



UNIVERSIDAD CÉSAR VALLEJO

FACULTAD DE INGENIERÍA Y ARQUITECTURA

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

**“Aplicación web progresiva para el proceso de facturación de la
empresa Continental Travel S.A.C.”**

**TESIS PARA OBTENER EL TÍTULO PROFESIONAL DE
INGENIERA DE SISTEMAS**

AUTOR:

Flores Príncipe, Orvic Yampool (ORCID: 0000-0001-5524-414X)

ASESOR:

DR. Pérez Farfán Iván Martin (ORCID: 0000-0001-5833-9400)

LÍNEA DE INVESTIGACIÓN:

Sistemas de Información y Comunicaciones

LIMA - PERÚ

2019

DEDICATORIA

A mis padres Richar y Carolina, a mis hermanas Dayeli y Estrella; por el apoyo incondicional y su ejemplo de esfuerzo, coraje y valentía. También a mis amigos y profesores los cuales me han acompañado en esta etapa universitaria.

AGRADECIMIENTO

A Dios, por darme la fuerza que necesitaba para seguir adelante.

A mi padre Richar y mi madre Carolina, por su apoyo, dedicación y consejos en el día a día, encaminándome de forma realista en la vida.

A cada uno de mis amigos y en especial a Sergio Moran, Jose Smith y Ronnie Alarcon, por abrirme las puertas de su empresa para poner en práctica los conocimientos adquiridos en el periodo de estudios universitarios.

AI DR. IVÁN MARTIN PÉREZ FARFÁN, por brindarme su asesoría permanente en el desarrollo y culminación de esta tesis.

PRESENTACIÓN

Señores miembros del Jurado:

Dando cumplimiento a las normas establecidas en el Reglamento de Grados y Títulos sección de Pregrado de la Universidad César Vallejo para la experiencia curricular de Metodología de la Investigación Científica, presento el trabajo de investigación pre-experimental denominado: “APLICACIÓN WEB PROGRESIVA PARA EL PROCESO DE FACTURACIÓN DE LA EMPRESA CONTINENTAL TRAVEL S.A.C.”.

La investigación, tiene como propósito fundamental: determinar cómo influye una Aplicación Web Progresiva para el Proceso de Facturación de la empresa Continental Travel S.A.C.”, en el año 2019.

La presente investigación está dividida en siete capítulos:

En el primer capítulo se expone el planteamiento del problema: incluye formulación del problema, los objetivos, la hipótesis, la justificación, los antecedentes y la fundamentación científica. En el segundo capítulo, que contiene el marco metodológico sobre la investigación en la que se desarrolla el trabajo de campo de la variable de estudio, diseño, población y muestra, las técnicas e instrumentos de recolección de datos y los métodos de análisis. En el tercer capítulo corresponde a la interpretación de los resultados. En el cuarto capítulo trata de la discusión del trabajo de estudio. En el quinto capítulo se construye las conclusiones, en el sexto capítulo las recomendaciones y finalmente en el séptimo capítulo están las referencias bibliográficas.

Señores miembros del jurado espero que esta investigación sea evaluada y merezca su aprobación.

ÍNDICE GENERAL

DEDICATORIA	ii
AGRADECIMIENTO	iii
PRESENTACIÓN.....	iv
ÍNDICE GENERAL.....	v
ÍNDICE DE FIGURAS	vii
ÍNDICE DE TABLAS	viii
RESUMEN.....	ix
ABSTRACT.....	x
I. INTRODUCCIÓN.....	11
1.1. Realidad Problemática	12
1.2. Trabajos previos	19
1.3. Teorías relacionadas al tema	26
1.4. Formulación del problema.....	53
1.5. Justificación del estudio	53
1.6. Hipótesis	56
1.7. Objetivo.....	56
II. MÉTODO.....	57
2.1. Diseño de investigación	58
2.2. Operacionalización de Variables	59
2.3. Población y muestra.....	71
2.4. Técnicas e instrumentos de recolección de datos, validez y confiabilidad	72
2.5. Métodos de análisis de datos	77
2.6. Aspectos éticos	81
III. RESULTADOS.....	82

3.1. Análisis Descriptivo	83
3.2. Análisis Inferencial	85
3.3. Prueba de Hipótesis	88
I. DISCUSIÓN	93
II. CONCLUSIONES	95
III. RECOMENDACIONES	97
REFERENCIA BIBLIOGRÁFICAS	99
ANEXOS	105

ÍNDICE DE FIGURAS

Figura 1. Grado de implantación de la FE en Europa a inicios del 2018	12
Figura 2. Factura electrónica en Europa	13
Figura 3. Evolución de la Cantidad de Facturas Electrónicas	14
Figura 4. Evolución de la cantidad de emisores electrónicos de FE	15
Figura 5. Comunicación en una Agencia de Viajes	16
Figura 6. Nivel de incidencias en emisiones.....	18
Figura 7. Margen de Utilidad Operativa.....	18
Figura 8. Proceso de Facturación	31
Figura 9. Comprobante Electrónico Digital emitidos desde los SEE	33
Figura 10. Arquitectura MVC.....	39
Figura 11. Principales Compañías propietarias de los GDS	44
Figura 12. PNR emitido como Reserva en Amadeus	46
Figura 13. Ciclo de vida de Extreme Programming	46
Figura 14. Ciclos, fases, iteraciones y flujos de trabajo de RUP	47
Figura 15. Ciclo de vida de Scrum	49
Figura 16. Proceso Scrum.....	50
Figura 17. Equipo Scrum	52
Figura 21. Nivel de Incidencias en Emisiones antes y después de implementado la Aplicación Web Progresiva	83
Figura 22. Margen de Utilidad Operativa generados antes y después de implementado la Aplicación Web Progresiva	84
Figura 23. Pre-Test: Nivel de Incidencias en Emisiones.....	85
Figura 24. Post-Test: Nivel de Incidencias en Emisiones	86
Figura 25. Pre-Test: Margen de Utilidad Operativa	87
Figura 26. Post-Test: Margen de Utilidad Operativa.....	87
Figura 27. Prueba T-Student – Nivel de Incidencias en Emisiones	90
Figura 28. Prueba T-Student – Margen de Utilidad Operativa	92

ÍNDICE DE TABLAS

Tabla 1. Validación de expertos	48
Tabla 2. Operacionalización de las variables	61
Tabla 3. Indicadores.....	62
Tabla 4: Determinación de la Población	71
Tabla 5. Juicio de Expertos	74
Tabla 6. Rango de Validez de Instrumentos.....	74
Tabla 7. Coeficiente de Correlación de Pearson	75
Tabla 8. Resultados SPSS – NIE	76
Tabla 9. Resultados SPSS – MUO.....	76
Tabla 10. Estadísticos descriptivos de Nivel de Incidencias en Emisiones	83
Tabla 11. Estadísticos descriptivos de Margen de Utilidad Operativo	84
Tabla 12. Pruebas de Normalidad del Nivel de Incidencias en Emisiones.....	85
Tabla 13. Pruebas de Normalidad del Margen de Utilidad Operativa	86
Tabla 14. Pruebas de T-Student del Nivel de Incidencias en Emisiones	89
Tabla 15. Pruebas de T-Student del Margen de Utilidad Operativa.....	91

RESUMEN

La presente tesis detalla el desarrollo de una Aplicación Web Progresiva para el Proceso de Facturación de la empresa Continental Travel S.A.C., la situación previa a la aplicación del sistema presentaba problemas en cuanto al proceso de facturación de comprobantes de pago. El objetivo de esta investigación fue determinar la influencia de una Aplicación Web Progresiva para el Proceso de Facturación de la empresa Continental Travel S.A.C.

Por lo tanto, se describe previamente todos los aspectos teóricos sobre el proceso de facturación; además de las metodologías que se utilizaron para el desarrollo del Sistema Web. Para lo cual, se utilizó como metodología a SCRUM, siendo esta la metodología que más satisfacía las exigencias del proyecto; además, se utilizó el lenguaje de programación C# con el framework Nettiers, el gestor de herramientas de Component One y el gestor de base de datos SQL Server 2012 R2.

El tipo de investigación fue aplicada, mientras el diseño de la investigación fue pre-experimental y el enfoque es cuantitativo. La técnica de recolección de datos fue el fichaje y el instrumento fue la ficha de registro, los cuales fueron validados por expertos.

La población para calcular los indicadores propuestos fue 151 comprobantes de pago con un tamaño de muestra de 108 comprobantes. El tipo de muestreo fue aleatorio probabilístico simple. La técnica de recolección de datos fue el fichaje y el instrumento fue la ficha de registro, los cuales fueron validados por expertos.

La implementación de la Aplicación Web Progresiva permitió disminuir el Nivel de Incidencias en emisiones del proceso de facturación del 33,20% al 12,27%, de la misma manera, se incrementó el Margen de Utilidad Operativa del 11,57% al 20,28%. Los resultados mencionados anteriormente, permitieron llegar al análisis que el Aplicación Web Progresiva permitió logro mejorar el proceso de facturación en el Área Contable.

Palabras clave: APLICACIÓN WEB PROGRESIVA, FACTURACIÓN, SCRUM.

ABSTRACT

The present thesis highlights the development of a Progressive Web Application for the Billing Process of the company Continental Travel S.A.C., the situation prior to the application of the system presented problems regarding the billing process of proof of payment. The objective of this research was to determine the influence of a Progressive Web Application for the Billing Process of the company Continental Travel S.A.C.

Therefore, all the theoretical aspects of the billing process are described in advance; in addition to the methodologies that were used for the development of the Web System. For this, SCRUM was used as a methodology, this being the methodology that most met the requirements of the project; In addition, the C-programming language was used with the Nettiers framework, the Component One tool manager, and the SQL Server 2012 R2 database manager.

The type of research was applied, while the research design was pre-experimental and the approach is quantitative. The data collection technique was the signing and the instrument was the registration sheet, which was validated by experts.

The population to calculate the proposed indicators was 151 proof of payment with a sample size of 108 receipts. The type of sampling was random probabilistic simple. The data collection technique was the signing and the instrument was the registration sheet, which was validated by experts.

The implementation of the Progressive Web Application allowed to reduce the level of emissions incidences of the billing process from 33,20% to 12,27%, in the same way, the Operating Profit Margin was increased from 11,57% to 20,28%. The results mentioned above, allowed us to arrive at the analysis that the Progressive Web Application allowed to improve the billing process in the Accounting Area.

Keywords: PROGRESSIVE WEB APPLICATION, BILLING, SCRUM.

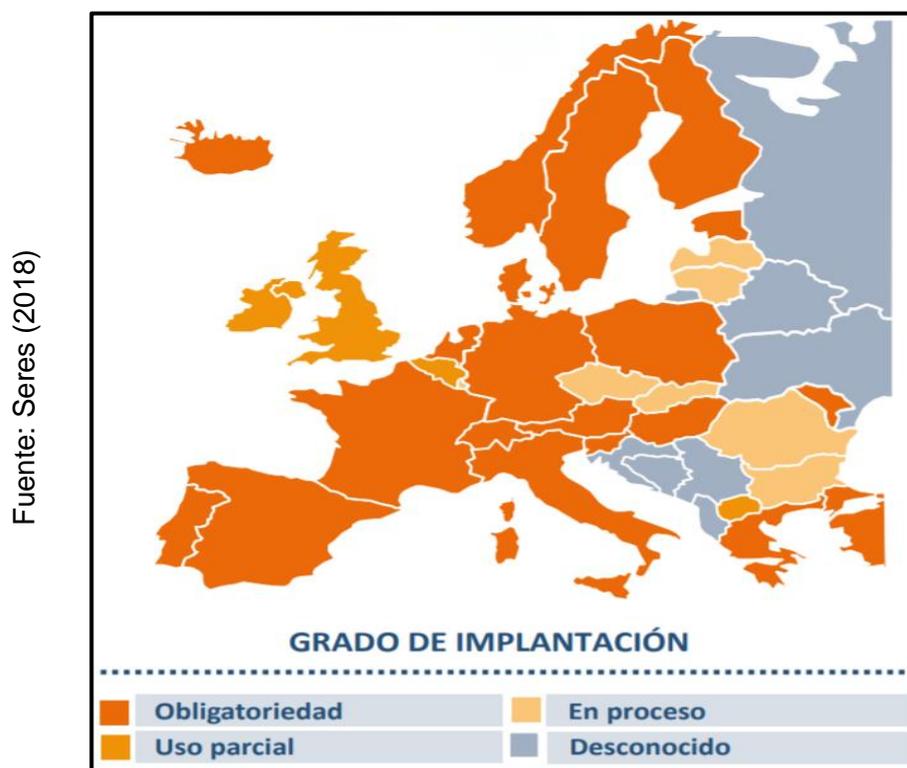
I. INTRODUCCIÓN

1.1. Realidad Problemática

La facturación implica una serie de venta o compras de un producto o servicio; en contexto internacional, Seres afirma lo siguiente:

La factura electrónica es uno de los elementos esenciales en la transformación digital de los países. Es impulsado en un movimiento internacional por la masificación de su uso por la Comisión Europea ya desde hace tiempo y tiene más fuerza que cada vez. Actualmente, se encuentra un proyecto de las Administraciones Públicas de la Unión Europea que normaliza el modelo de facturación electrónica. Estas documentaciones recopilan los avances y las situaciones que son esperados en Europa la facturación electrónica. (2018, p.2)

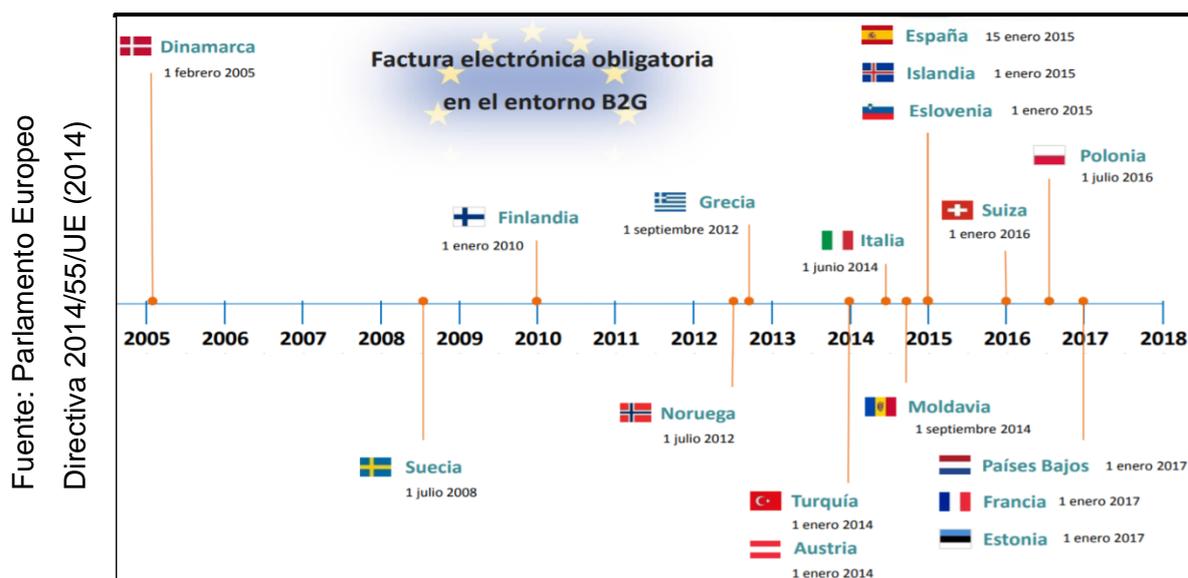
Figura 1. Grado de implantación de la FE en Europa a inicios del 2018



En los últimos años por toda la Unión Europea se masificó la facturación electrónica. Asimismo, Seres afirma lo siguiente:

En abril de 2014 fue aprobado el Parlamento Europeo la Directiva 2014/55/UE unificando los formatos de las facturaciones electrónicas (...) El avance desde ese entonces en la facturación electrónica en abril de 2019 a dado lugar a que los Estados miembros, implementen la factura electrónica de manera obligatoria entre sus proveedores y el sector público, es decir, en la relación B2G. (2018, p.3)

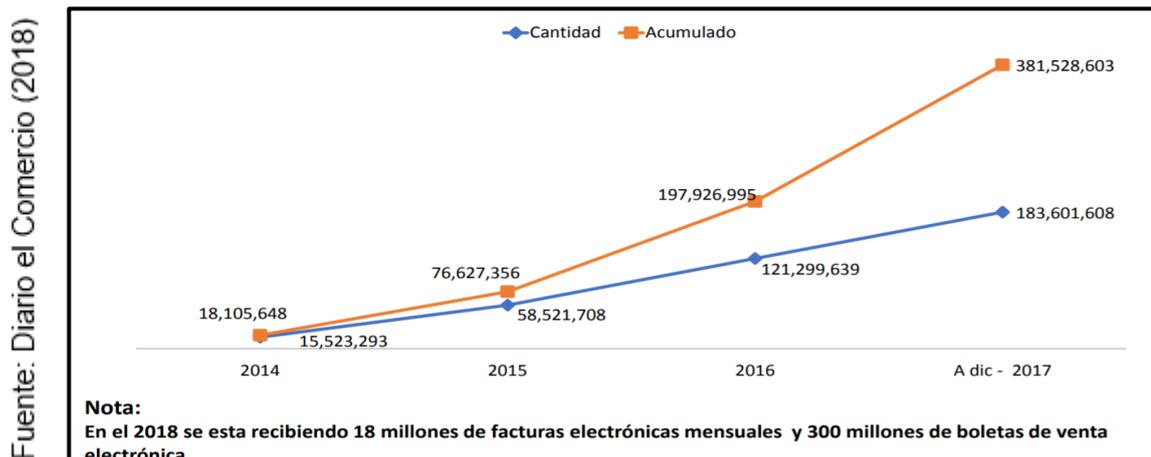
Figura 2. Factura electrónica en Europa



Sin embargo, a nivel nacional, en una nota periodística del Diario El Comercio, menciona lo siguiente:

En el país, se ha implementado más de 100.000 la facturación electrónica, según la SUNAT. En América Latina se ha posicionado entre los países con un mayor avance en la materia. Por lo cual, antes no se sabía que dentro de los países en el mundo el Perú en mayor medida fue implementando el uso de las facturas electrónicas. En el plano global, lideran su implementación ubicándose dentro de los diez países (2018, parr.1)

Figura 3. Escala de Facturas Electrónicas en el Perú desde 2014



Además, la SUNAT como ente regulador de este proceso ha publicado para todos los contribuyentes una serie de manuales y guías, entre otros; con el fin de facilitar la declaración de sus sustentos económicos como empresa. Asimismo, el Ingeniero Carlos Drago Llanos, jefe del Programa de Transformación Digital de SUNAT en una entrevista para el Diario El Comercio, menciona al respecto que:

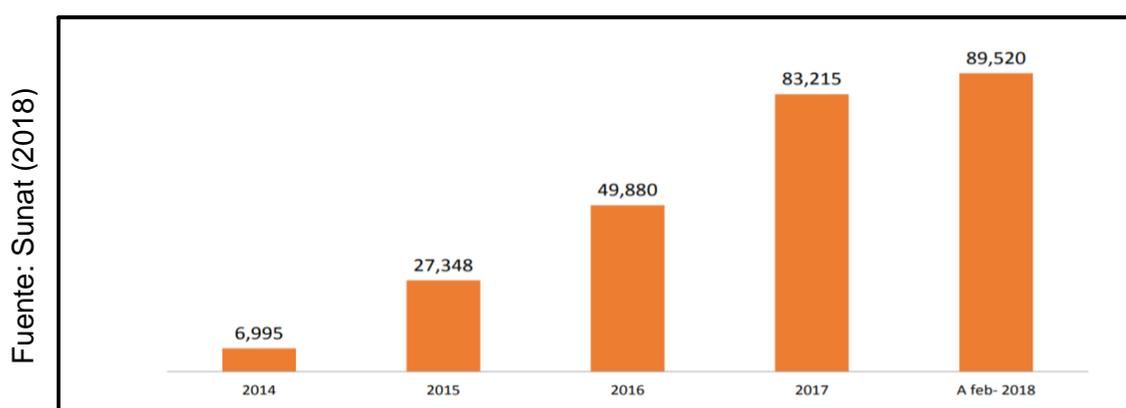
En el Perú, hace cuatro años se inició el proyecto y ha sido favorable por el uso de este instrumento: a finales del 2017, ya ascendía a 100.000 el número de empresas, que emiten por voluntad propia un alrededor del 85%. “Se ha evidenciado un avance en las facturas electrónicas en América Latina. México, por ejemplo, en un lapso de 4 a 5 años lo ha masificado al 100%. Chile comenzó en el 2003 y en febrero de este año terminó. Ya se ubica en esa línea el Perú como uno de los cinco países de la región más avanzados (2018, parr.1).

Lo cual, en el 2019, la SUNAT incorpore gradualmente a empresas pequeñas y medianas en el Perú, por ello el Ingeniero Carlos Drago Llanos en el Diario El Comercio para una entrevista afirma lo siguiente:

El objetivo es que entre el 2019 y el 2020 la mayoría de empresas estén emitiendo comprobantes de pago electrónicos. Lo cual vendría a ser viable y posicionando, en un corto tiempo, a la par

de Brasil, Chile y México. El modelo de México se viene adoptando por la SUNAT, continuando con la tendencia de rápida de masificación. Así, para este año en agosto, será incluido en la SUNAT como contribuyente, operando servicios electrónicos certificados. Sobre especialistas en las facturas electrónicas que validarán y recibirán el comprobante de pago electrónico, además de proveer otros servicios relacionados a la facturación (2018, parr.2)

Figura 4. Evolución de la cantidad de emisores electrónicos de FE



Lo cual se menciona que, se emplea una serie de facturas tradicionales por algunas empresas; durante el transcurso de los años se fue transformando en un proceso de mejoras que actualmente se denomina “Facturación Electrónica”. Asimismo, Tsinghua University afirma que:

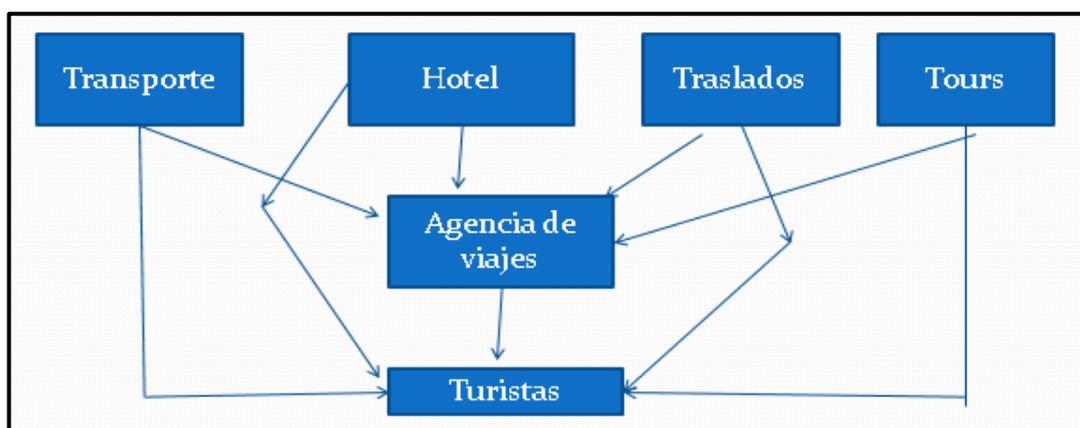
Las facturas de papel tradicionales su costo va en un rápido aumento y se vuelve ineficaz, con una gestión compleja, sin poder satisfacer las características del comercio electrónico. En las facturas tradicionales en el periodo del proceso de emisión, a veces por iniciativa propia el usuario no solicita la factura, esto ocasiona que no se emita por el beneficiario la boleta o facturación, ocasionando una masa de pérdida fiscal. El costo es muy alto a causa de la recaudación de la inspección y los impuestos, aparecen problemas en algunas facturas por la gran carga que se añade, esto hace

que los departamentos fiscales comprueben todos los libros y facturas de contabilidad, por lo tanto, un gran desafío al que se enfrentan basados en las facturas en papel por el sistema de gestión fiscal. Además, estas facturas gastan muchos recursos cada año, reduce las emisiones de carbono y incumpliendo los requisitos de conservación de energía. (2014, p.24)

La empresa Continental Travel S.A.C. está ubicado en la provincia de Lima, distrito de San Isidro; es una empresa privada, perteneciente al rubro de Turismo y Hotelería, con el propósito de vender y organizar a la clientela los distintos servicios turísticos con ciertas condiciones a favor y con conveniente precio.

Fuente: Romero y Vallespín (2014)

Figura 5. Comunicación en una Agencia de Viajes



Dada la entrevista realizada al Jefe de Sistema de la empresa Continental Travel S.A.C. (Anexo N°3) se pudo examinar que el principal problema era en los comprobantes de pago que contaban con una mala gestión; asimismo, esto estaba relacionado a su proceso de ventas actual. Sin embargo, durante la entrevista se destacan problemas como:

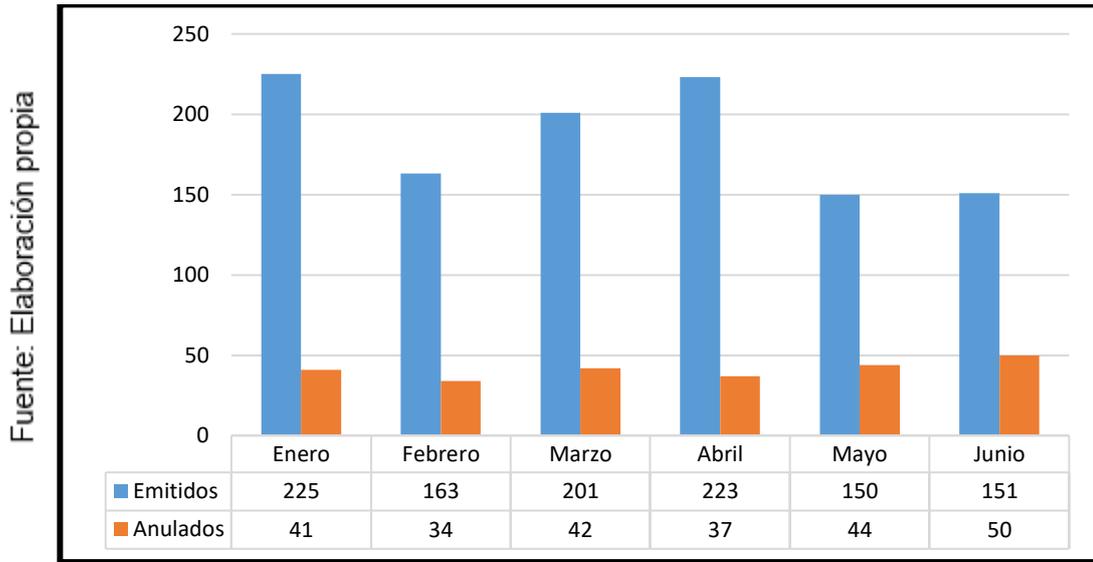
- Ineficiente salida de dinero y control en las entradas por motivos fraudulentos.
- La emisión de comprobante de pago tanto físicos como electrónico; facturas, boletas, reportes personalizados entre otros generaba costos, tiempo y mano de obra, donde cada departamento solicitaba para realizar el proceso respectivo

- Emisiones tardías de los documentos contables para entidades del estado o proveedores, clientes, el cual generaba daño a la reputación y multas económicas de la empresa.
- Falsificación o alteración de documentos contables, en cual tenía información irrelevante como la existencia del personal o empresas fantasmas en donde se emitían cobros o pagos del servicio.
- Los registros en papel físico eran expuestos a dañarse o perderse.

La fase de factura es iniciada cuando el facturador personal administra las guías de servicio y recepción de tickets emitidos por el Counter (Agente de Viajes) registrándose en el BackOffice, con la finalidad de emitir los comprobantes de pagos al usuario; lo cual es verificado por el personal antes que los datos estén relacionados y sean verídicos con el ticket emitido por el GDS, tomando aproximadamente 5 minutos. Si se presenta algún error, el personal de venta que lo emitió es contactado para corregir los datos en el sistema, si por algún motivo la modificación está limitada, el área de Sistemas debe de resolverlo; a causa de este problema tiende a demorar su emisión del comprobante de pago, después debe ser enviado al cliente con el Courier. Sin embargo, existe clientes que se les proporciona comprobantes electrónicos donde será registrado a través del Facturador, con tiempo adicional que se toma entre 3 a 10 minutos, según lo emitido por la glosa.

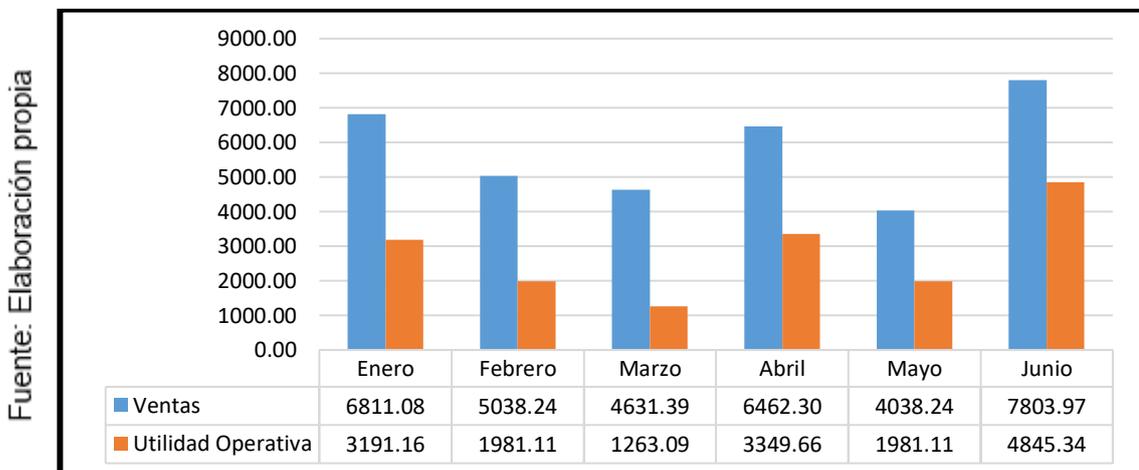
Uno de los problemas álgidos que requiere de atención inmediata es el nivel de incidencias en emisiones, que está 11% a causa de que no hay una adecuada comunicación entre los facturadores y los Counters, un proceso de gestión irregular y para la atención especializada esta la falta de capacitación al personal, observándose en la Figura 6.

Figura 6. Nivel de incidencias en emisiones



Así como también, otro de los problemas prioritarios se asocia a los gastos operacionales, estos montos incluyen gastos de representación, administrativos y financieros, como impresiones, eventos, licencias, incentivos, alquiler de sucursales, servicios de terceros, viajes de negocio, mantenimientos de equipos, muebles y oficinas entre otros. Por lo que se observa un promedio de 6.86% de gastos no planificados; como se evidencia en a Figura 7.

Figura 7. Margen de Utilidad Operativa



Por ello, la persistencia de estos problemas conlleva en el comprobante de pago tanto electrónicos como físico tiendan a presentar errores en la digitación y no se desarrollen a tiempo. Por lo tanto, surge la siguiente pregunta: ¿Qué sucederá si se sigue teniendo los mismos problemas en la empresa Continental Travel S.A.C.?, en respuesta a dicha pregunta, si se sigue realizando una inadecuada gestión de los comprobantes de pago, la compañía no tendrá un crecimiento estable a futuro, los clientes seguirán insatisfechos con la atención que se está recibiendo, esto ocasiona una imagen negativa para la compañía.

1.2. Trabajos previos

Para el desarrollo del proyecto de investigación se ha revisado diversas fuentes, proporcionando una base teórica que sustenta la investigación planteada.

Nacionales

- En el año 2018, Bustamante Ysela y Pacheco Flor; realizaron la investigación titulada “Incidencias: Ventajas y Desventajas de la Implementación del Sistema de Emisión Electrónica frente al Sistema de Emisión Física en principales contribuyentes en la ciudad de Arequipa 2018”. La investigación tiene como objetivo identificar las ventajas y desventajas del Sistema de Emisión Electrónica frente al Sistema tradicional en los Principales Contribuyentes de la ciudad de Arequipa. El tipo de estudio es Aplicada-Experimental y su diseño elegido fue Pre-Experimental. La muestra fue de 100 contribuyentes; por lo que en los resultados se obtuvo algunas ventajas como los beneficios en la declaración de sus comprobantes por estar al día, disminución de la evasión tributaria, ahorro de impresiones y simplificar sus procesos. En cuanto a las desventajas se encontró problemas en la plataforma del sistema de Facturación SUNAT, esto genera que se perciban a las normas de SUNAT como complicadas y cambiantes, se generó un

presupuesto adicional adquirido de un tercero para la implementación del sistema para los contribuyentes. De acuerdo al resultado durante la investigación demuestra el desarrollo de una Aplicación Web a medida propuesto por el autor es económicamente factible, lo cual permita mejorar el control fiscal.

Del presente estudio se tomará como referencia las teorías relacionadas, ya que diferencia de lo que es una facturación física y facturación electrónica; el impacto que conlleva a migrar a la Facturación Electrónica.

- En el año 2017, Navarro Theany; realizó un estudio titulado “Sistema de facturación electrónica para la gestión de comprobantes de pago basado en ISO/IEC19845:2015 en Acgenesys S.A.C.”. El autor determinó que la empresa presenta algunos gastos e incidencias no definidos en el comprobante de pago. Como objetivo principal propuesto fue determinar los efectos de la implementación de los sistemas de facturación gestionando los comprobantes de pago. Es de tipo experimental - aplicado y el diseño de tipo pre-experimental. La población fue por la cantidad de comprobante de pago emitido en 30 días, y como muestra la población es la misma cantidad. Como resultados se evidenció que las incidencias en emisiones antes fue 4.5491 y luego de la implementación fue 0.9111, una reducción de 79.97%. Además, para el indicador porcentaje de gastos operacionales, antes se tenía un valor de 0.0278 y luego de la implementación este valor se redujo a 0.0171, con una reducción porcentual de 61.551%. Por último, tuvo un efecto positivo al implementar un sistema de facturación electrónica en la gestión de comprobante de pago.

El antecedente del presente estudio resalta la relación con la problemática dentro del proceso de facturación en la empresa. El constante avance de la tecnología, como los dispositivos denominados

portátiles o móviles, lo cual nos permite de manera más rápida tener un mayor acceso a la información y eficaz para la tomar decisiones.

- En el año 2017, Araujo Andrés; realizo la investigación titulada “Sistema Web para el proceso comercial en la empresa Equisoft Perú S.A.C. El autor determino que la empresa presenta deficiencias en los comprobantes de pago emitidos. El objetivo de esta investigación fue determinar de qué manera un sistema web influye un en el proceso comercial en la Empresa Equisoft. Se empleó la metodología SCRUM, ya que en el transcurso del desarrollo del sistema los cambios son más flexibles, como lo plantea el autor. El estudio es de tipo aplicada y su diseño fue elegido Pre-Experimental. Se optimizo el proceso comercial por la implementación del sistema web, el índice de efectividad de cotizaciones era de 24,7% y aumento en 37,7 %; y con el índice de comercialidad era de un 1.48% y fue aumentando en un 3.35%. Finalmente, influyó de manera positiva el sistema web para el proceso comercial en la compañía Equisoft Perú S.A.C; aumentando el índice de comercialidad y el índice de efectividad de cotizaciones, en la satisfacción del cliente final y usuario.

De este estudio, como aporte se tomó el uso de la Metodología SCRUM para la construcción y especificación de requisitos del producto software a desarrollar, dirigido a un entorno empresarial.

- En el año 2015, Ordaya Rita; realizo la investigación titulada “Implementación de un Sistema de Información para una MYPE Comercial con componentes de Libros y Facturación electrónica”, de la Universidad Católica del Perú. Se observó en el estudio que, en el registro de transacción existe un deficiente proceso. Además, para realizar manualmente los registros para los usuarios les resulta complicado. Actualmente las tecnologías actuales mediante herramientas que nos brindan, se permitió sistematizar estos con el objetivo de facilitar las tareas, agilizando y reduciendo los errores. El

autor recalca que, en la Resolución de Superintendencia N° 374-2013/SUNAT hace referencia a los documentos de facturación, en la cual debe de ser generado digitalmente en un formato. Debido a ello, se propone el desarrollo de un sistema de información para la gestión del proceso de facturación, gestionando sus recursos, automatizando su proceso y cumplir con la normativa vigente.

El estudio es de tipo Aplicado-Experimental y su diseño Pre-Experimental. El tamaño poblacional es conformado por los comprobantes de pago emitidos durante el primer trimestre del 2015, y en la muestra la población es la misma cantidad. Como resultados se obtuvo el nivel de efectividad antes fue 74.31% y luego de la implementación fue 90.91%, lo que significa un aumento de 16.61%. De acuerdo a los resultados se alcanzó un desarrollo ordenado de la solución, por lo que fue posible tener claros sus funcionalidades y los procesos.

Del presente estudio se incluirá como referencia para la problemática y las teorías relacionadas, ya que define la situación actual de proceso de ventas relacionado al proceso de facturación, causando un gran impacto en el procesamiento de datos relevantes para una empresa.

Internacionales

- En el año 2018, Le, Ha Giang; realizo la investigación titulada “Adoption of E-invoicing In Vietnam Case: Electricity of Vietnam Group, Danang Power Company”, de la Lahti University of Applied Sciences de Vietnam. El autor resalto que hoy en día, la facturación es una parte crítica del funcionamiento de la compañía. Las creaciones de las nuevas tecnologías implican la facturación electrónica junto a la tradicional en papel, también ha sido de enfoque de organizaciones y gobiernos en todo el mundo. Sin embargo, en Vietnam, las empresas siguen siendo escépticas a implementar la factura electrónica. El objetivo de la investigación es obtener información sobre la factura electrónica y cómo

la facturación electrónica puede ser beneficioso para una empresa. Además, el autor pretende identificar los obstáculos que Danang Power Company, como aborda la facturación electrónica. La investigación fue descriptiva y con un diseño experimental; además, utilizo un enfoque deductivo, mientras que la parte teórica del estudio presenta todas las teorías relevantes. Incluye entrevistas con el personal de la compañía y una encuesta online realizada entre sus clientes. En el resultado de la investigación, la facturación electrónica es capaz de mejorar en un 90% la eficiencia de la compañía e impulsar a sus clientes hacia un nuevo sistema de facturación electrónica.

De la presente investigación sirvió de aporte para profundizar más en el análisis de la problemática, así como también al marco teórico que aborda la facturación electrónica.

- En el año 2017, Wenjing Yang; realizo la investigación “Analysis on Online Payment Systems of E-Commerce”, de Finlandia. El autor menciona que para el proceso del uso de la tecnología la venta y compra, se volvió crucial este proceso para el desarrollo a largo plazo de una empresa. El objetivo es analizar el estado actual, los desafíos y las expectativas futuras de Sistemas de pago en línea en China y Europa y especialmente en Finlandia. El tipo de estudio fue Cualitativa y contuvo un diseño longitudinal. Dentro de los resultados se muestra que actualmente son populares los sistemas de pagos en línea. La mayoría de encuestados tienen experiencia en pagos en línea. Net Bank y Tarjeta de débito (Visa o MasterCard) son los más populares, no solo en Europa sino también en China y demás países. El segundo son los sistemas de pagos en línea por terceros; PayPal en Europa es más popular. En China, por el contrario, AliPay es el sistema principal de pagos online. Los dos aspectos principalmente que contribuyen a la elección del sistema de pagos online son la rapidez de transacción y la conveniencia. La vulnerabilidad y los problemas técnicos a delitos cibernéticos son principalmente ventajas en línea; mientras que los problemas financieros

y los ataques de malware principalmente son desafíos de pagos online. Lo cual para un ambiente confiable y seguro de pago en línea es importante.

Del presente estudio sirvió como aporte para escatimar más en el desarrollo del software ofreciendo un ambiente seguro y amigable hacia el cliente final.

- En el año 2015, Benjamin Pessi; realizó la investigación titulada “The Impact of Implementation of the Electronic Purchase Invoice System on a Company on the Exmple of Hahle Group”, de Estonia. La investigación destacó la importancia de reducir los gastos para la competitiva de una empresa tanto como sea posible y acelerándolo; aumentando las operaciones de los mercados empresariales para alcanzar el máximo beneficio. En una administración financiera electrónica la transición no es sólo una cuestión de utilizar los canales electrónicos y de la automatización contable; además de la reducción de costos y la racionalización de las operaciones, esto a la transición de compras electrónicas. Los sistemas de facturación permiten flexibilidad en la organización de las operaciones de toda la empresa, una actualizada gestión. El objetivo es identificar los principales defectos que un Sistema de Facturación Electrónica puede lograr para una empresa. El tipo de investigación fue descriptiva y contuvo un diseño experimental; además, como tamaño poblacional conformado por las emisiones de los comprobantes de pago durante el año 2015. En los resultados el cambio que se mostró a un sistema de facturación electrónica fue una razonable decisión y saliendo bien desde principio a fin, sin problemas mayores. Permitted informar de manera oportuna, la eliminación del trabajo duplicado, conllevando la rapidez del proceso. Asimismo, la necesidad de espacio de almacenamiento es eliminado, y sea disponible el material independientemente del tiempo y la ubicación. Como resultado, tiene menos desventajas y beneficios mayores en la empresa que de un proceso de facturación física.

Del presente estudio sirvió de aporte para profundizar más en el análisis de la problemática, así como también al desarrollo de la solución propuesta el cual permite que el proceso sea mucho más rápido.

- En el año 2015, Howard Tugume, Justine Kobusinge y Justine Nanteza; realizó la investigación titulada “The Physical and Electronic Payment Interface and its Influence on Consumer Payment Choices and Informal/Fraudulent Practices: A Case Study of the National Water and Sewerage Corporation (Nwsc) Uganda”, de Estados Unidos. El autor resalto el rápido crecimiento en el uso del dinero móvil como método de pago y su aceptación por parte de corporaciones como la Corporación Nacional de Agua y Alcantarillado (NWSC). Este estudio utilizó un modelo LOGIT para analizar los factores de elección del consumidor al usar el dinero móvil como método de pago al NWSC. El estudio es de tipo Aplicada y contuvo un diseño transversal; la cual conto con un tamaño poblacional conformado 238 clientes nacionales de NWSC en los distritos de Kampala, Luwero y Mukono. Como resultado del estudio, el nivel de ingresos y la ubicación podrían aumentar la probabilidad de que un consumidor use dinero móvil para pagar facturas electrónicas. Los clientes de NWSC han adoptado métodos de pago modernos, especialmente dinero móvil y aún no han apreciado las innovaciones bancarias como la banca por Internet y otras transferencias directas. Los atributos de las innovaciones ayudan claramente a explicar por qué los clientes de NWSC utilizaron un determinado método de pago, y sus características podrían ayudar a predecir la probabilidad de que un consumidor utilice dinero móvil para pagar las facturas electrónicas.

De la presente investigación sirvió de aporte para un análisis externo de la facturación electrónica en relación al problema planteado en este proyecto de investigación, por lo que incluir medios de pago aceleraría los procesos de pago.

1.3. Teorías relacionadas al tema

Variable Dependiente: Proceso de Facturación

Comprobante de Pago

Para definir un Comprobantes de Pago, el Servicio de Ayuda a Domicilio (SAD) sostiene al respecto:

Son documentos en las cuales se refleja la información detallada sobre un producto en específico o servicio contratado y su costo es durante un periodo concreto. Estas se pueden clasificar en tres tipos: ordinarias, rectificativas o correctivas y recapitulativas o de resumen abreviado.” (2015, p.81)

IICA (2016) afirma que: “Todas las operaciones de contabilidad deben estar basado es comprobantes, internos o externos, o ambos en algunos casos; estos deben evidenciar los servicios o bienes adquiridos o vendidos, han sido recibidos a satisfacción.” (p.33)

Asimismo, la Superintendencia Nacional de Aduanas y de Administración Tributaria afirma que:

Son documentos que acreditan la prestación de servicios o la entrega en uso y transferencia de bienes. Esto deberá emitirse y/o ser impreso según este conformado las normas del Reglamento de Comprobantes de Pago (Resolución de Superintendencia N° 007-99-SUNAT).” (2019, parr.2)

Además, SUNAT (2019) afirma que: “Existen cuatro comprobantes de pago contables para la SUNAT; Nota de Crédito, Boleta, Factura, Nota de Débito” (parr.5)

Proceso de Facturación

Lo que se ofrece en toda empresa ya sea un servicio o bien, es obligatorio declarar sus sustentos económicos mediante los comprobantes de pago, es

denominado, como un proceso crucial para toda empresa. Asimismo, Catacora (2015) afirma que: “Tiene como objetivo procesar y registrar en relación a todas las operaciones o actividades las cuales como objetivo es aumentar y mantener las ventas” (p.282).

Leuro y Oviedo (2016) lo define como: “conjunto de actividades que conlleva a la atención al usuario que autoriza la liquidación a la prestación de servicios. Es cuando un usuario solicita su pago se activa este proceso, su atención y un servicio en la institución” (p.85).

Además, Leuro y Oviedo (2016) afirma que: “El proceso de facturación en el pedido es una serie de condiciones fijadas, está sujeta por el estado como normativa y permitirá el cumplimiento de la factura que justificará la operación” (p. 24).

Etapas del proceso de facturación

Contratación de Servicio: Para Leuro y Oviedo, sostienen al respecto que:

El CONTRATO es un compromiso, convenio o pacto, acuerdo que es consecuente del proceso entre dos partes de negociación (una o más personas), por cosa determinada o sobre materia, obligándose mutuamente y su cumplimiento este recíprocamente obligados. (2016, p. 34)

Admisión del Usuario: “Al ingresar es iniciado la atención con la recepción del servicio, siendo responsable de la clasificación, verificación de derechos e identificación, para que se inicie el proceso financiero como paso primordial” (Leuro y Oviedo, 2016, p.34).

Prestación del servicio y elaboración de Registros Asistenciales y/o Administrativos: Para Leuro y Oviedo, del presente proceso afirma que:

Los auxiliares o tecnólogos profesionales, técnicos que forman parte de la prestación de los servicios son los encargados de realizar el cargo de datos mediante los procesos de

ordenamiento o solicitud, administración, registro de la atención o el uso de insumos. Este paso es de importancia, ya que, si la información no es diferenciada correctamente y oportuna para cada uno de los responsables, la facturación puede ser errónea o deficiente. (2016, p.35)

Liquidación y elaboración de la factura: Para Leuro y Oviedo, sostiene al respecto que liquidar y elaborar la factura:

Para realizar el cargo diario de los insumos, prestaciones, etc. Se debe de elaborar la pre-factura de acuerdo a la norma o al convenio establecido, con una base tarifaria en la Ley o en el contrato. El facturador debe tener presente el manual tarifario convenido para la realización de los análisis de los datos contenidos, la normatividad legal y entre las partes los acuerdos de voluntad vigente. Posteriormente se procede a la emisión de las facturas definitivas o liquidar individualmente. (2016, p. 35)

Impresión y/o recolección de documentos soporte: “Es el soporte de la factura que incluyen los registros administrativos, la autorización de la prestación de servicios y consumos.” (Leuro y Oviedo, 2016, p. 35).

Preparación de la cuenta: Para Leuro y Oviedo, durante el proceso para preparar la cuenta, afirma que:

Durante este paso, la facturación es recibida por medio de un determinado período, la auditoría previa debe realizarse con el fin de corregir y los errores ser detectados. Verificada su calidad se aprueba y cada cuenta correspondiente se arma a cada entidad discriminada por periodo y responsable del pago; posteriormente estas cuentas se envían al área responsable (2016, p. 36).

Radicación de la cuenta: Asimismo, Leuro y Oviedo define el proceso propuesto como:

El paso final de los procesos de presentación y preparación. Por medio de este proceso, se informa a la entidad o empresa

responsable del pago. Es necesario resaltar que, al ser un procedimiento simple, no es limitado la entrega de documentos a una empresa, al contrario: Se debe asegurar la recepción de los documentos. Dejar constancia escrita de la aceptación y el recibo de la cuenta de cobro. En ella se obtiene las constancias de las fechas de entrega. (2016, p. 36)

Gestión de la cartera: “La cuenta entregada, a la aseguradora como objeto de cobro, el seguimiento es necesario las cuales incluyen: glosas, el manejo de objeciones; aplicación y control del indicador de gestión de la cartera” (Leuro y Oviedo,2016, p. 36).

Recaudo: Para Leuro y Oviedo, sostiene al respecto que este proceso:

La recaudación del ingreso o los pagos de los montos cobrados por venta de servicios, cumpliendo continuar con su objetivo. Es importante conocer la cancelación de las cuentas y las fechas en que son realizados cada pago para determinar en los plazos establecidos si existe o no cumplimiento (2016, p. 36).

Comprobante de Pago Electrónico

SUNAT (2019) afirma que. “Es través del sistema de emisión electrónica, para los efectos del Impuesto a la Renta esto permite sustentar el costo-gasto, así como el uso del crédito fiscal del Impuesto General a las Ventas.” (parr.1)

Para el Servicio de Apoyo Domiciliario (SAD), sostiene al respecto que un Comprobante de Pago Electrónico - CPE:

Es aquel documento electrónico que, al igual que el comprobante de pago en formato papel, garantizan la autenticidad de origen (empresa, organismo, etc.) y la integridad del contenido. Los comprobantes de pago electrónicos poseen las mismas equivalencias en cuanto a justificante de entrega de una provisión o producto de servicios entre un emisor y un receptor que los

comprobantes de pago en papel, pero para validarse necesitan de certificados, firmas y/o sellos digitales (2015, p.81).

Asimismo, la Superintendencia Nacional de Aduanas y de Administración Tributaria (SUNAT) afirma que un Comprobante de Pago Electrónico:

Se considera otorgada en el momento de su emisión. La emisión de la factura electrónica es regulada en el artículo 5° de la Resolución de Superintendencia N° 007-1999/SUNAT. En el caso que la transferencia sea por telefax, teléfono, Internet u similares medios, efectuándose el pago mediante las tarjetas de débito o crédito y/o abono en cuenta, se emitirá la factura electrónica según las fechas que es recibida la conformidad de las operaciones del medio de pago por parte del administrador o si se percibe los ingresos, de acuerdo a la situación (2019, parr.5).

Operatividad

Un comprobante de pago electrónico sigue en aplicación a las normas sujetas a la SUNAT; el cual el CPE emitido deberá estar estructurado de acuerdo a Norma Internacional de la UBL, que conjuntamente con la SUNAT, se emplea la versión 2.1. Por ello, la Superintendencia Nacional de Aduanas y de Administración Tributaria – SUNAT (2019, parr.2) sostiene lo siguiente:

- La factura es emitida, de acuerdo al formato electrónico establecido por la SUNAT.
- El emisor entrega y/o envía las facturas electrónicas a través de una página web a su clientela (receptores) en un formato electrónico, servicio web, correo electrónico, y otros medios. El emisor define el medio de entrega.
- A la SUNAT un ejemplar con la fecha de emisión consignada debe enviarse en la factura electrónica o vinculada en la nota electrónica.
- El emisor tiene un periodo de tiempo de 7 días que es contado en el primer día que se emitió para el envío, según lo establecido en el anexo 6 de la

Resolución de Superintendencia N° 097-2012/SUNAT. Ref: Art. 12° de la Resolución de Superintendencia N° 097-2012/2012, sustituido por el Art. 1° de la Resolución de Superintendencia N° 374-2013.

- La información enviada es validada por la SUNAT, el emisor envía una Constancia de Recepción– CDR en un medio similar en que envió los comprobantes de pago electrónicos, generando los estados siguientes Aceptada, Rechazada y Aceptada con observación.
- A disposición de su clientela (receptores) el emisor debe poner una opción de consulta emitidos en los comprobantes (boletas de venta, facturas y notas de débito y de crédito), por medio de una página web, en un tiempo no menor a un año. Un mecanismo de seguridad debe ser definido para acceder a esa consulta, permitiendo resguardar de la información su confidencialidad, de tal modo que, para acceder a ella, solo el cliente pueda hacerlo.
- Adicionalmente, la SUNAT para los comprobantes electrónicos emitidos a disposición del contribuyente, una opción de consulta. Por medio de ello, se observará la información tributaria del comprobante.

Figura 8. Proceso de Facturación

Fuente: Sunat (2019)



SUNAT nos señala que un comprobante de pago electrónico emitida deberá cumplir ciertos criterios, el cual nos afirma que:

Los comprobantes de pago emitidos, el formato obligado a usar debe ser respetado. (...) Luego de ello, el documento emitido se procederá a enviar una copia a SUNAT a partir de la fecha emitida en un plazo no mayor de 7 días calendario. Luego de que la información es validada por SUNAT, se enviará una constancia de recepción – CDR, se evaluará de acuerdo a tres pautas: Aceptación, que implica total validez tributaria. Aceptada con observación, indicando poseer validez tributaria, pero en una auditoría hay datos que podrían ser restaurados. Rechazada que indica una invalidez tributaria, necesitando realizar una nueva factura electrónica (2019, parr.4).

Asimismo, la SUNAT (2019) afirma que: “Es obligatorio que el contribuyente debe poner en consulta con 365 días los comprobantes a sus clientes. Además, estos comprobantes electrónicos emitidos la SUNAT pone a disposición la consulta, en donde se visualizará la información tributaria.” (parr. 13)

Ventajas

La migración hacia la Facturación electrónica implica ventajas, tanto para el emisor como para el receptor. Por ello, la SUNAT afirma lo siguiente:

Se logra la simplificación del proceso de emisión de facturas reduciendo los tiempos y costos de los procesos de facturación. El ahorro de tiempo y los costos de los transportes. La eliminación del tiempo, costos de impresión y almacenaje de los comprobantes de pago físicos. Además, mediante el uso de la factura electrónica se genera el importante impacto ecológico positivo, no será necesario usar el papel que se realizan anualmente para los millones de operaciones comerciales (2019, parr.11).

Figura 9. Comprobante Electrónico Digital emitidos desde los SEE

Fuente: Sunat (2019)

Cantidad	Unidad	Medida	Descripción	Valor Unitario
1.00	UNIDAD		SILLA GIRATORIA PARA OFICINA	100.00

Sub Total Ventas :	S/ 100.00
Anticipos :	S/ 0.00
Descuentos :	S/ 0.00
Valor Venta :	S/ 100.00
ISC :	S/ 0.00
IGV :	S/ 18.00
Otros Cargos :	S/ 0.00
Otros Tributos :	S/ 0.00
Importe Total :	S/ 118.00

Valor de Venta de Operaciones Gratuitas : S/ 0.00

SON: CIENTO DIECIOCHO Y 00/100 SOLES

Esta es una representación impresa de la factura electrónica, generada en el Sistema de SUNAT. Puede verificarla utilizando su clave SOL.

Dimensiones e Indicadores

En este apartado se hará un hincapié hacia las dimensiones e indicadores que abarcan el estudio propuesto.

Dimensión: Liquidación y elaboración de la factura

Se determinaron las dimensiones que serán evaluadas dentro del proceso, y se necesitarán medir desde cuando se elabora la factura hasta la emisión a la entidad supervisora.

Indicador: Nivel de Incidencias en emisiones

Cuylen [et al.] (2016), menciona que; “Las incidencias en emisiones se limitan a el número de documentos que se anulan durante el mes en la empresa.” (p.47).

Según lo citado anteriormente; Cuylen [et al.] (2016) sostienen la siguiente fórmula para el caso propuesto:

$$PIE = \frac{TA}{TE}$$

Dónde:

PIE: Porcentaje de incidencias en emisiones.

TA: Total de documentos anulados.

TE: Total de documentos emitidos.

Indicador: Margen de Utilidad Operativa (MUO)

Bernard, sostiene que el indicador propuesto sostiene al respecto que:

El margen de utilidad Operativa indica cuánto dinero hace una compañía (antes de interese e impuesto) por cada dólar de venta. Este indicador puede utilizarse para determinar si los costos de operación de un negocio son demasiados elevados en relación con su volumen de producción o ventas, especialmente cuando se compara con otras compañías del mismo sector (2016, p.52).

Según lo citado anteriormente; Bernard (2016) sostiene la siguiente fórmula para el caso propuesto:

$$MUO = \frac{UO}{IT} * 100$$

Dónde:

MUO: Margen de Utilidad Operativa.

UO: Utilidad Operativa.

IT: Ingresos Totales.

Con estos indicadores alcanzados se podrán utilizar para obtener cifras que sean medibles en la etapa de Liquidación y elaboración del comprobante del proceso de facturación.

Variable Independiente: Aplicación Web Progresiva

Para Microsoft, una Aplicación Web Progresiva (PWA) son: “Son aplicativos web que progresivamente son mejoradas con similares características a las aplicaciones nativas en motores y plataformas de navegación compatibles, la instalación de inicio de la pantalla desde casa, las notificaciones push y la asistencia fuera de línea” (2019, parr.1).

Esta nueva tecnología denominada PWA con su llegada, incrementaran la funcionalidad de una Aplicación Tradicional dando inicio a una nueva generación de aplicaciones. SolidGearGroup, afirma lo siguiente:

PWA es el acrónimo de Progressive Web Apps y es sobre un tipo de aplicativos webs que se incorporará caracteres propios de aplicaciones nativas que desde avances actuales hace uso en los diferentes webs API, un compuesto de buenas prácticas y navegadores web. Como ejemplo, se usará las notificaciones PUSH, siendo una capacidad más popular de las aplicaciones nativas. (2018, parr.1)

Asimismo, para su impulsor Google, las PWA permitirán una interacción directa entre el Usuario y la Aplicación Web. Por ello, Google Partners sostiene al respecto que:

Las Progressive Web Apps (PWA) son aplicativos webs modernas sin conexión principalmente orientadas a dispositivos móviles, con opciones mejoradas para sistemas nativos. Fueron diseñados con el objetivo de ser el nuevo estándar de la Web móvil. (...) participando conjuntamente con las aplicaciones nativas con el alcance de la Web. Es una pieza fundamental de la plataforma web. La creación de una Progressive Web App de alta calidad ofrece ventajas, aumentando la participación, multiplicar las conversiones y facilitar la tarea de deleitar a los usuarios (2018, parr.3).

Las Aplicaciones Web Progresivas funcionan en todas partes, independientemente de si los usuarios están en un dispositivo móvil. Google Partners (2018, parr.7), menciona algunas de las ventajas:

Son dignas de estar en la pantalla principal: Cuando se cumplen los criterios de Progressive Web App, Chrome inicia preguntando al usuario si desea agregar a su pantalla principal aplicaciones web progresivas.

Funcionan de forma confiable sin importar las condiciones de la red: A través de los service workers, Konga para cargar la página inicial pudo enviar un 63% menos de datos y para completar la primera transacción un 84% menos de datos.

Aumentan la participación: Las notificaciones de aplicación web ayudaron a que eXtra Electronics cuadruplicara la participación. Además, esos usuarios estuvieron el doble de tiempo en el sitio.

Mejoran las conversiones: La capacidad de brindar al usuario una experiencia de alta calidad ayudó a que AliExpress a los usuarios nuevos mejore un 104% las conversiones en todos los navegadores y las conversiones en iOS aumentara un 82%.

Las Progressive Web Apps (PWA) son experiencias de usuario que tienen al alcance de la Web, y para Google Partners (2018, parr.4) mencionan las siguientes características:

Son confiables: Cuando se inician desde la pantalla principal del usuario, los service workers permiten que una Progressive Web App se cargue al instante, sin importar el estado de la red. Un service worker es como un proxy de cliente, escrito en JavaScript, que brinda control sobre la memoria caché y la forma de responder a las solicitudes de recursos. Como los recursos clave se almacenan previamente en la caché, es posible eliminar la dependencia de la red y así, garantizar una experiencia instantánea y confiable para los usuarios.

Son rápidas: El 53% de los usuarios abandonan un sitio si demora más de 3 segundos en cargarse. Y, una vez cargado, esperan que sea rápido, con interfaces ágiles y desplazamiento fluido.

Son atractivas: Las Progressive Web Apps pueden instalarse y residir en la pantalla principal de los usuarios, sin necesidad de pasar por la tienda de aplicaciones. Estas aplicaciones brindan una experiencia de pantalla completa envolvente con ayuda de un archivo de manifiesto de aplicaciones web y también permiten

volver a interactuar con los usuarios mediante notificaciones de aplicación web.

Funcionamiento

Asimismo, Google Partners, sostiene que el funcionamiento de una Aplicación Web Progresiva es:

Un elemento fundamental para el funcionamiento y la interacción de las Progressive Web Apps es el Service Worker, se trata de un script que el navegador ejecuta en segundo plano, en separación de la página web o aplicación, generalmente está destinado a funciones que no ameritan la interacción con la propia página web o el usuario. Además, el Service Worker proporciona la base técnica basadas en características como las notificaciones push, sincronizaciones periódicas en segundo plano y las experiencias fuera de línea” (2018, parr.7).

Service Workers

Para SolidGearGroup, sostiene al respecto que un service worker es:

El motor principal de una PWA. A alto nivel, la primera vez que accedes a una PWA desde tu navegador, el service worker se inicia como proceso dentro del navegador y ya siempre quedará corriendo, incluso cuando cierres la PWA, él seguirá funcionando. Se encarga de gestionar todo el tráfico de red de tu PWA, las notificaciones push, la cache para la carga instantánea, la actualización de la app en segundo plano y muchas cosas más. A nivel más técnico, un service worker es un archivo JavaScript, que se ejecuta en segundo plano y se basa en eventos (2018, parr.9).

App Shell

Asimismo, SolidGearGroup (2018), sostiene que un App Shell es “un concepto de diseño mediante el cual la carga inicial del esqueleto de la aplicación es lo primero que se hace, mientras que el contenido de la

aplicación se carga después. Adoptar el uso del App Shell no es obligatorio, pero si una buena práctica para lograr una carga instantánea de nuestra aplicación” (parr.12).

Cache

SolidGearGroup (2018), afirma que: “para dotar nuestra PWA de capacidad para trabajar offline y cargar de manera instantánea es necesario disponer de un sistema para almacenar datos. Los tradicionales API web no nos sirven para este caso de uso, por eso se ha desarrollado un nuevo cache API para poder ser usado directamente desde los service workers” (parr.15).

App Manifest

Asimismo, SolidGearGroup (2018), afirma que el App Manifest es “un archivo JSON con un formato específico donde caracterizamos nuestra PWA y sus metadatos. Es la clave para convertir nuestra aplicación en una aplicación instalable. En ese fichero se especifica el nombre, la URL de inicio, el tema, los iconos” (parr.19).

Arquitectura del Sistema de Información Propuesto

Talledo (2015) define la Arquitectura del Sistema como el “Arte de planear, diseñar y construir un Sistema” (p. 38). En la cual se debe de tomar en cuenta los siguientes apartados para la Aplicación propuesta:

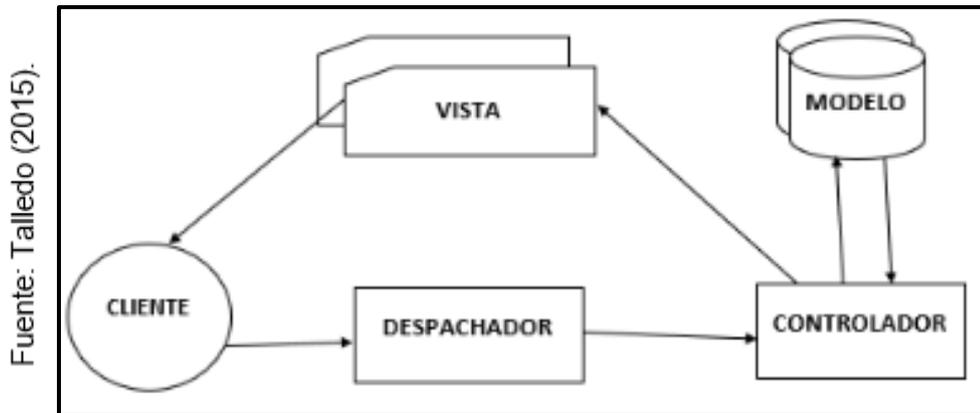
Accesibilidad: Talledo (2015) nos indica que “refiere a las posibilidades de acceso que es independiente de sus características físicas o individuales del contexto de uso (tecnologías disponibles)” (p.38).

Navegabilidad: Talledo (2015) nos define este aspecto como “la facilidad con que un usuario se puede desplazar atreves del sitio web sin perderse, Si un sitio web es claro, sencillo, comprensible, ofreciendo al usuario una satisfactoria experiencia” (p.38)

Usabilidad: Talledo (2015) lo define como “el término que es utilizado de la mejor forma para analizar los diseños de sitios web para que interactúen las

personas con ellos de una manera más intuitiva, cómoda y fácil posible” (p.39).

Figura 10. Arquitectura MVC



Talledo (2015) define a los modelos vista controlador (MVC) como “un patrón de arquitectura de un software que separa las aplicaciones y los datos de las interfaces del usuario y las comunicaciones que es gestionado por el módulo encargado y los eventos” (p.94).

Modelo: Para Talledo (2015) lo define como “la capa que trabaja con los datos. Es importante que contenga mecanismos para actualizar su estado y también acceder a la información. Se debe tener en cuenta que la petición de acceso y/o los datos manipulados o información que llegan al “modelo” por medio de controladores” (p.94).

Vista: Talledo (2015) nos dice que “esta parte se encarga de mostrar la información solicitada. Para mostrar dicha información contendrá el código de la aplicación necesaria. En entornos web, mayormente será, mediante código CSS, HTML y JavaScript que será el encargado de interpretarlo” (p.95).

Controlador: Según Talledo (2015), afirma que “Es la responsable a responder a eventos, por ejemplo, si se requiere hacer una compra a petición del usuario, búsqueda de información, visualizar un elemento, etc. Esto

genera una petición al “modelo”, de esta forma generar el resultado correspondiente a la información del que se dispone” (p. 95).

Herramientas para el desarrollo del Sistema de Información Propuesto Visual Studio

Microsoft Press (2016), afirma que: “Visual Studio es parte del entorno familiar de desarrollos integrados (IDE) Ligadas al desarrollo de aplicaciones que soporta numerosas actividades, tales como la inclusión de pruebas unitarias o de código de fuente y la creación de interfaces gráficas.” (p.17)

Para el desarrollo del Sistema propuesto de opto por la versión 2015, ya que es la más estable; además que cuenta con un repositorio más amplio, en la que se incluirán diferentes herramientas para su desarrollo, las cuales se definirán más adelante.

Nettiers

Asimismo, Microsoft Press define al framework Nettiers como:

Una plantilla de código abierto para CodeSmith escrito en C#. La intención de esta plantilla es asistir a los desarrolladores eliminando el código repetitivo, que tenemos que escribir cada vez que desarrollamos una aplicación con conexión a datos. Al mismo tiempo nos provee de un framework completo para comenzar a trabajar en las partes más importantes de nuestra aplicación (2016, p.14).

Para el desarrollo se tomó la versión 2.32 del Nettiers, ya que es la más estable a la fecha en compatibilidad con ASP.NET 4.5.2 y el entorno Visual Studio 2015.

ASP.NET

Microsoft Press (2016) lo define a la herramienta ASP.NET como:

Los fabricantes de software servidor reaccionan a CGI con propuestas de interfaces web (NSAPI de Netscape, ISAPI de Microsoft), e introduciendo paginas dinámicas. [...] Estas tecnologías son basados en el uso de un lenguaje específico, de una dedicada a la Base de Datos, de un HANDLER indicando a los Servidores Web páginas que no se debe de descargar y a la decodificación de Consultas HTTP, pero si se deben ejecutar. Dicho de otro modo, una nueva extensión es reconocido por los servidores web (ASP en el caso de Microsoft) para ejecutar las paginas correspondientes (2016, p.17).

Para el desarrollo se optó por la versión 4.5.2, ya que es la más estable en compatibilidad con el framework Nettiers propuesto.

Base de Datos SQL Server

Ademas, Gabillaud afirma que la Base de Datos SQL Server es:

Un SGBDR (Sistema de Gestión de Base de Datos Relacional) integrados a Windows, a nivel de administración realizando simplificaciones numerosas, que ofrecen un máximo de posibilidades. [...] Una gran capacidad que le confiere de su coherencia, conservar la integridad y gestionando los datos (2014, p.15).

Para el desarrollo del Sistema propuesto de opto por la versión 2012, ya que es con que la empresa tiene instalada y configurada en su servidor para dotar a sus diferentes aplicaciones web y escritorios instalados.

Report Builder

Microsoft Press (2016) afirma que, “es una herramienta de creación de informes de Microsoft que le permite desarrollar informes simples y complejos, y paneles basados en una variedad de fuentes de datos.” (p.15).

Para el desarrollo de la plataforma propuesta se optó por la versión 2012 SP3, ya que es la más estable y actualizada a fecha en compatibilidad con el Gestor de Base de Datos propuesto.

Component One

De igual manera, GrapeCity sostiene que la herramienta Component One se define lo siguiente:

Un repositorio que incluye cientos de controles para ocho plataformas de desarrollo creando aplicaciones web y desktop responsive ajustados al presupuesto y las fechas de entrega. Con controles que incluyen IU, visualización de datos complejos y se incluyen plataformas como WPF, ASP.NET, WinRT y WinForms, donde se tendrá lo necesario para crear aplicaciones táctiles más modernas y también se pueda actualizar y mantener aplicaciones heredadas. (2019, p.15).

Para el desarrollo de la plataforma propuesta se optó por la versión 20173.1.9, ya que es con que la empresa mantiene una licencia pagada para el desarrollo de los aplicativos webs, escritorio y/o móviles.

Agencia de Viajes

De la Torre, sostiene que una agencia de viajes se denomina lo siguiente:

Es una empresa turística que actúa con fines de lucro como agente intermediario activo entre prestadores de servicios específicos y sujetos de desplazamiento turístico. [...] En una agencia de viajes que casi la mayoría de sus ingresos se derivan de la comisión, que les otorgan prestadores de servicio sobre diversas líneas de productos (2017, p.11).

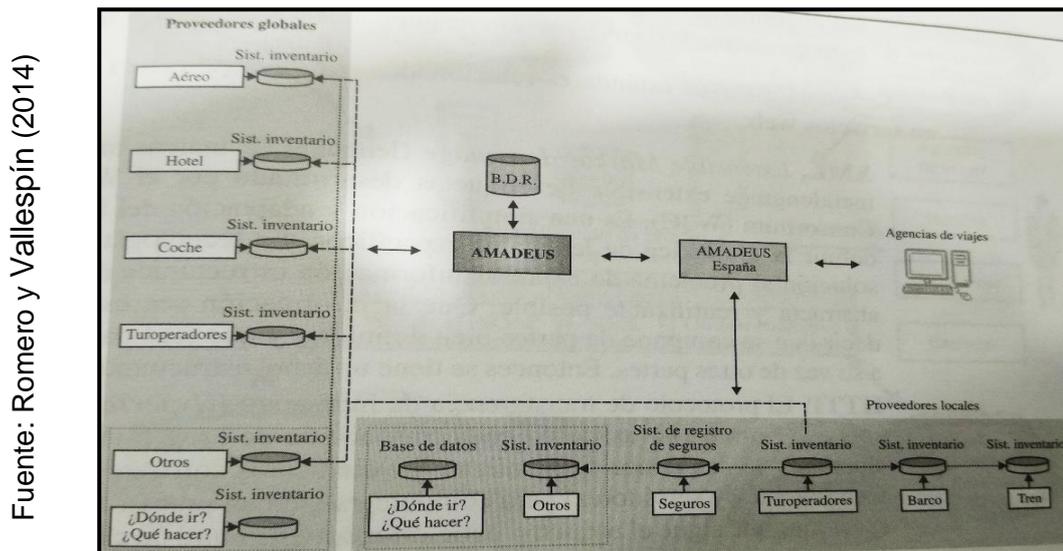
Global Distribution System (GDS)

Un GDS es un intermediario entre compañías aéreas y agencias que venden sus servicios a un cliente final, por lo que los ingresos que generar son a

partir de las comisiones por cada servicio vendido. Por ello, Romero afirma lo siguiente:

Los GDS son como bases de datos grandes centralizados que permitiendo el acceso a los agentes abonados. Los GDS tienen la capacidad de actualizar y almacenar instantáneamente cantidades enormes de informes sobre la oferta de una gama amplia de empresas turísticas. Estos sistemas GDS engloban consorcios de servicios turísticos y proveedores de productos (hoteles, cruceros, operadores, aseguradoras, rent-a-car, tour, vuelos etc.), suministrando información en tiempo real sobre precios, características y disponibilidades. De este modo, los terminales de los GDS que son instaladas en las agencias de viaje, por medio de ello se compran y reservan los servicios ofertados. Además, el sistema liquida las deudas correspondientes e imprime los billetes, configurando una red mundial gigantesca que se dedica a la venta de productos turísticos (2014, p.64).

Figura 7. Sistema de comunicaciones del GDS Amadeus



La empresa Continental Travel SAC, maneja las licencias de 2 de los GDS más conocidos, los cuales son Amadeus y Sabre. Asimismo, Romero y

Vallespín (2014, p.65) afirma que los productos comercializados por los GDS son:

Reserva de hoteles: permite el acceso al sistema de inventario de un hotel en tiempo real con las últimas ofertas, precios, tipo de habitación y aspectos de interés diferentes para el potencial cliente.

Producto Aéreo: suministran información que tiene la capacidad de cubrir todas las necesidades de comercialización de la oferta de las compañías aéreas de una manera eficiente y segura.

Transporte Ferroviario: proporciona en tiempo real de una amplia gama de trenes, [...], sus reservas, disponibilidad y horarios.

Transporte Marítimo: obteniendo información de sus, tarifas, reservas de plazas, horarios, disponibilidad y emisiones automáticas de billetes de todos los productos que ofrece la empresa de transporte.

Alquiler de coches: facilita a la agencia la reserva de vehículos y con un modo estandarizado el acceso a la información, incluyendo servicios en el momento en que la reserva de vuelo es realizada.

Figura 11. Principales Compañías propietarias de los GDS

	Amadeus	Sabre	Travelport	
			Galileo	Worldspan
Año fundación	1987	1987	1993	1990
Fundada por	Air France, Iberia, Lufthansa, SAS.	American Airlines.	Aer Lingus, Air Canda, Alitalia, Austrian Airlines, KLM, Olympic, Airwaya, Swissair, Tap, United Airlines, US Airwaya.	Delta Airlines, Northwest, TWA.
Sede	Madrid, España	Southlake, Texas, USA	Atlanta, Georgia, USA	Atlanta, Georgia, USA
Agencias de viajes conectadas	66.859	56.000	43.500	16.000
Países en los que opera	195 países	195 países	170 países	

Fuente: Romero y Vallespín (2014)

Passenger Name Record (PNR)

Un GDS gestiona diferentes ofertas de servicios, porque su manejo implica gestionar correctamente las consultas PNR. Por ello, Romero y Vallespín afirma lo siguiente:

El PNR son reservas (de hotel, de avión, etc.), un itinerario que abarca el registro de uno o grupos de viajeros. Es decir, el PNR relacionadas con el viaje contiene toda la información y los detalles de la reserva. Para este propósito, LA IATA, se definió como un estándar sobre las disposiciones y en la actualidad abarca los PNR que sigue en vigor para las reservas aéreas. (2014, p.51)

La estructura de un PNR es crucial en un GDS, ya que una errónea digitación implica comprar, reservar o incluso anular servicios; por lo que se debe tener un conocimiento adecuado antes de emitir un PNR. Asimismo, Romero y Vallespín (2014) afirma que: “Un PNR son denominados elementos, como un conjunto de unidades de información, existentes en un mismo PNR con un máximo de 999. Pero, al menos 5 elementos obligatorios deben contener” (p. 69).

Asimismo, Romero y Vallespín (2014, p.69) menciona y recalca la estructura que deberá contar un PNR para su próxima emisión, por ello nos menciona los datos principales de su estructura:

- Apellidos y nombres de Clientes.
- Itinerarios, fecha/clase/vuelo/días del origen/semana y destino/información/situación adicional/hora de llegada/hora de salida/observaciones.
- Información de contacto de los clientes.
- Situación de emisión de billete(ticketing)
- Firma del responsable (Oficina y Agente que llevaron a cabo la reserva).

Figura 12. PNR emitido como Reserva en Amadeus

Fuente: Romero y Vallespín (2014)

Cabecera...	RP/MADIB230/MADIB0230	0999ZZ	02SEP / 1012Z	C04ECJ
Pasajeros...	1. SALCEDO/JUAN MR	2. HERNANDEZ / PEDRO MR		
Itinerario...	3 IB3162 C	23SEP 3 MADLHR HK2	0920 1030 320	0
	4 IB3163 C	25SEP 5 LHRMAD HK2	1145 1445 320	0
Teléfonos...	5 AP MAD 91	2450381 -H		
Ticketing...	6 TK	TL01SEP / MADIB0230		
Cabecera: Oficina responsable-Oficina para mensajes-código agente-fecha y hora creación PNR- localizador				

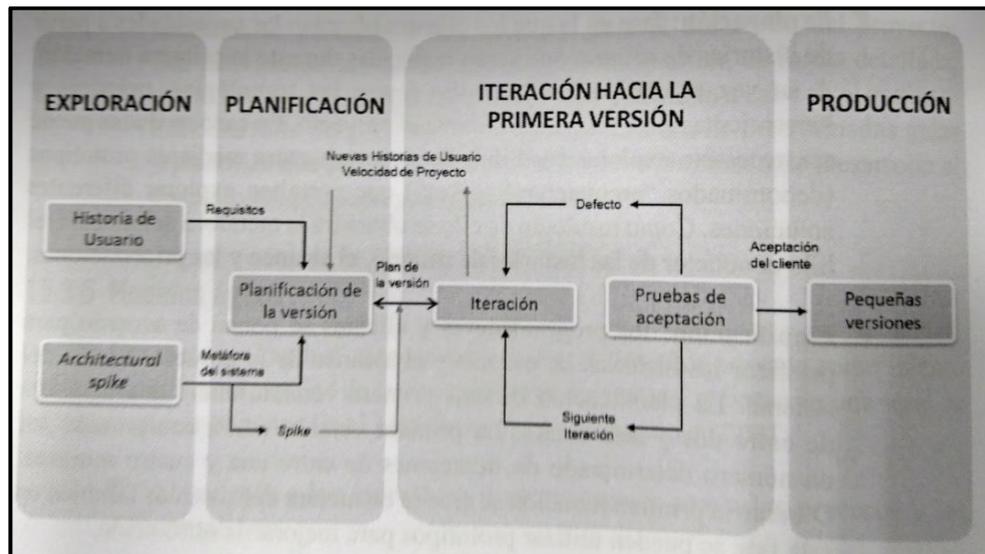
Metodologías de desarrollo para el Sistema de Información Propuesto

Extreme Programing (XP)

Según RA-MA (2015), define a extreme programming (XP) que “Se fusa para desarrollo del software de alta calidad, más sencilla y rápida, donde es beneficiado el cliente y pone más énfasis en la adaptabilidad” (p.319).

Figura 13. Ciclo de vida de Extreme Programming

Fuente: RA-MA (2010)

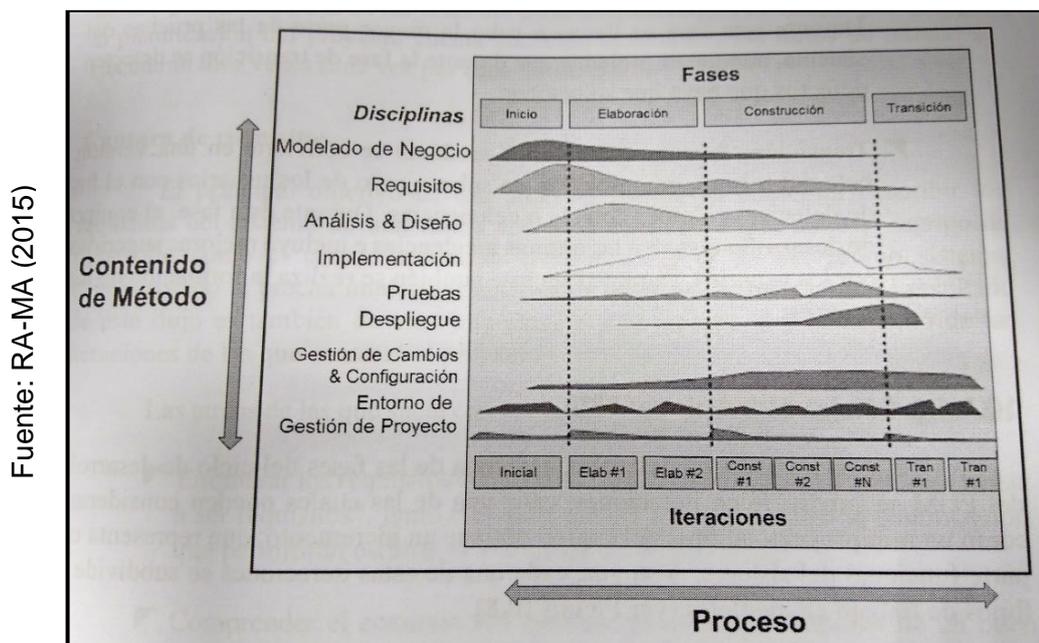


Rational Unified Process (RUP)

Asimismo, RA-MA define al Proceso Unificado de Rational (RUP) lo siguiente:

Metodologías que propone dividir el proyecto en fragmentos más pequeños o mini proyectos, cada uno de los mini proyectos recibe el nombre de iteración y produce como resultado un incremento. El desarrollo de un producto software siguiendo el RUP se extiende a lo largo de un ciclo completo. El ciclo de desarrollo se tiene cuatro fases (Inicio, Elaboración, Construcción y Transición), de las cuales formado por un conjunto de artefactos que deben haber sido desarrollados. Una vez finalizado el ciclo se obtiene una versión del producto lista para entregar al cliente. Si con el paso del tiempo surgieron nuevos requisitos, o el sistema tuviera que ser sometido a tareas de mantenimiento, se iniciara un nuevo ciclo de desarrollo tras cual, se obtendría una nueva versión del sistema adaptado a las nuevas necesidades (2015, p.312).

Figura 14. Ciclos, fases, iteraciones y flujos de trabajo de RUP



SCRUM

Asimismo, según Portillo define a la metodología SCRUM lo siguiente:

Término que describe la forma en que se desarrolla los productos iniciados en Japón. Este término, ya en 1987 Hirotaka Takeuchi y Ikujiro Nonaka abarcaron un método utilizado en rugby, la cual cada integrante del grupo deberá actuar en equipo para ganar el partido, de esta manera denominan un tipo de proceso nuevo de desarrollo de productos. Esta designación es escogida por las semejanzas que existía entre el juego del rugby y los tipos de procesos: rápido, con pocos descansos, auto-organizable y adaptable, SCRUM son procesos para el control y las gestiones de los productos, eliminando lo complejo en estas áreas centrándose en satisfacer las necesidades del negocio con la construcción de software. Es escalable y simple, porque no está establecido las prácticas de ingeniería de los softwares, más bien se combina o aplica, mas fácil, con otras prácticas de ingeniería (2014, p.128).

Criterio de evaluación y selección de la metodología:

Al considerar las 3 metodologías que son evaluadas en este estudio, se decidió realizar un juicio de expertos para considerar aspectos claves generales. Apoyándonos de 3 evaluadores quienes calificaron cada metodología, se puede apreciar en el Anexo N° 4:

Tabla 1. Validación de expertos

Expertos	Metodologías		
	SCRUM	RUP	XP
Juanita Isabel Cueva Villavicencio	21	11	18
Romero Valencia Mónica Patricia	19	16	15
Gálvez Tapia Orleans	35	23	34
Total	75	50	67

Fuente: Elaboración propia.

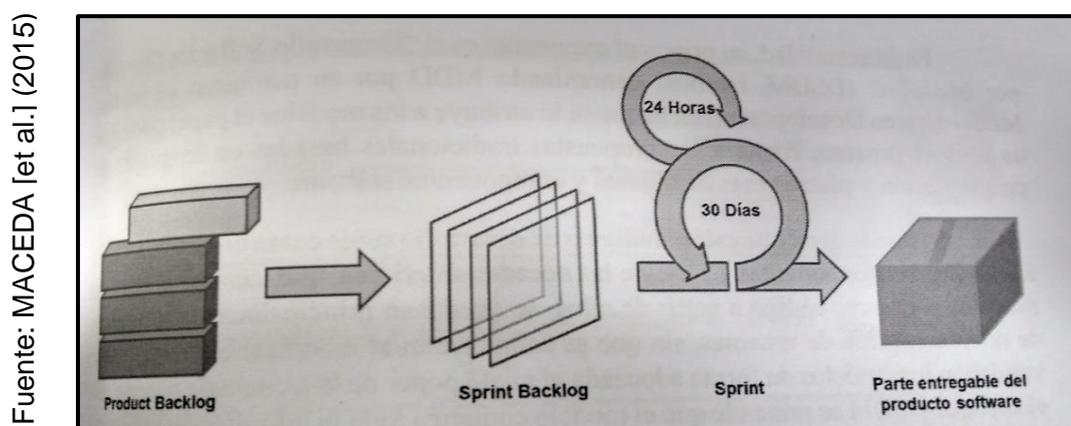
La metodología seleccionada para el desarrollo de una Aplicación Web Progresiva para los procesos de facturación en la empresa CONTINENTAL TRAVEL S.A.C. será SCRUM por ser una metodología ágil permitiendo ser más flexible con respecto a las necesidades del equipo de desarrollo y de los clientes. Sin embargo, en el mercado actual, al laborar en una consultora de desarrollo de software, las empresas demandan en la entrega rapidez y calidad, esta metodología se centra más en ciclos cortos del desarrollo del producto que puedan satisfacer la demanda del cliente.

Metodología seleccionada: SCRUM

Según Maceda [et al.], define a la metodología SCRUM lo siguiente:

Un método ágil para generar soporte a la administración que se aplica a casi cualquier proyecto. Debido a su uso actualmente es más conocido en el desarrollo de software de proyectos. Como método ágil, Scrum de manera incremental e iterativa permite el desarrollo de software. En Scrum un equipo auto-organizado y multifuncional que crea gradualmente en varias iteraciones cortas un producto. Se valora el rendimiento del equipo con cada iteración, como también el resultado del producto (2015, p.59).

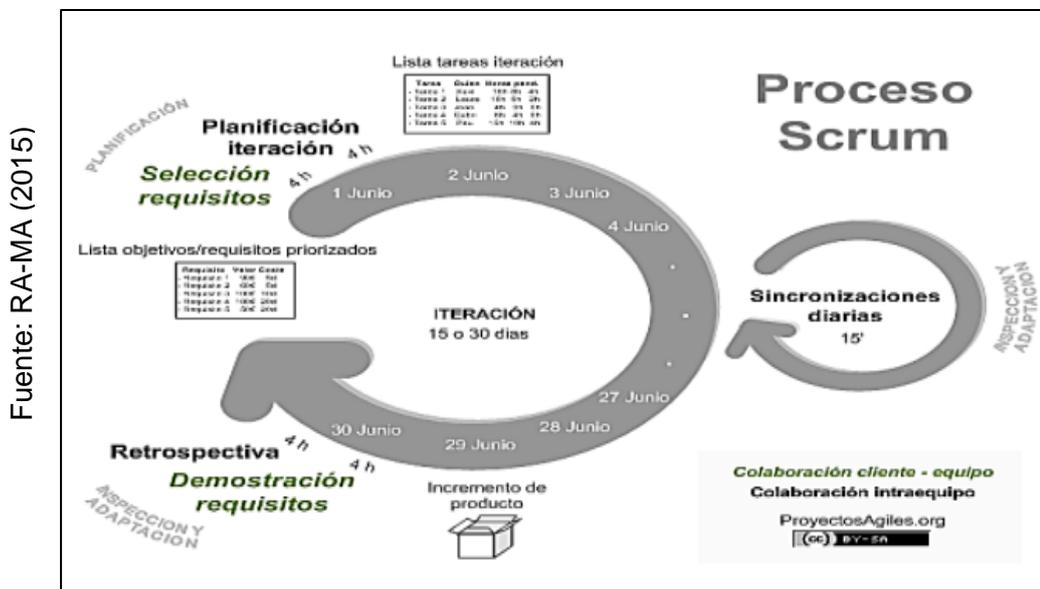
Figura 15. Ciclo de vida de Scrum



Asimismo, RA-MA (2015) afirma que: “Cada sprint mantiene una secuencia de características diferentes de un ciclo de vida en cascada” (p.34). Por ello el autor cita lo siguiente:

- El diseño es continuo, así como la integración durante el desarrollo las pruebas: estas actividades son realizadas continuamente a lo largo del sprint. Es un ciclo de vida en cascada se realiza de manera secuencial, tratándose de compartimientos estancos.
- El equipo de trabajo: en Scrum el equipo de desarrollo realizan diferentes tareas (pruebas, diseño, desarrollo, estimación, etc.), sin que exista una especialización, como sucede en el ciclo de vida en cascada.
- La duración del sprint no puede modificarse: en un proyecto que sigue una metodología en cascada se puede extender el tiempo planificado, mientras que un sprint no, aunque no se pueda cumplir con las actividades planificadas para ese sprint.

Figura 16. Proceso Scrum



Roles Principales

SCRUM al ser una metodología ágil, posee una estructura practica en relación al equipo de trabajo, por ello según RA-MA (2015), define lo siguiente:

- **Product Owner**

Según RA-MA (2015), menciona que “Al representar la voz del cliente se debe asegurar la forma de trabajo adecuado del equipo Scrum. El Product

Owner redacta las historias de los usuarios, y luego es colocado en el Product Backlog.” (p.112).

- **Scrum Master**

Según RA-MA (2015), menciona que “Es primordial que el grupo logre el objetivo del sprint eliminando los obstáculos que lo impiden. El Scrum Master asegura que los procesos Scrum sea utilizado como corresponde. El Scrum Master es encargado de hacer cumplir las reglas.” (p.112).

- **Scrum Team**

Según RA-MA (2015), menciona que “Para la entrega del producto el equipo es responsable. Un equipo pequeño que realice las habilidades transversales que son necesarios para el cumplimiento del trabajo (diseño, desarrollo, análisis, documentación, pruebas, etc.” (p.112).

Elementos

Asimismo, según RA-MA (2015) afirma que, durante el desarrollo de la metodología, se construye diferentes productos: de las cuales se definen como:

- **Product Backlog**

Según RA-MA (2015), menciona que “Las funcionalidades del producto es por medio de una “wish list”. Es trabajado por el Product Owner y para el negocio son priorizadas las funciones de acuerdo a lo que es menos y de más importancia” (p.114).

- **Sprint Backlog**

Según RA-MA (2015), menciona que “son subconjuntos de ítems del Product Backlog, seleccionados por el equipo para realizarlo a lo largo del Sprint lo que se trabajará. La duración es establecida por el equipo de cada Sprint” (p.114).

- **Sprint Planning Meeting o Reuniones**

Según RA-MA (2015), menciona que “al comenzar cada Sprint se ejecuta la reunión y es definida en cómo se enfoca el proyecto que vendría a ser del Product Backlog los plazos y las etapas” (p.114).

- **Daily Scrum o Stand-up Meeting**

Según RA-MA (2015), menciona que “Durante el periodo de Sprint es una reunión breve que es realizado a diario. Individualmente se responde tres preguntas: ¿Qué ayuda necesito?, ¿Qué hare hoy?, ¿Qué hice ayer?, El Scrum Master debe intentar la solución a los obstáculos y problemas que se pueden presentar” (p.114).

- **Sprint Review:**

Según RA-MA (2015), menciona que “el sprint terminado se revisa, con una avance tangible y claro para ser presentado al cliente” (p.115).

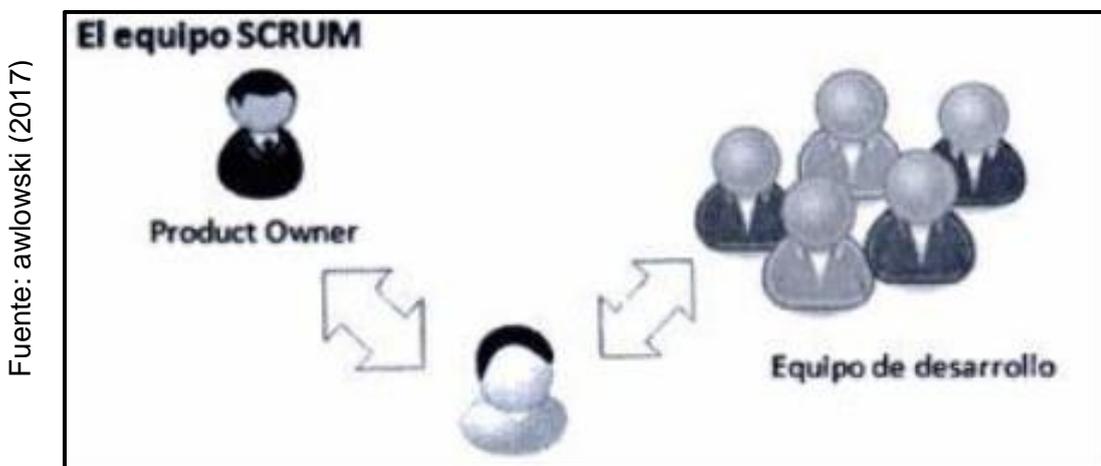
- **Sprint Retrospective:**

Según RA-MA (2015), menciona que “Los objetivos cumplidos del Sprint terminado son revisados por el equipo. Se anota lo malo y lo bueno, para que los errores no se vuelvan a repetir. Esta etapa del proceso de desarrollo sirve para implementar mejoras” (p.115).

- **Historias de Usuario**

Según Scrum Manager Book(2015), es “una funcionalidad que tiene que incorporar un sistema de software, y al implementarse aportará valor al cliente.” (parr.3)

Figura 17. Equipo Scrum



1.4. Formulación del problema

Problema General:

- ¿De qué manera influye una Aplicación Web Progresiva en el proceso de facturación de la empresa Continental Travel S.A.C.?

Problemas Secundarios:

- ¿De qué manera influye una Aplicación Web Progresiva en el Nivel de incidencias en emisiones para el proceso de facturación de la empresa Continental Travel S.A.C.?
- ¿De qué manera influye una Aplicación Web Progresiva en el Margen de Utilidad Operativa para el proceso de facturación de la empresa Continental Travel S.A.C.?

1.5. Justificación del estudio

El propósito de esta tesis es la implementación de esta tecnología web, que permitirá al proceso de facturación tener una mejor gestión de los comprobantes de pago electrónicos, permitiéndole ser más rápido y eficaz a la hora de procesar los ticket generados por los Agentes de Viajes; con un costo más que justificado por su desarrollo este sistema gestionara la información de clientes, proveedores, comprobantes de pago, aerolíneas entre otros datos relevantes en el proceso de facturación teniendo un control adecuado de la información entre las diferentes áreas de la empresa lo solicite de manera más rápida y segura permitiendo agilizar el tiempo de respuesta para tomar decisiones.

El presente proyecto hará su aporte en cuatro ámbitos, que serán explicados a continuación:

- **Justificación Tecnológica**

En la actualidad, "Estos cambios de los sistemas de información en que operan las organizaciones, importantes mejoras son logradas mediante su uso, automatizando los procesos de la empresa, facilidad del logro de

ventajas competitivas al implementarlo en las empresas” (Escorsa y Valls, 2016, parr.4).

El desarrollo de la Aplicación Web Progresiva hoy en día es un software que va en crecimiento, cambiará la manera en que la compañía Continental Travel S.A.C. interactúa en todo momento con sus clientes, es decir que será un canal de comunicación directa.

- **Justificación Económica**

Para De Pablos (2015), “el ahorro en costes de recursos humanos es provocado por usar sistemas de información, aumento en calidad en general, ahorro en la disminución de errores, y en tiempos de procesamiento de información” (p.12).

La empresa Continental Travel S.A.C., es una agencia de viajes que procesa información las 24 horas al día y los 7 días de la semana, por ello los beneficios económicos serán evidenciados tanto para la empresa como para los clientes, que requiere optimizar los tiempos de procesamiento de datos, disminuyendo el error humano el cual provoca que el comprobante de pago sea anulado o en el peor de los casos sea ignorado; el cual el contador realiza un resumen mensual pero si se presenta incoherencia, la SUNAT le impone una multa.

- **Justificación Institucional**

Asimismo, según Huidobro sostiene al respecto que:

En cada mesa de trabajo la PC como el teléfono comparten un lugar destacado, entre ellos no existía ninguna relación hasta hace poco, muy pronto será una realidad entre ellos por la convergencia hacia un único equipo, está empezando a aparecer en el mercado gracias a las nuevas aplicaciones y que las empresas hacen uso para mejorar el servicio a los clientes y su productividad (2015, p.25).

Por ello, he propuesto el desarrollo de una Aplicación Web Progresiva como una alternativa eficaz y eficiente para el procesamiento de datos;

accediendo así a toda la información, no solo a la información que se procesa para emitir los comprobantes de pago, además de obtener acceso a la información de la clientela o los proveedores como, paquetes turísticos, hoteles, vuelos, etc., en diferente lugar y momento, a su vez permite, a la empresa mejorar no solo su proceso sino también su competitividad y su imagen, de esta forma consolidar su estado en el mercado.

- **Justificación Operativa**

Según Cardador (2014) menciona que, “esta amigable interfaz y manejable comprensión y manejo, que desempeña de forma cómoda y en tiempos de respuesta generando una mayor efectividad” (p.22).

La implementación de una Aplicación Web Progresiva permitirá controlar adecuadamente los tiempos de control y desarrollo, cumpliéndose los plazos de objetivos y entrega. También, en las 24 horas del día los usuarios tendrán acceso al sistema web y se podrá revisar en el transcurso día los comprobantes de información y pago de los servicios, cancelarlos o corregirlos, todo ello vendría a estar en la solución del software.

1.6. Hipótesis

Hipótesis General:

- Una Aplicación Web Progresiva mejora el proceso de facturación de la empresa Continental Travel S.A.C.

Hipótesis Específicas:

- Una Aplicación Web Progresiva disminuye el nivel de incidencias en emisiones en el proceso de facturación de la empresa Continental Travel S.A.C.
- Una Aplicación Web Progresiva incrementa el Margen de Utilidad Operativa en el proceso de facturación de la empresa Continental Travel S.A.C.

1.7. Objetivo

Objetivo Principal

- Determinar de qué manera influye una Aplicación Web Progresiva en el proceso de facturación de la empresa Continental Travel S.A.C.

Objetivo Específicos

- Determinar de qué manera influye una Aplicación Web Progresiva en el Nivel de incidencias en emisiones en el proceso de facturación de la empresa Continental Travel S.A.C.
- Determinar de qué manera influye una Aplicación Web Progresiva en el Margen de Utilidad Operativa en el proceso de facturación de la empresa Continental Travel S.A.C.

II. MÉTODO

2.1. Diseño de investigación

2.1.1. Método de investigación

Para Cegarra, define método hipotético – deductivo como:

Busca solucionar los problemas planteados, de manera que lo empleamos corrientemente en la la investigación científica como en la vida ordinaria, en la que radica en emitir hipótesis al problema planteado acerca de las soluciones posibles (2015, pág. 82).

Para el manejo de datos este método ayudó para poder describirlos y en cómo llega a demostrar el objetivo de la investigación, permitiendo demostrar mediante valores con una secuencia de pasos los resultados de la investigación.

Tipo de Estudio

Con relación al estudio el autor Lozada (2014), declara que la investigación aplicada “es el procedimiento que toma conclusiones genéricas para aclaraciones específicas, es decir, se examinara las hipótesis planteadas” (p.39).

Además, Lozada (2014) menciona que: “la investigación aplicada tiene como objetivo construir en un conocimiento científico, por lo que se necesita desarrollar y profundizar cada vez nuestro conocimiento de la realidad, por lo que se pretende obtener contextos cada vez mayores como hipótesis, leyes, teorías entre otros” (parr.39).

Sin embargo, para Arias (2016), nos menciona que la investigación experimental: “es un proceso que somete al grupo de individuos o a un objeto, a condiciones determinadas (variable independiente), los efectos que se producen al observarlo (variable dependiente)” (p.27).

Se concluye que la presente investigación sigue un tipo Aplicada – Experimental, con el objeto de conocer el contexto de la investigación; se aplicará una Aplicación Web Progresiva (causa), y con ello analizar la influencia en los procesos de facturación (efecto) de la compañía Continental Travel S.A.C.

2.1.2. Diseño

Para Hernández [et. al], nos menciona que los diseños pre-experimentales:

Se caracterizan por tener el grado de control mínimo, por lo que administra un determinado tratamiento a un grupo de individuos o datos, para con ello se aplica una medición individual o en más variables y con el objetivo de observar el nivel grupal en estas condiciones propuestas (2014, p.24).

Para el presente estudio se concluyó en un diseño pre-experimental, ya serán medidos los indicadores propuestos en dos diferentes tiempos, por lo que se realizara una comparación entre dos diferentes de resultado.

$$G = O_1 X O_2$$

En dónde:

O1: Es el proceso de facturación antes de la implementación de la Aplicación Web Progresiva en la empresa Continental Travel S.A.C.

O2: Es el proceso de facturación antes de la implementación de la Aplicación Web en la empresa Continental Travel S.A.C.

X: Variable Independiente: Aplicación Web Progresiva

G: Grupo Experimental: Pre Test.

2.2. Operacionalización de Variables

2.2.1. Variable Independiente: Aplicación Web Progresiva

Definición Conceptual:

Para Microsoft, nos menciona que las aplicaciones web progresivas:

Progresivamente mejoran con características semejantes a las aplicaciones nativas motores de navegador y en plataformas compatibles, instalándose desde el hogar la pantalla de inicio, las notificaciones push y la asistencia fuera de línea (2018, parr.1).

Definición Operacional:

Es una herramienta tecnológica que al usuario le permite interactuar ya sea con o sin conexión a Internet con la página web desde cualquier plataforma, permitiéndole gestionar comprobantes de ticket, pago, y demás; para la empresa Continental Travel S.A.C.

2.2.2. Variable Dependiente: Proceso de facturación**Definición Conceptual:**

Para Catacora (2015), nos menciona que el proceso de facturación “tiene como función principal, el de registrar y procesar una serie de actividades relacionadas, cuyo objetivo no es solo conservar sino aumentar el número de ventas en una empresa” (p.282).

Definición Operacional:

En un conjunto de actividades, en la que se desenvuelven procesos las cuales son registrados y gestionan todo aquel comprobante de pago que son realizados dentro de la compañía Continental Travel S.A.C., en donde se relaciona unos con otros en que cada proceso, como se factura, si es recibido o son cobrados los servicios en la compañía.

Tabla 2. Matriz de Operacionalización de las variables

Tipo de variable	Variable	Dimensión	Indicador	Escala de Medición
Variable Independiente	Aplicación Web Progresiva			
Variable Dependiente	Proceso de Facturación	Liquidación y elaboración de la factura	Nivel de Incidencias en Emisiones	Razón o Proporción
			Margen de Utilidad Operativa	Razón o Proporción

Fuente: Elaboración propia.

Tabla 3. Matriz de Análisis de Indicadores

Dimensión	Indicador	Descripción	Técnica	Instrumento	Unidad de Medida	Formula
Liquidación y elaboración de la factura	Nivel de Incidencias en emisiones	Se analizará la cantidad de incidencia presentadas respecto al total de comprobantes de pago	Fichaje	Ficha de Registro	Unidad	$PIE = \frac{TA}{TE}$ <p>Dónde: PIE: Porcentaje de incidencias en emisiones. TA: Total de documentos anulados. TE: Total de documentos emitidos.</p>
	Margen de Utilidad Operativa	Se analizará la utilidad operativa sobre el total de ingresos de los comprobantes de pago válidos.	Fichaje	Ficha de Registro	Unidad	$MUO = \frac{UO}{IT} * 100$ <p>Dónde: MUO: Margen de Utilidad Operativa. UO: Utilidad Operativa. IT: Ingresos Totales.</p>

Fuente: Elaboración Propia

2.3. Población y muestra

La tesis será desarrollada en la empresa Continental Travel S.A.C., cuyo mercado está enfocado en la venta servicios de tipo turístico, el objeto de estudio es monitorear la cantidad de comprobante de pago validos requeridos para dicho proceso, así como también la comprobante de pago que han sido rechazado o anulados.

2.3.1. Población

Según el autor Tamayo (2014), define como “un conjunto que es finito o infinito, constituido de elementos como; seres o cosas, las cuales presentan atributos o características similares, las cuales son susceptibles a la observación. Por lo que se deduce que existe un universo de familias, empresas, instituciones, votantes, automóviles, etc.” (p.176).

Tabla 4: Matriz de Análisis de la Población

Población	Tiempo	Indicador
151 comprobantes emitidos en Mayo de 2019	1 mes calendario	Nivel de Incidencias en Emisiones
		Margen de Utilidad Operativa

Fuente: Elaboración Propia

2.3.2. Muestra

Según Tamayo (2014) afirma que es como “la parte de la población, donde se obtiene la información necesaria para poder desarrollar nuestra investigación; esta muestra, es representativa sobre una población, a partir de cual se aplicara la observación y medición de las variables (2014, p. 176).

$$n = \frac{Z^2 N}{Z^2 + 4N(EE^2)}$$

Donde:

- n = Tamaño de muestra
- Z = Nivel de confianza al 95% (1.96) elegido para esta investigación
- N = Población total del estudio
- EE = Error estimado (al 5%)

Resolución:

$$n = \frac{(1.96^2)(151)}{(1.96)^2 + 4(151)(0.05)^2}$$

$$n = \frac{(3.8416)(139)}{(3.8416) + (604)(0.0025)}$$

$$n = \frac{533.9824}{(3.8416) + 1.51}$$

$$n = \frac{580.0816}{5.3516}$$

$$n = 108.39405$$

Por lo tanto, de acuerdo a los resultados la medida de la muestra estará conformado con una cantidad de 108 comprobantes de pago emitidos durante el mes de Mayo de 2019.

2.3.3. Muestreo

A partir del análisis de la Población y Muestra de la investigación realizada, se concluyó que para la presente tesis se utilizara un muestreo probabilístico, de tipo aleatorio simple; además según el autor Valderrama (2015) nos indica que “su realización es por medio de un conjunto de variables en la cual se escoge al azar las variables u individuos completando el tamaño de la muestra previsto” (p.72).

2.4. Técnicas e instrumentos de recolección de datos, validez y confiabilidad

Para Chávez (2016), indica que “son procesos de obtención de información empírica permitiendo la evaluación de cada una de las variables de acuerdo a la unidad de análisis previstas, con el objetivo de obtener para el estudio del problema los datos necesarios [...]. Por lo tanto, a partir de ello los resultados se obtendrán en la investigación (p.53).

2.4.1. Técnicas de Recolección de Datos

- **Entrevista**

“es definido como la ejecución del cuestionario a un determinado individuo con el objetivo de obtener ciertos datos relevantes para un resultado establecido.” (Hernández [et. al], 2014, p.38). Que fue realizada al jefe de sistemas de la compañía Continental Travel (Anexo 6).

- **Fichaje**

“es conocer las fichas en cómo pueden ser usadas en los mecanismos plasmados por escrito información importante encontrados en los procesos de información y búsqueda, en cualquier momento que se desea tener al alcance” (Gavagnin, 2014, pág. 38). Por lo que, esta técnica recolectara los datos necesarios para los indicadores que son procesados en las facturaciones de la compañía Continental Travel S.A.C.

2.4.2. Instrumentos de Recolección de Datos

- **Ficha de Registro**

Para Gavagnin (2014) afirma que “son mecanismos de la investigación documentas que va admitiendo registros de los datos relevantes, que además de contener una extensión, genera un valor y una unidad” (pág. 38).

2.4.3. Validez

Según Mokkink (2014), definen que la validez como “la capacidad que tiene un instrumento para realizar su medición de acuerdo a lo que se pretende medir y cuál es el objeto de su diseño” (p. 539).

Juicio de expertos para la evaluación del instrumento

Según Hernández [et. al] (2014) afirman que el juicio de expertos es “la sugerencia que ofrecen los profesionales especializados en una disciplina, con coordinación de la investigación que se está realizando”. (p.204)

Los instrumentos de recopilación de datos utilizados en el estudio “Aplicación Web Progresiva para el proceso de facturación de la empresa Continental Travel S.A.C.”, mediante juicio de expertos han sido validados, por los profesionales especializados tal como se puede observar en la Tabla N° 5 (ver Anexo 5):

Tabla 5. Matriz de Resultados del Juicio de Expertos

Expertos	Nivel de Incidencias en emisiones	Margen de Utilidad Operativa
Mg. Juanita Isabel Cueva Villavicencio	80.00%	80.00%
Mg. Romero Valencia Mónica Patricia	88.00%	88.50%
Mg. Gálvez Tapia Orleans	80.00%	80.00%
Total	82.67%	82.83%

Elaboración Propia

De acuerdo Tabla N° 6 se aprecia la Tabla de Rango de validez para los instrumentos propuestos; y según las evaluaciones de expertos el instrumento para el Nivel de Incidencia en emisiones es elevado ya que el 82.67% está en el rango de <75% -100%> y el instrumento para el Margen de Utilidad Operativa también es alto debido al 82.83% está en el rango de <75% - 100%>.

Tabla 6. Tabla de Rango de Validez de Instrumentos

Intervalo	Nivel
0 %	Nula
0 – 25 %	Muy Baja
25 – 50 %	Regular
50 – 75 %	Aceptable
75 – 100 %	Elevada
100%	Total o Perfecta

Fuente: Hernández [et. al] (2014, p. 207)

2.4.4. Confiabilidad

Según Palella y Martins (2012) afirma que la confiabilidad “tiene por definición a la seguridad que el investigador presenta sobre los resultados de la medición, es decir como poder sostener que las mediciones realizadas son comparablemente aceptables”. (p.102)

Método Test – Retest

Según Aiken, sostiene al respecto que el método Test – Retest:

El procedimiento test-retest identifica como medición segura a la confiabilidad, el estudio es aplicado en dos momentos diferentes a una muestra de sujetos y están correlacionadas ambas medidas. Lo cual, si es correcta la prueba, muy similares deberán ser las medidas y con un coeficiente de confiabilidad es a 1. Si cumple estos criterios se aprueba la confiabilidad del test por ser indiferente a quien lo empleo, debido a las dimensiones como resultado serán las mismas (2014, p. 86 – 87).

Técnica: Coeficiente de correlación de Pearson

Asimismo, según Soler sostiene al respecto que:

Entre dos variables aleatorias cuantitativas es la medición de su relación lineal. La escala es independiente de las medidas de las variables. Lo cual se define como un índice que se utiliza siempre que sean cuantitativas para la calibración del nivel de asociación de las variables (Soler, 2015, p. 10).

Tabla 7. Matriz de Análisis del Coeficiente de Correlación de Pearson

Rango	Confiabilidad (Dimensión)
0,81-1	Muy Alta
0,61-0,80	Alta
0,41 -0,60	Media
0,21 -0,40	Baja
0-0,20	Muy baja

Fuente: Palella Santa (2012)

Aplicando la Correlación de Pearson sobre el indicador “Nivel de incidencias en emisiones” se obtuvo un 0.758; por lo tanto, el nivel de confiabilidad sobre el indicador propuesto, es una “Correlación Alta” por lo que se considera Aceptable.

Tabla 8. Resultados SPSS – NIE

		PRE_TEST	RE_TEST
PRE_TEST	Correlación de Pearson	1	,758**
	Sig. (bilateral)		,001
	N	15	15
RE_TEST	Correlación de Pearson	,758**	1
	Sig. (bilateral)	,001	
	N	15	15

** . La correlación es significativa en el nivel 0,01 (bilateral).

Fuente: Elaboración propia.

Asimismo, para el indicador “Margen de Utilidad Operativa” se obtuvo un 0.804; por lo tanto, el nivel de confiabilidad sobre el indicador propuesto, es una “Correlación Alta” por lo que se considera Aceptable.

Tabla 9. Resultados SPSS – MUO

		PRE_TEST	RE_TEST
PRE_TEST	Correlación de Pearson	1	,804**
	Sig. (bilateral)		,005
	N	10	10
RE_TEST	Correlación de Pearson	,804**	1
	Sig. (bilateral)	,005	
	N	10	10

** . La correlación es significativa en el nivel 0,01 (bilateral).

Fuente: Elaboración propia.

2.5. Métodos de análisis de datos

En la tesis el estudio ejecuta y procesa la información adquirida a por medio de herramientas de recolección de datos; la cual una vez ya dado el procesamiento de los datos se hace uso estadístico para evaluar los resultados adquiridos y a partir de ello se plasma la comprobación de las hipótesis específicas siendo estadísticamente una herramienta para utilizar el IBM SPSS Statistics; con el objetivo de comprobar si la hipótesis es correcta de acuerdo a la comparación de los anteriores resultados (Pre Test) y posterior (Post Test).

Pruebas de Normalidad

Según Vilalta Perdomo (2016) sostiene al respecto que:

Para demostrar la normalidad por separada de las variables es la prueba de Shapiro-Wilk, encargado de conocer el grado de correspondencia distribuidos en un conjunto de valores en las cuales son las muéstrales y la distribución teórica específica. Es basado por el hecho de que la muestra es inferior a 50. Por ello se ha realizado observaciones y encuestas (2016, p.43).

2.5.1. Hipótesis de Investigación

Hipótesis General

- **Hipótesis H_0 :** La Aplicación Web Progresiva no mejora el proceso de facturación de la empresa Continental Travel S.A.C.
- **Hipótesis H_a :** La Aplicación Web Progresiva mejora el proceso de facturación de la empresa Continental Travel S.A.C.

Hipótesis Específica (HE_1)

Una Aplicación Web Progresiva disminuye el nivel de incidencias en emisiones en el proceso de facturación de la empresa Continental Travel S.A.C.

Definición de Variables:

NIE_a: Nivel de incidencias en emisiones antes de utilizar el Aplicación Web Progresiva.

NIE_d: Nivel de incidencias en emisiones después de utilizar el Aplicación Web Progresiva.

Hipótesis Estadística

Hipótesis Nula (H₀): La Aplicación Web Progresiva aumenta el nivel de incidencias en emisiones para el proceso de facturación de la empresa Continental Travel S.A.C.

$$H_0 : NIE_d > NIE_a$$

Se deduce que el indicador sin la Aplicación Web Progresiva es mejor que el indicador con la Aplicación Web Progresiva.

Hipótesis Alterna (H_a): La Aplicación Web Progresiva reduce el nivel de incidencias en emisiones para el proceso de facturación de la empresa Continental Travel S.A.C.

$$H_a : NIE_d \leq NIE_a$$

Se deduce que el indicador con la Aplicación Web Progresiva es mejor que el indicador sin la Aplicación Web Progresiva.

Hipótesis Especifica (HE₂)

Una Aplicación Web Progresiva incrementa el Margen de Utilidad Operativa en el proceso de facturación de la empresa Continental Travel S.A.C.

Definición de Variables:

MUO_a: Margen de Utilidad Operativa antes de utilizar el Aplicación Web Progresiva.

MUO_d: Margen de Utilidad Operativa después de utilizar el Aplicación Web Progresiva.

Hipótesis Estadística

Hipótesis Nula (H₀): La Aplicación Web Progresiva aumenta el Margen de Utilidad Operativa para el proceso de facturación de la empresa Continental Travel S.A.C.

$$H_0 : MUO_d > MUO_a$$

Se deduce que el indicador sin la Aplicación Web Progresiva es mejor que el indicador con la Aplicación Web Progresiva.

Hipótesis Alterna (H_a): La Aplicación Web Progresiva disminuye el Margen de Utilidad Operativa para el proceso de facturación de la empresa Continental Travel S.A.C.

$$H_a : MUO_d \leq MUO_a$$

Se deduce que el indicador con la Aplicación Web Progresiva es mejor que el indicador sin la Aplicación Web Progresiva.

Estadística (T-Student)

Se efectuará la prueba T- Student para la validación del pre test y post test lo cual permitirá a dos grupos diferentes su evaluación a través de la estadística, lo cual permitirá la comparación de la variable en estudio por medio de la medición.

Donde:

$$t = \frac{\bar{x} - \mu}{s/\sqrt{n}}$$

- μ = media de la población
- \bar{x} = media de la distribución de los datos
- n = tamaño de la muestra
- s = error estándar de la muestra

Región de rechazo

La región de rechazo es $t = t_x$

Donde t_x es tal que:

$P [t > t_x] = 0.05$, donde $t_x = \text{Valor Tabular}$

Luego Región de Rechazo: $t > t_x$

Cálculo de la Media:

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$$

Cálculo de la Varianza

$$S^2 = \frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n - 1}$$

Desviación Estándar

$$S^2 = \frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n - 1}$$

Donde:

\bar{x} = Media

δ^2 = Varianza

S^2 = Desviación Estándar

X_i = Dato i que está entre $(0, n)$

\bar{X} = Promedio de los datos

n = Número de datos

2.6. Aspectos éticos

El desarrollo de este estudio, como autor me comprometo a ser prudente y respetar en la claridad de la consecuencia obtenida y la información brindada por la empresa Continental Travel S.A.C.

Los antecedentes del presente estudio se obtuvieron de la empresa Continental Travel S.A.C. para ser procesados sin adulteraciones y de forma adecuada, debido son orientados estos datos para los instrumentos, por lo que se adjunta la carta de aceptación donde se confirma y autoriza la realización del estudio.

Toda la información del estudio fue citada con autenticidad, respetando la propiedad intelectual, reglamentos y lineamientos de la Universidad César Vallejo en la construcción de este proyecto de investigación. Además, respecto a las revistas, libros y otros documentos que sirvieron como fuente de conocimiento, se mantiene a través de las citas textuales y/o referencias bibliográficas a través del estándar ISO 690, con el objeto de proteger la información del autor.

III. RESULTADOS

3.1. Análisis Descriptivo

INDICADOR N° 1: Nivel de Incidencias en Emisiones

De acuerdo a la recolección de datos y posteriores resultados sobre el Nivel de Incidencias en Emisiones, se observan los resultados descriptivos en la siguiente Tabla.

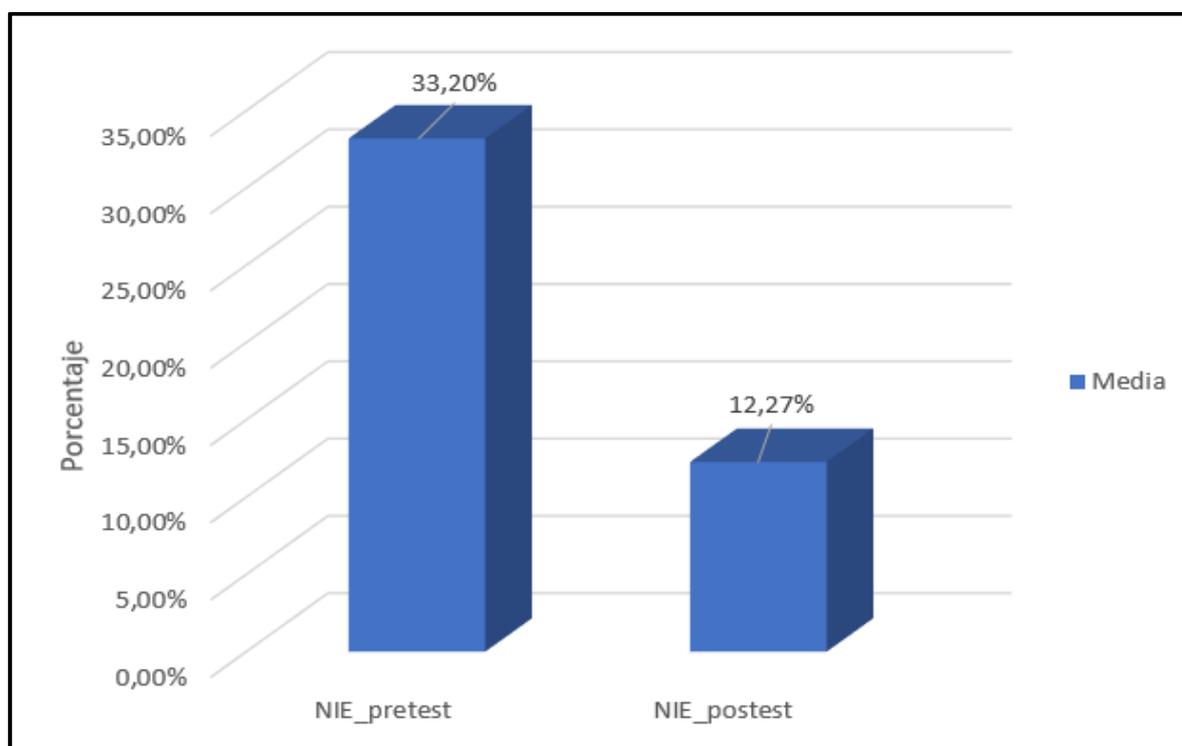
Tabla 10. Matriz Descriptiva del Indicador Nivel de Incidencias en Emisiones

	N	Mínimo	Máximo	Media	Desviación estándar
NIE_PRE	15	0,14	0,67	0,3320	0,15086
NIE_POST	15	0,00	0,33	0,1227	0,09498
N válido (por lista)	15				

Fuente: Elaboración propia

Por consiguiente, en la recolección y procesamiento de datos del pre-test el valor que se obtuvo es un 15.08%, en cambio en el análisis post-test fue de un 9,49% tal como se estima en la Figura N° 18; por lo que esto revela una mayor diferencia antes y después de la implementación de una Aplicación Web Progresiva; así mismo, se observa que el indicador propuesto tuvo antes una máxima de 67%, y después de la implementación de la Aplicación Web Progresiva tuvo un 33%.

Figura 18. Gráfico del Nivel de Incidencias en Emisiones antes y después de implementado la Aplicación Web Progresiva



Fuente: Elaboración propia

INDICADOR 2: Margen de Utilidad Operativo

De acuerdo a la recolección de datos y posteriores resultados sobre el Margen de Utilidad Operativo, se observan los resultados descriptivos en la siguiente Tabla.

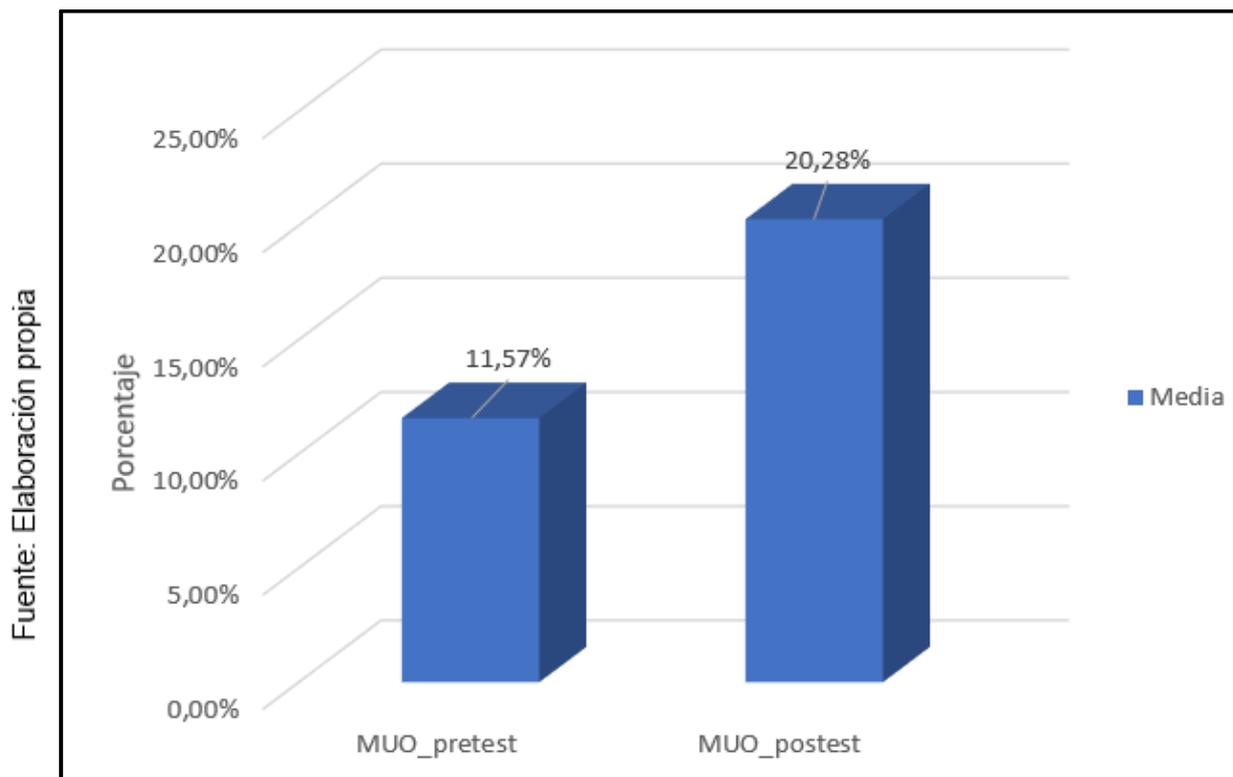
Tabla 11. Matriz Descriptiva del Margen de Utilidad Operativo

	N	Mínimo	Máximo	Media	Desviación estándar
MUO_PRE	15	3,91	22,49	11,5747	5,32848
MUO_POST	15	12,39	33,27	20,2827	6,05788
N válido (por lista)	15				

Fuente: Elaboración propia

Por consiguiente, en la recolección y procesamiento de datos del pre-test el valor que se obtuvo es un 11.57%, en cambio en el análisis post-test fue de un 20.28% tal como se estima en la Figura N° 19; por lo que esto revela una mayor diferencia antes y después de la implementación de una Aplicación Web Progresiva; así mismo, se observa que el indicador propuesto tuvo antes una mínima de 3.91%, y después de la implementación de la Aplicación Web Progresiva tuvo un 12.39%.

Figura 19. Gráfico del Margen de Utilidad Operativa generados antes y después de implementado la Aplicación Web Progresiva



3.2. Análisis Inferencial

Prueba de Normalidad

Para la realización de las pruebas de normalidad para los indicadores propuestos mediante la prueba Shapiro-Wilk, el tamaño de la muestra está constituido por 15 días de registros de comprobantes de pago a lo largo de un mes.

INDICADOR N° 1: Nivel de Incidencias en Emisiones

Tabla 12. Pruebas de Shapiro Wilk sobre el Nivel de Incidencias en Emisiones

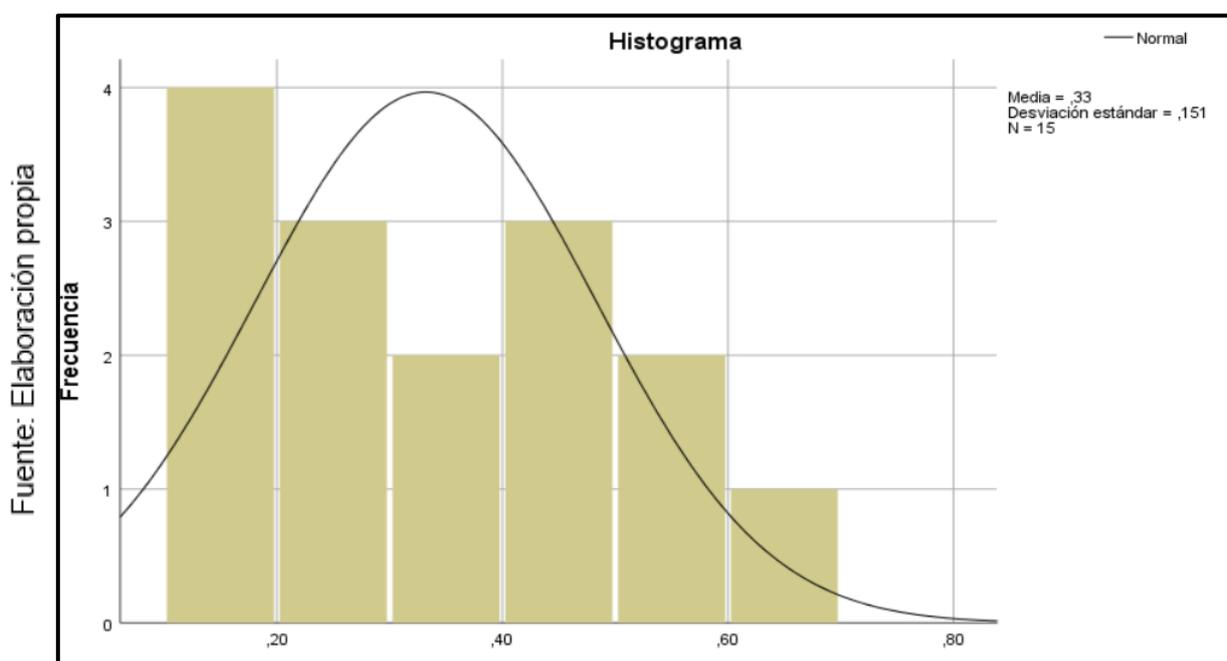
Prueba de Normalidad : Shapiro Wilk			
	Estadístico	gl	Sig.
NIE_PRE	,936	15	,332
NIE_POST	,909	15	,129
N válido (por lista)			

Fuente: Elaboración propia

Por consiguiente, de acuerdo al análisis de la Tabla N° 12, se obtuvo lo siguiente:

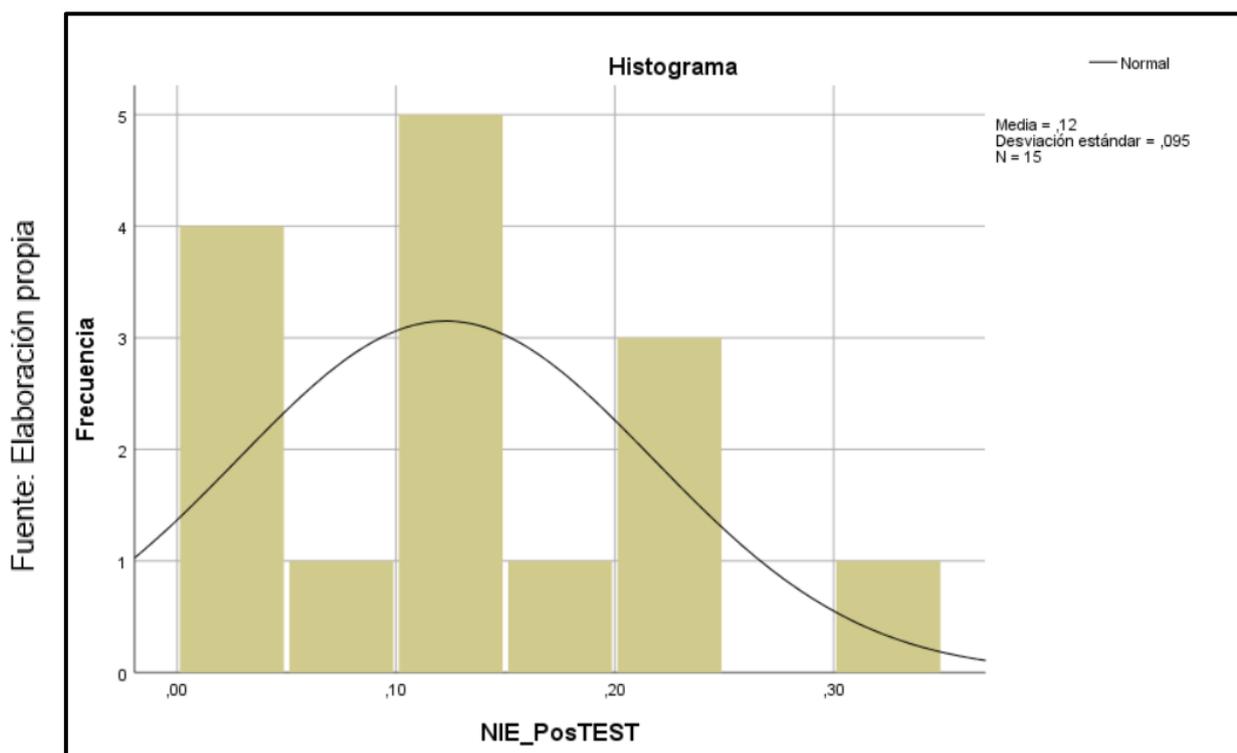
- De acuerdo, a los resultados del método pre-test fue de un 0,332, cuyo valor es mayor que un 0.05, lo cual revela que el indicador propuesto presenta una distribución normal.
- Asimismo, según los resultados del método post -test fue de un 0,129, siendo un valor mayor que un 0.05, lo cual revela que el indicador propuesto se distribuye normalmente.

Figura 20. Histograma del Pre-Test: Nivel de Incidencias en Emisiones



Fuente: Elaboración propia

Figura 21. Histograma del Post-Test: Nivel de Incidencias en Emisiones



INDICADOR N° 2: Margen de Utilidad Operativa

Tabla 13. Pruebas de Shapiro Wilk del Margen de Utilidad Operativa

Prueba de Normalidad : Shapiro Wilk			
	Estadístico	gl	Sig.
MUO_PRE	,907	15	,122
MUO_POST	,950	15	,524
N válido (por lista)			

Fuente: Elaboración propia

Por consiguiente, de acuerdo al análisis de la Tabla N° 13, se obtuvo lo siguiente:

- De acuerdo, a los resultados del método pre-test fue de un 0,122, cuyo valor es mayor que un 0.05, lo cual revela que el indicador presenta una distribución normal.
- Asimismo, según los resultados del método post-test fue de un 0,524, siendo un valor mayor que un 0.05, lo cual revela que el indicador propuesto se distribuye normalmente.

Figura 22. Histograma del Pre-Test: Margen de Utilidad Operativa

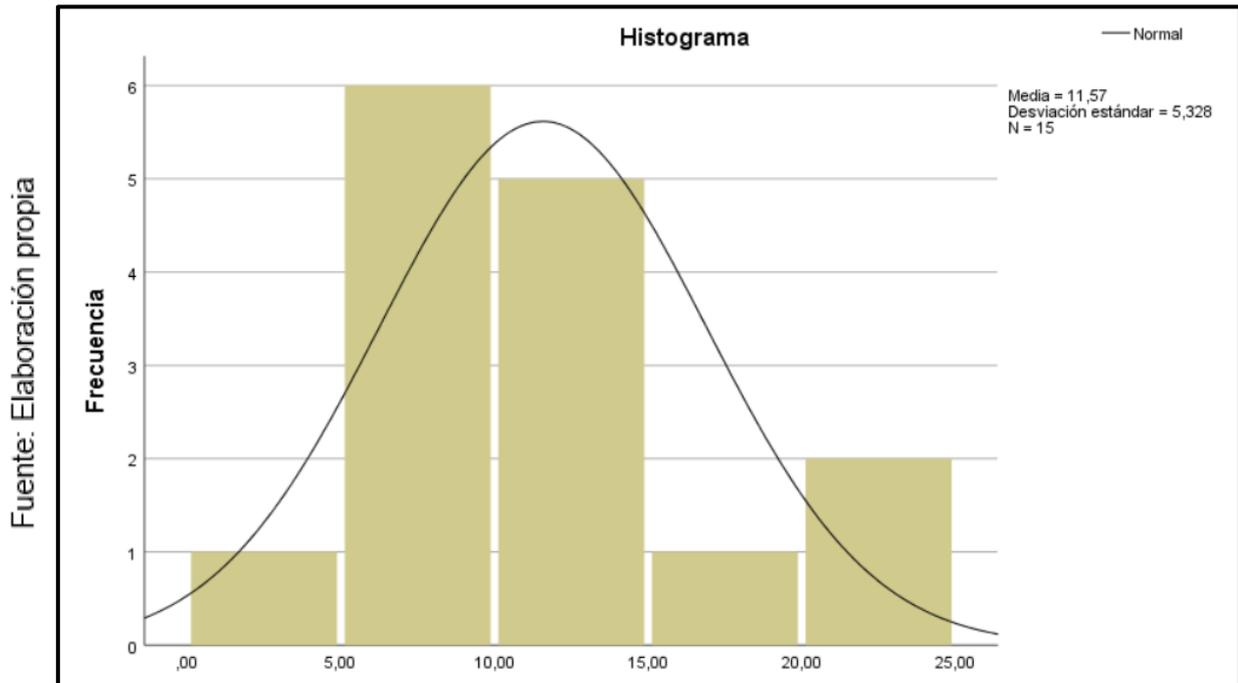
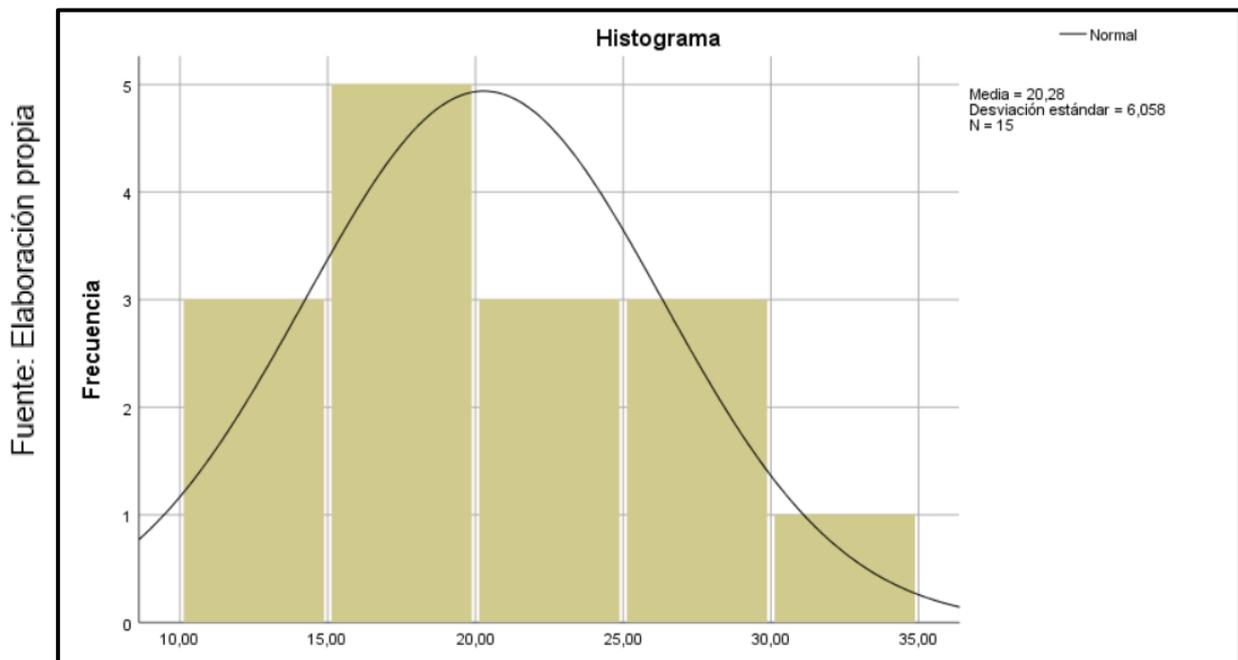


Figura 23. Histograma del Post-Test: Margen de Utilidad Operativa



3.3. Prueba de Hipótesis

Hipótesis de Investigación N° 1

- H1: Una Aplicación Web Progresiva disminuye el Nivel de Incidencias en Emisiones en el proceso de Facturación de la empresa Continental Travel S.A.C

Definiciones de variables

NIEa: El Nivel de Incidencias en Emisiones antes de utilizar el sistema.

NIEd: El Nivel de Incidencias en Emisiones después de utilizar el sistema.

Hipótesis Estadísticas

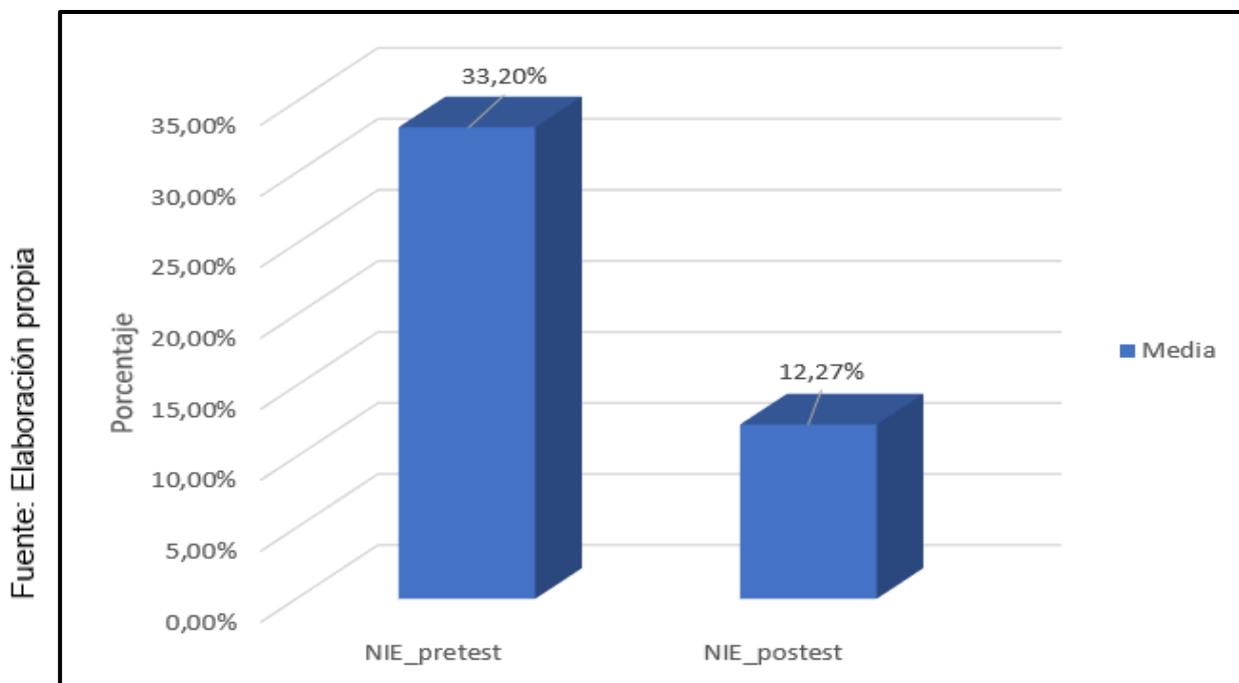
H0: Una Aplicación Web Progresiva no incrementa el Nivel de Incidencias en Emisiones en el proceso de Facturación de la empresa Continental Travel S.A.C

H0: NIEa \geq NIEd

HA: Una Aplicación Web Progresiva incrementa Nivel de Incidencias en Emisiones en el proceso de Facturación de la empresa Continental Travel S.A.C

H0: NIEa \leq NIEd

Figura 24: Grafico Comparativo - Nivel de Incidencias en emisiones



Por consiguiente, se concluye que el indicador con la Aplicación Web Progresiva es mejor que el indicador sin la Aplicación Web Progresiva; asimismo se observa un descenso en la Figura N° 24, por lo que indica que el indicador propuesto en el análisis del Pre-Test, obtuvo un 33.20% y posteriormente en el Post-Test desciende a un 12.27%.

Tabla 14. Pruebas de T-Student del Nivel de Incidencias en Emisiones

Prueba T-Student					
	Desviación	Media	t	gl	Sig.(bilateral)
NIE_PRE – NIE_POST	0,160778	0,3320	5,043	14	0.00
		0,1227			

Fuente: Elaboración propia

Por otro lado, en el resultado de contraste de la hipótesis propuesta se aplicó la Prueba T-Student, donde el valor de T contraste es de 5,043, indicando que es notoriamente menor que 1.7613 de acuerdo a la clasificación T-Student.

Análisis T-Student del Nivel de Incidencias en Emisiones

$$T_c = \frac{x - u}{S / \sqrt{n}}$$

$$T_c = \frac{0,3320 - 0,122667}{0,160778 \sqrt{15}}$$

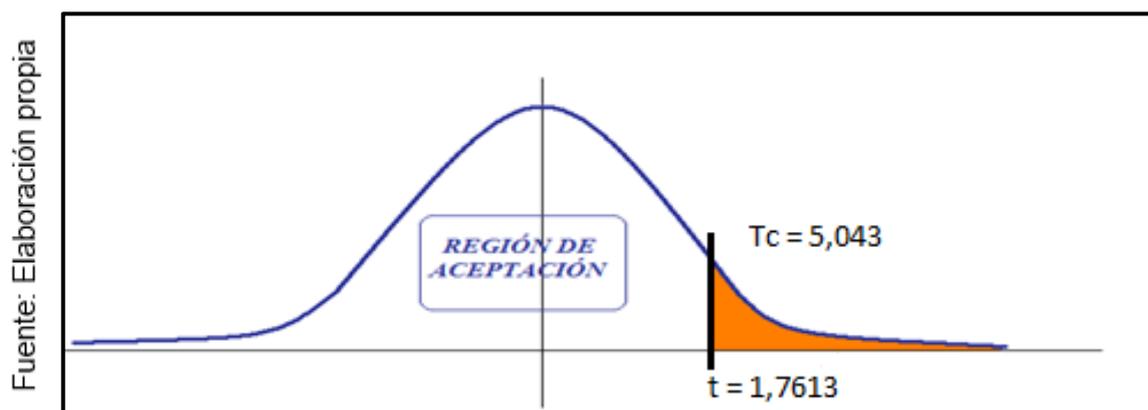
$$T_c = \frac{0,3320 - 0,122667}{0,160778 / 3,872983}$$

$$T_c = \frac{0,209333}{0,04151}$$

$$T_c = 5.04295$$

En suma, al análisis de los resultados se rechaza la hipótesis nula, aceptando la hipótesis alterna con un 95% de confianza, y el valor obtenido Tc se posiciona en la zona de rechazo como los indica la Figura N° 25; se concluye que la Aplicación Web Progresiva disminuye el Nivel de Incidencias en Emisiones en el Proceso de Facturación de la empresa Continental Travel S.A.C.

Figura 25. Grafica T-Student – Nivel de Incidencias en Emisiones



Hipótesis de Investigación N° 2

- H1: Una Aplicación Web Progresiva incrementa el Margen de Utilidad Operativa en el proceso de Facturación de la empresa Continental Travel S.A.C

Definiciones de variables:

MUOa: El Margen de Utilidad Operativa antes de utilizar el sistema.

MUOd: El Margen de Utilidad Operativa en después de utilizar el sistema.

Hipótesis Estadísticas:

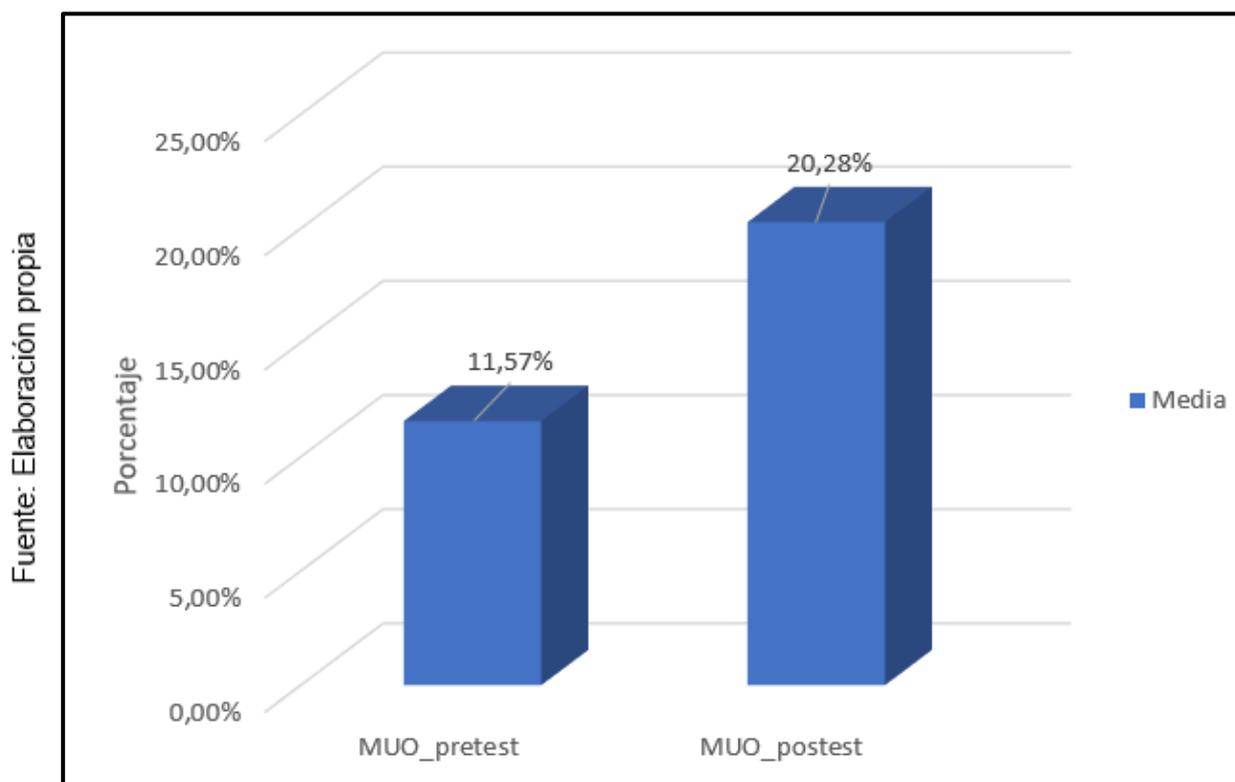
H0: Una Aplicación Web Progresiva no incrementa el Margen de Utilidad Operativa en el proceso de Facturación de la empresa Continental Travel S.A.C

$$H0: NIEa \geq NIEd$$

Ha: Una Aplicación Web Progresiva incrementa el Margen de Utilidad Operativa en el proceso de Facturación de la empresa Continental Travel S.A.C

$$H0: NIEa < NIEd$$

Figura 26: Grafico Comparativo - Margen de Utilidad Operativa



Por consiguiente, se concluye que el indicador con la Aplicación Web Progresiva es mejor que el indicador sin la Aplicación Web Progresiva; asimismo se observa un ascenso en la Figura N° 26, por lo que indica que el indicador propuesto en el análisis del Pre-Test, obtuvo un 11.57% y posteriormente en el Post-Test asciende a un 20.28%.

Tabla 15. Pruebas de T-Student del Margen de Utilidad Operativa

Prueba T-Student					
	Desviación	Media	t	gl	Sig.(bilateral)
MUO_PRE – MUO_POST	6,39942	11,5747	-5,270	14	0.000
		20,2827			

Fuente: Elaboración propia

Por otro lado, en el resultado de contraste de la hipótesis propuesta se aplicó la Prueba T-Student, donde el valor de T contraste es de -5,270, lo cual indica que es claramente menor que -1.7613 de acuerdo a la clasificación T-Student.

Análisis T-Student del Nivel de Incidencias en Emisiones

$$T_c = \frac{x - u}{S / \sqrt{n}}$$

$$T_c = \frac{11,5747 - 20,2827}{6,39942 \sqrt{15}}$$

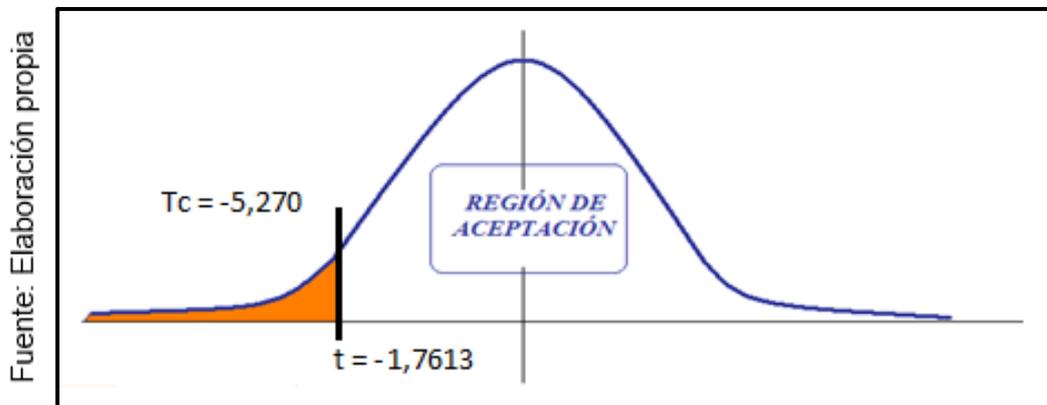
$$T_c = \frac{11,5747 - 20,2827}{6,39942 / 3,872983}$$

$$T_c = \frac{-8,708}{1,6523}$$

$$T_c = -5.27022$$

En suma, al análisis de los resultados se rechaza la hipótesis nula, aceptando la hipótesis alterna con un 95% de confianza, y el valor T_c obtenido se posiciona en la zona de rechazo como lo indica la Figura N° 27; se concluye que la Aplicación Web Progresiva incrementa el Margen de Utilidad Operativa en el Proceso de Facturación de la empresa Continental Travel S.A.C.

Figura 27. Grafica T-Student – Margen de Utilidad Operativa



I. DISCUSIÓN

De acuerdo a la investigación propuesta, la Aplicación Web Progresiva tuvo como resultado el descenso del Nivel de Incidencias en emisiones en el Proceso de Facturación de un 33,20% a un 12.27%, lo que representa una disminución de 20.93%.

Asimismo, Navarro Theany, en la investigación denominada “Sistema de facturación electrónica para la gestión de comprobantes de pago basado en ISO/IEC19845:2015 en Acgenesys S.A.C.”, se concluyó que la aplicación de un Sistema Web disminuyó la Tasa de entregas incompletas de 45.13% a un 19.97%.

Además, la Aplicación Web Progresiva tuvo como resultado el aumento en el Margen de Utilidad Operativa en el Proceso de Facturación de 11,57% a obtener un 20.28%, lo que representa a un crecimiento de 8,71%.

Asimismo, Araujo Andrés, en su presente investigación denominada “Sistema Web para el proceso comercial en la empresa Equisoft Perú S.A.C”, se determinó que la aplicación de un Sistema Web incrementó el Índice de Efectividad de Cotizaciones aumento de 12.62% a un 37,7 % de lo que equivale a un crecimiento de 25.08%.

Por consiguiente, según los resultados se comprueba que el uso de la Aplicación Web Progresiva como herramienta tecnológica, mejora el proceso de facturación en la empresa Continental Travel S.A.C. logrando disminuir el Nivel de Incidencias en Emisiones en un 20,93% e incrementa el Margen de Utilidad Operativa en un 8.71%.

II. CONCLUSIONES

PRIMERO: La Aplicación Web Progresiva disminuyó el Nivel de Incidencias en Emisiones en 20,93%. Teniendo inicialmente un 33,20% y posteriormente un 12,27%; además en el análisis de resultado sobre la hipótesis se realizó la aplicación de una Prueba TStudent, donde se alcanzó un valor de T contraste de 5,043, el cual es claramente mayor que la región de rechazo de 1.7613. Por consiguiente, en el Proceso de Facturación se afirma que una Aplicación Web Progresiva disminuyó el Nivel de Incidencias en Emisiones.

SEGUNDO: La Aplicación Web Progresiva incremento el Margen de Utilidad Operativa en 8,71%. Teniendo inicialmente un 11,57% y posteriormente un 20,28%; además en el análisis de resultado sobre la hipótesis se realizó la aplicación de una Prueba TStudent, donde se alcanzó un valor de T contraste de -5,270, lo cual es menor que la región de rechazo de -1.7613. Por consiguiente, en el Proceso de Facturación se afirma que una Aplicación Web Progresiva incremento el Margen de Utilidad Operativa.

TERCERO: Por último, se concluyó que a partir de los resultados obtenidos de los indicadores propuestos, la influencia de la Aplicación Web Progresiva mejora el Proceso de Facturación en la empresa Continental Travel S.A.C.

III. RECOMENDACIONES

Para incentivar el continuo desarrollo del instrumento y contexto de la investigación del presente proyecto, se propone las recomendaciones siguientes:

- Es imperativo recibir las capacitaciones en relación uso apropiado de la Aplicación PWA a todo personal implicado al uso del software propuesto en el presente proyecto.
- En cuanto a el software propuesto, se recomienda a largo plazo, mejorar la Aplicación PWA mediante versiones que apliquen actualizaciones sobre mejoras y/o mantenimiento del software; así también como procesos relacionados a la gestión de comprobantes de pago.

Finalmente, recomendamos ejecutar investigaciones relacionadas a la investigación propuesta, que conlleven a la mejora del proceso que involucra la gestión del comprobante de pago como los procesos de venta y la Generación de Tickets de Servicio que procesa la empresa Continental Travel S.A.C.

REFERENCIA BIBLIOGRÁFICAS

- ARIAS, Fidias. El proyecto de Investigación: Introducción a la metodología científica. Venezuela: Episteme, 2016. ISBN: 9800785299
- ARAUJO, Andrés. Sistema Web para el proceso comercial en la empresa Equisoft Perú S.A.C. Tesis (Pregrado). Perú: Universidad Cesar Vallejo, 2017.
- BENJAMIN, Pessi. The Impact of Implementation of the Electronic Purchase Invoice System on a Company on the Exmple of Hahle Group [en línea]. Tesis. Estonia: Tallinn University of Technology, 2015. Disponible en: <https://digi.lib.ttu.ee/i/file.php?DLID=9591&t=1>
- BEUD. Itil® v3. Barcelona: Ediciones Eni, 2016. ISBN: 9782409001789
- BUSTAMANTE, Ysela y PACHECO, Flor. Incidencias: Ventajas y Desventajas de la Implementación del Sistema de Emisión Electrónica frente al Sistema de Emisión Física en principales contribuyentes en la ciudad de Arequipa 2018 [en línea]. Tesis (Pregrado). Perú: Universidad Católica San Pablo, 2018. Disponible en: http://repositorio.ucsp.edu.pe/bitstream/UCSP/15616/1/BUSTAMANTE_AQUINO_YSE_INC.pdf
- CARDADOR, A. Implementación de aplicaciones web en entorno internet, intranet y extranet IFCD210. 2014. ISBN: 9788416433094
- CATAORA. F. (2015). Sistemas y Procedimientos Contables. (4ta edición) Editorial Mc Graw Hill Interamericana de Venezuela
- CEGARRA, José. Los métodos de investigación. 1a. ed. Madrid: Ediciones Díaz de Santos. 2015. ISBN: 9788499693910
- CHANG, C., CHIN-JUI, K., HUAI-CHIEN, C., CHEN-YUAN, C., TSUNG-HAO, C. and PEI-YIN, C. Ergonomic Techniques for a Mobile E-Invoice System: Operational Requirements of an Information Management System. Human Factors and Ergonomics in Manufacturing, Nov, 2014, vol. 23, no. 6. pp. 582 ProQuest Central. ISSN 10908471. Disponible en: <http://repositorio.unsm.edu.pe/bitstream/handle/UNSM/2676/CONTABILIDAD%20-%20Danny%20Daniel%20C%C3%B3rdova%20Pe%C3%B1a%20%26%20Karen%20Vanesa%20Nu%C3%B1ez%20Uriarte.pdf?sequence=1&isAllowed=y>

- CUYLEN, A., KOSCH, L. and BREITNER, M.H., 2016. Development of a Maturity Model for Electronic Invoice Processes. *Electronic Markets*, 05, vol. 26, no. 2, pp. 115-127 ProQuest Central. ISSN 10196781.
- DE LA TORRE. *Agencia de Viajes: Estructura y Operación*, Mexico:2014. ISBN: 978849236236
- DIARIO EL COMERCIO. Factura electrónica: ¿Sabías que el Perú lidera su uso en la región? [en línea], 2018. Disponible en: <https://elcomercio.pe/publiirreportaje/factura-electronica-sabias-peru-lidera-region-noticia-512837>
- EL SERVICIO DE AYUDA A DOMICILIO (SAD). *Apoyo domiciliario y alimentación familiar* [en línea], 2015. Disponible en: <https://books.google.com.pe/books?id=PvQwloWvd9MC&printsec=frontcover#v=onepage&q&f=false>
- ESCORSA Pere. y VALLS Jaume. *Tecnología e innovación en la empresa* [en línea], 2016. Disponible en: <https://www.gestiopolis.com/sistemas-de-informacion-en-las-organizaciones/>
- GABILLAUD, Jérôme. *SQL Server 2014: Administración de una base de datos transaccional con SQL Server Management Studio* [en línea], 2014. Disponible en: <https://books.google.com.pe/books?id=LvPWdGufkboC&printsec=frontcover#v=onepage&q&f=false>
- GARCÍA, Juan. *Costo beneficio del uso de la factura electrónica en los Principales contribuyentes de la Región San Martín - Tarapoto, Perú: Universidad Nacional de San Martín, Facultad de Ingeniería Agroindustrial*, 2014. GoodBarber. *Progressive Web Apps* [En línea], 2018. Disponible en: https://es.goodbarber.com/pwa/create/?gclid=Cj0KCQjwgOzdBRDIARIsAJ6_HNnX869KRv5yXz9FIA6Ez1BgGHBg2DET_WVqAZbLZHHskfk49j3WxEkaAqh bEALw_wcB
- GOOGLE PARTNERS. *PWA, The Progressive Web Apps* [En línea], 2018. Disponible en: <https://support.google.com/partners/answer/7336532?hl=es-419>
- GRAPECITY. *Component One Studio Enterprise* [en línea], 2019. Disponible en: <https://www.grapecity.com/dmn/c1-focus->

3b?utm_source=Adwords&utm_medium=CPC&utm_campaign=C1-Brand&utm_term=Componentone&gclid=CjwKCAiA5o3vBRBUEiwA9PVzaqJoYZwpcFVARJ__5Ht5jSHZJY-VW65dS7LkR2xLUHJ-xVjKIYoyyhoC0CoQAvD_BwE&utm_expId=.xRmYrFvjRF6vrq324Nlg3w.0&utm_referrer=https%3A%2F%2Fwww.google.com%2F

- HERNANDEZ, Roberto, FERNANDEZ, Carlos y BAPTISTA, Pilar. Metodología de la Investigación (5a). México: McGraw Hill, 2014. ISBN: 978-607-15-0291-9
- HUIDOBRO, José y CONCESA, Rafael. Redes de banda ancha. España: S.A. MCGRAW-HILL / Interamericana de España, 2015. ISBN: 9788448140267
- GÓMEZ, Alberto y DE ABAJO, Nicolas. Los Sistemas de Información en la Empresa. Oviedo: Universidad de Oviedo, 2015. ISBN: 978-8483-1703-6-6.
- Instituto Americano de Ciencias Agrícolas de la OEA (IICA). Manual de procedimientos administrativos [en línea], 2016. Disponible en: <https://books.google.com.pe/books?id=aeEOAQAIAAJ&printsec=frontcover#v=onepage&q&f=false>
- LE, Ha Giang. Adoption of E-invoicing In Vietnam Case: Electricity of Vietnam Group, Danang Power Company [en línea]. Tesis. Vietnam: Lahti University of Applied Sciences, 2018. Disponible en: https://www.theseus.fi/bitstream/handle/10024/144072/Giang_Le_Ha.pdf?sequence=2&isAllowed=y
- LOZADA, José. Investigación Aplicada: Definición, propiedad intelectual e Industria. Ecuador: Universidad Tecnología Indoamerica, 2014. ISBN 1390-9592
- MICROSOFT. Las aplicaciones web progresivas [En línea], 2018. Disponible en: <https://docs.microsoft.com/es-es/microsoft-edge/progressive-web-apps/get-started>
- MACEDA, Humberto, CASTRO, Luis y VELASCO, Perla. Arquitectura de Software, México: McGraw Hill, 2015. ISBN: 9786075224565
- LEURO, Mauricio y OVIEDO, Irsa. Facturación & auditoría de cuentas en salud (5ta) México: EcoEdiciones, 2016. ISBN: 9789587712964
- MENDOZA, Henry y BAUTISTA, Gloria. Bioestadística Fundamental. Colombia: Universidad Nacional de Colombia. 2015.

- MICROSOFT PRESS. Inside Microsoft Visual Studio .NET [en línea], 2016. Disponible en: <https://books.google.com.pe/books?id=Ra1-sXvTly4C&printsec=frontcover#v=onepage&q&f=false>
- NAVARRO, Theany. Sistema de facturación electrónica para la gestión de comprobantes de pago basado en ISO/IEC19845:2015 en Acgenesys S.A.C. Tesis (Pregrado). Perú: Universidad Cesar Vallejo, 2017.
- ORDAYA, Rita. Implementación de un Sistema de Información para una MYPE Comercial con componentes de Libros y Facturación electrónica [en línea]. Tesis (Pregrado). Perú: Universidad Católica del Perú, 2015. Disponible en: http://tesis.pucp.edu.pe/repositorio/bitstream/handle/123456789/6852/ORDAYA_RITA_SISTEMA_MYPE_COMERCIAL_LIBROS_FACTURACION_ELECTRONICA.pdf?sequence=1&isAllowed=y
- DE PABLOS, Carmen. Dirección y gestión de los sistemas de información en la empresa: Una visión integradora. Madrid. 2015. ISBN: 84-7356-445-6
- RA-MA. Calidad de Sistema de Información. México: 2015. ISBN: 9923523523623
- ROMERO, María José y VALLESPÍN, Arán María. UF0079: Gestión de sistemas de distribución global(GDS), 2014.
- SCRUM MANAGER BOOK. Historia de usuario, 2014. Disponible en: http://www.scrummanager.net/bok/index.php/Historia_de_usuario/
- SERES. Situación de la Factura Electrónica, Grado de Implantación en Europa [en línea], 2018. Disponible en: https://www.economiadehoy.es/adjuntos/33662/SERES_-_Situacion_Factura_Electronica_en_Europa_2018.pdf
- SOLIDGEARGROUP. Progressive Web Apps [En línea], 2018. Disponible en: <https://solidgeargroup.com/pwa-la-nueva-generacion-de-aplicaciones-web?lang=es>
- SUNAT. Facturación Electrónica [En línea],2018. Disponible en: <http://orientacion.sunat.gob.pe/index.php/empresas-menu/comprobantes-de-pago-empresas/comprobantes-de-pago-electronicos-empresas/see-desde-los-sistemas-del-contribuyente/2-comprobantes-que-se-pueden-emitir-desde-see-sistemas-del-contribuyente/factura-electronica-desde-see-del-contribuyente>

- TALLEDO, J. Implantación de aplicaciones web en entorno internet, intranet y extranet. 2015. España: Ediciones Paraninfo S.A. ISBN: 978-84-283-9734-6
- TAMAYO, Mario. El proceso de la investigación científica. México: Limusa, 2014. ISBN: 9681858727
- TORRES, Raul. Las Progressive Web Apps [En línea],2018. Disponible en: https://comenzandodecero.com/que-son-las-pwa-progressive-web-apps/#Importancia_de_las_PWA_en_el_moldeado_futuro_de_la_web
- TUGUME, Howard; KOBUSINGE, Justine y NANTEZA, Justine. The Physical and Electronic Payment Interface and its Influence on Consumer Payment Choices and Informal/Fraudulent Practices: A Case Study of the National Water and Sewerage Corporation (Nwsc) Uganda [en línea]. Tesis. Estados Unidos: Institute for Money, Technology & Financial Inclusion of University of California, 2015. Disponible en: <https://www.imtfin.uci.edu/files/docs/2015/Howard%20Tugume%20Final%20Report.pdf>
- TSINGHUA UNIVERSITY. "Electronic Invoice Issuing System for Electronic Commerce Website" in Patent Application Approval Process. Information Technology Newsweekly, Mar 11, 2014. pp. 307 ProQuest Central. ISSN 1944-1797.
- VALDERRAMA, Santiago. Pasos para elaborar Proyectos y tesis de investigación científica. 1a. ed. Lima: Editorial San Marcos, 2015. ISBN: 9786123028787
- WENJING Yang. T Analysis on Online Payment Systems of E-Commerce [en línea]. Tesis. Finlandia: Oulu University of Applied Sciences, 2017. Disponible en: https://www.theseus.fi/bitstream/handle/10024/139600/Yang_Wenjing.pdf?sequence=1
- ZULEIMA Flores. Estrategia para mejorar el proceso de facturación de la empresa Comercializadora Trapecio, C.A [En línea],2016. Disponible en: <https://bibliovirtualujap.files.wordpress.com/2013/05/teg-zuleima-elisa-flores-romero.pdf>

ANEXOS

Anexo N° 1: Modelado de la Matriz de Consistencia de la Investigación

PROBLEMAS	OBJETIVOS	HIPÓTESIS	VARIABLES	DIMENSIONES	INDICADOR	METODOLOGÍA
<p>Principal</p> <p>PG: ¿Cómo influye una Aplicación Web Progresiva en el Proceso de Facturación en la empresa Continental Travel S.A.C.?</p>	<p>General</p> <p>OG: Determinar la influencia de una Aplicación Web Progresiva en el Proceso de Facturación en la empresa Continental Travel S.A.C.</p>	<p>General</p> <p>HG: Una Aplicación Web Progresiva mejora el Proceso de Facturación en la empresa Continental Travel S.A.C.</p>	<p>Independiente</p> <p>X1: Aplicación Web Progresiva</p>			<p>Método de Investigación: Hipotético-Deductivo</p> <p>Tipo de Investigación: Aplicada</p> <p>Diseño de la Investigación: Pre-Experimental</p>
<p>Secundario</p> <p>P1: ¿Cómo influye una Aplicación Web Progresiva en el Nivel de Incidencias de Emisiones en el Proceso de Facturación en la empresa Continental Travel S.A.C.?</p>	<p>Específico</p> <p>O1: Determinar la influencia de una Aplicación Web Progresiva en el Nivel de Incidencias de Emisiones en el Proceso de Facturación en la empresa Continental Travel S.A.C.</p>	<p>Específicas</p> <p>H1: Una Aplicación Web Progresiva disminuye el Nivel de Incidencias en Emisiones en el Proceso de Facturación en la empresa Continental Travel S.A.C.</p> <p>H1: Una Aplicación Web Progresiva aumenta el Margen de Utilidad Operativa en el Proceso de Facturación en la empresa Continental Travel S.A.C.</p>	<p>Dependientes</p> <p>Y1: Proceso de Facturación</p>	<p>D1: Liquidación y elaboración de la factura</p>	<p>I1: Nivel de Incidencias en emisiones</p> <p>I2: Margen de Utilidad Operativa</p>	<p>Población Está conformada por 151 comprobantes de pago emitidos en el mes de Mayo de 2019</p> <p>MUESTRA El tamaño de la muestra está conformado por 108 comprobantes de pago emitidos durante el mes de Mayo de 2019, estratificadas en 15 días mediante el muestreo estratificado proporcional.</p> <p>TÉCNICAS E INSTRUMENTOS Entrevista: - Cuestionario Observación: - Ficha de Registro</p>

Anexo 2: Ficha Técnica del Instrumento (Formato)

Ficha de Registro			
Investigador	Orvic Yampool Flores Principe		
Empresa	Continental Travel S.A.C		
Tema de Investigación	Aplicación Web Progresiva en el Proceso de Facturación de la empresa Continental Travel S.A.C.		
Fecha Inicio	XX/XX/2019	Fecha Final	XX/XX/2019

Elección de Técnica de Instrumento			
Dimensión	Indicador	Técnica	Instrumento
Liquidación y elaboración de la Factura	Nivel de Incidencias en Emisiones	Fichaje	Ficha de Registro
	Margen de Utilidad Operativa	Fichaje	Ficha de Registro

Anexo N° 3: Fichas para el Registro del Instrumento

Ficha de Registro – Pre Test			
Investigador	Orvic Yampool Flores Principe		
Empresa	Continental Travel S.A.C		
Tema de Investigación	Aplicación Web Progresiva en el proceso de facturación de la empresa Continental Travel S.A.C.		
Fecha Inicio	01/05/2019	Fecha Final	31/05/2019

Indicador	Nivel de Incidencias en emisiones (NIE)		
Fórmula	$NIE = \frac{TA}{TE}$ <p>Dónde: NIE: Nivel de incidencias en emisiones. TA: Total de documentos anulados. TE: Total de documentos emitidos.</p>		

ITEM	FECHA EMISIÓN	NÚMERO DE COMPROBANTES EMITIDOS		NIE
		EMITIDAS	ANULADAS	
1	03/05/2019	3	1	0.33
2	04/05/2019	4	2	0.50
3	05/05/2019	7	2	0.29
4	06/05/2019	12	5	0.42
5	07/05/2019	7	1	0.14
6	08/05/2019	13	4	0.31
7	10/05/2019	10	2	0.20
8	12/05/2019	16	3	0.19
9	13/05/2019	3	2	0.67
10	14/05/2019	5	2	0.40
11	15/05/2019	6	1	0.17
12	17/05/2019	5	2	0.40
13	18/05/2019	7	2	0.29
14	20/05/2019	6	1	0.17
15	21/05/2019	4	2	0.50


 CONTINENTAL TRAVEL S.A.C.
 AGENCIA DE VIAJES Y TURISMO

Ficha de Registro – Re Test			
Investigador	Orvic Yampool Flores Principe		
Empresa	Continental Travel S.A.C		
Tema de Investigación	Aplicación Web Progresiva en el proceso de facturación de la empresa Continental Travel S.A.C.		
Fecha Inicio	01/06/2019	Fecha Final	30/06/2019

Indicador	Nivel de Incidencias en emisiones (NIE)		
Fórmula	$NIE = \frac{TA}{TE}$		Dónde: NIE: Nivel de incidencias en emisiones. TA: Total de documentos anulados. TE: Total de documentos emitidos.

ITEM	FECHA EMISIÓN	NÚMERO DE COMPROBANTES EMITIDOS		NIE
		EMITIDAS	ANULADAS	
1	01/06/2019	18	5	0.28
2	02/06/2019	7	3	0.43
3	03/06/2019	6	2	0.33
4	04/06/2019	7	3	0.43
5	05/06/2019	16	3	0.19
6	06/06/2019	4	2	0.50
7	08/06/2019	3	1	0.33
8	09/06/2019	7	2	0.29
9	10/06/2019	2	2	1.00
10	11/06/2019	10	4	0.40
11	12/06/2019	3	1	0.33
12	13/06/2019	4	2	0.50
13	15/06/2019	8	3	0.38
14	16/06/2019	5	2	0.40
15	17/06/2019	8	3	0.38


CONTINENTAL TRAVEL S.A.C.
AGENCIA DE VIAJES Y TURISMO

Ficha de Registro – Post Test			
Investigador	Orvic Yampool Flores Principe		
Empresa	Continental Travel S.A.C		
Tema de Investigación	Aplicación Web Progresiva en el proceso de facturación de la empresa Continental Travel S.A.C.		
Fecha Inicio	01/10/2019	Fecha Final	31/10/2019

Indicador	Nivel de Incidencias en emisiones (NIE)		
Fórmula	$NIE = \frac{TA}{TE}$	Dónde: NIE: Nivel de incidencias en emisiones. TA: Total de documentos anulados. TE: Total de documentos emitidos.	

ITEM	FECHA EMISIÓN	NÚMERO DE COMPROBANTES EMITIDOS		NIE
		EMITIDAS	ANULADAS	
1	07/10/2019	9	0	0.00
2	08/10/2019	7	1	0.14
3	09/10/2019	10	2	0.20
4	10/10/2019	5	1	0.20
5	11/10/2019	5	0	0.00
6	14/10/2019	3	1	0.33
7	15/10/2019	5	0	0.00
8	16/10/2019	7	1	0.14
9	17/10/2019	13	1	0.08
10	18/10/2019	12	2	0.17
11	19/10/2019	8	1	0.13
12	21/10/2019	9	1	0.11
13	22/10/2019	3	0	0.00
14	24/10/2019	7	1	0.14
15	26/10/2019	5	1	0.20


CONTINENTAL TRAVEL S.A.C.
AGENCIA DE VIAJES Y TURISMO

Ficha de Registro – Pre Test			
Investigador	Orvic Yampool Flores Principe		
Empresa	Continental Travel S.A.C		
Tema de Investigación	Aplicación Web Progresiva en el proceso de facturación de la empresa Continental Travel S.A.C.		
Fecha Inicio	01/05/2019	Fecha Final	31/05/2019

Indicador	Margen de Utilidad Operativa (MUO)		
Fórmula	$MUO = \frac{UO}{IT} * 100$ <p>Dónde: MUO: Margen de Utilidad Operativa. UO: Utilidad Operativa. IT: Ingresos Totales</p>		

ITEM	FECHA EMISIÓN	COMPROBANTES EMITIDOS		MUO
		INGRESOS TOTALES	UTILIDAD OPERATIVA	
1	03/05/2019	570.05	78.95	13.85
2	04/05/2019	1,408.10	157.90	11.21
3	05/05/2019	930.34	99.66	10.71
4	06/05/2019	6,700.28	354.72	5.29
5	07/05/2019	1,025.89	101.69	9.91
6	08/05/2019	3,205.99	264.01	8.23
7	10/05/2019	6,045.89	517.55	8.56
8	12/05/2019	2,503.75	547.58	21.87
9	13/05/2019	3,438.56	298.56	8.68
10	14/05/2019	1,059.26	158.68	14.98
11	15/05/2019	1,596.39	244.11	15.29
12	17/05/2019	1,980.92	199.08	10.05
13	18/05/2019	2,845.69	244.31	8.59
14	20/05/2019	3,895.91	152.48	3.91
15	21/05/2019	467.56	105.15	22.49


CONTINENTAL TRAVEL S.A.C.
AGENCIA DE VIAJES Y TURISMO

Ficha de Registro – Re Test			
Investigador	Orvic Yampool Flores Principe		
Empresa	Continental Travel S.A.C		
Tema de Investigación	Aplicación Web Progresiva en el proceso de facturación de la empresa Continental Travel S.A.C.		
Fecha Inicio	01/06/2019	Fecha Final	30/06/2019

Indicador	Margen de Utilidad Operativa (MUO)		
Fórmula	$MUO = \frac{UO}{IT} * 100$ <p>Dónde: MUO: Margen de Utilidad Operativa. UO: Utilidad Operativa. IT: Ingresos Totales</p>		

ITEM	FECHA EMISIÓN	COMPROBANTES EMITIDOS		MUO
		INGRESOS TOTALES	UTILIDAD OPERATIVA	
1	01/06/2019	4,270.06	487.25	11.41
2	02/06/2019	218.90	20.00	9.14
3	03/06/2019	1,439.24	179.76	12.49
4	04/06/2019	2,787.62	202.38	7.26
5	05/06/2019	13,765.70	1,199.30	8.71
6	06/06/2019	1,771.55	108.45	6.12
7	08/06/2019	1,348.35	106.65	7.91
8	09/06/2019	1,025.79	192.68	18.78
9	10/06/2019	0.00	0.00	0.00
10	11/06/2019	6,271.40	733.80	11.70
11	12/06/2019	1,645.00	95.00	5.78
12	13/06/2019	822.43	60.00	7.30
13	15/06/2019	2,608.92	231.08	8.86
14	16/06/2019	1,566.04	95.47	6.10
15	17/06/2019	7,054.50	435.50	6.17


CONTINENTAL TRAVEL S.A.C.
AGENCIA DE VIAJES Y TURISMO

Ficha de Registro – Post Test			
Investigador	Orvic Yampool Flores Principe		
Empresa	Continental Travel S.A.C		
Tema de Investigación	Aplicación Web Progresiva en el proceso de facturación de la empresa Continental Travel S.A.C.		
Fecha Inicio	01/10/2019	Fecha Final	31/10/2019

Indicador	Margen de Utilidad Operativa (MUO)		
Fórmula	$MUO = \frac{UO}{IT} * 100$ <p>Dónde: MUO: Margen de Utilidad Operativa. UO: Utilidad Operativa. IT: Ingresos Totales</p>		

ITEM	FECHA EMISIÓN	COMPROBANTES EMITIDOS		MUO
		INGRESOS TOTALES	UTILIDAD OPERATIVA	
1	07/10/2019	2,255.10	429.76	19.06
2	08/10/2019	1,914.21	498.00	26.02
3	09/10/2019	941.53	313.27	33.27
4	10/10/2019	2,375.56	294.44	12.39
5	11/10/2019	691.27	149.73	21.66
6	14/10/2019	53.10	12.35	23.26
7	15/10/2019	1,225.20	164.20	13.40
8	16/10/2019	1,279.04	345.94	27.05
9	17/10/2019	3,321.50	611.50	18.41
10	18/10/2019	4,280.86	549.14	12.83
11	19/10/2019	1,647.49	251.62	15.27
12	21/10/2019	2,028.58	381.42	18.80
13	22/10/2019	2,781.29	608.71	21.89
14	24/10/2019	3,654.37	555.63	15.20
15	26/10/2019	1,194.65	307.35	25.73


CONTINENTAL TRAVEL S.A.C.
AGENCIA DE VIAJES Y TURISMO

Anexo 4: Matriz de la Base de Datos del Proyecto de Investigación

Dia	Nivel de Incidencias en Emisiones			Margen de Utilidad Operatia		
	Pre Test		Post Test	Pre Test		Post Test
	Mayo	Junio	Octubre	Mayo	Junio	Octubre
1	0.33	0.28	0.00	13.85	11.41	19.06
2	0.50	0.43	0.14	11.21	9.14	26.02
3	0.29	0.33	0.20	10.71	12.49	33.27
4	0.42	0.43	0.20	5.29	7.26	12.39
5	0.14	0.19	0.00	9.91	8.71	21.66
6	0.31	0.50	0.33	8.23	6.12	23.26
7	0.20	0.33	0.00	8.56	7.91	13.40
8	0.19	0.29	0.14	21.87	18.78	27.05
9	0.67	1.00	0.08	8.68	0.00	18.41
10	0.40	0.40	0.17	14.98	11.70	12.83
11	0.17	0.33	0.13	15.29	5.78	15.27
12	0.40	0.50	0.11	10.05	7.30	18.80
13	0.29	0.38	0.00	8.59	8.86	21.89
14	0.17	0.40	0.14	3.91	6.10	15.20
15	0.50	0.38	0.20	22.49	6.17	25.73

[Handwritten signature]

CONTINENTAL TRAVEL S.A.S.
AGENCIA DE VIAJES Y TURISMO

Anexo N° 5: Resultados de la Confiabilidad del Instrumento de Investigación

Indicador N° 1: Nivel de Incidencias en emisiones

MUESTRA TEST

	Fecha	NIE	TEST
1	03/05/2019	0.33	
2	04/05/2019	0.50	
3	05/05/2019	0.29	
4	06/05/2019	0.42	
5	07/05/2019	0.14	
6	08/05/2019	0.31	
7	10/05/2019	0.20	
8	12/05/2019	0.19	
9	13/05/2019	0.67	
10	14/05/2019	0.40	
11	15/05/2019	0.17	
12	17/05/2019	0.40	
13	18/05/2019	0.29	
14	20/05/2019	0.17	
15	21/05/2019	0.50	

MUESTRA RE TEST

	Fecha	NIE	RE TEST
1	01/06/2019	0.28	
2	02/06/2019	0.43	
3	03/06/2019	0.33	
4	04/06/2019	0.43	
5	05/06/2019	0.19	
6	06/06/2019	0.50	
7	08/06/2019	0.33	
8	09/06/2019	0.29	
9	10/06/2019	1.00	
10	11/06/2019	0.40	
11	12/06/2019	0.33	
12	13/06/2019	0.50	
13	15/06/2019	0.38	
14	16/06/2019	0.40	
15	17/06/2019	0.38	

		PRE_TEST	RE_TEST
PRE_TEST	Correlación de Pearson	1	,758**
	Sig. (bilateral)		,001
	N	15	15
RE_TEST	Correlación de Pearson	,758**	1
	Sig. (bilateral)	,001	
	N	15	15

** . La correlación es significativa en el nivel 0,01 (bilateral).

Anexo N° 5: Resultados de la Confiabilidad del Instrumento de Investigación

Indicador N° 2: Margen de Utilidad Operativa

MUESTRA TEST

	Fecha	MUO	TEST
1	03/05/2019	13.85	
2	04/05/2019	11.21	
3	05/05/2019	10.71	
4	06/05/2019	5.29	
5	07/05/2019	9.91	
6	08/05/2019	8.23	
7	10/05/2019	8.56	
8	12/05/2019	21.87	
9	13/05/2019	8.68	
10	14/05/2019	14.98	
11	15/05/2019	15.29	
12	17/05/2019	10.05	
13	18/05/2019	8.59	
14	20/05/2019	3.91	
15	21/05/2019	22.49	

MUESTRA RE TEST

	Fecha	MUO	RE TEST
1	01/06/2019	11.41	
2	02/06/2019	9.14	
3	03/06/2019	12.49	
4	04/06/2019	7.26	
5	05/06/2019	8.71	
6	06/06/2019	6.12	
7	08/06/2019	7.91	
8	09/06/2019	18.78	
9	10/06/2019	0.00	
10	11/06/2019	11.70	
11	12/06/2019	5.78	
12	13/06/2019	7.30	
13	15/06/2019	8.86	
14	16/06/2019	6.10	
15	17/06/2019	6.17	

		PRE_TEST	RE_TEST
PRE_TEST	Correlación de Pearson	1	,804**
	Sig. (bilateral)		,005
	N	10	10
RE_TEST	Correlación de Pearson	,804**	1
	Sig. (bilateral)	,005	
	N	10	10

** . La correlación es significativa en el nivel 0,01 (bilateral).

Anexo N° 6: Constancia de Evaluación sobre el Juicio de los Expertos

TABLA DE EVALUACION DE EXPERTOS

Apellidos y Nombres del Experto: Gálvez Tapia Orleans Moisés

Título y/o Grado:

Doctor... () Magister... () Ingeniero... () Licenciado... () Otros... ()

Universidad que labora: Universidad Cesar Vallejo – Sede Lima Norte

Fecha: 18/10/2019

TÍTULO DE TESIS

APLICACIÓN WEB PROGRESIVA PARA EL PROCESO DE FACTURACION DE LA EMPRESA CONTINENTAL TRAVEL S.A.C.

Tabla de Evaluación de Expertos para la elección de la Metodología

Mediante la tabla de evaluación de expertos, usted tiene la facultad de calificar las metodologías involucradas, mediante una serie de preguntas con puntuaciones especificadas al final de la tabla.

ITEM	PREGUNTAS	METODOLOGÍA			OBSERVACIONES
		SCRUM	XP	RUP	
1	Califique Ud. como manejan la comunicación con el cliente las siguientes metodologías.	5	3	4	
2	Califique Ud. como se adaptan a las necesidades del cliente las siguientes metodologías.	5	3	4	
3	Califique Ud. como manejan la gestión de prioridades las siguientes metodologías.	5	3	3	
4	Califique Ud. como reaccionan ante los cambios las siguientes metodologías.	5	3	4	
5	Califique Ud. como agilizan el tiempo las siguiente metodologías.	5	3	4	
6	Califique Ud. como representan el avance del proyecto las siguientes metodologías.	5	3	4	
7	Califique Ud. que metodología brinda resultados más rápidos.	5	4	5	
TOTAL					

Evaluar con la siguiente calificación:

1: Muy Malo 2: Malo 3: Regular 2: Bueno 3: Muy Bueno

Sugerencias:

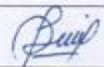

Firma del Experto

TABLA DE EVALUACION DE EXPERTOS

Apellidos y Nombres del Experto: Rodrigo Valenzuela Morúa Patricia

Título y/o Grado:

Doctor... () Magister... (X) Ingeniero... () Licenciado... () Otros... ()

Universidad que labora: **Universidad Cesar Vallejo – Sede Lima Norte**

Fecha: 6/09/2019

TÍTULO DE TESIS

APLICACIÓN WEB PROGRESIVA PARA EL PROCESO DE FACTURACION DE LA EMPRESA CONTINENTAL TRAVEL S.A.C.

Tabla de Evaluación de Expertos para la elección de la Metodología

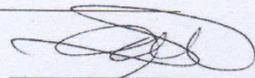
Mediante la tabla de evaluación de expertos, usted tiene la facultad de calificar las metodologías involucradas, mediante una serie de preguntas con puntuaciones especificadas al final de la tabla.

ITEM	PREGUNTAS	METODOLOGÍA			OBSERVACIONES
		SCRUM	XP	RUP	
1	Califique Ud. como manejan la comunicación con el cliente las siguientes metodologías.	2	3	3	
2	Califique Ud. como se adaptan a las necesidades del cliente las siguientes metodologías.	3	2	2	
3	Califique Ud. como manejan la gestión de prioridades las siguientes metodologías.	3	2	2	
4	Califique Ud. como reaccionan ante los cambios las siguientes metodologías.	2	2	3	
5	Califique Ud. como agilizan el tiempo las siguiente metodologías.	3	2	2	
6	Califique Ud. como representan el avance del proyecto las siguientes metodologías.	3	2	2	
7	Califique Ud. que metodología brinda resultados más rápidos.	3	2	2	
TOTAL		19	15	16	

Evaluar con la siguiente calificación:

1: Malo 2: Regular 3: Bueno

Sugerencias:



 Firma del Experto

TABLA DE EVALUACION DE EXPERTOS

Apellidos y Nombres del Experto: Cabrera Agosto, Juan S.

Título y/o Grado:

Doctor... () Magister... (x) Ingeniero... () Licenciado... () Otros... ()

Universidad que labora: Universidad Cesar Vallejo – Sede Lima Norte

Fecha: 11/07/2019

TÍTULO DE TESIS

APLICACIÓN WEB PROGRESIVA PARA EL PROCESO DE FACTURACION DE LA EMPRESA CONTINENTAL TRAVEL S.A.C.

Tabla de Evaluación de Expertos para la elección de la Metodología

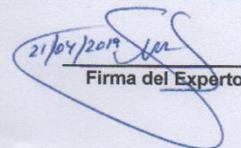
Mediante la tabla de evaluación de expertos, usted tiene la facultad de calificar las metodologías involucradas, mediante una serie de preguntas con puntuaciones especificadas al final de la tabla.

ITEM	PREGUNTAS	METODOLOGÍA			OBSERVACIONES
		SCRUM	XP	RUP	
1	Califique Ud. como manejan la comunicación con el cliente las siguientes metodologías.	3	2	1	
2	Califique Ud. como se adaptan a las necesidades del cliente las siguientes metodologías.	3	3	2	
3	Califique Ud. como manejan la gestión de prioridades las siguientes metodologías.	3	2	2	
4	Califique Ud. como reaccionan ante los cambios las siguientes metodologías.	3	3	2	
5	Califique Ud. como agilizan el tiempo las siguiente metodologías.	3	3	1	
6	Califique Ud. como representan el avance del proyecto las siguientes metodologías.	3	2	2	
7	Califique Ud. que metodología brinda resultados más rápidos.	3	3	1	
TOTAL		21	18	11	

Evaluar con la siguiente calificación:

1: Malo 2: Regular 3: Bueno

Sugerencias:



 21/07/2019

 Firma del Experto

VALIDACIÓN DE INSTRUMENTO

I. DATOS GENERALES

Apellidos y Nombres: *Cálvez Tapu Orleans Moises*
 Cargo e Institución donde labora: *Mg. en Ingeniería de Sistemas*
 Universidad Cesar Vallejo, Escuela Académica de Ingeniería de Sistemas
 Nombre del Instrumento motivo de Evaluación:
 Ficha de Observación – Nivel de Incidencias en Emisiones (NIE)
 Título de Investigación:
**APLICACIÓN WEB PROGRESIVA PARA EL PROCESO DE FACTURACIÓN DE LA
 EMPRESA CONTINENTAL TRAVEL S.A.C.**
 Autor: Flores Principe, Orvic Yampool

II. ASPECTOS DE VALIDACIÓN

INDICADORES	CRITERIOS	Deficiente 0-20%	Regular 21-50%	Bueno 51-70%	Muy Bueno 71-80%	Excelente 81-100%
1. CLARIDAD	Esta formulado con el lenguaje apropiado.				80%	
2. OBJETIVIDAD	Esta expresado en conducta reservable.				80%	
3. ACTUALIDAD	Es adecuado el avance de la ciencia y tecnología.				80%	
4. ORGANIZACIÓN	Existe una organización lógica.				80%	
5. SUFICIENCIA	Comprende los aspectos de cantidad y claridad.				80%	
6. INTENCIONALIDAD	Adecuado para valorar los aspectos del sistema metodológico y científico.				80%	
7. CONSISTENCIA	Está basado en aspectos teóricos, científicos acordes a la tecnología educativa.				80%	
8. COHERENCIA	Entre los indices, indicadores y dimensiones.				80%	
9. METODOLOGIA	Responde al propósito de trabajo bajo los objetivos a lograr.				80%	
10. PERTINENCIA	El instrumento es adecuado al tipo de investigación.				80%	
PROMEDIO DE EVALUACIÓN					80%	

III. PROMEDIO DE LA VALORACIÓN: _____

IV. OPCION DE APLICABILIDAD:

- () El instrumento puede ser aplicado, tal como está elaborado.
 () El instrumento debe ser mejorado, antes de ser aplicado.
 Considere las recomendaciones y aplicar al trabajo

Fecha: _____

[Firma]

 Firma del Experto

VALIDACIÓN DE INSTRUMENTO

I. DATOS GENERALES

Apellidos y Nombres: Cesva Gallacencio Loarife J.

Cargo e Institución donde labora: Regíster

Universidad Cesar Vallejo, Escuela Académica de Ingeniería de Sistemas

Nombre del Instrumento motivo de Evaluación:

Ficha de Observación – Nivel de Incidencias en Emisiones (NIE)

Título de Investigación:

**APLICACIÓN WEB PROGRESIVA PARA EL PROCESO DE FACTURACIÓN DE LA
EMPRESA CONTINENTAL TRAVEL S.A.C.**

Autor: Flores Principe, Orvic Yampool

II. ASPECTOS DE VALIDACIÓN

INDICADORES	CRITERIOS	Deficiente 0-20%	Regular 21-50%	Bueno 51-70%	Muy Bueno 71-80%	Excelente 81-100%
1. CLARIDAD	Esta formulado con el lenguaje apropiado.				80%	
2. OBJETIVIDAD	Esta expresado en conducta reservable.				80%	
3. ACTUALIDAD	Es adecuado el avance de la ciencia y tecnología.				80%	
4. ORGANIZACIÓN	Existe una organización lógica.				80%	
5. SUFICIENCIA	Comprende los aspectos de cantidad y claridad.				80%	
6. INTENCIONALIDAD	Adecuado para valorar los aspectos del sistema metodológico y científico.				80%	
7. CONSISTENCIA	Está basado en aspectos teóricos, científicos acordes a la tecnología educativa.				80%	
8. COHERENCIA	Entre los índices, indicadores y dimensiones.				80%	
9. METODOLOGIA	Responde al propósito de trabajo bajo los objetivos a lograr.				80%	
10. PERTINENCIA	El instrumento es adecuado al tipo de investigación.				80%	
PROMEDIO DE EVALUACIÓN						

III. PROMEDIO DE LA VALORACIÓN: _____

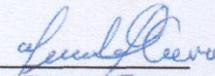
IV. OPCION DE APLICABILIDAD:

() El instrumento puede ser aplicado, tal como está elaborado.

() El instrumento debe ser mejorado, antes de ser aplicado.

Considere las recomendaciones y aplicar al trabajo

Fecha:


Firma del Experto

VALIDACIÓN DE INSTRUMENTO

I. DATOS GENERALES

Apellidos y Nombres: *Romero Valenzuela Maria Patricia*

Cargo e Institución donde labora:

Universidad Cesar Vallejo, Escuela Académica de Ingeniería de Sistemas

Nombre del Instrumento motivo de Evaluación:

Ficha de Observación – Nivel de Incidencias en Emisiones (NIE)

Título de Investigación:

**APLICACIÓN WEB PROGRESIVA PARA EL PROCESO DE FACTURACIÓN DE LA
EMPRESA CONTINENTAL TRAVEL S.A.C.**

Autor: Flores Principe, Orvic Yampool

II. ASPECTOS DE VALIDACIÓN

INDICADORES	CRITERIOS	Deficiente 0-20%	Regular 21-50%	Bueno 51-70%	Muy Bueno 71-80%	Excelente 81-100%
1. CLARIDAD	Esta formulado con el lenguaje apropiado.					85
2. OBJETIVIDAD	Esta expresado en conducta reservable.					90
3. ACTUALIDAD	Es adecuado el avance de la ciencia y tecnología.					90
4. ORGANIZACIÓN	Existe una organización lógica.					85
5. SUFICIENCIA	Comprende los aspectos de cantidad y claridad.					90
6. INTENCIONALIDAD	Adecuado para valorar los aspectos del sistema metodológico y científico.					90
7. CONSISTENCIA	Está basado en aspectos teóricos, científicos acordes a la tecnología educativa.					90
8. COHERENCIA	Entre los índices, indicadores y dimensiones.					85
9. METODOLOGIA	Responde al propósito de trabajo bajo los objetivos a lograr.					90
10. PERTINENCIA	El instrumento es adecuado al tipo de investigación.					85
PROMEDIO DE EVALUACIÓN						

III. PROMEDIO DE LA VALORACIÓN: _____

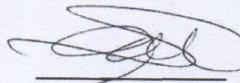
IV. OPCION DE APLICABILIDAD:

() El instrumento puede ser aplicado, tal como está elaborado.

() El instrumento debe ser mejorado, antes de ser aplicado.

Considere las recomendaciones y aplicar al trabajo

Fecha: _____


Firma del Experto

VALIDACIÓN DE INSTRUMENTO

I. DATOS GENERALES

Apellidos y Nombres: *Calvez Tapia Orleans Moises*
 Cargo e Institución donde labora: *Mg. en Ingeniería de Sistemas*
 Universidad Cesar Vallejo, Escuela Académica de Ingeniería de Sistemas
 Nombre del Instrumento motivo de Evaluación:
 Ficha de Observación – Margen de Utilidad Operativa (MUO)
 Título de Investigación:
APLICACIÓN WEB PROGRESIVA PARA EL PROCESO DE FACTURACIÓN DE LA EMPRESA CONTINENTAL TRAVEL S.A.C.
 Autor: Flores Principe, Orvic Yampool

II. ASPECTOS DE VALIDACIÓN

INDICADORES	CRITERIOS	Deficiente 0-20%	Regular 21-50%	Bueno 51-70%	Muy Bueno 71-80%	Excelente 81-100%
1. CLARIDAD	Esta formulado con el lenguaje apropiado.				80%	
2. OBJETIVIDAD	Esta expresado en conducta reservable.				80%	
3. ACTUALIDAD	Es adecuado el avance de la ciencia y tecnología.				80%	
4. ORGANIZACIÓN	Existe una organización lógica.				80%	
5. SUFICIENCIA	Comprende los aspectos de cantidad y claridad.				80%	
6. INTENCIONALIDAD	Adecuado para valorar los aspectos del sistema metodológico y científico.				80%	
7. CONSISTENCIA	Está basado en aspectos teóricos, científicos acordes a la tecnología educativa.				80%	
8. COHERENCIA	Entre los índices, indicadores y dimensiones.				80%	
9. METODOLOGIA	Responde al propósito de trabajo bajo los objetivos a lograr.				80%	
10. PERTINENCIA	El instrumento es adecuado al tipo de investigación.				80%	
PROMEDIO DE EVALUACIÓN					80%	

III. PROMEDIO DE LA VALORACIÓN: _____

IV. OPCION DE APLICABILIDAD:

- () El instrumento puede ser aplicado, tal como está elaborado.
 () El instrumento debe ser mejorado, antes de ser aplicado.
 Considere las recomendaciones y aplicar al trabajo

Fecha: _____

Orvic Yampool

Firma del Experto

VALIDACIÓN DE INSTRUMENTO

I. DATOS GENERALES

Apellidos y Nombres: *Cueva Gallagüencio Juanita J.*

Cargo e Institución donde labora: *Magister*

Universidad Cesar Vallejo, Escuela Académica de Ingeniería de Sistemas

Nombre del Instrumento motivo de Evaluación:

Ficha de Observación – Margen de Utilidad Operativa (MUO)

Título de Investigación:

**APLICACIÓN WEB PROGRESIVA PARA EL PROCESO DE FACTURACIÓN DE LA
EMPRESA CONTINENTAL TRAVEL S.A.C.**

Autor: Flores Principe, Orvic Yampool

II. ASPECTOS DE VALIDACIÓN

INDICADORES	CRITERIOS	Deficiente 0-20%	Regular 21-50%	Buena 51-70%	Muy Buena 71-80%	Excelente 81-100%
1. CLARIDAD	Esta formulado con el lenguaje apropiado.				90%	
2. OBJETIVIDAD	Esta expresado en conducta reservable.				90%	
3. ACTUALIDAD	Es adecuado el avance de la ciencia y tecnología.				80%	
4. ORGANIZACIÓN	Existe una organización lógica.				90%	
5. SUFICIENCIA	Comprende los aspectos de cantidad y claridad.				80%	
6. INTENCIONALIDAD	Adecuado para valorar los aspectos del sistema metodológico y científico.				80%	
7. CONSISTENCIA	Está basado en aspectos teóricos, científicos acordes a la tecnología educativa.				80%	
8. COHERENCIA	Entre los índices, indicadores y dimensiones.				80%	
9. METODOLOGIA	Responde al propósito de trabajo bajo los objetivos a lograr.				80%	
10. PERTINENCIA	El instrumento es adecuado al tipo de investigación.				80%	
PROMEDIO DE EVALUACIÓN					80%	

III. PROMEDIO DE LA VALORACIÓN: _____

IV. OPCION DE APLICABILIDAD:

() El instrumento puede ser aplicado, tal como está elaborado.

() El instrumento debe ser mejorado, antes de ser aplicado.

Considere las recomendaciones y aplicar al trabajo

Fecha: _____

Juanita Cueva
Firma del Experto

VALIDACIÓN DE INSTRUMENTO

I. DATOS GENERALES

Apellidos y Nombres: *Romero Valerín María Patricia*

Cargo e Institución donde labora:

Universidad Cesar Vallejo, Escuela Académica de Ingeniería de Sistemas

Nombre del Instrumento motivo de Evaluación:

Ficha de Observación – Margen de Utilidad Operativa (MUO)

Título de Investigación:

**APLICACIÓN WEB PROGRESIVA PARA EL PROCESO DE FACTURACIÓN DE LA
EMPRESA CONTINENTAL TRAVEL S.A.C.**

Autor: Flores Príncipe, Orvic Yampool

II. ASPECTOS DE VALIDACIÓN

INDICADORES	CRITERIOS	Deficiente 0-20%	Regular 21-50%	Bueno 51-70%	Muy Bueno 71-80%	Excelente 81-100%
1. CLARIDAD	Esta formulado con el lenguaje apropiado.					90
2. OBJETIVIDAD	Esta expresado en conducta reservable.					85
3. ACTUALIDAD	Es adecuado el avance de la ciencia y tecnología.					90
4. ORGANIZACIÓN	Existe una organización lógica.					90
5. SUFICIENCIA	Comprende los aspectos de cantidad y claridad.					85
6. INTENCIONALIDAD	Adecuado para valorar los aspectos del sistema metodológico y científico.					90
7. CONSISTENCIA	Está basado en aspectos teóricos, científicos acordes a la tecnología educativa.					90
8. COHERENCIA	Entre los índices, indicadores y dimensiones.					85
9. METODOLOGIA	Responde al propósito de trabajo bajo los objetivos a lograr.					90
10. PERTINENCIA	El instrumento es adecuado al tipo de investigación.					90
PROMEDIO DE EVALUACIÓN						

III. PROMEDIO DE LA VALORACIÓN: _____

IV. OPCION DE APLICABILIDAD:

() El instrumento puede ser aplicado, tal como está elaborado.

() El instrumento debe ser mejorado, antes de ser aplicado.

Considere las recomendaciones y aplicar al trabajo

Fecha: _____


Firma del Experto

Anexo N° 7: Cédula de la Entrevista

FICHA DE ENTREVISTA N° 01

Se realiza esta ficha con el fin de obtener información real de la situación en el proceso de facturación de la empresa CONTINETAL TRAVEL S.A.C.

Responsable: Sr. Sergio Moran Palacios

Cargo: Jefe de Sistemas

1. ¿Qué herramienta usan para el control del proceso de facturación?
Actualmente se usa un sistema de escritorio denominado BackOffice, el cual se usa para generar las cotizaciones (tickets y guías de servicios), a partir de estos se generan los comprobantes de pago el cual se gestiona en el módulo de facturación, en el que incluye la gestión de las facturas, boletas, notas de crédito y débito; el cual es muy limitados en sus funciones.
2. ¿Cómo se almacena la información actualmente?
Parte de la información de los comprobantes de pago se almacena tanto en físico, ubicados en archiveros; además en una base de datos en SQL SERVER 2012, gestionadas por el BackOffice.
3. ¿En qué formato usted genera los comprobantes de pago?
Tanto Físico como Virtual, ya que algunos clientes son empresas privadas a las cuales se les tiene emitir por obligación un comprobante virtual, además del XML.
4. ¿Cuál es el procedimiento al generar un comprobante de pago?
La emisión de comprobantes de pago, inicia con los registros de tickets y las guías de servicios, consta de registros de boletos aéreos, hospedaje y/o servicios turísticos. Se debe procesarse inmediatamente y enviar el comprobante de pago con el Courier, para el posterior pago del cliente. Estos procesos son registrados por las agentes manualmente en el BackOffice; sin embargo, bajo la presión de realizarlas con rapidez, ha provocado errores de cálculo en registros de manera frecuente.

Sin embargo, existe dependencia del negocio hacia algunas personas que conozca los datos sobre precios, proveedores, lugares, ofertas relacionadas a los servicios, de manera diaria. En este caso se requiere la presencia de esta persona para la mayoría de actividades y dado que una sola persona no puede cubrir todas estas actividades en el tiempo programado, suelen haber retrasos en las mismas.

Una vez procesado los tickets y/o guías de servicio; el personal del área de facturación verifica de manera manual tanto el registro del BackOffice como

el ticket registrado en el GDS, lo cual toma 5 minutos aproximados, después de verificar todo esto se factura el comprobante de pago, pero de presentarse errores, el personal de facturación se comunica con la Agente que emitió para su respectiva corrección, en caso el Agente no pueda corregir se comunica con el Área de Sistemas para que lo modifique en la base de datos, sin embargo esto es considerado una mala práctica lo cual se quiere evitar. Después de facturar el comprobante de pago, el sistema se encarga de indicar el número de factura y guardar los datos, lo que es una gran ventaja puesto que esto permite acceder a la información, los comprobantes físicos originales son entregados como comprobante de venta al cliente final, dos copias para el área de cobranzas, uno para el área de facturación y uno más para el área contable.

Pero en el proceso de facturación, se ve afectadas por otros conflictos presentados en la actual sistema, como es el caso del número de facturas anuladas por errores de digitalización, la mala toma de reservas o el rechazo de reserva totales o parciales, provocando la anulación del documento; lo cual resulta un tanto trabajoso para la empresa, ya que según lo establece SUNAT, la empresa puede anular facturas dentro de los 7 días, pero en caso sea más días se que tiene que emitir una nota de crédito que sustente esa factura y/o boleta, con el fin de evitar cualquier multa por parte de la SUNAT; esta situación se presenta semanalmente recargando el flujo de trabajo de las áreas involucradas en todo este proceso, se sabe que el 15% de este total son rechazados mensualmente cifra que resulta preocupante.

Todas estas situaciones impactan directamente en la satisfacción del cliente, causa de problemas administrativos; por tanto, se genera una actitud de rechazo al servicio, casos en la que el cliente se abstiene del pago debido a los errores en el documento impreso o electrónico que se declaró, posteriormente incitando a un queja e incomodidad por parte del cliente, donde finalmente se genera ya sea nota de crédito o débito, además de un nuevo comprobante de pago con los montos correctos; todo ello afectando a las utilidades, incrementando el costo e impactando en los ingresos y finanzas de la empresa.

Mensualmente el área contable realiza la declaración de impuestos en las oficinas de la empresa la misma que hasta el momento no ha tenido problemas, pero si contratiempos como falta de facturas, boletas, nota de crédito o débito ya sea por extravió, deterioro y/o pago faltante, en la realización de este proceso. Debido a que resulta complicado ordenar y procesar todas las transacciones realizadas para luego rendir la tributación correspondiente, en el mejor de los casos, se contrata como apoyo a un contador que realice la contabilidad ya sea mensual o anual, y en el peor de los casos simplemente no se realiza la tributación. Se resalta de este hecho

que la información que se presenta no se encuentra ordenada y no es real, las cuales afectan directamente los ingresos y el flujo de caja a corto y mediano plazo de la empresa.

5. ¿Qué dificultades que se presentan durante el proceso de facturación?

El sistema actual no genera XML propuesto con la SUNAT para la facturación electrónica por lo que se tiene que generar manualmente en el Facturador de la SUNAT, lo cual toma de 3 a 10 minutos, de acuerdo a la glosa que se emitió, por lo que este proceso se vuelve lento y frecuentemente con muchos errores. Esto provoca re-trabajo que, usualmente, implica realizar un esfuerzo adicional, por lo que se ve afectado el funcionamiento de la empresa. Todo esto se debe a que no existen métodos claros y controlados para el desarrollo de este proceso, lo que genera malestar entre los empleados debido a que se presentan situaciones de estrés, conflictos de comunicación e ineficiencia laboral.

6. ¿Considera Ud. que es necesario implementar un sistema web para mejorar el proceso de facturación?

Pienso que sería necesario; ya que, por la información publicada por la SUNAT, se debe emitir electrónicamente los comprobantes de pago a partir del 01/03/2019.

CONTINENTAL TRAVEL S.A.C.
AGENCIA DE VIAJES Y TURISMO

Firma y Sello

Anexo N° 8: Constancia de Aceptación - Continental Travel S.A.C.



Av. Canaval y Moreira 285 San Isidro Lima 27- Perú
Central Telefónica: (511) 415-5050 Fax: (511) 415-5070
continental@continentaltravel.com.pe
www.continentaltravel.com.pe



CONSTANCIA

Por medio del presente, se da por aceptada la propuesta de desarrollo de la **"Aplicación Web Progresiva para el proceso de facturación en la empresa Continental Travel S.A.C."** de nuestro representado.

El proyecto, se encontrará a cargo del Sr. **ORVIC YAMPOOL FLORES PRINCIPE**, identificado con DNI **71554511**, quien realiza sus prácticas pre profesionales en la empresa J&S Consultores con el objeto de elaborar su trabajo de tesis. Dicha propuesta de solución cumple con nuestra expectativa y mejora el área comercial, permitiendo el ahorro de recursos, minimizar el tiempo y brindar un servicio eficiente a nuestro clientes y usuarios.

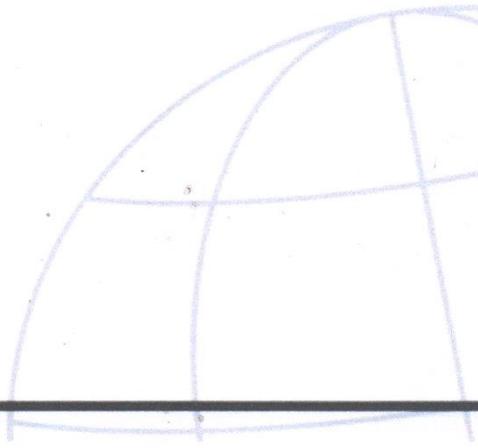
Para la elaboración de este sistema se compartió información confidencial de nuestra institución. Además, que el sistema se encuentra publicada en nuestra intranet y disponible para los usuarios pertinentes.

Se expide la presente a solicitud del interesado para los fines que se estime conveniente.

San Isidro, 26 de Noviembre de 2019.

CONTINENTAL TRAVEL S.A.C.


Gian Marco Piera Testino
SUB - GERENTE GENERAL



ANEXO 9
DESARROLLO DE LA METODOLOGÍA

Metodología de desarrollo de Software del Proyecto de Investigación

1. DEFINICIÓN DE LA METODOLOGÍA

1.1. Fundamentación

Esta documentación refiere la ejecución de la metodología ágil denominada SCRUM; esto nos permitirá responder a los cambios y las necesidades y así tomar mejores decisiones y no tener problemas a largo plazo con SUNAT como prueba de envío y recepción de comprobantes de pagos electrónicos.

El proyecto se basa en la metodología SCRUM, porque se adapta mejor a la forma en que trabaja la empresa, en la que se cumplen las historias de usuarios establecidas por la metodología; también incluirá los sprints necesarios, que nos proporcionarán una visión general de todos los procesos.

Alcance de la Metodología

Considerando lo que se analizó en cada uno de los objetivos específicos, se considera apropiado que en el proyecto propuesto logre los objetivos prioritarios:

- El sistema podrá generar cada comprobante de pago emitido por BackOffice, en función de las ventas generadas en el día.
- El sistema permite registrar el tipo de cambio del día.
- El sistema admitirá la remisión y recepción de XML para rastrear los movimientos de los comprobantes electrónicos generados.
- El sistema permite el envío, recepción y cancelación de los comprobantes electrónicos, evitando así inconvenientes posteriores.
- El sistema proporciona informes de cada tipo de recibo de pago, así como informes mensuales.

Posteriormente, debe tenerse en cuenta que durante el avance del software y el seguimiento de cada sprint, el ajuste del Producto Backlog produjo nuevas versiones actualizadas.

1.2. Roles de la Metodología

Tabla 1: Roles del Proyecto

Rol	Nombre
Scrum Master	Ronnie Alarcon Morales
Team Member	Orvic Yampool Flores Principe Wilmer Alama Seminario Jimmy Pazos Ramirez Joseph Palomino
Product Owner	Sergio Moran Palacios

Fuente: Elaboración propia

Tabla 2: Implicados del Proyecto

Comprometidos	Implicados
Scrum Master	Equipo de desarrollo
Team Member	
Product Owner	Continental Travel S.A.C.

Fuente: Elaboración propia

2. ANÁLISIS Y DESARROLLO DEL PRODUCTO

2.1. Historias de Usuario

A continuación, se describe los requerimientos que el usuario necesita para con el sistema, discutidas con el Product Owner en las reuniones anteriores.

Tabla 3: HU01 - Login

Historia de Usuario	
Numero: 1	Usuario: Administrador / Usuario
Nombre de Historia: Login	
Prioridad en Negocio: Alta	Riesgo de desarrollo: Medio
Descripción: El sistema debe permitir a un usuario o cliente, el ingreso a la Intranet; en el que acceda a la interfaz principal, denominada Menú principal.	
Condiciones y Restricciones: El usuario o cliente debe digitar de manera obligatoria su nombre de usuario y una contraseña.	

Fuente: Elaboración propia

Tabla 4: HU02 - Mantenimiento de Cliente

Historia de Usuario	
Numero: 2	Usuario: Administrador
Nombre de Historia: Mantenimiento de Cliente	
Prioridad en Negocio: Alta	Riesgo de desarrollo: Alta
Descripción: Como Administrador del sistema nos permite modificar una Cliente (cada cliente va a contener un Nombre, Tipo de documento de Identidad, Razón social, Dirección, Teléfono, Nombre corto, Numero de documento de identidad, Dirección de Facturación, Fax), para poder gestionar sus datos previamente registrados.	
Condiciones y Restricciones: Solo podrá digitar un usuario y una contraseña valida.	

Fuente: Elaboración propia

Tabla 5: HU03 - Mantenimiento de Empresa

Historia de Usuario	
Numero: 3	Usuario: Administrador
Nombre de Historia: Mantenimiento de Empresa	
Prioridad en Negocio: Alta	Riesgo de desarrollo: Alta
Descripción: Como Administrador del sistema nos permite modificar una Empresa (cada empresa va a contener un Razón Social, Dirección, Usuario SUNAT, Numero RUC, Porcentaje de IGV, Contraseña de SUNAT, Contraseña del certificado, Código de País), para poder gestionar sus datos previamente registrados.	
Condiciones y Restricciones: Es necesario que los campos Razón Social, Dirección, Usuario SUNAT, Numero RUC, Porcentaje de IGV, Contraseña de SUNAT, Contraseña del certificado y Código de País sean obligatorios.	

Fuente: Elaboración propia

Tabla 6: HU04 - Mantenimiento de Tipo de Cambio

Historia de Usuario	
Numero: 4	Usuario: Administrador
Nombre de Historia: Mantenimiento de Tipo de Cambio	
Prioridad en Negocio: Alta	Riesgo de desarrollo: Alta
Descripción: Como Administrador del sistema nos permite registrar, modificar y eliminar un Tipo de Cambio (cada Tipo de Cambio va a contener un Fecha de Cambio, Moneda y Monto de Cambio), para poder gestionar sus datos previamente registrados.	
Condiciones y Restricciones: Es necesario que los campos Fecha de Cambio, Moneda y Monto de Cambio sean obligatorios.	

Fuente: Elaboración propia

Tabla 7: H05 - Mantenimiento de Moneda

Historia de Usuario	
Numero: 5	Usuario: Administrador
Nombre de Historia: Mantenimiento de Moneda	
Prioridad en Negocio: Alta	Riesgo de desarrollo: Alta
Descripción: Como Administrador del sistema nos permite registrar, modificar y eliminar Monedas (cada Moneda va a contener un Nombre y Símbolo), para poder gestionar sus datos previamente registrados.	
Condiciones y Restricciones: Es necesario que los campos Nombre y Símbolo sean obligatorios.	

Fuente: Elaboración propia

Tabla 8: H06 - Mantenimiento de Usuario

Historia de Usuario	
Numero: 6	Usuario: Administrador / Usuario
Nombre de Historia: Mantenimiento de Usuario	
Prioridad en Negocio: Alta	Riesgo de desarrollo: Alta
Descripción: Como Administrador del sistema nos permite registrar, modificar y eliminar Usuarios (cada Usuario va a contener un Login, Nombre, Correo, Cliente, Apellidos, Teléfono y Contraseña), para poder gestionar sus datos previamente registrados.	
Condiciones y Restricciones: Es necesario que los campos Login, Nombre, Correo, Apellidos, Teléfono sean obligatorios.	

Fuente: Elaboración propia

Tabla 9: H07 - Envío y Recepción Comprobantes de Pago

Historia de Usuario	
Numero: 7	Usuario: Administrador / Usuario
Nombre de Historia: Envío y Recepción Comprobantes de Pago Electrónicos	
Prioridad en Negocio: Alta	Riesgo de desarrollo: Alta
Descripción: Como Usuario Logueado del Sistema podrá enviar comprobantes de pago electrónicos y generar un reporte para enviarlo al Cliente con el correo electrónico registrado previamente.	
Condiciones y Restricciones: Solo podrá enviar los comprobantes de pago electrónico dentro de los siete días hábiles, y aquellos que aún no aun registrados por la SUNAT. En caso este fuera de fecha, se lanzara una alerta.	

Fuente: Elaboración propia

Tabla 10: H08 - Envío y Recepción de Cierre Diarios de Boletas

Historia de Usuario	
Numero: 8	Usuario: Administrador / Usuario
Nombre de Historia: Envío y Recepción de Cierre Diarios de Boletas Electrónicas	
Prioridad en Negocio: Alta	Riesgo de desarrollo: Alta
Descripción: Como Usuario Logueado del Sistema podrá enviar, recepcionar y reenviar los Cierre Diarios de Comprobantes de Pago Electrónico registrado previamente durante el día, además de poder generar un reporte mensual.	
Condiciones y Restricciones: Solo podrá enviar el Cierre Diarios dentro de los siete días hábiles. En caso ha sido registrado por la SUNAT, lanzara una alerta indicando ya sea un aviso o un error en la generación del XML o problemas de envío a la SUNAT.	

Fuente: Elaboración propia

Tabla 11: HU09 - Envío y Recepción Resúmenes de Anulaciones

Historia de Usuario	
Numero: 9	Usuario: Administrador / Usuario
Nombre de Historia: Envío y Recepción Resúmenes de Anulaciones de Comprobantes de Pago	
Prioridad en Negocio: Alta	Riesgo de desarrollo: Medio
Descripción: Como Usuario Logueado del Sistema podrá enviar, recepcionar y reenviar los Resúmenes de Anulaciones de Comprobantes de Pago Electrónico registrado previamente durante el día, además de poder generar un reporte mensual.	
Condiciones y Restricciones: Solo podrá enviar el Resumen de Anulación solo sino ha sido registrado por la SUNAT, lanzara una alerta indicando ya sea un aviso o un error en la generación del XML o problemas de envío a la SUNAT.	

Fuente: Elaboración propia

Tabla 12: HU10 - Consultar Nivel de incidencias en Emisiones

Historia de Usuario	
Numero: 10	Usuario: Administrador / Usuario
Nombre de Historia: Consultar Nivel de incidencias en Emisiones	
Prioridad en Negocio: Alta	Riesgo de desarrollo: Medio
Descripción: Como Usuario Logueado del Sistema podrá generar reportes en base fechas seleccionadas previamente..	
Condiciones y Restricciones: Solo se podrá generar el reporte con un mínimo de 5 días y un máximo de 365 días.	

Fuente: Elaboración propia

Tabla 13: HU11 - Consultar el Margen de Utilidad Operativa

Historia de Usuario	
Numero: 11	Usuario: Administrador / Usuario
Nombre de Historia: Consultar Margen de Utilidad Operativa	
Prioridad en Negocio: Alta	Riesgo de desarrollo: Medio
Descripción: Como Usuario Logueado del Sistema podrá generar reportes en base fechas seleccionadas previamente..	
Condiciones y Restricciones: Solo se podrá generar el reporte con un mínimo de 5 días y un máximo de 365 días.	

Fuente: Elaboración propia

2.2. Product Backlog

Tabla 14: Requerimientos Funcionales

CODIGO	REQUERIMIENTOS FUNCIONALES	PRIORIDAD
RF01	El sistema debe permitir la autenticación de usuarios, mediante un nombre de usuario y una contraseña.	ALTA
RF02	El sistema debe permitir listar, registrar, modificar y eliminar usuarios para gestionar el proceso de facturación electrónica.	ALTA
RF03	El sistema debe permitir listar y modificar los clientes relacionados al comprobante de pago electrónico emitido.	ALTA
RF04	El sistema debe permitir listar, modificar los datos de la empresa que emitirá la facturación electrónica.	ALTA
RF05	El sistema debe permitir listar y modificar los productos relacionados al comprobante de pago electrónico emitido.	ALTA
RF06	El sistema debe permitir listar, registrar y modificar tipo de cambio a la fecha a emitir.	ALTA
RF07	El sistema debe permitir listar, registrar y modificar moneda a emitir según el comprobante de pago electrónico generado.	ALTA
RF08	El sistema permitirá filtrar los comprobantes de pago de acuerdo a la fecha de emisión, cliente, tipo de comprobante, estado , serie y numero de comprobante.	ALTA
RF09	El sistema permitirá generar un XML y enviar un comprobante de pago electrónico a la SUNAT.	ALTA
RF10	El sistema permitirá recepcionar el XML de respuesta por parte de la SUNAT e interpretar su estado actual.	ALTA
RF11	El Sistema permitirá generar un reporte individual por cada comprobante de pago electrónico generado.	ALTA
RF12	El sistema permitirá enviar un XML y un reporte del comprobante de pago electrónico al Cliente, para su posterior confirmación.	ALTA
RF13	El sistema permitirá reenviar un XML de un comprobante de pago electrónico que haya presentado inconvenientes en su envío a la SUNAT.	ALTA
RF14	El sistema permitirá anular los comprobantes de pago electrónicos emitidos, y enviar un XML de anulación a la SUNAT.	ALTA

RF15	El sistema permitirá filtrar los cierres diarios generados en el día, mediante un filtro de fecha de emisión y estado del comprobante.	ALTA
RF16	El sistema permitirá generar un XML y enviar un nuevo cierre diario de los comprobantes de pago relacionados a la Boleta, para sus posterior registro en la SUNAT.	ALTA
RF17	El sistema permitirá obtener un XML de respuesta del cierre diario de los comprobantes de pago electrónicos relacionados a la Boleta, para sus posterior validación.	ALTA
RF18	El sistema permitirá reenviar el XML de un cierre diario de los comprobantes de pago electrónicos relacionados a la Boleta, para sus posterior registro en la SUNAT.	ALTA
RF19	El sistema permitirá listar los comprobantes de pago electrónicos relacionados a la factura que hayan sido anulados anteriormente, mediante un filtro de fecha de emisión y estado del comprobante.	ALTA
RF20	El Sistema permitirá obtener el XML de envió de un comprobante de pago electrónico anulado anteriormente.	ALTA
RF21	El Sistema permitirá obtener XML de respuesta de un comprobante de pago electrónico anulado anteriormente.	ALTA
RF22	El sistema permitirá listar un historial de los comprobantes de pago electrónicos generados en el BackOffice.	ALTA
RF23	El sistema permitirá obtener información de un comprobante de pago electrónico generado en el BackOffice.	ALTA
RF24	El sistema permitirá listar un historial de los comprobantes de pago electrónicos enviados a la SUNAT.	ALTA
RF25	El sistema permitirá obtener información sobre el estado de un comprobante de pago electrónico enviado a la SUNAT.	ALTA
RF26	El sistema permitirá generar reportes mensuales de los comprobantes de pago electrónicos anulados.	ALTA
RF27	El sistema permitirá generar reportes mensuales de los cierres diarios generados de acuerdo comprobantes de pago electrónicos emitidos durante el día.	ALTA
RF28	El sistema permitirá descargar los XML generados de los comprobantes de pago electrónicos.	ALTA
RF29	El sistema permitirá exportar la información a un archivo de tipo Excel, que contengan datos de los comprobantes de pago electrónicos de acuerdo al filtro propuesto en la interfaz.	ALTA
RF30	El sistema permitirá descargar de manera masiva los XML y los reportes de cada uno de los comprobantes de pago electrónico emitidos de acuerdo al filtro propuesto en la interfaz.	ALTA
RF31	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Cliente.	ALTA
RF32	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Producto.	ALTA

RF33	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Tipo Cambio.	ALTA
RF34	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Moneda.	ALTA
RF35	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Usuario.	ALTA
RF36	El sistema permitirá generar reportes en base al indicador Nivel de incidencias en emisiones.	ALTA
RF37	El sistema permitirá generar reportes en base al Margen de Utilidad Operativa	ALTA
RF38	Solo el usuario no registrado podrá consultar la emisión de los comprobantes electrónicos si cumple los filtros de número de serie, numero de documento, RUC del Cliente, fecha de emisión y monto total del comprobante.	ALTA

Fuente: Elaboración propia

Tabla 15: Requerimientos No Funcionales

CODIGO	NIVEL	REQUERIMIENTOS NO FUNCIONALES	PRIORIDAD
RNF01	Seguridad	El acceso al sistema solo será para los usuarios registrados.	ALTA
RNF02	Disponibilidad	El sistema está disponible las 24 horas del día y 365 días del año.	
RNF03	Accesibilidad	El sistema es compatible con los navegadores: Google Chrome, Mozilla Firefox e Internet Explore.	ALTA
RNF04	Escalabilidad	El sistema permite la integración de nuevos requerimientos de acuerdo a las necesidades de la empresa.	ALTA
RNF05	Usabilidad	El sistema cuenta con interfaces amigables para la interacción con el usuario.	ALTA
RNF06	Usabilidad	El sistema posee mensajes de errores informativos y orientados al usuario.	ALTA
RNF07	Portabilidad	El sistema posee un diseño responsive con la finalidad de garantizar que la aplicación se adapte a cualquier dispositivo	ALTA

Fuente: Elaboración propia

2.3. Planeamiento del Sprint

2.3.1. Sprint Backlog

Tabla 16: General Sprint Backlog

SPRINT	REQUERIMIENTO	ESTIMACIÓN (DIAS)
SPRINT 0: Diseño del Sistema	Diseño de la Base de Datos y Casos de Uso. Diseño de Prototipos. RF01, RNF01, RNF05, RNF06, RNF07	30
SPRINT 1: Módulo de Administración	RF02, RF03, RF04, RF05, RF06, RF07	13
SPRINT 2: Módulo de Facturación	RF08, RF12, RF15, RF19, RF22, RF23, RF24, RF25, RF28, RF29, RF30, RF38	12
SPRINT 3: Módulo de Envío y/o Recepción de Comprobantes de pago electrónicos	RF09, RF10, RF13, RF14, RF16, RF17, RF18, RF20, RF21	8
SPRINT 4: Módulo de Reportes	RF11, RF26, RF27, RF31, RF32, RF33, RF34, RF35, RF36, RF37	12

Fuente: Elaboración propia

Figura 1. Diagrama Gant del Proyecto

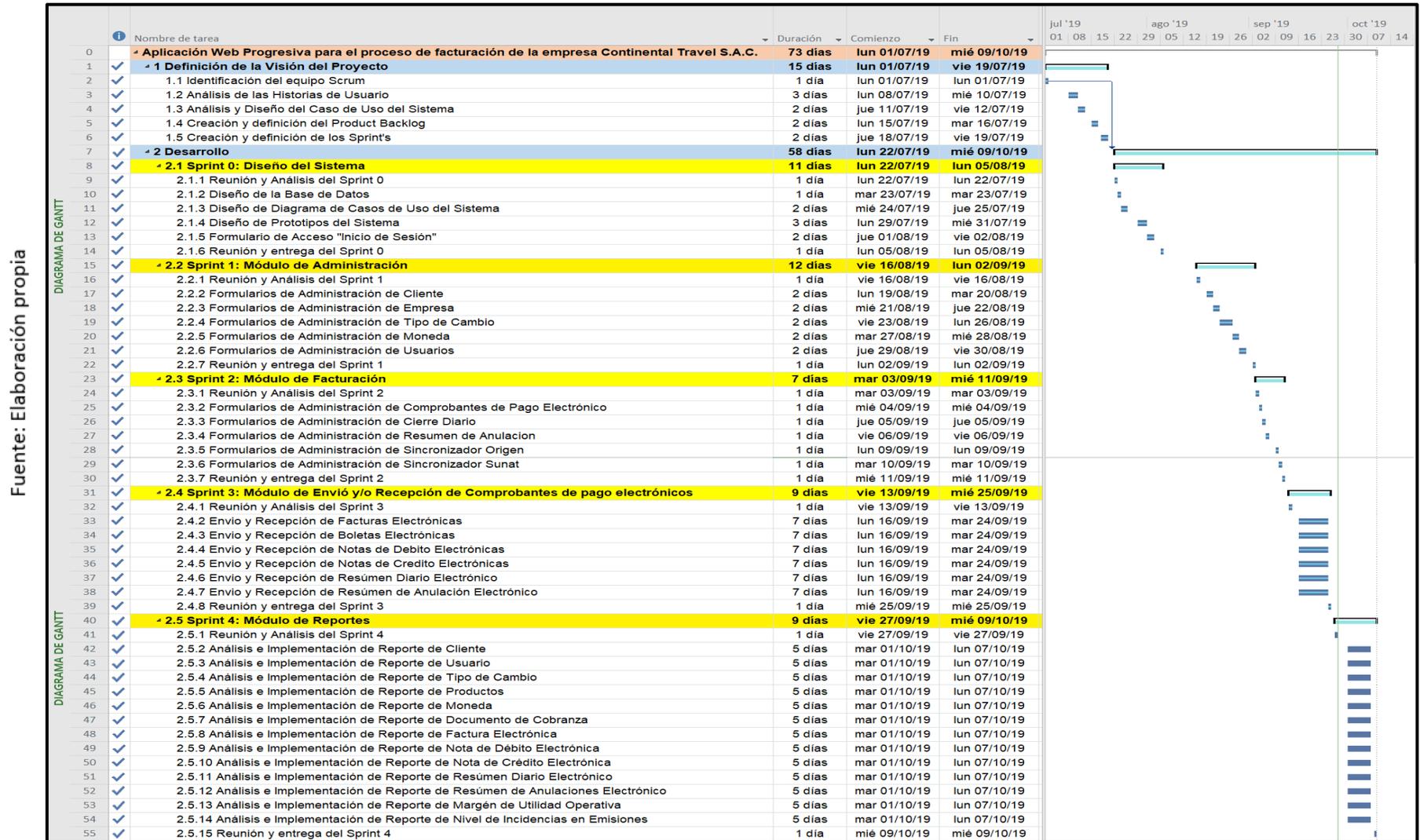


Tabla 17: Sprint Backlog

SPRINT	CODIGO	TAREA	ESTADO	TIPO	ROL
SPRINT 0	SP 0.1	Diseño de Casos de Uso del Sistema.	Finalizado	Desarrollo	TEAM
	SP 0.2	Diseño de la Base de Datos.	Finalizado	Desarrollo	TEAM
	SP 0.3	Desarrollo del Diccionario de Datos.			
	SP 0.4	Diseño de Prototipos del sistema.	Finalizado	Desarrollo	TEAM
	RF01	El sistema debe permitir la autenticación de usuarios, mediante un nombre de usuario y una contraseña.	Finalizado	Desarrollo	TEAM
	RNF01	El acceso al sistema solo será para los usuarios registrados.	Finalizado	Desarrollo	TEAM
	RNF05	El sistema cuenta con interfaces amigables para la interacción con el usuario.	Finalizado	Desarrollo	TEAM
	RNF06	El sistema posee mensajes de errores informativos y orientados al usuario.	Finalizado	Desarrollo	TEAM
	RNF07	El sistema posee un diseño responsive con la finalidad de garantizar que la aplicación se adapte a cualquier dispositivo	Finalizado	Desarrollo	TEAM
SPRINT 1	RF02	El sistema debe permitir listar, registrar, modificar y eliminar usuarios para gestionar el proceso de facturación electrónica.	Finalizado	Desarrollo	TEAM
	RF03	El sistema debe permitir listar y modificar los clientes relacionados al comprobante de pago electrónico emitido.	Finalizado	Desarrollo	TEAM
	RF04	El sistema debe permitir listar, modificar los datos de la empresa que emitirá la facturación electrónica.	Finalizado	Desarrollo	TEAM
	RF05	El sistema debe permitir listar y modificar los productos relacionados al comprobante de pago electrónico emitido.	Finalizado	Desarrollo	TEAM
	RF06	El sistema debe permitir listar, registrar y modificar tipo de cambio a la fecha a emitir.	Finalizado	Desarrollo	TEAM
	RF07	El sistema debe permitir listar, registrar y modificar moneda a emitir según el comprobante de pago electrónico generado.	Finalizado	Desarrollo	TEAM

SPRINT 2	RF08	El sistema permitirá filtrar los comprobantes de pago de acuerdo a la fecha de emisión, cliente, tipo de comprobante, estado , serie y numero de comprobante.	Finalizado	Desarrollo	TEAM
	RF12	El sistema permitirá enviar un XML y un reporte del comprobante de pago electrónico al Cliente, para su posterior confirmación.	Finalizado	Desarrollo	TEAM
	RF15	El sistema permitirá filtrar los cierres diarios generados en el día, mediante un filtro de fecha de emisión y estado del comprobante.	Finalizado	Desarrollo	TEAM
	RF19	El sistema permitirá listar los comprobantes de pago electrónicos relacionados a la factura que hayan sido anulados anteriormente, mediante un filtro de fecha de emisión y estado del comprobante.	Finalizado	Desarrollo	TEAM
	RF22	El sistema permitirá listar un historial de los comprobantes de pago electrónicos generados en el BackOffice.	Finalizado	Desarrollo	TEAM
	RF23	El sistema permitirá obtener información de un comprobante de pago electrónico generado en el BackOffice.	Finalizado	Desarrollo	TEAM
	RF24	El sistema permitirá listar un historial de los comprobantes de pago electrónicos enviados a la SUNAT.	Finalizado	Desarrollo	TEAM
	RF25	El sistema permitirá obtener información sobre el estado de un comprobante de pago electrónico enviado a la SUNAT.	Finalizado	Desarrollo	TEAM
	RF28	El sistema permitirá descargar los XML generados de los comprobantes de pago electrónicos.	Finalizado	Desarrollo	TEAM
RF29	El sistema permitirá exportar la información a un archivo de tipo Excel, que contengan datos de los comprobantes de pago electrónicos de acuerdo al filtro propuesto en la interfaz.	Finalizado	Desarrollo	TEAM	

	RF30	El sistema permitirá descargar de manera masiva los XML y los reportes de cada uno de los comprobantes de pago electrónico emitidos de acuerdo al filtro propuesto en la interfaz.	Finalizado	Desarrollo	TEAM
	RF38	Solo el usuario no registrado podrá consultarla emisión de los comprobantes electrónicos si cumple los filtros de número de serie, numero de documento, RUC del Cliente, fecha de emisión y monto total del comprobante.	Finalizado	Desarrollo	TEAM
SPRINT 3	RF09	El sistema permitirá generar un XML y enviar un comprobante de pago electrónico a la SUNAT.	Finalizado	Desarrollo	TEAM
	RF10	El sistema permitirá recepcionar el XML de respuesta por parte de la SUNAT e interpretar su estado actual.	Finalizado	Desarrollo	TEAM
	RF13	El sistema permitirá reenviar un XML de un comprobante de pago electrónico que haya presentado inconvenientes en su envió a la SUNAT.	Finalizado	Desarrollo	TEAM
	RF14	El sistema permitirá anular los comprobantes de pago electrónicos emitidos, y enviar un XML de anulación a la SUNAT.	Finalizado	Desarrollo	TEAM
	RF16	El sistema permitirá generar un XML y enviar un nuevo cierre diario de los comprobantes de pago relacionados a la Boleta, para sus posterior registro en la SUNAT.	Finalizado	Desarrollo	TEAM
	RF17	El sistema permitirá obtener un XML de respuesta del cierre diario de los comprobantes de pago electrónicos relacionados a la Boleta, para sus posterior validación.	Finalizado	Desarrollo	TEAM
	RF18	El sistema permitirá reenviar el XML de un cierre diario de los comprobantes de pago electrónicos relacionados a la Boleta, para sus posterior registro en la SUNAT.	Finalizado	Desarrollo	TEAM

	RF20	El Sistema permitirá obtener el XML de envío de un comprobante de pago electrónico anulado anteriormente.	Finalizado	Desarrollo	TEAM
	RF21	El Sistema permitirá obtener XML de respuesta de un comprobante de pago electrónico anulado anteriormente.	Finalizado	Desarrollo	TEAM
SPRINT 4	RF11	El Sistema permitirá generar un reporte individual por cada comprobante de pago electrónico generado.	Finalizado	Desarrollo	TEAM
	RF26	El sistema permitirá generar reportes mensuales de los comprobantes de pago electrónicos anulados.	Finalizado	Desarrollo	TEAM
	RF27	El sistema permitirá generar reportes mensuales de los cierres diarios generados de acuerdo comprobantes de pago electrónicos emitidos durante el día.	Finalizado	Desarrollo	TEAM
	RF31	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Cliente.	Finalizado	Desarrollo	TEAM
	RF32	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Producto.	Finalizado	Desarrollo	TEAM
	RF33	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Tipo Cambio.	Finalizado	Desarrollo	TEAM
	RF34	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Moneda.	Finalizado	Desarrollo	TEAM
	RF35	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Usuario.	Finalizado	Desarrollo	TEAM
	RF36	El sistema permitirá generar reportes en base al indicador Nivel de incidencias en emisiones.	Finalizado	Desarrollo	TEAM
	RF37	El sistema permitirá generar reportes en base al Tasa de entregas completas y puntuales.	Finalizado	Desarrollo	TEAM

Fuente: Elaboración propia

3. DESARROLLO DEL SPRINT

3.1. Desarrollo del Sprint 0

Tabla 18: Sprint 0 – Análisis y Diseño del Sistema

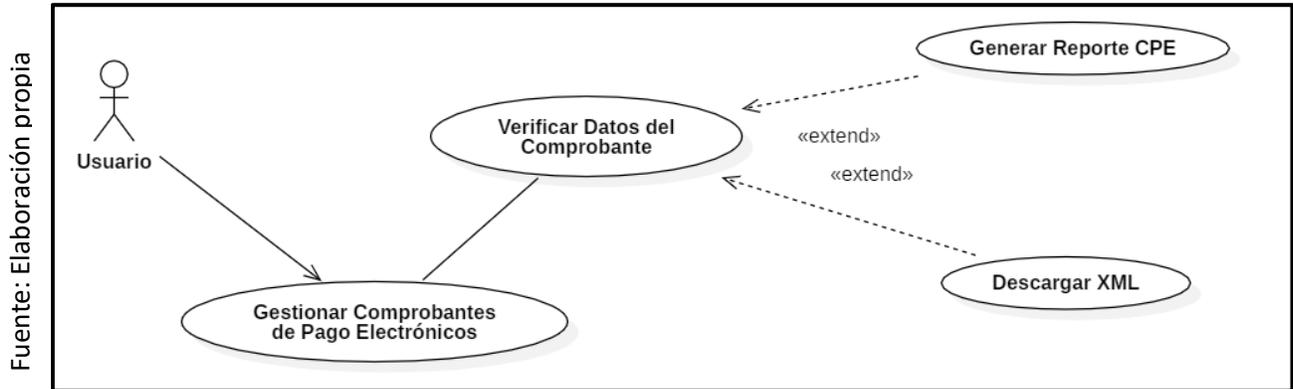
SPRINT	CODIGO	TAREA	ESTIMACIÓN (DIAS)	PRIORIDAD
SPRINT 0	SP 0.1	Diseño de Casos de Uso del Sistema.	1	ALTA
	SP 0.2	Diseño de la Base de Datos.	2	ALTA
	SP 0.3	Desarrollo del Diccionario de Datos.		
	SP 0.4	Diseño de Prototipos del sistema.	3	ALTA
	RF01	El sistema debe permitir la autenticación de usuarios, mediante un nombre de usuario y una contraseña.	2	ALTA
	RNF01	El acceso al sistema solo será para los usuarios registrados.	2	ALTA
	RNF05	El sistema cuenta con interfaces amigables para la interacción con el usuario.	2	ALTA
	RNF06	El sistema posee mensajes de errores informativos y orientados al usuario.	2	ALTA
	RNF07	El sistema posee un diseño responsive con la finalidad de garantizar que la aplicación se adapte a cualquier dispositivo	2	ALTA

Fuente: Elaboración propia

SP 0.1: Diseño de Casos de Uso del Sistema

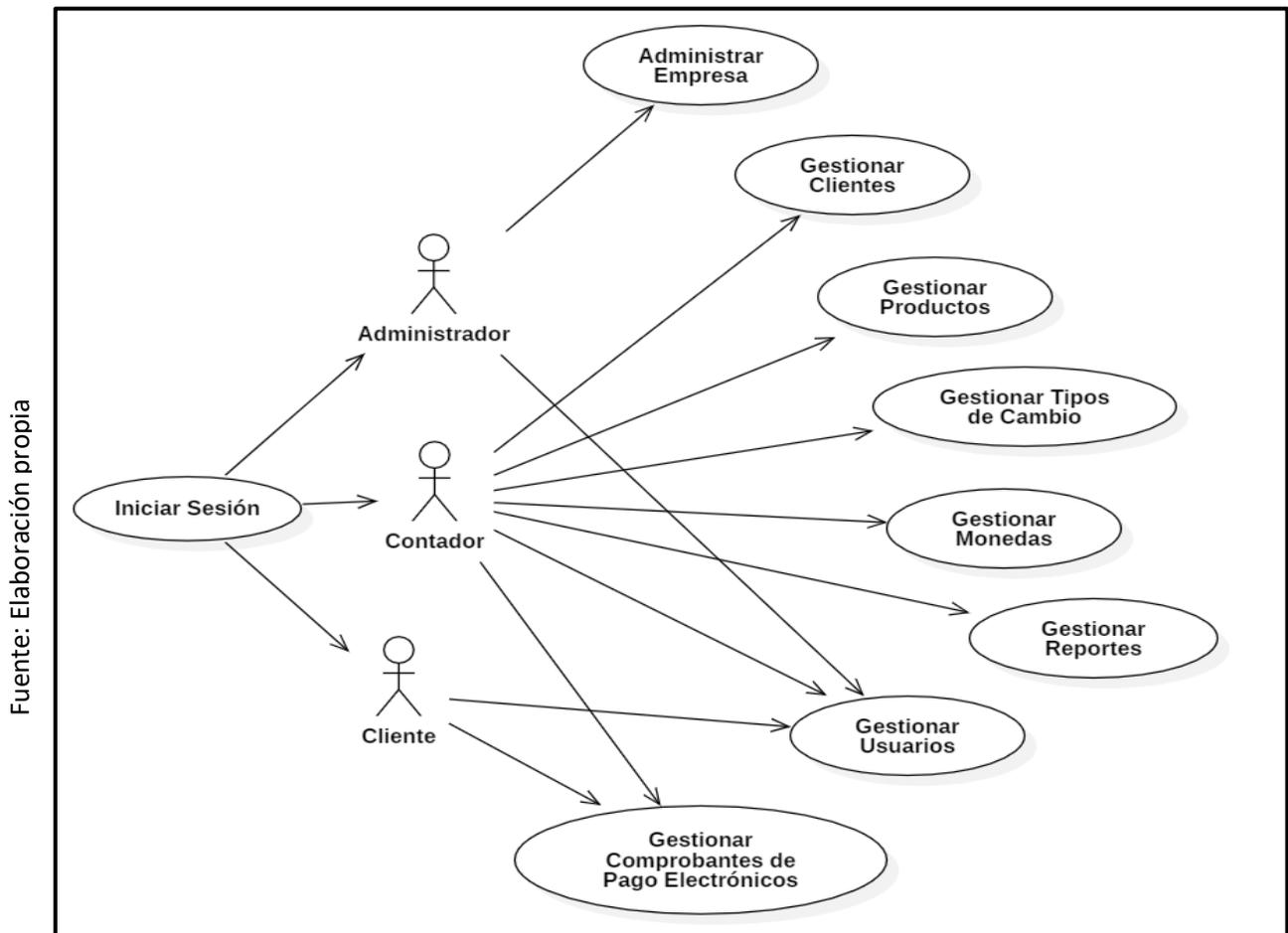
Caso de Uso – Extranet

Figura 2: Diagrama de Caso de Uso de Extranet



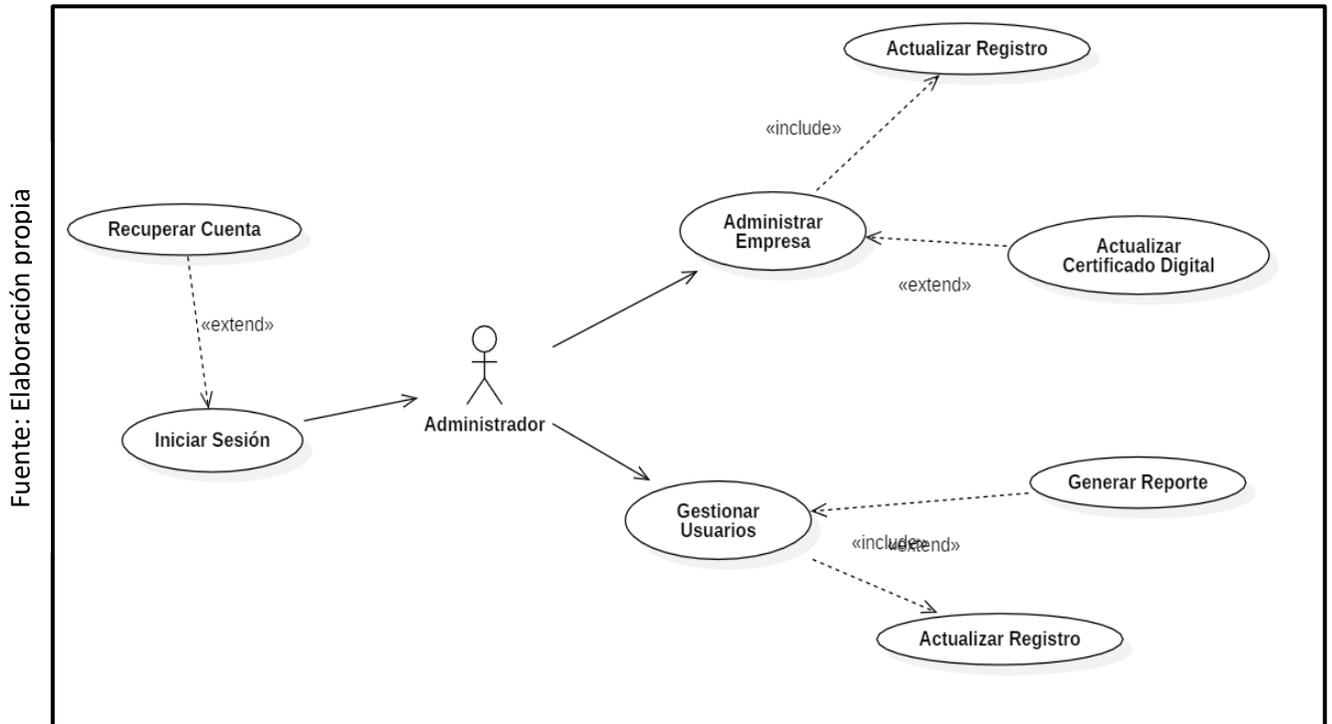
Caso de Uso – Intranet

Figura 3: Diagrama de Caso de Uso de Intranet



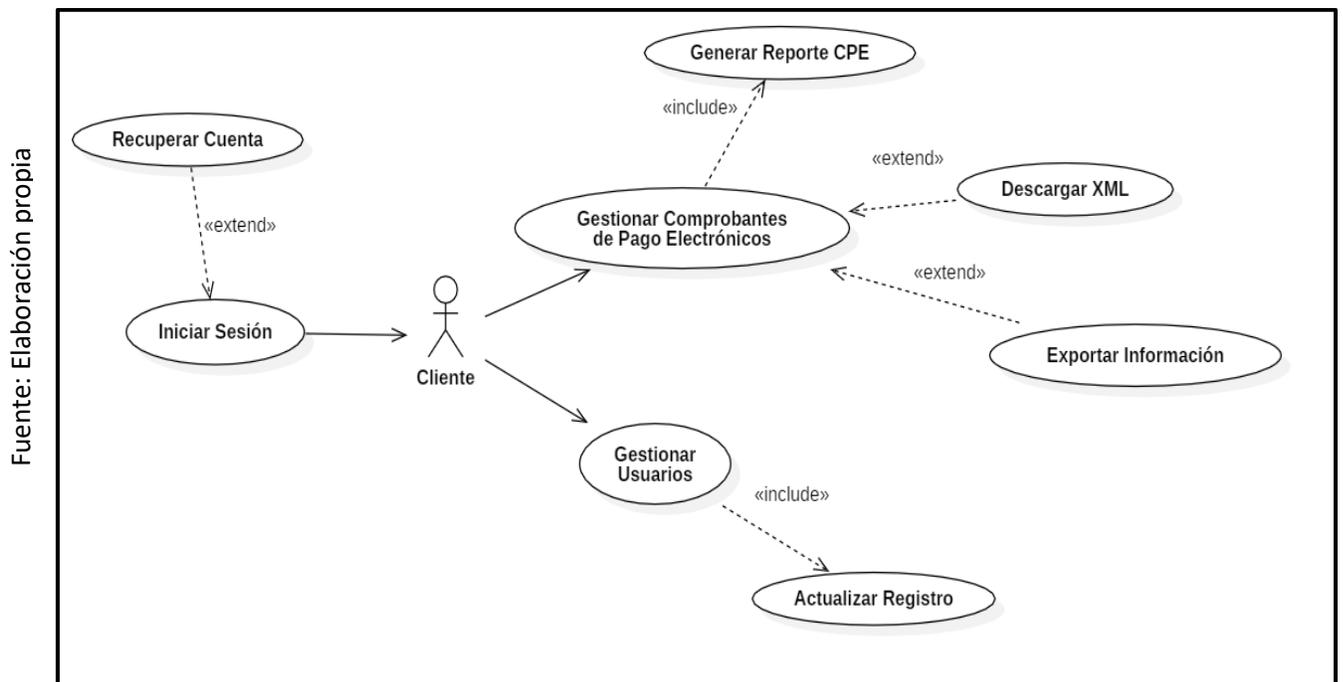
Caso de Uso – Intranet – Administrador

Figura 4: Caso de Uso del Administrador



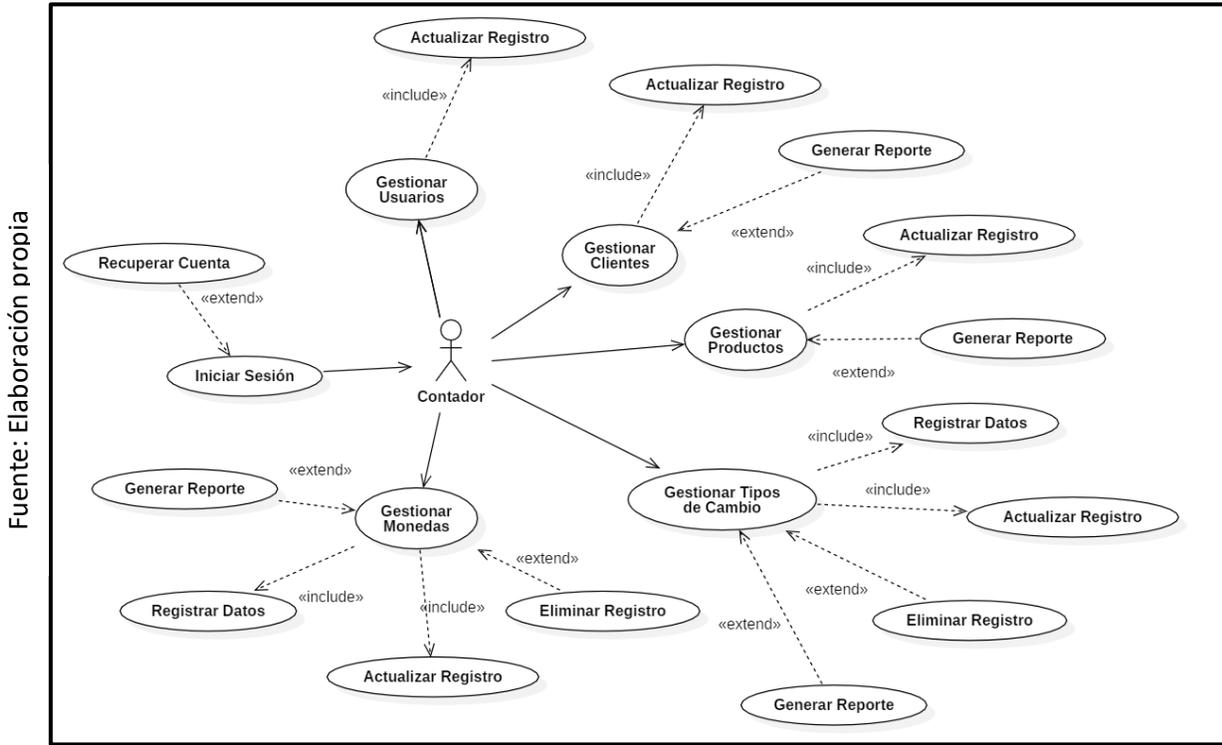
Caso de Uso – Intranet – Cliente

Figura 5: Caso de Uso del Cliente



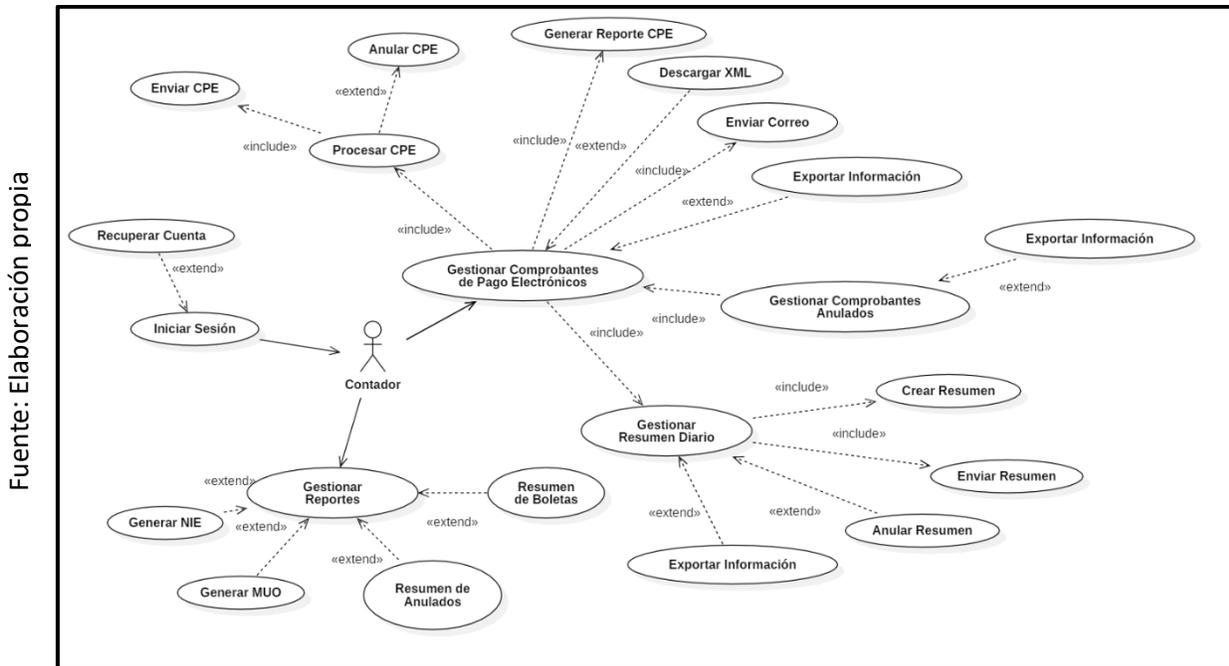
Caso de Uso – Intranet – Contador

Figura 6: Caso de Uso del Contador – Parte 1



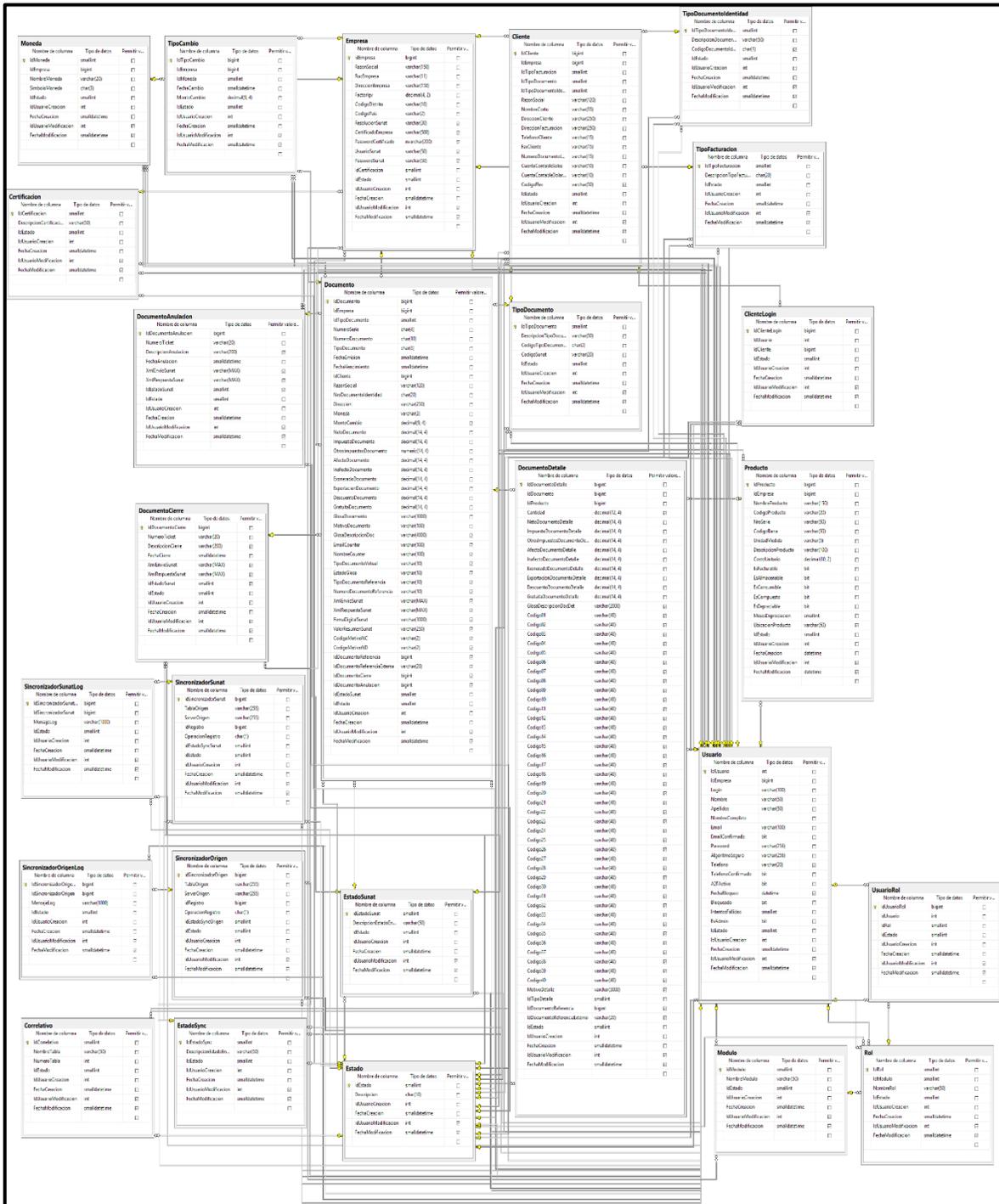
Caso de Uso del Usuario

Figura 7: Caso de Uso del Contador – Parte 2



SP 0.2: Diseño de la Base de Datos

Figura 8: Modelo Físico de la Base de Datos



Fuente: Elaboración propia

SP 0.3: Desarrollo del Diccionario de Datos

La Aplicación Web Progresiva para el proceso de facturación consta de 25 tablas, las cuales detallare en las siguientes tablas.

Tabla 19: Tabla Certificación

Columna	Descripción	Tipo	Nulo	Observaciones
IdCertificacion	Código de Certificación	smallint	No	Clave Primaria
DescripcionCertificacion	Descripción de la Certificación	varchar(50)	No	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 20: Tabla Cliente

Columna	Descripción	Tipo	Nulo	Observaciones
IdCliente	Código de Cliente	bigint	No	Clave Primaria
IdEmpresa	Código de Empresa	bigint	No	Clave Foránea proveniente de la tabla Empresa
IdTipoFacturacion	Código de Tipo de Facturacion	smallint	No	Clave Foránea proveniente de la tabla TipoFacturacion
IdTipoDocumento	Código de Tipo de Documento	smallint	No	Clave Foránea proveniente de la tabla TipoDocumento
IdTipoDocumentoIdentidad	Código de Tipo de Documento de Identidad	varchar(120)	No	Clave Foránea proveniente de la tabla TipoDocumentoIdentidad

RazonSocial	Razón Social del Cliente	varchar(35)	No	
NombreCorto	Nombre Corto del Cliente	varchar(250)	No	
DireccionCliente	Dirección de Cliente	varchar(250)	No	
DireccionFacturacion	Dirección de Facturación del Cliente	varchar(15)	No	
TelefonoCliente	Teléfono del Cliente	varchar(15)	No	
FaxCliente	Fax del Cliente	varchar(15)	No	
NumeroDocumentoIdentidad	Numero de Documento de Identidad	varchar(15)	No	
CuentaContableSoles	Número de Cuenta Contable en Soles	varchar(15)	No	
CuentaContableDolares	Número de Cuenta Contable Dólares	varchar(15)	No	
CodigoFlex	Código Flexline	varchar(15)	Si	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Tbl_Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Tbl_Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 21: Tabla ClienteLogin

Columna	Descripción	Tipo	Nulo	Observaciones
IdClienteLogin	Código de Cliente Login	bigint	No	Clave Primaria
IdUsuario	Código de Usuario relacionado al Cliente	int	No	Clave Foránea proveniente de la tabla Usuario
IdCliente	Código de Cliente	bigint	No	Clave Foránea proveniente de la tabla Cliente
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado

IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 22: Tabla Documento

Columna	Descripción	Tipo	Nulo	Observaciones
IdDocumento	Código de Documento	bigint	No	Clave Primaria
IdEmpresa	Código de Empresa	bigint	No	Clave Foránea proveniente de la tabla Empresa
IdTipoDocumento	Código de Tipo de Documento	smallint	No	Clave Foránea proveniente de la tabla TipoDocumento
NumeroSerie	Número de Serie del Comprobante	char(4)	No	
NumeroDocumento	Número de Comprobante	char(10)	No	
TipoDocumento	Tipo de Comprobante	char(3)	No	
FechaEmision	Fecha de Emisión de Comprobante	smalldatetime	No	
FechaVencimiento	Fecha de Vencimiento de Comprobante	smalldatetime	No	
IdCliente	Código de Cliente relacionado al Documento	bigint	No	Clave Foránea proveniente de la tabla Cliente
RazonSocial	Razón Social del Cliente	varchar(120)	No	
NroDocumentoidentidad	Numero de Documento de Identidad del Cliente	char(20)	No	
Direccion	Dirección del Cliente	varchar(250)	No	
Moneda	Tipo de Moneda emitido en el Comprobante	varchar(3)	No	

MontoCambio	Monto de Cambio del Comprobante	decimal(14,4)	Si	
NetoDocumento	Monto Neto del Comprobante	decimal(9,4)	No	
ImpuestoDocumento	Monto Impuesto del Comprobante	decimal(9,4)	No	
OtrosImpuestoDocumento	Monto Otros Impuesto del Comprobante	decimal(9,4)	No	
AfectoDocumento	Monto Afecto del Comprobante	decimal(9,4)	No	
InafectoDocumento	Monto Inafecto del Comprobante	decimal(9,4)	No	
ExoneradoDocumento	Monto Exonerado del Comprobante	decimal(9,4)	No	
ExportacionDocumento	Monto Exportación del Comprobante	decimal(9,4)	No	
DescuentoDocumento	Monto Descuento del Comprobante	decimal(9,4)	No	
GratisDocumento	Monto Gratuito del Comprobante	decimal(9,4)	No	
GlosaDocumento	Glosa del Documento emitido	varchar(255)	No	
MotivoDocumento	Motivo del Documento emitido	varchar(100)	No	
GlosaDescripcionDoc	Contiene más Información de la Glosa del Documento Emitido	varchar(MAX)	Si	
EmailCounter	Email del Counter	varchar(100)	Si	
NombreCounter	Nombre del Counter	varchar(100)	Si	
TipoDocumentoVirtual	Tipo de Documento a Imprimir	varchar(10)	Si	
EstadoGlosa	Estado de la Glosa a imprimir	varchar(10)	Si	
TipoReferenciaDocumento	Tipo de Documento a la que hace referencia solo si es Nota de Débito o Crédito	varchar(10)	Si	
NumeroReferenciaDocumento	Número de Documento a la que hace referencia solo si es Nota de Débito o Crédito	varchar(10)	Si	

XmlEnvioSunat	Contiene el XML que envía a la SUNAT	varchar(MAX)	Si	
XmlRespuestaSunat	Contiene el XML que recepciona de la SUNAT	varchar(MAX)	Si	
FirmaDigitalSunat	Contiene la firma contenida en el XML que envía a la SUNAT	varchar(MAX)	Si	
ValorResumenSunat	Contiene el resumen encriptado que se recepciona de la SUNAT	varchar(250)	Si	
CodigoMotivoNC	Código del Motivo de una Nota de Crédito según SUNAT	varchar(2)	Si	
CodigoMotivoND	Código del Motivo de una Nota de Débito según SUNAT	varchar(2)	Si	
IdDocumentoReferencia	Código del Documento que se relaciona al Comprobante solo si es Nota de Crédito o Débito	bigint	Si	Clave Foránea proveniente de la tabla Documento
IdDocumentoReferenciaExterna	Código del Documento Importado	varchar(20)	Si	
IdDocumentoCierre	Código del Cierre Diario a la que se relaciona	bigint	Si	Clave Foránea proveniente de la tabla DocumentoCierre
IdDocumentoAnulacion	Código del Resumen de Anulación a la que se relaciona	bigint	Si	Clave Foránea proveniente de la tabla DocumentoAnulacion
IdEstadoSunat	Código de Estado SUNAT a la que se relaciona el Comprobante	smallint	No	Clave Foránea proveniente de la tabla EstadoSunat
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado

IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Tbl_Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Tbl_Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 23: Tabla DocumentoDetalle

Columna	Descripción	Tipo	Nulo	Observaciones
IdDocumentoDetalle	Código de Documento Detalle	bigint	No	Clave Primaria
IdDocumento	Código de Documento relacionado al Documento Detalle	bigint	No	Clave Foránea proveniente de la tabla Documento
IdProducto	Código de Producto relacionado al Documento	smallint	No	Clave Foránea proveniente de la tabla Producto
Cantidad	Cantidad de Productos	decimal(12,4)	No	
NetoDocumentoDetalle	Monto Neto del Detalle del Comprobante	decimal(14,4)	No	
ImpuestoDocumentoDetalle	Monto Impuesto del Detalle del Comprobante	decimal(14,4)	No	
OtrosImpuestoDocumentoDetalle	Monto Otros Impuesto del Detalle del Comprobante	decimal(14,4)	No	
AfectoDocumentoDetalle	Monto Afecto del Comprobante	decimal(14,4)	No	
InafectoDocumentoDetalle	Monto Inafecto del Detalle del Comprobante	decimal(14,4)	No	
ExoneradoDocumentoDetalle	Monto Exonerado del Detalle del Comprobante	decimal(14,4)	No	

ExportacionDocumentoDetalle	Monto Exportación del Detalle del Comprobante	decimal(14,4)	No	
DescuentoDocumentoDetalle	Monto Descuento del Detalle del Comprobante	decimal(14,4)	No	
GratisDocumentoDetalle	Monto Gratuito del Detalle del Comprobante	decimal(14,4)	No	
GlosaDescripcionDocDet	Contiene más Información de la Glosa del Documento Detalle Emitido	varchar(MAX)	Si	
Codigo01	Descripción del Tipo de Servicio	varchar(40)	Si	
Codigo02	Código del Tipo de Servicio	varchar(40)	Si	
Codigo03	Razón Social de la Aerolínea	varchar(40)	Si	
Codigo04	Código IATA de la Aerolínea	varchar(40)	Si	
Codigo05	Sigla IATA de la Aerolínea	varchar(40)	Si	
Codigo06	Nombre del Proveedor del Ticket	varchar(40)	Si	
Codigo07	Descripción del Área de donde se originó el Ticket	varchar(40)	Si	
Codigo08	Nombre de Counter	varchar(40)	Si	
Codigo09	Código del Counter	varchar(40)	Si	
Codigo10	Nombre del GDS	varchar(40)	Si	
Codigo11	Descripción del Tipo del Proveedor	varchar(40)	Si	
Codigo12	Código IATA	varchar(40)	Si	
Codigo13	Código Agente MPD	varchar(40)	Si	
Codigo14	Código del Servicio Adquirido	varchar(40)	Si	
Codigo15	Fecha de Reserva	varchar(40)	Si	
Codigo16	Código de Reserva	varchar(40)	Si	
Codigo17	Numero de Ticket	varchar(40)	Si	
Codigo18	Tipo de Pasajero	varchar(40)	Si	
Codigo19	Nombre del Pasajero	varchar(40)	Si	
Codigo20	Numero File	varchar(40)	Si	
Codigo21	Numero de Pasajero	varchar(40)	Si	

Codigo22	Nombre del Solicitante del Ticket	varchar(40)	Si	
Codigo23	Ruta del Ticket	varchar(40)	Si	
Codigo24	Destino del Ticket	varchar(40)	Si	
Codigo25	Tipo de Cambio	varchar(40)	Si	
Codigo26	Tipo de Tarifa	varchar(40)	Si	
Codigo27	Monto Comisión	varchar(40)	Si	
Codigo28	Código de Conjunción de Ticket	varchar(40)	Si	
Codigo29	Fecha In (Salida a Destino)	varchar(40)	Si	
Codigo30	Fecha Out (Llegada a Destino)	varchar(40)	Si	
Codigo31	Motivo del Ticket Solicitado	varchar(40)	Si	
Codigo32	Descripción del Tipo de Documento Emitido	varchar(40)	Si	
Codigo33	Descripción del Tipo de Documento Procesado	varchar(40)	Si	
Codigo34	Código de la Ciudad de Salida de la Ruta	varchar(40)	Si	
Codigo35	Cantidad de Vuelos del Ticket	varchar(40)	Si	
Codigo36	Fecha Salida del Vuelo de la Ruta	varchar(40)	Si	
Codigo37	Fecha Llegada del Vuelo de la Ruta	varchar(40)	Si	
Codigo38	Nombre de la Aerolínea de la Ruta	varchar(40)	Si	
Codigo39	Código de Vuelo	varchar(40)	Si	
Codigo40	Descripción del Itinerario	varchar(MAX)	Si	
MotivoDetalle	Motivo del Documento emitido	varchar(MAX)	Si	
IdTipoDetalle	Contiene más Información de la Glosa del Documento Emitido	smallint	Si	

IdDocumentoReferencia	Código del Documento Importado	bigint	Si	
IdDocumentoReferenciaExterno	Código del DocumentoDetalle Importado	varchar(20)	Si	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Tbl_Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Tbl_Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 24: Tabla DocumentoAnulacion

Columna	Descripción	Tipo	Nulo	Observaciones
IdDocumentoAnulacion	Código de Documento Anulación	bigint	No	Clave Primaria
NumeroTicket	Numero de Ticket asignado por la SUNAT	varchar(20)	No	
DescripcionAnulacion	Descripción de la Anulación asignado por tipo de Operación	varchar(200)	Si	
FechaAnulacion	Fecha en que se realiza el cierre	smalldatetime	No	
XmlEnvioSunat	Contiene la data del XML enviado	varchar(MAX)	Si	
XmlRespuestaSunat	Contiene la data del XML recibido	varchar(MAX)	Si	
IdEstadoSunat	Código de Estado Sunat relacionado al Documento Cierre	smallint	Si	Clave Foránea proveniente de la tabla EstadoSunat

IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 25: Tabla DocumentoCierre

Columna	Descripción	Tipo	Nulo	Observaciones
IdDocumentoCierre	Código de Documento Cierre	bigint	No	Clave Primaria
NumeroTicket	Numero de Ticket asignado por la SUNAT	varchar(20)	No	
DescripcionCierre	Descripción de Cierre asignado por tipo de Operación	varchar(200)	Si	
FechaCierre	Fecha en que se realiza el cierre	smalldatetime	No	
XmlEnvioSunat	Contiene la data del XML enviado	varchar(MAX)	Si	
XmlRespuestaSunat	Contiene la data del XML recibido	varchar(MAX)	Si	
IdEstadoSunat	Código de Estado Sunat relacionado al Documento Cierre	smallint	Si	Clave Foránea proveniente de la tabla EstadoSunat
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario

FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	
-------------------	---------------------------------------	---------------	----	--

Fuente: Elaboración propia

Tabla 26: Tabla Empresa

Columna	Descripción	Tipo	Nulo	Observaciones
IdEmpresa	Código de Empresa	bigint	No	Clave Primaria
RazonSocial	Razón Social de la Empresa	varchar(150)	No	
RucEmpresa	Numero RUC de la Empresa	varchar(11)	No	
DireccionEmpresa	Dirección de la Empresa	varchar(150)	No	
FactorIgv	Monto IGV de la Empresa	decimal(4,2)	No	
CodigoDistrito	Código de Distrito donde se ubica la empresa	varchar(10)	No	
CodigoPais	Código del País donde se ubica la Empresa	varchar(2)	No	
ResolucionSunat	Numero de Resolución Sunat donde indique a la empresa que tipo de contribuyente ha sido marcado.	varchar(30)	Si	
CertificadoEmpresa	Descripción de la ubicación del certificado SUNAT de Tipo PFX	varchar(500)	Si	
PasswordCertificado	Contraseña del Certificado SUNAT	varchar(200)	Si	
UsuarioSunat	Usuario SUNAT	varchar(50)	Si	
PasswordSunat	Password SUNAT	varchar(50)	Si	
IdCertificacion	Código de Certificado relacionado a la Empresa	smallint	No	Clave Foránea proveniente de la tabla Certificacion
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	

IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 27: Tabla Estado

Columna	Descripción	Tipo	Nulo	Observaciones
IdEstado	Código de Estado	smallint	No	Clave Primaria
DescripcionEstado	Descripción del Estado	char(10)	No	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 28: Tabla EstadoSunat

Columna	Descripción	Tipo	Nulo	Observaciones
IdEstadoSunat	Código de Estado Sunat	smallint	No	Clave Primaria
DescripcionEstadoEnvioSunat	Descripción del estado de Envío a la SUNAT	varchar(50)	No	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario

FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	
-------------------	---------------------------------------	---------------	----	--

Fuente: Elaboración propia

Tabla 29: Tabla EstadoSync

Columna	Descripción	Tipo	Nulo	Observaciones
IdEstadoSync	Código de Estado Sincronización	smallint	No	Clave Primaria
DescripcionEstadoSync	Descripción del estado de Sincronización de importación	varchar(50)	No	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 30: Tabla Modulo

Columna	Descripción	Tipo	Nulo	Observaciones
IdModulo	Código de Modulo	smallint	No	Clave Primaria
NombreModulo	Nombre del Modulo	varchar(50)	No	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 31: Tabla Moneda

Columna	Descripción	Tipo	Nulo	Observaciones
IdMoneda	Código de Moneda	smallint	No	Clave Primaria
IdEmpresa	Código de Empresa relacionado al Usuario	bigint	No	Clave Foránea proveniente de la tabla Empresa
NombreMoneda	Nombre de la Moneda	varchar(20)	No	
SimboloMoneda	Símbolo de la Moneda	char(3)	No	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 32: Tabla Producto

Columna	Descripción	Tipo	Nulo	Observaciones
IdProducto	Código de Producto	bigint	No	Clave Primaria
IdEmpresa	Código de Empresa relacionado al Usuario	bigint	No	Clave Foránea proveniente de la tabla Empresa
NombreProducto	Nombre del Producto	varchar(150)	No	
CodigoProducto	Código del Producto	varchar(20)	No	
NroSerie	Número de Serie del Producto	varchar(50)	No	
CodigoBarra	Código de Barra del Producto	varchar(50)	No	
UnidadMedida	Unidad de Medida del Producto	varchar(5)	No	
DescripcionProducto	Descripción del Producto	varchar(100)	No	
CostoUnitario	Costo Unitario del Producto	decimal(10,2)	No	

EsFacturable	Si el Producto es Facturable	bit	No	
EsAlmacenable	Si el Producto es Almacenable	bit	No	
EsConsumible	Si el Producto es Consumible	bit	No	
EsCompuesto	Si el Producto es Compuesto por otro Producto	bit	No	
EsDepreciable	Si el Producto es Depreciable	bit	No	
MesesDepreciacion	Cantidad de Meses de Depreciación del Producto	smallint	No	
UbicacionProducto	Descripción de la Ubicación del Producto	varchar(50)	Si	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 33: Tabla Rol

Columna	Descripción	Tipo	Nulo	Observaciones
IdRol	Código de Rol	smallint	No	Clave Primaria
IdModulo	Código de Modulo relacionado al Rol	smallint	No	Clave Foránea proveniente de la tabla Modulo
NombreRol	Descripción del Rol	varchar(50)	No	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	

IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 34: Tabla SincronizadorOrigen

Columna	Descripción	Tipo	Nulo	Observaciones
IdSincronizadorOrigen	Código Sincronizador Sunat Log	bigint	No	Clave Primaria
TablaOrigen	Nombre de la Tabla a realizar la operación	varchar(255)	No	
ServerOrigen	Nombre del medio de importación	varchar(255)	No	
IdRegistro	Código del Registro Importado	bigint	No	
OperacionRegistro	Código de Tipo de Registro	char(1)	No	
IdEstadoSyncOrigen	Código del Estado Sincronizador a realizar en el Sistema	smallint	No	Clave Foránea proveniente de la tabla EstadoSync
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 35: Tabla SincronizaodrOrigenLog

Columna	Descripción	Tipo	Nulo	Observaciones
IdSincronizadorOrigen Log	Código Sincronizador Origen Log	bigint	No	Clave Primaria
IdSincronizadorOrigen	Código de Sincronizador Origen relacionado a sus detalles	bigint	No	Clave Foránea proveniente de la tabla SincronizadorOrigen
MensajeLog	Mensaje del Log	varchar(1000)	No	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 36: Tabla SincronizadorSunat

Columna	Descripción	Tipo	Nulo	Observaciones
IdSincronizadorSunat	Código Sincronizador Sunat Log	bigint	No	Clave Primaria
TablaOrigen	Nombre de la Tabla a realizar la operacion	varchar(255)	No	
ServerOrigen	Nombre del medio de a operar	varchar(255)	No	
IdRegistro	Codigo del Registro Importado	bigint	No	
OperacionRegistro	Codigo de Tipo de Registro	char(1)	No	
IdEstadoSyncSunat	Codigo del Estado Sincronizador a realizar en la SUNAT	smallint	No	Clave Foránea proveniente de la tabla EstadoSync
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado

IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 37: Tabla SincronizadorSunatLog

Columna	Descripción	Tipo	Nulo	Observaciones
IdSincronizadorSunatLog	Código Sincronizador Sunat Log	bigint	No	Clave Primaria
IdSincronizadorSunat	Código de Sincronizador Sunat relacionado a sus detalles	bigint	No	Clave Foránea proveniente de la tabla SincronizadorSunat
MensajeLog	Mensaje del Log	varchar(1000)	No	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 38: Tabla TipoCambio

Columna	Descripción	Tipo	Nulo	Observaciones
IdTipoCambio	Código de Tipo de Cambio	int	No	Clave Primaria
IdEmpresa	Código de Empresa relacionado al Tipo de Cambio	bigint	No	Clave Foránea proveniente de la tabla Empresa
IdMoneda	Código de Moneda relacionado al Tipo de Cambio	smallint	No	Clave Foránea proveniente de la tabla Moneda
FechaCambio	Fecha de Tipo de Cambio	smalldatetime	No	
MontoCambio	Monto de Tipo de Cambio	decimal(9,4)	No	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 39: Tabla TipoDocumento

Columna	Descripción	Tipo	Nulo	Observaciones
IdTipoDocumento	Código de Tipo de Documento	smallint	No	Clave Primaria
DescripcionTipoDocumento	Descripción del Tipo de Documento	varchar(30)	No	
CodigoTipoDocumento	Código de Tipo de Documento	char(2)	No	
CodigoSunat	Código de Tipo de Documento según SUNAT	varchar(20)	No	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario

FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 40: Tabla TipoDocumentoidentidad

Columna	Descripción	Tipo	Nulo	Observaciones
IdTipoDocumentoidentidad	Código de Tipo de Documento de Identidad	smallint	No	Clave Primaria
DescripcionDocumentoidentidad	Descripción del Tipo de Documento de Identidad	varchar(50)	No	
CodigoDocumentoidentidad	Código de Tipo de Documento de Identidad según SUNAT	char(1)	Si	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 41: Tabla TipoFacturacion

Columna	Descripción	Tipo	Nulo	Observaciones
IdTipoFacturacion	Código de Tipo Facturacion	smallint	No	Clave Primaria
DescripcionTipoFacturacion	Descripción del Tipo de Facturación	char(20)	No	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado

IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 42: Tabla Usuario

Columna	Descripción	Tipo	Nulo	Observaciones
IdUsuario	Código de Usuario	smallint	No	Clave Primaria
IdEmpresa	Código de Empresa relacionado al Usuario	bigint	No	Clave Foránea proveniente de la tabla Empresa
Login	Nombre de Login	varchar(100)	No	
Nombre	Nombres del Usuario	varchar(50)	No	
Apellidos	Apellidos del Usuario	varchar(50)	No	
NombreCompleto	Nombre Completo del Usuario		No	Autogenerado
Email	Email del Usuario	varchar(100)	No	
EmailConfirmado	Si el Email del Usuario ha sido confirmado	bit	No	
Password	Contraseña del Usuario con encriptación md5	varchar(256)	No	
AlgoritmoSeguro	Algoritmo usado para la encriptación de la contraseña	varchar(256)	No	
Telefono	Teléfono del Usuario	varchar(20)	Si	
TelefonoConfirmado	Si el Teléfono del Usuario ha sido confirmado	bit	No	
A2FActivo	Si el Tipo de autenticación A2F está activa	bit	No	
FechaBloqueo	Fecha de Bloqueo de sesión	datetime	Si	
Bloqueado	Si el Usuario está Bloqueado	bit	No	

IntentosFallidos	Numero de Intentos fallidos al iniciar la sesión	smallint	No	
EsAdmin	Si el Usuario es Administrador	bit	No	
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

Tabla 43: Tabla UsuarioRol

Columna	Descripción	Tipo	Nulo	Observaciones
IdUsuarioRol	Código de Usuario Rol	smallint	No	Clave Primaria
IdUsuario	Código de Usuario relacionado al Rol	int	No	Clave Foránea proveniente de la tabla Usuario
IdRol	Código de Rol	smallint	No	Clave Foránea proveniente de la tabla Rol
IdEstado	Código de Estado	smallint	No	Clave Foránea proveniente de la tabla Estado
IdUsuarioCreacion	Código de Usuario que ha creado la data	int	No	Clave Foránea proveniente de la tabla Usuario
FechaCreacion	Fecha en que se ha creado la data	smalldatetime	No	
IdUsuarioModificacion	Código de Usuario que ha modificado la data	int	Si	Clave Foránea proveniente de la tabla Usuario
FechaModificacion	Fecha en que se ha modificado la data	smalldatetime	Si	

Fuente: Elaboración propia

SP 0.3: Diseño de Prototipos del sistema

Para la presentación de la Aplicación Web Progresiva al Product Owner, se preparó una serie de prototipos del software propuesto; con el fin de obtener su visto bueno e inicie el proceso de desarrollo del software propuesto, además de corregir e implementar las observaciones del software presentadas en la reunión.

Figura 9: Vista Principal del Sistema

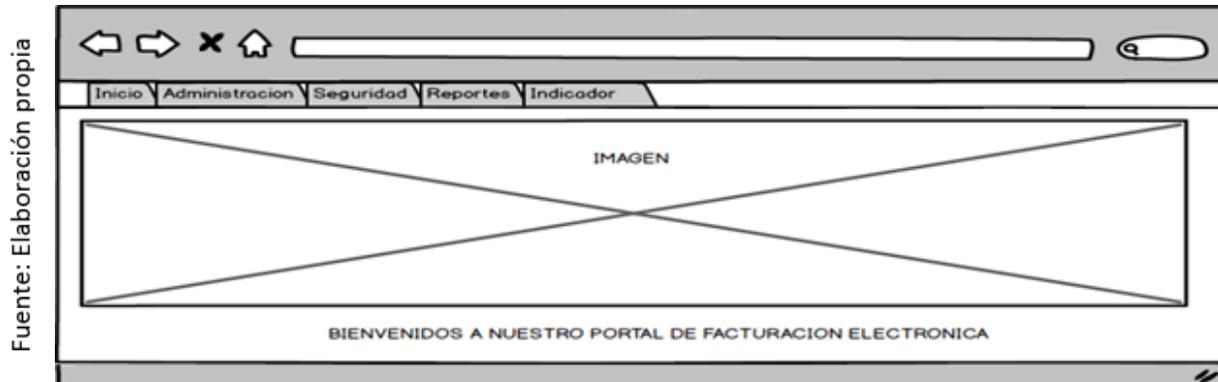


Figura 10: Login del Sistema

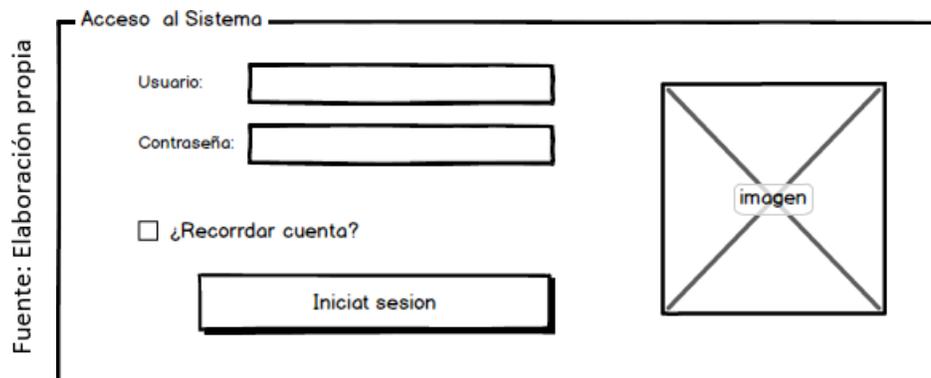


Figura 11. Vista Crear Usuario

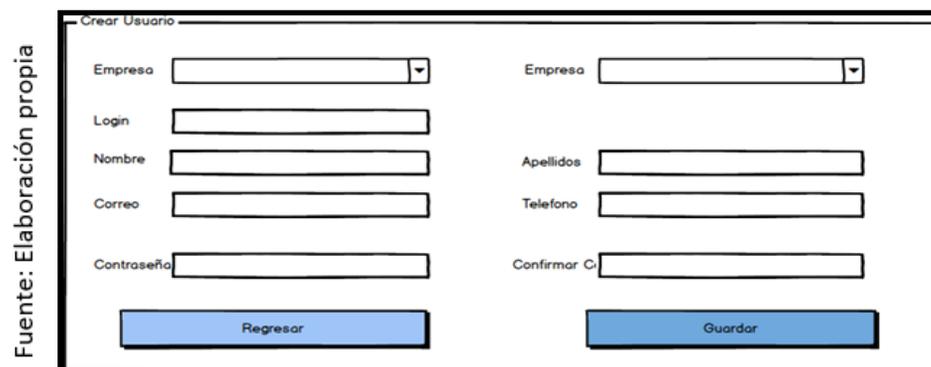


Figura 12: Vista Editar Usuario

Fuente: Elaboración propia

Edit Usuario

Empresa

Login

Nombre Apellidos

Correo Telefono

Figura 13: Vista Cambiar Contraseña

Fuente: Elaboración propia

CAMBIO CONTRASEÑA

Contraseña Actual

Contraseña Nueva

Confirme la contraseña nueva

Figura 14: Vista Lista de Tipo de Cambio

Fuente: Elaboración propia

Tipo Cambio

inicio: // Fin: // Empresa: ComboBox

tipo cambio	fecha cambio	Monto	monedas	empresa	usuario creacion	fecha creacion	usuario modificacion	estado
-------------	--------------	-------	---------	---------	------------------	----------------	----------------------	--------

Figura 15: Vista Lista de Moneda

Fuente: Elaboración propia

Moneda

inicio: // Fin: // Empresa: ComboBox

Empresa	monbre moneda	simbolo moneda	usuario creacion	fecha creacion	usuario modificacion	fecha modificacion	estado
---------	---------------	----------------	------------------	----------------	----------------------	--------------------	--------

Figura 16: Vista Nueva Moneda

Fuente: Elaboración propia

NUEVA MONEDA

Nombre

Simbolo

Empresa J&S CONSULTORES

Estado ACTIVO

Regresar Guardar

Figura 17: Vista Lista de Producto

Producto

Fuente: Elaboración propia

⏪ ⏩ ↺ ↻ 📄

Buscar 🔍

Cod producto	nombre	unidade	costo de medida	costo Unitario	Nº Serie	facturable	Almacenada	consumible	compuesto
--------------	--------	---------	-----------------	----------------	----------	------------	------------	------------	-----------

Figura 18: Vista Nuevo Producto

Nuevo Producto

Fuente: Elaboración propia

Codigo Producto	<input type="text"/>	Nombre	<input type="text"/>
Nº Serie	<input type="text"/>	Código de barra	<input type="text"/>
Descripcion Producto	<input type="text"/>	Ubicación del Pro	<input type="text"/>
Unidades Medidas	<input type="text"/>	Costo Uni	<input type="text"/>
Meses Depreciacion	<input type="text"/>	Estado	<input type="text" value="ACTIVO"/>

Facturable Consumible Depreciable Almacenable Compuesto

Figura 19: Vista Editar Producto

Edit Producto

Codigo Producto: 17007-007B Nombre: A315-51-50DQ INTEL CORE

N° Serie: N Código de barra: MERCADERIA

Descripcion Producto: AMBOS Ubicación del Pro: ITECH-GIFTS

Unidades Medidas: Costo Uni: 2300.00

Meses Depreciacion: Estado: ACTIVO

Facturable
 Consumible
 Depreciable
 Almacenable
 Compuesto

Regresar Guardar

Fuente: Elaboración propia

Figura 20: Vista Lista de Comprobantes de Pago

Documentos

inicio: / /
 Fin: / /
 Empresa: TODOS

Cliente: TODOS
 Tipo: TODOS
 serie N°Doc
 Est.Sunat: TODOS

Tipo	serie	numero	razonsocial	mon	neto	igv	otros	Desc	Total	Estado	Sunat
------	-------	--------	-------------	-----	------	-----	-------	------	-------	--------	-------

Fuente: Elaboración propia

Figura 21: Vista Comprobantes de Pago – Opción Enviar

Fuente: Elaboración propia

Documentos

⏪ ▶ ⏩ inicio: / / Fin: / / Empresa: 🔍 📄
 Cliente: Tipo: serie: N°Doc: Est.Sunat:

Tipo	serie	numero	Fecha	razonsocial	mon	neto	igv	otros	Desc	Total	Estado Sunat	
03	B001	0000001	03/05/2018	GÓMEZ ESTRADA LISSY	PEN	183.81	33.09	0.00	0.00	216.90	ACEPTADO	✎ 📄 📤 📥 🔍 📄
04	F001	0000001	03/05/2018	RIVERA SALINAS JEAN	PEN	183.81	33.09	0.00	0.00	216.90	ACEPTADO	✎ 📄 📤 📥 🔍 📄

EL BOTON PARA ENVIAR A LA SUNAT ,DICHO PROCESO AGILIZA EL TIEMPO

Figura 22: Vista Comprobantes de Pago – Opción Ver y/o Descargar XML Enviado/Recibido

Fuente: Elaboración propia

Documentos

⏪ ▶ ⏩ inicio: / / Fin: / / Empresa: 🔍 📄
 Cliente: Tipo: serie: N°Doc: Est.Sunat:

Tipo	serie	numero	Fecha	razonsocial	mon	neto	igv	otros	Desc	Total	Estado Sunat	
03	B001	0000001	03/05/2018	GÓMEZ ESTRADA LISSY	PEN	183.81	33.09	0.00	0.00	216.90	ACEPTADO	✎ 📄 📤 📥 🔍 📄
04	F001	0000001	03/05/2018	RIVERA SALINAS JEAN	PEN	183.81	33.09	0.00	0.00	216.90	ACEPTADO	✎ 📄 📤 📥 🔍 📄

el primer boton es para ver el XML de envio (informacion que se envia)

el Segundo boton es para ver el XML de Respuesta que nos manda la sunat

Figura 23: Vista Comprobantes de Pago – Opción Enviar XML a la SUNAT

Fuente: Elaboración propia

Documentos

⏪ ▶ ⏩ inicio: Fin: Empresa:

Cliente: Tipo: serie: N°Doc: Est.Sunat:

Tipo	serie	numero	Fecha	razonsocial	mon	neto	igv	otros	Desc	Total	Estado Sunat	
03	B001	0000001	03/05/2018	GÓMEZ ESTRADA LISSY	PEN	183.81	33.09	0.00	0.00	216.90	ACEPTADO	
04	F001	0000001	03/05/2018	RIVERA SALINAS JEAN	PEN	183.81	33.09	0.00	0.00	216.90	ACEPTADO	

el boton es para enviar el documento a los anulados ,cuando un documento esta mal facturado o este mal la informacion se puede anular ,sin necesidad de llamar al a sunat

Figura 24: Vista Reporte de Comprobantes de Pago Emitidos

Fuente: Elaboración propia

Documentos Emitidos

inicio: Fin:

Figura 25: Vista Reporte de Comprobantes de Pago Anulados

Fuente: Elaboración propia

Documentos Anulados

inicio: Fin:

RF01: El sistema debe permitir la autenticación del usuario utilizando un nombre de usuario y contraseña.

Figura 26: Vista Modulo Logueo

Fuente: Elaboración propia

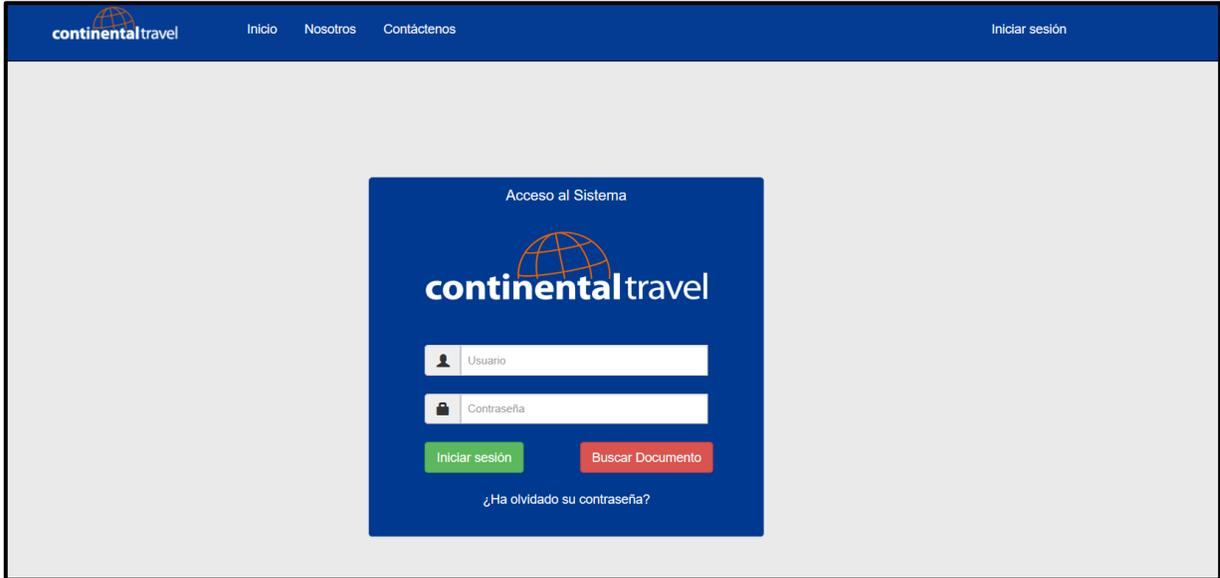


Figura 27: Estructura del Código Parte 1– Modulo Logueo

Fuente: Elaboración propia

```
@using JSUBinet.WebUblNet.Models
@model LoginViewModel
@{
    Layout = "~/Views/Shared/_LoginLayout.cshtml";
}
@{
    ViewBag.Title = "Iniciar sesión";
}
<div class="form-horizontal" id="loginForm">
    @using (Html.BeginForm("Login", "Account", new { returnUrl = ViewBag.ReturnUrl }, FormMethod.Post, new { @class = "form-horizontal", role = "form" }))
    {
        @Html.AntiForgeryToken()
        @Html.ValidationSummary(true, "", new { @class = "text-danger" })
        <div class="row">
            <div class="col-lg-6 bi">
                <div class="row">
                    <div class="col-lg-12">
                        <div class="input-group">
                            <span class="input-group-addon"><span class="glyphicon glyphicon-user"></span></span>
                            @Html.TextBoxFor(m => m.Login, new { @class = "form-control", @Style = "height:34px", @placeholder = "Usuario" })
                            @Html.ValidationMessageFor(m => m.Login, "", new { @class = "text-danger" })
                        </div>
                    </div>
                </div>
                <div class="row">
                    <div class="col-lg-12">
                        <br />
                    </div>
                </div>
                <div class="row">
                    <div class="col-lg-12">
                        <div class="input-group">
                            <span class="input-group-addon"><span class="glyphicon glyphicon-lock"></span></span>
                            @Html.PasswordFor(m => m.Password, new { @class = "form-control", @Style = "height:34px", @placeholder = "Contraseña" })
                            @Html.ValidationMessageFor(m => m.Password, "", new { @class = "text-danger" })
                        </div>
                    </div>
                </div>
                <div class="row">
                    <div class="col-lg-12 bi">
                        <div class="col-lg-12 checkbox">
                            @Html.CheckBoxFor(m => m.RememberMe)
                            @Html.LabelFor(m => m.RememberMe)
                        </div>
                    </div>
                </div>
            </div>
        </div>
    }
</div>
```

Figura 28: Estructura del Código Parte 2 – Modulo Logueo

Fuente: Elaboración propia

```

<div class="row">
  <div class="col-lg-5 bi">
    <button type="submit" class="btn btn-danger" style="width:auto">Iniciar sesión</button>
  </div>
  <div class="col-lg-5 bi">
    <div class="btn btn-primary" style="width:155px ; height : auto">
      @Html.ActionLink("Buscar Documento", "Index", "DocumentoBuscar", routeValues: null,
        htmlAttributes: new { style = "color:#fff;border:none;text-decoration: none;" })
    </div>
  </div>
</div>
<br />
<p>
  @Html.ActionLink("&#iquest;Ha olvidado su contrase&#223a;a?", "ForgotPassword")
</p>
</div>
<div class="col-lg-6 visible-lg" style="padding-left:0px">
  
</div>
</div>
</div>
<section Scripts {
  @Scripts.Render("~/bundles/jqueryval")
}

```

Figura 29: Estructura del Código Parte 3 – Modulo Logueo

Fuente: Elaboración propia

```

// POST: /Account/Login
[HttpPost]
[AllowAnonymous]
[ValidateAntiForgeryToken]
--referencias
public async Task<ActionResult> Login(LoginViewModel model, string returnUrl)
{
  if (!ModelState.IsValid)
  {
    return View(model);
  }

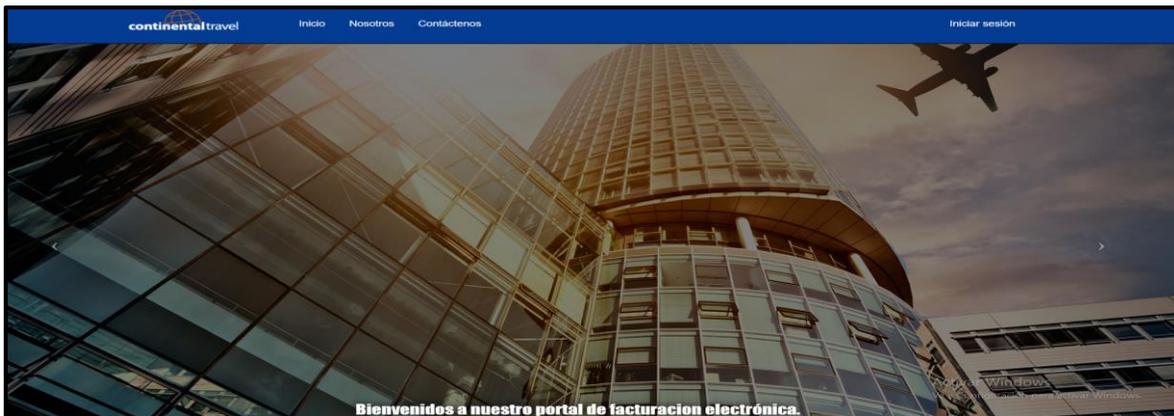
  // No cuenta los errores de inicio de sesi&#243;n para el bloqueo de la cuenta
  // Para permitir que los errores de contrase&#223a;a desencadenen el bloqueo de la cuenta, cambie a shouldLockout: true
  var result = await SignInManager.PasswordSignInAsync(model.Login, model.Password, model.RememberMe, shouldLockout: true);
  switch (result)
  {
    case SignInStatus.Success:
      return RedirectToLocal(returnUrl);
    case SignInStatus.LockedOut:
      return View("Lockout");
    case SignInStatus.RequiresVerification:
      return RedirectToAction("SendCode", new { ReturnUrl = returnUrl, RememberMe = model.RememberMe });
    case SignInStatus.Failure:
    default:
      ModelState.AddModelError("", "Intento de inicio de sesi&#243;n no v&#223;lido.");
      return View(model);
  }
}

```

RNF01: El acceso al sistema solo serßa para los usuarios registrados

Figura 30: Vista Mená Principal - Extranet –PC

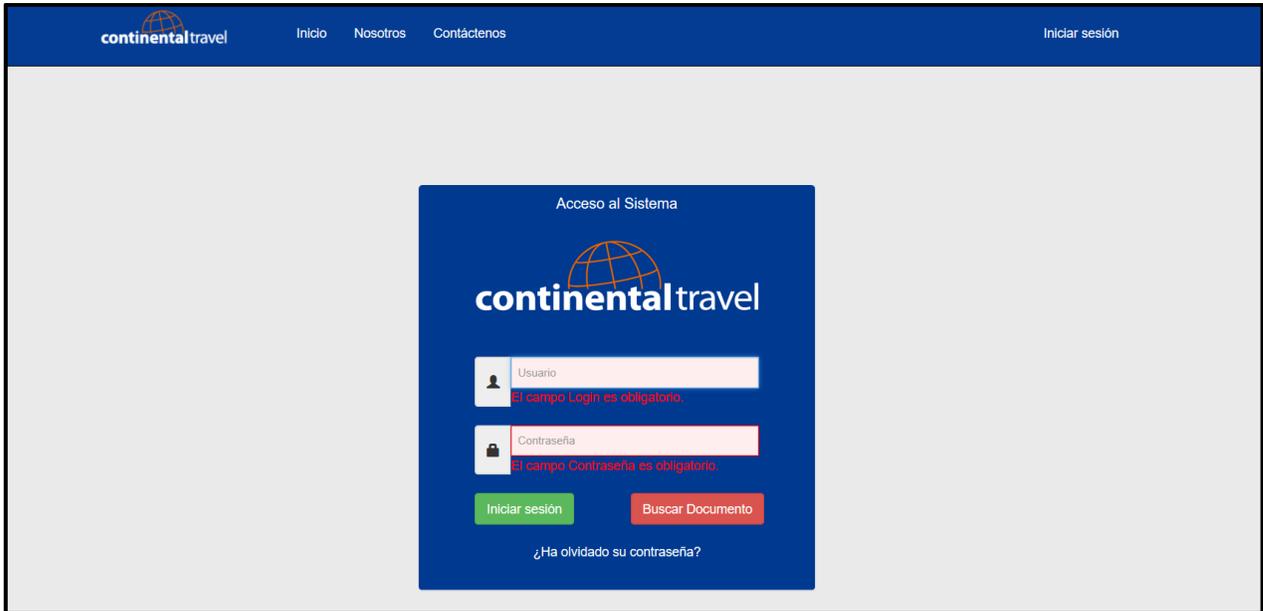
Fuente: Elaboración propia



RNF06: El sistema posee mensajes de errores informativos y orientados al usuario

Figura 31: Vista Modulo Logueo – Error Informativos

Fuente: Elaboración propia



RNF07: El sistema posee un diseño responsive con la finalidad de garantizar que la aplicación se adapte a cualquier dispositivo

Figura 32: Vista Menú Principal – Intranet – Equipo Tipo Escritorio

Fuente: Elaboración propia

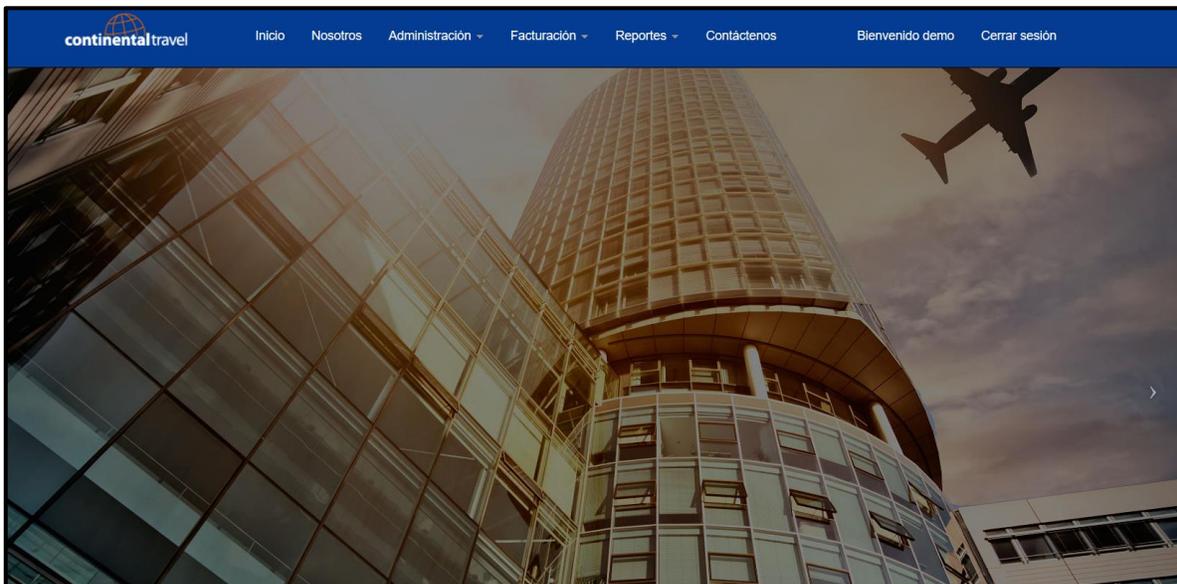


Figura 33: Vista Menú Principal – Dispositivo Móvil

Fuente: Elaboración propia



Figura 34: Estructura de Código Parte 1 – Vista Menú Principal

Fuente: Elaboración propia

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta name="description" content="">
  <meta name="author" content="">
  <link href="Content/images/ct_icon.ico" rel="icon" onerror="this.src = '../Content/images/ct_icon.ico';">
  <title>Continental Travel - Facturación Electrónica</title>
  <!-- CSS -->
  @Styles.Render("~/Content/css")
  @Scripts.Render("~/bundles/modernizr")
  @Html.C1().Styles()
  @RenderSection("styles", required: false)
  @Html.C1().Scripts().Basic()
</head>
<body style="padding-bottom:0px">
  <div class="container-fluid">
    @Html.MvcSiteMap().Menu("BootstrapMenuHelperModel")
  </div>
  <div class="container-fluid">
    @RenderBody()
  </div>
  @Scripts.Render("~/bundles/jquery")
  @Scripts.Render("~/bundles/bootstrap")
  @Html.C1().Scripts().Culture("es-PE")
  @RenderSection("scripts", required: false)
  @Html.C1().DeferredScripts()
  @Scripts.Render("~/bundles/MyApp")
  <script>
    $(function() {
      $('#.carousel').carousel({
        interval: 5000
      })
    })
  </script>
</body>
</html>
```

Figura 35: Estructura de Código Parte 2 – Vista Menú Principal

```

using Microsoft.AspNet.Identity;
<?xml version="1.0" encoding="utf-8" ?>
<!--
Layout = "~/Views/Shared/_Layout.cshtml";
ViewBag.Title = "Home Page";
-->
<div id="myCarousel" class="carousel slide visible-lg">
  <ol class="carousel-indicators">
    <li data-target="#myCarousel" data-slide-to="0" class="active"></li>
    <li data-target="#myCarousel" data-slide-to="1"></li>
    <li data-target="#myCarousel" data-slide-to="2"></li>
  </ol>
  <div class="carousel-inner">
    <div class="item active">
      <div class="fill" style="background-image:url('@Url.Content("~/Content/images/carrusel_edificio.png")');"></div>
      <div class="carousel-caption">
        <h3 class="opa">Bienvenidos a nuestro portal de facturacion electrónica.</h3>
      </div>
    </div>
    <div class="item">
      <div class="fill" style="background-image:url('@Url.Content("~/Content/images/carrusel_map-image.png")');"></div>
      <div class="carousel-caption">
        <h3 class="opa">Tu tiempo es valioso. Todos los datos de los Servicios emitidos en un click.</h3>
      </div>
    </div>
    <div class="item">
      <div class="fill" style="background-image:url('@Url.Content("~/Content/images/carrusel_invoice.png")');"></div>
      <div class="carousel-caption">
        <h3 class="opa">Revisa los Comprobantes de Pago, desde tu panel de control y en tiempo real.</h3>
      </div>
    </div>
  </div>
  <a class="left carousel-control" href="#myCarousel" data-slide="prev">
    <span class="icon-prev"></span>
  </a>
  <a class="right carousel-control" href="#myCarousel" data-slide="next">
    <span class="icon-next"></span>
  </a>
</div>
<div class="container-fluid hidden-lg">
  <div class="item">
    <div class="fill" style="background-image:url('@Url.Content("~/Content/images/carrusel_edificio.png")');"></div>
    <div class="carousel-caption">
      <h3 class="opa">Bienvenidos a nuestro portal de facturacion electrónica.</h3>
    </div>
  </div>
</div>

```

Fuente: Elaboración propia

Figura 36: Estructura de Código Parte 3 – Vista Menú Principal

```

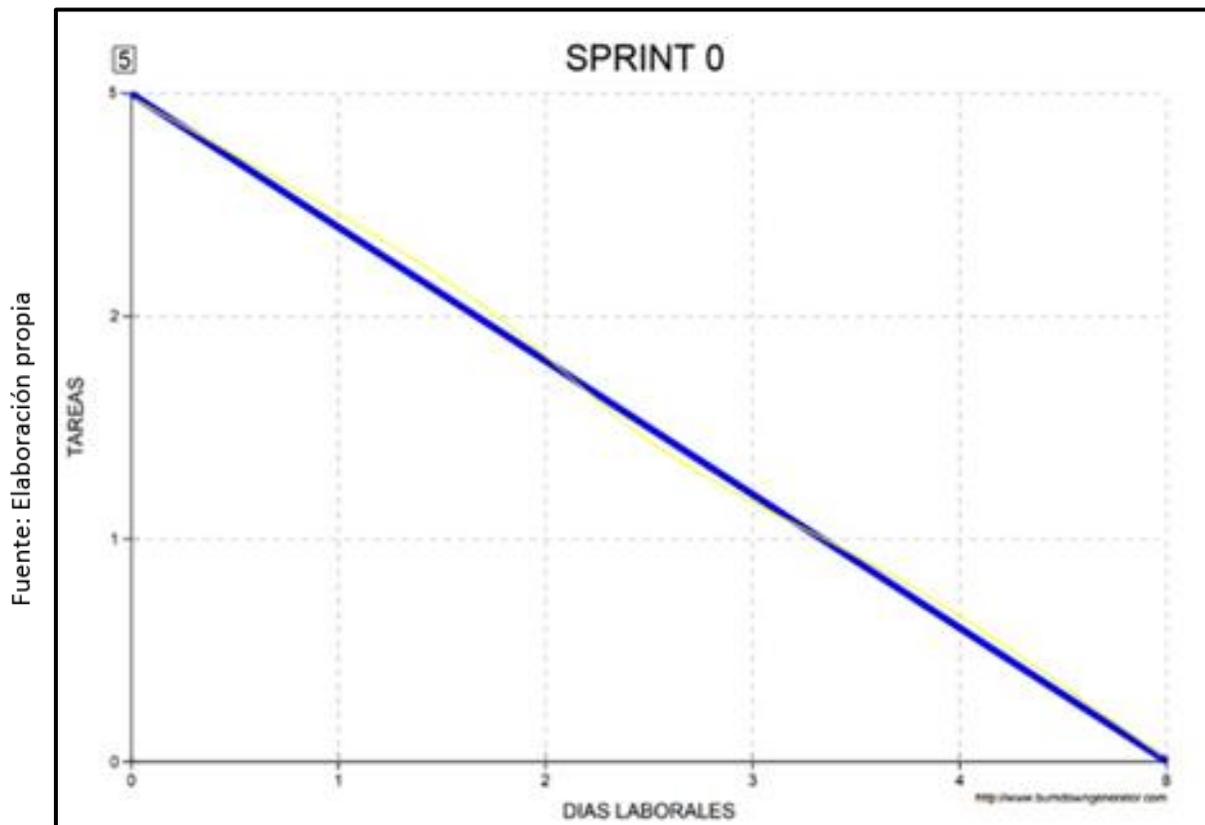
<?xml version="1.0" encoding="utf-8" ?>
<!--
Layout = "~/Views/Shared/_Layout.cshtml";
ViewBag.Title = "Home Page";
-->
<div id="myCarousel" class="carousel slide visible-lg">
  <ol class="carousel-indicators">
    <li data-target="#myCarousel" data-slide-to="0" class="active"></li>
    <li data-target="#myCarousel" data-slide-to="1"></li>
    <li data-target="#myCarousel" data-slide-to="2"></li>
  </ol>
  <div class="carousel-inner">
    <div class="item active">
      <div class="fill" style="background-image:url('@Url.Content("~/Content/images/carrusel_edificio.png")');"></div>
      <div class="carousel-caption">
        <h3 class="opa">Bienvenidos a nuestro portal de facturacion electrónica.</h3>
      </div>
    </div>
    <div class="item">
      <div class="fill" style="background-image:url('@Url.Content("~/Content/images/carrusel_map-image.png")');"></div>
      <div class="carousel-caption">
        <h3 class="opa">Tu tiempo es valioso. Todos los datos de los Servicios emitidos en un click.</h3>
      </div>
    </div>
    <div class="item">
      <div class="fill" style="background-image:url('@Url.Content("~/Content/images/carrusel_invoice.png")');"></div>
      <div class="carousel-caption">
        <h3 class="opa">Revisa los Comprobantes de Pago, desde tu panel de control y en tiempo real.</h3>
      </div>
    </div>
  </div>
  <a class="left carousel-control" href="#myCarousel" data-slide="prev">
    <span class="icon-prev"></span>
  </a>
  <a class="right carousel-control" href="#myCarousel" data-slide="next">
    <span class="icon-next"></span>
  </a>
</div>
<div class="container-fluid hidden-lg">
  <div class="item">
    <div class="fill" style="background-image:url('@Url.Content("~/Content/images/carrusel_edificio.png")');"></div>
    <div class="carousel-caption">
      <h3 class="opa">Bienvenidos a nuestro portal de facturacion electrónica.</h3>
    </div>
  </div>
</div>

```

Fuente: Elaboración propia

GRÁFICA DE BURN DOWN – SPRINT 0

Figura 37: Burn Down del Sprint N° 0



ACTA DE PLANIFICACION SPRINT N° 0

Figura 38: Acta de Planificación del Sprint N°0

ACTA DE PLANIFICACION DEL SPRINT N° 0

La reunión se lleva a cabo en presencia de los siguientes participantes:

N°	Nombres y Apellidos	Rol
1	Sergio Moran Palacios	Product Owner
2	Ronnie Alarcon Morales	Scrum Master
3	Orvic Yampool Flores Principe	Team Member

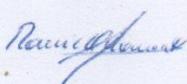
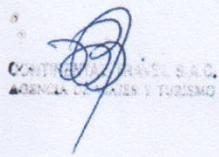
El gerente de la empresa Continental Travel S.A.C. realizó la exposición de los requerimientos e indica los requerimientos con mayor prioridad.

CODIGO	TAREA
SP0.1	Diseño de Casos de Uso del Sistema.
SP0.2	Diseño de la Base de Datos.
SP0.3	Desarrollo del Diccionario de Datos.
SP0.4	Diseño de Prototipos del sistema.
RF01	El sistema debe permitir la autenticación de usuarios, mediante un nombre de usuario y una contraseña.
RNF01	El acceso al sistema solo será para los usuarios registrados.
RNF05	El sistema cuenta con interfaces amigables para la interacción con el usuario.
RNF06	El sistema posee mensajes de errores informativos y orientados al usuario.
RNF07	El sistema posee un diseño responsive con la finalidad de garantizar que la aplicación se adapte a cualquier dispositivo

Mediante la presente acta se válida y se da conformidad los objetivos a realizarse en el Sprint 0 por parte del equipo de desarrollo, como también, las tareas propuestas del producto (Historias de Usuarios) que contiene el sprint mencionado.

Los asistentes impartirán su aprobación de acuerdo a lo presentado en la planificación del Sprint 0, indicando que la fecha de entrega de este Sprint sería el día 5 de Agosto del 2019.

Lima, 22 de julio del 2019.

Fuente: Elaboración propia

ACTA DE REUNION SPRINT N° 0

Figura 39: Acta de Reunión del Sprint N° 0

ACTA DE REUNION DEL SPRINT N° 0

La reunión se lleva a cabo en presencia de los siguientes participantes:

Presentes:

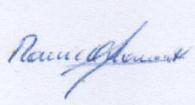
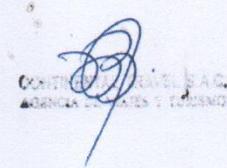
N°	Nombres y Apellidos	Rol
1	Sergio Moran Palacios	Product Owner
2	Ronnie Alarcon Morales	Scrum Master
3	Orvic Yampool Flores Principe	Team Member
4	Wilmer Alama	Team Member
5	Joseph Palomino	Team Member
6	Jimmy Pazos	Team Member

En la presente reunión, se obtuvo los requerimientos necesarios para el desarrollo del software, dichos requerimientos se encuentran detallados en las Historias de Usuario.

Asimismo, se identificó la necesidad de acceder al sistema a través de una página web, considerando los requerimientos no funcionales como seguridad, portabilidad y movilidad.

Por consiguiente, se brinda aprobación del Sprint N° 0 ya concluido, para el desarrollo de la Aplicación Web Progresiva para el proceso de facturación de la empresa Continental Travel S.A.C.

Lima, 5 de agosto del 2019.

Fuente: Elaboración propia

3.2. Desarrollo del Sprint 1

Tabla 44: Sprint 1 – Módulo de Administración

SPRINT	CODIGO	TAREA	ESTIMACIÓN (DIAS)	PRIORIDAD
SPRINT 1	RF02	El sistema debe permitir listar, registrar, modificar y eliminar usuarios para gestionar el proceso de facturación electrónica.	2	ALTA
	RF03	El sistema debe permitir listar y modificar los clientes relacionados al comprobante de pago electrónico emitido.	2	ALTA
	RF04	El sistema debe permitir listar, modificar los datos de la empresa que emitirá la facturación electrónica.	2	ALTA
	RF05	El sistema debe permitir listar y modificar los productos relacionados al comprobante de pago electrónico emitido.	2	ALTA
	RF06	El sistema debe permitir listar, registrar y modificar tipo de cambio a la fecha a emitir.	2	ALTA
	RF07	El sistema debe permitir listar, registrar y modificar moneda a emitir según el comprobante de pago electrónico generado.	2	ALTA

Fuente: Elaboración propia

RF02. El sistema debe permitir enumerar, registrar, modificar y eliminar usuarios para gestionar el proceso de facturación electrónica.

- Modulo Usuario – Operación Listar

Figura 40: Vista Modulo Usuario – Operación Listar

Fuente: Elaboración propia

Empresa	Login	Nombres Completos	Email	Telefono	Estado
CONTINENTAL TRAVEL S.A.C.	admin	Admin CTravel	admin@ctravel.com.pe		ACTIVO
CONTINENTAL TRAVEL S.A.C.	20100812542	ROSTER S.A	svaldivieso@roster.com.pe		ACTIVO
CONTINENTAL TRAVEL S.A.C.	17907416	CLARA LOPEZ	20100128056@demois.com.pe		ACTIVO
CONTINENTAL TRAVEL S.A.C.	70984164	MORIYA YOHEI	70984164@realtoours.com.pe		ACTIVO
CONTINENTAL TRAVEL S.A.C.	20601272611	CENTRO DE AGENCIAS CORPORATIVAS S.A.C. CENTRO D...	20601272611@realtoours.com.pe		ACTIVO
CONTINENTAL TRAVEL S.A.C.	20307246741	AEROVIAS DE MEXICO S.A. DE C.V. SUCURSAL PERU	20307246741@realtoours.com.pe		ACTIVO
CONTINENTAL TRAVEL S.A.C.	16696040	MIYAKAWA RANDY	16696040@realtoours.com.pe		ACTIVO
CONTINENTAL TRAVEL S.A.C.	0123456789	REAL TOURS TOKYO	0123456789@realtoours.com.pe		ACTIVO
CONTINENTAL TRAVEL S.A.C.	40599598	SURITA MANCHA WILSON FRANK	40599598@realtoours.com.pe		ACTIVO
CONTINENTAL TRAVEL S.A.C.	116085106	TSUCHIKAME AKEMI	116085106@realtoours.com.pe		ACTIVO
CONTINENTAL TRAVEL S.A.C.	TS1094197	UMEZAKI YORIKO	TS1094197@realtoours.com.pe		ACTIVO
CONTINENTAL TRAVEL S.A.C.	TZ1277271	TANAKA TAKAHIRO	TZ1277271@realtoours.com.pe		ACTIVO
CONTINENTAL TRAVEL S.A.C.	117150507	CAMONES ELSA	117150507@realtoours.com.pe		ACTIVO
CONTINENTAL TRAVEL S.A.C.	116163107	SANCHEZ AURELIO	116163107@realtoours.com.pe		ACTIVO
CONTINENTAL TRAVEL S.A.C.	08236924	LUIS PUERTAS HURTADO	08236924@realtoours.com.pe		ACTIVO
CONTINENTAL TRAVEL S.A.C.	43930436	BRYAN EYZAGUIRRE LINDO	43930436@realtoours.com.pe		ACTIVO

Figura 41: Estructura de Código – Vista Modulo Usuario – Listar Parte 1

Fuente: Elaboración propia

```

using System.Net.WebUtility;
@model UsuarioListaViewModel
@{
    ViewBag.Title = "Usuarios";
}
@{Html.CI().CollectionViewService<UsuarioListaViewModel>()
    .Id("collectionViewService")
    .PageSize(Model.PageSize)
}
<div class="well grid-sort-group">
    @{Html.CI().Pager().Owner("collectionViewService")}
    <button id="NewBtn"
        class="btn btn-default" title="Nuevo" @(User.IsInRole("Usuario Crear") ? "" : "disabled")
        onclick="location.href=@Url.Action("create")">
        <span class="glyphicon glyphicon-edit"></span>
    </button>
    <label> Empresa </label>
    @{Html.CI().AutoComplete()
        .Width(210)
        .Height(34)
        .Bind(Model.empresas)
        .Id("cboEmpresa")
        .MaxItems(10)
        .CssMatch("highlight")
        .DisplayMemberPath("name")
        .SelectedValuePath("value")
        .Disabled(!User.IsInRole("ADMINISTRADOR"))
    }
    <label>Buscar :</label>
    <input type="text" class="form-inline" id="Nombre" placeholder="Buscar nombre" style="height:34px;font-size:14px;border-radius:4px;border: 1px solid #cccccc">
    <button id="btnFilter"
        class="btn btn-default" title="Filtro">
        <span class="glyphicon wj-glyph-filter"></span>
    </button>
    <button id="btnReset" onclick="resetRow()"
        class="btn btn-default" title="Quitar Filtro">
        <span class="glyphicon glyphicon-refresh"></span>
    </button>
    <button id="btnReport" onclick="reportRow()"
        class="btn btn-default" title="Reporte">
        <span class="glyphicon wj-glyph-file"></span>
    </button>
</div>
<script id="EditTemplate" type="text/html">
<div>
    <div class="boton-new" @(User.IsInRole("Usuario Editar") ? "" : "disabled")>
        <span class="glyphicon glyphicon-pencil ft" onclick="editRow()" title="Editar"></span>
    </div>
</div>
</script>
<script id="DeleteTemplate" type="text/html">
<div>
    <div class="boton-new" @(User.IsInRole("Usuario Eliminar") ? "" : "disabled")>
        <span class="glyphicon glyphicon-trash ft" onclick="deleteRow()" title="Eliminar"></span>
    </div>
</div>
</script>

```

Figura 42: Estructura de Código – Vista Modulo Usuario – Listar Parte 2

Fuente: Elaboración propia

```

@{Html.CI().FlexGrid<UsuarioListaViewModel>()
    .Id("pagingGrid")
    .Height(440)
    .ItemsSourceId("collectionViewService")
    .IsReadOnly(true)
    .AllowDelete(true)
    .AutoGenerateColumns(false)
    .Columns(csb =>
        {
            csb.Add(cb => cb.Binding("NombreEmpresa").Header("Empresa").Width("200"));
            csb.Add(cb => cb.Binding("Login").Header("Login").Width("100"));
            csb.Add(cb => cb.Binding("NombreCompleto").Header("Nombres Completos").Width("350"));
            csb.Add(cb => cb.Binding("Email").Header("Email").Format("0").Width("220"));
            csb.Add(cb => cb.Binding("Telefono").Header("Telefono").Width("100"));
            csb.Add(cb => cb.Binding("DescripcionEstado").Header("Estado").Width("80"));
            csb.Add().Name("EditButton").CellTemplate(b => b.TemplateId("EditTemplate")).Width("25");
            csb.Add().Name("DeleteButton").CellTemplate(b => b.TemplateId("DeleteTemplate")).Width("25");
        }
    )
}
<div class="well grid-sort-group">
    @{Html.CI().Pager().Owner("pagingGrid")}
</div>
<section Scripts {
    @Scripts.Render("~/bundles/jqueryval")
    <script type="text/javascript">
        var pagingGrid, txtNombre,txtUsuario, oCollection, cboEmpresa, cboCliente;
        ci.mvc.Utils.documentReady(function () {
            pagingGrid = wjmo.Control.getControl("#pagingGrid");
            cboEmpresa = wjmo.Control.getControl("#cboEmpresa");
            txtNombre = $("#Nombre");
            txtUsuario = @Model.UserId;
            if (pagingGrid) {
                oCollection = pagingGrid.collectionView;
                pagingGrid.rows.defaultSize = 25;
                updateUrl(@Model.PageInitial);
                oCollection.refresh();
            }

            var dvFilter = document.getElementById("btnFilter");
            dvFilter.addEventListener("click", function () {
                updateUrl(0);
                oCollection.refresh();
            });
            oCollection.currentChanging.addHandler(function () {
                updateUrl(0);
            });
        });
        function editRow() {
            if (!pagingGrid || !oCollection) {
                return;
            }
            pagingGrid.invalidate();
            window.location.href = '@Url.Action("Edit", "Usuarios")?Id=' + oCollection.currentItem.IdUsuario + '&PageInitial=' + oCollection.pageIndex;
        }
    </script>
}

```

Figura 43: Estructura de Código – Vista Modulo Usuario – Listar Parte 3

Fuente: Elaboración propia

```
function reportRow() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    var url = '@Url.Action("Report", "Usuarios")?Nombre='
        + encodeURIComponent(txtNombre.val()) + "&IdEmpresa="
        + encodeURIComponent(cboEmpresa.selectedValue) + "&PageInitial="
        + oCollection.pageIndex;
    window.open(url, 'Reporte Usuario');
};
function resetRow() {
    $('#Nombre').val('');
    updateUrl(0);
    oCollection.refresh();
};
function deleteRow() {
    if (!confirm("Esta seguro que desea borrar este registro.") || !pagingGrid || !oCollection) {
        return;
    }
    oCollection.moveCurrentToPosition(-1);
    pagingGrid.invalidate();
    var rowIndex = pagingGrid.selection.row;
    oCollection.removeAt(rowIndex);
    oCollection.moveCurrentToPosition(-1);
};
function updateUrl(PageInitial) {
    oCollection._readActionUrl = '@Url.Action("RemoteBind_Read", "Usuarios")?Nombre='
    + encodeURIComponent(txtNombre.val()) + "&IdEmpresa="
    + encodeURIComponent(cboEmpresa.selectedValue) + "&UserId="
    + encodeURIComponent(txtUsuario) + "&PageInitial="
    + encodeURIComponent(PageInitial);

    oCollection._deleteActionUrl = '@Url.Action("Delete", "Usuarios")?Nombre='
    + encodeURIComponent(txtNombre.val()) + "&IdEmpresa="
    + encodeURIComponent(cboEmpresa.selectedValue);
};
</script>
}
```

Figura 44: Estructura de Código – Controlador Modulo – Usuario Parte 1

Fuente: Elaboración propia

```
#region Listar
[Authorize(Roles = "Usuario Listar")]
public ActionResult Index()
{
    UsuarioListViewModel oUsuarioDetalleViewModel = new UsuarioListViewModel() { IdEmpresa = oUsuario.IdEmpresa, empresas = new List<AutoCompleteElemento<long>>() };
    DataRepository.UsuarioProvider.DeepLoad(oUsuario, true, DeepLoadType.IncludeChildren, new Type[] { typeof(TList<ClienteLogin>), typeof(Cliente), typeof(Empresa), typeof(Estado) });
    oUsuarioDetalleViewModel.empresas.Add(new AutoCompleteElemento<long>() { name = oUsuario.IdEmpresaSource.RazonSocial, value = oUsuario.IdEmpresa });
    oUsuarioDetalleViewModel.PageSize = 16;
    oUsuarioDetalleViewModel.UserId = oUsuario.IdUsuario;
    return View(oUsuarioDetalleViewModel);
}
#endregion
```

Figura 45: Estructura de Código – Controlador Modulo – Usuario Parte 2

Fuente: Elaboración propia

```
public ActionResult RemoteBind_Read([JsonRequest] CollectionViewRequest<UsuarioListViewModel> requestData, string Nombre, long IdEmpresa, int? oUserId, int PageInitial = 0)
{
    int nCount = 0;
    string oValores = string.Empty; string cWhere = string.Empty;
    requestData.PageIndex = (PageInitial == 0 ? requestData.PageIndex : PageInitial);
    ClienteLogin oClienteLoginBus = DataRepository.ClienteLoginProvider.GetByIdUsuario(oUserId.Value).FirstOrDefault();
    Usuario oUsuario = DataRepository.UsuarioProvider.GetByIdUsuario(oUserId.Value);

    if (oUsuario.EsAdmin == false)
    {
        TList<ClienteLogin> oClienteLogin = DataRepository.ClienteLoginProvider.GetByIdCliente(oClienteLoginBus.IdCliente);
        foreach (ClienteLogin oVal in oClienteLogin)
        {
            oValores = oValores + oVal.IdUsuario.ToString() + ",";
        }
        oValores = oValores.Substring(0, oValores.Length - 1);
        cWhere = string.Format("[NombreCompleto] LIKE '{0}%' AND ((1 = -1 OR IdEmpresa = {1}) AND IdUsuario IN ({2})", Nombre.Trim(), IdEmpresa, oValores);
    }
    else
    {
        cWhere = string.Format("[NombreCompleto] LIKE '{0}%' AND ((1 = -1 OR IdEmpresa = {1})", Nombre.Trim(), IdEmpresa, oValores);
    }

    TList<Usuario> oUsuarios = DataRepository.UsuarioProvider.GetPaged(cWhere, "", requestData.PageIndex, requestData.PageSize, out nCount);
    if (oUsuarios.Count > 0)
        DataRepository.UsuarioProvider.DeepLoad(oUsuarios, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Estado), typeof(Empresa), typeof(Cliente), typeof(Estado) });

    List<UsuarioListViewModel> oUsuarioModelViews = Mapper.MapList<Usuario, List<UsuarioListViewModel>>(oUsuarios.ToList());
    CollectionViewResponse<UsuarioListViewModel> responseData = CollectionViewHelper.Read(requestData, oUsuarioModelViews);
    responseData.PageIndex = requestData.PageIndex;
    responseData.TotalItemCount = nCount;
    return this.Json(responseData);
}
```

- **Modulo Usuario – Operación Registrar**

Figura 46: Vista Modulo Usuario – Operación Registrar

Fuente: Elaboración propia

Figura 47: Estructura de Código – Vista Modulo Usuario – Registrar

Fuente: Elaboración propia

```

@model TSubInet.WebUI.Net.Models.UsuarioDetalleViewModel
@{
    ViewBag.Title = "Create";
}
using (Html.BeginForm())
{
    @Html.AntiForgeryToken()
    @Html.HiddenFor(model => model.IdCliente)
    <div class="container a1">
        <div class="panel panel-default">
            <div class="panel-heading">
                <h3 class="panel-title">Crear Usuario</h3>
                @Html.ValidationSummary(false, "", new { @class = "text-danger" })
            </div>
            <div class="panel-body">
                <div class="form-horizontal">
                    <div class="row">
                        <div class="form-group">
                            <div class="col-lg-6">
                                @Html.LabelFor(model => model.IdEmpresa, htmlAttributes: new { @class = "control-label col-md-2" })
                                @Html.DropDownListFor(m => m.IdEmpresa, new SelectList(Model.empresas, "value", "name", Model.IdEmpresa), htmlAttributes: new { @class = "form-control text-box single-line" })
                                @Html.ValidationMessageFor(m => m.IdEstado, "", new { @class = "text-danger" })
                            </div>
                            <div class="col-lg-6">
                                @Html.LabelFor(model => model.IdCliente, htmlAttributes: new { @class = "control-label col-md-2" })
                                @Html.CI().AutoComplete().Width(320).Height(25).Bind(Model.clientes).ShowDropDownButton(true).CssStyle("font-size", "12px").Id("cboCliente").IsRequired(Model.EsAdmin).MaxItems(10).CssMatch("highlight").ItemsSourceAction(Url.Action("GetCliente")).DisplayMemberPath("name").SelectedValuePath("value")
                            </div>
                        </div>
                    </div>
                    <div class="row">
                        <div class="row">
                            <div class="row">
                                <div class="row">
                                    <p class="text-danger">...</p>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
}
@section Scripts {
    @Scripts.Render("~/bundles/jqueryval")
    <script type="text/javascript">
        var cboCliente
        $.mvcUtils.documentReady(function () {
            cboCliente = $.mvc.Control.getControl("#cboCliente");
            cboCliente.selectedIndexChanged.addHandler(function (e, data) {
                var hidIdCliente = $("#idCliente");
                hidIdCliente.val(0);
                if (e.selectedValue) {
                    hidIdCliente.val(e.selectedValue);
                }
            });
        });
    </script>
}

```

Figura 48: Estructura de Código – Controlador Modulo Usuario – Registrar Parte 1

Fuente:
Elaboración propia

```
#region Crear
[Authorize(Roles = "Usuario Crear")]
--referencias
public ActionResult Create()
{
    UsuarioDetalleViewModel oUsuarioDetalleViewModel = new UsuarioDetalleViewModel()
    {
        IdEmpresa = oUsuario.IdEmpresa,
        empresas = new List<AutoCompleteElemento<long>>(),
        clientes = new List<AutoCompleteElemento<long>>()
    };
    DataRepository.UsuarioProvider.DeepLoad(oUsuario, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Empresa), typeof(Cliente) });
    oUsuarioDetalleViewModel.empresas.Add(new AutoCompleteElemento<long>() { name = oUsuario.IdEmpresaSource.RazonSocial, value = oUsuario.IdEmpresa });

    oUsuarioDetalleViewModel.EsAdmin = oUsuario.EsAdmin;
    oUsuarioDetalleViewModel.PageInitial = 0;
    return View(oUsuarioDetalleViewModel);
}
}
```

Figura 49: Estructura de Código – Controlador Modulo Usuario – Registrar Parte 2

Fuente: Elaboración propia

```
[Authorize(Roles = "Usuario Crear")]
[HttpPost]
[ValidateAntiForgeryToken]
--referencias
public async Task<ActionResult> Create(UsuarioDetalleViewModel oUsuarioDetalleViewModel)
{
    try
    {
        if (oUsuario.EsAdmin == false)
        {
            if (oUsuarioDetalleViewModel.IdCliente == 0)
                throw new Exception("Debe seleccionar un cliente.");
        }

        if (!ModelState.IsValid)
            throw new Exception("Formulario Invalido.");

        oUsuarioDetalleViewModel.Password = (string.IsNullOrEmpty(oUsuarioDetalleViewModel.Password) ? oUsuarioDetalleViewModel.Login : oUsuarioDetalleViewModel.Password);
        var user = new IdentityUser
        {
            IdEmpresa = oUsuarioDetalleViewModel.IdEmpresa,
            UserName = oUsuarioDetalleViewModel.Login,
            Email = oUsuarioDetalleViewModel.Email,
            Nombre = oUsuarioDetalleViewModel.Nombre,
            Apellidos = oUsuarioDetalleViewModel.Apellidos,
            IdEstado = 1,
            IdUsuarioCreacion = oUsuario.IdUsuario,
            FechaCreacion = DateTime.Now
        };
        var result = await UserManager.CreateAsync(user, oUsuarioDetalleViewModel.Password);
        if (!result.Succeeded)
        {
            AddErrors(result);
            throw new Exception("Formulario Invalido.");
        }
        Usuario oNewUsuario = new Usuario();
        if (oUsuarioDetalleViewModel.IdCliente == 0){
            oUsuarioDetalleViewModel.Bloqueado = true;
            var roles = new short[] { 28, 29, 32, 33
        };
        oNewUsuario = DataRepository.UsuarioProvider.GetByIdEmpresaLogin(oUsuarioDetalleViewModel.IdEmpresa, oUsuarioDetalleViewModel.Login);
        oNewUsuario.UsuarioRolCollectionGetByIdUsuario.AddRange((from Rol oRol in DataRepository.RolProvider.GetByIdEstado(1)
                                                                    where roles.Contains(oRol.IdRol)
                                                                    select new UsuarioRol()
                                                                    {
                                                                        IdRol = oRol.IdRol,
                                                                        IdEstado = 1,
                                                                        IdUsuarioCreacion = oUsuario.IdUsuario,
                                                                        FechaCreacion = DateTime.Now,
                                                                        EntityState = EntityState.Added
                                                                    }).ToList());
    }
}
```

Figura 50: Estructura de Código – Controlador Modulo Usuario – Registrar Parte 3

Fuente: Elaboración propia

```

else
{
    var roles = new short[] { 28, 29 };
    oUsuario = DataRepository.UsuarioProvider.GetByIdEmpresaLogin(oUsuarioDetalleViewModel.IdEmpresa, oUsuarioDetalleViewModel.Login);
    oNewUsuario.UsuarioRolCollectionGetByIdUsuario.AddRange((from Rol oRol in DataRepository.RolProvider.GetByIdEstado(1)
                                                            where roles.Contains(oRol.IdRol)
                                                            select new UsuarioRol()
                                                            {
                                                                IdRol = oRol.IdRol,
                                                                IdEstado = 1,
                                                                IdUsuarioCreacion = oUsuario.IdUsuario,
                                                                FechaCreacion = DateTime.Now,
                                                                EntityState = EntityState.Added
                                                            }).ToList());
}

if (oUsuarioDetalleViewModel.IdCliente != 0)
{
    oNewUsuario.ClienteLoginCollectionGetByIdUsuario.Add(new ClienteLogin()
    {
        IdCliente = oUsuarioDetalleViewModel.IdCliente,
        IdEstado = 1,
        IdUsuarioCreacion = oUsuario.IdUsuario,
        FechaCreacion = DateTime.Now,
        EntityState = EntityState.Added
    });
}

if (oUsuario.EsAdmin == true)
{
    oNewUsuario.EsAdmin = oUsuario.EsAdmin;
}

DataRepository.UsuarioProvider.DeepSave(oNewUsuario);
this.AddNotification("Registro grabado.", NotificationType.SUCCESS);
return RedirectToAction("Index");
}
catch (Exception ex)
{
    this.AddNotification(ex.Message, NotificationType.ERROR);
    Empresa oEmpresa = DataRepository.EmpresaProvider.GetByIdEmpresa(oUsuarioDetalleViewModel.IdEmpresa);
    oUsuarioDetalleViewModel.empresas = new List<AutoCompleteElemento<long>>() { new AutoCompleteElemento<long>() { name = oEmpresa.RazonSocial, value = oEmpresa.IdEmpresa } };
    oUsuarioDetalleViewModel.clientes = new List<AutoCompleteElemento<long>>();
    Cliente oCliente = DataRepository.ClienteProvider.GetByIdCliente(oUsuarioDetalleViewModel.IdCliente);
    if (oCliente != null)
    {
        oUsuarioDetalleViewModel.clientes = new List<AutoCompleteElemento<long>>() { new AutoCompleteElemento<long>() { name = oCliente.RazonSocial, value = oCliente.IdCliente } };
    }
    return View(oUsuarioDetalleViewModel);
}
}
#endregion
    
```

- Modulo Usuario – Operación Modificar

Figura 51: Vista Modulo Usuario – Operación Modificar

Fuente: Elaboración propia

Figura 52: Estructura de Código – Vista Modulo Usuario – Modificar

Fuente: Elaboración propia

```

@model JSUBInet.usbUBInet.Models.UsuarioDetalleViewModel
@{
    ViewBag.Title = "Edit";
}
using (Html.BeginForm())
{
    @Html.AntiForgeryToken()
    @Html.HiddenFor(m => m.IdUsuario)
    @Html.HiddenFor(m => m.EmailConfirmado)
    @Html.HiddenFor(m => m.FechaBloqueo)
    @Html.HiddenFor(m => m.Bloqueado)
    @Html.HiddenFor(m => m.IntentosFallidos)
    @Html.HiddenFor(m => m.IdEstado)
    @Html.HiddenFor(m => m.IdUsuarioCreacion)
    @Html.HiddenFor(m => m.FechaCreacion)
    <div class="container al">
        <div class="panel panel-default">
            <div class="panel-heading">
                <h3 class="panel-title">Editar Usuario</h3>
                @Html.ValidationSummary(false, "", new { @class = "text-danger" })
            </div>
            <div class="panel-body">
                <div class="form-horizontal">
                    <div class="row">
                        <div class="form-group">
                            <div class="col-lg-6">
                                @Html.LabelFor(model => model.IdEmpresa, htmlAttributes: new { @class = "control-label col-md-2" })
                                @Html.DropDownListFor(m => m.IdEmpresa, new SelectList(Model.empresas, "value", "name", Model.IdEmpresa)
                                , htmlAttributes: new { @class = " form-control text-box single-line" })
                                @Html.ValidationMessageFor(m => m.IdEmpresa, "", new { @class = "text-danger" })
                            </div>
                            <div class="col-lg-6">
                                &nbsp;
                            </div>
                        </div>
                    </div>
                    <div class="row">
                        <div class="form-group">
                            <div class="col-lg-6">
                                @Html.LabelFor(model => model.Login, htmlAttributes: new { @class = "control-label col-md-2" })
                                @Html.EditorFor(m => m.Login, new { htmlAttributes = new { @class = " form-control text-box single-line" } })
                                @Html.ValidationMessageFor(m => m.Login, "", new { @class = "text-danger" })
                            </div>
                            <div class="col-lg-6">
                                &nbsp;
                            </div>
                        </div>
                    </div>
                    <div class="row">
                        <div class="row">...</div>
                        <div class="row">...</div>
                        <div class="row">...</div>
                    </div>
                </div>
            </div>
        </div>
    </div>
}
@section Scripts {
    @Scripts.Render("~/bundles/jqueryval")
}

```

Figura 53: Estructura de Código – Controlador Modulo Usuario - Modificar

Fuente: Elaboración propia

```

[Authorize(Roles = "Usuario Editar")]
public ActionResult Edit(int id, int PageInitial = 0)
{
    Usuario oUsuario = DataRepository.UsuarioProvider.GetByIdUsuario(id);
    DataRepository.UsuarioProvider.DeepLoad(oUsuario, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Estado), typeof(Empresa), typeof(Cliente) });
    UsuarioDetalleViewModel oUsuarioDetalleView = Mapper.Map<Usuario, UsuarioDetalleViewModel>(oUsuario);
    oUsuarioDetalleView.PageInitial = PageInitial;
    return View(oUsuarioDetalleView);
}

[Authorize(Roles = "Usuario Editar")]
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult Edit(int id, UsuarioDetalleViewModel oUsuarioDetalleViewModel)
{
    try
    {
        Usuario oNewUsuario = DataRepository.UsuarioProvider.GetByIdUsuario(id);
        oUsuarioDetalleViewModel.Password = oNewUsuario.Password;
        oUsuarioDetalleViewModel.ConfirmarPassword = oNewUsuario.Password;
        oUsuarioDetalleViewModel.AlgoritmoSeguro = oNewUsuario.AlgoritmoSeguro;
        if (!ModelState.IsValid)
            throw new Exception("Formulario Invalido.");
        oNewUsuario = Mapper.Map<UsuarioDetalleViewModel, Usuario>(oUsuarioDetalleViewModel);
        oNewUsuario.EntityState = EntityState.Changed;
        oNewUsuario.IdUsuarioModificacion = oUsuario.IdUsuario;
        oNewUsuario.FechaModificacion = DateTime.Now;
        DataRepository.UsuarioProvider.Save(oNewUsuario);
        this.AddNotification("Registro grabado.", NotificationType.SUCCESS);
        if (User.IsInRole("Usuario Listar"))
            return RedirectToAction("Index", "Home");
        return RedirectToAction("Index", new { PageInitial = oUsuarioDetalleViewModel.PageInitial });
    }
    catch (Exception ex)
    {
        this.AddNotification(ex.Message, NotificationType.ERROR);
        Empresa oEmpresa = DataRepository.EmpresaProvider.GetByIdEmpresa(oUsuarioDetalleViewModel.IdEmpresa);
        oUsuarioDetalleViewModel.empresas = new List<AutoCompleteElemento>() { new AutoCompleteElemento() { name = oEmpresa.RazonSocial, value = oEmpresa.IdEmpresa } };
        return View(oUsuarioDetalleViewModel);
    }
}

```

- **Modulo Usuario – Operación Cambiar Contraseña**

Figura 54: Vista Modulo Usuario – Operación Cambiar Contraseña

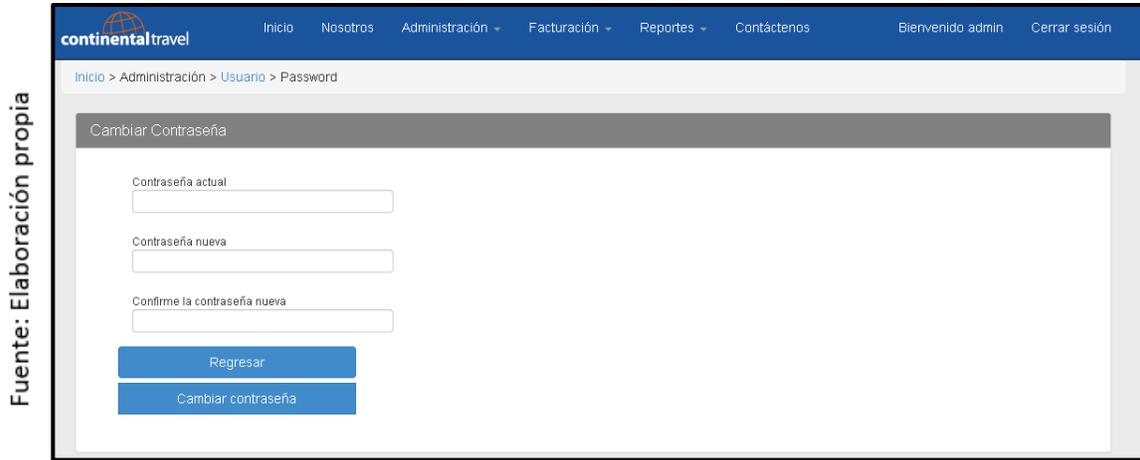


Figura 55: Estructura de Código – Vista Modulo Usuario – Cambiar Contraseña



Figura 56: Estructura de Código – Controlador Modulo Usuario – Cambiar Contraseña

Fuente: Elaboración propia

```

#region Password
--referencias
public ActionResult CambiarPassword(int id, int PageInitial)
{
    CambiarPasswordViewModel oCambiarPasswordViewModel = new CambiarPasswordViewModel() { IdUsuario = id, PageInitial = PageInitial };
    return View(oCambiarPasswordViewModel);
}
[HttpPost]
[ValidateAntiForgeryToken]
--referencias
public async Task

```

- **RF02: Modulo Usuario – Operación Gestionar Accesos**

Figura 57: Vista Modulo Usuario – Operación Gestionar Accesos

Fuente: Elaboración propia

The screenshot displays the 'Accesos Sistema' page in a web application. The navigation bar at the top includes the 'continentaltravel' logo and links for Inicio, Nosotros, Administración, Facturación, Reportes, Contáctenos, Bienvenido admin, and Cerrar sesión. The breadcrumb trail shows 'Inicio > Administración > Usuario > Accesos'. The main content area features a table titled 'Accesos Sistema' with the following data:

Id	NombreModulo	Nombre Regla	Permitir
▲ Moneda (1 ítems)			
20293	Moneda	Moneda Listar	<input checked="" type="checkbox"/>
▲ Tipo Cambio (1 ítems)			
20294	Tipo Cambio	Tipo Cambio Listar	<input checked="" type="checkbox"/>
▲ Usuario (4 ítems)			
20295	Usuario	Usuario Crear	<input checked="" type="checkbox"/>
20296	Usuario	Usuario Editar	<input checked="" type="checkbox"/>
20297	Usuario	Usuario Eliminar	<input checked="" type="checkbox"/>
20298	Usuario	Usuario Listar	<input checked="" type="checkbox"/>
▲ Documentos (1 ítems)			
20299	Documentos	Documento Listar	<input checked="" type="checkbox"/>

At the bottom of the table area, there is a blue button labeled 'Regresar'.

Figura 58: Estructura de Código – Vista Modulo Usuario – Gestionar Accesos

Fuente: Elaboración propia

```

ViewBag.Title = "Cambiar Accesos";
using (Html.BeginForm())
{
    @Html.Ci().CollectionViewService(oUsuarioRolListViewModel)()
        .Id("collectionViewService")
}
@Html.AntiForgeryToken()
<div class="container al">
    <div class="panel panel-default">
        <div class="panel-heading">...</div>
        <div class="panel-body">
            <div class="form-horizontal">
                <div class="row">
                    <div class="form-group">
                        <div class="col-lg-8">
                            @Html.Ci().FieldId(oUsuarioRolListViewModel)()
                                .Id("pagingGrid")
                                .Height(320)
                                .ItemsSourceId("collectionViewService")
                                .AutoGenerateColumns(false)
                                .Columns(csb =>
                                {
                                    csb.Add(cb => cb.Binding("IdUsuarioRol").Width("60").Align("right").Header("Id").Format("0").IsReadOnly(true));
                                    csb.Add(cb => cb.Binding("NombreModulo").Header("Nombre Modulo").Width("230").IsReadOnly(true));
                                    csb.Add(cb => cb.Binding("NombreRol").Header("Nombre Rol").Width("230").IsReadOnly(true));
                                    csb.Add(cb => cb.Binding("PermitirAcceso").Header("Permitir").Width("60"));
                                })
                        </div>
                    </div>
                    <div class="col-lg-4">...</div>
                </div>
            </div>
        </div>
    </div>
}
@section Scripts {
<script type="text/javascript">
var pagingGrid, oCollection
$.swc.util.ready(function () {
    pagingGrid = wijmo.Control.getControl("#pagingGrid");
    if (pagingGrid) {
        oCollection = pagingGrid.collectionView;
        pagingGrid.rows.defaultSize = 25;
        updateUrl();
        oCollection.refresh();
        var groupDescriptions = pagingGrid.collectionView.groupDescriptions;
        pagingGrid.beginUpdate();
        groupDescriptions.push(new wijmo.collections.PropertyGroupDescription("NombreModulo"));
        pagingGrid.endUpdate();
    }
});
function updateUrl() {
    oCollection.readActionUrl = "@Url.Action("UsuarioRol_Read", "Usuarios");IdUsuario=" + encodeURIComponent($("#IdUsuario"));
    oCollection.updateActionUrl = "@Url.Action("UsuarioRol_Update", "Usuarios");IdUsuario=" + encodeURIComponent($("#IdUsuario"));
}
</script>

```

Figura 59: Estructura de Código – Controlador Modulo Usuario – Gestionar Accesos

Fuente: Elaboración propia

```

#region Accesos
[Authorize(Roles = "Usuario Administrar")]
public ActionResult CambiarAccesos(int id, int PageInitial)
{
    Usuario oUsuario = DataRepository.UsuarioProvider.GetByIdUsuario(id);
    DataRepository.UsuarioProvider.DeepLoad(oUsuario, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Estado), typeof(Empresa), typeof(Cliente) });
    UsuarioDetalleViewModel oUsuarioDetalleView = Mapper.Map<Usuario, UsuarioDetalleViewModel>(oUsuario);
    oUsuarioDetalleView.PageInitial = PageInitial;
    return View(oUsuarioDetalleView);
}
}
[Authorize(Roles = "Usuario Administrar")]
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult CambiarAccesos(UsuarioDetalleViewModel oUsuarioDetalleViewModel)
{
    return View();
}
}
public ActionResult ControlLogin(int id, int PageInitial)
{
    Usuario oUsuario = DataRepository.UsuarioProvider.GetByIdUsuario(id);
    DataRepository.UsuarioProvider.DeepLoad(oUsuario, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Estado), typeof(Empresa), typeof(Cliente) });
    UsuarioDetalleViewModel oUsuarioDetalleView = Mapper.Map<Usuario, UsuarioDetalleViewModel>(oUsuario);
    oUsuarioDetalleView.PageInitial = PageInitial;
    return View(oUsuarioDetalleView);
}
}

```

- RF02: Modulo Usuario – Operación Eliminar

Figura 60: Vista Modulo Usuario – Operación Eliminar

Fuente: Elaboración propia

Empresa	Login	Nombres Completos	Email	Telefono	Estado		
CONTINENTAL TRAVEL S.A.C.	admin	Admin CTravel	admin@ctravel.com.pe		ACTIVO		
CONTINENTAL TRAVEL S.A.C.	20100812542	ROSTER S.A	svaldivieso@roster.com.pe		ACTIVO		
CONTINENTAL TRAVEL S.A.C.	17907416	CLARA LOPEZ	20100128056@demojs.com.pe		ACTIVO		
CONTINENTAL TRAVEL S.A.C.	70984164	MORIYA YOHEI	70984164@realtours.com.pe		ACTIVO		
CONTINENTAL TRAVEL S.A.C.	20601272611	CENTRO DE AGENCIAS CORPORATIVAS S.A.C. CENTRO D...	20601272611@realtours.com.pe		ACTIVO		
CONTINENTAL TRAVEL S.A.C.	20307246741	AEROVIAS DE MEXICO S.A. DE C.V. SUCURSAL PERU	20307246741@realtours.com.pe		ACTIVO		
CONTINENTAL TRAVEL S.A.C.	16696040	MIYAKAWA RANDY	16696040@realtours.com.pe		ACTIVO		
CONTINENTAL TRAVEL S.A.C.	0123456789	REAL TOURS TOKYO	0123456789@realtours.com.pe		ACTIVO		
CONTINENTAL TRAVEL S.A.C.	40599588	SURITA MANCHA WILSON FRANK	40599588@realtours.com.pe		ACTIVO		
CONTINENTAL TRAVEL S.A.C.	116085106	TSUCHIKAME AKEMI	116085106@realtours.com.pe		ACTIVO		
CONTINENTAL TRAVEL S.A.C.	TS1094197	UMEZAKI YORIKO	TS1094197@realtours.com.pe		ACTIVO		
CONTINENTAL TRAVEL S.A.C.	TZ1277271	TANAKA TAKAHIRO	TZ1277271@realtours.com.pe		ACTIVO		
CONTINENTAL TRAVEL S.A.C.	117150507	CAMONES ELSA	117150507@realtours.com.pe		ACTIVO		
CONTINENTAL TRAVEL S.A.C.	116163107	SANCHEZ AURELIO	116163107@realtours.com.pe		ACTIVO		
CONTINENTAL TRAVEL S.A.C.	08236824	LUIS PUERTAS HURTADO	08236824@realtours.com.pe		ACTIVO		
CONTINENTAL TRAVEL S.A.C.	43930436	BRYAN EYZAGUIRRE LINDO	43930436@realtours.com.pe		ACTIVO		

Figura 61: Estructura de Código – Vista Modulo Usuario - Eliminar

Fuente: Elaboración propia

```
function deleteRow() {
    if (!confirm("Esta seguro que desea borrar este registro.?) || !pagingGrid || !oCollection) {
        oCollection.moveCurrentToPosition(-1);
        return;
    }
    pagingGrid.invalidate();
    var rowIndex = pagingGrid.selection.row;
    oCollection.removeAt(rowIndex);
    oCollection.moveCurrentToPosition(-1);
};
```

Figura 62: Estructura de Código – Controlador Modulo Usuario - Eliminar

Fuente: Elaboración propia

```
#region Eliminar
[HttpPost]
public ActionResult Delete([FromBody]CollectionViewEditRequest<UsuarioListViewModel> requestData, long IdEmpresa)
{
    int nCount = 0;

    CollectionViewResponse<UsuarioListViewModel> responseData = CollectionViewHelper.Edit(requestData,
        item =>
        {
            string error = string.Empty;
            bool success = true;

            try
            {
                DataRepository.UsuarioProvider.Delete((int)item.IdUsuario);
            }
            catch (Exception e)
            {
                error = e.Message;
                if (error.Contains("FK"))
                {
                    error = "No se puede eliminar el usuario .";
                }
                success = false;
            }
            return new CollectionViewItemResult<UsuarioListViewModel>
            {
                Error = error,
                Success = success && ModelState.IsValid,
                Data = item
            };
        }, () =>
        {
            string cWhere = string.Format("({0} = -1 OR IdEmpresa = {0})", IdEmpresa);
            TList<Usuario> oUsuario = DataRepository.UsuarioProvider.GetPaged(cWhere, "", requestData.PageIndex, requestData.PageSize, out nCount);
            if (oUsuario.Count > 0)

                DataRepository.UsuarioProvider.DeepLoad(oUsuario, true, DeepLoadType.IncludeChildren, new Type[] { typeof(TList<ClienteLogin>), typeof(Cliente), typeof(Empresa), typeof(Estado) });
            List<UsuarioListViewModel> oUsuarioListViewModels = Mapper.Map<List<Usuario>, List<UsuarioListViewModel>>(oUsuario.ToList());
            return oUsuarioListViewModels;
        });
    responseData.PageIndex = requestData.PageIndex;
    responseData.TotalItemCount = nCount;
    return this.Json(responseData);
}
#endregion
```

RF03: El sistema debe permitir listar y modificar los clientes relacionados al comprobante de pago electrónico emitido.

- Modulo Cliente – Operación Listar

Figura 63: Vista Modulo Cliente – Operación Listar

Fuente: Elaboración propia

Tipo Do...	Ruc	Razon Social	Dirección	Estado
RUC	20506646431	OCUTECH S.A.C.	JR. MANUEL IRRIBARREN NRO. 1325 (CRUCE AV. D. ORUE CON MANUEL IR...	ACTIVO
RUC	20492189589	FULLTECH PERU S.A.C.	AV. ISAAC NEWTON 138, URB. IND. SAN FRANCISCO, ATE	ACTIVO
RUC	20100016681	IMPORTACIONES HIRACKA S.A.C.	AV. LA MARINA 2650 - SAN MIGUEL	ACTIVO
PASAP...	118065114	GUERRA IRMA	LIMA	ACTIVO
PASAP...	73689755	SIU TORRES JUAN JAVIER	AV. ARIOSTO MATELINI 473 - CHORRILLOS	ACTIVO

Figura 64: Estructura de Código – Vista Modulo Cliente – Lista Parte 1

Fuente: Elaboración propia

```

using System.Web.Mvc;
using System.Web.Mvc.Model;
using System.Web.Mvc.PagedList;
using System.Web.Mvc.PagedListExtensions;

namespace ContinentalTravel.Models
{
    public class ClienteListarViewModel
    {
        public string PageTitle { get; set; }
    }
}

@model ClienteListarViewModel
@{
    ViewBag.Title = "Clientes";
}

@Html.Ci().CollectionViewService(ClienteListarViewModel)
    .Id("collectionviewservice")
    .PageSize(Model.PageSize)
    .Render()
    .DivClass("well grid-sort-group")
    .DivId("collectionviewservice")
    .PageHeader(
        @<div class="form-inline">
            @<input type="text" class="form-control" placeholder="Buscar Razon Social" style="height:34px;font-size:14px;border-radius:4px;border: 1px solid #cccccc">
            @<input type="text" class="form-control" placeholder="Ingreso N° Ruc" style="height:34px;font-size:14px;border-radius:4px;border: 1px solid #cccccc" maxlength="11">
            @<button type="button" class="btn btn-default">
                @<span class="glyphicon glyphicon-filter">
            @</button>
            @<button id="btnReset" onclick="resetRow()" class="btn btn-default" title="Quitar filtro">
            @</button>
            @<button class="glyphicon glyphicon-refresh">
            @</button>
            @<button id="btnReport" onclick="reportRow()" class="btn btn-default" title="Reporte">
            @<span class="glyphicon glyphicon-file">
            @</button>
        @</div>
        @<div id="editTemplate" type="text/html">
            @<div class="form-inline">
                @<input type="text" class="form-control" placeholder="Buscar Razon Social" style="height:34px;font-size:14px;border-radius:4px;border: 1px solid #cccccc">
                @<input type="text" class="form-control" placeholder="Ingreso N° Ruc" style="height:34px;font-size:14px;border-radius:4px;border: 1px solid #cccccc" maxlength="11">
                @<button type="button" class="btn btn-default">
                    @<span class="glyphicon glyphicon-filter">
                @</button>
                @<button id="btnReset" onclick="resetRow()" class="btn btn-default" title="Quitar filtro">
                @</button>
                @<button class="glyphicon glyphicon-refresh">
                @</button>
                @<button id="btnReport" onclick="reportRow()" class="btn btn-default" title="Reporte">
                @<span class="glyphicon glyphicon-file">
                @</button>
            @</div>
        @</div>
        @<div id="grid">
            @<table border="1">
                @<thead>
                    @<tr>
                        @<th>Tipo Do...</th>
                        @<th>Ruc</th>
                        @<th>Razon Social</th>
                        @<th>Dirección</th>
                        @<th>Estado</th>
                    @</tr>
                @</thead>
                @<tbody>
                    @<tr>
                        @<td>RUC</td>
                        @<td>20506646431</td>
                        @<td>OCUTECH S.A.C.</td>
                        @<td>JR. MANUEL IRRIBARREN NRO. 1325 (CRUCE AV. D. ORUE CON MANUEL IR...</td>
                        @<td>ACTIVO</td>
                    @</tr>
                    @<tr>
                        @<td>RUC</td>
                        @<td>20492189589</td>
                        @<td>FULLTECH PERU S.A.C.</td>
                        @<td>AV. ISAAC NEWTON 138, URB. IND. SAN FRANCISCO, ATE</td>
                        @<td>ACTIVO</td>
                    @</tr>
                    @<tr>
                        @<td>RUC</td>
                        @<td>20100016681</td>
                        @<td>IMPORTACIONES HIRACKA S.A.C.</td>
                        @<td>AV. LA MARINA 2650 - SAN MIGUEL</td>
                        @<td>ACTIVO</td>
                    @</tr>
                    @<tr>
                        @<td>PASAP...</td>
                        @<td>118065114</td>
                        @<td>GUERRA IRMA</td>
                        @<td>LIMA</td>
                        @<td>ACTIVO</td>
                    @</tr>
                    @<tr>
                        @<td>PASAP...</td>
                        @<td>73689755</td>
                        @<td>SIU TORRES JUAN JAVIER</td>
                        @<td>AV. ARIOSTO MATELINI 473 - CHORRILLOS</td>
                        @<td>ACTIVO</td>
                    @</tr>
                @</tbody>
            @</table>
        @</div>
    @</div>

```

Figura 65: Estructura de Código – Vista Modulo Cliente – Lista Parte 2

Fuente: Elaboración propia

```

@section Scripts {
    @Scripts.Render("~/bundles/jqueryval")
    <script type="text/javascript">
        var pagingGrid, oCollection, cboEmpresa, txtRazonSocial, txtRuc;
        cl.mvc.utils.documentReady(function () {
            pagingGrid = wijmo.controls.gridcontrol("#pagingGrid");
            cboEmpresa = wijmo.controls.combobox("#cboEmpresa");
            txtRazonSocial = $("#RazonSocial");
            txtRuc = $("#NumeroDocumentoIdentidad");
            if (pagingGrid) {
                oCollection = pagingGrid.collectionView;
                pagingGrid.rows.defaults.pageSize = 25;
                updateUrl(@Url.Action("Index", "Clientes"));
                oCollection.refresh();
            }
            var dfilter = document.getElementById("dfilter");
            dfilter.addEventListener("click", function () {
                updateUrl(0);
                oCollection.refresh();
            });
            oCollection.currentChanging.addHandler(function () {
                updateUrl(0);
            });
            function editRow() {
                if (pagingGrid || oCollection) {
                    return;
                }
                pagingGrid.invalidate();
                window.location.href = "@Url.Action('Edit', 'Clientes')?id=" + oCollection.currentItem.idCliente + "&PageInitial=" + oCollection.pageIndex;
            }
            function reportRow() {
                if (pagingGrid || oCollection) {
                    return;
                }
                pagingGrid.invalidate();
                var url = "@Url.Action('Report', 'Clientes')?idEmpresa="
                    + encodeURIComponent(cboEmpresa.selectedValue) + "&RazonSocial="
                    + encodeURIComponent(txtRazonSocial.val()) + "&NumeroDocumentoIdentidad="
                    + encodeURIComponent(txtRuc.val()) + "&PageInitial="
                    + oCollection.pageIndex;
                window.open(url, "Reporte Cliente");
            }
            function resetRow() {
                $("#RazonSocial, #NumeroDocumentoIdentidad").val("");
                updateUrl(0);
                oCollection.refresh();
            }
            function updateUrl(PageInitial) {
                oCollection.readActionUrl = "@Url.Action('RemoteBind_Read', 'Clientes')?idEmpresa="
                    + encodeURIComponent(cboEmpresa.selectedValue) + "&RazonSocial="
                    + encodeURIComponent(txtRazonSocial.val()) + "&NumeroDocumentoIdentidad="
                    + encodeURIComponent(txtRuc.val()) + "&PageInitial="
                    + encodeURIComponent(PageInitial);
                oCollection._selectActionUrl = "@Url.Action('Delete', 'Clientes')?idEmpresa="
                    + encodeURIComponent(cboEmpresa.selectedValue);
            }
        });
    </script>
}
    
```

Figura 66: Estructura de Código – Controlador Modulo Cliente – Listar Parte 1

Fuente: Elaboración propia

```

#region View Principal
[Authorize(Roles = "Cliente Listar")]
--referencias
public ActionResult Index(int PageInitial = 0)
{
    ClienteListViewModel oClienteDetalleViewModel = new ClienteListViewModel()
    {
        IdEmpresa = -1,
        empresas = new List<AutoCompleteElemento<long>>(),
    };

    DataRepository.UsuarioProvider.DeepLoad(oUsuario, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Estado), typeof(Empresa) });
    oClienteDetalleViewModel.empresas.Add(new AutoCompleteElemento<long>() { name = oUsuario.IdEmpresaSource.RazonSocial, value = oUsuario.IdEmpresa });

    if (User.IsInRole("ADMINISTRADOR"))
        oClienteDetalleViewModel.empresas.Insert(0, new AutoCompleteElemento<long>() { name = "TODOS", value = -1 });

    oClienteDetalleViewModel.PageSize = 16;
    oClienteDetalleViewModel.PageInitial = PageInitial;
    return View(oClienteDetalleViewModel);
}
#endregion
    
```

Figura 67: Estructura de Código – Controlador Modulo Cliente – Lista Parte 2

Fuente: Elaboración propia

```

public ActionResult RemoteBind_Read([CllsonRequest] CollectionViewRequest<ClienteListViewModel> requestData, long IdEmpresa, string RazonSocial, string NumeroDocumentoIdentidad, int PageIni
{
    int nCount = 0;
    requestData.PageIndex = (PageInitial == 0 ? requestData.PageIndex : PageInitial);
    object[] cparams = { IdEmpresa, RazonSocial.Trim(), NumeroDocumentoIdentidad.Trim() };
    string cwhere = string.Format(" ('{0}' = -1 OR IdEmpresa = '{0}') AND RazonSocial like '{1}%' AND NumeroDocumentoIdentidad like '{2}%' ", cparams);
    IList<Cliente> oClientes = DataRepository.ClienteProvider.GetPaged(cwhere, "", requestData.PageIndex, requestData.PageSize, out nCount);
    if (oClientes.Count > 0)
        DataRepository.ClienteProvider.DeepLoad(oClientes, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Estado), typeof(Empresa), typeof(TipoDocumentoIdentidad) });

    List<ClienteListViewModel> oClienteModelViews = Mapper.Map<List<Cliente>, List<ClienteListViewModel>>(oClientes.ToList());
    CollectionViewResponse<ClienteListViewModel> responseData = CollectionViewHelper.Read(requestData, oClienteModelViews);
    responseData.PageIndex = requestData.PageIndex;
    responseData.TotalItemCount = nCount;
    return this.Cllson(responseData);
}
    
```

- **Modulo Cliente – Operación Modificar**

Figura 68: Vista Modulo Cliente – Operación Modificar

Fuente: Elaboración propia

Figura 69: Estructura de Código – Vista Modulo Cliente – Lista Parte 1

Fuente: Elaboración propia

```

@model JSUBInet.WebUIInet.Models.ClienteDetalleVistaModel
@{
    ViewBag.Title = "Edit";
}
using (Html.BeginForm())
{
    @Html.AntiForgeryToken()
    @Html.HiddenFor(model => model.IdCliente)
    @Html.HiddenFor(model => model.IdTipoDocumento)
    @Html.HiddenFor(model => model.CuentaContableSoltes)
    @Html.HiddenFor(model => model.CuentaContableDolares)
    @Html.HiddenFor(model => model.IdUsuarioCreacion)
    @Html.HiddenFor(model => model.FechaCreacion)
    @Html.HiddenFor(model => model.PageInitial)
    <div class="container al">
        <div class="panel panel-default">
            <div class="panel-heading">
                <h3 class="panel-title">Editar Cliente</h3>
                @Html.ValidationSummary(false, "", new { @class = "text-danger" })
            </div>
            <div class="panel-body">
                <div class="form-horizontal">
                    <div class="row">
                        <div class="form-group">
                            <div class="col-lg-6">
                                @Html.LabelFor(model => model.IdTipoDocumentoIdentidad, htmlAttributes: new { @class = "control-label col-md-4" })
                                @Html.DropDownListFor(m => m.IdTipoDocumentoIdentidad, new SelectList(Model.tipoDocumentoIdentidad, "IdTipoDocumentoIdentidad", "DescripcionDocumentoIdentidad", Model.IdTipoDocumentoIdentidad, "Selecciona"), htmlAttributes: new { @class = "form-control text-box single-line", @style = "width:25%" })
                                @Html.ValidationMessageFor(m => m.IdTipoDocumentoIdentidad, "", new { @class = "text-danger" })
                            </div>
                            <div class="col-lg-6">
                                @Html.LabelFor(model => model.NumeroDocumentoIdentidad, htmlAttributes: new { @class = "control-label col-md-4" })
                                @Html.EditorFor(model => model.NumeroDocumentoIdentidad, new { htmlAttributes = new { @class = "form-control", @style = "width:25%" } })
                                @Html.ValidationMessageFor(model => model.NumeroDocumentoIdentidad, "", new { @class = "text-danger" })
                            </div>
                        </div>
                    </div>
                    <div class="row"></div>
                    <div class="row"></div>
                    <div class="row"></div>
                    <div class="row"></div>
                    <div class="row">
                        <div class="form-group">
                            <div class="col-lg-12">
                                <div class="col-lg-6">
                                    @Html.ActionLink("Regresar", "Index", null, htmlAttributes: new { @class = "btn btn-primary", @style = "color: white; font-family:Arial;width:100%" })
                                </div>
                                <div class="col-lg-6">
                                    <button type="submit" class="btn btn-primary" style="width:100%">Guardar</button>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
}

```


RF04: El sistema debe permitir listar, modificar los datos de la empresa que emitirá la facturación electrónica.

- Modulo Empresa – Operación Listar

Figura 72: Vista Modulo Empresa – Operación Listar

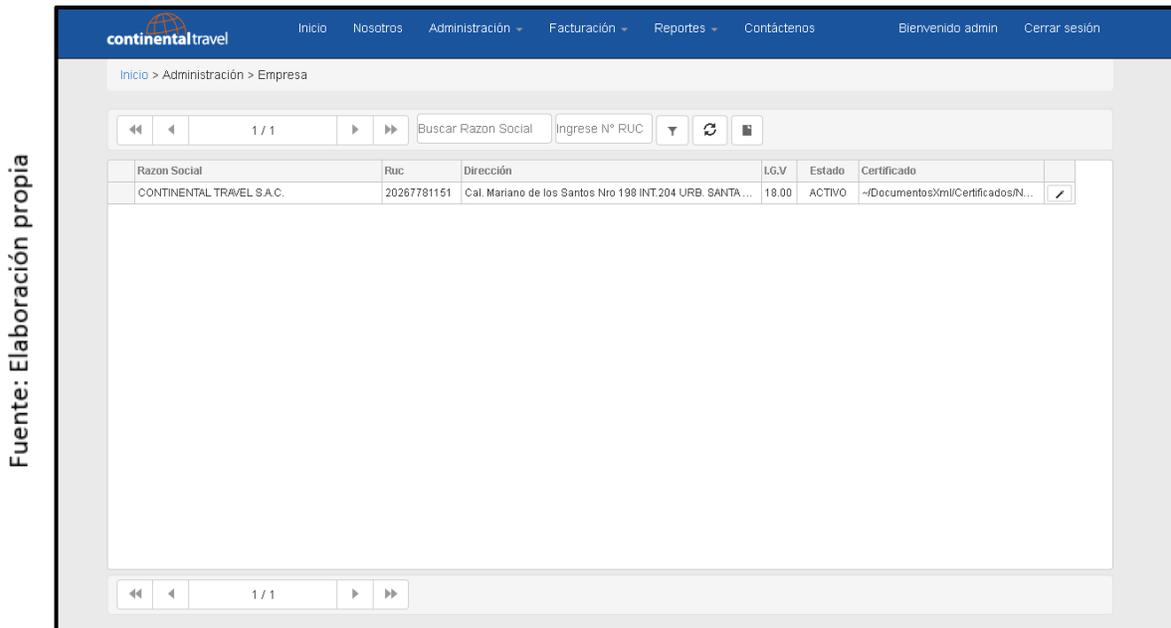


Figura 73: Estructura de Código – Vista Modulo Empresa – Listar Parte 1



Figura 74: Estructura de Código – Vista Modulo Empresa – Listar Parte 2

Fuente: Elaboración propia

```

<script type="text/javascript">
var pagingGrid, txtRazonSocial,txtRuc, oCollection;
cl.mvc.Utils.documentReady(function () {
    pagingGrid = wjmo.Control.getControl("#pagingGrid");
    txtRazonSocial = $("#RazonSocial");
    txtRuc = $("#RucEmpresa");
    if (pagingGrid) {
        oCollection = pagingGrid.collectionView;
        pagingGrid.rows.defaultSize = 25;
        updateUrl(PageInitial);
        oCollection.refresh();
    };
    var dvFilter = document.getElementById('btnFilter');
    dvFilter.addEventListener("click", function () {
        updateUrl(0);
        oCollection.refresh();
    });
    oCollection.currentChanging.addHandler(function () {
        updateUrl(0);
    });
});
function editRow() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    window.location.href = '@Url.Action("Edit", "Empresas")?id=' + oCollection.currentItem.IdEmpresa + '&PageInitial=' + oCollection.pageIndex;
};
function reportRow() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    var url = '@Url.Action("Report", "Empresas")?RazonSocial=' + encodeURIComponent(txtRazonSocial.val()) + "&PageInitial=" + oCollection.pageIndex;
    window.open(url, 'Reporte Empresa');
};
function resetRow() {
    $("#RazonSocial, #RucEmpresa").val('');
    updateUrl(0);
    oCollection.refresh();
};
function updateUrl(PageInitial) {
    oCollection._readActionUrl = '@Url.Action("RemoteBind_Read", "Empresas")?RazonSocial='
        + encodeURIComponent(txtRazonSocial.val()) + "&RucEmpresa="
        + encodeURIComponent(txtRuc.val()) + "&PageInitial="
        + encodeURIComponent(PageInitial);
    oCollection._deleteActionUrl = '@Url.Action("Delete", "Empresas")?RazonSocial='
        + encodeURIComponent(txtRazonSocial.val());
};
</script>

```

Figura 75: Estructura de Código – Controlador Modulo Empresa – Listar

Fuente: Elaboración propia

```

#region Listar
- referencias
public ActionResult Index(int PageInitial = 0)
{
    EmpresaListViewModel oEmpresaDetalleViewModel = new EmpresaListViewModel() { IdEmpresa = -1, empresas = new List<AutoCompleteElemento<long>>() };

    DataRepository.UsuarioProvider.DeepLoad(oUsuario, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Empresa) });
    if (User.IsInRole("ADMINISTRADOR"))
        oEmpresaDetalleViewModel.empresas.Insert(0, new AutoCompleteElemento<long>() { name = "TODOOS", value = -1 });

    oEmpresaDetalleViewModel.PageSize = 16;
    oEmpresaDetalleViewModel.PageInitial = PageInitial;
    return View(oEmpresaDetalleViewModel);
}
#endregion

```

- Modulo Empresa – Operación Modificar

Figura 76: Vista Modulo Empresa – Operación Modificar

Fuente: Elaboración propia

The screenshot shows a web application interface for editing a company. The header includes the 'continentaltravel' logo and navigation links like 'Inicio', 'Nosotros', 'Administración', 'Facturación', 'Reportes', and 'Contáctenos'. The breadcrumb trail is 'Inicio > Administración > Empresa > Editar'. The main content area is titled 'Editar Empresa' and contains a form with the following fields:

- Razón Social: CONTINENTAL TRAVELS A.C.
- Dirección: Cal. Mariano de los Santos Nro 198 INT.204 URE
- Usuario Sunat: XXXXXXXXXXXXXXXXXXXX
- Certificado Empresa: [Seleccionar archivo] No se eligió archivo
~/Documentos/Xml/Certificados/NUOVOSETCT1JXWIMTTdtak9QMA==.pfx
- Codigo de Distrito: 150103
- Estado: ACTIVO
- Ruc: 20267781151
- I.G.V: 18.00
- Password Sunat: XXXXXXXX
- Password Certificado: XXXXXXXXXXXXXXXX
- Codigo de Pais: PE
- Certificación: PRUEBA

At the bottom of the form are two buttons: 'Regresar' and 'Guardar'.

Figura 77: Estructura de Código – Vista Modulo Usuario – Modificar Parte 1

Fuente: Elaboración propia

```

@model JSUBInet.WebUbiNet.Models.EmpresaDetalleViewModel
@{
    ViewBag.Title = "Edit";
}
using (Html.BeginForm("Edit", "Empresas", null, FormMethod.Post, new { enctype = "multipart/form-data" }))
{
    @Html.AntiForgeryToken()
    @Html.HiddenFor(model => model.CertificadoEmpresa)
    @Html.HiddenFor(model => model.IdEmpresa)
    @Html.HiddenFor(model => model.IdUsuarioCreacion)
    @Html.HiddenFor(model => model.FechaCreacion)
    @Html.HiddenFor(model => model.IdUsuarioModificacion)
    @Html.HiddenFor(model => model.FechaModificacion)
    <div class="container al">
        <div class="panel panel-default">
            <div class="panel-heading">
                <h3 class="panel-title">Editar Empresa</h3>
                @Html.ValidationSummary(false, "", new { @class = "text-danger" })
            </div>
            <div class="panel-body">
                <div class="form-horizontal">
                    <div class="row">
                        <div class="form-group">
                            <div class="col-lg-6">
                                @Html.LabelFor(model => model.RazonSocial, htmlAttributes: new { @class = "control-label col-md-4" })
                                @Html.EditorFor(model => model.RazonSocial, new { htmlAttributes = new { @class = "form-control" } })
                                @Html.ValidationMessageFor(model => model.RazonSocial, "", new { @class = "text-danger" })
                            </div>
                            <div class="col-md-6">
                                @Html.LabelFor(model => model.RucEmpresa, htmlAttributes: new { @class = "control-label col-md-4" })
                                @Html.EditorFor(model => model.RucEmpresa, new { htmlAttributes = new { @class = "form-control", @style = "width:25%" } })
                                @Html.ValidationMessageFor(model => model.RucEmpresa, "", new { @class = "text-danger" })
                            </div>
                        </div>
                    </div>
                    <div class="row"></div>
                    <div class="row"></div>
                    <div class="row"></div>
                    <div class="row"></div>
                    <div class="row"></div>
                    <div class="row">
                        <div class="form-group">
                            <div class="col-lg-12">
                                <div class="col-lg-6">
                                    @Html.ActionLink("Regresar", "Index", null, htmlAttributes: new { @class = "btn btn-primary", @style = "color: white; font-family: 'Arial'; width: 100%" })
                                </div>
                                <div class="col-lg-6">
                                    <button type="submit" class="btn btn-primary" style="width: 100%">Guardar</button>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
}

```


RF05: El sistema debe permitir listar y modificar los productos relacionados al comprobante de pago electrónico emitido.

- Modulo Productos – Operación Listar

Figura 80: Vista Modulo Productos – Operación Listar

Fuente: Elaboración propia

Cod.producto	Nombre	Unidad de Medida	Costo unitario	N° de serie	Facturable	Almacenable	Consumible	Compuesto	Depreciable	Estado
SF	SERVICE FEE	UNI	0.00	SF	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVO
TK	TICKETS AEREOS	UNI	0.00	TK	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVO
CP	COMISION DE PROVEEDORES	UNI	0.00	CP	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVO
AJ	ASESORIA VISA JAPON	UNI	0.00	AJ	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVO
TA	TARJETA ASISTENCIA-TARJET...	UNI	0.00	TA	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVO
PN	PAQUETE NACIONAL	UNI	0.00	PN	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVO
PI	PAQUETE INTERNACIONAL	UNI	0.00	PI	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVO
AU	ASESORIA VISA USA	UNI	0.00	AU	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVO
HA	HOTEL ALQUAMIENTO	UNI	0.00	HA	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVO
BC	BACKEND	UNI	0.00	BC	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVO
PA	PAGO DE ASIENTOS	UNI	0.00	PA	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVO
CH	ASESORIA VISA CHINA	UNI	0.00	CH	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVO
ME	MALETA EXTRA	UNI	0.00	ME	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVO
SE	SERVICIOS "ESTA"	UNI	0.00	SE	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVO
JP	JAPAN RAIL PASS	UNI	0.00	JP	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVO
PR	PAGO DE REPORTE	UNI	0.00	PR	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ACTIVO

Figura 81: Estructura de Código – Vista Modulo Productos – Listar Parte 1

Fuente: Elaboración propia

```

@model ProductListaViewModel
@{
    ViewBag.Title = "Productos";
}
@{Html.C1().CollectionViewService<ProductListaViewModel>()
    .Id("collectionViewService")
    .PageSize(Model.PageSize)
}
<div class="well grid-sort-group">
    @Html.C1().Pager().Owner("collectionViewService")
    <label> Empresa :</label>
    @Html.C1().AutoComplete()
        .Width(210)
        .Height(34)
        .Bind(Model.empresas)
        .Id("cboEmpresa")
        .MaxItems(10)
        .CssMatch("highlight")
        .DisplayMemberPath("name")
        .SelectedValuePath("value")
        .Disabled(!User.IsInRole("ADMINISTRADOR"))
}
<input type="text" class="form-inline" id="NombreProducto" placeholder="Nombre Producto" style="height:34px;font-size:14px;border-radius:4px;border: 1px solid #cccccc">
<input type="text" class="form-inline" id="CodigoProducto" placeholder="Cod. Producto" style="height:34px;width:95px;font-size:14px;border-radius:4px;border: 1px solid #cccccc">
<button id="btnFilter" class="btn btn-default">
    <span class="glyphicon wj-glyph-filter"></span>
</button>
<button id="btnReset" onclick="resetRow()" class="btn btn-default" title="Reset">
    <span class="glyphicon glyphicon-refresh"></span>
</button>
<button id="btnReport" onclick="reportRow()" class="btn btn-default" title="Reporte">
    <span class="glyphicon wj-glyph-file"></span>
</button>
</div>
<script id="EditTemplate" type="text/html">
    <div>
        <div class="boton-new" @((User.IsInRole("Producto Editar") ? "" : "disabled")>
            <span class="glyphicon glyphicon-pencil ft" onclick=" editRow()" title="Editar"></span>
        </div>
    </div>
</script>

```

Figura 82: Estructura de Código – Vista Modulo Productos – Listar Parte 2

Fuente: Elaboración propia

```

@{Html.C1().FlexGrid<ProductoListaViewModel>()
    .Id("pagingGrid")
    .Height(440)
    .ItemsSourceId("collectionViewService")
    .IsReadOnly(true)
    .AllowDelete(true)
    .AutoGenerateColumns(false)
    .Columns(c1sb =>
    {
        c1sb.Add(cb => cb.Binding("CodigoProducto").Align("center").Header("Cod.producto").Width("90"));
        c1sb.Add(cb => cb.Binding("NombreProducto").Header("Nombre").Width("190"));
        c1sb.Add(cb => cb.Binding("UnidadMedida").Align("center").Header("Unidad de Medida").Width("110"));
        c1sb.Add(cb => cb.Binding("CostoUnitario").Header("Costo unitario").Width("90").Format("n2").Align("right"));
        c1sb.Add(cb => cb.Binding("NroSerie").Header("Nº de serie").Width("75").Align("center"));
        c1sb.Add(cb => cb.Binding("EsFacturable").Header("Facturable").Width("80"));
        c1sb.Add(cb => cb.Binding("EsAlmacenable").Header("Almacenable").Width("85"));
        c1sb.Add(cb => cb.Binding("EsConsumible").Header("Consumible").Width("80"));
        c1sb.Add(cb => cb.Binding("EsCompuesto").Header("Compuesto").Width("80"));
        c1sb.Add(cb => cb.Binding("EsDepreciable").Header("Depreciable").Width("80"));
        c1sb.Add(cb => cb.Binding("DescripcionEstado").Header("Estado").Width("75"));
        c1sb.Add().Name("EditButton").CellTemplate(b => b.TemplateId("EditTemplate")).Width("33");
    }
    )
}
<div class="well grid-sort-group">
    @{Html.C1().Pager().Owner("pagingGrid")}
</div>

```

Figura 83: Estructura de Código – Controlador Modulo Productos - Listar

Fuente: Elaboración propia

```

#region Listar
[AuthorizeApp(Roles = "Producto Listar")]
--referencias
public ActionResult Index(int PageInitial = 0)
{
    ProductoListaViewModel oProductoViewModel = new ProductoListaViewModel()
    {
        IdEmpresa = -1,
        empresas = new List<AutoCompleteElemento<long>>(),
    };

    DataRepository.UsuarioProvider.DeepLoad(oUsuario, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Empresa), typeof(Estado) });
    oProductoViewModel.empresas.Add(new AutoCompleteElemento<long>() { name = oUsuario.IdEmpresaSource.RazonSocial, value = oUsuario.IdEmpresa });
    if (User.IsInRole("ADMINISTRADOR"))
        oProductoViewModel.empresas.Insert(0, new AutoCompleteElemento<long>() { name = "TODOS", value = -1 });

    oProductoViewModel.PageSize = 16;
    oProductoViewModel.PageInitial = PageInitial;
    return View(oProductoViewModel);
}
#endregion

#region Filtrar
--referencias
public ActionResult RemoteBind_Read([C1JsonRequest] CollectionViewRequest<ProductoListaViewModel> requestData, long IdEmpresa, string NombreProducto, string CodigoProducto, int
{
    int nCount = 0;
    requestData.PageIndex = (PageInitial == 0 ? requestData.PageIndex : PageInitial);
    object[] cparams = { IdEmpresa, NombreProducto, CodigoProducto };
    string cwhere = string.Format("({0} = -1 OR IdEmpresa = {0}) AND NombreProducto like '{1}%' AND CodigoProducto like '{2}%' ", cparams);
    TList<Producto> oProductos = DataRepository.ProductoProvider.GetPaged(cwhere, "", requestData.PageIndex, requestData.PageSize, out nCount);
    if (oProductos.Count > 0)
        DataRepository.ProductoProvider.DeepLoad(oProductos, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Empresa), typeof(Estado) });

    List<ProductoListaViewModel> oProductoModelViews = Mapper.Map<List<Producto>, List<ProductoListaViewModel>>(oProductos.ToList());
    CollectionViewResponse<ProductoListaViewModel> responseData = CollectionViewHelper.Read(requestData, oProductoModelViews);
    responseData.PageIndex = requestData.PageIndex;
    responseData.TotalItemCount = nCount;
    return this.C1Json(responseData);
}
#endregion

```

- **Modulo Productos – Operación Modificar**

Figura 84: Vista Modulo Productos – Operación Modificar

Fuente: Elaboración propia

Figura 85: Estructura de Código – Vista Modulo Productos – Modificar Parte 1

Fuente: Elaboración propia

```

@model JSUBInet.WebUI.Net.Models.ProductoDetalleViewModel
@{
    ViewBag.Title = "Edit";
}
using (Html.BeginForm())
{
    @Html.AntiForgeryToken()
    @Html.HiddenFor(model => model.IdEmpresa)
    @Html.HiddenFor(model => model.IdProducto)
    @Html.HiddenFor(model => model.IdUsuarioCreacion)
    @Html.HiddenFor(model => model.FechaCreacion)
    @Html.HiddenFor(model => model.IdUsuarioModificacion)
    @Html.HiddenFor(model => model.FechaModificacion)
    @Html.HiddenFor(model => model.NroSerie)
    @Html.HiddenFor(model => model.CodigoBarra)
    <div class="container al">
        <div class="panel panel-default">
            <div class="panel-heading">
                <h3 class="panel-title">Editar Producto/</h3>
                @Html.ValidationSummary(false, "", new { @class = "" })
            </div>
            <div class="panel-body">
                <div class="form-horizontal">
                    <div class="row">
                        <div class="form-group">
                            <div class="col-lg-6">
                                @Html.LabelFor(model => model.CodigoProducto, htmlAttributes: new { @class = "control-label col-md-4" })
                                @Html.EditorFor(model => model.CodigoProducto, new { htmlAttributes = new { @class = "form-control" } })
                                @Html.ValidationMessageFor(model => model.CodigoProducto, "", new { @class = "text-danger" })
                            </div>
                            <div class="col-lg-6">
                                @Html.LabelFor(model => model.NombreProducto, htmlAttributes: new { @class = "control-label col-md-4" })
                                @Html.EditorFor(model => model.NombreProducto, new { htmlAttributes = new { @class = "form-control" } })
                                @Html.ValidationMessageFor(model => model.NombreProducto, "", new { @class = "text-danger" })
                            </div>
                        </div>
                    </div>
                    <div class="row">
                        <div class="row"></div>
                        <div class="row"></div>
                        <div class="row"></div>
                        <div class="row"></div>
                        <div class="row"></div>
                    </div>
                    <div class="form-group">
                        <div class="col-lg-12">
                            <div class="col-lg-6">
                                @Html.ActionLink("Regresar", "Index", null, htmlAttributes: new { @class = "btn btn-primary", @style = "color: white; font-family: Arial;width:100%" })
                            </div>
                            <div class="col-lg-6">
                                <button type="submit" class="btn btn-primary" style="width:100%">Guardar</button>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
}

```

Figura 86: Estructura de Código – Vista Modulo Productos – Modificar Parte 2

Fuente: Elaboración propia

```
<div class="panel-footer pf">
  <div class="row">
    <div class="form-group">
      <div class="col-lg-6">
        <label class="control-label col-md-4"><span style="border:1px solid white" class="glyphicon glyphicon-user"></span> Creado por:</label> @Model.UsuarioCreacion
      </div>
      <div class="col-lg-6">
        <label class="control-label col-md-6">
          <span class="glyphicon glyphicon-calendar"></span> Fecha Creacion:
        </label>
        @Model.FechaCreacion.ToShortDateString()
        @Model.FechaCreacion.ToShortTimeString()
      </div>
    </div>
  </div>
  <div class="row">
    <div class="form-group">
      <div class="col-lg-6">
        <label class="control-label col-md-4"><span style="border:1px solid white" class="glyphicon glyphicon-user"></span> Modificado por:</label> @Model.UsuarioModificacion
      </div>
      <div class="col-lg-6">
        <label class="control-label col-md-6">
          <span class="glyphicon glyphicon-calendar"></span> Fecha Modificacion:
        </label>
        @if (Model.FechaModificacion.HasValue)
        {
          @Model.FechaModificacion.Value.ToShortDateString() + " " +
          @Model.FechaModificacion.Value.ToShortTimeString()
        }
      </div>
    </div>
  </div>
</div>
</div>
</div>
@section Scripts {
  @Scripts.Render("~/bundles/jqueryval")
}
```

Figura 87: Estructura de Código – Controlador Modulo Productos - Modificar

Fuente: Elaboración propia

```
#region Editar
[Authorize(Roles = "Producto Editar")]
public ActionResult Edit(int id, int PageInitial = 0)
{
    ProductoDetalleViewModel oProductoDetalleModel = new ProductoDetalleViewModel();
    Producto oProducto = DataRepository.ProductoProvider.GetByIdProducto(id);
    DataRepository.ProductoProvider.DeepLoad(oProducto, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Usuario), typeof(Estado), typeof(Empresa) });
    oProductoDetalleModel = Mapper.Map<Producto, ProductoDetalleViewModel>(oProducto);
    oProductoDetalleModel.PageInitial = PageInitial;
    oProductoDetalleModel.estados = DataRepository.EstadoProvider.GetAll().ToList();
    return View(oProductoDetalleModel);
}
[Authorize(Roles = "Producto Editar")]
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult Edit(int id, ProductoDetalleViewModel oProductoDetalleViewModel)
{
    try
    {
        if (!ModelState.IsValid)
            throw new Exception("Formulario Invalido");
        Producto oProducto = Mapper.Map<ProductoDetalleViewModel, Producto>(oProductoDetalleViewModel);
        oProducto.IdEmpresa = oUsuario.IdEmpresa;
        oProducto.NroSerie = oProductoDetalleViewModel.NroSerie;
        oProducto.EntityState = EntityState.Changed;
        oProducto.IdUsuarioModificacion = oUsuario.IdUsuario;
        oProducto.FechaModificacion = DateTime.Now;
        DataRepository.ProductoProvider.Save(oProducto);
        this.AddNotification("Registro Grabado...", NotificationType.SUCCESS);
        return RedirectToAction("Index", new { PageInitial = oProductoDetalleViewModel.PageInitial });
    }
    catch (Exception ex)
    {
        this.AddNotification(ex.Message, NotificationType.ERROR);
        Empresa oEmpresa = DataRepository.EmpresaProvider.GetByIdEmpresa(oProductoDetalleViewModel.IdEmpresa);
        oProductoDetalleViewModel.estados = DataRepository.EstadoProvider.GetAll().ToList();
        oProductoDetalleViewModel.empresas = new List<AutoCompleteElemento<long>>() { new AutoCompleteElemento<long>() { name = oEmpresa.RazonSocial, value = oEmpresa.IdEmpresa } };
        return View(oProductoDetalleViewModel);
    }
}
#endregion
```

RF06: El sistema debe permitir listar, registrar y modificar tipo de cambio a la fecha a emitir.

- Modulo Tipo de Cambio – Operación Listar

Figura 88: Vista Modulo Tipo de Cambio – Operación Listar

Fuente: Elaboración propia

Tipo Cambio	Fecha Cambio	Monto	monedas	Empresa	Usuario Creación	Fecha Creación	Usuario Modificación	Fecha Modificación	Estado		
18	20/08/2018	3.30	USD	CONTINENTAL TRAVEL S.A.C.	Admin CTravel	20/08/2018			ACTIVO	/	
19	21/08/2018	3.30	USD	CONTINENTAL TRAVEL S.A.C.	Admin CTravel	21/08/2018			ACTIVO	/	
22	22/08/2018	3.30	USD	CONTINENTAL TRAVEL S.A.C.	Admin CTravel	21/08/2018			ACTIVO	/	
23	23/08/2018	3.30	USD	CONTINENTAL TRAVEL S.A.C.	Admin CTravel	23/08/2018			ACTIVO	/	
24	24/08/2018	3.30	USD	CONTINENTAL TRAVEL S.A.C.	Admin CTravel	24/08/2018			ACTIVO	/	
25	25/08/2018	3.30	USD	CONTINENTAL TRAVEL S.A.C.	Admin CTravel	25/08/2018			ACTIVO	/	
26	27/08/2018	3.30	USD	CONTINENTAL TRAVEL S.A.C.	Admin CTravel	27/08/2018			ACTIVO	/	
27	28/08/2018	3.30	USD	CONTINENTAL TRAVEL S.A.C.	Admin CTravel	28/08/2018			ACTIVO	/	
28	29/08/2018	3.32	USD	CONTINENTAL TRAVEL S.A.C.	Admin CTravel	29/08/2018			ACTIVO	/	
29	31/08/2018	3.31	USD	CONTINENTAL TRAVEL S.A.C.	Admin CTravel	31/08/2018			ACTIVO	/	

Figura 89: Estructura de Código – Vista Modulo Tipo de Cambia – Listar Parte 1

Fuente: Elaboración propia

```

using System.Web.UI.WebControls;
@model TipoCambioListaViewModel
@{
    ViewBag.Title = "TipoCambios";
    var fecini = DateTime.Now.AddMonths(-1).Date;
    var fecfin = DateTime.Now.Date.AddDays(1).Date;
}
@Html.Ci().CollectionViewService<TipoCambioListaViewModel>()
    .Id("collectionViewService")
    .PageSize(Model.PageSize)
}
<div class="well grid-sort-group">
    @Html.Ci().Pager().Owner("collectionViewService")
    <button id="NewBtn" class="btn btn-default" title="Nuevo" @User.IsInRole("Tipo Cambio Crear") ? "" : "disabled"
        onclick="location.href=@Url.Action("create")"> <span class="glyphicon glyphicon-edit"></span>
    </button>
    <label>Inicio :</label>
    @Html.Ci().InputDate().Id("FechaInicio").Width(120).Value(fecini)
    <label>Fin : </label>
    @Html.Ci().InputDate().Id("FechaFinal").Width(120).Value(fecfin)
    <label> Empresa :</label>
    @Html.Ci().AutoComplete().Width(210)
        .Bind(Model.OfEmpresas).Id("cbOfEmpresas")
        .MaxItems(10)
        .CssMatch("highlight")
        .DisplayMemberPath("name")
        .SelectedValuePath("value")
        .Disabled(User.IsInRole("ADMINISTRADOR"))
    <button id="btnFilter" class="btn btn-default" title="Filtro">
        <span class="glyphicon wj-glyph-filter"></span>
    </button>
    <button id="btnReport" onclick="reportRow()" class="btn btn-default" title="Reporte">
        <span class="glyphicon wj-glyph-file"></span>
    </button>
</div>
<script id="EditTemplate" type="text/html">
<div>
    <div class="boton-nuevo" @User.IsInRole("Tipo Cambio Editor") ? "" : "disabled">
        <span class="glyphicon glyphicon-pencil ft" onclick="editRow()" title="Editar"></span>
    </div>
</div>
</script>
<script id="DeleteTemplate" type="text/html">
<div>
    <div class="boton-nuevo" @User.IsInRole("Tipo Cambio Eliminar") ? "" : "disabled">
        <span class="glyphicon glyphicon-trash ft" onclick="deleteRow()" title="Eliminar"></span>
    </div>
</div>
</script>
<script id="PrintTemplate" type="text/html">
<div>
    <button class="btn btn-default btn-sm" onclick="PrintRow()">
        <span class="glyphicon glyphicon-print"></span>
    </button>
</div>
</script>

```

Figura 90: Estructura de Código – Vista Modulo Tipo de Cambio – Listar Parte 2

Fuente: Elaboración propia

```

@Html.C1().FlexGrid(<tipoCambioListaViewModel>{
    .Id("pagingGrid")
    .Height(465)
    .ItemsSourceId("collectionViewService")
    .IsReadOnly(true)
    .AllowDelete(true)
    .AutoGenerateColumns(false)
    .Columns(c1sb =>
        {
            c1sb.Add(cb => cb.Binding("IdTipoCambio").Header("Tipo Cambio").Width("80").Align("center"));
            c1sb.Add(cb => cb.Binding("FechaCambio").Header("Fecha Cambio").Width("100").Align("center"));
            c1sb.Add(cb => cb.Binding("MontoCambio").Header("Monto").Width("50").Format("n2").Align("right"));
            c1sb.Add(cb => cb.Binding("monedas").Header("monedas").Width("60").Align("center"));
            c1sb.Add(cb => cb.Binding("empresas").Header("Empresa").Width("195"));
            c1sb.Add(cb => cb.Binding("UsuarioCreacion").Header("Usuario Creación").Width("110"));
            c1sb.Add(cb => cb.Binding("FechaCreacion").Header("Fecha Creación").Width("110").Format("dd/MM/yyyy"));
            c1sb.Add(cb => cb.Binding("UsuarioModificacion").Header("Usuario Modificación").Width("130"));
            c1sb.Add(cb => cb.Binding("FechaModificacion").Header("Fecha Modificación").Width("120").Format("dd/MM/yyyy"));
            c1sb.Add(cb => cb.Binding("DescripcionEstado").Header("Estado").Width("70").Align("center"));
            c1sb.Add().Name("EditButton").CellTemplate(b => b.TemplateId("EditTemplate")).Width("33");
            c1sb.Add().Name("DeleteButton").CellTemplate(b => b.TemplateId("DeleteTemplate")).Width("33");
        }
    )
}
</div class="well grid-sort-group">
@Html.C1().Pager().Owner("pagingGrid")
</div>
@section Scripts {
    @Scripts.Render("~/bundles/jqueryval")
    <script type="text/javascript">
        var pagingGrid, fechaInicio, fechaFinal, oCollection, cboEmpresa;
        c1.mvc.Utils.documentReady(function () {
            pagingGrid = wijmo.Control.getControl("#pagingGrid");
            fechaInicio = wijmo.Control.getControl("#fechaInicio");
            fechaFinal = wijmo.Control.getControl("#fechaFinal");
            cboEmpresa = wijmo.Control.getControl("#cboEmpresa");
            if (pagingGrid) {
                oCollection = pagingGrid.collectionView;
                pagingGrid.rows.defaultSize = 25;
                updateUrl();
                oCollection.refresh();
            }

            var dvFilter = document.getElementById('btnFilter');
            dvFilter.addEventListener("click", function () {
                updateUrl();
                oCollection.refresh();
            });
            oCollection.currentChanging.addHandler(function () {
                updateUrl();
            });
        });
        function editRow() {
            if (!pagingGrid || !oCollection) {
                return;
            }
            pagingGrid.invalidate();
            window.location.href = '@Url.Action("Edit", "TipoCambios")?id=' + oCollection.currentItem.IdTipoCambio + '&PageInitial=' + oCollection.pageIndex;
        }
    </script>
}

```

Figura 91: Estructura de Código – Vista Modulo Tipo de Cambio – Listar Parte 3

Fuente: Elaboración propia

```

function reportRow() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    var url = '@Url.Action("Report", "TipoCambios")?FechaInicio='
        + encodeURIComponent(formatDate(fechaInicio.value)) + "&FechaFinal="
        + encodeURIComponent(formatDate(fechaFinal.value)) + "&IDEmpresa="
        + encodeURIComponent(cboEmpresa.selectedValue) + "&PageInitial="
        + oCollection.pageIndex;
    window.open(url, 'Reporte Empresa');
};

function deleteRow() {
    if (!confirm("Esta seguro que desea borrar este registro.") || !pagingGrid || !oCollection) {
        oCollection.moveCurrentToPosition(-1);
        return;
    }
    pagingGrid.invalidate();
    var rowIndex = pagingGrid.selection.row;
    oCollection.removeAt(rowIndex);
    oCollection.moveCurrentToPosition(-1);
};

function updateUrl() {
    oCollection._readActionUrl = '@Url.Action("RemoteBind_Read", "TipoCambios")?FechaInicio='
        + encodeURIComponent(formatDate(fechaInicio.value)) + "&FechaFinal="
        + encodeURIComponent(formatDate(fechaFinal.value)) + "&IDEmpresa="
        + encodeURIComponent(cboEmpresa.selectedValue) + "&PageInitial=0";

    oCollection._deleteActionUrl = '@Url.Action("Delete", "TipoCambios")?FechaInicio='
        + encodeURIComponent(formatDate(fechaInicio.value)) + "&FechaFinal="
        + encodeURIComponent(formatDate(fechaFinal.value)) + "&IDEmpresa="
        + encodeURIComponent(cboEmpresa.selectedValue);
};
</script>

```

Figura 92: Estructura de Código – Controlador Modulo Tipo de Cambio – Listar

Fuente: Elaboración propia

```

#region Listar
[Authorize(Roles = "Tipo Cambio Listar")]
--referencias
public ActionResult Index(int PageInitial = 0)
{
    TipoCambioListaViewModel oTipoCambioDetalleViewModel = new TipoCambioListaViewModel() { IdEmpresa = -1, oEmpresa = new List<AutoCompleteElemento<long>>() };

    DataRepository.UsuarioProvider.DeepLoad(oUsuario, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Empresa) });
    oTipoCambioDetalleViewModel.oEmpresa.Add(new AutoCompleteElemento<long>() { name = oUsuario.IdEmpresaSource.RazonSocial, value = oUsuario.IdEmpresa });
    if (User.IsInRole("ADMINISTRADOR"))
        oTipoCambioDetalleViewModel.oEmpresa.Insert(0, new AutoCompleteElemento<long>() { name = "0005", value = -1 });

    oTipoCambioDetalleViewModel.PageSize = 16;
    oTipoCambioDetalleViewModel.PageInitial = PageInitial;
    return View(oTipoCambioDetalleViewModel);
}
#endregion

#region Operacion Remotas
--referencias
public ActionResult RemoteBind_Read([CllsonRequest] CollectionViewRequest<TipoCambioListaViewModel> requestData, DateTime FechaInicio, DateTime FechaFinal, long IdEmpresa, int PageInitial = 0)
{
    int nCount = 0;
    requestData.PageIndex = (PageInitial == 0 ? requestData.PageIndex : PageInitial);

    object[] cparams;
    if (ConfigurationManager.AppSettings["Region"].ToString().Equals("US"))
    {
        object[] cparamsTipo = { FechaInicio.ToString("yyyy/MM/dd"), FechaFinal.ToString("yyyy/MM/dd"), IdEmpresa };
        cparams = cparamsTipo;
    }
    else
    {
        object[] cparamsTipo = { FechaInicio.ToString("dd/MM/yyyy"), FechaFinal.ToString("dd/MM/yyyy"), IdEmpresa };
        cparams = cparamsTipo;
    }

    string cWhere = string.Format("[FechaCambio] BETWEEN '{0}' AND '{1}' AND ((2) = -1 OR IdEmpresa = {2})", cparams);
    TList<TipoCambio> oTipoCambios = DataRepository.TipoCambioProvider.GetPaged(cWhere, "", requestData.PageIndex, requestData.PageSize, out nCount);
    if (oTipoCambios.Count > 0)
        DataRepository.TipoCambioProvider.DeepLoad(oTipoCambios, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Estado), typeof(Usuario), typeof(Empresa), typeof(Moneda) });

    List<TipoCambioListaViewModel> oTipoCambioModelViews = Mapper.Map<List<TipoCambio>, List<TipoCambioListaViewModel>>(oTipoCambios.ToList());
    CollectionViewResponse<TipoCambioListaViewModel> responseData = CollectionViewHelper.Read(requestData, oTipoCambioModelViews);
    responseData.PageIndex = requestData.PageIndex;
    responseData.TotalItemCount = nCount;
    return this.Cllson(responseData);
}
}

```

- **Modulo Tipo de Cambio – Operación Registrar**

Figura 93: Vista Modulo Tipo de Cambio – Opción Registrar

Fuente: Elaboración propia

The screenshot shows a web application interface for creating a new exchange rate. At the top, there is a navigation bar with the 'continentaltravel' logo and menu items: Inicio, Nosotros, Administración, Facturación, Reportes, Contáctenos, Bienvenido admin, and Cerrar sesión. Below the navigation bar, the breadcrumb trail reads 'Inicio > Administración > Tipo Cambio > Crear'. The main content area is titled 'Nuevo Tipo de Cambio' and contains a form with the following fields:

- Fecha de Cambio:** A date picker set to 06/04/2019.
- Monto de Cambio:** A text input field containing 0.00.
- Empresa:** A dropdown menu showing 'CONTINENTAL TRAVEL S.A.C'.
- Estado:** A dropdown menu showing 'ACTIVO'.
- Monedas:** A dropdown menu showing 'DOLAR'.

At the bottom of the form, there are two blue buttons: 'Regresar' and 'Guardar'.

- **Modulo Tipo de Cambio – Operación Modificar**

Figura 96: Vista Modulo Tipo de Cambio – Operación Modificar

Fuente: Elaboración propia

Figura 97: Estructura de Código – Vista Modulo Tipo de Cambio – Modificar Parte 1

Fuente: Elaboración propia

```

@model System.Web.Mvc.Models.TipoCambioDetalleViewModel
@{
    ViewBag.Title = "Edit";
}
@using (Html.BeginForm())
{
    @Html.AntiForgeryToken()
    @Html.HiddenFor(model => model.IdTipoCambio)
    @Html.HiddenFor(model => model.IdEmpresa)
    @Html.HiddenFor(model => model.IdUsuarioCreacion)
    @Html.HiddenFor(model => model.FechaCreacion)
    @Html.HiddenFor(model => model.IdUsuarioModificacion)
    @Html.HiddenFor(model => model.FechaModificacion)
    @Html.HiddenFor(model => model.PageInitial)
    <div class="container al">
        <div class="panel panel-default">
            <div class="panel-heading">
                <h3 class="panel-title">Editar Tipo de Cambio/</h3>
                @Html.ValidationSummary(false, "", new { @class = "text-danger" })
            </div>
            <div class="panel-body">
                <div class="form-horizontal">
                    <div class="form-group">
                        <div class="col-lg-6">
                            @Html.LabelFor(model => model.FechaCambio, htmlAttributes: new { @class = "control-label col-md-4" })
                            @Html.EditorFor(model => model.FechaCambio, new { @htmlAttributes = new { @class = "form-control", @style = "width:150%" } })
                            @Html.ValidationMessageFor(model => model.FechaCambio, "", new { @class = "text-danger" })
                        </div>
                        <div class="col-lg-6">
                            @Html.LabelFor(model => model.MontoCambio, htmlAttributes: new { @class = "control-label col-md-4" })
                            @Html.EditorFor(model => model.MontoCambio, new { @htmlAttributes = new { @class = "form-control", @style = "width:150%" } })
                            @Html.ValidationMessageFor(model => model.MontoCambio, "", new { @class = "text-danger" })
                        </div>
                    </div>
                    <div class="form-group">
                        <div class="col-lg-12">
                            @Html.LabelFor(model => model.Empresa, htmlAttributes: new { @class = "control-label col-md-4" })
                            @Html.EditorFor(model => model.Empresa, new { @htmlAttributes = new { @class = "form-control", @style = "width:150%" } })
                            @Html.ValidationMessageFor(model => model.Empresa, "", new { @class = "text-danger" })
                        </div>
                    </div>
                    <div class="form-group">
                        <div class="col-lg-12">
                            @Html.LabelFor(model => model.Estado, htmlAttributes: new { @class = "control-label col-md-4" })
                            @Html.EditorFor(model => model.Estado, new { @htmlAttributes = new { @class = "form-control", @style = "width:150%" } })
                            @Html.ValidationMessageFor(model => model.Estado, "", new { @class = "text-danger" })
                        </div>
                    </div>
                    <div class="form-group">
                        <div class="col-lg-12">
                            @Html.LabelFor(model => model.Monedas, htmlAttributes: new { @class = "control-label col-md-4" })
                            @Html.EditorFor(model => model.Monedas, new { @htmlAttributes = new { @class = "form-control", @style = "width:150%" } })
                            @Html.ValidationMessageFor(model => model.Monedas, "", new { @class = "text-danger" })
                        </div>
                    </div>
                </div>
                <div class="form-group">
                    <div class="col-lg-12">
                        @Html.ActionLink("Regresar", "Index", null, htmlAttributes: new { @class = "btn btn-primary", @style = "color: white; font-family: Arial; width:100%" })
                        <div class="col-lg-6">
                            <button type="submit" class="btn btn-primary" style="width:100%">Guardar</button>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
}

```


RF07: El sistema debe permitir listar, registrar y modificar moneda a emitir según el comprobante de pago electrónico generado.

- Modulo Moneda – Operación Listar

Figura 100: Vista Modulo Moneda – Operación Listar

Fuente: Elaboración propia

Empresa	Nombre Moneda	Símbolo Moneda	Usuario Creación	Fecha Creación	Usuario Modificación	Fecha Modificación	Estado
CONTINENTAL TRAVEL S.A.C.	SOLES	S/	Admin CTravel	01/08/2018			ANULADO
CONTINENTAL TRAVEL S.A.C.	DOLAR	USD	Admin CTravel	01/08/2018			ACTIVO
CONTINENTAL TRAVEL S.A.C.	EUROS	€	Admin CTravel	01/08/2018			ACTIVO
CONTINENTAL TRAVEL S.A.C.	PESO	P/	Admin CTravel	01/08/2018			ACTIVO

Figura 101: Estructura de Código – Vista Modulo Moneda – Listar Parte 1

Fuente: Elaboración propia

```

using JSUBInet.WebUIInet.Models;
@model MonedaListViewModel
@{
    ViewBag.Title = "Monedas";
}
@(Html.CI().CollectionViewService<MonedaListViewModel>()
    .Id("collectionViewService")
    .PageSize(Model.PageSize)
)
<div class="well grid-sort-group">
    @(Html.CI().Pager().Owner("collectionViewService"))
    <button id="NewBtn"
        class="btn btn-default title="Nuevo" @(User.IsInRole("Moneda Crear") ? "" : "disabled")
        onclick="Location.href=@Url.Action("create")">
        <span class="glyphicon glyphicon-edit"></span>
    </button>
    <label> Empresa : </label>
    @(Html.CI().AutoComplete().Width(210).Height(34)
        .Bind(Model.oEmpresa)
        .Id("cboEmpresa")
        .MaxItems(10)
        .CssMatch("highlight")
        .DisplayMemberPath("name")
        .SelectedValuePath("value")
        .Disabled(!User.IsInRole("ADMINISTRADOR")))
    )
    <label> Buscar: </label>
    <input type="text" class="form-inline" id="NombreMoneda" placeholder="Nombre Moneda" style="height:34px;font-size:14px;border-radius:4px;border: 1px solid #cccccc">
    <button id="btnFilter" class="btn btn-default" title="Filtro">
        <span class="glyphicon wj-glyph-filter"></span>
    </button>
    <button id="btnReset" onclick="resetRow()" class="btn btn-default" title="Reset">
        <span class="glyphicon glyphicon-refresh"></span>
    </button>
    <button id="btnReport" onclick="reportRow()" class="btn btn-default" title="Reporte">
        <span class="glyphicon wj-glyph-file"></span>
    </button>
</div>
<script id="EditTemplate" type="text/html">
    <div>
        <div class="boton-new" @(User.IsInRole("Moneda Editar") ? "" : "disabled")>
            <span class="glyphicon glyphicon-pencil ft" onclick=" editRow()" title="Editar"></span>
        </div>
    </div>
</script>
<script id="DeleteTemplate" type="text/html">
    <div>
        <div class="boton-new" @(User.IsInRole("Moneda Eliminar") ? "" : "disabled")>
            <span class="glyphicon glyphicon-trash ft" onclick="deleteRow()" title="Eliminar"></span>
        </div>
    </div>
</script>

```

Figura 102: Estructura de Código – Vista Modulo Moneda – Listar Parte 2

Fuente: Elaboración propia

```

@Html.C1().FlexGrid<MonedaListViewModel>()
    .Id("pagingGrid")
    .Height(465)
    .ItemsSourceId("collectionViewService")
    .IsReadOnly(true)
    .AllowDelete(true)
    .AutoGenerateColumns(false)
    .Columns(clsb =>
    {
        clsb.Add(cb => cb.Binding("NombreEmpresa").Header("Empresa").Width("192"));
        clsb.Add(cb => cb.Binding("NombreMoneda").Header("Nombre Moneda").Width("120"));
        clsb.Add(cb => cb.Binding("SimboloMoneda").Header("Simbolo Moneda").Width("150"));
        clsb.Add(cb => cb.Binding("UsuarioCreacion").Header("Usuario Creación").Width("120"));
        clsb.Add(cb => cb.Binding("FechaCreacion").Header("Fecha Creación").Width("120").Format("dd/MM/yyyy"));
        clsb.Add(cb => cb.Binding("UsuarioModificacion").Header("Usuario Modificación").Width("130"));
        clsb.Add(cb => cb.Binding("FechaModificacion").Header("Fecha Modificación").Width("120").Format("dd/MM/yyyy"));
        clsb.Add(cb => cb.Binding("IdEstado").Binding("DescripcionEstado").Header("Estado").Width("80"));
        clsb.Add().Name("EditButton").CellTemplate(b => b.TemplateId("EditTemplate")).Width("35");
        clsb.Add().Name("DeleteButton").CellTemplate(b => b.TemplateId("DeleteTemplate")).Width("35");
    })
)
<div class="well grid-sort-group">
    @Html.C1().Pager().Owner("pagingGrid")
</div>
@section Scripts {
    @Scripts.Render("~/bundles/jqueryval")
    <script type="text/javascript">
        var pagingGrid, txtNombreMoneda, oCollection, cboEmpresa;
        cl.mvc.Utils.documentReady(function () {
            pagingGrid = wjmo.Control.getControl("#pagingGrid");
            cboEmpresa = wjmo.Control.getControl("#cboEmpresa");
            txtNombreMoneda = $("#NombreMoneda");
            if (pagingGrid) {
                oCollection = pagingGrid.collectionView;
                pagingGrid.rows.defaultSize = 25;
                updateUrl(@Model.PageInitial);
                oCollection.refresh();
            };
            var dvFilter = document.getElementById('btnFilter');
            dvFilter.addEventListener("click", function () {
                updateUrl(0);
                oCollection.refresh();
            });
            oCollection.currentChanging.addHandler(function () {
                updateUrl(0);
            })
        });
    </script>
}
    
```

Figura 103: Estructura de Código – Vista Modulo Moneda – Listar Parte 3

Fuente: Elaboración propia

```

function editRow() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    window.location.href = '@Url.Action("Edit", "Monedas")?id=' + oCollection.currentItem.IdMoneda + '&PageInitial=' + oCollection.pageIndex;
};

function reportRow() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    var url = '@Url.Action("Report", "Monedas")?NombreMoneda='
        + encodeURIComponent(txtNombreMoneda.val()) + '&IdEmpresa='
        + encodeURIComponent(cboEmpresa.selectedValue) + '&PageInitial='
        + oCollection.pageIndex;
    window.open(url, 'Reporte Moneda');
};

function resetRow() {
    $('#NombreMoneda').val('');
    updateUrl(0);
    oCollection.refresh();
};

function deleteRow() {
    if (!confirm("Esta seguro que desea borrar este registro.") || !pagingGrid || !oCollection) {
        oCollection.moveCurrentToPosition(-1);
        return;
    }
    pagingGrid.invalidate();
    var rowIndex = pagingGrid.selection.row;
    oCollection.removeAt(rowIndex);
    oCollection.moveCurrentToPosition(-1);
};

function updateUrl(PageInitial) {
    oCollection._readActionUrl = '@Url.Action("RemoteBind_Read", "Monedas")?NombreMoneda='
        + encodeURIComponent(txtNombreMoneda.val()) + '&IdEmpresa='
        + encodeURIComponent(cboEmpresa.selectedValue) + '&PageInitial='
        + encodeURIComponent(PageInitial);

    oCollection._deleteActionUrl = '@Url.Action("Delete", "Monedas")?NombreMoneda='
        + encodeURIComponent(txtNombreMoneda.val()) + '&IdEmpresa='
        + encodeURIComponent(cboEmpresa.selectedValue);
};
</script>
    
```

Figura 104: Estructura de Código Controlador Modulo Moneda – Listar

Fuente: Elaboración propia

```

#region Principal
[Authorize(Roles = "Moneda Listar")]
--referencias
public ActionResult Index(int PageInitial = 0)
{
    MonedaListViewModel oMonedaViewModel = new MonedaListViewModel() { IdEmpresa = -1, oEmpresa = new List<AutoCompleteElemento<long>>() };

    DataRepository.UsuarioProvider.DeepLoad(oUsuario, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Empresa) });
    oMonedaViewModel.oEmpresa.Add(new AutoCompleteElemento<long>() { name = oUsuario.IdEmpresaSource.RazonSocial, value = oUsuario.IdEmpresa });
    if (User.IsInRole("ADMINISTRADOR"))
        oMonedaViewModel.oEmpresa.Insert(0, new AutoCompleteElemento<long>() { name = "TODOS", value = -1 });

    oMonedaViewModel.PageSize = 16;
    oMonedaViewModel.PageInitial = PageInitial;
    return View(oMonedaViewModel);
}
#endregion

#region Filtro
--referencias
public ActionResult RemoteBind_Read([CancellationToken] CancellationToken requestData, string NombreMoneda, long IdEmpresa, int PageInitial = 0)
{
    int nCount = 0;
    requestData.PageIndex = (PageInitial == 0 ? requestData.PageIndex : PageInitial);
    string cWhere = string.Format("[NombreMoneda] LIKE '%{0}%' AND ((1) = -1 OR IdEmpresa = {1})", NombreMoneda.Trim(), IdEmpresa);
    TList<Moneda> oMoneda = DataRepository.MonedaProvider.GetPaged(cWhere, "", requestData.PageIndex, requestData.PageSize, out nCount);
    if (oMoneda.Count > 0)
        DataRepository.MonedaProvider.DeepLoad(oMoneda, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Estado), typeof(Usuario), typeof(Empresa) });

    List<MonedaListViewModel> oMonedaModelViews = Mapper.Map<List<Moneda>, List<MonedaListViewModel>>(oMoneda.ToList());
    CollectionViewResponse<MonedaListViewModel> responseData = CollectionViewHelper.Read(requestData, oMonedaModelViews);
    responseData.PageIndex = requestData.PageIndex;
    responseData.TotalItemCount = nCount;
    return this.CIJson(responseData);
}
#endregion
    
```

- Modulo Moneda – Operación Registrar

Figura 105: Vista Modulo Moneda – Operación Registrar

Fuente: Elaboración propia

The screenshot shows a web application interface for creating a new currency. At the top, there is a navigation bar with the 'continentaltravel' logo and menu items: Inicio, Nosotros, Administración, Facturación, Reportes, Contáctenos, Bienvenido admin, and Cerrar sesión. Below the navigation bar, the breadcrumb trail reads 'Inicio > Administración > Moneda > Crear'. The main content area is titled 'Nueva Moneda' and contains a form with the following fields: 'Nombre' (text input), 'Simbolo' (text input), 'Empresa' (dropdown menu showing 'CONTINENTAL TRAVEL S.A.C'), and 'Estado' (dropdown menu showing 'ACTIVO'). At the bottom of the form, there are two blue buttons: 'Regresar' and 'Guardar'.

- **Modulo Moneda – Operación Modificar**

Figura 108: Vista Modulo Moneda – Operación Modificar

Fuente: Elaboración propia

continentaltravel Inicio Nosotros Administración Facturación Reportes Contáctenos Bienvenido admin Cerrar sesión

Inicio > Administración > Moneda > Editar

Editar Moneda

Nombre: DOLAR Empresa: CONTINENTAL TRAVEL S.A.C.

Simbolo: USD Estado: ACTIVO

Regresar Guardar

Creado por: Admin CTravel
 Modificado por:

Fecha Creacion: 1/08/2018 10:32
 Fecha Modificacion:

Figura 109: Estructura de Código – Vista Modulo Moneda – Modificar Parte 1

Fuente: Elaboración propia

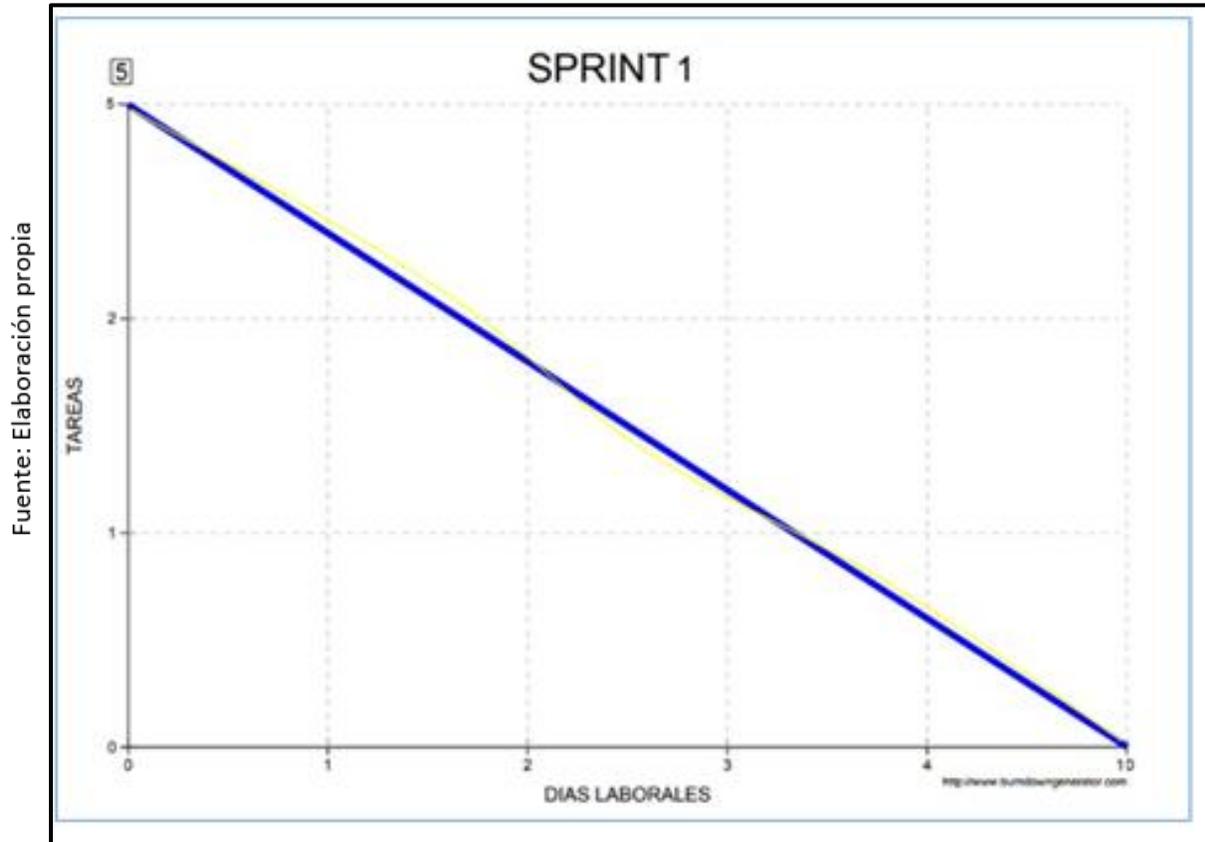
```

@model JSUBinet.WebUI.Net.Models.MonedaDetalleViewModel
@{
    ViewBag.Title = "Edit";
}
using (Html.BeginForm())
{
    @Html.AntiForgeryToken()
    @Html.HiddenFor(model => model.IdMoneda)
    @Html.HiddenFor(model => model.IdUsuarioCreacion)
    @Html.HiddenFor(model => model.FechaCreacion)
    <div class="container al">
        <div class="panel panel-default">
            <div class="panel-heading">
                <h3 class="panel-title">Editar Moneda</h3>
                @Html.ValidationSummary(false, "", new { @class = "text-danger" })
            </div>
            <div class="panel-body">
                <div class="form-horizontal">
                    <div class="row">
                        <div class="form-group">
                            <div class="col-lg-6">
                                @Html.LabelFor(model => model.NombreMoneda, htmlAttributes: new { @class = "control-label col-md-4" })
                                @Html.EditorFor(model => model.NombreMoneda, new { htmlAttributes = new { @class = "form-control" } })
                                @Html.ValidationMessageFor(model => model.NombreMoneda, "", new { @class = "text-danger" })
                            </div>
                            <div class="col-lg-6">
                                @Html.LabelFor(model => model.IdEmpresa, htmlAttributes: new { @class = "control-label col-md-4" })
                                @Html.DropDownListFor(model => model.IdEmpresa, new SelectList(Model.ObEmpresa, "value", "name", Model.IdEmpresa), htmlAttributes: new { @class = "form-control" })
                                @Html.ValidationMessageFor(model => model.IdEmpresa, "", new { @class = "text-danger" })
                            </div>
                        </div>
                    </div>
                    <div class="row">
                        <div class="form-group">
                            <div class="col-lg-12">
                                @Html.ActionLink("Regresar", "Index", null, htmlAttributes: new { @class = "btn btn-primary", @style = "color: white; font-family:'Arial';width:100%" })
                            </div>
                            <div class="col-lg-6">
                                <button type="submit" class="btn btn-primary" style="width:100%">Guardar</button>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
}

```


GRÁFICA DE BURN DOWN – SPRINT 1

Figura 112: Burn Down del Sprint N° 1



ACTA DE PLANIFICACION SPRINT N° 1

Figura 113: Acta de Planificación del Sprint N°1

Fuente: Elaboración propia

ACTA DE PLANIFICACION DEL SPRINT N° 1

La reunión se lleva a cabo en presencia de los siguientes participantes:

N°	Nombres y Apellidos	Rol
1	Sergio Moran Palacios	Product Owner
2	Ronnie Alarcon Morales	Scrum Master
3	Orvic Yampool Flores Principe	Team Member

El gerente de la empresa Continental Travel S.A.C. realizó la exposición de los requerimientos e indica los requerimientos con mayor prioridad.

CODIGO	TAREA
RF02	El sistema debe permitir listar, registrar, modificar y eliminar usuarios para gestionar el proceso de facturación electrónica.
RF03	El sistema debe permitir listar y modificar los clientes relacionados al comprobante de pago electrónico emitido.
RF04	El sistema debe permitir listar, modificar los datos de la empresa que emitirá la facturación electrónica.
RF05	El sistema debe permitir listar y modificar los productos relacionados al comprobante de pago electrónico emitido.
RF06	El sistema debe permitir listar, registrar y modificar tipo de cambio a la fecha a emitir.
RF07	El sistema debe permitir listar, registrar y modificar moneda a emitir según el comprobante de pago electrónico generado.

Mediante la presente acta se válida y se da conformidad los objetivos a realizarse en el Sprint 1 por parte del equipo de desarrollo, como también, las tareas propuestas del producto (Historias de Usuarios) que contiene el sprint mencionado.

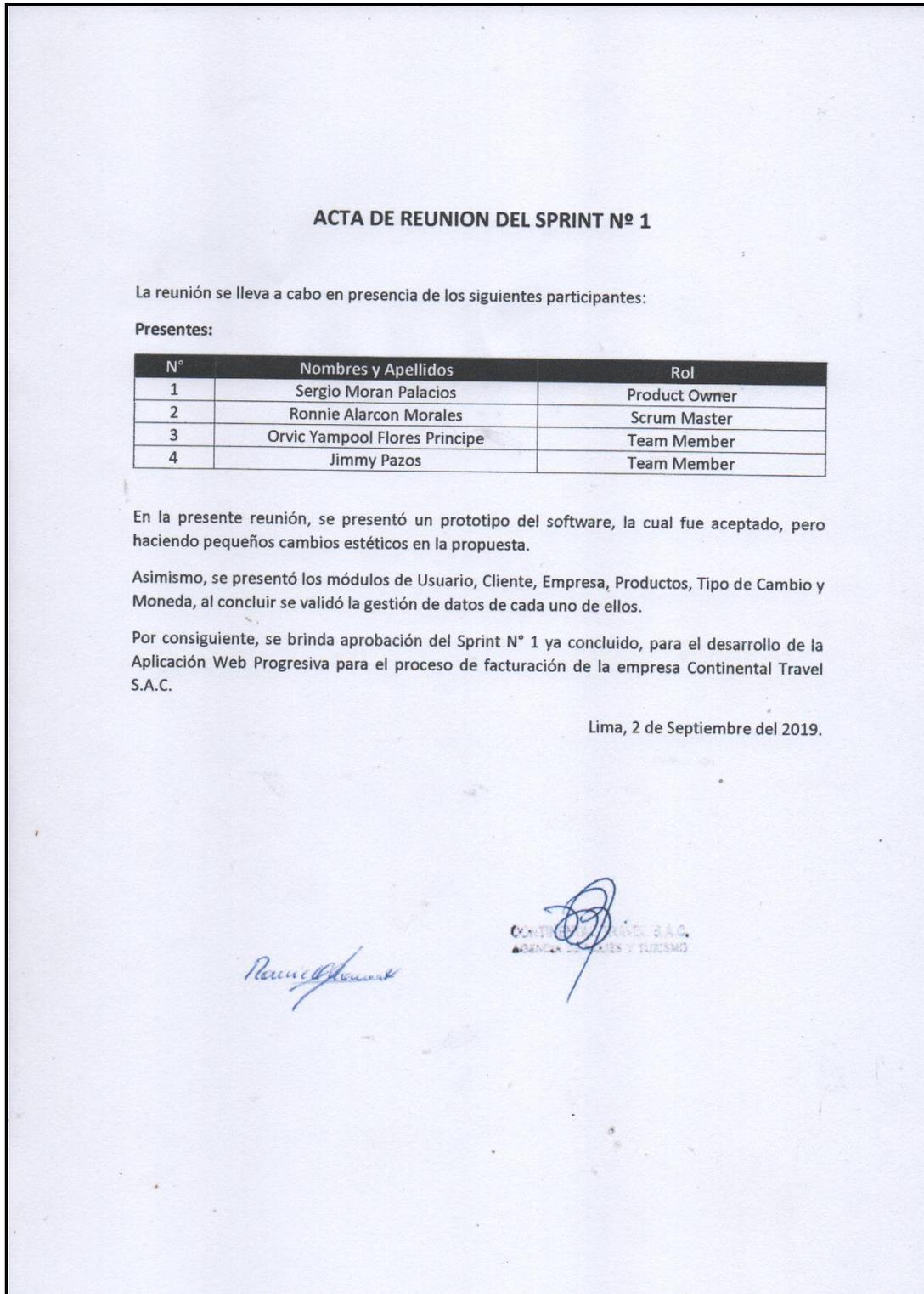
Los asistentes impartirán su aprobación de acuerdo a lo presentado en la planificación del Sprint 0, indicando que la fecha de entrega de este Sprint sería el día 2 de Septiembre del 2019.

Lima, 16 de Agosto del 2019.

CONTINENTAL TRAVEL S.A.C.
AGENCIA DE VIAJES Y TURISMO

ACTA REUNION SPRINT N° 1

Figura 114: Acta de Reunión del Sprint N° 1



Fuente: Elaboración propia

3.3. Desarrollo del Sprint 2

Tabla 45: Sprint 2 – Módulo Facturación

SPRINT	CODIGO	TAREA	ESTIMACIÓN (DIAS)	PRIORIDAD
SPRINT 2	RF08	El sistema permitirá filtrar los comprobantes de pago de acuerdo a la fecha de emisión, cliente, tipo de comprobante, estado, serie y numero de comprobante.	2	ALTA
	RF12	El sistema permitirá enviar un XML y un reporte del comprobante de pago electrónico al Cliente, para su posterior confirmación.	2	ALTA
	RF15	El sistema permitirá filtrar los cierres diarios generados en el día, mediante un filtro de fecha de emisión y estado del comprobante.	2	ALTA
	RF19	El sistema permitirá listar los comprobantes de pago electrónicos relacionados a la factura que hayan sido anulados anteriormente, mediante un filtro de fecha de emisión y estado.	2	ALTA
	RF22	El sistema permitirá listar un historial de los comprobantes de pago electrónicos generados en el BackOffice.	2	ALTA
	RF23	El sistema permitirá obtener información de un comprobante de pago electrónico generado en el BackOffice.	2	ALTA
	RF24	El sistema permitirá listar un historial de los comprobantes de pago electrónicos enviados a la SUNAT.	2	ALTA
	RF25	El sistema permitirá obtener información sobre el estado de un comprobante de pago electrónico enviado a la SUNAT.	2	ALTA
	RF28	El sistema permitirá descargar los XML generados de los comprobantes de pago electrónicos.	2	ALTA
	RF29	El sistema permitirá exportar la información a un archivo de tipo Excel, que contengan datos de los comprobantes de pago electrónicos de acuerdo al filtro propuesto.	2	ALTA
	RF30	El sistema permitirá descargar de manera masiva los XML y los reportes de cada uno de los comprobantes de pago electrónico emitidos.	2	ALTA
	RF38	Solo el usuario no registrado podrá consultarla emisión de los comprobantes electrónicos si cumple los filtros de número de serie, numero de documento, RUC del Cliente, fecha de emisión y monto total del comprobante.	2	ALTA

Fuente: Elaboración propia

RF08: El sistema permitirá filtrar los comprobantes de pago de acuerdo a la fecha de emisión, cliente, tipo de comprobante, estado, serie y numero de comprobante.

Figura 115: Vista Modulo Comprobantes de Pago - Filtrar

Fuente: Elaboración propia

Tipo	Serie	Número	Fecha EM.	RazonSocial	Mon	Neto	IGV	Otros	Dcto	Total	Estado Sunat
01	F001	0000316	01/02/2019	SOCIETE AIR FRANCE SUCURSA...	USD	8.58	1.54	0.00	0.00	10.12	ACEPTADO
01	F001	0000317	01/02/2019	MEDIX TRAVEL AID S.A.C.	USD	6.27	1.13	0.00	0.00	7.40	ACEPTADO
03	B001	0000670	01/02/2019	REYES RUBEN	USD	52.11	9.38	0.00	0.00	61.49	ACEPTADO
03	B001	0000671	01/02/2019	URATA FUTOSHI	USD	83.47	15.02	0.00	0.00	98.49	ACEPTADO
36	001	0027954	01/02/2019	REYES RUBEN	USD	856.00	154.08	78.43	0.00	1,088.51	ACEPTADO
36	001	0027955	01/02/2019	URATA FUTOSHI	USD	1,206.00	217.08	78.43	0.00	1,501.51	ACEPTADO
36	001	0027956	01/02/2019	TABUCHI GUSHIKEN CINDY RURI	USD	37.00	0.00	0.00	0.00	37.00	ACEPTADO
01	F001	0000318	04/02/2019	CENTRO DE AGENCIAS CORPOR...	USD	219.79	39.56	0.00	0.00	259.35	ACEPTADO
01	F001	0000319	04/02/2019	REAL TOURS TOKYO	USD	60.97	10.97	0.00	0.00	71.94	ACEPTADO
03	B001	0000672	04/02/2019	ZUIKO ARTURO	USD	31.68	5.70	0.00	0.00	37.38	ACEPTADO
03	B001	0000673	04/02/2019	GALVEZ CINTHYA	USD	31.68	5.70	0.00	0.00	37.38	ACEPTADO
03	B001	0000674	04/02/2019	ASATO BERTA	USD	35.01	6.30	0.00	0.00	41.31	ACEPTADO
03	B001	0000675	04/02/2019	NOMOTO KIYOKO	USD	60.43	10.88	0.00	0.00	71.31	ACEPTADO
03	B001	0000676	04/02/2019	ESPIÑOZA VENNNIA	USD	164.61	29.63	0.00	0.00	194.24	ACEPTADO
03	B001	0000677	04/02/2019	JAVE ARROYO YASUKE SAKAY	USD	82.31	14.81	0.00	0.00	97.12	ACEPTADO
36	001	0027957	04/02/2019	REAL TOURS S.A.	USD	417.00	75.06	145.16	0.00	637.22	ACEPTADO

Figura 116: Estructura de Código – Vista Modulo Comprobantes de Pago – Filtrar Parte 1

Fuente: Elaboración propia

```

using JSUBInet.WebUIInet.Models;
@model DocumentoListaViewModel
@{
    ViewBag.Title = "Documentos";
    var fecini = DateTime.Now.AddDays(-7).Date;
    var fecfin = DateTime.Now.Date;
}
@{Html.CI().CollectionViewService<DocumentoListaViewModel>()
    .Id("collectionViewService")
    .PageSize(Model.PageSize)}
<div class="container-fluid">
    <div class="row">
        <div class="col-md-4">
            @Html.CI().Pager().Owner("collectionViewService").Width(300)
        </div>
        <div class="col-md-4">
            <label>Inicio :</label>
            @Html.CI().InputDate().Id("FechaInicio").Width(140).Value(fecini)
            <label>Final :</label>
            @Html.CI().InputDate().Id("FechaFinal").Width(140).Value(fecfin)
        </div>
        <div class="col-md-4">
            <label>Empresa :</label>
            @Html.CI().AutoComplete()
                .Width(175)
                .Bind(Model.empresas)
                .Id("cboEmpresa")
                .MaxItems(10)
                .CssMatch("highlight")
                .DisplayMemberPath("name")
                .SelectedValuePath("value")
                .Disabled(!User.IsInRole("ADMINISTRADOR"))
            <button id="btnExcel" class="btn btn-default"></button>
            <button id="btnXML" class="btn btn-default"></button>
        </div>
    </div>
    <div class="row" style="padding-bottom:20px">
        <div class="col-md-4"></div>
        <div class="col-md-4"></div>
        <div class="col-md-4"></div>
    </div>
</div>
<script id="EditTemplate"></script>
<script id="PrintTemplate"></script>
<script id="RequestXml"></script>
<script id="ResponseXml"></script>
<script id="Email"></script>
<script id="EditTemplate"></script>
<script id="DownloadTemplate"></script>
<script id="DeleteTemplate"></script>
<script></script>
<script></script>
<script id="detailTemplate"></script>

```

Figura 117: Estructura de Pago – Vista Modulo Comprobantes de Pago – Filtrar Parte 2

Fuente: Elaboración propia

```

@Html.C1().FlexGrid(DocumentoListaViewModels)
    .Id("pagingGrid")
    .ShowDetailRow(d => d.DetailRowTemplateId("detailTemplate"), RowHasDetail("hasDetail"), DetailVisibilityMode(C1.Web.Mvc.Grid.DetailVisibilityMode.ExpandMulti))
    .Height(440)
    .ItemsSourceId("collectionViewService")
    .IsReadOnly(true)
    .AllowDelete(true)
    .AutoGenerateColumns(false)
    .Columns(clsb =>
    {
        clsb.Add().Name("ApprovalStatus").Width("35");
        clsb.Add(cb => cb.Binding("TipoDocumento").Header("Tipo").Width("40"));
        clsb.Add(cb => cb.Binding("NumeroSerie").Header("Serie").Width("40").Align("right"));
        clsb.Add(cb => cb.Binding("NumeroDocumento").Header("Número").Width("65").Align("right"));
        clsb.Add(cb => cb.Binding("FechaEmision").Header("Fecha Em.").Width("70").Format("dd/MM/yyyy"));
        clsb.Add(cb => cb.Binding("RazonSocial").Header("RazonSocial").Width("210"));
        clsb.Add(cb => cb.Binding("Moneda").Header("Mon").Width("40"));
        clsb.Add(cb => cb.Binding("NetoDocumento").Header("Neto").Format("n2").Width("80"));
        clsb.Add(cb => cb.Binding("ImpuestoDocumento").Header("IGV").Format("n2").Width("70"));
        clsb.Add(cb => cb.Binding("OtrosImpuestosDocumento").Header("Otros").Format("n2").Width("60"));
        clsb.Add(cb => cb.Binding("DescuentoDocumento").Header("Dscto").Format("n2").Width("60"));
        clsb.Add(cb => cb.Binding("TotalDocumento").Header("Total").Format("n2").Width("90"));
        clsb.Add(cb => cb.Binding("DescripcionEstadosunat").Header("Estado Sunat").Width("90").Align("center"));
        clsb.Add().Name("EditButton").CellTemplate(b => b.TemplateId("EditTemplate")).Width("25").Visible(Model.EsGlobalUser);
        clsb.Add().Name("PrintButton").CellTemplate(b => b.TemplateId("PrintTemplate")).Width("25");
        clsb.Add().Name("XmlRequestButton").CellTemplate(b => b.TemplateId("RequestXml")).Width("25").Visible(Model.EsGlobalUser);
        clsb.Add().Name("XmlResponseButton").CellTemplate(b => b.TemplateId("ResponseXml")).Width("25").Visible(Model.EsGlobalUser);
        clsb.Add().Name("EmailButton").CellTemplate(b => b.TemplateId("Email")).Width("25").Visible(Model.EsGlobalUser);
        clsb.Add().Name("DeleteButton").CellTemplate(b => b.TemplateId("DeleteTemplate")).Width("25").Visible(Model.EsGlobalUserDelete);
    })
    .ItemFormatter("pagingGridFormatter")
}

<div class="well grid-sort-group">
    @Html.C1().Pager().Owner("pagingGrid")
</div>
<div id="modal-dialog" class="modal fade in">
    <div class="modal-dialog" style="width: 50%;">
        <div class="modal-content">
            <div id="modal-content"></div>
        </div>
    </div>
</div>

```

Figura 118: Estructura de Código – Vista Modulo Comprobantes de Pago – Filtrar Pate 3

Fuente: Elaboración propia

```

@section Scripts {
    @Scripts.Render("~/bundles/jqueryval")
    <script type="text/javascript">
        var pagingGrid, oCollection, cboTipoDocumento, cboEmpresa, cboCliente, fechainicio, fechafinal, txtnumerodoc, txtseriedoc, cboEstadoSunat;
        cl.mvc.Utils.documentReady(function () {
            pagingGrid = wijmo.Control.getControl("#pagingGrid");
            cboCliente = wijmo.Control.getControl("#cboCliente");
            cboTipoDocumento = wijmo.Control.getControl("#cboTipoDocumento");
            cboEmpresa = wijmo.Control.getControl("#cboEmpresa");
            cboEstadoSunat = wijmo.Control.getControl("#cboEstadoSunat");
            fechainicio = wijmo.Control.getControl("#FechaInicio");
            fechafinal = wijmo.Control.getControl("#FechaFinal");
            txtnumerodoc = $("#NumeroDocumento");
            txtseriedoc = $("#SerieDocumento");
            if (pagingGrid) {
                oCollection = pagingGrid.collectionView;
                pagingGrid.rows.defaultSize = 25;
                updateUrl(@Model.PageInitial);
                oCollection.refresh();
            };
            var dvFilter = document.getElementById("btnFilter");
            dvFilter.addEventListener("click", function () {
                updateUrl(0);
                oCollection.refresh();
            });
            oCollection.currentChanging.addHandler(function () {
                updateUrl(0);
            });
            var dvExcel = document.getElementById("btnExcel");
            dvExcel.addEventListener("click", function () {
                if (!pagingGrid || !oCollection) {
                    return;
                }
                window.location.href = @Url.Action("DownloadToExcel", "Documentos")?FechaInicio=
                    + encodeURIComponent(formatDate(fechainicio.value)) + "&FechaFinal="
                    + encodeURIComponent(formatDate(fechafinal.value)) + "&IdCliente="
                    + encodeURIComponent(cboCliente.selectedValue) + "&IdEmpresa="
                    + encodeURIComponent(cboEmpresa.selectedValue) + "&IdTipoDocumento="
                    + encodeURIComponent(cboTipoDocumento.selectedValue) + "&IdEstadoSunat="
                    + encodeURIComponent(cboEstadoSunat.selectedValue);
            });
        });
    </script>
}

```

Figura 119: Estructura de Código – Controlador Modulo Cmpte. de Pago – Filtrar Parte 1

Fuente: Elaboración propia

```
[Authorize(Roles = "Documento Listar")]
public ActionResult Index()
{
    DataRepository.UsuarioProvider.DeepLoad(oUsuario, true, DeepLoadType.IncludeChildren, new Type[] { typeof(TList<ClienteLogin>), typeof(Cliente), typeof(Empresa) });

    DocumentoListaViewModel oClienteDetalleViewModel = new DocumentoListaViewModel()
    {
        IdEmpresa = -1,
        IdCliente = -1,
        empresas = new List<AutoCompleteElemento<long>>(),
        clientes = (from ClienteLogin u in oUsuario.ClienteLoginCollectionGetByIdUsuario
            select new AutoCompleteElemento<long>()
            {
                name = u.IdClienteSource.RazonSocial,
                value = u.IdCliente
            }).ToList(),
        TipoDocumentos = (from TipoDocumento oTipo in DataRepository.TipoDocumentoProvider.GetAll().Where(x => x.IdEstado == 1)
            select new AutoCompleteElemento<short>()
            {
                value = oTipo.IdTipoDocumento,
                name = oTipo.DescripcionTipoDocumento
            }).ToList(),
        EstadosSunat = (from EstadoSunat oSunat in DataRepository.EstadoSunatProvider.GetAll()
            select new AutoCompleteElemento<short>()
            {
                value = oSunat.IdEstadoSunat,
                name = oSunat.DescripcionEstadoEnvioSunat
            }).ToList(),
    };

    oClienteDetalleViewModel.TipoDocumentos.Insert(0, new AutoCompleteElemento<short>() { name = "TODOS", value = -1 });
    oClienteDetalleViewModel.EstadosSunat.Insert(0, new AutoCompleteElemento<short>() { name = "TODOS", value = -1 });

    oClienteDetalleViewModel.empresas.Add(new AutoCompleteElemento<long>() { name = oUsuario.IdEmpresaSource.RazonSocial, value = oUsuario.IdEmpresa });
    if (User.IsInRole("Documento Administrar"))
        oClienteDetalleViewModel.empresas.Insert(0, new AutoCompleteElemento<long>() { name = "TODOS", value = -1 });

    if (User.IsInRole("Documento Administrar"))
        oClienteDetalleViewModel.clientes.Insert(0, new AutoCompleteElemento<long>() { name = "TODOS", value = -1 });

    if (User.IsInRole("Documento Administrar"))
        oClienteDetalleViewModel.EsGlobalUser = true;
    else
        oClienteDetalleViewModel.EsGlobalUser = false;

    if (User.IsInRole("Documento Anular"))
        oClienteDetalleViewModel.EsGlobalUserDelete = true;

    oClienteDetalleViewModel.PageSize = 16;
    return View(oClienteDetalleViewModel);
}
```

Figura 120: Estructura de Código – Controlador Modulo Cmpte. de Pago – Filtrar Parte 2

Fuente: Elaboración propia

```
public ActionResult RemoteBind_Read([FromBody] Request request) collectionViewRequest(DocumentoListaViewModel) requestData, DateTime FechaInicio, DateTime FechaFinal, long IdCliente, long IdEmpresa, short IdTipoDocumento,
{
    int nCount = 0;
    requestData.PageIndex = (PageInitial == 0 ? requestData.PageIndex : PageInitial);

    object[] cparams;
    if (ConfigurationManager.AppSettings["Region"].ToString().Equals("US"))
    {
        object[] cparamsTipo = { FechaInicio.ToString("yyyy/MM/dd"), FechaFinal.ToString("yyyy/MM/dd"), IdCliente, IdEmpresa, IdTipoDocumento, NumeroSerie, NumeroDocumento, IdEstadoSunat };
        cparams = cparamsTipo;
    }
    else
    {
        object[] cparamsTipo = { FechaInicio.ToString("dd/MM/yyyy"), FechaFinal.ToString("dd/MM/yyyy"), IdCliente, IdEmpresa, IdTipoDocumento, NumeroSerie, NumeroDocumento, IdEstadoSunat };
        cparams = cparamsTipo;
    }

    string cwhere = string.Format("[FechaEmision] BETWEEN '{0}' AND '{1}' AND ( '{2}' = -1 OR IdCliente = '{2}') AND ( '{3}' = -1 OR IdEmpresa = '{3}') AND ((4) = -1 OR IdTipoDocumento = {4}) AND NumeroSerie LI
TList<Documento> oDocumentos = DataRepository.DocumentoProvider.GetPaged(cwhere, "FechaEmision, TipoDocumento, NumeroSerie, NumeroDocumento", requestData.PageIndex, requestData.PageSize, out nCount);

    if (oDocumentos.Count > 0)
    {
        // en dos fases para impedir el Over Memory
        DataRepository.DocumentoProvider.DeepLoad(oDocumentos, true, DeepLoadType.IncludeChildren, new Type[] { typeof(EstadoSunat) });
        DataRepository.DocumentoProvider.DeepLoad(oDocumentos, true, DeepLoadType.IncludeChildren, new Type[] { typeof(TList<DocumentoDetalle>), typeof(Producto) });
    }

    List<DocumentoListaViewModel> oDocumentoModelViews = Mapper.Map<List<Documento>, List<DocumentoListaViewModel>>(oDocumentos.ToList());
    CollectionViewResponse<DocumentoListaViewModel> responseData = CollectionViewHelper.Read(requestData, oDocumentoModelViews);
    responseData.PageIndex = requestData.PageIndex;
    responseData.TotalItemCount = nCount;
    return this.C13son(responseData);
}
```

RF12: El sistema permitirá enviar un XML y un reporte del comprobante de pago electrónico al Cliente, para su posterior confirmación.

Figura 121: Vista Modulo Comprobantes de Pago – Operación Enviar Correo Electrónico

Fuente: Elaboración propia

	Tipo	Serie	Número	Fecha EM.	RazonSocial	Mon	Neto	IGV	Otros	Dcto	Total	Estado Sanat					
+	✓	01	F001	0000316	01/02/2019	SOCIETE AIR FRANCE SUCURSA...	USD	8.58	1.54	0.00	0.00	10.12	ACEPTADO	🔍	🔄	🗑️	📧
+	✓	01	F001	0000317	01/02/2019	MEDK TRAVEL AID S.A.C.	USD	6.27	1.13	0.00	0.00	7.40	ACEPTADO	🔍	🔄	🗑️	📧
+	✓	03	B001	0000670	01/02/2019	REYES RUBEN	USD	52.11	9.38	0.00	0.00	61.49	ACEPTADO	🔍	🔄	🗑️	📧
+	✓	03	B001	0000671	01/02/2019	URATA FUTOSHI	USD	83.47	15.02	0.00	0.00	98.49	ACEPTADO	🔍	🔄	🗑️	📧
+	✓	36	001	0027954	01/02/2019	REYES RUBEN	USD	856.00	154.00	78.43	0.00	1 088.51	ACEPTADO	🔍	🔄	🗑️	📧

Figura 122: Estructura de Código – Vista Modulo C.P. – Enviar Correo E.

Fuente: Elaboración propia

```
function SendEmail() {
  if (!pagingGrid || !oCollection) {
    return;
  }
  pagingGrid.invalidate();
  $.ajax({
    cache: false,
    url: Url.Action("SendEmail", "Documentos"),
    async: false,
    data: { id: oCollection.currentItem.IdDocumento },
    success: function (data) {
      if (data == "OK") {
        alert("Envio Satisfactorio.");
      } else {
        alert(data);
      }
    },
    error: function (req, status, errorObj) {
      alert(req.responseText);
    }
  });
};
```

Figura 123: Estructura de Código – Controlador Modulo C.P. – Enviar Correo E: Parte 1

Fuente: Elaboración propia

```
#region Correo
- referencias
public ActionResult SendEmail(long id)
{
  try
  {
    Utils.SendMail.SendEmailDocumento(id);
    return Json("OK", JsonRequestBehavior.AllowGet);
  }
  catch (Exception ex)
  {
    return Json(ex.Message, JsonRequestBehavior.AllowGet);
  }
}
#endregion
```

Figura 124: Estructura de Código – Controlador Modulo C.P. – Enviar Correo E. Parte 2

Fuente: Elaboración propia

```
public static void SendEmailDocumento(long id)
{
    lock (sync)
    {
        DataSet oDataSet = null; string rPath = string.Empty;
        ReportDataSource oReportDataSource = null; List<ReportParameter> oReportParameters = new List<ReportParameter>();

        Documento oDocumento = DataRepository.DocumentoProvider.GetByIdDocumento(id);
        DataRepository.DocumentoProvider.DeepLoad(oDocumento, true, DeepLoadType.IncludeChildren, new Type[] { (typeof(Empresa)), typeof(TipoDocumentoIdentidad),
        DataRepository.ClienteLoginProvider.DeepLoad(oDocumento.IdClienteSource.ClienteLoginCollection, true, DeepLoadType.IncludeChildren, new Type[] { (typeof(U

generamos el pdf y xml)

//Envia Datos
List<string> ccUsuarios = new List<string>();
if (!string.IsNullOrEmpty(oDocumento.DocumentoDetalleCollection.FirstOrDefault().Codigo35))
    if (oDocumento.DocumentoDetalleCollection.FirstOrDefault().Codigo35.Contains("@"))
        ccUsuarios.Add(oDocumento.DocumentoDetalleCollection.FirstOrDefault().Codigo35.ToString());
Utils.SendMail.SendEmailMessage(cAsunto, htmlBody.ToString(), oUsuariosEmail.ToArray(), ccUsuarios.ToArray(), pdfPath, xmlPath, id);
    }
}
```

Figura 125: Estructura de Código – Controlador Modulo C.P. – Enviar Correo E. Parte 3

Fuente: Elaboración propia

```
public static Boolean SendEmailMessage(string cAsunto, string cCuerpo, string[] aDestinatarios, string[] aCopiados, string cArchivoPdf, string cArchivoXml, long id)
{
    string NumeroDocumento = string.Empty;
    Documento oDocumento = DataRepository.DocumentoProvider.GetByIdDocumento(id);
    Empresa oEmpresa = DataRepository.EmpresaProvider.GetAll().SingleOrDefault();
    NumeroDocumento = oEmpresa.RucEmpresa.Trim() + "-" + oDocumento.TipoDocumento.Trim() + "-" + oDocumento.NumeroSerie.Trim() + "-" + oDocumento.NumeroDocumento.Trim();

    string HostAdd = ConfigurationManager.AppSettings["Host"].ToString();
    string FromEmailId = ConfigurationManager.AppSettings["FromMail"].ToString();
    string Password = ConfigurationManager.AppSettings["Password"].ToString();

    generamos el contenido

    //network and security related credentials
    System.Net.ServicePointManager.ServerCertificateValidationCallback = new System.Net.Security.RemoteCertificateValidationCallback(validarCertificado);

    try
    {
        SmtplibClient smtp = new SmtplibClient();
        smtp.Host = HostAdd;
        smtp.Port = 25;
        smtp.EnableSsl = false;
        //smtp.EnableSsl = true;
        smtp.UseDefaultCredentials = true;
        smtp.Credentials = new NetworkCredential(mailMessage.From.Address, Password);
        smtp.Send(mailMessage); //sending Email
    }
    catch (Exception ex)
    {
        return false;
    }

    return true;
}
```


Figura 128: Estructura de Código – Vista Modulo Cierre Comprobantes – Filtrar Parte 2

Fuente: Elaboración propia

```

@section Scripts {
    @Scripts.Render("~/bundles/jqueryval")
    <script type="text/javascript">
        var pagingGrid, oCollection, pagingGridDetail, oCollectionDetail, fechaInicial, fechaFinal, IdEstadoSunat, fechaCierreResumen;
        cl.mvc.Utils.documentReady(function () {
            pagingGrid = wijmo.Control.getControl("#pagingGrid");
            fechaInicial = wijmo.Control.getControl("#FechaInicial");
            fechaFinal = wijmo.Control.getControl("#FechaFinal");
            fechaCierreResumen = wijmo.Control.getControl("#FechaCierreResumen");
            IdEstadoSunat = $("#cboEstadoSunat");
            if (pagingGrid) {
                oCollection = pagingGrid.collectionView;
                pagingGrid.rows.defaultSize = 25;
                updateUrl(@Model.PageInitial);
                oCollection.refresh();
            };
            var dvFilter = document.getElementById('btnFilter');
            dvFilter.addEventListener("click", function () {
                updateUrl(0);
                oCollection.refresh();
            });
            oCollection.currentChanging.addHandler(function () {
                updateUrl(0);
            });
        });

        function updateUrl(PageInitial) {
            oCollection._readActionUrl = "@Url.Action("RemoteBind_Read", "DocumentoCierres")?fechaInicial="
                + encodeURIComponent(formatDate(fechaInicial.value)) + "&fechaFinal="
                + encodeURIComponent(formatDate(fechaFinal.value)) + "&cboEstadoSunat="
                + encodeURIComponent(IdEstadoSunat.val()) + "&PageInitial="
                + encodeURIComponent(PageInitial);
        };
    </script>
}
    
```

Figura 129: Estructura de Código – Controlador Modulo Cierre Comprobantes – Filtrar

Fuente: Elaboración propia

```

#region Principal
[Authorize(Roles = "Cierre Documentos Listar")]
public ActionResult Index(int PageInitial = 0)
{
    DocumentoCierreListaModel oCierreDocumento = new DocumentoCierreListaModel();
    oCierreDocumento.Estadosunat = new SelectListItems() {
        new SelectListItem { Text = "Pendiente", Value = "1", Selected = true },
        new SelectListItem { Text = "Aceptado", Value = "2"},
        new SelectListItem { Text = "Rechazado", Value = "3"},
    };
    oCierreDocumento.PageSize = 16;
    oCierreDocumento.PageInitial = PageInitial;
    return View(oCierreDocumento);
}
#endregion

#region Filtro
public ActionResult RemoteBind_Read([ClisonRequest] CollectionViewRequest<DocumentoCierreListaModel> requestData, DateTime fechaInicial, DateTime fechaFinal, short cboEstadoSunat, int PageInitial = 0)
{
    int nCount = 0;
    requestData.PageIndex = (PageInitial == 0 ? requestData.PageIndex : PageInitial);

    object[] cparams;
    if (ConfigurationManager.AppSettings["Region"].ToString().Equals("US"))
    {
        object[] cparamsTipo = { fechaInicial.ToString("yyyy/MM/dd"), fechaFinal.ToString("yyyy/MM/dd"), cboEstadoSunat };
        cparams = cparamsTipo;
    }
    else
    {
        object[] cparamsTipo = { fechaInicial.ToString("dd/MM/yyyy"), fechaFinal.ToString("dd/MM/yyyy"), cboEstadoSunat };
        cparams = cparamsTipo;
    }

    string chere = string.Format("[FechaCierre] BETWEEN '{0}' AND '{1}' AND IdEstadoSunat='{2}'", cparams);
    IList<DocumentoCierre> oCierreDocumento = DataRepository.DocumentoCierreProvider.GetPaged(chere, "", requestData.PageIndex, requestData.PageSize, out nCount);
    if (oCierreDocumento.Count > 0)
        DataRepository.DocumentoCierreProvider.DeepLoad(oCierreDocumento, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Estado), typeof(Usuario) });
    List<DocumentoCierre> oCierreModels = Mapper.Map<IList<DocumentoCierre>, List<DocumentoCierreListaModel>>(oCierreDocumento.ToList());
    CollectionViewResponse<DocumentoCierreListaModel> responseData = CollectionViewHelper.Read(requestData, oCierreModels);
    responseData.PageIndex = requestData.PageIndex;
    responseData.TotalItemCount = nCount;
    return this.Clison(responseData);
}
#endregion
    
```

RF19: El sistema permitirá listar los comprobantes de pago electrónicos relacionados a la factura que hayan sido anulados anteriormente, mediante un filtro de fecha de emisión y estado.

Figura 130: Vista Modulo Anulación Comprobante – Operación Filtrar

Fuente: Elaboración propia

N° ticket	Descripción Anulación	Fecha Anulación	Usuario Creacion	Fecha Creacion	Usuario Modificacion	Fecha Modificacion	Estado		
20180309390...	ANULACION DE DOCUMENTO	22/12/2018	Admin CTravel	22/12/2018	Admin CTravel	22/12/2018	ACTIVO		
20190318521...	ANULACION DE DOCUMENTO	10/01/2019	Admin CTravel	10/01/2019	Admin CTravel	10/01/2019	ACTIVO		

Figura 131: Estructura de Código – Vista Modulo Anulación Comprobante – Filtrar Parte 1

Fuente: Elaboración propia

```

using JSOBInet.WebUI.Net.Models;
@model DocumentoAnulacionLista@Model
@{
    ViewBag.Title = "Anulacion Documento";
    var fecini = DateTime.Now.AddDays(-15).Date;
    var fecfin = DateTime.Now.Date;
}
@Html.Ci().CollectionViewService<DocumentoAnulacionListaModel>()
    .Id("collectionViewService")
    .PageSize(Model.PageSize)
}
<div class="well grid-sort-group">
    @Html.Ci().Pager().Owner("collectionViewService")
    <label>Inicio :</label>
    @Html.Ci().InputDate().Id("FechaInicial").Width(120).Value(fecini)
    <label>Final :</label>
    @Html.Ci().InputDate().Id("FechaFinal").Width(120).Value(fecfin)
    @Html.DropDownList("cboEstadoSunat", Model.Estadosunat, htmlAttributes: new { @class = "form-control", @style = "width:120px;height:30px" })
    <button id="btnFiltro"
        class="btn btn-default" title="Filtrar">
        <span class="glyphicon wj-glyph-filter"></span>
    </button>
</div>
<script id="PrintTemplate">...</script>
<script id="RequestXml">...</script>
<script id="ResponseXml">...</script>
<script id="RecuperarResumen">...</script>
<script>...</script>
<script id="detailTemplate">...</script>
@Html.Ci().FlexGrid<DocumentoAnulacionListaModel>()
    .Id("pagingGrid")
    .ShowDetailRow(d => d.DetailRowTemplateId("detailTemplate").RowHasDetail("hasDetail").DetailVisibilityMode(Ci.Web.Mvc.Grid.DetailVisibilityMode.ExpandMulti))
    .Height(440)
    .ItemsSourceId("collectionViewService")
    .IsReadOnly(true)
    .AllowDelete(true)
    .AutoGenerateColumns(false)
    .Columns(cls =>
    {
        clsb.Add(cb => cb.Binding("NumeroTicket").Header("N° ticket").Width("100"));
        clsb.Add(cb => cb.Binding("DescripcionAnulacion").Header("Descripción Anulación").Width("200"));
        clsb.Add(cb => cb.Binding("FechaAnulacion").Header("Fecha Anulación").Width("105").Format("dd/MM/yyyy"));
        clsb.Add(cb => cb.Binding("UsuarioCreacion").Header("Usuario Creación").Width("160"));
        clsb.Add(cb => cb.Binding("FechaCreacion").Header("Fecha Creación").Format("dd/MM/yyyy"));
        clsb.Add(cb => cb.Binding("UsuarioModificacion").Header("Usuario Modificación").Width("130"));
        clsb.Add(cb => cb.Binding("FechaModificacion").Header("Fecha Modificación").Format("dd/MM/yyyy").Width("125"));
        clsb.Add(cb => cb.Binding("DescripcionEstado").Header("Estado").Width("70"));
        clsb.Add().Name("XmlRequestButton").CellTemplate(b => b.TemplateId("RequestXml")).Width("25");
        clsb.Add().Name("XmlResponseButton").CellTemplate(b => b.TemplateId("ResponseXml")).Width("25");
        clsb.Add().Name("Recuperar").CellTemplate(b => b.TemplateId("RecuperarResumen")).Width("25");
    })
}
<div class="well grid-sort-group">
    @Html.Ci().Pager().Owner("pagingGrid")
</div>

```

Figura 132: Estructura de Código – Vista Modulo Anulación Comprobante – Filtrar Parte 2

Fuente: Elaboración propia

```

@section Scripts {
    @Scripts.Render("~/bundles/jqueryval")
    <script type="text/javascript">
        var pagingGrid, oCollection, fechaInicial, fechaFinal, IdEstadoSunat, fechaCierreResumen;
        cl.mvc.Utils.documentReady(function () {
            pagingGrid = wijmo.Control.getControl("#pagingGrid");
            fechaInicial = wijmo.Control.getControl("#FechaInicial");
            fechaFinal = wijmo.Control.getControl("#FechaFinal");
            fechaCierreResumen = wijmo.Control.getControl("#FechaCierreResumen");
            IdEstadoSunat = $("#cboEstadoSunat");
            if (pagingGrid) {
                oCollection = pagingGrid.collectionView;
                pagingGrid.rows.defaultSize = 25;
                updateUrl(PageInitial);
                oCollection.refresh();
            };
            var dvFilter = document.getElementById("btnFilter");
            dvFilter.addEventListener("click", function () {
                updateUrl(0);
                oCollection.refresh();
            });
            oCollection.currentChanging.addHandler(function () {
                updateUrl(0);
            });
        });
        function updateUrl(PageInitial) {
            oCollection.readActionUrl = @Url.Action("RemoteBind_Read", "DocumentoAnulaciones"?fechaInicial="
                + encodeURIComponent(formatDate(fechaInicial.value)) + "&fechaFinal="
                + encodeURIComponent(formatDate(fechaFinal.value)) + "&cboEstadoSunat="
                + encodeURIComponent(IdEstadoSunat.val()) + "&PageInitial="
                + encodeURIComponent(PageInitial);
        };
    </script>
    
```

Figura 133: Estructura de Pago – Controlador Modulo Anulación Comprobante – Filtrar

Fuente: Elaboración propia

```

#region Principal
[Authorize(Roles = "Anulacion Documentos Listar")]
public ActionResult Index(int PageInitial = 0)
{
    DocumentoAnulacionListaModel oAnulacionDocumento = new DocumentoAnulacionListaModel();
    oAnulacionDocumento.EstadoSunat = new List<SelectListItem>()
    {
        new SelectListItem { Text = "Pendiente", Value = "1", Selected = true },
        new SelectListItem { Text = "Aceptado", Value = "2"},
        new SelectListItem { Text = "Rechazado", Value = "3"},
    };
    oAnulacionDocumento.PageSize = 16;
    oAnulacionDocumento.PageInitial = PageInitial;
    return View(oAnulacionDocumento);
}

public ActionResult RemoteBind_Read([CancellationToken] CancellationToken requestData, DateTime fechaInicial, DateTime fechaFinal, short cboEstadoSunat, int PageInitial = 0)
{
    int nCount = 0;
    requestData.PageIndex = (PageInitial == 0 ? requestData.PageIndex : PageInitial);

    object[] cparams;
    if (ConfigurationManager.AppSettings["Region"].ToString().Equals("US"))
    {
        object[] cparamsTipo = { fechaInicial.ToString("yyyy/MM/dd"), fechaFinal.ToString("yyyy/MM/dd"), cboEstadoSunat };
        cparams = cparamsTipo;
    }
    else
    {
        object[] cparamsTipo = { fechaInicial.ToString("dd/MM/yyyy"), fechaFinal.ToString("dd/MM/yyyy"), cboEstadoSunat };
        cparams = cparamsTipo;
    }

    string cwhere = string.Format("[FechaCreacion] BETWEEN '{0}' AND '{1}' AND IdEstadoSunat='{2}'", cparams);
    IList<DocumentoAnulacion> oAnulacionDocumento = DataRepository.DocumentoAnulacionProvider.GetPage(cwhere, "", requestData.PageSize, out nCount);
    if (oAnulacionDocumento.Count > 0)
    {
        DataRepository.DocumentoAnulacionProvider.DeepLoad(oAnulacionDocumento, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Estado), typeof(Usuario) });
        DataRepository.DocumentoAnulacionProvider.DeepLoad(oAnulacionDocumento, true, DeepLoadType.IncludeChildren, new Type[] { typeof(TList<Documento>) });
    }

    List<DocumentoAnulacionListaModel> oAnulacionModels = Mapper.Map<IList<DocumentoAnulacion>, List<DocumentoAnulacionListaModel>>(oAnulacionDocumento.ToList());
    CollectionViewResponse<DocumentoAnulacionListaModel> responseData = CollectionViewHelper.Read(requestData, oAnulacionModels);
    responseData.PageIndex = requestData.PageIndex;
    responseData.TotalItemCount = nCount;
    return this.Cllson(responseData);
}
#endregion
    
```

RF22: El sistema permitirá listar un historial de los comprobantes de pago electrónicos generados en el BackOffice.

Figura 134: Vista Modulo Sincronizador Origen – Operación Filtrar

Fuente: Elaboración propia

Tabla Origen	Server Origen	N°Registro	N° Operacion	Usuario Creacion	Fecha Creacion	Usuario Modificacion	Fecha Modificacion	Estado
+ DOCUMENTOS	BACKOFFICE	55526	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+ DOCUMENTOS	BACKOFFICE	55527	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+ DOCUMENTOS	BACKOFFICE	55528	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+ DOCUMENTOS	BACKOFFICE	55529	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+ DOCUMENTOS	BACKOFFICE	55530	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+ DOCUMENTOS	BACKOFFICE	55531	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+ DOCUMENTOS	BACKOFFICE	55532	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+ DOCUMENTOS	BACKOFFICE	55533	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+ DOCUMENTOS	BACKOFFICE	55534	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+ DOCUMENTOS	BACKOFFICE	55535	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+ DOCUMENTOS	BACKOFFICE	55536	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+ DOCUMENTOS	BACKOFFICE	55537	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+ DOCUMENTOS	BACKOFFICE	55538	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+ DOCUMENTOS	BACKOFFICE	55539	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+ DOCUMENTOS	BACKOFFICE	55540	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+ DOCUMENTOS	BACKOFFICE	55541	I	Admin CTravel	23/03/2019	Admin CTravel	23/03/2019	ACTIVO

Figura 135: Estructura de Código – Vista Modulo Sincronizador Origen – Filtrar

Fuente: Elaboración propia

```

using JSUBInet.Web03Inet.Models;
model SincronizadorOrigenListaViewModel
{
    ViewBag.Title = "Sincronizador Origen";
    var fecini = DateTime.Now.AddDays(-15).Date;
    var fecfin = DateTime.Now.Date;
}
@(Html.C1().CollectionViewService<SincronizadorOrigenListaViewModel>()
    .Id("collectionViewService")
    .PageSize(Model.PageSize)
)
<div class="well grid-sort-group">
    @(Html.C1().Pager().Owner("collectionViewService"))
    <label>Inicio :</label>
    @(Html.C1().InputDate().Id("FechaInicial").Width(120).Value(fecini))
    <label>Final :</label>
    @(Html.C1().InputDate().Id("FechaFinal").Width(120).Value(fecfin))
    @(Html.DropDownList("cbotEstado", Model.EstadosOrigen, htmlAttributes: new { @class = "form-control", @style = "width:120px;height:30px" }));
    <button id="btnFilter"
        class="btn btn-default" title="filtro">
        <span class="glyphicon wj-glyph-filter"></span>
    </button>
</div>
<script>...</script>
<script id="detailTemplate"...</script>
@(Html.C1().FlexGrid<SincronizadorOrigenListaViewModel>()
    .Id("pagingGrid")
    .ShowDetailRow(d => d.DetailRowTemplateId("detailTemplate").RowHasDetail("hasDetail").DetailVisibilityMode(C1.Web.Mvc.Grid.DetailVisibilityMode.ExpandMulti))
    .Height(465)
    .ItemsSourceId("collectionViewService")
    .IsReadOnly(true)
    .AllowDelete(true)
    .AutoGenerateColumns(false)
    .Columns(c1sb =>
        {
            c1sb.Add(cb => cb.Binding("TablaOrigen").Header("Tabla Origen").Width("196"));
            c1sb.Add(cb => cb.Binding("ServerOrigen").Header("Server Origen").Width("196"));
            c1sb.Add(cb => cb.Binding("IDRegistro").Header("N°Registro").Format("N").Width("90").Align("center"));
            c1sb.Add(cb => cb.Binding("OperacionRegistro").Header("N° Operacion").Width("90"));
            c1sb.Add(cb => cb.Binding("UsuarioCreacion").Header("Usuario Creacion").Format("G").Width("110"));
            c1sb.Add(cb => cb.Binding("FechaCreacion").Header("Fecha Creacion").Format("dd/MM/yyyy").Width("100").Align("center"));
            c1sb.Add(cb => cb.Binding("UsuarioModificacion").Header("Usuario Modificacion").Format("G").Width("130"));
            c1sb.Add(cb => cb.Binding("FechaModificacion").Header("Fecha Modificacion").Format("dd/MM/yyyy").Width("120").Align("center"));
            c1sb.Add(cb => cb.Binding("DescripcionEstado").Header("Estado").Width("70"));
        }
    )
)
<div class="well grid-sort-group">
    @(Html.C1().Pager().Owner("pagingGrid"))
</div>
<section Scripts {
    <Scripts.Render("~/bundles/jqueryval")
    <script>...</script>
}

```

Figura 136: Estructura de Código – Controlador Modulo Sincronizador Origen – Filtrar

Fuente: Elaboración propia

```

#region Principal
[Authorize(Roles = "Sincronizador Origen Lister")]
public ActionResult Index(int PageInitial = 0)
{
    SincronizadorOrigenListViewModel oSincronizadoresOrigen = new SincronizadorOrigenListViewModel();
    oSincronizadoresOrigen.EstadosOrigen = new List<SelectListItem>(){
        new SelectListItem { Text = "Encolado", Value = "1" },
        new SelectListItem { Text = "Procesado", Value = "2", Selected = true},
    };
    oSincronizadoresOrigen.PageSize = 16;
    oSincronizadoresOrigen.PageInitial = PageInitial;
    return View(oSincronizadoresOrigen);
}

public ActionResult RemoteBind_Read([Cj3sonRequest] CollectionViewRequest<SincronizadorOrigenListViewModel> requestData, DateTime fechaInicial, DateTime fechaFinal, short IdOrigen, int PageInitial = 0)
{
    int nCount = 0;
    requestData.PageIndex = (PageInitial == 0 ? requestData.PageIndex : PageInitial);

    object[] cparams;
    if (ConfigurationManager.AppSettings["Region"].ToString().Equals("US"))
    {
        object[] cparamsTipo = { fechaInicial.ToString("yyyy/MM/dd"), fechaFinal.ToString("yyyy/MM/dd"), IdOrigen };
        cparams = cparamsTipo;
    }
    else
    {
        object[] cparamsTipo = { fechaInicial.ToString("dd/MM/yyyy"), fechaFinal.ToString("dd/MM/yyyy"), IdOrigen };
        cparams = cparamsTipo;
    }

    string cwhere = string.Format("[FechaCreacion] BETWEEN '{0}' AND '{1} 23:59:59' AND IdEstadoSyncOrigen={2}", cparams);
    IList<SincronizadorOrigen> oSincronizadorOrigen = DataRepository.SincronizadorOrigenProvider.GetPaged(cwhere, "", requestData.PageIndex, requestData.PageSize, out nCount);
    if (oSincronizadorOrigen.Count > 0)
    {
        DataRepository.SincronizadorOrigenProvider.DeepLoad(oSincronizadorOrigen, true, DeepLoadType.IncludeChildren, new Type[] { typeof(TList<SincronizadorOrigenLog>), typeof(Usuario) });
        DataRepository.SincronizadorOrigenProvider.DeepLoad(oSincronizadorOrigen, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Estado), typeof(Usuario), typeof(SincronizadorOrigenLog) });
    }

    IList<SincronizadorOrigenListViewModel> oSincronizadorOrigenListViewModels = Mapper.Map<List<SincronizadorOrigen>, List<SincronizadorOrigenListViewModel>>(oSincronizadorOrigen.ToList());
    CollectionViewResponse<SincronizadorOrigenListViewModel> responseData = CollectionViewHelper.Read(requestData, oSincronizadorOrigenListViewModels);
    responseData.PageIndex = requestData.PageIndex;
    responseData.TotalItemCount = nCount;
    return this.Cj3son(responseData);
}
#endregion

```

RF23: El sistema permitirá obtener información de un comprobante de pago electrónico generado en el BackOffice:

Figura 137: Vista Modulo Sincronizador Origen – Operación Obtener Información

Fuente: Elaboración propia

Tabla Origen	Server Origen	N°Registro	N° Operacion	Usuario Creacion	Fecha Creacion	Usuario Modificacion	Fecha Modificacion	Estado
+ DOCUMENTOS	BACKOFFICE	55526	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
- DOCUMENTOS	BACKOFFICE	55527	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
Mensaje Log				Usuario Creacion	Fecha Creacion	Usuario Modificación	Fecha Modificación	
Los Datos complementarios del Documento 001 -0028089 se han registrado correctamente				Admin CTravel	22/03/2019			
El Documento 001 -0028089 y su detalle se han registrado satisfactoriamente				Admin CTravel	22/03/2019			
+ DOCUMENTOS	BACKOFFICE	55528	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO

Figura 138: Estructura de Código – Vista Modulo S:O – Obtener Información

Fuente: Elaboración propia

```

<script>
function hasDetail(row) {
return row.dataItem.SincronizadorOrigenLogLista.length > 0;
}
</script>
<script id="detailTemplate" type="text/template">
@(Html.C1().FlexGrid()
.Height("100px")
.AutoGenerateColumns(false)
.IsReadOnly(true)
.TemplateBind("ItemsSource", "SincronizadorOrigenLogLista")
.Columns(columns =>
{
columns.Add(column => column.Binding("MensajeLog").Header("Mensaje Log").Width("550"));
columns.Add(column => column.Binding("UsuarioCreacion").Header("Usuario Creacion").Width("110").Align("center"));
columns.Add(column => column.Binding("FechaCreacion").Header("Fecha Creacion").Width("100").Align("center"));
columns.Add(column => column.Binding("UsuarioModificacion").Header("Usuario Modificación").Width("130"));
columns.Add(column => column.Binding("FechaModificacion").Header("Fecha Modificación").Width("120"));
})
.ToTemplate()
)
</script>
@(Html.C1().FlexGrid(SincronizadorOrigenListaViewModel)()
.Id("pagingGrid")
.ShowDetailRow(d => d.DetailRowTemplateId("detailTemplate").RowHasDetail("hasDetail").DetailVisibilityMode(C1.Web.Mvc.Grid.DetailVisibilityMode.ExpandMulti))
.Height(465)
.ItemsSourceId("collectionViewService")
.IsReadOnly(true)
.AllowDelete(true)
.AutoGenerateColumns(false)
.Columns(clsb =>
{
clsb.Add(cb => cb.Binding("TablaOrigen").Header("Tabla Origen").Width("196"));
clsb.Add(cb => cb.Binding("ServerOrigen").Header("Server Origen").Width("196"));
clsb.Add(cb => cb.Binding("IdRegistro").Header("N°Registro").Format("R").Width("90").Align("center"));
clsb.Add(cb => cb.Binding("OperacionRegistro").Header("N° Operacion").Width("90"));
clsb.Add(cb => cb.Binding("UsuarioCreacion").Header("Usuario Creacion").Format("G").Width("110"));
clsb.Add(cb => cb.Binding("FechaCreacion").Header("Fecha Creacion").Format("dd/MM/yyyy").Width("100").Align("center"));
clsb.Add(cb => cb.Binding("UsuarioModificacion").Header("Usuario Modificación").Format("G").Width("130"));
clsb.Add(cb => cb.Binding("FechaModificacion").Header("Fecha Modificación").Format("dd/MM/yyyy").Width("120").Align("center"));
clsb.Add(cb => cb.Binding("DescripcionEstado").Header("Estado").Width("70"));
})
)
)

```

RF24: El sistema permitirá listar un historial de los comprobantes de pago electrónicos enviados a la SUNAT.

Figura 139: Vista Modulo Sincronizador Sunat – Operación Filtrar

Fuente: Elaboración propia

Tabla Origen	Server Origen	N°Registro	N° Operacion	Usuario Creacion	Fecha Creacion	Usuario Modificacion	Fecha Modificacion	Estado	
+	DOCUMENTOS	BACKOFFICE	51,920	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+	DOCUMENTOS	BACKOFFICE	51,921	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+	DOCUMENTOS	BACKOFFICE	51,922	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+	DOCUMENTOS	BACKOFFICE	51,923	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+	DOCUMENTOS	BACKOFFICE	51,924	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+	DOCUMENTOS	BACKOFFICE	51,925	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+	DOCUMENTOS	BACKOFFICE	51,926	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+	DOCUMENTOS	BACKOFFICE	51,927	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+	DOCUMENTOS	BACKOFFICE	51,928	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+	DOCUMENTOS	BACKOFFICE	51,929	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+	DOCUMENTOS	BACKOFFICE	51,930	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+	DOCUMENTOS	BACKOFFICE	51,931	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+	DOCUMENTOS	BACKOFFICE	51,932	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+	DOCUMENTOS	BACKOFFICE	51,933	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+	DOCUMENTOS	BACKOFFICE	51,934	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
+	CIERRE DOCUMENTO	BACKOFFICE	50,200	I	Admin CTravel	22/03/2019			ACTIVO

Figura 140: Estructura de Código – Vista Modulo Sincronizador Sunat – Filtrar

Fuente: Elaboración propia

```

using JSUBinet.WebUI.Models;
@model SincronizadorSunatListaViewModel
@{
    ViewBag.Title = "Sincronizador Sunat";
    var fecini = DateTime.Now.AddDays(-15).Date;
    var fecfin = DateTime.Now.Date;
}
@Html.C1().CollectionViewService(SincronizadorSunatListaViewModel)
    .Id("collectionViewService")
    .PageSize(Model.PageSize)
}
<div class="well grid-sort-group">
    @Html.C1().Pager().Owner("collectionViewService").Width(300)
    <label>Inicio </label>
    @Html.C1().InputDate().Id("FechaInicial").Width(120).Value(fecini)
    <label>Final </label>
    @Html.C1().InputDate().Id("FechaFinal").Width(120).Value(fecfin)
    @Html.DropDownList("cboEstado", Model.EstadosSunat, htmlAttributes: new { @class = "form-control", @style = "width:120px;height:30px" })
    <button id="btnFilter"
        class="btn btn-default" title="filtro">
        <span class="glyphicon wj-glyph-filter"></span>
    </button>
</div>
<script id="SendTemplate">...</script>
<script>...</script>
<script id="detailTemplate">...</script>
@Html.C1().FlexGrid(SincronizadorSunatListaViewModel)
    .Id("pagingGrid")
    .ShowDetailRow(d => d.DetailRowTemplateId("detailTemplate").RowHasDetail("hasDetail").DetailVisibilityMode(C1.Web.Mvc.Grid.DetailVisibilityMode.ExpandMulti))
    .Height(465)
    .ItemsSourceId("collectionViewService")
    .IsReadOnly(true)
    .AllowDelete(true)
    .AutoGenerateColumns(false)
    .Columns(cls =>
    {
        clsb.Add(cb => cb.Binding("TablaOrigen").Header("Tabla Origen").Width("100"));
        clsb.Add(cb => cb.Binding("ServerOrigen").Header("Server Origen").Width("200"));
        clsb.Add(cb => cb.Binding("IdRegistro").Header("Nº Registro").Width("80").Align("center"));
        clsb.Add(cb => cb.Binding("OperacionRegistro").Header("Nº Operacion").Width("90"));
        clsb.Add(cb => cb.Binding("UsuarioCreacion").Header("Usuario Creacion").Format("G").Width("110"));
        clsb.Add(cb => cb.Binding("FechaCreacion").Header("Fecha Creacion").Format("dd/MM/yyyy").Width("100"));
        clsb.Add(cb => cb.Binding("UsuarioModificacion").Header("Usuario Modificacion").Format("G").Width("130"));
        clsb.Add(cb => cb.Binding("FechaModificacion").Header("Fecha Modificacion").Format("dd/MM/yyyy").Width("120"));
        clsb.Add(cb => cb.Binding("DescripcionEstado").Header("Estado").Width("70").Align("center"));
        clsb.Add().Name("Reenviar").CellTemplate(b => b.TemplateId("SendTemplate")).Width("25");
    })
}
<div class="well grid-sort-group">
    @Html.C1().Pager().Owner("pagingGrid")
</div>

```

Figura 141: Estructura de Código – Controlador Modulo Sincronizador Sunat - Filtrar

Fuente: Elaboración propia

```

#region Principal
[Authorize(Roles = "Sincronizador Sunat Lister")]
public ActionResult Index(int PageInitial = 0)
{
    SincronizadorSunatListaViewModel oSincronizadoresSunat = new SincronizadorSunatListaViewModel();
    oSincronizadoresSunat.EstadosSunat = new List<SelectListItem>() {
        new SelectListItem { Text = "Pendiente", Value = "1", Selected = true },
        new SelectListItem { Text = "Procesado", Value = "2" },
        new SelectListItem { Text = "Rechazado", Value = "3" },
        new SelectListItem { Text = "EnProceso", Value = "4" }
    };
    oSincronizadoresSunat.PageSize = 16;
    oSincronizadoresSunat.PageInitial = PageInitial;
    return View(oSincronizadoresSunat);
}

public ActionResult RemoteBind_Read([C1]JsonRequest) CollectionViewRequest<SincronizadorSunatListaViewModel> requestData, DateTime fechaInicial, DateTime fechaFinal, short IdSunat, int PageInitial = 0)
{
    int nCount = 0;
    requestData.PageIndex = (PageInitial == 0 ? requestData.PageIndex : PageInitial);

    object[] cparams;
    if (ConfigurationManager.AppSettings["Region"].ToString().Equals("US"))
    {
        object[] cparamsTipo = { fechaInicial.ToString("yyyy/MM/dd"), fechaFinal.ToString("yyyy/MM/dd"), IdSunat };
        cparams = cparamsTipo;
    }
    else
    {
        object[] cparamsTipo = { fechaInicial.ToString("dd/MM/yyyy"), fechaFinal.ToString("dd/MM/yyyy"), IdSunat };
        cparams = cparamsTipo;
    }

    string cWhere = string.Format("[FechaCreacion] BETWEEN '{0}' AND '{1} 23:59:59' AND IdEstadoSyncSunat={2}", cparams);
    TList<SincronizadorSunat> oSincronizadorSunat = DataRepository.SincronizadorSunatProvider.GetPaged(cWhere, "", requestData.PageIndex, requestData.PageSize, out nCount);
    if (oSincronizadorSunat.Count > 0)
    {
        DataRepository.SincronizadorSunatProvider.DeepLoad(oSincronizadorSunat, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Estado), typeof(Usuario) });
        DataRepository.SincronizadorSunatProvider.DeepLoad(oSincronizadorSunat, true, DeepLoadType.IncludeChildren, new Type[] { typeof(TList<SincronizadorSunatLog>), typeof(Usuario) });
    }

    List<SincronizadorSunatListaViewModel> oSincronizadorSunatListaViewModels = Mapper.Map(List<SincronizadorSunat>, List<SincronizadorSunatListaViewModel>)(oSincronizadorSunat.ToList());
    CollectionViewResponse<SincronizadorSunatListaViewModel> responseData = CollectionViewHelper.Read(requestData, oSincronizadorSunatListaViewModels);
    responseData.PageIndex = requestData.PageIndex;
    responseData.TotalItemCount = nCount;
    return this.C1Json(responseData);
}
#endregion

```

RF25: El sistema permitirá obtener información sobre el estado de un comprobante de pago electrónico enviado a la SUNAT.

Figura 142: Vista Modulo Sincronizador Sunat – Obtener Información

Fuente: Elaboración propia

Tabla Origen	Server Origen	N°Registro	N° Operacion	Usuario Creacion	Fecha Creacion	Usuario Modificacion	Fecha Modificacion	Estado
DOCUMENTOS	BACKOFFICE	51,920	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
DOCUMENTOS	BACKOFFICE	51,921	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO
DOCUMENTOS	BACKOFFICE	51,922	I	Admin CTravel	22/03/2019	Admin CTravel	22/03/2019	ACTIVO

Mensaje Log	Usuario Creacion	Fecha Creacion	Usuario Modificación	Fecha Modificación
El Documento F001-0000373 esta listo para ser enviado a la Sunat	Admin CTravel	22/03/2019		
El Documento F001-0000373, ha sido Aceptado por la Sunat	Admin CTravel	22/03/2019		

Figura 143: Estructura de Código – Vista Modulo S:O – Obtener Información

Fuente: Elaboración propia

```

<script>
function hasDetail(row) {
    return row.detailItem.SincronizadorSunatLogLista.length > 0;
}
</script>
<script id="detailTemplate" type="text/template">
<Html.C1().FlexGrid()
    .Height("100px")
    .AutoGenerateColumns(false)
    .IsReadOnly(true)
    .TemplateBind("ItemsSource", "SincronizadorSunatLogLista")
    .Columns(columns =>
    {
        columns.Add(column => column.Binding("MensajeLog").Header("Mensaje Log").Width("450"));
        columns.Add(column => column.Binding("UsuarioCreacion").Header("Usuario Creacion").Width("120").Align("center"));
        columns.Add(column => column.Binding("FechaCreacion").Header("Fecha Creacion").Width("100").Align("center"));
        columns.Add(column => column.Binding("UsuarioModificacion").Header("Usuario Modificación").Width("130").Align("center"));
        columns.Add(column => column.Binding("FechaModificacion").Header("Fecha Modificación").Width("130").Align("center"));
    })
    .ToTemplate()
</script>
<Html.C1().FlexGrid<SincronizadorSunatListaViewModel>()
    .Id("pagingGrid")
    .ShowDetailRow(d => d.DetailRowTemplateId("detailTemplate").RowHasDetail("hasDetail").DetailVisibilityMode(C1.Web.Mvc.Grid.DetailVisibilityMode.ExpandMulti))
    .Height(465)
    .ItemsSourceId("collectionViewService")
    .IsReadOnly(true)
    .AllowDelete(true)
    .AutoGenerateColumns(false)
    .Columns(csb =>
    {
        csb.Add(cb => cb.Binding("TablaOrigen").Header("Tabla Origen").Width("180"));
        csb.Add(cb => cb.Binding("ServerOrigen").Header("Server Origen").Width("200"));
        csb.Add(cb => cb.Binding("IDRegistro").Header("N°Registro").Width("80").Align("center"));
        csb.Add(cb => cb.Binding("OperacionRegistro").Header("N° Operacion").Width("90"));
        csb.Add(cb => cb.Binding("UsuarioCreacion").Header("Usuario Creacion").Format("G").Width("110"));
        csb.Add(cb => cb.Binding("FechaCreacion").Header("Fecha Creacion").Format("dd/MM/yyyy").Width("100"));
        csb.Add(cb => cb.Binding("UsuarioModificacion").Header("Usuario Modificación").Format("G").Width("130"));
        csb.Add(cb => cb.Binding("FechaModificacion").Header("Fecha Modificación").Format("dd/MM/yyyy").Width("120"));
        csb.Add(cb => cb.Binding("DescripcionEstado").Header("Estado").Width("70").Align("center"));
        csb.Add(cb => cb.Binding("Reenviar").CellTemplate(b => b.TemplateId("SendTemplate")).Width("25"));
    })
    )
    
```

RF28: El sistema permitirá descargar los XML generados de los comprobantes de pago electrónicos:

Figura 144: Vista Modulo Comprobantes de Pago – Operación Descarga XML

Fuente: Elaboración propia

	Tipo	Serie	Número	Fecha EM.	RazonSocial	Mon	Neto	IGV	Otros	Dcto	Total	Estado Suma				
+ ✓	01	F001	0000316	01/02/2019	SOCIETE AIR FRANCE SUCURSA...	USD	8.58	1.54	0.00	0.00	10.12	ACE				
+ ✓	01	F001	0000317	01/02/2019	MEDIX TRAVEL AID S.A.C.	USD	6.27	1.13	0.00	0.00	7.40	ACEPTADO				
+ ✓	03	B001	0000670	01/02/2019	REYES RUBEN	USD	52.11	9.38	0.00	0.00	61.49	ACEPTADO				
+ ✓	03	B001	0000671	01/02/2019	URATA FUTOSHI	USD	83.47	15.02	0.00	0.00	98.49	ACEPTADO				
+ ✓	36	001	0027954	01/02/2019	REYES RUBEN	USD	856.00	154.08	78.43	0.00	1,088.51	ACEPTADO				
+ ✓	36	001	0027955	01/02/2019	URATA FUTOSHI	USD	1,206.00	217.08	78.43	0.00	1,501.51	ACEPTADO				

Figura 145: Estructura de Código – Modulo Comprobantes de Pago – Descargar XML

Fuente: Elaboración propia

```
function RequestXml() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    var url = '@Url.Action("RequestXml", "Documentos")?id=' + oCollection.currentItem.IdDocumento;
    window.open(url, 'Visor Xml');
};

function ResponseXml() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    var url = '@Url.Action("ResponseXml", "Documentos")?id=' + oCollection.currentItem.IdDocumento;
    window.open(url, 'Visor Xml');
};
```

Figura 146: Estructura de Código – Controlador C:P: – Descargar XML Parte 1

Fuente: Elaboración propia

```

public ActionResult RequestXml(long id)
{
    string NumeroDocumento = string.Empty;
    Documento oDocumento = DataRepository.DocumentoProvider.GetByIdDocumento(id);
    Empresa oEmpresa = DataRepository.EmpresaProvider.GetAll().SingleOrDefault();
    NumeroDocumento = oEmpresa.RucEmpresa.Trim() + "-" + oDocumento.TipoDocumento.Trim() + "-" + oDocumento.NumeroSerie.Trim() + "-" + oDocumento.NumeroDocumento.Trim();

    var path = string.Format("{0}\\{1}-{2}", FilesHelper.RutaXmlRepositorio, NumeroDocumento, ".xml");

    if (System.IO.File.Exists(path))
    {
        return File(path, "text/xml", NumeroDocumento + ".xml");
    }
    else
    {
        String Result = "";
        MemoryStream MS = new MemoryStream();
        XmlTextWriter W = new XmlTextWriter(MS, Encoding.Unicode);
        XmlDocument D = new XmlDocument();

        try
        {
            // Load the XmlDocument with the XML.
            D.LoadXml(oDocumento.XmlEnvioSunat);

            W.Formatting = Formatting.Indented;

            // Write the XML into a Formatting XmlTextWriter
            D.WriteContentTo(W);
            W.Flush();
            MS.Flush();
            MS.Position = 0;

            // Read MemoryStream contents into a StreamReader.
            StreamReader SR = new StreamReader(MS);

            // Extract the text from the StreamReader.
            String FormattedXML = SR.ReadToEnd();

            Response.Clear();
            Response.ContentType = "text/xml";
            Response.AddHeader("content-disposition", "attachment; filename=" + oDocumento.NroDocumentoIdentidad.Trim() + "-" + oDocumento.TipoDocumento.Trim() + "-" + oDocumento.NumeroSerie.Trim() + "-" + oDocumento.NumeroDocumento.Trim() + ".xml");
            Response.Write(FormattedXML);
            Response.End();

            var serializer = new System.Xml.Serialization.XmlSerializer(D.InnerXml.GetType());
            serializer.Serialize(Response.OutputStream, FormattedXML);

            Result = FormattedXML;
        }
        catch (XmlException)
        {
        }

        MS.Close();
        W.Close();

        return View();
    }
}

```

Figura 147: Estructura de Código – Controlador C:P: – Descargar XML Parte 2

Fuente: Elaboración propia

```

public ActionResult ResponseXml(long id)
{
    string NumeroDocumento = string.Empty;
    Empresa oEmpresa = DataRepository.EmpresaProvider.GetAll().SingleOrDefault();
    Documento oDocumento = DataRepository.DocumentoProvider.GetByIdDocumento(id);

    NumeroDocumento = "R-" + oEmpresa.RucEmpresa.Trim() + "-" + oDocumento.TipoDocumento.Trim() + "-" + oDocumento.NumeroSerie.Trim() + "-" + oDocumento.NumeroDocumento.Trim() + ".xml";

    var cNombreXmlRespuesta = string.Format("{0}\\{1}", FilesHelper.RutaXmlSunat, NumeroDocumento);

    if (System.IO.File.Exists(cNombreXmlRespuesta))
    {
        if (oDocumento.XmlRespuestaSunat == null || oDocumento.XmlRespuestaSunat == string.Empty)
        {
            using (StreamReader sr = new StreamReader(cNombreXmlRespuesta))
            {
                oDocumento.XmlRespuestaSunat = sr.ReadToEnd();
                DataRepository.DocumentoProvider.Save(oDocumento);
            }
        }

        return File(cNombreXmlRespuesta, "text/xml", NumeroDocumento + ".xml");
    }
    else
    {
        if (oDocumento.XmlRespuestaSunat == null || oDocumento.XmlRespuestaSunat == string.Empty)
        {
            return Json("No se ha encontrado ningún CDR del Comprobante Electrónico", JsonRequestBehavior.AllowGet);
        }
        else
        {
            Generamos el XML a partir de la data guardada

            return View();
        }
    }
}

```

RF29: El sistema permitirá exportar la información a un archivo de tipo Excel, que contengan datos de los comprobantes de pago electrónicos de acuerdo al filtro propuesto.

Figura 148: Vista Modulo Comprobantes de Pago- Operación Descargar Formato Excel

Fuente:
Elaboración propia

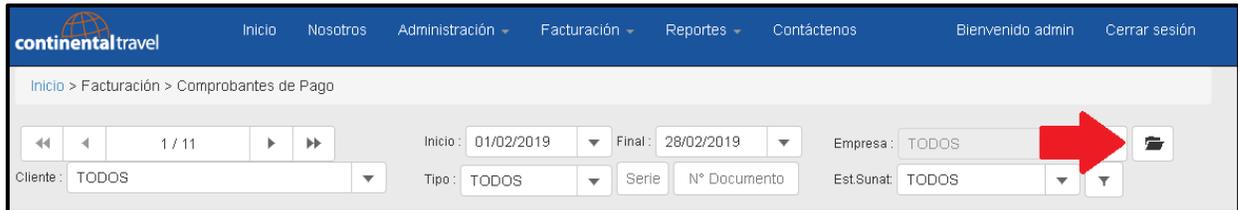


Figura 149: Estructura de Código – Vista Modulo C.P. – Descargar Formato Excel

Fuente: Elaboración propia

```

<script type="text/javascript">
var pagingGrid, oCollection, cboTipoDocumento, cboEmpresa, cboCliente, fechainicio, fechafinal, txtnumerodoc, txtseriedoc, cboEstadoSunat;
cl.mvc.Utils.documentReady(function () {
    pagingGrid = wijmo.Control.getControl("#pagingGrid");
    cboCliente = wijmo.Control.getControl("#cboCliente");
    cboTipoDocumento = wijmo.Control.getControl("#cboTipoDocumento");
    cboEmpresa = wijmo.Control.getControl("#cboEmpresa");
    cboEstadoSunat = wijmo.Control.getControl("#cboEstadoSunat");
    fechainicio = wijmo.Control.getControl("#FechaInicio");
    fechafinal = wijmo.Control.getControl("#FechaFinal");
    txtnumerodoc = $("#NumeroDocumento");
    txtseriedoc = $("#SerieDocumento");
    if (pagingGrid) {
        oCollection = pagingGrid.collectionView;
        pagingGrid.rows.defaultSize = 25;
        updateUrl(@Model.PageInitial);
        oCollection.refresh();
    };
    var dvFilter = document.getElementById("btnFilter");
    dvFilter.addEventListener("click", function () {
        updateUrl(0);
        oCollection.refresh();
    });
    oCollection.currentChanging.addHandler(function () {
        updateUrl(0);
    });
    var dvExcel = document.getElementById("btnExcel");
    dvExcel.addEventListener("click", function () {
        if (!pagingGrid || !oCollection) {
            return;
        }
        window.location.href = @Url.Action("DownloadToExcel", "Documentos")?FechaInicio='
            + encodeURIComponent(formatDate(fechainicio.value)) + "&FechaFinal='
            + encodeURIComponent(formatDate(fechafinal.value)) + "&IdCliente="
            + encodeURIComponent(cboCliente.selectedValue == null ? -1 : cboCliente.selectedValue) + "&IdEmpresa="
            + encodeURIComponent(cboEmpresa.selectedValue) + "&IdTipoDocumento="
            + encodeURIComponent(cboTipoDocumento.selectedValue) + "&IdEstadoSunat="
            + encodeURIComponent(cboEstadoSunat.selectedValue);
    });
});

```

Figura 150: Estructura de Código – Controlador C:P – Descarga Formato Excel

```

public void DownloadToExcel(DateTime FechaInicio, DateTime FechaFinal, long IdCliente, long IdEmpresa, short IdTipoDocumento, short IdEstadoSunat)
{
    capturamos el filtro de comprobantes
    if (oDocumentos.Count > 0)
    {
        DataRepository.DocumentoProvider.DeepLoad(oDocumentos, true, DeepLoadType.IncludeChildren, new Type[] { typeof(EstadoSunat) });
        DataRepository.DocumentoProvider.DeepLoad(oDocumentos, true, DeepLoadType.IncludeChildren, new Type[] { typeof(TList<DocumentoDetalle>), typeof(Producto) });

        DataTable dt = new DataTable();

        Generamos la Cabecera del documento a exportar
        Capturamos los datos a exportar
        foreach (var item in oListDocumentos)
        {
            dt.Rows.Add(item.TipoDocumento,
                string.Format("{0}-{1}", item.NumeroSerie.Trim(), item.NumeroDocumento.Trim()),
                item.FechaEmision.ToShortDateString(),
                normalizarTexto(item.RazonSocial.ToString()),
                item.TipoDocumentoIdentidad.Trim(),
                item.NroDocumentoIdentidad.Trim(),
                item.Moneda.Trim(),
                Math.Round(item.NetoDocumento, 2).ToString(),
                Math.Round(item.ImpuestoDocumento, 2).ToString(),
                Math.Round(item.OtrosImpuestosDocumento, 2).ToString(),
                Math.Round(item.DescuentoDocumento, 2).ToString(),
                Math.Round(item.AfectoDocumento, 2).ToString(),
                Math.Round(item.InafectoDocumento, 2).ToString(),
                Math.Round(item.ExoneradoDocumento, 2).ToString(),
                Math.Round(item.ExportacionDocumento, 2).ToString(),
                Math.Round(item.NetoDocumento + item.ImpuestoDocumento + item.OtrosImpuestosDocumento + item.ExportacionDocumento + item.ExoneradoDocumento - item.DescuentoDocumento, 2).ToString(),
                item.DescripcionEstadosSunat.Trim());
        }

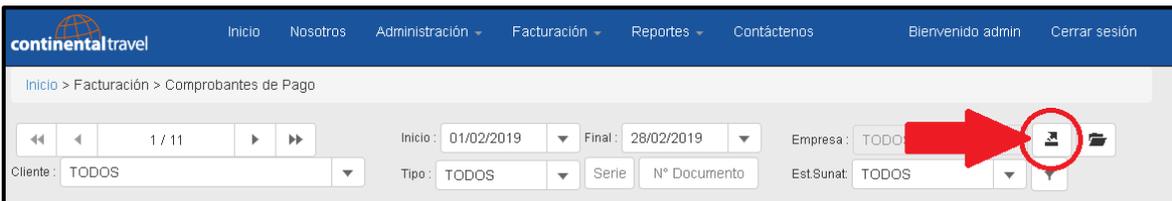
        string attachment = "attachment; filename=ExpordataToExcel_" + FechaInicio.ToString("dd/MM/yyyy") + "_To_" + FechaFinal.ToString("dd/MM/yyyy") + ".xls";
        Response.ClearContent();
        Response.AddHeader("content-disposition", attachment);
        Response.ContentType = "application/vnd.openxmlformats-officedocument.spreadsheetml.sheet";
        string tab = "";
        foreach (DataColumn dc in dt.Columns)
        {
            Response.Write(tab + dc.ColumnName);
            tab = "\t";
        }
        Response.Write("\n");
        int i;
        foreach (DataRow dr in dt.Rows)
        {
            tab = "";
            for (i = 0; i < dt.Columns.Count; i++)
            {
                Response.Write(tab + dr[i].ToString());
                tab = "\t";
            }
            Response.Write("\n");
        }
        Response.End();
    }
}

```

Fuente: Elaboración propia

RF30: El sistema permitirá descargar de manera masiva los XML y los reportes de cada uno de los comprobantes de pago electrónico emitidos.

Figura 151: Vista Comprobantes de Pago – Operación Descarga Masiva



Fuente: Elaboración propia

Figura 152: Estructura de Código – Vista Modulo C.P. – Descarga Masiva

Fuente: Elaboración propia

```
function DownloadXmlPdf() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    $.ajax({
        cache: false,
        url: '@Url.Action("DownloadToXMLsAndPDFs", "Documentos")',
        async: true,
        data: {
            FechaInicio: encodeURIComponent(formatDate(fechainicio.value)),
            FechaFinal: encodeURIComponent(formatDate(fechafinal.value)),
            IdCliente: encodeURIComponent(cboCliente.selectedValue == null ? -1 : cboCliente.selectedValue),
            IdEmpresa: encodeURIComponent(cboEmpresa.selectedValue),
            IdTipoDocumento: encodeURIComponent(cboTipoDocumento.selectedValue),
            IdEstadoSunat: encodeURIComponent(cboEstadoSunat.selectedValue)
        },
        success: function (data) {
            setTimeout(function () {
                $('#c1-grid-mask').hide();
            }, 1000);
            if (data.isRedirect) {
                window.open(data.redirectUrl);
            }
            else
                alert(data);
        },
        error: function (req, status, errorObj) {
            alert(req.responseText);
        }
    });
};
```

Figura 153: Estructura de Código – Controlador Modulo C.P. – Descarga Masiva

Fuente: Elaboración propia

```
public ActionResult DownloadXMLsAndPDFs(string FechaInicio, string FechaFinal, long IdCliente, long IdEmpresa, short IdTipoDocumento, short IdEstadoSunat)
{
    try
    {
        DeleteFilesXMLandPDF();
        Capturamos el filtro

        String nameZip = string.Format("ExportDatatoXML_and_PDF_{0}_to_{1}.zip", dateInicio.ToShortDateString(), dateFinal.ToShortDateString());

        if (oDocumentos.Count > 0)
        {
            String _pathEnProceso = string.Format("{0}\\ListXMLandPDF", FilesHelper.RutaXmlRepositorio);
            String _pathProcesado = string.Format("{0}\\{1}", FilesHelper.RutaXmlRepositorio, nameZip);

            Directory.CreateDirectory(_pathEnProceso);
            DirectoryInfo oDirectorio = new DirectoryInfo(_pathEnProceso);
            DirectoryInfo oDirectorioRepositorio = new DirectoryInfo(FilesHelper.RutaXmlRepositorio);
            bool nCrearTicket = false; bool nCrearPDF = false;

            foreach (Documento oDocu in oDocumentos)
            {
                Generamos el XML del Comprobante
                Generamos el PDF
            }

            if (nCrearTicket == true || nCrearPDF == true)
            {
                if (System.IO.File.Exists(_pathProcesado))
                {
                    System.IO.File.Delete(_pathProcesado);
                }
                FileInfo[] FilesInfo = oDirectorio.GetFiles("*.**", SearchOption.TopDirectoryOnly);
                if (FilesInfo.Length != 0)
                {
                    ZipFile.CreateFromDirectory(_pathEnProceso, _pathProcesado.Replace("/", "-"));
                    string rutaZip = string.Format("Utils/RepositorioXML/{0}", nameZip.Replace("/", "-"));
                    return Json(new { redirectUrl = rutaZip, isRedirect = true }, JsonRequestBehavior.AllowGet);
                }
                else
                {
                    throw new Exception("Ocurrió un problema al comprimir los XMLs y/o PDF disponibles");
                }
            }
            else
            {
                throw new Exception("Ocurrió un problema al guardar los XMLs y/o PDF disponibles");
            }
        }
        else
        {
            throw new Exception("Ocurrió un problema al procesar los XMLs y/o PDF disponibles");
        }
    }
    catch (Exception ex)
    {
        return Json(ex.Message, JsonRequestBehavior.AllowGet);
    }
}
```


Figura 156: Estructura de Código – Vista Modulo Buscar Comprobante – Buscar Parte 2

Fuente: Elaboración propia

```

@Html.Ci().FlexGrid<DocumentoBuscarDetalleViewModels>()
    .Id("pagingGrid")
    .Height(254)
    .Bind(Model.documentos)
    .IsReadOnly(true)
    .AllowDelete(true)
    .AutoGenerateColumns(false)
    .Columns(clsb =>
        {
            clsb.Add(cb => cb.Binding("TipoDocumento").Header("Tipo").Width("40"));
            clsb.Add(cb => cb.Binding("NumeroSerie").Header("Serie").Width("40"));
            clsb.Add(cb => cb.Binding("NumeroDocumento").Header("Numero").Width("90"));
            clsb.Add(cb => cb.Binding("FechaEmision").Header("Emitido").Width("85").Format("dd/MM/yyyy"));
            clsb.Add(cb => cb.Binding("RazonSocial").Header("RazonSocial").Width("365"));
            clsb.Add(cb => cb.Binding("Moneda").Header("Mon").Width("40"));
            clsb.Add(cb => cb.Binding("NetoDocumento").Header("Neto").Format("n2").Width("65"));
            clsb.Add(cb => cb.Binding("ImpuestoDocumento").Header("IGV").Format("n2").Width("70"));
            clsb.Add(cb => cb.Binding("OtrosImpuestosDocumento").Header("Otros").Format("n2").Width("60"));
            clsb.Add(cb => cb.Binding("TotalDocumento").Header("Total").Format("n2").Width("75"));
            clsb.Add(cb => cb.Binding("DescripcionEstadosSumat").Header("Estado Sumat").Width("100"));
            clsb.Add().Name("PrintButton").CellTemplate(b => b.TemplateId("PrintTemplate")).Width("25");
            clsb.Add().Name("XmlResponseButton").CellTemplate(b => b.TemplateId("ResponseXml")).Width("25");
            clsb.Add().Name("XmlRequestButton").CellTemplate(b => b.TemplateId("RequestXml")).Width("25");
        }
    )
}

@section Scripts {
    @Scripts.Render("~/bundles/jqueryval")

    <script type="text/javascript">
        var pagingGrid, oCollection, cboCliente;
        cl.mvc.Utils.documentReady(function () {
            pagingGrid = wijmo.Control.getControl("#pagingGrid");
            cboCliente = wijmo.Control.getControl("#cboCliente");
            if (pagingGrid) {
                oCollection = pagingGrid.collectionView;
                pagingGrid.rows.defaultSize = 25;
            };
            cboCliente.selectedIndexChanged.addHandler(function (e, data) {
                $('#RucDocumento').val(e.selectedValue);
            });
        });
    </script>
}
    
```

Figura 157: Estructura de Código – Controlador Modulo Buscar Comprobante- Buscar

Fuente: Elaboración propia

```

#region Principal
[HttpGet]
public ActionResult Index()
{
    DocumentoBuscarListaViewModels oDocumentoListaModel = new DocumentoBuscarListaViewModels()
    {
        FechaEmision = DateTime.Now.Date,
        TipoDocumentos = (from TipoDocumento oTipodoc in DataRepository.TipoDocumentoProvider.GetAll()
            select new AutoCompleteElement<short>()
            {
                value = oTipodoc.IdTipoDocumento,
                name = oTipodoc.DescripcionTipoDocumento
            }).ToList();
    };
    return View(oDocumentoListaModel);
}

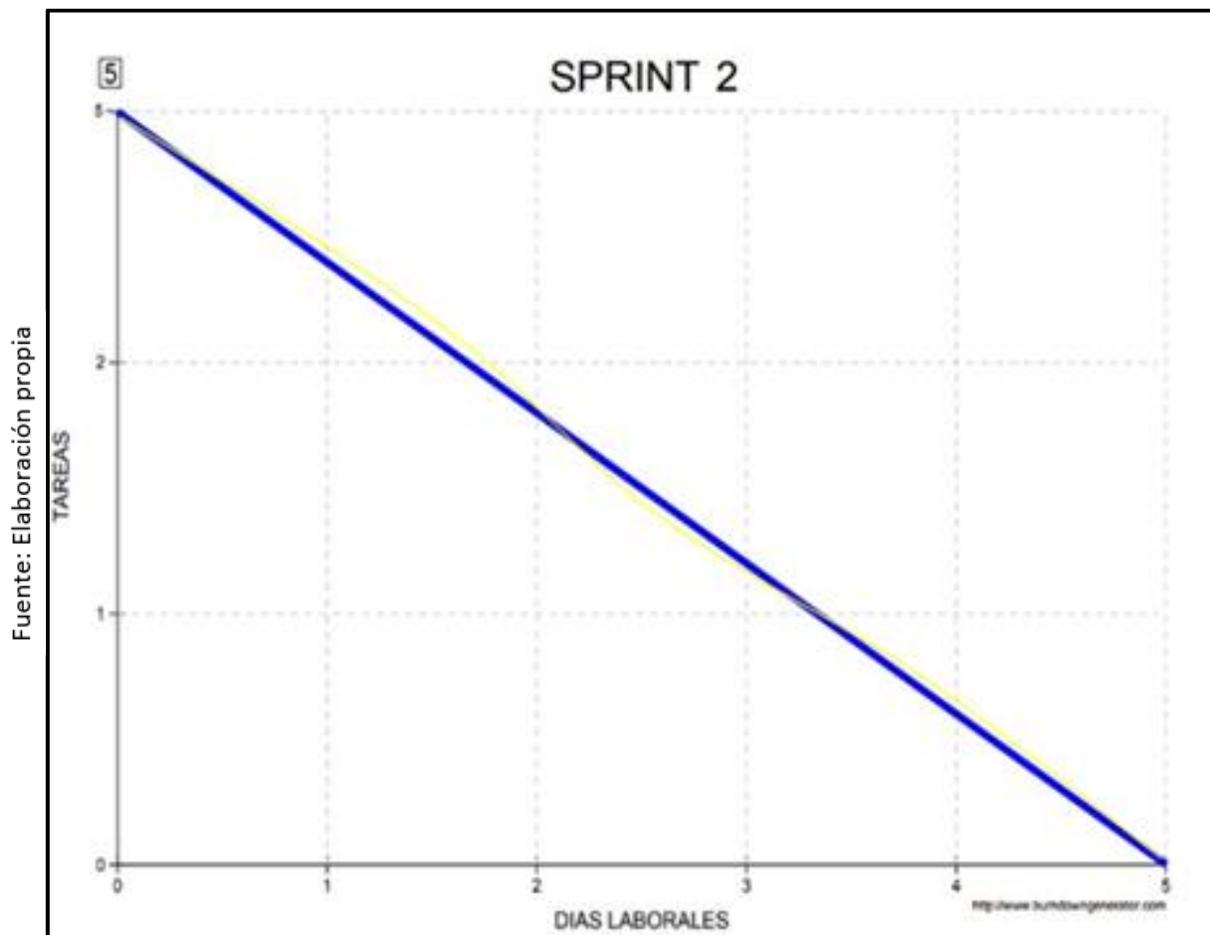
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult Index(DocumentoBuscarListaViewModels oDocumentoListaModel)
{
    oDocumentoListaModel.TipoDocumentos = (from TipoDocumento oTipodoc in DataRepository.TipoDocumentoProvider.GetAll()
        where(x => x.IdEstado == 1 && (x.IdTipoDocumento == 1 || x.IdTipoDocumento == 2 || x.IdTipoDocumento == 7 || x.IdTipoDocumento == 9))
        select new AutoCompleteElement<short>()
        {
            value = oTipodoc.IdTipoDocumento,
            name = oTipodoc.DescripcionTipoDocumento
        }).ToList();

    if (!this.IsCaptchaValid("El valor capcha es invalido. "))
        return View(oDocumentoListaModel);
    TList<Documento> oDocumentos = new TList<Documento>();
    DocumentoQuery oDocumentoQuery = new DocumentoQuery();
    oDocumentoQuery.AppendLike(DocumentoColumn.IdTipoDocumento, oDocumentoListaModel.IdTipoDocumento.ToString());
    oDocumentoQuery.AppendLike(DocumentoColumn.FechaEmision, oDocumentoListaModel.FechaEmision.Date.ToShortDateString());
    oDocumentoQuery.AppendLike(DocumentoColumn.NumeroDocumento, oDocumentoListaModel.NumeroDocumento.Trim().ToString());
    oDocumentoQuery.AppendLike(DocumentoColumn.NumeroSerie, oDocumentoListaModel.SerieDocumento.Trim().ToString());

    TList<Documento> oDocumentosIstlar = DataRepository.DocumentoProvider.Find(oDocumentoQuery.Parameters);
    foreach (Documento oDocumento in oDocumentosIstlar)
    {
        decimal oTotal = (oDocumento.NetoDocumento - oDocumento.DescuentoDocumento) + oDocumento.ImpuestoDocumento + oDocumento.OtrosImpuestosDocumento;
        if (oTotal == oDocumentoListaModel.TotalDocumento)
            oDocumentos.Add(oDocumento);
    }
    if (oDocumentos.Count > 0)
    {
        // en dos fases para impedir el Over Memory
        DataRepository.DocumentoProvider.DeepLoad(oDocumentos, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Estado), typeof(TipoDocumento), typeof(EstadoSumat) });
    }
    else
    {
        this.AddNotification("Documento No Encontrado.", NotificationType.SUCCESS);
        return View(oDocumentoListaModel);
    }
    oDocumentoListaModel.documentos = Mapper.Map<List<Documento>, List<DocumentoBuscarDetalleViewModels>>(oDocumentos.ToList());
    return View(oDocumentoListaModel);
}
}
    
```

GRÁFICA DE BURN DOWN – SPRINT 2

Figura 158: Burn Down Sprint del N° 2



ACTA DE PLANIFICACION SPRINT N° 2

Figura 159: Acta de Planificación del Sprint N°2

Fuente: Elaboración propia

ACTA DE PLANIFICACION DEL SPRINT N° 2

La reunión se lleva a cabo en presencia de los siguientes participantes:

N°	Nombres y Apellidos	Rol
1	Sergio Moran Palacios	Product Owner
2	Ronnie Alarcon Morales	Scrum Master
3	Orvic Yampool Flores Principe	Team Member

El gerente de la empresa Continental Travel S.A.C. realizó la exposición de los requerimientos e indica los requerimientos con mayor prioridad.

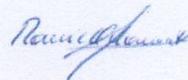
CODIGO	TAREA
RF08	El sistema permitirá filtrar los comprobantes de pago de acuerdo a la fecha de emisión, cliente, tipo de comprobante, estado, serie y numero de comprobante.
RF12	El sistema permitirá enviar un XML y un reporte del comprobante de pago electrónico al Cliente, para su posterior confirmación.
RF15	El sistema permitirá filtrar los cierres diarios generados en el día, mediante un filtro de fecha de emisión y estado del comprobante.
RF19	El sistema permitirá listar los comprobantes de pago electrónicos relacionados a la factura que hayan sido anulados anteriormente, mediante un filtro de fecha de emisión y estado del comprobante.
RF22	El sistema permitirá listar un historial de los comprobantes de pago electrónicos generados en el BackOffice.
RF23	El sistema permitirá obtener información de un comprobante de pago electrónico generado en el BackOffice.
RF24	El sistema permitirá listar un historial de los comprobantes de pago electrónicos enviados a la SUNAT.
RF25	El sistema permitirá obtener información sobre el estado de un comprobante de pago electrónico enviado a la SUNAT.
RF28	El sistema permitirá descargar los XML generados de los comprobantes de pago electrónicos.
RF29	El sistema permitirá exportar la información a un archivo de tipo Excel, que contengan datos de los comprobantes de pago electrónicos de acuerdo al filtro propuesto en la interfaz.
RF30	El sistema permitirá descargar de manera masiva los XML y los reportes de cada uno de los comprobantes de pago electrónico emitidos de acuerdo al filtro propuesto en la interfaz.
RF38	Solo el usuario no registrado podrá consultar la emisión de los comprobantes electrónicos si cumple los filtros de número de serie, numero de documento, RUC del Cliente, fecha de emisión y monto total del comprobante.

Mediante la presente acta se válida y se da conformidad los objetivos a realizarse en el Sprint 2 por parte del equipo de desarrollo, como también, las tareas propuestas del producto (Historias de Usuarios) que contiene el sprint mencionado.

Los asistentes impartirán su aprobación de acuerdo a lo presentado en la planificación del Sprint 0, indicando que la fecha de entrega de este Sprint sería el día 11 de Septiembre del 2019.

Lima, 3 de Septiembre del 2019.


CONTINENTAL TRAVEL S.A.C.
AGENCIA DE VIAJES Y TURISMO



ACTA DE REUNION SPRINT N° 2

Figura 160: Acta de Reunión del Sprint N° 2

Fuente: Elaboración propia

ACTA DE REUNION DEL SPRINT N° 2

La reunión se lleva a cabo en presencia de los siguientes participantes:

Presentes:

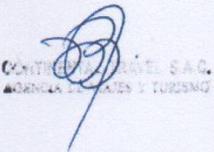
N°	Nombres y Apellidos	Rol
1	Sergio Moran Palacios	Product Owner
2	Ronnie Alarcon Morales	Scrum Master
3	Orvic Yampool Flores Principe	Team Member

En la presente reunión, se presentó un prototipo del software y las observaciones levantadas propuestas en la anterior reunión, el cual fue aceptado; dándonos una conformidad positiva de los usuarios presentes en la reunión.

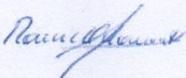
Asimismo, se presentó los módulos de Comprobantes de Pago, Anulación de Comprobantes, Cierre de Comprobantes, Sincronizador Origen y Sincronizador Sunat; al concluir se validó la gestión de datos de cada uno de ellos.

Por consiguiente, se brinda aprobación del Sprint N° 2 ya concluido, para el desarrollo de la Aplicación Web Progresiva para el proceso de facturación de la empresa Continental Travel S.A.C.

Lima, 11 de Septiembre del 2019.



CONTINENTAL TRAVEL S.A.C.
AGENCIA DE VIAJES Y TURISMO



3.4. Desarrollo del Sprint 3

Tabla 46. Sprint 3 – Módulo de Envío y/o Recepción de Comprobantes de Pago Electrónico

SPRINT	CODIGO	TAREA	ESTIMACIÓN (DIAS)	PRIORIDAD
SPRINT 3	RF09	El sistema permitirá generar un XML y enviar un comprobante de pago electrónico a la SUNAT.	5	ALTA
	RF10	El sistema permitirá recepcionar el XML de respuesta por parte de la SUNAT e interpretar su estado actual.	5	ALTA
	RF13	El sistema permitirá reenviar un XML de un comprobante de pago electrónico que haya presentado inconvenientes en su envío a la SUNAT.	5	ALTA
	RF14	El sistema permitirá anular los comprobantes de pago electrónicos emitidos, y enviar un XML de anulación a la SUNAT.	5	ALTA
	RF16	El sistema permitirá generar un XML y enviar un nuevo cierre diario de los comprobantes de pago relacionados a la Boleta, para su posterior registro en la SUNAT.	5	ALTA
	RF17	El sistema permitirá obtener un XML de respuesta del cierre diario de los comprobantes de pago electrónicos relacionados a la Boleta, para su posterior validación.	5	ALTA
	RF18	El sistema permitirá reenviar el XML de un cierre diario de los comprobantes de pago electrónicos relacionados a la Boleta, para su posterior registro en la SUNAT.	5	ALTA
	RF20	El Sistema permitirá obtener el XML de envío de un comprobante de pago electrónico anulado anteriormente.	5	ALTA
	RF21	El Sistema permitirá obtener XML de respuesta de un comprobante de pago electrónico anulado anteriormente.	5	ALTA

Fuente: Elaboración propia

RF09: El sistema permitirá generar un XML y enviar un comprobante de pago electrónico a la SUNAT.

Figura 161: Vista Modulo Reprocesar CPE – Operación Enviar XML

Fuente: Elaboración propia

Tipo	Serie	Número	Fecha EM	RazonSocial	Mon	Neto	IGV	Otros	Dsccto	Total	Estado Sunat				
03	B001	0000790	29/03/2019	KRISTIAN GARCIA MEZA	USD	50.85	9.15	0.00	0.00	60.00	PENDIENTE				
03	B001	0000792	29/03/2019	ECHEVERRIA VILMA	USD	86.26	15.53	0.00	0.00	101.79	PENDIENTE				
01	F001	0000389	29/03/2019	REAL TOURS TOKYO	USD	3.22	0.58	0.00	0.00	3.80	PENDIENTE				

Figura 162: Estructura de Código – Vista Modulo Reprocesar CPE – Enviar XML

Fuente: Elaboración propia

```
function SendSunat() {
    if (!confirm("Esta seguro que desea enviar este comprobante? ") || !pagingGrid || !oCollection) {
        return;
    }
    $.ajax({
        type: 'GET',
        cache: false,
        url: Url.Action("SendXml", "Facturacion"),
        async: false,
        data: { id: oCollection.currentItem.IdDocumento },
        success: function (data) {
            alert(data);
        }
    });
    updateUrl(0)
    oCollection.refresh();
};
```

Figura 163: Estructura de Código – Controlador Reprocesar CPE – Enviar XML

Fuente: Elaboración propia

```
public ActionResult SendXml(long id)
{
    Documento oDocumento = DataRepository.DocumentoProvider.GetByIdDocumento(id);
    SincronizadorSunat oSincronizadorSunat = DataRepository.SincronizadorSunatProvider.GetByServerOrigenTablaOrigenIdRegistro(ConfigurationManager.AppSettings["Empresa"], "D");

    if (oDocumento != null)
    {
        if (oDocumento.IdEstadoSunat == 3)
            return Json("Comprobante Rechazado, no valido para reenviar.", JsonRequestBehavior.AllowGet);
        else
        {
            if (oDocumento.IdEstadoSunat == 5)
                return Json("Comprobante Anulado, no valido para reenviar.", JsonRequestBehavior.AllowGet);
            else
            {
                if (oDocumento.IdEstadoSunat == 2)
                    return Json("Comprobante Aceptado, no valido para reenviar.", JsonRequestBehavior.AllowGet);
                else
                {
                    string oNumeroDocumento = string.Empty; string oTag = string.Empty;
                    oNumeroDocumento = oDocumento.NumeroSerie.Trim() + "-" + oDocumento.NumeroDocumento.Trim();

                    WsNet.WsDataImport oWsDataImport = new WsNet.WsDataImport();
                    try
                    {
                        int oResult = 0;
                        if (ConfigurationManager.AppSettings["URL"].ToString().Equals("21"))
                            oResult = oWsDataImport.ProcesarDocumentoElectronico_v21(oDocumento.IdDocumento, ref oNumeroDocumento);
                        else
                            oResult = oWsDataImport.ProcesarDocumentoElectronico(oDocumento.IdDocumento, ref oNumeroDocumento);

                        if (oResult == 0)
                            throw (new Exception("Error al generar el XML y/o enviar el Comprobante de Pago."));

                        Guardo los resultados del Envio del comprobante
                        Verifico XML de respuesta obtenido

                        return Json(oTag, JsonRequestBehavior.AllowGet);
                    }
                    catch (Exception ex)
                    {
                        SincronizadorSunat oSincronizadorSunatError = oSincronizadorSunat;
                        oSincronizadorSunatError.Tag = string.Format("El Documento {0} presenta errores {1}", oNumeroDocumento, ex.Message);
                        oSincronizadorSunatError.IdEstadoSyncSunat = 3;
                        oWsDataImport.SincronizadorSunatProvider.CrearSincronizadorSunat(ref oSincronizadorSunatError);
                        return Json(string.Format("Error insertando el comprobante {0}-{1} {2}", oDocumento.NumeroSerie, oDocumento.NumeroDocumento, ex.Message), JsonRequestBehavior.AllowGet);
                    }
                }
            }
        }
    }
    else
        return Json("Ocurrió un error al enviar el Comprobante, contacte con el personal de Soporte.", JsonRequestBehavior.AllowGet);
}
```

RF10: El sistema permitirá recepcionar el XML de respuesta por parte de la SUNAT e interpretar su estado actual.

Figura 164: Vista Modulo Reprocesar CPE – Operación Recepción XML

Fuente: Elaboración propia

Tipo	Serie	Número	Fecha EM	RazonSocial	Mon	Neto	IGV	Otros	Dscto	Total	Estado Sunat	Acciones			
03	B001	0000790	29/03/2019	KRISTIAN GARCIA MEZA	USD	50.85	9.15	0.00	0.00	60.00	PENDIENTE	[Iconos]			
03	B001	0000792	29/03/2019	ECHEVERRIA VILMA	USD	86.26	15.53	0.00	0.00	101.79	PENDIENTE	[Iconos]			
01	F001	0000389	29/03/2019	REAL TOURS TOKYO	USD	3.22	0.58	0.00	0.00	3.80	PENDIENTE	[Iconos]			

Figura 165: Estructura de Código – Vista Modulo Reprocesar CPE – Recepcionar XML

Fuente: Elaboración propia

```
function CdrSunat() {
    if (!confirm("Esta seguro que desea recuperar el CDR de este comprobante? ") || !pagingGrid || !oCollection) {
        return;
    }
    $.ajax({
        type: 'GET',
        cache: false,
        url: '@Url.Action("FindCDR", "Facturacion")',
        async: false,
        data: { id: oCollection.currentItem.IdDocumento },
        success: function (data) {
            if (data == "OK") {
                alert("Enhorabuena, CDR Recuperado.");
            }
            else if (data == "ERROR") {
                alert("Ocurrio un problema al recuperar el CDR.");
            }
            else {
                alert(data);
            }
        }
    });
    updateUrl(0);
    oCollection.refresh();
};
```

Figura 166: Estructura de Código – Controlador Reprocesar CPE – Recepcionar XML

Fuente: Elaboración propia

```
public ActionResult FindCDR(long id)
{
    string oNumeroDocumento = string.Empty;
    Empresa oEmpresa = DataRepository.EmpresaProvider.GetAll().SingleOrDefault();
    Documento oDocumento = DataRepository.DocumentoProvider.GetById(id);
    WsNet.WsDataImport oWsDataImport = new WsNet.WsDataImport();
    SincronizadorSunat oSincronizadorSunat = DataRepository.SincronizadorSunatProvider.GetByServerOrigenTablaOrigenIdRegistro(ConfigurationManager.AppSettings["Empres

oNumeroDocumento = oDocumento.NumeroSerie.Trim() + "-" + oDocumento.NumeroDocumento.Trim();

if (oDocumento.NumeroSerie.Substring(0, 1) == "B")
    return Json("Solo se puede recuperar los CDR de la Factura o Notas de Crédito de Factura por esta vista.", JsonRequestBehavior.AllowGet);

string NumeroDocumento = string.Empty;
NumeroDocumento = "R-" + oEmpresa.RucEmpresa.Trim() + "-" + oDocumento.TipoDocumento.Trim() + "-" + oDocumento.NumeroSerie.Trim() + "-" + oDocumento.NumeroDocumen

var cNombreXmlRespuesta = string.Format("{0}\\{1}", FilesHelper.RutaXmlSunat, NumeroDocumento);

if (System.IO.File.Exists(cNombreXmlRespuesta))
{
    verificamos si XML esta almacenado y es valido
    FilesHelper.verificarEstadoCdrNodoXML(oDocumento);
    return Json("El CDR ya existe.", JsonRequestBehavior.AllowGet);
}
else
{
    try
    {
        int oResult = oWsDataImport.RecuperarCdrDocumento(oSincronizadorSunat.IdRegistro, ref oNumeroDocumento);
        guardamos resultado de la recepcion del XML de respuesta
        oWsDataImport.DocumentoProvider_GuardarEstadoSunat(oDocumento, oResult);
        FilesHelper.verificarEstadoCdrNodoXML(oDocumento);
        return Json("OK", JsonRequestBehavior.AllowGet);
    }
    catch (Exception ex)
    {
        SincronizadorSunat oSincronizadorSunatError = oSincronizadorSunat;
        oSincronizadorSunatError.Tag = string.Format("El Documento {0} presenta errores {1}", oNumeroDocumento, ex.Message);
        oSincronizadorSunatError.IdEstadoSyncSunat = 3;
        oWsDataImport.SincronizadorSunatProvider_CrearSincronizadorSunat(ref oSincronizadorSunatError);
        return Json(string.Format("Error recuperando el comprobante {0}-{1} {2}", oDocumento.NumeroSerie, oDocumento.NumeroDocumento, ex.Message), JsonRequestBehaviorBeha
    }
}
```

RF14: El sistema permitirá anular los comprobantes de pago electrónicos emitidos, y enviar un XML de anulación a la SUNAT.

Figura 167: Vista Modulo Reprocesar CPE – Operación Anular Comprobante

Fuente: Elaboración propia

Tipo	Serie	Número	Fecha EM	RazonSocial	Mon	Neto	IGV	Otros	Dcto	Total	Estado Sunat	Icons
03	B001	0000790	29/03/2019	KRISTIAN GARCIA MEZA	USD	50.85	9.15	0.00	0.00	60.00	PENDIENTE	[Icons]
03	B001	0000792	29/03/2019	ECHEVERRIA VILMA	USD	86.26	15.53	0.00	0.00	101.79	PENDIENTE	[Icons]
01	F001	0000389	29/03/2019	REAL TOURS TOKYO	USD	3.22	0.58	0.00	0.00	3.80	PENDIENTE	[Icons]

Figura 168: Estructura de Código – Vista Reprocesar CPE – Anular Comprobante

Fuente: Elaboración propia

```
function DeleteSunat() {
    if (!confirm("Esta seguro que desea anular este comprobante?") || !pagingGrid || !oCollection) {
        return;
    }
    $.ajax({
        type: 'GET',
        cache: false,
        url: '@Url.Action("AnularComprobante", "Facturacion")',
        async: false,
        data: { id: oCollection.currentItem.IdDocumento },
        success: function (data) {
            if (data == "OK") {
                alert("Enhorabuena, Comprobante Anulado listo para ser enviado.");
            }
            else {
                alert(data);
            }
        }
    });
    updateUrl(0);
    oCollection.refresh();
};
```

Figura 169: Estructura de Código – Controlador Modulo Reprocesar CPE – Anular Comprobante

Fuente: Elaboración propia

```
public ActionResult AnularComprobante(long id)
{
    MsNet.WsDataImport oWsDataImport = new MsNet.WsDataImport();
    Documento oDocumentoAnulado = DataRepository.DocumentoProvider.GetByIdDocumento(id);

    if (oDocumentoAnulado.IdEstadoSunat == 4 && oDocumentoAnulado.IdEstado == 2)
        return Json("El Comprobantes ya ha sido marcado para anularse, esta a espera de la respuesta.", JsonRequestBehavior.AllowGet);
    else if (oDocumentoAnulado.IdEstadoSunat == 5)
        return Json("El Comprobantes ya esta anulado.", JsonRequestBehavior.AllowGet);
    else if (oDocumentoAnulado.IdEstadoSunat == 4 || oDocumentoAnulado.IdEstadoSunat == 1)
        return Json("El Comprobantes no ha sido aceptado por la SUNAT, se recomienda que haya sido ACEPTADO antes de anular.", JsonRequestBehavior.AllowGet);
    else if (oDocumentoAnulado.IdEstadoSunat == 3)
        return Json("El Comprobantes esta rechazado no es posible anular.", JsonRequestBehavior.AllowGet);

    try
    {
        string esRechazado = oWsDataImport.Documento_VerificarAnulados(oDocumentoAnulado.IdDocumentoReferenciaExterna.ToString().Trim());
        if (esRechazado == "3")
        {
            return Json("El Comprobantes no ha sido aceptado por la SUNAT, se recomienda que haya sido ACEPTADO antes de anular.", JsonRequestBehavior.AllowGet);
        }

        if (oDocumentoAnulado.TipoDocumento != "03")
        {
            SincronizadorOrigen oSincronizadorOrigen = new SincronizadorOrigen()
            {
                EntityState = EntityState.Added,
                TablaOrigen = "ANULACION DOCUMENTO",
                ServerOrigen = ConfigurationManager.AppSettings["Empresa"],
                IdRegistro = (long)Convert.ToInt32(oDocumentoAnulado.IdDocumentoReferenciaExterna),
                FechaCreacion = DateTime.Now,
                Tag = string.Format("El Documento anulado {0}-{1} se ha registrado correctamente", oDocumentoAnulado.NumeroSerie, oDocumentoAnulado.NumeroDocumento)
            };
            oWsDataImport.SincronizadorOrigenProvider_CrearSincronizadorOrigen(ref oSincronizadorOrigen);
        }

        return Json("Comprobante ha sido marcado para Anularse, listo para crear un Resumen de Anulaciones.", JsonRequestBehavior.AllowGet);
    }
    catch (Exception ex)
    {
        SincronizadorOrigen oSincronizadorOrigen = new SincronizadorOrigen()
        {
            EntityState = EntityState.Added,
            TablaOrigen = "ANULACION DOCUMENTO",
            ServerOrigen = ConfigurationManager.AppSettings["Empresa"],
            IdRegistro = (long)Convert.ToInt32(oDocumentoAnulado.IdDocumentoReferenciaExterna),
            Tag = string.Format("Error insertando el documento anulado {0}-{1} {2}", oDocumentoAnulado.NumeroSerie, oDocumentoAnulado.NumeroDocumento, ex.Message)
        };
        oWsDataImport.SincronizadorOrigenProvider_CrearSincronizadorOrigen(ref oSincronizadorOrigen);
        return Json(string.Format("Error insertando el documento anulado {0}-{1} {2}", oDocumentoAnulado.NumeroSerie, oDocumentoAnulado.NumeroDocumento, ex.Message), JsonRequestBehavior.AllowGet);
    }
}
```

RF16: El sistema permitirá generar un XML y enviar un nuevo cierre diario de los comprobantes de pago relacionados a la Boleta, para su posterior registro en la SUNAT.

Figura 170: Vista Modulo Resumen Diario- Operación Enviar XML

Fuente:
Elaboración propia



Figura 171: Estructura de Código – Vista Modulo Resumen Diario – Enviar XML Parte 1

Fuente:
Elaboración propia

```
<label style="margin-left:10px">Crear Resumen:</label>
@(Html.Ci().InputDate().Id("FechaCierreResumen").Width(120).Value(fecfin))
<button id="btnNew" class="btn btn-default" title="Nuevo Resumen Diario">
<span class="glyphicon glyphicon-log-in"></span>
</button>
```

Figura 172: Estructura de Código – Vista Modulo Resumen Diario – Enviar XML Parte 2

Fuente: Elaboración propia

```
var dvNewCierre = document.getElementById('btnNew');
dvNewCierre.addEventListener("click", function () {
$.ajax({
cache: false,
url: '@Url.Action("NuevoResumen", "DocumentoCierres")',
async: false,
data: { fechacierre: encodeURIComponent(formatDate(fechaCierreResumen.value)) },
success: function (data) {
alert(data);
updateUrl(0);
oCollection.refresh();
},
error: function (req, status, errorObj) {
alert(req.responseText);
}
});
});
```

Figura 173: Estructura de Código – Controlador Modulo Resumen Diario – Enviar XML

Fuente: Elaboración propia

```
public ActionResult NuevoResumen(string fechacierre)
{
    WsNet.WsDataImport oWsDataImport = new WsNet.WsDataImport();

    try
    {
        DateTime dayCierre;
        dayCierre = DateTime.Parse(fechacierre.Replace("%2F", "/").Substring(0, 10));

        long idDocumentoCierre = oWsDataImport.DocumentoProvider_EnviarResumenBoletas(dayCierre, ConfigurationManager.AppSettings["Empresa"], "CIERRE DOCUMENTO");

        SincronizadorSunat oSincronizadorSunat = new SincronizadorSunat()
        {
            EntityState = EntityState.Added,
            TablaOrigen = "CIERRE DOCUMENTO",
            ServerOrigen = ConfigurationManager.AppSettings["Empresa"],
            IdRegistro = idDocumentoCierre,
            Tag = string.Format("El Resumen Diario de Boletas y Notas relacionadas estan listos para ser enviados a la Sunat")
        };

        oWsDataImport.SincronizadorSunatProvider_CrearSincronizadorSunat(ref oSincronizadorSunat);

        return Json("Resumen enviado", JsonRequestBehavior.AllowGet);
    }
    catch (Exception ex)
    {
        return Json(ex.Message, JsonRequestBehavior.AllowGet);
    }
}
```

RF17: El sistema permitirá obtener un XML de respuesta del cierre diario de los comprobantes de pago electrónicos relacionados a la Boleta, para sus posteriores validaciones.

Figura 174: Vista Resumen Diario – Operación Obtener Respuesta XML

Fuente: Elaboración propia

N° ticket	Descripción Cierre	Fecha Cierre	Usuario Creación	Fecha Creación	Usuario Modificación	Fecha Modificación	Estado				
201903541499208	RESUMEN DE BOLETAS	28/03/2019	demos demos123	28/03/2019	demos demos123	28/03/2019	ACTIVO				
201903547299016	RESUMEN DE BOLETAS	29/03/2019	demos demos123	29/03/2019	demos demos123	29/03/2019	ACTIVO				
201903555974318	RESUMEN DE BOLETAS	26/03/2019	demos demos123	01/04/2019	demos demos123	01/04/2019	ACTIVO				
201903555975846	RESUMEN DE BOLETAS	27/03/2019	demos demos123	01/04/2019	demos demos123	01/04/2019	ACTIVO				
201903595956579	RESUMEN DE BOLETAS	01/04/2019	demos demos123	01/04/2019	demos demos123	01/04/2019	ACTIVO				
201903574505608	RESUMEN DE BOLETAS	04/04/2019	demos demos123	04/04/2019	demos demos123	04/04/2019	ACTIVO				
201903578983784	RESUMEN DE BOLETAS	05/04/2019	demos demos123	05/04/2019	demos demos123	05/04/2019	ACTIVO				
201903583079529	RESUMEN DE BOLETAS	06/04/2019	demos demos123	06/04/2019	demos demos123	06/04/2019	ACTIVO				

Figura 175: Estructura de Código – Vista Modulo R. Diario – Obtener Respuesta XML

Fuente: Elaboración propia

```
function RecuperarRespuesta() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    $.ajax({
        cache: false,
        url: 'Url.Action("RecuperarResumen", "DocumentoCierres")',
        async: false,
        data: { id: oCollection.currentItem.IdDocumentoCierre },
        success: function (data) {
            alert(data);
            updateUrl(0)
            oCollection.refresh();
        },
        error: function (req, status, errorObj) {
            alert(req.responseText);
        }
    });
};
```

Figura 176: Estructura de Código – Controlador Modulo R: Diario – Obtener Respuesta

Fuente: Elaboración propia

```
public ActionResult RecuperarResumen(long id)
{
    WsNet.WsDataImport oWsDataImport = new WsNet.WsDataImport();
    DocumentoCierre oDocumentoCierre = DataRepository.DocumentoCierreProvider.GetByIdDocumentoCierre(id);

    if (oDocumentoCierre.DescripcionCierre == "ANULACION DE RESUMEN")
        return Json("Los resúmenes de anulaciones se obtienen al final del día.", JsonRequestBehavior.AllowGet);

    try
    {
        if (oDocumentoCierre.IdEstadoSunat == 2)
            return Json("Documento ya ha sido Aceptado, revise los documentos a la fecha registrada.", JsonRequestBehavior.AllowGet);
        if (oDocumentoCierre.IdEstado == 2)
            return Json("Documento ya ha sido Anulado, revise los documentos a la fecha registrada.", JsonRequestBehavior.AllowGet);
        if (oDocumentoCierre.IdEstado == 5)
            return Json("Documento ya ha sido Cerrado, revise los documentos a la fecha registrada.", JsonRequestBehavior.AllowGet);

        DataRepository.UsuarioProvider.DeepLoad(oUsuario, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Empresa) });

        bool recuperacion = oWsDataImport.DocumentoProvider_ProcesarTicketCierre(Configuration.AppSettings["Empresa"], "CIERRE DOCUMENTO", oUsuario.IdEmpresa);
        if (recuperacion == false)
        {
            throw (new Exception());
        }

        return Json("Se inició el proceso de recuperación, por favor espere", JsonRequestBehavior.AllowGet);
    }
    catch (Exception ex)
    {
        return Json(ex.Message, JsonRequestBehavior.AllowGet);
    }
}

#endregion
```

RF18: El sistema permitirá reenviar el XML de un cierre diario de los comprobantes de pago electrónicos relacionados a la Boleta, para su posterior registro en la SUNAT.

Figura 177: Vista Modulo Resumen Diario – Operación Reenviar XML

Fuente: Elaboración propia

N° ticket	Descripción Cierre	Fecha Cierre	Usuario Creación	Fecha Creación	Usuario Modificación	Fecha Modificación	Estado	Acciones
201903541499208	RESUMEN DE BOLETAS	28/03/2019	demoss demos123	28/03/2019	demoss demos123	28/03/2019	ACTIVO	[Reenviar XML]
201903547299016	RESUMEN DE BOLETAS	29/03/2019	demoss demos123	29/03/2019	demoss demos123	29/03/2019	ACTIVO	[Reenviar XML]
201903555974318	RESUMEN DE BOLETAS	26/03/2019	demoss demos123	01/04/2019	demoss demos123	01/04/2019	ACTIVO	[Reenviar XML]
201903555975846	RESUMEN DE BOLETAS	27/03/2019	demoss demos123	01/04/2019	demoss demos123	01/04/2019	ACTIVO	[Reenviar XML]
201903559556579	RESUMEN DE BOLETAS	01/04/2019	demoss demos123	01/04/2019	demoss demos123	01/04/2019	ACTIVO	[Reenviar XML]
201903574505608	RESUMEN DE BOLETAS	04/04/2019	demoss demos123	04/04/2019	demoss demos123	04/04/2019	ACTIVO	[Reenviar XML]
201903578983784	RESUMEN DE BOLETAS	05/04/2019	demoss demos123	05/04/2019	demoss demos123	05/04/2019	ACTIVO	[Reenviar XML]
201903583079529	RESUMEN DE BOLETAS	06/04/2019	demoss demos123	06/04/2019	demoss demos123	06/04/2019	ACTIVO	[Reenviar XML]

Figura 178: Estructura de Código – Vista Modulo Resumen Diario – Reenviar XML

Fuente: Elaboración propia

```

function ReenviarCierre() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    $.ajax({
        cache: false,
        url: '@Url.Action("ReenviarResumen", "DocumentoCierres")',
        async: false,
        data: { id: oCollection.currentItem.IdDocumentoCierre },
        success: function (data) {
            alert(data);
            updateUrl(0);
            oCollection.refresh();
        },
        error: function (req, status, errorObj) {
            alert(req.responseText);
        }
    });
}
    
```

Figura 179: Estructura de Código – Controlador Modulo Resumen Diario – Reenviar XML

Fuente: Elaboración propia

```

public ActionResult ReenviarResumen(long id)
{
    WsNet.WsDataImport oWsDataImport = new WsNet.WsDataImport();
    DocumentoCierre oDocumentoCierre = DataRepository.DocumentoCierreProvider.GetByIdDocumentoCierre(id);

    try
    {
        if (oDocumentoCierre.IdEstado == 2)
            return Json("Documento ya ha sido Anulado, revise los documentos a la fecha registrada.", JsonRequestBehavior.AllowGet);
        if (oDocumentoCierre.IdEstado == 5)
            return Json("Documento ya ha sido Cerrado, revise los documentos a la fecha registrada.", JsonRequestBehavior.AllowGet);

        DocumentoQuery oDocumentosCierreDiarioQuery = new DocumentoQuery();
        oDocumentosCierreDiarioQuery.AppendEquals(DocumentoColumn.FechaEmision, oDocumentoCierre.FechaCierre.ToShortDateString());
        oDocumentosCierreDiarioQuery.AppendEquals(DocumentoColumn.IdEstadoSuntat, "4");
        oDocumentosCierreDiarioQuery.AppendLike(DocumentoColumn.NumeroSerie, "88");
        TList<Documento> oDocumentosCierreDiaros = DataRepository.DocumentoProvider.Find(oDocumentosCierreDiarioQuery.GetParameters());

        foreach (Documento oDocumento in oDocumentosCierreDiaros)
        {
            oDocumento.EntityState = EntityState.Changed;
            oDocumento.IdDocumentoCierre = null;
            DataRepository.DocumentoProvider.Save(oDocumento);
        }

        long idReenvio = oWsDataImport.DocumentoProvider_EnviarResumenBoletas(oDocumentoCierre.FechaCierre, ConfigurationManager.AppSettings["Empresa"], "CIERRE DOCUMENTO");
        if (idReenvio == 0)
        {
            throw (new Exception());
        }

        SincronizadorSuntat oSincronizadorSuntat = new SincronizadorSuntat()
        {
            EntityState = EntityState.Added,
            TablaOrigen = "CIERRE DOCUMENTO",
            ServerOrigen = ConfigurationManager.AppSettings["Empresa"],
            IdRegistro = idReenvio,
        };

        oSincronizadorSuntat.Tag = "El Resumen Diario de Boletas y Notas relacionadas estan listo para ser reenviado a la Sunat";
        oWsDataImport.SincronizadorSuntatProvider_CrearSincronizadorSuntat(ref oSincronizadorSuntat);

        oDocumentoCierre.IdEstado = 5;
        oDocumentoCierre.IdUsuarioModificacion = oUsuario.IdUsuario;
        oDocumentoCierre.FechaModificacion = DateTime.Now;
        DataRepository.DocumentoCierreProvider.Save(oDocumentoCierre);

        return Json("Resumen Reenviado", JsonRequestBehavior.AllowGet);
    }
    catch (Exception ex)
    {
        return Json(ex, JsonRequestBehavior.AllowGet);
    }
}
    
```

RF20: El Sistema permitirá obtener el XML de envío de un comprobante de pago electrónico anulado anteriormente.

Figura 180: Vista Modulo Resumen Anulación – Operación Obtener XML Envió

Fuente: Elaboración propia

N° ticket	Descripción Anulación	Fecha Anulación	Usuario Creacion	Fecha Creacion	Usuario Modificacion	Fecha Modificacion	Estado		
201803093900...	ANULACION DE DOCUMENTO	22/12/2018	demos demos123	22/12/2018	demos demos123	22/12/2018	ACTIVO		
201903185215...	ANULACION DE DOCUMENTO	10/01/2019	demos demos123	10/01/2019	demos demos123	10/01/2019	ACTIVO		

Figura 181: Estructura de Código – Vista Modulo Resumen Anulación – Obtener XML Envió

Fuente: Elaboración propia

```
function RequestXml() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    var url = '@Url.Action("RequestXml")?id=' + oCollection.currentItem.IdDocumentoAnulacion;
    window.open(url, 'Visor Xml');
};
```

Figura 182: Estructura de Código – Controlador Modulo Resumen Anulación – Obtener XML Envió

Fuente: Elaboración propia

```
public ActionResult RequestXml(long id)
{
    DocumentoAnulacion oDocumentoAnulacion = DataRepository.DocumentoAnulacionProvider.GetByIdDocumentoAnulacion(id);
    return this.Content(oDocumentoAnulacion.XmlEnvioSunat, "text/xml");
}
```

RF21: El Sistema permitirá obtener XML de respuesta de un comprobante de pago electrónico anulado anteriormente.

Figura 183: Vista Modulo Resumen Anulaciones – Operación Obtener XML Respuesta

Fuente:
Elaboración propia

N° ticket	Descripción Anulación	Fecha Anulación	Usuario Creacion	Fecha Creacion	Usuario Modificacion	Fecha Modificacion	Estado		
201803093900...	ANULACION DE DOCUMENTO	22/12/2018	demos demos123	22/12/2018	demos demos123	22/12/2018	ACTIVO		
201903185215...	ANULACION DE DOCUMENTO	10/01/2019	demos demos123	10/01/2019	demos demos123	10/01/2019	ACTIVO		

Figura 184: Estructura de Código – Vista Modulo R:A – Obtener XML Respuesta

Fuente:
Elaboración propia

```
function ResponseXml() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    var url = '@Url.Action("ResponseXml")?id=' + oCollection.currentItem.IdDocumentoAnulacion;
    window.open(url, 'Visor Xml');
};
```

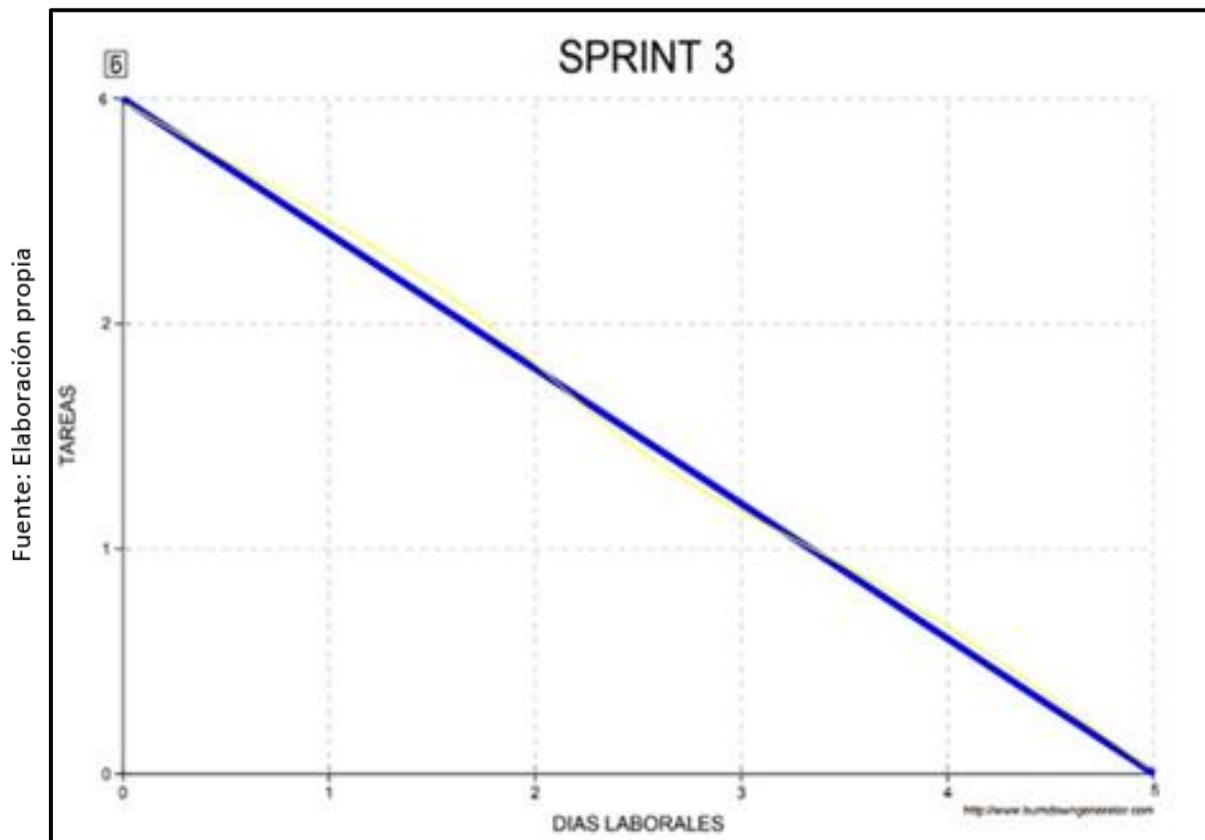
Figura 185: Estructura de Código – Controlador Modulo R:A – Obtener XML Respuesta

Fuente:
Elaboración propia

```
public ActionResult ResponseXml(long id)
{
    DocumentoAnulacion oDocumentoAnulacion = DataRepository.DocumentoAnulacionProvider.GetByIdDocumentoAnulacion(id);
    return this.Content(oDocumentoAnulacion.XmlRespuestaSunat, "text/xml");
}
```

GRÁFICA DE BURN DOWN – SPRINT 3

Figura 186: Burn Down del Sprint N° 3



ACTA DE PLANIFICACION SPRINT N° 3

Figura 187: Acta de Planificación del Sprint N°3

Fuente: Elaboración propia

ACTA DE PLANIFICACION DEL SPRINT N° 3

La reunión se lleva a cabo en presencia de los siguientes participantes:

N°	Nombres y Apellidos	Rol
1	Sergio Moran Palacios	Product Owner
2	Ronnie Alarcon Morales	Scrum Master
3	Orvic Yampool Flores Principe	Team Member

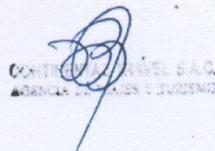
El gerente de la empresa Continental Travel S.A.C. realizó la exposición de los requerimientos e indica los requerimientos con mayor prioridad.

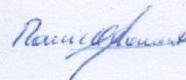
CODIGO	TAREA
RF09	El sistema permitirá generar un XML y enviar un comprobante de pago electrónico a la SUNAT.
RF10	El sistema permitirá recepcionar el XML de respuesta por parte de la SUNAT e interpretar su estado actual.
RF13	El sistema permitirá reenviar un XML de un comprobante de pago electrónico que haya presentado inconvenientes en su envío a la SUNAT.
RF14	El sistema permitirá anular los comprobantes de pago electrónicos emitidos, y enviar un XML de anulación a la SUNAT.
RF16	El sistema permitirá generar un XML y enviar un nuevo cierre diario de los comprobantes de pago relacionados a la Boleta, para sus posterior registro en la SUNAT.
RF17	El sistema permitirá obtener un XML de respuesta del cierre diario de los comprobantes de pago electrónicos relacionados a la Boleta, para sus posterior validación.
RF18	El sistema permitirá reenviar el XML de un cierre diario de los comprobantes de pago electrónicos relacionados a la Boleta, para sus posterior registro en la SUNAT.
RF20	El Sistema permitirá obtener el XML de envío de un comprobante de pago electrónico anulado anteriormente.
RF21	El Sistema permitirá obtener XML de respuesta de un comprobante de pago electrónico anulado anteriormente.

Mediante la presente acta se válida y se da conformidad los objetivos a realizarse en el Sprint 3 por parte del equipo de desarrollo, como también, las tareas propuestas del producto (Historias de Usuarios) que contiene el sprint mencionado.

Los asistentes impartirán su aprobación de acuerdo a lo presentado en la planificación del Sprint 0, indicando que la fecha de entrega de este Sprint sería el día 25 de Septiembre del 2019.

Lima, 13 de Septiembre del 2019.

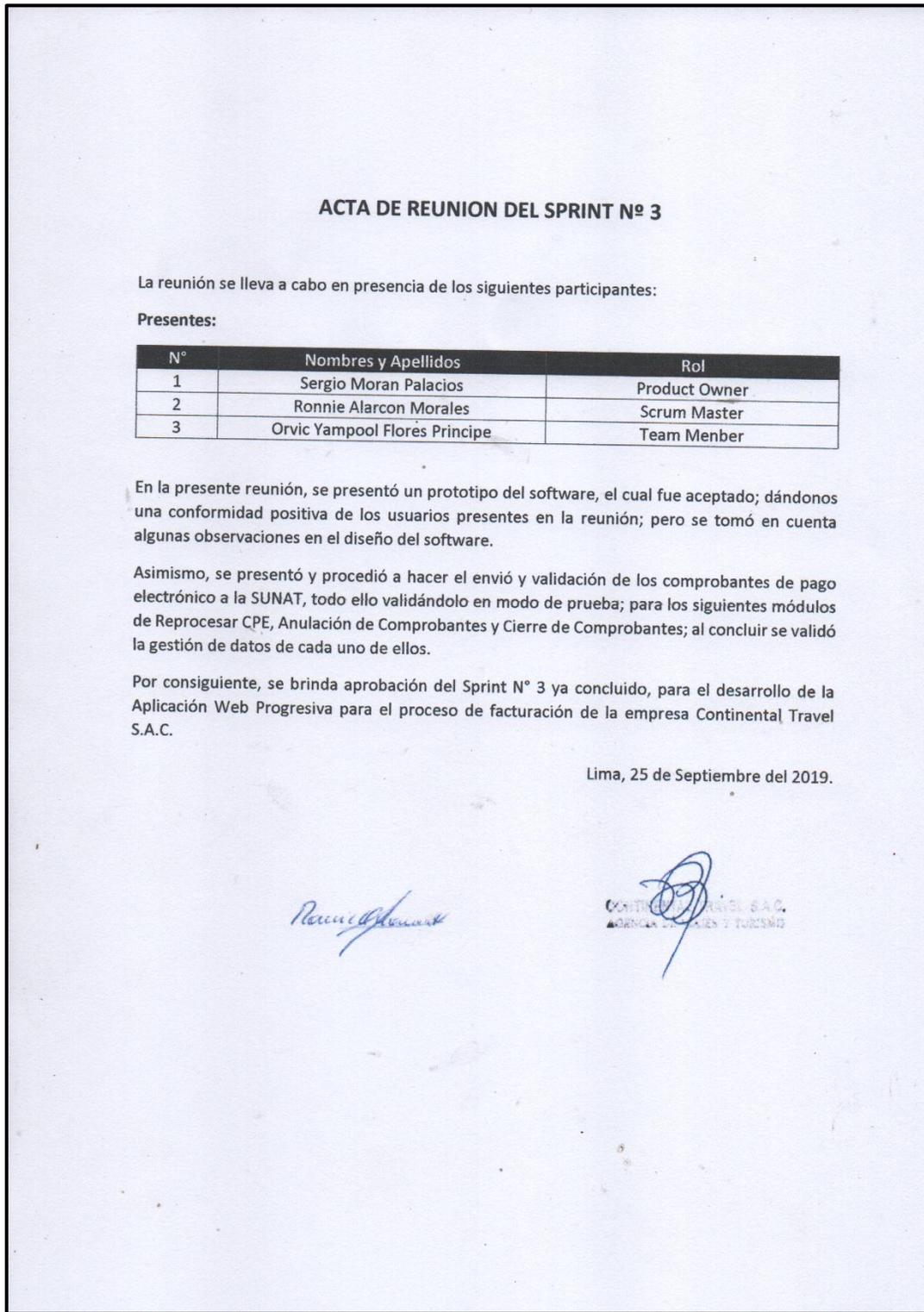

CONTINENTAL TRAVEL S.A.C.
AGENCIA DE VIAJES Y TURISMO



ACTA DE REUNION SPRINT N° 3

Figura 188: Acta de Reunión del Sprint N° 3

Fuente: Elaboración propia



3.5. Desarrollo del Sprint 4

Tabla 47: Sprint 4 – Módulo de Reportes

SPRINT	CODIGO	TAREA	ESTIMACIÓN (DIAS)	PRIORIDAD
SPRINT 4	RF11	El Sistema permitirá generar un reporte individual por cada comprobante de pago electrónico generado.	4	ALTA
	RF26	El sistema permitirá generar reportes mensuales de los comprobantes de pago electrónicos anulados.	4	ALTA
	RF27	El sistema permitirá generar reportes mensuales de los cierres diarios generados de acuerdo comprobantes de pago electrónicos emitidos durante el día.	4	ALTA
	RF31	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Cliente.	4	ALTA
	RF32	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Producto.	4	ALTA
	RF33	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Tipo Cambio.	4	ALTA
	RF34	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Moneda.	4	ALTA
	RF35	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Usuario.	4	ALTA
	RF36	El sistema permitirá generar reportes en base al indicador Nivel de incidencias en emisiones.	4	ALTA
RF37	El sistema permitirá generar reportes en base al Margen de Utilidad Operativa.	4	ALTA	

Fuente: Elaboración propia

RF11. El Sistema permitirá generar un reporte individual por cada comprobante de pago electrónico generado.

Figura 189: Vista Modulo Comprobantes de Pago – Operación Reporte

Fuente: Elaboración propia

Tipo	Serie	Número	Fecha EM.	RazonSocial	Mon	Neto	IGV	Otros	Dscto	Total	Estado Sunat
01	F001	0000388	28/03/2019	ASOCIACION DE PRODUCTORES...	USD	40.17	7.23	0.00	0.00	47.40	PENDIENTE
36	001	0028115	28/03/2019	ASOCIACION DE PRODUCTORES...	USD	245.58	0.00	47.02	0.00	292.60	PENDIENTE
01	F001	0000389	29/03/2019	REAL TOURS TOKYO	USD	3.22	0.58	0.00	0.00	3.80	PENDIENTE
03	B001	0000790	29/03/2019	KRISTIAN GARCIA MEZA	USD	50.85	9.15	0.00	0.00	60.00	PENDIENTE
03	B001	0000792	29/03/2019	EHEVERRIA VILMA	USD	86.26	15.53	0.00	0.00	101.79	PENDIENTE
36	001	0028118	29/03/2019	EHEVERRIA VILMA	USD	1,034.00	186.12	78.09	0.00	1,298.21	PENDIENTE
36	001	0028119	07/04/2019	PINEDA JAVIER	USD	442.00	79.56	61.31	0.00	582.87	PENDIENTE

Figura 190: Vista Modulo Comprobantes de Pago – Reporte

Fuente: Elaboración propia

CONTINENTAL TRAVEL S.A.C.
 Cal. Marcano de los Gerres No. 108 Int. 204. URB. SANTA ANA - SAN SEBASTIÁN - LIMA
 Central Telefónica : (01) 415-5071
 E-mail : facturacion@continentaltravel.com.pe
 http://www.continentaltravel.com.pe/

RUC N° 20267781151
FACTURA ELECTRÓNICA
F001 - [REDACTED]

Nombre / Razon social : [REDACTED] Moneda : USD
 Dirección : [REDACTED] Fecha Emisión : 28/03/2019
 RUC / DOC : [REDACTED] Fecha Vencimiento : 28/03/2019
 N° File : 1703814
 Vendedor : [REDACTED]

Descripción	Total
CAROSOS SERV. EMISION TKT N° 006 - 3624040481 PAX: [REDACTED]	47.40

SON : CUARENTA Y SIETE Y 40/100 DOLARES AMERICANOS
 Cancelar antes del: 28/03/2019
 Representación impresa de la Factura Electrónica.
 Puede ser consultada en : <http://www.continentaltravel.com.pe/>

Total Gravado	USD	40.17
Total No Gravado	USD	0.00
Total Exonerado	USD	0.00
Total Anticipo	USD	0.00
Total I.G.V.	USD	7.23
Otros Impuesto	USD	0.00
Descuento	USD	0.00
Importe Total	USD	47.40

NO SE ACEPTAN CAMBIOS NI DEVOLUCIONES
 El incumplimiento de pago al vencimiento, se devengarán los intereses monetarios y compensatorios que se computarán con las tasas máximas permitidas por la ley, en caso sea cancelable en efectivo en nuestras ventanillas o cheque girado nombre de CONTINENTAL TRAVEL S.A.C.

CONTINENTAL TRAVEL S.A.C.

Figura 191: Estructura de Código – Vista Modulo Comprobantes de Pago – Reporte

Fuente: Elaboración propia

```
function PrintRow() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    var url = '@Url.Action("Report", "Documentos")?id=' + oCollection.currentItem.IdDocumento;
    window.open(url, 'Reporte Cliente')
};
```

Figura 192: Estructura de Código – Controlador Comprobantes de Pago – Reporte

Fuente: Elaboración propia

```
public ActionResult Report(long id)
{
    List<ReportParameter> oReportParameters = new List<ReportParameter>();
    DBUblnet oDBUblnet = new DBUblnet();
    Documento oDocumento = DataRepository.DocumentoProvider.GetByIdDocumento(id);
    DataRepository.DocumentoProvider.DeepLoad(oDocumento, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Empresa), typeof(EstadoSunat) });

    Generamos la Imagen QR

    Carga el logo de la empresa con la que se logeará

    #region Carga de Parametro del Reporte
    //Parametros de la empresa
    string Direccion = oDocumento.IdEmpresaSource.DireccionEmpresa;
    string Telefono = ConfigurationManager.AppSettings["telefono"].ToString();
    string Fax = ConfigurationManager.AppSettings["fax"].ToString();
    string Url = ConfigurationManager.AppSettings["web"].ToString();

    //Parametros del reporte
    oReportParameters.Add(new ReportParameter("cDireccion", oDocumento.IdEmpresaSource.DireccionEmpresa.ToString()));
    oReportParameters.Add(new ReportParameter("cTelefono", Telefono));
    oReportParameters.Add(new ReportParameter("cFax", Fax));
    oReportParameters.Add(new ReportParameter("cUrl", Url));
    oReportParameters.Add(new ReportParameter("cEmpresa", DataRepository.EmpresaProvider.GetAll().First().RazonSocial));
    #endregion

    Validamos el Tipo de Reporte

    this.AddNotification("No se puede mostrar el formato virtual de este documento", NotificationType.ERROR);
    return View("~/Views/Shared/DocumentsViewers/ViewFormatoInvalido.cshtml");
}
```

RF26: El sistema permitirá generar reportes mensuales de los comprobantes de pago electrónicos anulados.

Figura 193: Vista Comprobantes Anulados – Operación Reporte

Fuente: Elaboración propia

continentaltravel Inicio Nosotros Administración Facturación Reportes Contáctenos Bienvenido admin Cerrar sesión

Inicio > Reportes > Documentos Anulados

1 / 1 Inicio: 31/10/2018 Final: 18/12/2018 [Printer Icon]

Tipo	Serie	Número	Fecha EM.	RazonSocial	Mon	Neto	IGV	Otros	Dcto	Total	Estado Sunat
03	B001	0000452	21/11/2018	ARIAS CORONADO ALFREDO ENRIQUE	USD	4,334.92	293.08	0.00	0.00	4,628.00	ANULADO
03	B001	0000513	14/12/2018	CLAUDIO AIKO	USD	1,450.00	261.00	81.98	0.00	1,792.98	ANULADO
01	F001	0000241	18/12/2018	DESTINOS MUNDIALES S.A.C.	USD	461.74	83.11	0.00	0.00	544.85	ANULADO

Figura 194: Vista Modulo Comprobantes Anulados – Reporte

Fuente:
Elaboración propia

REPORTE DE DOCUMENTOS ANULADOS					
TIPO DOCUMENTO	RAZON SOCIAL / CLIENTE	RUC / DNI	DOCUMENTO	FECHA	MONTO
FACTURA	DESTINOS MUNDIALES S.A.C.	20109796841	F001-0000241	18/12/2018	USD 544.8500
BOLETA	ARIAS CORONADO ALFREDO ENRIQUE	06109603	B001-0000452	21/11/2018	USD 4628.0019
	CLAUDIO AIKO	118205904	B001-0000513	14/12/2018	USD 1792.9800

Figura 195: Estructura de Código – Vista Modulo C: Anulados – Reporte Parte 1

Fuente:
Elaboración propia

```
<button id="btnExport"
    onclick="btnExport()"
    class="btn btn-default" title="Descargar Reporte">
    <span class="glyphicon glyphicon-print"></span>
</button>
```

Figura 196: Estructura de Código – Vista Modulo C: Anulados – Reporte Parte 2

Fuente:
Elaboración propia

```
function btnExport() {
    window.location.href = '@Url.Action("Anulados", "ReporteDocumento")?FechaInicio='
    + encodeURIComponent(formatDate(fechainicio.value)) + "&FechaFin="
    + encodeURIComponent(formatDate(fechafinal.value))
};
```

Figura 197: Estructura de Código – Controlador Modulo C: Anulados – Reporte

Fuente: Elaboración propia

```
public ActionResult Anulados(DateTime FechaInicio, DateTime FechaFin)
{
    try {
        DataSet oDataSet = DataRepository.DocumentoProvider.Formato_Informe(FechaInicio, FechaFin);
        if (oDataSet.Tables[0].Rows.Count != 0)
        {
            string iddocumento = oDataSet.Tables[0].Rows[0][0].ToString();
            Documento oDocumento = DataRepository.DocumentoProvider.GetByIdDocumento(Convert.ToInt32(iddocumento));
            DataRepository.DocumentoProvider.DeepLoad(oDocumento, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Empresa) });

            List<ReportParameter> oReportParameters = new List<ReportParameter>();
            System.Drawing.Image oLogImage = System.Drawing.Image.FromFile(string.Format("{0}\\{1}.png", FilesHelper.RutaLogo, oDocumento.IdEmpresaSource.RucEmpresa.Trim()));
            string BaseLogo64 = string.Empty;
            using (MemoryStream lms = new MemoryStream())
            {
                oLogImage.Save(lms, System.Drawing.Imaging.ImageFormat.Png);
                byte[] logoBytes = lms.ToArray();
                BaseLogo64 = Convert.ToBase64String(logoBytes);
                oReportParameters.Add(new ReportParameter("cLogo", BaseLogo64.ToString()));
            }

            ReportDataSource oReportDataSource = new ReportDataSource("Cust_Documento_Formato_Factura", oDataSet.Tables[0]);
            ReportViewer oNewReportView = new ReportViewer();
            oNewReportView.ProcessingMode = ProcessingMode.Local;
            oNewReportView.LocalReport.ReportPath = string.Format("{0}\\{1}.rdl", FilesHelper.RutaFormato, "RptFormatoAnulado");
            oNewReportView.LocalReport.DataSources.Add(oReportDataSource);
            oNewReportView.LocalReport.SetParameters(oReportParameters);

            string pDeviceInfo = string.Empty; Warning[] warnings; string[] streamIds;
            string mimeType = string.Empty; string encoding = string.Empty; string extension = string.Empty;
            byte[] bytes = oNewReportView.LocalReport.Render("PDF", pDeviceInfo, out mimeType, out encoding, out extension, out streamIds, out warnings);

            Response.Clear();
            Response.ClearContent();
            Response.ClearHeaders();
            Response.ContentType = "application/pdf";
            Response.AddHeader("content-disposition", "attachment; filename=" + "Reporte de documentos anulados" + "." + extension);
            Response.AddHeader("Content-Length", bytes.Length.ToString());
            Response.BinaryWrite(bytes);
            Response.AddHeader("Accept-Ranges", "ascii");
            Response.AddHeader("Cache-Control", "post-check=0, pre-check=0");
            Response.AddHeader("Cache-Control", "no-store, no-cache, must-revalidate");
            Response.AddHeader("Pragma", "no-cache");
            Response.Flush();
            Response.End();

            return View();
        }
        else
        {
            this.AddNotification("No ha encontrado ningún documento anulado en el rango de fechas que especifico", NotificationType.WARNING);
            return RedirectToAction("DocumentoAnulado");
        }
    }
    catch
    {
        this.AddNotification("No ha encontrado ningún documento anulado en el rango de fechas que especifico", NotificationType.WARNING);
        return RedirectToAction("DocumentoAnulado");
    }
}
```

RF27: El sistema permitirá generar reportes mensuales de los cierres diarios generados de acuerdo comprobantes de pago electrónicos emitidos durante el día.

Figura 198: Vista Modulo Cierre Comprobantes – Operación Reporte

Fuente:
Elaboración propia



Figura 199: Vista Modulo Cierre Comprobantes – Reporte

Fuente:
Elaboración propia

RESUMEN DIARIO DE BOLETAS DE VENTA					
TIPO DOCUMENTO	RAZON SOCIAL / CLIENTE	RUC / DNI	DOCUMENTO	FECHA	MONTO
BOLETA DE VENTA	NAITO YOSHIO	TR9735922	B001-0000679	07/02/2019	USD 42.54
	DE LA CRUZ JOSE	118018131	B001-0000680	07/02/2019	USD 82.13
	SANTA CRUZ PATRICIA	C016633	B001-0000681	07/02/2019	USD 97.28

Figura 200: Estructura de Código – Vista Modulo Cierre Comprobantes – Reporte

Fuente: Elaboración propia

```

@{
    ViewBag.Title = "Documentos";
    var fecini = DateTime.Now.Date;
}
<div class="container-fluid">
    <div class="row">
        <div class="col-md-3">
            <label>Fecha Cierre :</label>
            @(Html.C1().InputDate().Id("Fecha").Width(140).Value(fecini))
        </div>
        <div class="col-md-4">
            <button id="btnExport"
                onclick="btnExport()"
                class="btn btn-default">
                <span class="glyphicon glyphicon-list-alt"></span>
            </button>
        </div>
    </div>
</div>
</div>
@section Scripts {
    @Scripts.Render("~/bundles/jqueryval")
    @Scripts.Render("~/Scripts/script.js")
    <script type="text/javascript">
        var fecha
        function btnExport() {
            fecha = wijmo.Control.getControl("#Fecha");
            window.location.href = '@Url.Action("Cierre", "ReporteDocumento")?Fecha=' + encodeURIComponent(formatDate(fecha.value))
        };
    </script>
}
    
```

Figura 201: Estructura de Código – Controlador Modulo Cierre Comprobantes – Reporte

Fuente: Elaboración propia

```

public ActionResult Cierre(DateTime Fecha)
{
    DataSet oDataSet = DataRepository.DocumentoProvider.Formato_Cierre(Fecha);
    if (oDataSet.Tables[0].Rows.Count == 0)
    {
        this.AddNotification("No ha encontrado ningún documento cierre en el rango de fechas que especifico", NotificationType.WARNING);
        return RedirectToAction("DocumentoCierre");
    }
    else
    {
        string iddocumento = oDataSet.Tables[0].Rows[0][0].ToString();
        Documento oDocumento = DataRepository.DocumentoProvider.GetByIdDocumento(Convert.ToInt32(iddocumento));
        DataRepository.DocumentoProvider.DeepLoad(oDocumento, true, DeepLoadType.IncludeChildren, new Type[] { typeof(Empresa) });

        List<ReportParameter> oReportParameters = new List<ReportParameter>();
        System.Drawing.Image oLogImage = System.Drawing.Image.FromFile(string.Format("{0}\\{1}.png", FilesHelper.RutaLogo, oDocumento.IdEmpresaSource.RucEmpresa.Trim()));
        string BaseLogo64 = string.Empty;
        using (MemoryStream lms = new MemoryStream())
        {
            oLogImage.Save(lms, System.Drawing.Imaging.ImageFormat.Png);
            byte[] logoBytes = lms.ToArray();
            BaseLogo64 = Convert.ToBase64String(logoBytes);
            oReportParameters.Add(new ReportParameter("cLogo", BaseLogo64.ToString()));
        }

        ReportDataSource oReportDataSource = new ReportDataSource("Cust_Documento_Formato_Factura", oDataSet.Tables[0]);
        ReportViewer oNewReportView = new ReportViewer();
        oNewReportView.ProcessingMode = ProcessingMode.Local;
        oNewReportView.LocalReport.ReportPath = string.Format("{0}\\{1}.rdl", FilesHelper.RutaFormato, "RptFormatoCierre");

        oNewReportView.LocalReport.DataSources.Add(oReportDataSource);
        oNewReportView.LocalReport.SetParameters(oReportParameters);

        string pDeviceInfo = string.Empty;
        Warning[] warnings; string[] streamIds;
        string mimeType = string.Empty; string encoding = string.Empty; string extension = string.Empty;
        byte[] bytes = oNewReportView.LocalReport.Render("PDF", pDeviceInfo, out encoding, out extension, out streamIds, out warnings);

        Response.Clear();
        Response.ClearContent();
        Response.ClearHeaders();
        Response.ContentType = "application/pdf";
        Response.AddHeader("content-disposition", "attachment; filename= " + "Reporte de documentos cierre" + "." + extension);
        Response.AddHeader("Content-Length", bytes.Length.ToString());
        Response.BinaryWrite(bytes);
        Response.AddHeader("Accept-Ranges", "ascii");
        Response.AddHeader("Cache-Control", "post-check=0, pre-check=0");
        Response.AddHeader("Cache-Control", "no-store, no-cache, must-revalidate");
        Response.AddHeader("Pragma", "no-cache");
        Response.Flush();
        Response.End();
        return View();
    }
}

```

RF31: El sistema permitirá generar, descargar e imprimir reportes en el Módulo Cliente.

Figura 202: Vista Modulo Cliente – Operación Reporte

Fuente:
Elaboración propia

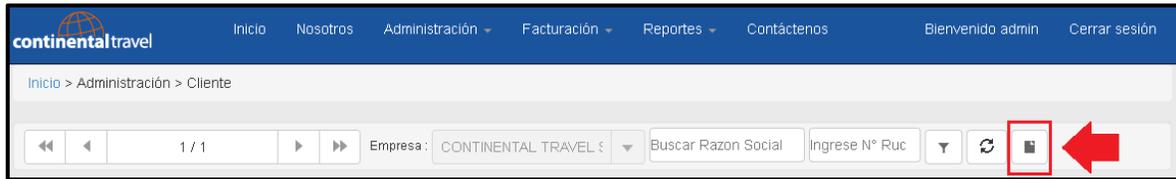


Figura 203: Vista Modulo Clientes – Reporte

Fuente: Elaboración propia

The screenshot shows the 'Reporte de Clientes' page. At the top, there is a search bar with the text 'Buscar | Siguiete' and a dropdown menu with options 'Excel', 'PDF', and 'Word'. Below the search bar, the title 'REPORTE DE CLIENTES' is displayed. The main content is a table with the following columns: Nombre, Dirección, RUC, and Telefono. The table contains 15 rows of client data.

Nombre	Dirección	RUC	Telefono
ROSTER S.A.	JR. MANUEL IRIBARREN NRO. 1325 (ALTURA	20100812542	
LOPEZ CLARA	LIMA	17907416	
MORIYA YOHEI	LIMA	70984164	
AG CORP	AV. JOSE PARDO NRO. 805 DPTO. 301 LIMA	20601272611	
AEROMEXICO	AV. BENAVIDES 1579 INT. 405 MIRAFLORES,	20307246741	
MIYAKAWA RANDY	LIMA	16696040	
SURITA WILSON	LIMA	40599588	
TSUCHIKAME AKEMI	LIMA	116085106	
UMEZAKI YORIKO	LIMA	TS1094197	
TANAKA TAKAHIRO	LIMA	TZ1277271	
CAMONES ELSA	LIMA	117150507	
SANCHEZ AURELIO	LIMA	116163107	
LUIS PUERTAS HURTADO	CA. 17(PETIROJOS) 408, CORPAC, SAN ISIDR	08236824	
BRYAN EYZAGUIRRE LIN	JR. SAN SALVADOR123, URB. HUAQUIYAY, COM	43930436	
PAREDES GUSHIKEN WML	CALLE GENOVESA MZ A2 LOTE 39 LOS CEDROS	6905303	941729355

Figura 204: Estructura de Código – Vista Modulo Clientes – Reporte

Fuente:
Elaboración propia

```
function reportRow() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    var url = '@Url.Action("Report", "Clientes")?IdEmpresa='
        + encodeURIComponent(cboEmpresa.selectedValue) + "&RazonSocial="
        + encodeURIComponent(txtRazonSocial.val()) + "&NumeroDocumentoIdentidad="
        + encodeURIComponent(txtRuc.val()) + "&PageInitial="
        + oCollection.pageIndex;
    window.open(url, 'Reporte Cliente');
};
```

Figura 205: Estructura de Código – Controlador Modulo Cliente – Reporte

Fuente:
Elaboración propia

```

#region Reporte
<small>Referencias</small>
public ActionResult Report(long IdEmpresa,string RazonSocial,string NumeroDocumentoIdentidad, int PageInitial)
{
    int nCount = 0;
    object[] cparams = { IdEmpresa,RazonSocial,NumeroDocumentoIdentidad };
    string cWhere = string.Format("( {0} = -1 OR IdEmpresa = {0}) AND RazonSocial like '{1}%' AND NumeroDocumentoIdentidad like '{2}%", cparams);
    TList<Cliente> oClientes = DataRepository.ClienteProvider.GetPaged(cWhere, "", 1, PageInitial, out nCount);
    return View(oClientes.ToList());
}
#endregion
    
```

RF32: El sistema permitirá generar, descargar e imprimir reportes en el Módulo Producto.

Figura 206: Vista Modulo Productos – Operación Reporte

Fuente:
Elaboración propia

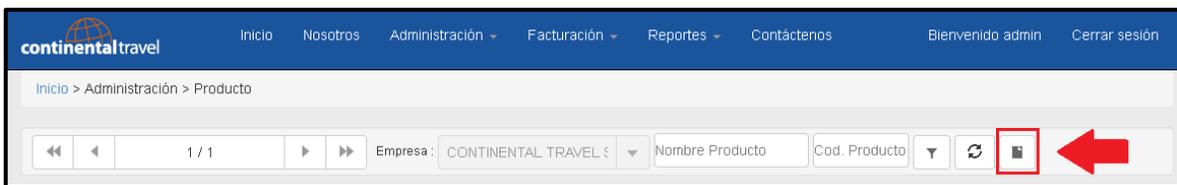


Figura 207: Vista Modulo Productos – Reporte

Fuente: Elaboración propia

The screenshot shows the 'Reporte de Productos' page. At the top, there is a search bar with '1 de 1' and a 'Buscar | Siguiete' button. A dropdown menu is open, showing options for 'Excel', 'PDF', and 'Word'. Below the menu, the title 'REPORTE DE PRODUCTOS' is displayed. The main content is a table with the following data:

Nombre	Codigo	Descripción	Costo Unitario
SERVICE FEE	SF	SERVICE FEE	0.0
TICKETS AEREOS	TK	TICKETS AEREOS	0.0
ASESORIA VISA JAPON	AJ	ASESORIA VISA JAPON	0.0

Figura 208: Estructura de Código – Vista Modulo Clientes – Reporte

Fuente:
Elaboración propia

```

function reportRow() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    var url = 'Url.Action("Report", "Productos"?IdEmpresa='
        + encodeURIComponent(cboEmpresa.selectedValue) + "&NombreProducto="
        + encodeURIComponent(txtNombreProducto.val()) + "&CodigoProducto="
        + encodeURIComponent(txtCodigo.val()) + "&PageInitial="
        + oCollection.pageIndex;
    window.open(url, 'Reporte Producto');
};
    
```

Figura 209: Estructura de Código – Controlador Modulo Productos – Reporte

Fuente: Elaboración propia

```
#region Reporte
<!--referencias-->
public ActionResult Report(long IdEmpresa, string NombreProducto, string CodigoProducto, int PageInitial)
{
    int nCount = 0;
    object[] cparams = { IdEmpresa, NombreProducto, CodigoProducto};
    string clwhere = string.Format(" ({0} = -1 OR IdEmpresa = {0}) AND NombreProducto like '{1}%' AND CodigoProducto like '{2}%' ", cparams);
    TList<Producto> oProducto = DataRepository.ProductoProvider.GetPaged(clwhere, "", 1, PageInitial, out nCount);
    return View(oProducto.ToList());
}
#endregion
```

RF33: El sistema permitirá generar, descargar e imprimir reportes en el Módulo Tipo Cambio.

Figura 210: Vista Modulo Tipo Cambio – Operación Reporte

Fuente: Elaboración propia

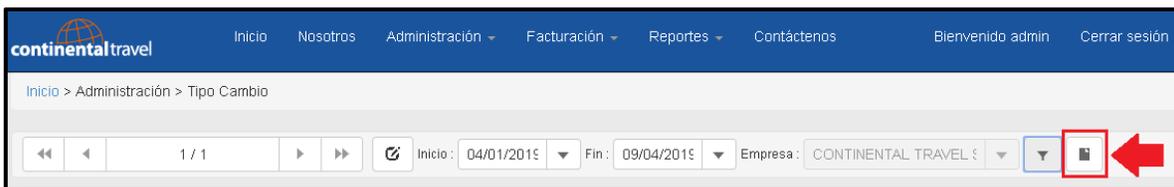


Figura 211: Vista Modulo Tipo Cambio - Reporte

Fuente: Elaboración propia

1 de 1 | Buscar | Siguiente

Excel | PDF | Word

1 of 1
08/04/2019

Fecha Cambio	Monto Cambio
08/03/2019	3,37
09/03/2019	3,37
11/03/2019	3,37
12/03/2019	3,37
13/03/2019	3,37
14/03/2019	3,37
15/03/2019	3,37
16/03/2019	3,37
17/03/2019	3,37
18/03/2019	3,37

Figura 212: Estructura de Código – Vista Modulo Tipo Cambio - Reporte

Fuente: Elaboración propia

```
function reportRow() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    var url = '@Url.Action("Report", "TipoCambios")?FechaInicio='
        + encodeURIComponent(formatDate(fechainicio.value)) + "&FechaFinal="
        + encodeURIComponent(formatDate(fechainicio.value)) + "&IdEmpresa="
        + encodeURIComponent(cboEmpresa.selectedValue) + "&PageInitial="
        + oCollection.pageIndex;
    window.open(url, 'Reporte Empresa');
};
```

Figura 213: Estructura de Código – Controlador Modulo Tipo Cambio - Reporte

Fuente: Elaboración propia

```
#region Reporte
@references
public ActionResult Report(DateTime FechaInicio, DateTime FechaFinal, long IdEmpresa, int PageInitial)
{
    int nCount = 0;
    object[] cparams;
    if (ConfigurationManager.AppSettings["Region"].ToString().Equals("US"))
    {
        object[] cparamsTipo = { FechaInicio.ToString("yyyy/MM/dd"), FechaFinal.ToString("yyyy/MM/dd"), IdEmpresa };
        cparams = cparamsTipo;
    }
    else
    {
        object[] cparamsTipo = { FechaInicio.ToString("dd/MM/yyyy"), FechaFinal.ToString("dd/MM/yyyy"), IdEmpresa };
        cparams = cparamsTipo;
    }
    string cwhere = string.Format("[FechaCambio] BETWEEN '{0}' AND '{1}' AND (({2} = -1 OR IdEmpresa = {2})", cparams);
    TList<TipoCambio> oTipoCambio = DataRepository.TipoCambioProvider.GetPaged(cwhere, "", 1, PageInitial, out nCount);
    return View(oTipoCambio.ToList());
}
#endregion
```

RF34: El sistema permitirá generar, descargar e imprimir reportes en el Módulo Moneda.

Figura 214: Vista Modulo Moneda – Operación Reporte

Fuente: Elaboración propia

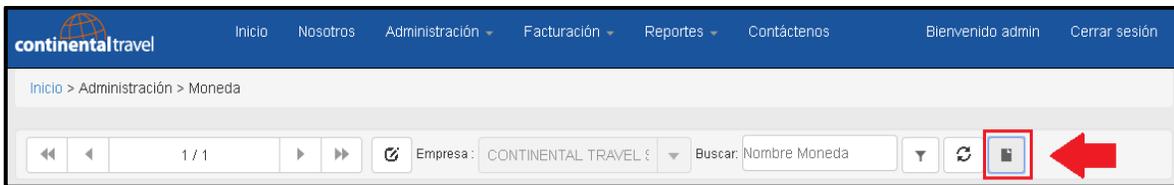


Figura 215: Vista Modulo Moneda – Reporte

Fuente: Elaboración propia



Figura 216: Estructura de Código – Vista Modulo Moneda - Reporte

Fuente:
Elaboración propia

```
function reportRow() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    var url = '@Url.Action("Report", "TipoCambios")?FechaInicio='
    + encodeURIComponent(formatDate(fechainicio.value)) + "&FechaFinal="
    + encodeURIComponent(formatDate(fechafinal.value)) + "&IdEmpresa="
    + encodeURIComponent(cboEmpresa.selectedValue) + "&PageInitial="
    + oCollection.pageIndex;
    window.open(url, 'Reporte Empresa');
};
```

Figura 217: Estructura de Código – Controlador Modulo Moneda - Reporte

Fuente:
Elaboración propia

```
#region Reporte
//referencias
public ActionResult Report(string NombreMoneda, long IdEmpresa, int PageInitial)
{
    int nCount = 0;
    object[] cparams = { NombreMoneda, IdEmpresa };
    string cWhere = string.Format("[NombreMoneda] LIKE '{0}' AND ((1) = -1 OR IdEmpresa = {1})", cparams);
    TList<Moneda> oMoneda = DataRepository.MonedaProvider.GetPaged(cWhere, "", 1, PageInitial, out nCount);
    return View(oMoneda.ToList());
}
#endregion
```

RF35: El sistema permitirá generar, descargar e imprimir reportes en el Módulo Usuario:

Figura 218: Vista Modulo Usuario – Operación Reporte

Fuente:
Elaboración propia

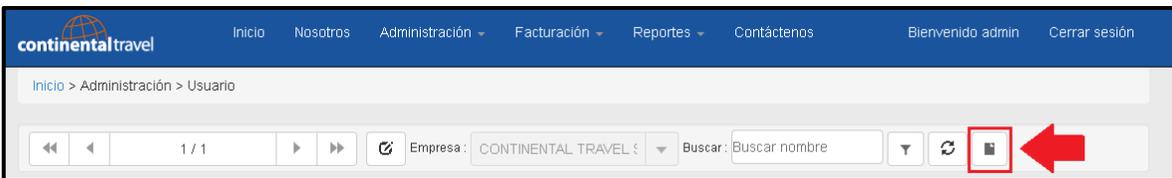


Figura 219: Vista Modulo Usuario – Reporte

Fuente: Elaboración propia

1 de 1 Buscar | Siguiente

Excel
PDF
Word

Fecha Impresión: 2019

Login	Nombre	Nombre Completo	Email	Telefono
admin	Admin	Admin CTravel	admin@ctravel.com.pe	014120012
0123456789	PERU TOURS TOKYO	PERU TOURS TOKYO	0123456789@ctravel.com.pe	012423423
118388942	ECHEVERRIA VILMA	ECHEVERRIA VILMA	118388942@ctravel.com.pe	013434202
216107953	PINEDA JAVIER	PINEDA JAVIER	216107953@ctravel.com.pe	012340343
20561174076	ASOCIACION DE PRODUCTORES INTEGRAL Y SOSTENIBLE DE	ASOCIACION DE PRODUCTORES INTEGRAL Y SOSTENIBLE DE	20561174076@ctravel.com.pe	012340022
40794257	KRISTIAN GARCIA	KRISTIAN GARCIA	40794257@ctravel.com.pe	013840145

Figura 220: Estructura de Código – Vista Modulo Usuario - Reporte

Fuente:
Elaboración propia

```
function reportRow() {
    if (!pagingGrid || !oCollection) {
        return;
    }
    pagingGrid.invalidate();
    var url = '@Url.Action("Report", "Usuarios")?Nombre='
        + encodeURIComponent(txtNombre.val()) + "&IdEmpresa="
        + encodeURIComponent(cboEmpresa.selectedValue) + "&PageInitial="
        + oCollection.pageIndex;
    window.open(url, 'Reporte Usuario');
};
```

Figura 221: Estructura de Código – Controlador Modulo Usuario – Reporte

Fuente:
Elaboración propia

```
#region Reporte
0 referencias
public ActionResult Report(string Nombre, long IdEmpresa, int PageInitial)
{
    int nCount = 0;
    string[] cparams = { Nombre.Trim(), IdEmpresa.ToString() };
    string cWhere = string.Format("[NombreCompleto] LIKE '{0}%' AND ((1) = -1 OR IdEmpresa = {1})", cparams);
    TList<Usuario> oUsuario = DataRepository.UsuarioProvider.GetPaged(cWhere, "", PageInitial, 16, out nCount);
    return View(oUsuario.ToList());
}
#endregion
```

RF36: El sistema permitirá generar reportes en base al indicador Nivel de incidencias en emisiones.

Figura 222: Vista Modulo Resumen Contable – Operación NIE

Fuente: Elaboración propia

Figura 223: Vista Modulo Resumen Contable –Reporte NIE

Fuente: Elaboración propia

REPORTE DE NIVEL DE INCIDENCIAS EN EMISIONES


CONTINENTAL TRAVEL S.A.C.
 AV. CANAVAL Y MOREYRA NRO. 285 URB. SANTA ANA / LIMA - LIMA - SAN ISIDRO
 Central Telefónica / Telephone Central : 01 415 5050
 E-mail : facturacion@continentaltravel.com.pe
 http://www.continentaltravel.com.pe

NIE = 25.85 %
 1/04/2019 AL 30/04/2019

PROCESO DE FACTURACION / BILLING PROCESS			
Fecha / Date	Emisidos / Issued	Anulados / Voided	NIE
1 de Abr.	18	5	27.78
2 de Abr.	7	2	28.57
3 de Abr.	6	1	16.67
4 de Abr.	7	2	28.57
5 de Abr.	10	5	31.25
6 de Abr.	4	2	50.00
7 de Abr.	0	0	(-)
8 de Abr.	1	0	(-)
9 de Abr.	9	2	22.22
10 de Abr.	2	1	50.00
11 de Abr.	10	2	20.00
12 de Abr.	3	1	33.33
13 de Abr.	3	1	33.33
14 de Abr.	0	0	(-)
15 de Abr.	0	0	(-)
16 de Abr.	4	1	25.00
17 de Abr.	8	2	25.00
18 de Abr.	0	0	(-)
19 de Abr.	0	0	(-)
20 de Abr.	0	0	(-)
21 de Abr.	0	0	(-)
22 de Abr.	19	4	21.05
23 de Abr.	0	0	(-)
24 de Abr.	11	2	18.18
25 de Abr.	1	0	(-)
26 de Abr.	2	1	50.00
27 de Abr.	6	2	33.33
28 de Abr.	0	0	(-)
29 de Abr.	10	2	20.00
30 de Abr.	0	0	(-)

Comprobantes Emisidos / Issued Vouchers **147.00**

Comprobantes Anulados / Canceled Vouchers **38.00**

CONTINENTAL TRAVEL S.A.C.

Representación impresa del Indicador de Nivel de Incidencias en Emisiones / Printed representation of the Level of Incidents in Emissions Indicator
 Puede ser consultada en / It can be consulted at: <http://www.continentaltravel.com.pe>

RF37: El sistema permitirá generar reportes en base al Margen de Utilidad Operativa.

Figura 226: Vista Modulo Resumen Contable –Operación MUO

Fuente: Elaboración propia

Figura 227: Vista Modulo Resumen Contable – Reporte MUO

Fuente: Elaboración propia

REPORTE DE MARGEN DE UTILIDAD OPERATIVA


CONTINENTAL TRAVELS A.C.
 AV. CANAVAL Y MOREYRA NRO. 285 URB. SANTA ANA / LIMA -
 LIMA - SAN ISIDRO
 Central Telefónica / Telephone Central : 01 415 5050
 E-mail : facturacion@continentaltravel.com.pe
 http://www.continentaltravel.com.pe

MUO = 0.56

1/04/2019 AL 30/04/2019

PROCESO DE FACTURACION / BILLING PROCESS

Fecha / Date	Utilidad Operativa / Operating Income	Ingresos / Income	MUO
1 de Abr.	2318.32	6076.79	0.38
2 de Abr.	142.14	3038.66	0.05
3 de Abr.	240.83	1729.17	0.14
4 de Abr.	451.14	2986.74	0.15
5 de Abr.	3161.31	18703.69	0.17
6 de Abr.	168.45	1771.05	0.10
7 de Abr.	0.00	0.00	(-)
8 de Abr.	0.00	0.00	(-)
9 de Abr.	408.34	3617.13	0.11
10 de Abr.	74.97	497.74	0.15
11 de Abr.	1208.41	6851.08	0.18
12 de Abr.	220.00	1645.00	0.13
13 de Abr.	187.57	822.43	0.23
14 de Abr.	0.00	0.00	(-)
15 de Abr.	0.00	0.00	(-)
16 de Abr.	171.88	244.00	0.70
17 de Abr.	449.35	3970.90	0.11
18 de Abr.	0.00	0.00	(-)
19 de Abr.	0.00	0.00	(-)
20 de Abr.	0.00	0.00	(-)
21 de Abr.	0.00	0.00	(-)
22 de Abr.	1490.75	15069.58	0.10
23 de Abr.	0.00	0.00	(-)
24 de Abr.	987.89	5960.70	0.17
25 de Abr.	0.00	0.00	(-)
26 de Abr.	0.00	0.00	Error
27 de Abr.	430.60	5489.40	0.07
28 de Abr.	0.00	0.00	(-)
29 de Abr.	764.17	4905.78	0.15
30 de Abr.	0.00	0.00	(-)
Total de Ventas / Total sales		110,016.39	
Utilidad Bruto / Gross Profit		23,327.56	

CONTINENTAL TRAVEL S.A.C.

Representación impresa del Indicador de Margen de Utilidad Operativa / Printed representation of the Operating Profit Margin Indicator
 Puede ser consultada en / It can be consulted at: <http://www.continentaltravel.com.pe>

Figura 228: Estructura de Código – Vista Modulo Resumen Contable - MUO

Fuente: Elaboración propia

```

using JSUBinet.WebUBI.Net.Models;
@model DocumentoListaViewModel
@{
    ViewBag.Title = "Reporte Anulados";
    var fecini = DateTime.Now.AddMonths(-1).Date;
    var fecfin = DateTime.Now.Date;
}
<div class="panel panel-default">
<div class="panel-body">
<div class="row">
<div class="col-md-5">
<label>Inicio :</label>
@Html.CI().InputDate().Id("FechaInicio").Width(140).Value(fecini)
<label>Final :</label>
@Html.CI().InputDate().Id("FechaFinal").Width(140).Value(fecfin)
</div>
<div class="col-md-4"></div>
<div class="col-md-3" style="text-align : right">
<button id="btnExport" class="btn btn-default">...</button>
<button id="btnExportMUO" onclick="btnExportMUO()"
class="btn btn-default" title="Descargar Reporte MUO">
<span class="glyphicon glyphicon-signal"></span> <b>MUO</b>
</button>
</div>
</div><br>
<div class="row">...</div><br>
</div>
</div>
@section Scripts {
@Scripts.Render("~/bundles/jqueryval")
<script type="text/javascript">
var fechaInicio, fechaFinal;
cl.mvc.Utilis.documentReady(function () {
    fechaInicio = wijmo.Control.getControl("#FechaInicio");
    fechaFinal = wijmo.Control.getControl("#FechaFinal");
});
function btnExportNIE() {
    window.location.href = '@Url.Action("ReporteNIE", "ReporteDocumento")?FechaInicio='
+ encodeURIComponent(formatDate(fechaInicio.value)) + "&FechaFin="
+ encodeURIComponent(formatDate(fechaFinal.value))
};
function btnExportMUO() {
    window.location.href = '@Url.Action("ReporteMUO", "ReporteDocumento")?FechaInicio='
+ encodeURIComponent(formatDate(fechaInicio.value)) + "&FechaFin="
+ encodeURIComponent(formatDate(fechaFinal.value))
};
</script>
}
    
```

Figura 229: Estructura de Código – Controlador Modulo Resumen Contable – MUO

Fuente: Elaboración propia

```

public ActionResult ReporteMUO(DateTime FechaInicio, DateTime FechaFin)
{
    try
    {
        Empresa oEmpresa = DataRepository.EmpresaProvider.GetAll().First();

        List<ReportParameter> oReportParameters = new List<ReportParameter>();
        DataSet oDataSet = DataRepository.DocumentoProvider.Formato_MUO(FechaInicio.ToShortDateString(), FechaFin.ToShortDateString());

        if (oDataSet.Tables[0].Rows.Count != 0)
        {
            ReportDataSource oReportDataSource = new ReportDataSource("Cust_Documento_Formato_MUO", oDataSet.Tables[0]);

            Carga el logo de la empresa con la que se logean
            Carga de Parametro del Reporte

            ReportViewer oNewReportView = new ReportViewer();
            oNewReportView.ProcessingMode = ProcessingMode.Local;
            oNewReportView.LocalReport.ReportPath = string.Format("{0}\\{1}.rdl", FilesHelper.RutaFormato, "RptFormatoMUO");
            oNewReportView.LocalReport.DataSources.Add(oReportDataSource);
            oNewReportView.LocalReport.SetParameters(oReportParameters);

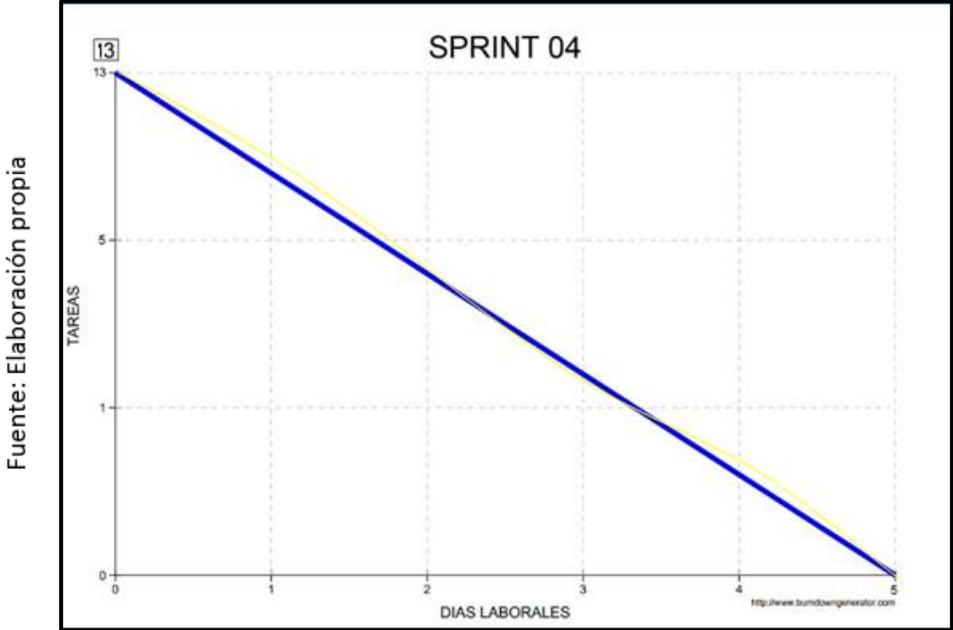
            string pDeviceInfo = string.Empty;
            Warning[] warnings; string[] streamIds;
            string mimeType = string.Empty; string encoding = string.Empty; string extension = string.Empty;
            byte[] bytes = oNewReportView.LocalReport.Render("PDF", pDeviceInfo, out mimeType, out encoding, out extension, out streamIds, out warnings);

            Carga PDF

            return View();
        }
        else
        {
            this.AddNotification("No ha encontrado ningun comprobante que procesar el rango de fechas que especifico", NotificationType.WARNING);
            return RedirectToAction("DocumentoIndicador");
        }
    }
    catch (Exception ex)
    {
        this.AddNotification(ex.Message, NotificationType.WARNING);
        return RedirectToAction("DocumentoIndicador");
    }
}
    
```

GRÁFICA DE BURN DOWN – SPRINT 4

Figura 230: Burn Down Sprint N° 4



ACTA DE PLANIFICACION SPRINT N° 4

Figura 231: Acta de Planificación del Sprint N° 4

Fuente: Elaboración propia

ACTA DE PLANIFICACION DEL SPRINT N° 4

La reunión se lleva a cabo en presencia de los siguientes participantes:

N°	Nombres y Apellidos	Rol
1	Sergio Moran Palacios	Product Owner
2	Ronnie Alarcon Morales	Scrum Master
3	Orvic Yampool Flores Principe	Team Member

El gerente de la empresa Continental Travel S.A.C. realizó la exposición de los requerimientos e indica los requerimientos con mayor prioridad.

CODIGO	TAREA
RF11	El Sistema permitirá generar un reporte individual por cada comprobante de pago electrónico generado.
RF26	El sistema permitirá generar reportes mensuales de los comprobantes de pago electrónicos anulados.
RF27	El sistema permitirá generar reportes mensuales de los cierres diarios generados de acuerdo comprobantes de pago electrónicos emitidos durante el día.
RF31	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Cliente.
RF32	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Producto.
RF33	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Tipo Cambio.
RF34	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Moneda.
RF35	El sistema permitirá generar, descargar e imprimir reportes en el Módulo Usuario.
RF36	El sistema permitirá generar reportes en base al indicador Nivel de incidencias en emisiones.
RF37	El sistema permitirá generar reportes en base al Tasa de entregas completas y puntuales.

Mediante la presente acta se válida y se da conformidad los objetivos a realizarse en el Sprint 4 por parte del equipo de desarrollo, como también, las tareas propuestas del producto (Historias de Usuarios) que contiene el sprint mencionado.

Los asistentes impartirán su aprobación de acuerdo a lo presentado en la planificación del Sprint 0, indicando que la fecha de entrega de este Sprint sería el día 9 de Octubre del 2019.

Lima, 27 de Septiembre del 2019.

CENTINELA TRAVEL S.A.C.
AGENCIA DE VIAJES Y TURISMO

ACTA DE REUNION DEL SPRINT N° 4

Figura 232: Acta de Sprint N° 4

Fuente: Elaboración propia

ACTA DE REUNION DEL SPRINT N° 4

La reunión se lleva a cabo en presencia de los siguientes participantes:

Presentes:

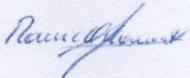
N°	Nombres y Apellidos	Rol
1	Sergio Moran Palacios	Product Owner
2	Ronnie Alarcon Morales	Scrum Master
3	Orvic Yampool Flores Principe	Team Member

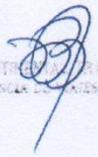
En la presente reunión, se presentó un prototipo del software y las observaciones levantadas propuestas en la anterior reunión, lo cual los usuarios presentes nos dieron su conformidad positiva.

Asimismo, se presentó los reportes en PDF de los módulos de Cliente, Usuario, Moneda, Tipo de Cambio, Productos, Comprobantes de Pago, Anulación de Comprobantes, Cierre de Comprobantes y Reprocesar CPE; al concluir se validó la gestión de datos de cada uno de ellos.

Por consiguiente, se brinda aprobación del Sprint N° 4 ya concluido, para el desarrollo de la Aplicación Web Progresiva para el proceso de facturación de la empresa Continental Travel S.A.C.

Lima, 9 de Octubre del 2019.




CONTINENTAL S.A.C.
AGENCIA DE VIAJES Y TURISMO