



**UNIVERSIDAD CÉSAR VALLEJO**

## **FACULTAD DE INGENIERÍA**

**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**

**“SISTEMA DE INFORMACIÓN PARA LA CONTABILIDAD  
ADMINISTRATIVA PARA LA EMPRESA CORPORACIÓN  
ARZAPALO S.A.C”**

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

**AUTOR:**

Br. Paraguay Arzapalo, Luis Enrique

**ASESOR:**

Mg. Gálvez Tapia, Orleans Moisés

**LÍNEA DE INVESTIGACIÓN:**

Desarrollo de software

**LIMA – PERÚ**

**2015**

El Jurado encargado de evaluar la tesis presentada por don (a):


PARAGUAY ARZAPALO, LUIS ENRIQUE

cuyo título es:

SISTEMA DE INFORMACIÓN PARA LA CONTABILIDAD ADMINISTRATIVA PARA LA EMPRESA CORPORACIÓN ARZAPALO S.A.C.

Reunido en la fecha, escuchó la sustentación y la resolución de preguntas por el estudiante, otorgándole el calificativo de: 14 (número) QUINCE (letras).

Lima, Martes 18 de Agosto del 2015



.....  
PRESIDENTE  
Dra. SALAZAR CHAVEZ, LILY DORIS



.....  
SECRETARIO  
Mgtr. PEREZ FARFAN, IVAN MARTIN



.....  
VOCAL  
Mgtr. GALVEZ TAPIA, ORLEANS  
MOISES

Elaboró	Dirección de Investigación	Revisó	Representante del SGC	Aprobó	Vicerrectorado de Investigación
---------	----------------------------	--------	-----------------------	--------	---------------------------------

## **Dedicatoria**

Dedico el presente trabajo a mi familia, amigos y a Dios por su incondicional apoyo durante el tiempo de estudios de pre – grado, proporcionándome motivación y fuerzas para el cumplimiento del primer objetivo de mi vida.

## **Agradecimientos**

Agradecimiento especial a la empresa Corporación Arzapalo S. A. C por la confianza depositada en mi persona, a mi equipo de trabajo por su compromiso y apoyo en la realización del presente trabajo, a mi asesor, el Mg. Orleans Gálvez, por su dedicación, esmero y compromiso para la realización de la presente tesis como también al Ing. Nicolás Nakasone por proporcionar los lineamientos generales sobre el gestor de base de datos MySQL.



## Declaratoria de autenticidad

Yo, Luis Enrique Paraguay Arzapalo, con DNI N° 46671622, a efecto de cumplir con las disposiciones vigentes consideradas en el Reglamento de Grados y Títulos de la Universidad César Vallejo, Facultad de Ingeniería, Escuela de Ingeniería de Sistemas, declaro bajo juramento que toda la documentación que acompaño es veraz y auténtica.

Así mismo, declaro también bajo juramento que todos los datos e información que se presenta en la presente tesis son auténticos y veraces.

En tal sentido, asumo la responsabilidad que corresponda ante cualquier falsedad, ocultamiento u omisión tanto de los documentos como de información aportada por lo cual me someto a lo dispuesto en las normas académicas de la Universidad César Vallejo.

Lima, 18 de agosto de 2015



Luis Enrique Paraguay Arzapalo  
DNI: 46671622

## **Presentación**

Señores miembros del Jurado:

En cumplimiento del Reglamento de Grados y Títulos de la Universidad César Vallejo, presento ante ustedes la tesis titulada “Sistema de información para la Contabilidad Administrativa para la empresa Corporación Arzapalo S. A. C” que tiene como objetivo principal de determinar la influencia de un sistema de información en la contabilidad administrativa de la empresa en mención.

Para ello, se procedió en dividir a la presente investigación en los siguientes capítulos:

El primer capítulo de esta investigación describe el problema hallado en la empresa, se detallarán las justificaciones y se mostrarán los antecedentes respectivos. Adicional a ello, se expondrán las teorías que avala la presente investigación. Se definen los conceptos relevantes de los sistemas de información y la contabilidad administrativa, y, en base a ello, se plantearán los objetivos de la investigación para posteriormente formular la hipótesis.

En el segundo capítulo, se describe el marco metodológico de la investigación. En ello, se describe el diseño de investigación escogido, la formulación de variables y su operacionalización, y la identificación de la población y muestra.

El tercer capítulo muestra los resultados obtenidos en la presente investigación a través de las pruebas de normalidad y pruebas de hipótesis entre el Pre y Post test.

En el cuarto capítulo se realizan la discusión de los resultados por cada indicador, y, finalmente, en los capítulos quinto y sexto, se detallan las conclusiones y sugerencias de la investigación respectivamente.

# Índice general

Dedicatoria .....	iii
Agradecimientos .....	iv
Presentación .....	vi
Índice general .....	vii
Índice de tablas .....	x
Índice de figuras .....	xi
Resumen .....	xiii
Abstract.....	xiv
<b>CAPÍTULO I: PROBLEMA DE INVESTIGACIÓN .....</b>	<b>15</b>
1.1. Realidad problemática.....	15
1.2. Trabajos previos.....	19
1.2.1. Antecedentes nacionales.....	19
1.2.2. Antecedentes internacionales.....	22
1.3. Teorías relacionadas al tema.....	25
1.3.1. Sistema de información .....	25
1.3.2. Clasificación de los sistemas de información.....	28
1.3.3. Arquitectura de un sistema de información .....	29
1.3.4. Metodología de desarrollo de un sistema de información.....	32
1.3.5. Metodología Scrum.....	35
1.3.6. Contabilidad administrativa .....	40
1.4. Formulación del problema .....	52
1.4.1. Problema general.....	52
1.4.2. Problema específico .....	52
1.5. Justificación del estudio .....	53
1.5.1. Justificación tecnológica .....	53
1.5.2. Justificación económica .....	53
1.5.3. Justificación institucional .....	53
1.5.4. Justificación operativa .....	54
1.6. Hipótesis .....	54
1.6.1. Hipótesis general .....	54

1.6.2. Hipótesis específica.....	54
1.7. Objetivos.....	55
1.7.1. Objetivo general .....	55
1.7.2. Objetivos específicos .....	55
CAPÍTULO II: MÉTODO .....	56
2.1. Diseño de investigación .....	56
2.2. Variables, operacionalización .....	57
2.2.1. Definición conceptual .....	57
2.2.2. Definición operacional.....	67
2.3. Población y muestra .....	67
2.3.1. Población.....	67
2.3.2. Muestra.....	68
2.4. Técnicas e instrumentos de recolección de datos, validez y confiabilidad .....	69
2.4.1. Técnicas .....	69
2.4.2. Instrumentos de recolección de datos.....	70
2.5. Métodos de análisis de datos.....	71
2.5.1. Definición.....	72
2.5.2. Hipótesis estadísticas.....	72
2.6. Aspectos éticos .....	76
CAPÍTULO III: RESULTADOS .....	77
3.1. Pruebas de normalidad .....	77
3.1.1. Test de Shapiro – Wilk para el primer indicador .....	78
3.1.2. Test de Shapiro – Wilk para el segundo indicador .....	79
3.2. Prueba de hipótesis .....	79
3.2.1. Prueba de hipótesis para el primer indicador.....	80
3.2.2. Prueba de hipótesis para el segundo indicador.....	84
CAPÍTULO IV: DISCUSIÓN.....	88
4.1. Indicador 1: Unidades de equilibrio por producto .....	88
4.2. Indicador 2: Razón margen de contribución por territorios de ventas.....	88
CAPÍTULO V: CONCLUSIONES .....	90
CAPÍTULO VI: RECOMENDACIONES.....	91
BIBLIOGRAFÍA .....	92

Anexo 1: Matriz de consistencia.....	97
Anexo 2: Ficha de revisión documental .....	98
Anexo 3: Evaluación de expertos para la metodología de desarrollo .....	104
Anexo 4: Entrevista al Gerente de Corporación Arzapalo S.A.C.....	107
Anexo 5: Carta de aprobación de la empresa.....	108
Anexo 6: Desarrollo de la metodología .....	109

## Índice de tablas

Tabla 1. Cuadro comparativo entre las metodologías ágiles y las metodologías tradicionales .....	33
Tabla 2. Matriz comparativa de las metodologías de desarrollo de software más utilizadas.....	34
Tabla 3. Diferencias entre la contabilidad administrativa y la financiera .....	41
Tabla 4. Tipo de empresa y su base de actividad .....	58
Tabla 5. Relación de distritos objetivos de la empresa Corporación Arzapalo S. A. C.....	68
Tabla 6. Distribución de la muestra .....	69
Tabla 7. Relación entre indicador, técnica e instrumento de recolección de datos .....	71
Tabla 8. Pruebas de normalidad aplicada según la muestra obtenida.....	77
Tabla 9. Resultado de la prueba de Shapiro – Wilk para el primer indicador.....	78
Tabla 10. Resultados de la prueba de Shapiro – Wilk para el segundo indicador.....	79
Tabla 11. Relación entre los indicadores y la prueba de hipótesis a utilizar .....	80
Tabla 12. Resultados de la prueba T – Student para el primer indicador .....	83
Tabla 13. Resultados de la prueba T – Student para el segundo indicador.....	87

## Índice de figuras

Figura 1. <i>Evolución de las unidades de equilibrio en la empresa Corporación Arzapalo S. A. C. en el 2° trimestre – 2014</i> .....	16
Figura 2. <i>Evolución de la razón margen de contribución por territorio de ventas entre abril y junio de 2014</i> .....	18
Figura 3. <i>Actividades básicas de un sistema de información</i> .....	28
Figura 4. <i>Arquitectura en capas</i> .....	31
Figura 5. <i>Diferencias entre arquitectura 3 Capas y arquitectura 3 Niveles</i> .....	31
Figura 6. <i>Etapas de la metodología Scrum</i> .....	36
Figura 7. <i>Proceso de la metodología Scrum</i> .....	36
Figura 8. <i>Reuniones habituales en Scrum</i> .....	38
Figura 9. <i>Artefactos utilizados por Scrum</i> .....	39
Figura 10. <i>Distribución clásica de roles para Scrum</i> .....	40
Figura 11. <i>Relación entre la variable independiente y dependiente en el diseño experimental</i> .....	56
Figura 12. <i>Gráfica del comportamiento de los costos fijos</i> .....	59
Figura 13. <i>Gráfica del comportamiento del costo variable</i> .....	59
Figura 14. <i>Gráfica del comportamiento de los costos mixtos</i> .....	60
Figura 15. <i>Análisis del costo variable por diagramas de dispersión</i> .....	61
Figura 16. <i>Método del punto ato – punto bajo de análisis de costos</i> .....	62
Figura 17. <i>Reporte del margen de contribución por territorios de ventas</i> .....	64
Figura 18. <i>Reporte del margen de contribución por la línea de producto</i> .....	65
Figura 19. <i>Reporte del margen de contribución por vendedor</i> .....	66
Figura 20. <i>Gráfica de la distribución t – Student</i> .....	74
Figura 21. <i>Diagrama de histograma para el indicador unidades de equilibrio por producto (Pre – Test)</i> .....	81
Figura 22. <i>Diagrama de histograma para el indicador unidades de equilibrio por producto (Post – Test)</i> .....	82
Figura 23. <i>Comparativa del Pre – Test y Post – Test para el indicador unidades de equilibrio por producto</i> .....	83
Figura 24. <i>Diagrama de histograma para el indicador razón margen de contribución por territorios de ventas (Pre – Test)</i> .....	85

Figura 25. <i>Diagrama de histograma para el indicador razón margen de contribución por territorios de ventas (Post – Test).....</i>	<i>85</i>
Figura 26. <i>Comparativa del Pre – Test y Post – Test para el indicador razón margen de contribución por territorios de ventas .....</i>	<i>86</i>



## Resumen

La presente tesis trata sobre el desarrollo de un sistema de información para la contabilidad administrativa para la empresa Corporación Arzapalo S. A. C.

El objetivo del presente trabajo de investigación fue determinar la influencia de un sistema de información en la contabilidad administrativa en la empresa mencionada. Para ello, se plantearon dos objetivos específicos que ayudaron a dividir el trabajo. El primero, fue en determinar la influencia del sistema de información desarrollado en las unidades de equilibrio por producto, y el segundo en determinar la influencia del mismo sistema en la rentabilidad del producto por territorios de ventas.

Para desarrollar el sistema de información, se utilizó la metodología Scrum por estar más orientado al usuario, además de permitir obtener resultados (entregables) en periodos de tiempos cortos y también porque permite la utilización de menos recursos humanos.

En relación con el lenguaje de programación, se utilizó Java y como gestor de base de datos a MySQL. En cuanto a las herramientas, se empleó el IDE Eclipse para la programación en Java y SQLyog para MySQL.

Para medir las unidades de equilibrio, se tomó una muestra de 42 productos de la empresa, y para medir la rentabilidad del producto por territorios de ventas, se tomó una muestra de 24 distritos de Lima Metropolitana.

Luego del análisis de datos, se concluyó que el sistema de información para la contabilidad administrativa reduce las unidades de equilibrio de los productos en un 34.94%. Del mismo modo, el mismo sistema de información también aumenta la rentabilidad del producto por territorios de ventas al aumentar la razón margen de contribución en un 16.12% con relación a los resultados del Pre – Test.

### **Palabras claves:**

Sistema de información – Contabilidad administrativa – Punto de equilibrio – Costeo variable – Scrum

## **Abstract**

This thesis is about the development of an information system for management accounting for Arzapalo Corporation Company S. A. C.

The objective of this research was to determine the influence of an information system in the administrative accounts of the company in question. To do this, two specific objectives that helped divide the present work were raised. The first was to determine the influence of the information system developed in balance by product units of the company, and the second to determine the influence of the system on the profitability of product sales territories.

For the development of the information system the Scrum methodology was used, which was selected to be more user-oriented, it produces results (deliverables) in short periods of time and, also, because it allows the use of fewer human resources.

In relation to the programming language, Java was used and MySQL as a database manager. As for the tools, the Eclipse IDE was used for programming in Java and SQLyog for MySQL.

To measure the equilibrium units, a sample of 42 products was taken from the company, and to measure the profitability of the product by sales territories, a sample of 24 districts of Metropolitan Lima was taken.

After data analysis, it is concluded that the information system for management accounting units reduces balance of products by 34.94%. Furthermore, the information system developed, increasing the profitability of the product sales territories by increasing the rate of contribution margin by 16.12% compared to the results of the Pre - Test.

### **Key Word:**

Information system – Management accounting – Breakeven – Variable costing  
– Scrum

## CAPÍTULO I: PROBLEMA DE INVESTIGACIÓN

### 1.1. Realidad problemática

El estudio realizado por el Ministerio de la Producción en el año 2012 señala que ocho de cada diez Mypes (Micro y pequeña empresa) fracasan en sus primeros cinco años de fundación. Para dicha institución, “uno de los principales factores es que los microempresarios no cuentan con información actualizada, consistente y oportuna que les permita tomar decisiones concretas y correctas para satisfacer la demanda de un determinado mercado en condiciones competitivas, sea éste nacional o internacional” (Ministerio de la Producción, 2012 pág. 7).

Parte de la información solicitada por los microempresarios es proporcionada por la contabilidad, el cual, a través de información cuantitativa, proporciona información de cómo está marchando la empresa, si esta genera utilidades o pérdidas. Sin embargo, Centrum Católica (2012, pág. 77) señala que los microempresarios ven a la contabilidad únicamente para cumplir con las obligaciones tributarias, es decir, el control de los ingresos y egresos lo realizan en cuadernos o incluso de manera mental. Además, los cálculos obtenidos son aproximados, válidos y suficientes para ellos, lo que provoca una mala gestión de la empresa.

Dentro de la información contable demandada por los microempresarios es proporcionada por la contabilidad administrativa, que es un tipo de contabilidad no obligatorio en la organización que está orientado hacia los administradores dentro de la organización (Garrison, y otros, 2007 pág. 7). Este tipo de contabilidad permite conocer las unidades a vender para equiparar los costos de producción y evitar pérdidas en la operación. De la misma manera, haciendo uso de la contabilidad administrativa, se puede analizar el rendimiento del producto por territorio de ventas.

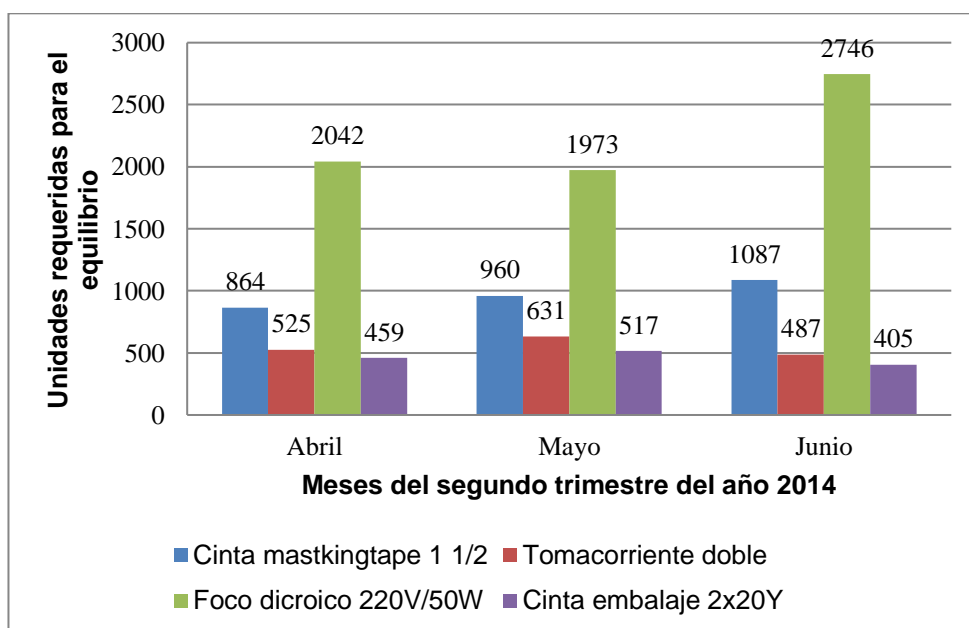
Por su parte, Corporación Arzapalo S. A. C., es una empresa comercial proveedora de productos de ferretería al por mayor con cinco años en el mercado limeño que necesitaba conocer la relación costos – ventas para determinar su utilidad y conocer las unidades a vender necesarias para

---

cubrir los costos de ventas, para que, de esta forma, pueda predecir sus ventas, determinar los costos para las ventas planificadas y se beneficie principalmente con los descuentos ofrecidos sus proveedores.

Las preguntas formuladas por el administrador a menudo eran ¿Cuánto realmente me cuesta vender un producto?, ¿Cuántas unidades debo vender para cubrir esos costos de ventas?, ¿estaré vendiendo las cantidades suficientes para cubrir mis costos de ventas?, ¿puedo aumentar mi utilidad sin incrementar el precio de venta de los productos?

**Figura 1.** Evolución de las unidades de equilibrio en la empresa Corporación Arzapalo S. A. C. en el 2° trimestre – 2014



Fuente: Elaboración propia (2015)

La Figura 1, grafica las unidades a vender necesarias para obtener el equilibrio con los costos de ventas para los cuatro (04) productos principales de la empresa Corporación Arzapalo S.A. C. En ella se observa que el producto tomacorriente doble es el producto que más se tuvo que vender para igualar su costo de venta con un promedio de 3,240 unidades entre abril y junio de 2014.

Si este problema continuara, hubiera generado incertidumbre acerca de los costos reales que interfiere en el proceso de ventas y, por ende, hubiera

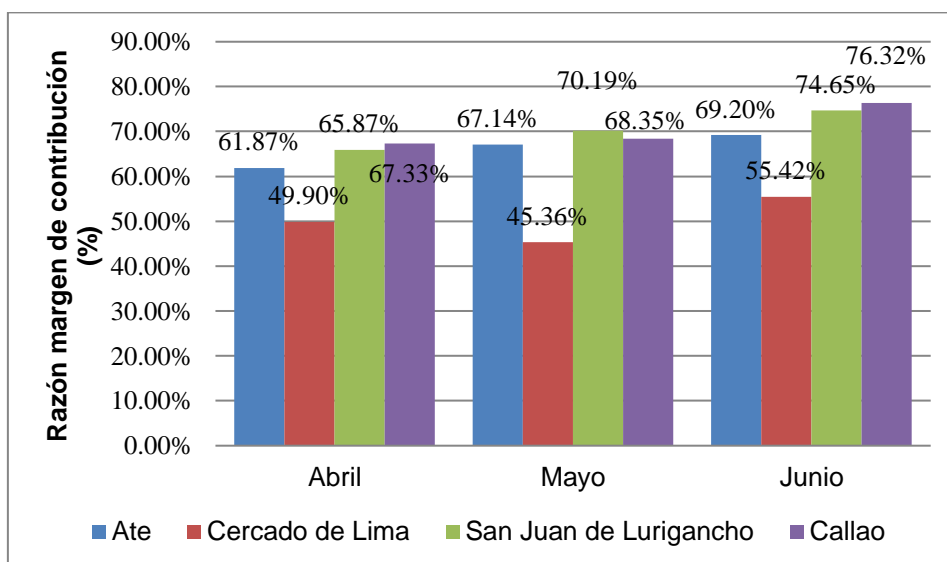
generado pérdidas a la empresa debido a que primero se hubieran puesto precios de ventas mal elaborados, lo que pudieron ocasionar la sobrevaloración o subvaloración de un producto que los clientes pudieran percibir y hubieran optado por comprar los mismos productos a la competencia; y segundo, no se hubieran conocido las unidades necesarias a vender para poder cubrir los costos de ventas, pudiendo adquirir menor mercadería de lo necesario o, por el contrario, adquirir más mercadería de lo demandado y correr el riesgo de generar capital estático o que algunos productos pudieran vencer.

Otro problema que se identificó fue que no se podía determinar los distritos con mayor rentabilidad por producto. Esta actividad era realizada en base a la intuición sin ningún sustento cuantitativo y en forma manual. Esto debido a que sólo se determinaba el rendimiento del producto por sus unidades vendidas. ¿Qué pasaría si un producto no se vende conforme a lo planificado?, ¿será producto de los costos?, ¿se podrá mejorar la rentabilidad de un producto si se hacen ajustes a uno de los costos implicados en las ventas?, ¿será correcto dejar el mercado por falta de rendimiento? ¿Se estará invirtiendo en vano en un producto que tiene menor rendimiento el cual pudiera ser utilizado para otro producto bien posicionado?

Para ello, parte de la contabilidad administrativa es el análisis de segmentos de mercado a través de la razón margen de contribución. Este indicador permite analizar el rendimiento del producto por territorio de ventas con el objetivo de enfatizar los planes de marketing en los distritos con mayor rentabilidad, reducir costos de promoción y ajustar las comisiones por ventas en distritos con menores ventas y/o aumentar el precio de venta en las zonas con menor razón margen de contribución para obtener una mayor utilidad.

La siguiente figura muestra la evolución del margen de contribución de los principales territorios de ventas entre abril y junio de 2014:

**Figura 2.** Evolución de la razón margen de contribución por territorio de ventas entre abril y junio de 2014



Fuente: Elaboración propia (2015)

La Figura 2, grafica que Callao es el territorio con mayor razón margen de contribución promedio de 70.67% entre abril y junio del 2014, seguido de San Juan de Lurigancho con una razón margen de contribución del 70.24% para el mismo periodo.

Si este problema hubiera continuado, la empresa pudo generar pérdidas debido a que se realizarían inversiones vanas en distritos donde el producto no es rentable, se pondría un precio erróneo a los productos en ciertos distritos el cual generaría el rechazo del mercado y por ende la pérdida de clientes y no realizar ajustes en las comisiones de ventas para que un producto pueda aumentar su rentabilidad en ciertos distritos.

De acuerdo con el estudio realizado, se plantearon las siguientes soluciones:

Dotar de un sistema de información que automatice las operaciones de cálculo de las unidades de equilibrio y determinar la rentabilidad del producto mediante el empleo del indicador razón margen de contribución. La principal desventaja que se encontró para esta solución consistía en elaborar un sistema independiente al utilizado en la actualidad, debido a que el software que posee la empresa fue desarrollado en entorno de

escritorio y no se cuenta con el código fuente de dicho sistema que permitiera la adición de nuevas funcionalidades. Otra solución tentativa fue en adquirir un software que abarque los procesos detallados anteriormente. La desventaja presentada fue el alto costo de adquisición, entre licencia y recursos hardware para la cantidad de usuarios que lo utilizarían, además de capacitarlos para su uso.

Finalmente, se optó por la primera alternativa, ya que implicó un menor costo de adquisición y se pudo personalizar de acuerdo con las necesidades de la organización.

## **1.2. Trabajos previos**

### **1.2.1. Antecedentes nacionales**

**A. Alpaca, A. (2012). El Uso de Costos Relevantes para la Toma de Decisiones Gerenciales y su Influencia en la Utilidad de las Empresas Industriales de la Ciudad de Tacna, año 2012 (Tesis para optar el título de profesional de contador público). Universidad nacional Jorge Basadre Grohmann, Tacna, Perú.**

Esta tesis señala que la toma de decisiones rutinarias se toma rápidamente, de manera inconsciente y carecen de un proceso detallado de consideración y que muchas de esas decisiones afectarían la rentabilidad de la empresa, inclusive ponen en riesgos las metas organizacionales. En ese sentido, los empresarios del sector industrial tienen que utilizar, de forma correcta, el análisis de los costos relevantes, debido a que podría mejorar los resultados corporativos.

Como objetivo principal se tuvo el de establecer si el uso de los costos relevantes en la toma de decisiones influye en la rentabilidad de las empresas industriales de la ciudad de Tacna y tuvo como población y muestra a ocho empresas medianas de la industria manufactureras ubicadas en Tacna. Las encuestas

fueros dirigidos a 50 personas directivas y contables, arrojando como resultado que el 90% de los encuestados afirma que el uso de los costos relevantes permite aumentar la utilidad de la empresa.

De esta tesis se extrajo información para el tema del planteamiento del problema, debido a que indica el problema de las Mypes en relación con el manejo de costos y su relación con la toma de decisiones.

**B. Yi, Z. (2008). Sistema de Información para el Costeo por Procesos de las Industrias Vitivinícolas (Tesis para optar el título de ingeniero informático). Pontificia Universidad Católica del Perú, Lima, Perú.**

La tesis proporcionó, a la industria vitivinícola, información oportuna sobre los costos incurridos en la producción de vinos. Del mismo modo, la autora logró la trazabilidad que involucra la preferencia y reconocimiento del público objetivo; además de mejorar la toma de decisiones al administrar y planear eficientemente las adquisiciones de los recursos.

El objetivo general fue de analizar y diseñar un sistema de información de costeo por procesos enfocado al sector vitivinícola.

Esta tesis fue desarrollada bajo un enfoque de investigación aplicada. Esto debido a que se presentó una solución informática el cual resolvió un problema práctico de la sociedad, que para este caso fue la industria vitivinícola.

A manera de conclusión, la autora mencionó que la implementación del sistema de información de costos por procesos ayudó a lograr la trazabilidad del producto terminado. Ello permitió vincular la condición del producto final con daños



consecuentes de la distribución, producción o adquisición de materia prima.

De esta tesis se pudo obtener información acerca del marco teórico para la variable independiente con relación a sistema de información y su metodología de desarrollo; además de obtener una visión general para la construcción de una solución informática en el tema de costeo y de las herramientas a utilizar para desarrollar un sistema de información.

**C. Florián, W. y Fernández, C. (2013). Sistema de costos por órdenes en la fijación de precios y control de recursos en la empresa Corporación Wamatray SAC en la ciudad de Trujillo periodo Enero – Julio 2013 (Tesis de Grado). Universidad Privada Antenor Orrego, Trujillo, Perú.**

El problema encontrado en la tesis fue que la empresa Wamatray necesitaba un precio de venta competitivo para afrontar la guerra de precios de sus competidores. Debido a ello, la empresa buscó determinar el costo real de sus servicios. Para ello, requirió una adecuada gestión administrativa para obtener información relevante con la finalidad de fijar sus precios y poder controlar sus recursos de una manera más eficaz.

Como objetivo principal, los tesisistas plantearon en demostrar como un sistema de costos por órdenes de servicios mejorarán la fijación de precios y el control de los recursos en la empresa Corporación Wamatray S.A.C. de la ciudad de Trujillo.

Para ello, los tesisistas utilizaron una población/muestra de 9 colaboradores y usaron la observación de campo, el análisis documental y entrevistas como técnicas de recolección de datos.

Finalmente, los investigadores llegaron a la conclusión que se obtuvo un punto de equilibrio de S/. 313,803.10, con una

facturación real de S/. 368,437.10. De esta manera, lograron demostrar que el nuevo cálculo de precios para el servicio brindado fue adecuado y ayudó a la toma de decisiones gerenciales.

De esta tesis, se extrajo información para el marco teórico, debido a que toca el tema de la relación costo – volumen – utilidad y su aplicación en el punto de equilibrio, además de los datos para hacer la discusión de datos.

### **1.2.2. Antecedentes internacionales**

**A. Maicán, P., Salazar, F. y Sánchez, D. (2013). Planificación de costos y utilidades en la empresa movistar zona franca celular's C.A, de cumana, estado sucre, año 2013 (Trabajo Especial de Grado presentado como requisito parcial para optar al título de Licenciado en Contaduría Pública). Universidad de Oriente, Núcleo de Sucre, Venezuela.**

El problema hallado en esta tesis fue que la empresa en mención consideró necesario analizar la planificación de costos y utilidades.

Como objetivo principal se planteó analizar la planificación de costos y utilidades en la empresa Movistar Zona Franca Celular's C.A del año 2013. El estudio que usaron los autores fue una investigación de campo, debido a que los datos se obtuvieron de manera directa.

Los autores tomaron una población – muestra de seis (06) empleados de la empresa durante el año 2013.

Finalmente, los autores concluyeron que el trabajo de investigación suministró a la empresa un análisis detallado de la planificación de costos y utilidades, el cual ayudó considerable a la alta gerencia en la elaboración planes de operación con

objetivos concretos para la comparación del desempeño real versus el esperado y en la presentación de información financiera.

De esta tesis, se extrajo información para el marco teórico, debido a que toca el tema de la relación costo – volumen – utilidad y su aplicación en el punto de equilibrio.

**B. Quinche, C. y Quizhpe, K. (2011). Análisis de la relación Costo – Volumen – Utilidad como base de planificación presupuestaria en la fábrica de embutidos La Escocesa de la ciudad de Loja. (Tesis de grado). Universidad Nacional de Loja, Ecuador.**

El problema hallado en la tesis fue que la empresa necesitaba conocer el presupuesto para la planificación de producción de los productos en base a los costos directos e indirectos para determinar la cantidad a producir de acuerdo con las ventas de los productos fabricados. El objetivo principal fue de analizar la relación costo – volumen – utilidad (CVU) como base de la planificación presupuestaria en la fábrica de embutidos.

La investigación fue de tipo descriptiva porque solo buscó explicar cómo funciona la relación CVU en la planificación presupuestaria. Para ello, las investigadoras tomaron una población y muestra de 12 productos. Esta investigación no presentó hipótesis. Como resultado se obtuvo que la empresa produjo 70,439.30 libras para cubrir los costos de operación, lo que equivalen a \$91,806.21 dólares americanos.

La conclusión que llegaron las investigadoras fue que la falta de un análisis costo – volumen – utilidad no permitió que el gerente cuente con información oportuna y planificada de los costos y gastos que inciden en la producción y por ende conocer la utilidad que ello genera ya que esta empresa tiene como único ingreso la venta de ahumados y embutidos.

---

De esta tesis se obtuvo datos cuantitativos con el objetivo de hacer una comparación con los datos conseguidos en el presente trabajo. Además, se obtuvo información para el marco teórico con relación al análisis costo – volumen – utilidad.

**C. Helfmann, G. (2009). Determinación de la rentabilidad, costos y márgenes de los principales productos de una empresa de telecomunicaciones (Tesis de grado). Universidad de Chile, Chile.**

El problema presentado en la tesis fue que la empresa Empresas de Telefónica Chile S.A. carecía de un modelo de determinación de precios que tome como principal entrada, la estructura de costos y márgenes de sus productos. Esto se debió a que la empresa no contaba con una adecuada información acerca de la distribución de costos de cada producto. Esto impidió la obtención de los ingresos por servicio, los márgenes y la rentabilidad real de cada uno de ellos.

Como objetivo principal se tuvo la de implementar un nuevo modelo de precios que tome la estructura de costos para obtener la rentabilidad real de los productos. La investigación fue de tipo descriptiva. Para ello, el investigador contó con una población/muestra de cuatro productos.

Los resultados que se obtuvieron fueron que la rentabilidad real del producto A alcanzó un -3%, y, además, los márgenes de los subproductos A.1 y A.2 fueron de -26% y 17% respectivamente. Adicional a ello, la rentabilidad del producto B fue de 68% y la de sus subproductos B.1 y B.2 un 70% y 53% respectivamente. Finalmente, el estudio arrojó las rentabilidades del producto C y producto D que obtuvieron un 36% y 54% respectivamente.

La investigación concluye que la política comercial ponía foco en la venta del subproducto A.2 por sobre el A.1 y un aumento del precio para ambos, y que La información obtenida resultó clave y

---

muy reveladora para la empresa, e implicó la implementación inmediata de políticas comerciales que mejoraron el resultado de los productos.

De esta tesis se obtuvo datos cuantitativos con la finalidad de comparar con los datos conseguidos en el presente trabajo de investigación. Además, se obtuvieron los lineamientos para el análisis de rentabilidad del producto.

### **1.3. Teorías relacionadas al tema**

#### **1.3.1. Sistema de información**

De Pablos, López, Martín - Romo, & Medina (2013, pág. 21), basándose en la Teoría General de Sistemas lo define como “un conjunto de recursos técnicos, humanos y económicos, interrelacionados entre sí, organizados en torno al objetivo de satisfacer las necesidades de información de una organización empresarial para la gestión y la correcta toma de decisiones”.

Por su parte Peña (2006, pág. 7), lo define como “un conjunto de elementos interrelacionados con el propósito de prestar atención a las demandas de información de una organización, para elevar el nivel de conocimientos que permitan un mejor apoyo a la toma de decisiones y desarrollo de acciones”.

Técnicamente, un sistema de información permite recuperar, procesar, almacenar y distribuir información para apoyar la toma de decisiones y el control en una organización (Laudon, y otros, 2008 pág. 14), ya sea mediante el uso de un simple archivo o un potente y complejo software sobre una computadora.

Entre los elementos que conforman los sistemas de información, Cohen y Asín (2000, pág. 4), menciona que los sistemas de información esta formados por los siguientes elementos:

- **El equipo computacional:** Llamado comúnmente como hardware. Ejemplos: computadoras, impresoras y periféricos que son necesarios para que el sistema de información opere (Cohen, y otros, 2000 pág. 4).
- **El recurso humano:** “Conformado por las personas que alimentan de datos al sistema de información” (Cohen, y otros, 2000 pág. 4).
- **Los datos o información fuente:** “Pieza básica de información que representa una palabra [...] En tanto la información es el conjunto de datos con un determinado sentido” (Lerma y Kirchner, y otros, 2012 pág. 297)
- **Los programas:** “Conjunto de instrucciones secuenciadas que hace que la computadora realice un proceso o tratamiento a los datos” (Lerma y Kirchner, y otros, 2012 pág. 297)
- **Las telecomunicaciones:** Conformado por el hardware y software que facilitan la transmisión de información de forma electrónica (Cohen, y otros, 2000 pág. 5).
- **Procedimientos:** “Políticas y reglas de operación de la parte funcional del proceso de negocio, como de los mecanismos para hacer operar una aplicación en la computadora” (Cohen, y otros, 2000 pág. 5).

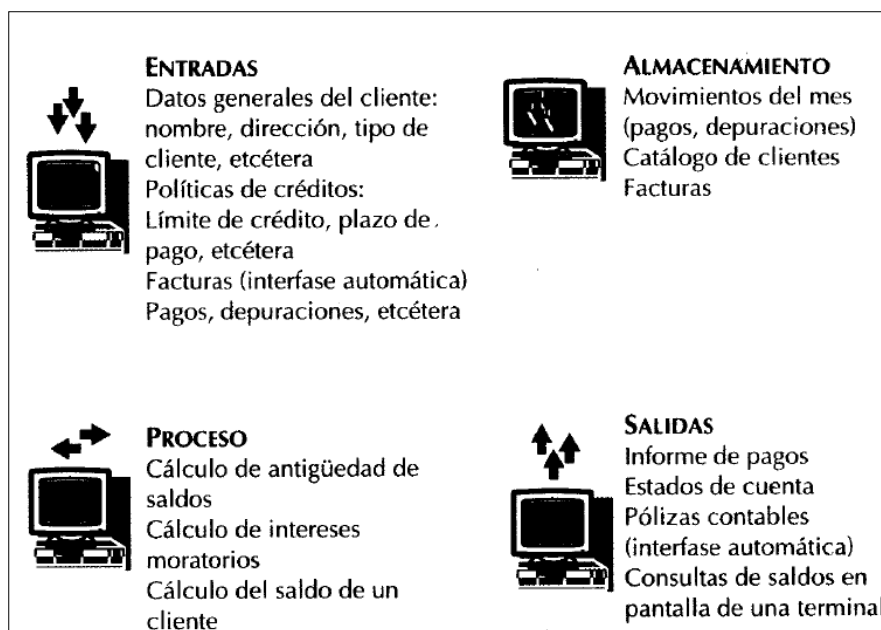
Cabe resaltar la diferencia entre un sistema de información y un software. Como se mencionó, el primero es un conjunto de elementos interrelacionados, mientras que un software es un elemento de ello. Pressman (2009, pág. 5), define al software como “un conjunto de instrucciones que al ser ejecutadas proporcionan funciones y el grado de desempeño deseado. Es un elemento lógico, de un sistema”. Esta definición coincide con Cohen y Asín (2000,

pág. 36), mencionándolo como uno de los elementos de un sistema de información.

Cohen y Asín (2001, pág. 5), establece que un sistema de información debe de cumplir con cuatro actividades básicas (ver figura 3), las cuales son:

- **Entrada de información:** “Es el proceso por el cual el sistema de información captura los datos ingresados por el usuario del sistema [...]. Pueden ser ingresados por cualquier periférico como un teclado, scanner, voz, etc” (Cohen, y otros, 2000 pág. 5).
- **Almacenamiento:** Propiedad que permite al sistema guardar la información para ser reutilizada cuando se necesite. Esta información suele estar guardada en estructuras lógicas (archivos) o en estructuras físicas (disco duro, memoria pen drive, etc.) (Cohen, y otros, 2000 pág. 5).
- **Procesamiento:** Es la capacidad que tiene los sistemas de información para realizar operaciones que transforman datos en información (Cohen, y otros, 2000 pág. 6).
- **Salida de información:** Es la capacidad donde un sistema de información permite mostrar la información procesada al exterior por medio de la pantalla, impresora, voz, etc. Cabe resaltar que esta salida puede ser la entrada para otro sistema. (Cohen, y otros, 2000 pág. 6).

**Figura 3. Actividades básicas de un sistema de información.**



Fuente: Cohen y Asin (2000)

La figura 3 muestra un resumen de las cuatro actividades básicas de un sistema de información.

### 1.3.2. Clasificación de los sistemas de información

Cohen y Asín (2000, pág. 9), clasifican a los sistemas de información de acuerdo a su uso en sistemas transaccionales, y los sistemas de apoyo a la toma de decisiones, que a su vez lo subdivide en sistema de apoyo a la toma de decisiones, sistemas para el apoyo a la toma de decisiones en grupo, sistemas expertos para la toma de decisiones, y sistemas de información para ejecutivos. También propone la existencia de un tercer tipo de sistema la cual denomina sistemas estratégicos, que se encargan de ayudar a las organizaciones a obtener ventajas competitivas mediante el uso de las TICS.

Por su parte, Kendall y Kendall (2005, pág. 2), los clasifica como sistema de procesamiento de transacciones (TPS, por sus siglas en inglés), sistema de automatización de oficinas (OAS, por sus siglas en inglés), los sistemas de trabajo del conocimiento (KWS, por sus



siglas en inglés), los sistemas de información gerencial (MIS, por sus siglas en inglés), sistemas de apoyo a la toma de decisiones (DSS, por sus siglas en inglés), y los sistemas expertos e inteligencia artificial (AI, por siglas en inglés).

De estos autores, podemos concluir que los sistemas se clasifican en tres grupos: los sistemas de transacciones, los sistemas de apoyo a la toma de decisiones y los sistemas expertos.

- **Sistema de procesamiento de transacciones**

Según Kendall y Kendall (2005, pág. 2), “son sistemas de información computarizada creados para procesar grandes cantidades de datos relacionadas con transacciones rutinarias de negocios, como las nóminas y los inventarios”.

Laudon y Laudon (2008, pág. 52), añade que “estos sistemas responden a las preguntas rutinarias y dar seguimiento al flujo de transacciones en la organización, como por ejemplo ¿Cuántas partes hay en la existencia?, ¿Qué ocurrió con el pago del señor Sánchez?”.

De Pablos, López, Martín - Romo, y Medina (2013, pág. 151) añade que estos sistemas “representan tareas iterativas y rutinarias que no necesitan de carga decisoria por lo que están prácticamente automatizados”.

Por su parte, Cohen y Asín (2000, pág. 8), señalan que estos sistemas son los primeros en implementarse debido a que reducen costos operativos de mano de obra.

### **1.3.3. Arquitectura de un sistema de información**

Grochow (citado por Pressman, 2009, pág. 276), define a la arquitectura de un sistema como “un marco conceptual completo que

describe su forma y estructura (sus componentes y la manera en que se integran)”.

“La arquitectura de un software o sistema de cómputo es la estructura o las estructuras del sistema que incluyen los componentes del sistema, las propiedades visibles externamente de esos componentes y las relaciones entre ellos” (Pressman, 2010 pág. 277).

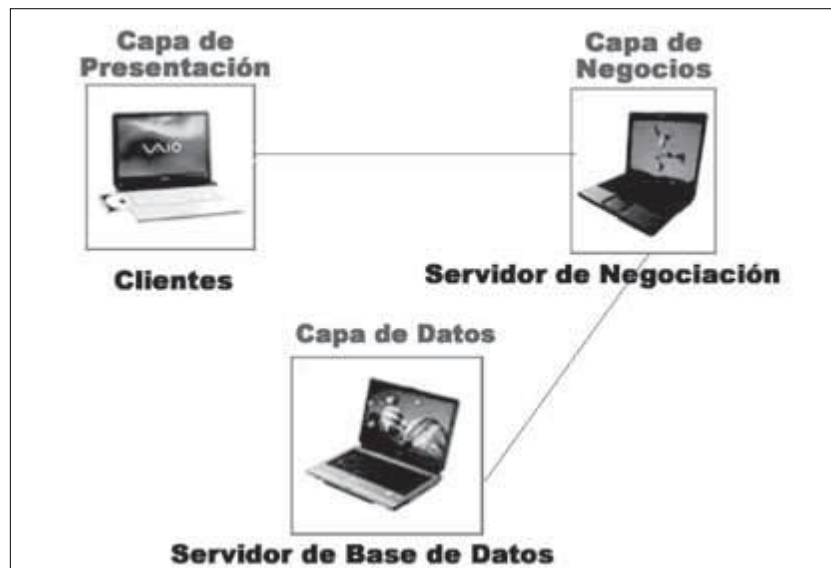
Hofmeister (citado por Sommerville, 2005, pág. 220), sugiere que la arquitectura de un sistema informático “sirve como un plan de diseño que se usa para negociar los requerimientos del sistema y como una forma de estructurar las discusiones con los clientes, desarrolladores y gestores”.

- **Arquitectura en capas**

Pressman (2009, pág. 284), lo define como una arquitectura estratificada, en donde se definen varias capas definidas y que cada una de ellas cumple una función en especial.

“El modelo en capas o arquitectura en capas, organiza el sistema en capas, cada una de las cuales proporciona un conjunto de servicios. Cada capa puede pensarse como una maquina abstracta cuyo lenguaje de maquina se define por los servicios proporcionados por la capa” (Sommerville, 2005 pág. 227).

**Figura 4. Arquitectura en capas**

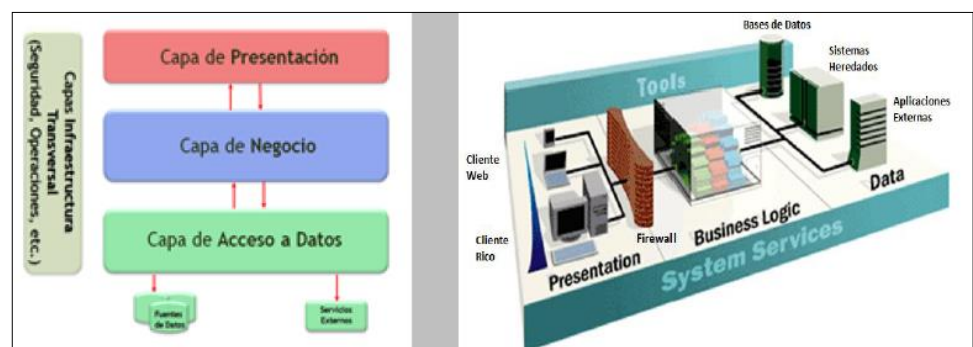


**Fuente:** Moquillaza, Vega & Guerra (2010)

La figura 4 muestra un ejemplo de la arquitectura con 3 capas: la capa de presentación, que hace referencia al cliente, la capa de negocios, centrado en un servidor de negociación, y la capa de datos representado por un servidor de base de datos.

De la Torre, Zorrilla, Ramos, y Calvarro (2010, pág. 33), hace énfasis en las diferencias entre Capa y Nivel, que a pesar de llevar los mismos nombres (2 – N, 3 – N), las capas se enfocan en la división lógica de componentes mientras que los niveles tratan de la distribución física considerando la topología de redes y locaciones remotas.

**Figura 5. Diferencias entre arquitectura 3 Capas y arquitectura 3 Niveles**



**Fuente:** De la Torre, Zorrilla, Ramos, y Calvarro (2010)

La figura 5 muestra la diferencia entre arquitectura 3 capas (lado izquierdo) y la arquitectura en 3 Niveles (lado derecho).

Moquillaza, Vega & Guerra (2010, p. 59), añade que el diseño más utilizado en la actualidad es el diseño en tres capas.

A continuación, se describen las capas básicas de una arquitectura 3 capas:

- a) Capa de presentación:** Es la parte gráfica de un sistema de información. Esta capa tiene una comunicación única con la capa de negocios para la recuperación de datos a mostrar.
- b) Capa de negocio:** Es la responsable todo el procesamiento del sistema de información. Para Moquillaza, Vega & Guerra (2010, p. 58), esta capa intermedia contiene objetos que se enlazan con las diversas entidades del sistema de información.
- c) Capa de datos:** Es responsable de la extracción de datos. Para Moquillaza, Vega & Guerra (2010, p. 58), en esta capa se implementan las conexiones a los servidores de base de datos, los cuales reciben peticiones de inserción o selección de datos provenientes de la capa de negocio.

#### **1.3.4. Metodología de desarrollo de un sistema de información**

Según la RAE (2014, Diccionario en línea), se entiende por metodología al “conjunto de métodos que se siguen en una investigación o en una exposición doctrinal. Partiendo de esta definición se puede deducir que las metodologías de desarrollo de sistemas de información son un conjunto de métodos utilizados para la construcción de un sistema de información”.

De Pablos, López, Martín - Romo, & Medina (2013, pág. 194) complementa que las metodologías de desarrollo son “un conjunto de técnicas, modelos y métodos que permiten planificar y controlar proyectos de construcción de un sistema de información”.

En la actualidad, existen dos grandes enfoques para el desarrollo de sistemas de información las cuales son descritos en el siguiente cuadro:

**Tabla 1.** Cuadro comparativo entre las metodologías ágiles y las metodologías tradicionales

Metodologías ágiles	Metodologías tradicionales
Scrum, XP	RUP
Basadas en buenas prácticas de desarrollo de software.	Basadas en normas, estándares rígidos.
Alta flexibilidad al cambio	Hay resistencia al cambio
Impuestas por el equipo	Impuestas por terceros.
Proceso menos controlado, con pocas políticas/normas	Proceso muy controlado, con demasiadas políticas/normas.
No existe contrato prefijado	Existencia de contratos
El cliente pertenece al equipo.	El cliente colabora con el equipo mediante reuniones.
Equipos reducidos	Equipos grandes.
Documentación necesaria.	Demasiada documentación.
Pocos roles.	Muchos roles.
Poco énfasis en la arquitectura del software.	Mucho énfasis en la arquitectura de software.

**Fuente:** Amaro y Valverde (2007)

De acuerdo con la Tabla 1, se puede apreciar las diferencias entre las metodologías tradicionales con las metodologías ágiles de desarrollo de software. De ellas se pueden destacar que las metodologías ágiles utilizan una menor cantidad de recursos humanos y son adaptables al cambio.

Para la presente tesis, se optó por utilizar una metodología ágil. Esto se debe a que el cliente, en este caso el Gerente General y el contador de la Corporación Arzapalo, fueron parte del equipo, ya que ellos proporcionaban los lineamientos del sistema de información y

aprobaban los diseños. Aparte de ello, se contó con un equipo de cinco (05) personas (incluyendo al Gerente General y el contador de la empresa) lo cual significó que se tuvo pocos roles durante el desarrollo del sistema de información.

Como señala la Tabla 1, se tiene dos metodologías ágiles (Scrum y XP). Para la elección de la metodología se consultó a tres (03) expertos en la materia las cuales recomendaron el uso de la metodología Scrum para el presente trabajo de investigación. Adicional a ello, se tomó en cuenta la comparación de las metodologías de acuerdo con la siguiente tabla:

**Tabla 2.** *Matriz comparativa de las metodologías de desarrollo de software más utilizadas.*

Característica	RUP	MSF	XP	SCRUM
Heredan modelos	X	X		
Independiente de tecnologías		X		X
Documentación estricta	X	X		
Estrictamente ordenado	X		X	
Enfocado en procesos	X	X		
Enfocado en el equipo			X	X
Resultados en corto tiempo			X	X
Cliente como parte del equipo			X	X
Tiempos establecidos	X	X	X	X
Reestructuración de código fuente			X	
Desarrollo incremental y cíclico	X	X	X	X
Alta flexibilidad a los cambios			X	X

**Fuente:** Pérez (2011)

Como muestra la tabla 2, se adoptó la metodología Scrum principalmente porque está más enfocado a los usuarios, es decir, busca que sea fácil de usar para el usuario y permite una rápida

adaptación por parte de ello y reducir el rechazo de ellos. También se escogió Scrum porque permite adaptarse a los cambios.

Entre Scrum y XP, se optó por Scrum debido a que no se contó con mucho personal para cumplir el paradigma de “desarrollo en parejas” presentada en la metodología XP, el cual una misma actividad debe de ser desarrollada por dos personas, una que desarrolla y otra que supervise.

### 1.3.5. Metodología Scrum

Fue diseñada por Hirotaka Takeuchi e Ikujiro Nonaka a mediados de los años 80 como una metodología ágil para el desarrollo de proyectos de diversa índole (Palacio, 2008 pág. 87).

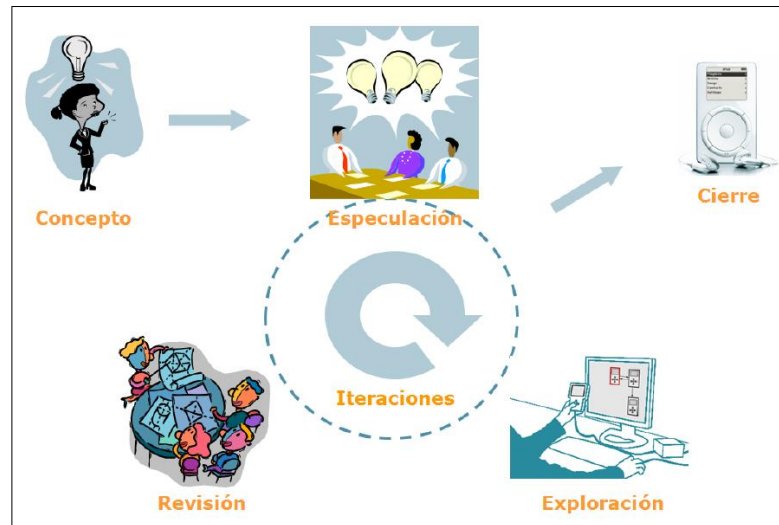
Por su parte Deemer, y otros (2009 pág. 5), añaden que es “un marco de trabajo iterativo e incremental para el desarrollo de proyectos, productos y aplicaciones”.

Scrum, como parte del manifiesto ágil, adopta sus etapas de desarrollo. Al respecto, Palacio (2008 pág. 62), lo divide en cinco (05) etapas, las cuales se describen a continuación:

- **Concepto:** En la fase se crea la visión del producto. Se selecciona al equipo que lo llevará a cabo.
- **Especulación:** Aquí se establecen las limitaciones impuestas por el negocio (costos y agendas principalmente) y se determina el primer alcance de lo que se puede producir.
- **Exploración:** Desarrollo de las tareas que determinó el equipo en la fase anterior para generar el siguiente incremento del producto.
- **Revisión:** En esta etapa, los miembros del equipo y los principales stakeholders evalúan las funcionalidades construidas en el sprint.

- **Cierre:** Es la entrega de una versión de producto esperado establecida en la fase de concepto.

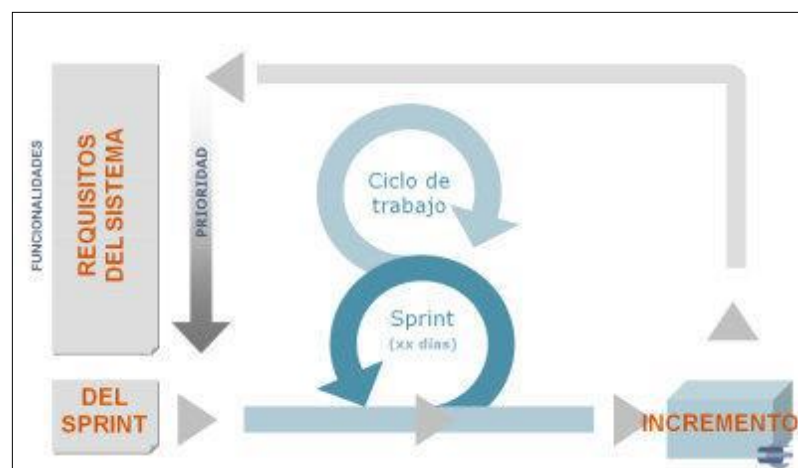
**Figura 6. Etapas de la metodología Scrum**



**Fuente:** Palacio (2008)

La figura 6 grafica las etapas de la metodología Scrum que comienza con la etapa de concepto y entra en un bucle que contiene las etapas de especulación, exploración y revisión, y sale del mismo en la etapa de cierre cuando se entrega el incremento.

**Figura 7. Proceso de la metodología Scrum**



**Fuente:** Herranz *et al.* (2011)



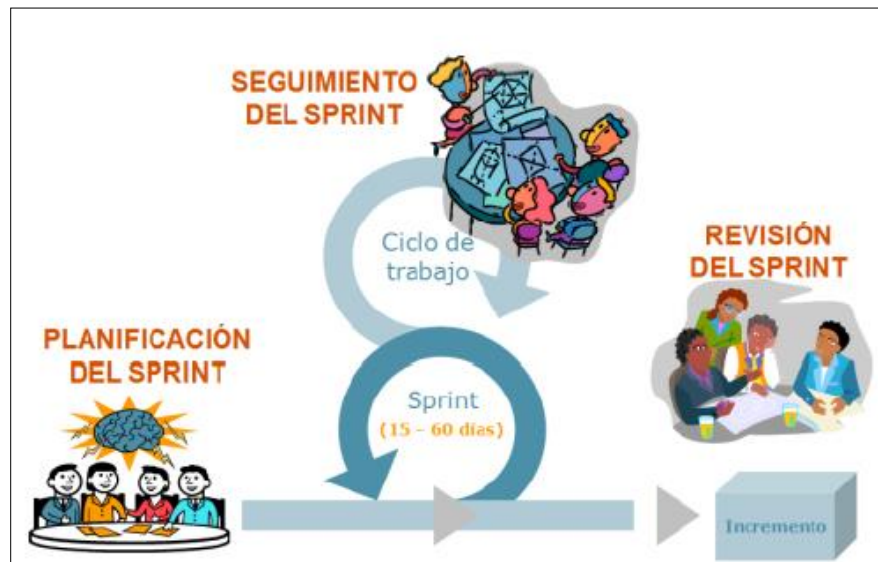
La figura 7 muestra el proceso de la metodología Scrum. Los requisitos del sistema son organizados en los Sprints por su prioridad. Estos son desarrollados y finalizan con el incremento para luego volver a revisar los requisitos del sistema y comenzar con un nuevo sprint.

Palacio y Ruata (2011, p. 61), señala que Scrum está conformada por los siguientes elementos:

#### **a) Las reuniones**

Al respecto, Herranz *et al.* (2011, p. 12), menciona que existen cuatro (04) tipos de reuniones para tratar diversos puntos del proyecto las cuales son:

- **Planificación del Sprint:** Esta reunión consiste en definir la Pila de Sprint. En una primera parte, el Product Owner explica al equipo de desarrollo los requerimientos a desarrollar.
- **Seguimiento del Sprint:** Es una reunión diaria (aproximadamente 15 minutos) donde el equipo responde a las siguientes preguntas, ¿Qué se hizo ayer?, ¿qué se hará hoy?, y ¿Qué impedimentos tuvo?
- **Revisión del Sprint:** Es una reunión donde el equipo Scrum presenta el incremento realizado de un determinado sprint al Product Owner.
- **Retrospectiva:** Llamada también reunión de mejora continua. Aquí el Equipo analiza los inconvenientes suscitados durante el sprint y establece las acciones mejorables que pueden ser utilizadas para el siguiente sprint.

**Figura 8.** Reuniones habituales en Scrum

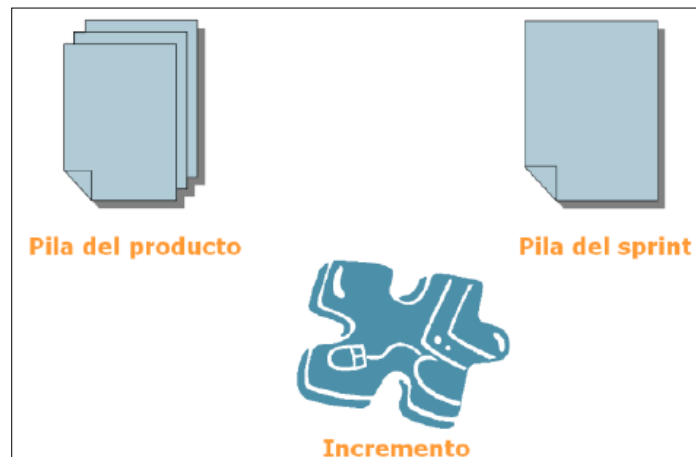
Fuente: Palacio y Ruata (2011)

La figura 8 muestra los tipos de reuniones utilizados en Scrum: las reuniones de planificación, seguimiento y de revisión del sprint.

#### b) Los artefactos

Herranz *et al.* (2011, p. 12), señalan que los artefactos utilizados por Scrum son: la pila del producto, la pila del sprint y los incrementos.

- **Pila del Producto (Product Backlog):** Es la lista de todos requisitos del producto (historias de usuario) proporcionados por el dueño del producto.
- **Pila del Sprint (Sprint Backlog):** Son las tareas desglosadas por cada historia de usuario que serán ejecutadas el equipo de desarrollo.
- **Incremento:** Es el resultado del desarrollo en cada Sprint.

**Figura 9.** *Artefactos utilizados por Scrum*

**Fuente:** Herranz *et al.* (2011)

En la figura 9 se puede apreciar en forma gráfica los artefactos utilizados en Scrum según Herranz *et al.* (2011, p. 12): la pila del producto, la pila del sprint y el incremento.

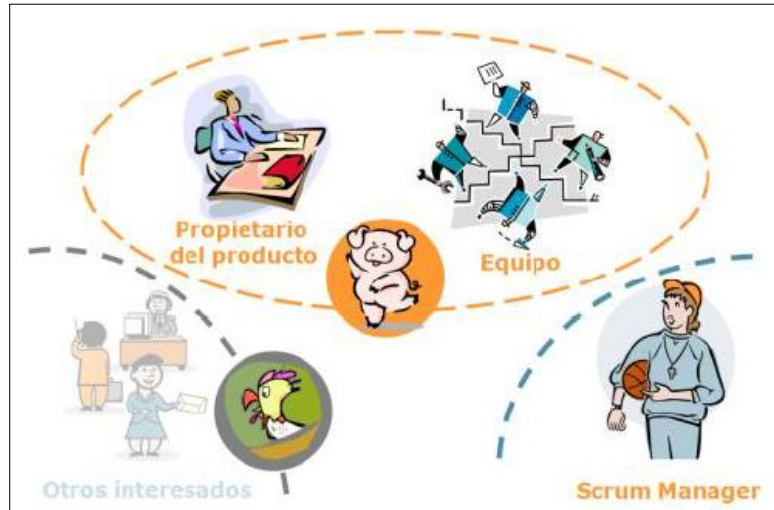
### c) Roles o responsabilidades

Para Palacio y Ruata (2011, p. 61), “todas las personas que intervienen, o tienen relación directa o indirecta con el proyecto, se clasifican en dos grupos: comprometidos e implicados, o también llamados cerdos y gallinas”. A continuación, se listan los roles habituales en Scrum:

- **Propietario del Producto (Product Owner):** Responsable de lograr el máximo valor del producto desarrollado para los usuarios y el resto de implicados.
- **Equipo de desarrollo (Team):** Es el responsable del desarrollo del producto en sí.
- **Otros interesados (stakeholders):** Pertenecen al grupo de los implicados y son los principales interesados del desarrollo del producto. Pueden definir el éxito o fracaso de un determinado proyecto.
- **Scrum máster:** Encargado de la correcta aplicación de la metodología Scrum en el equipo. Para Herranz *et al.*

(2011, p. 12), “no debe ser considerado ni comprometido ni implicado debido que sus responsabilidades no son del proyecto, sino del grupo de procesos y métodos de la organización”.

**Figura 10.** *Distribución clásica de roles para Scrum*



**Fuente:** Palacio & Ruata (2011)

En la figura 10 se puede apreciar los roles que intervienen en Scrum según Palacio y Ruata (2011, p. 61).

### 1.3.6. Contabilidad administrativa

La contabilidad administrativa “es un proceso al servicio de las necesidades internas de la administración, orientado a facilitar las funciones administrativas de planeación y control, así como la toma de decisiones”. (Guajardo, y otros, 2008 pág. 18).

Horngrén, Sundem y Stratton (2007, pág. 5), agrega que este tipo de contabilidad genera información de apoyo a los administradores para identificar, medir, acumular, analizar, preparar, interpretar y comunicar información para cumplir los objetivos organizacionales.

Según Warren, Reeve, y Duchac (2010, pág. 2), la contabilidad administrativa está diseñada para satisfacer las necesidades

específicas de la administración de una empresa. Esta información incluye los siguientes:

1. Datos históricos, los cuales proporcionan medidas objetivas de operaciones pasadas.
2. Datos estimados, que proporcionan estimaciones subjetivas acerca de las decisiones futuras.

En varias oportunidades se suele confundir a la contabilidad administrativa con la contabilidad financiera, “mientras la primera se concentra en la presentación de información cuantitativa y cualitativa orientada al futuro, la contabilidad financiera presenta información cuantitativa e histórica para terceros” (Yermanos, y otros pág. 7).

En siguiente tabla se presentan las diferencias entre la contabilidad financiera y la contabilidad administrativa:

**Tabla 3.** *Diferencias entre la contabilidad administrativa y la financiera*

	<b>Contabilidad administrativa</b>	<b>Contabilidad financiera</b>
Usuario objetivo	Administradores de distintos niveles	Agentes externos. Ejemplo: inversionistas, instituciones públicas, etc.
Elección de las medidas contables	Restringidos en la relación de costos con los beneficios para la toma de decisiones.	Están restringidos por los PCGA
Enfoque de tiempo	Se orienta al futuro. Ejemplo: presupuesto 2001 versus rendimiento real 2001.	Se orienta al pasado. Ejemplo rendimiento real 2003 versus rendimiento real 2002.
Horizonte de tiempo	Flexible. Se puede analizar en periodos de una hora hasta 15 años.	Menos flexible. Por lo general un año o un trimestre.

Nivel de detalle en los reportes	Son detallados: incluyen detalles de cada área de la organización, productos, territorios, etc.	Son resumidos: Reflejan los resultados globales de la organización
Campo de acción	Mayor uso de ciencias económicas, de decisión y de comportamiento.	Menor uso de dichas disciplinas

**Fuente:** Horngren, Sundem, y Stratton, 2010

Con respecto a la diferencia entre la contabilidad administrativa y la contabilidad de costos, Horngren, Datar y Foster (2007, pág. 2), señalan lo siguiente:

“la contabilidad de costos proporciona información para dirigir la contabilidad administrativa y financiera, es decir, mide, analiza y presenta información financiera y no financiera relacionada con los costos de adquirir o utilizar recursos en una organización. Por ejemplo, calcular el costo de un producto es una función de la contabilidad de costos que responde a las necesidades de valoración de inventarios de la contabilidad financiera y a las necesidades de toma de decisiones de la contabilidad administrativa en materia de la elección de los productos a ofrecer”.

Warren, Reeve, y Duchac (2010 pág. 39), menciona que la contabilidad administrativa se puede estudiar desde las siguientes perspectivas.

**a) Costeo por órdenes de producción:** “Registra y resume los costos de producción por cada trabajo, orden o pedido. El total resultante y los costos del producto por unidad se pueden comparar con órdenes similares, comparar a través del tiempo o

comparar con los costos esperados para evaluar y controlar los costos” (Warren, y otros, 2010 pág. 54).

- b) Costeo por proceso:** Registra los costos del producto para cada departamento o proceso de manufactura.

Para analizar este tipo de costeo, Warren, Reeve, & Duchac (2010 pág. 87) “toma en cuenta el reporte del costo de producción, que resume la información de producción y costo para un departamento, y es usada por los administradores para tomar decisiones que involucran el control y la mejora de operaciones”.

- c) Comportamiento del costo:** “El comportamiento del costo es de utilidad para los administradores porque permite predecir ganancias conforme cambian los volúmenes de ventas y producción” (Warren, y otros, 2010 pág. 130).

Para determinar el comportamiento del costo, se suelen emplear el siguiente indicador:

- Costo variable por unidad (CVU):

$$CVU = \frac{\text{Diferencia en costo total}}{\text{Diferencia en produccion}}$$

- Costo fijo (CF):

$$CF = \text{Costos totales} - (CVU * \text{unidades producidas})$$

- d) Análisis costo – volumen – utilidad:** “Permite a los administradores analizar los efectos sobre las utilidades de cambiar los precios de ventas, analizar los efectos sobre las utilidades de cambiar los costos, analizar los efectos sobre las utilidades de cambiar el volumen, establecer precios de ventas, seleccionar la mezcla de productos para vender y elegir las estrategias de marketing” (Warren, y otros, 2010 pág. 138)

Para este análisis, se emplea los siguientes indicadores:

- Margen de contribución unitario (MCU):

$$MCU = \text{ventas} - \text{costo variable unitario}$$

- Razón margen de contribución (RMCU):

$$RMCU = \frac{\text{Margen de contribucion unitario}}{\text{Ventas}}$$

- Ventas de equilibrio (unidades) (VEU):

$$VEU = \frac{\text{Costos fijos}}{\text{Margen de contribucion unitario}}$$

- Ventas de equilibrio (nuevos soles) (VEM):

$$VEM = \frac{\text{Costos fijos}}{\text{Razón margen de contribucion unitario}}$$

- Ventas objetivo (unidades) (VOU):

$$VOU = \frac{\text{costos fijos} + \text{utilidad objetivo}}{\text{margen de contribucion unitario}}$$

- Ventas objetivo (nuevos soles) (VOM):

$$VOM = \frac{\text{costos fijos} + \text{utilidad objetivo}}{\text{razon de margen de contribuion unitario}}$$

- Apalancamiento operativo (AO):

$$AO = \frac{\text{Margen de contribucion}}{\text{Utilidad de la operacion}}$$

- Margen de seguridad (MS):

$$MS = \frac{\text{Ventas} - \text{punto de equilibrio (monto)}}{\text{ventas}}$$



- e) Costeo variable y análisis de segmentos de mercado:** “El costeo variable se usa con frecuencia para realizar análisis a corto plazo de segmentos de mercados. Mediante este análisis, se puede determinar la rentabilidad por territorios de ventas, rentabilidad del producto y rentabilidad del vendedor” (Warren, y otros, 2010 pág. 189).

Para los análisis de segmentos de mercado, se emplea el siguiente indicador:

- Razón margen de contribución (RMC):

$$RMC = \frac{\text{Margen de contribucion}}{\text{Ventas}}$$

- f) Costeo basado en actividades:** “Este método se enfoca en el costo de actividades y después asigna esos costos a los productos al usar una gran variedad de base de actividad” (Warren, y otros, 2010 pág. 453).

Para este tipo de costeo, se emplea el siguiente indicador:

- Tasa por actividad

Por otro lado, Horngren, Sundem, y Stratton (2010), afirman que la contabilidad administrativa se estudia desde las siguientes perspectivas:

- a) Comportamiento del costo y relaciones costo – volumen:** Desde esta perspectiva, se explica “cómo influyen los causantes del costo el comportamiento de este, y también se muestra cómo afectan las variaciones en los niveles de actividad de los causantes del costo a los costos variables y fijos. También, se describe el punto de equilibrio en importe y cantidad” (Horngren, y otros, 2007 pág. 42)

Para ello, se usan los siguientes indicadores:

- Contribución marginal unitaria (CMU)

$$CMU = \text{Precio de venta unit.} - \text{Costo variable unit.}$$

- Contribución marginal total (CMT)

$$CMT = \text{Unidades vendidas} * CMU$$

- Porcentaje del costo variable (CV%)

$$CV\% = \frac{\text{Costos variables totales}}{\text{ventas totales}}$$

- Porcentaje de contribución marginal (CM%)

$$CM\% = \text{Precio de venta} - \text{Gastos variables (\%)}$$

- Utilidad neta (UN)

$$UN = \text{ventas} - \text{costos fijos} - \text{costos variables}$$

- Volumen de punto de equilibrio en unidades (VPEU)

$$VPEU = \frac{\text{costos fijos}}{\text{contribucion marginal por unidad}}$$

- Volumen de punto de equilibrio en importe (VPEI)

$$VPEI = \frac{\text{costos fijos}}{\text{razon de contribucion marginal}}$$

**b) Presupuestación de capital:** Mediante esta perspectiva, “se describen las decisiones de presupuestación del capital, y también para evaluar proyectos mediante el uso del análisis de sensibilidad” (Horngren, y otros, 2007 pág. 470).

**c) Asignación del costo:** Desde esta perspectiva, se fijan costos fijos y variables a los diversos departamentos organizacionales empleando métodos directo y escalonado de asignación de costo (Horngren, y otros, 2007 pág. 520).

Para la asignación de costos, se emplean los siguientes indicadores:

- Valor de ventas en el punto de separación
- Asignación de costos conjuntos

Desde el punto de vista de Garrison, Noreen, y Brewer (2007), la contabilidad administrativa se estudia desde las siguientes perspectivas:

**a) Costeo por órdenes:** Costeo utilizado cuando se producen múltiples productos diferentes en simultaneo. Por ejemplo, una fábrica de ropa confeccionaría muchos modelos de polos tanto para hombres como para mujeres durante un mes (Garrison, y otros, 2007 pág. 95).

Para los autores, se usan los siguientes indicadores:

- Tasa predeterminada de costos indirectos
- Costeo indirecto para un trabajo en particular
- Costos indirectos sub aplicados

**b) Costeo por procesos:** Costeo utilizado cuando se produce muchas unidades de un solo producto. El enfoque básico del costeo por procesos consiste en acumular los costos de una operación o un departamento en particular durante todo un periodo (mes, trimestre, año), y luego dividir este costo total entre la cantidad de unidades producidas durante el periodo” (Garrison, y otros, 2007 pág. 94).

Este tipo de costeo se mide por el siguiente indicador:

- Costo por unidades de proceso
- Unidades equivalentes de producción

- c) Comportamiento de costos:** Hace referencia a cómo reaccionan los costos cuando varían los niveles de actividad (Garrison, y otros, 2007 pág. 190).

En esta sección se hace uso de los siguientes indicadores:

- Costo variable por unidad (CVU):

$$CVU = \frac{\text{Cambio en el costo}}{\text{Cambio en la actividad}}$$

- Elemento de costo fijo (ECF):

$$ECF = \text{Costo total} - \text{elemento de costo variable}$$

- d) Relaciones costo – volumen – utilidad:** “Es una herramienta que los administradores tienen a su servicio que les ayuda a entender las relaciones entre costo, volumen y utilidades en una organización” (Garrison, y otros, 2007 pág. 236).

Estas relaciones se miden con los siguientes indicadores:

- Margen de contribución (MC):

$$MC = \text{Precio de venta} - \text{costo variable}$$

- Margen de contribución porcentual (MCP):

$$MCP = \frac{\text{Margen de contribucion}}{\text{Ventas}}$$

- Utilidad de la operación (UO):

$$UO = (\text{Ventas} - \text{Gastos variables}) - \text{Gastos fijos}$$

- Punto de equilibrio en unidades vendidas (PEUV):

$$PEUV = \frac{\text{Gastos fijos}}{\text{Margen de contribucion por unidad}}$$

- Punto de equilibrio en unidades monetarias (PEUM):

$$PEUM = \frac{\text{Gastos fijos}}{\text{Margen de contribucion porcentual}}$$

- Ventas por unidades para conseguir la utilidad meta (VUUN):

$$VUUN = \frac{\text{Gastos fijos} + \text{Utilidad meta}}{\text{Margen de contribucion por unidad}}$$

- Unidades para alcanzar la utilidad meta (UUM):

$$UUM = \frac{\text{Gastos fijos} + \text{Utilidad meta} / (1 - \text{tasa})}{\text{Margen de contribucion unitario}}$$

- Margen de seguridad (MS):

$$MS = \text{Total de las ventas} - \text{Ventas del pto. de equilibrio}$$

- Porcentaje del margen de seguridad (PMS):

$$PMS = \frac{\text{Margen de seguridad en unidades monetarias}}{\text{Total de las ventas presupuestadas (o reales)}}$$

- Grado de apalancamiento operativo (GAO):

$$GAO = \frac{\text{Margen de contribucion}}{\text{Utilidad de la operacion}}$$

- e) Costeo variable:** Este tipo de costeo toma en cuenta los costos que varían en función de la producción. Tales costos pueden ser materias primas, mano de obra y otros gastos sólo si estos son utilizados directamente para producir un producto. (Garrison, y otros, 2007 pág. 284).

Para este costeo, se usan los siguientes indicadores:

- Costo unitario de producción
- Margen de contribución

**f) Costeo por actividades:** Proporciona información acerca de costos de producción que no afectan costos fijos y capacidad operativa, y que facilitan la toma de decisiones a los administradores. (Garrison, y otros, 2007 pág. 322).

Por otro lado, Lazo (2013) señala que la contabilidad se puede analizar desde las siguientes perspectivas:

**a) Costo – Volumen – Utilidad:** “Este análisis permite conocer los cambios en el nivel de los ingresos y los costos solo se producen debido a variaciones en la cantidad de unidades producidas y ventas” (Lazo Palacios, 2013 pág. 97).

Según el autor, este análisis presenta los siguientes indicadores:

- Utilidad de la operación (UO):

$$UO = \text{Total de ingresos operativos} - \text{Costos operativos}$$

- Utilidad neta (UN):

$$UN = \text{Utilidad de la operación} - \text{Impuesto sobre la renta}$$

- Contribución marginal unitaria (CMU):

$$CMU = \text{precio de venta} - \text{costo variable por unidad}$$

- Contribución marginal (CM):

$$CM = CMU * \text{unidades vendidas}$$

- Porcentaje de contribución marginal (%CM):

$$\%CM = \frac{\text{Contribucion marginal unitaria}}{\text{precio de venta}}$$

- CVU en unidades (CVUU):

$$CVUU = \frac{\text{Utilidad objetivo} + \text{Costos fijos totales}}{\text{Margen de contribucion} * \text{unidades}}$$

- CVU en cantidades (CVUC):

$$CVUC = \frac{1 - \%IR + \text{Costos fijos totales}}{\text{Coeficiente del margen de contribucion}}$$

- b) Punto de equilibrio:** “Es la cantidad de producción con la que los ingresos totales igualan a los costos totales; es decir, con la que la utilidad de operación es cero” (Lazo Palacios, 2013 pág. 105).

Para su análisis, el autor menciona los siguientes indicadores:

- Punto de equilibrio en unidades (PEU):

$$PEU = \frac{\text{Costos fijos totales}}{\text{Precio unitario} - \text{Costo variable unitario}}$$

- Punto de equilibrio en cantidades (PEC):

$$PEC = \frac{\text{Costos fijos totales}}{\text{Coeficiente del margen de contribucion}}$$

- Margen de seguridad (MS):

$$MS = \frac{\text{Ventas esperadas} - \text{Ventas pto. equilibrio}}{\text{Ventas esperadas}}$$

- c) Planeación financiera:** “Proceso continuo de elaborar proyecciones y toma sistemáticamente decisiones a corto, mediano y largo plazo, con el mejor conocimiento que se pueda tener sobre el futuro y organizar los esfuerzos necesarios para hacer que estas decisiones se cumplan” (Lazo Palacios, 2013 pág. 113)

Para el autor, la planeación financiera se mide en base a lo siguiente:

- Balance general
- Proyección de ventas
- Proyección de egresos

- Proyección de la depreciación
- Proyección de las inversiones a realizar

**d) Planificación empresarial:** “Son las provisiones de los recursos financieros, materiales y humanos que realiza toda empresa, con la finalidad de enfrentar con éxito a la competencia, dentro de un proceso de globalización de mercados” (Lazo Palacios, 2013 pág. 118).

Para el autor, la planificación empresarial se mide tomando en cuenta lo siguiente:

- Presupuesto de ventas
- Presupuesto de compras
- Presupuesto de cobranza
- Presupuesto de caja

## **1.4. Formulación del problema**

### **1.4.1. Problema general**

- ¿De qué forma un sistema de información influye en la contabilidad administrativa de la empresa Corporación Arzapalo S. A. C?

### **1.4.2. Problema específico**

- ¿En qué medida un sistema de información para la contabilidad administrativa influye en las unidades de equilibrio por producto de la empresa Corporación Arzapalo S.A.C.?
- ¿De qué forma un sistema información para la contabilidad administrativa influye en la rentabilidad del producto por territorios de ventas de la empresa Corporación Arzapalo S.A.C.?



## **1.5. Justificación del estudio**

### **1.5.1. Justificación tecnológica**

Cohen y Asín (2000, pág. 64), considera que “el uso de las tecnologías de información apoya o da forma a la estrategia competitiva de la organización, a su plan para mantener o incrementar la ventaja competitiva o bien, para reducir la ventaja de sus rivales”.

A través de la presente investigación, se desarrolló una plataforma que permite iniciar la integración con el sistema de ventas de la. Adicional a ello, esta investigación permitió el uso de herramientas de software libre compatibles con las herramientas implementadas con anterioridad.

### **1.5.2. Justificación económica**

Es posible considerar a la tecnología de sistemas de información como un factor de producción que se puede sustituir por el capital o el trabajo tradicional, dando como resultado la reducción en el número de gerentes del nivel intermedio y oficinistas, en cuanto la tecnología de información los sustituye en sus labores. (Laudon, y otros, 2008 pág. 90)

A través de esta investigación, se logró la reducción de costos de hombre/hora en un 22%, producto del sobretiempo de los colaboradores encargados en la generación de los reportes contables (ver anexo 07).

### **1.5.3. Justificación institucional**

Para Cohen y Asín (2000, pág. 65), “los sistemas de información aportan beneficios mediante el incremento del volumen de ventas, mejora del servicio al cliente, aumento de la productividad, reducción de costos y mejora de la eficiencia en el manejo de los recursos económicos de la empresa”.

Esta investigación logró incrementar la confiabilidad de la información demandada por la Alta Gerencia, y como consecuencia de ello, la toma de decisiones se realiza con sustento cuantitativo, el cual se ve reflejado con mejores promociones hacia los clientes y la percepción de ellos sobre un precio justo.

#### **1.5.4. Justificación operativa**

Para Laudon y Laudon (2008, pág. 8), “los sistemas y las tecnologías de información son algunas de las herramientas disponibles más importantes para que los gerentes consigan niveles de eficiencia y productividad más altos en las operaciones empresariales, especialmente cuando se conjuntan con cambios en las prácticas de negocios y el comportamiento administrativo”.

Con la presente investigación, se logró la fluidez de la información y compartir la misma con los colaboradores del área contable y la Administración, el cual permitió la visualización de la información recopilada por el sistema oportunamente para elaborar estrategias tácticas de ventas.

### **1.6. Hipótesis**

#### **1.6.1. Hipótesis general**

- El sistema de información optimiza la contabilidad administrativa de la empresa Corporación Arzapalo S.A.C.

#### **1.6.2. Hipótesis específica**

- El sistema de información para la contabilidad administrativa reduce las unidades de equilibrio por producto de la empresa Corporación Arzapalo S.A.C.
- El sistema información para la contabilidad administrativa incrementa la rentabilidad del producto por territorios de ventas de la empresa Corporación Arzapalo S.A.C.

## **1.7. Objetivos**

### **1.7.1. Objetivo general**

- Determinar la influencia de un sistema de información en la contabilidad administrativa de la empresa Corporación Arzapalo S.A.C

### **1.7.2. Objetivos específicos**

- Determinar la influencia de un sistema de información para la contabilidad administrativa en las unidades de equilibrio por producto de la empresa Corporación Arzapalo S.A.C.
- Determinar la influencia de un sistema de información para la contabilidad administrativa la de rentabilidad del producto por territorios de ventas de la empresa Corporación Arzapalo S.A.C.

## CAPÍTULO II: MÉTODO

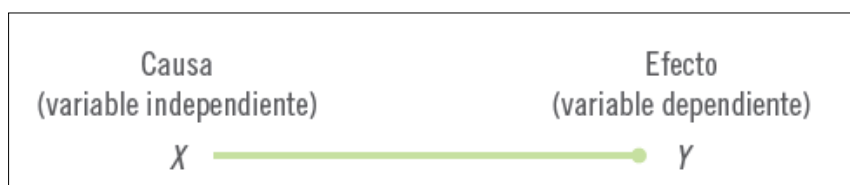
### 2.1. Diseño de investigación

El diseño no es otra cosa que la estrategia utilizada para obtener la información deseada (Hernández, y otros, 2010 pág. 120).

Por otra parte, (Arias, 1999 pág. 20), señala que el diseño de la investigación “es la estrategia que adopta el investigador para responder al problema planteado”.

Para la presente investigación, se adoptó el diseño Experimental, debido a que el presente proceso “consiste en someter a un objeto o grupo de individuos a determinadas condiciones o estímulos (variable independiente), para observar los efectos que se producen (variable dependiente) y se diferencia de la investigación de campo por la manipulación y control de variables” (Arias, 1999 pág. 21).

**Figura 11.** *Relación entre la variable independiente y dependiente en el diseño experimental*



**Fuente:** Hernández, Baptista, & Fernández (2010)

Como se puede apreciar en la figura 11, para el diseño experimental existe una relación de causa – efecto entre la variable independiente y la variable dependiente, es decir, busca medir en qué medida la variable independiente influye en la variable dependiente.

Dentro del diseño experimental, se eligió el diseño pre - experimental debido a que es “un diseño de un solo grupo cuyo grado de control es mínimo, es decir, que a un grupo se le aplica una prueba previa al estímulo o tratamiento experimental (pre – test), después se le administra el

tratamiento y finalmente se le aplica una prueba posterior al estímulo (post – test)” (Hernández, y otros, 2010 pág. 136).

Para la presente tesis se usó la siguiente relación:

$$G : O_1 \quad X \quad O_2$$

- *G: Grupo Control de la variable dependiente contabilidad administrativa.*
- *O1: Resultados del pretest, sin la implementación del Sistema de información.*
- *X: Sistema de información para la contabilidad administrativa.*
- *O2: Resultados del pretest, empleando el Sistema de información para la contabilidad administrativa.*

## **2.2. Variables, operacionalización**

### **2.2.1. Definición conceptual**

#### **A. Variable independiente: Sistema de información**

“Conjunto de componentes interrelacionados que recolectan (o recuperan), procesan, almacenan y distribuyen información para apoyar la toma de decisiones y el control de una organización” (Laudon, y otros, 2008 pág. 14)

#### **B. Variable dependiente: Contabilidad administrativa**

Para el presente trabajo de investigación se usaron los planteamientos de Garrison, Noreen, y Brewer (2007) y de Warren, Reeve, y Duchac (2012). Esto debido a que tuvieron concordancia con la problemática planteada en la presente tesis, que son la determinación de las unidades de equilibrio mediante el uso del análisis costo – volumen – utilidad con el indicador punto de equilibrio en unidades, y la rentabilidad del producto por territorios de ventas, a través del análisis de segmentos de mercado empleando el indicador la razón margen de

contribución, ambos aplicados al área de ventas de una empresa comercial.

Según Warren, Reeve, y Duchac (2010, pág. 132), la comprensión del comportamiento del costo depende de dos factores:

- Identificar las actividades que causan el cambio del costo. Dichas actividades se llaman base de actividad o generadores de actividad.
- Especificar el rango de actividad sobre el cual son de interés los cambios en el costo. Este rango de actividad se llama rango relevante.

**Tabla 4.** *Tipo de empresa y su base de actividad*

<b>Tipo de empresa</b>	<b>Base de actividad</b>
Universidad	Número de clases
Aerolínea comercial	Número de millas en vuelo
Manufacturera	Cantidad de unidades producidas
Comercial	Cantidad de unidades adquiridas
Hospital	Número de pacientes
Hotel	Número de huéspedes
Banco	Número de transacciones bancarias.

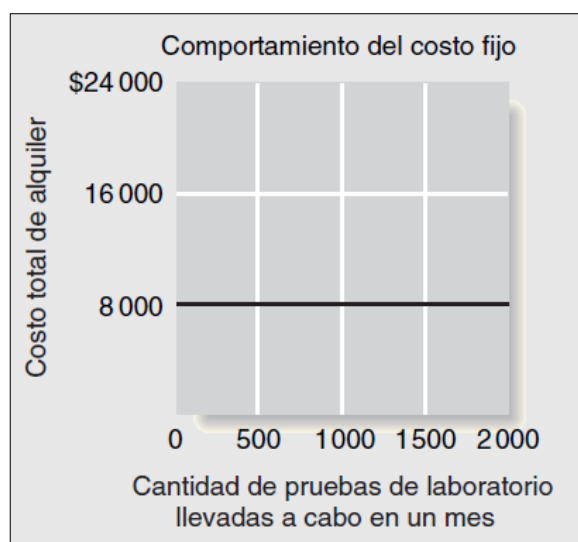
Fuente: Warren, Reeve, y Duchac (2010)

La tabla 4 muestra el tipo de empresa y su base de actividad respectiva. Como se puede apreciar, la base de actividad para una empresa comercial son las unidades adquiridas.

Para entender el comportamiento del costo, es necesario definir los tipos de costos existentes como son los costos fijos, los costos variables y los costos mixtos.

Por costos fijos se entiende que “aquellos que permanecen constantes en cantidad monetaria total conforme cambia la base de actividad. Cuando la base de utilidad es unidades producidas, muchos costos indirectos, como la depreciación en línea recta, se clasifican como costos fijos” (Warren, y otros, 2010 pág. 134).

**Figura 12.** *Gráfica del comportamiento de los costos fijos*

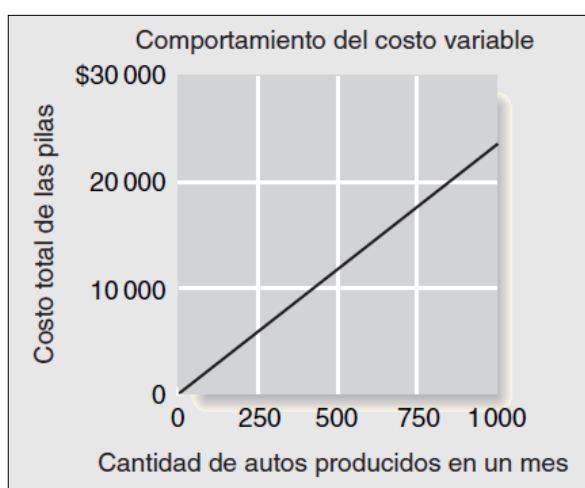


**Fuente:** Garrison, Noreen, & Brewer (2007)

La figura 12 muestra en forma gráfica el comportamiento del costo fijo. Como se puede apreciar, el costo fijo se mantiene estable sin importar la cantidad de pruebas realizadas en un determinado mes. Es decir, es independiente a las pruebas de laboratorio realizadas.

Con relación a los costos variables, Warren, Reeve, y Duchac (2010) señalan que “son aquellos que se modifican en proporción a los cambios en la base de actividad”.

**Figura 13.** *Gráfica del comportamiento del costo variable*

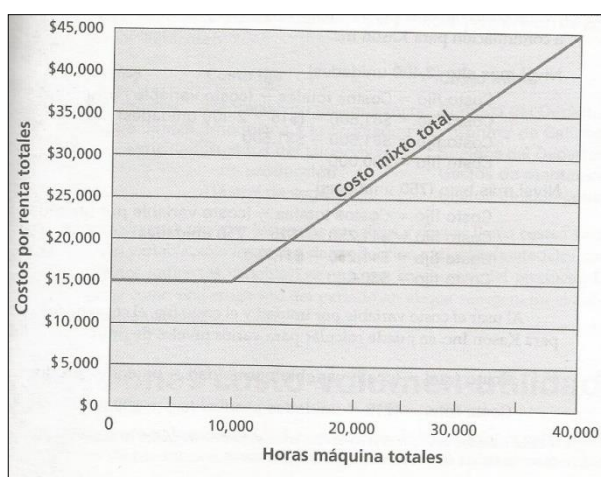


**Fuente:** Garrison, Noreen, y Brewer (2007)

La figura 13 muestra en forma gráfica el comportamiento de los costos variables. Como se puede apreciar, a medida que se producen autos en el mes, aumenta el costo total de las pilas, es decir, el costo total de las pilas depende de la producción de autos.

Por último, los costos mixtos son “aquellos que tienen características tanto del costo variable como del costo fijo” (Warren, y otros, 2010 pág. 135).

**Figura 14.** Gráfica del comportamiento de los costos mixtos



**Fuente:** Warren, Reeve, y Duchac (2010)

La figura 14 muestra en forma gráfica el comportamiento de los costos mixtos. Para el ejemplo mostrado, el costo por renta total se mantiene fijo hasta consumir las 10 mil horas/máquinas. Una vez consumido ello, el costo total por renta dependerá de las horas máquinas consumidas.

Para calcular los costos variables, Garrison, Noreen, y Brewer (2007) propone 3 métodos:

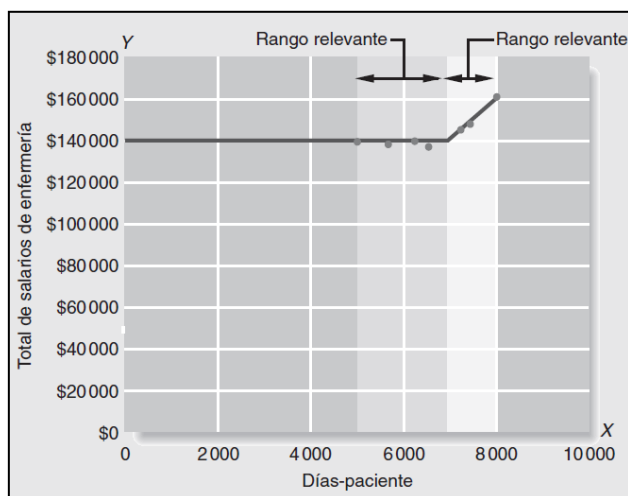
- **Método por diagramas de dispersión:**

Garrison, Noreen, y Brewer (2007, pág. 206) lo define como un método rápido e inusual utilizado en



estimaciones de los componentes de los costos mixtos (costos fijos y costos variables).

**Figura 15.** *Análisis del costo variable por diagramas de dispersión*



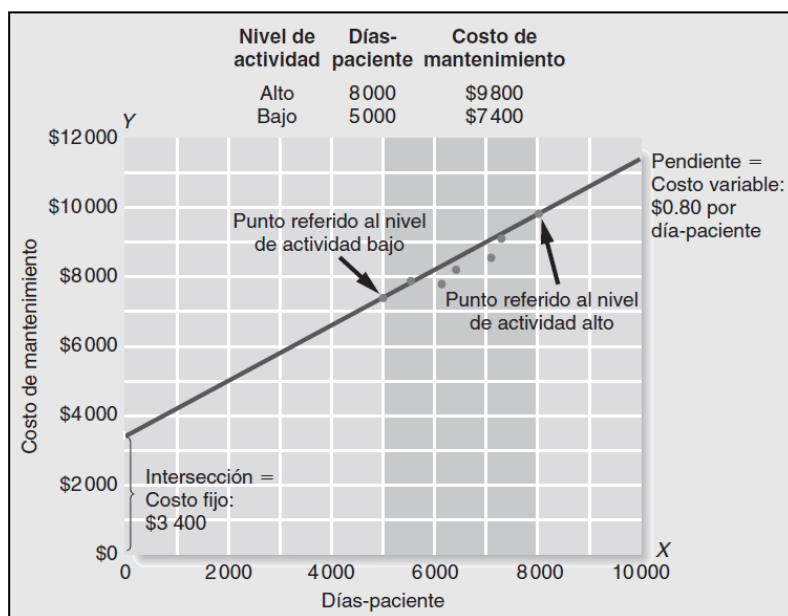
**Fuente:** Garrison, Noreen, & Brewer (2007)

La figura 15 muestra el método por diagramas de dispersión para una empresa de salud. En ella se aprecia que en el eje Y se colocan los costos de mantenimiento (variable dependiente) y en el eje X la actividad, que para el ejemplo es días – paciente (variable independiente).

- **Método del punto alto – punto bajo**

El método punto alto-punto bajo, en forma práctica, identifica el periodo con menor nivel de actividad y el periodo con mayor nivel de actividad para determinar la pendiente de una línea recta, ya que actúa como la relación las variables independiente y dependiente. (Garrison, y otros, 2007 pág. 208).

**Figura 16.** Método del punto alto – punto bajo de análisis de costos



**Fuente:** Garrison, Noreen, & Brewer (2007)

La figura 16 muestra en forma gráfica el método del punto alto – punto bajo para una clínica. En ella se puede observar el trazo de una línea recta entre los puntos correspondientes a los niveles alto y bajo de actividad.

Para Garrison, Noreen, y Brewer, 2007, pág. 208), el costo variable está representado, matemáticamente, por la pendiente de la recta y, para estimar el costo variable, se emplea la siguiente la fórmula algebraica:

$$\text{Costo variable (CV)} = \text{pendiente de la recta} = \frac{Y_2 - Y_1}{X_2 - X_1}$$

De la relación se puede concluir lo siguiente:

$$CV = \frac{\text{Costo}_{\text{nivel gran actividad}} - \text{Costo}_{\text{nivel baja actividad}}}{\text{Alto nivel de actividad} - \text{Bajo nivel de actividad}}$$

En pocas palabras:

$$\text{Costo variable} = \frac{\text{Cambio en el costo}}{\text{Cambio en la actividad}}$$

Tomando como base la relación planteada por Garrison, Noreen, y Brewer (2007, pág. 209) sobre el costo variable y la base de actividad de la empresa en estudio, se llegó a la siguiente relación:

$$CVU = \frac{\text{Costo total}_{m\acute{a}x} - \text{Costo total}_{m\acute{i}n}}{\text{Unidades compradas}_{m\acute{a}x} - \text{Unidades compradas}_{m\acute{i}n}}$$

Para el cálculo del costo variable, Warren, Reeve, y Duchac (2010, pág. 134) recomienda utilizar el método punto alto – punto bajo.

Los costos fijos (CF) se estima al restar los costos variables totales de los costos totales por las unidades producidas, como se muestran a continuación:

$$CF = \text{costos totales}_{m\acute{a}x \text{ o } m\acute{i}n} - (CVU * \text{unid. producidas}_{m\acute{a}x \text{ o } m\acute{i}n})$$

Con relación al análisis de segmentos de mercado, se utilizó el costeo variable o directo. En este tipo de costeo, “el costo de producción incluye solamente los costos de producción variable, es decir, en los materiales directos, mano de obra directa y costos indirectos variables” (Warren, y otros, 2010 pág. 179).

Para complementar ello, Garrison, Noreen, y Brewer (2007, pág. 284) menciona que el costeo variable es muy utilizado por los administradores para la toma de decisiones internas y debe utilizarse para preparar el balance de resultados de acuerdo con el enfoque de margen de contribución.

Warren, Reeve, y Duchac (2010, pág. 189) menciona también que los “administradores deben usar el costeo variable para

evaluar el rendimiento de la operación y que con frecuencia el costeo variable se usa para análisis de mercado a corto plazo”.

Para ello, Warren, Reeve, y Duchac, (2010, pág. 190) dividen el análisis de segmentos de mercados en: análisis de rentabilidad por territorios de ventas, el análisis de rentabilidad del producto y el análisis de rentabilidad del vendedor.

Según Warren, Reeve, y Duchac, (2010, pág. 190), “un análisis de rentabilidad por territorios de ventas con frecuencia se usa para evaluar el desempeño pasado y dirigir es esfuerzo en ventas futuras a través del margen de contribución”. El mismo autor añade que este análisis “puede conducir a la administración a hacer lo siguiente: reducir los costos en territorios de ventas de ganancias menores o aumentar los esfuerzos de ventas en territorios de ganancias mayores” (Warren, y otros, 2010 pág. 190).

**Figura 17.** *Reporte del margen de contribución por territorios de ventas*

Camelot Fragrance Company Margen de contribución por territorio de ventas para el mes terminado el 31 de marzo 2010			
	Territorio Norte		Territorio Sur
Ventas .....		\$80,000	\$80,000
Costo de ventas variable.....		<u>9,600</u>	<u>9,600</u>
Margen de producción .....		\$70,400	\$70,400
Gastos de venta variables:			
Costos de promoción.....	\$22,000		\$19,000
Comisiones por ventas .....	<u>14,000</u>	<u>36,000</u>	<u>30,000</u>
Margen de contribución .....		<u>\$34,400</u>	<u>\$40,400</u>
Razón de margen de contribución.....		<u>43%</u>	<u>50.5%</u>

**Fuente:** Warren, Reeve, & Duchac (2010)

La figura 17 muestra un ejemplo del reporte de análisis de rentabilidad por territorios de ventas, la misma que compara entre dos territorios las ventas para un determinado periodo (mes de marzo para el ejemplo), el margen de producción y los gastos de ventas variables. Como resultado de ello, se obtiene la

razón margen de contribución de cada territorio de venta, de donde se puede concluir que el territorio sur es más rentable que el territorio norte para el ejemplo propuesto por el autor.

Con respecto al análisis de rentabilidad del producto, Warren, Reeve, y Duchac (2010, pág. 191) menciona que “una empresa debe enfocar sus esfuerzos de ventas sobre productos que proporcionan el mayor margen de contribución. Para ello, con frecuencia se suele usar el análisis de rentabilidad del producto para tomar decisiones en cuanto a ventas de producto esfuerzos promocionales”.

**Figura 18.** Reporte del margen de contribución por la línea de producto

Camelot Fragrance Company				
Margen de contribución por línea de producto				
para el mes finalizado el 31 de marzo 2010				
	Gwenevere		Lancelot	
Ventas .....		\$90,000		\$70,000
Costo de ventas variable.....		<u>10,800</u>		<u>8,400</u>
Margen de producción .....		\$79,200		\$61,600
Gastos de venta variables:				
Costos de promoción.....	\$27,000		\$14,000	
Comisiones por ventas .....	<u>18,000</u>	<u>45,000</u>	<u>7,000</u>	<u>21,000</u>
Margen de contribución .....		<u>\$34,200</u>		<u>\$40,600</u>
Razón de margen de contribución.....		<u>38%</u>		<u>58%</u>

**Fuente:** Warren, Reeve, y Duchac (2010)

La figura 18 muestra un ejemplo del reporte del margen de contribución por la línea de producto e indica que la razón margen de contribución de Lancelot es mayor que la de Gwenevere, debido que los costos de promoción y comisiones por ventas son más bajos. Teniendo en cuenta ello, según Warren, Reeve, y Duchac (2010, pág. 192), “los administradores deben considerar en enfatizar los planes de marketing en Lancelot, reducir los costos de promoción y de comisiones por ventas de Gwenevere o aumentar el precio de venta Gwenevere”.

Por otro lado, un reporte de rentabilidad del vendedor es útil para evaluar el desempeño de los vendedores, el cual incluye las ventas totales, costo de ventas variable, gastos de ventas variables, margen de contribución y a razón margen de contribución para cada vendedor.

**Figura 19.** Reporte del margen de contribución por vendedor

Camelot Fragrance Company				
Margen de contribución por vendedor-Territorio Norte para el mes finalizado el 31 de marzo 2010				
	Inés Rodríguez	Tom Ginger	Beth Williams	Total territorio Norte
Ventas .....	\$20,000	\$20,000	\$40,000	\$80,000
Costo de ventas variable .....	2,400	2,400	4,800	9,600
Margen de producción .....	<u>\$17,600</u>	<u>\$17,600</u>	<u>\$35,200</u>	<u>\$70,400</u>
Gastos de venta variables:				
Costos de promoción .....	\$ 5,000	\$ 5,000	\$12,000	\$22,000
Comisiones por ventas .....	<u>3,000</u>	<u>3,000</u>	<u>8,000</u>	<u>14,000</u>
	<u>\$ 8,000</u>	<u>\$ 8,000</u>	<u>\$20,000</u>	<u>\$36,000</u>
Margen de contribución .....	<u>\$ 9,600</u>	<u>\$ 9,600</u>	<u>\$15,200</u>	<u>\$34,400</u>
Razón de margen de contribución ...	<u>48%</u>	<u>48%</u>	<u>38%</u>	<u>43%</u>
				<u>25%</u>

**Fuente:** Warren, Reeve, y Duchac (2010)

La figura 19 muestra un ejemplo del reporte de margen de contribución por vendedor. En ella se aprecia que Beth Williams produjo el mayor margen de contribución, pero tuvo la menor razón de margen de contribución. Este reporte permite al administrador del territorio Norte para alentar a Inés a Tom a vender más productos o aumentar las comisiones de ventas.

Para efectos de cálculos, Warren, Reeve, y Duchac (2010, pág. 187) señalan que los costos variables se calculan de la siguiente manera:

$$\text{Costos variables} = \text{unidades vendidas} * \text{CVU (periodo analizado)}$$

En donde:

- CVU hace referencia al costo variable unitario.

Por otra parte, el mismo autor señala que el margen de producción es llamado margen de adquisición o margen de compras para empresas no manufactureras (pág. 188).

### **2.2.2. Definición operacional**

#### **A. Sistema de información**

Un sistema de información consta de una validación de acceso al sistema, consulta de datos para generar reportes de las unidades de equilibrio por productos y la razón margen de contribución por territorios de ventas.

#### **B. Contabilidad administrativa**

Es el proceso en la cual la información recabada por el sistema de información se resume con la finalidad de proveer, al gerente general, de información confiable y oportuna para la correcta toma de decisiones.

## **2.3. Población y muestra**

### **2.3.1. Población**

“Conjunto de elementos que consiste en personas, objetos, etc. En los que se pueden observar o medir una o más características de naturaleza cualitativa o cuantitativa” (Córdova, 2009, p. 2).

Conforme con la definición dada, se distribuyó la población de la siguiente manera:

- Para el indicador “ventas de equilibrio por producto”, se tomó población los 42 productos que la empresa ofrece a la clientela.
- Para el indicador “razón margen de contribución”, se tuvo como población a 24 distritos pertenecientes a Lima

Metropolitana y Callao en donde se tiene presencia de ventas como lo muestra la siguiente tabla:

**Tabla 5. Relación de distritos objetivos de la empresa Corporación Arzapalo S. A. C.**

N°	Nombre del distrito
1	Ancon
2	Ate - Vitarte
3	Breña
4	Cercado de Lima
5	Chorrillos
6	Jesus Maria
7	La Molina
8	La victoria
9	Lince
10	Lurin
11	Magdalena del mar
12	Miraflores
13	San Borja
14	San Isidro
15	San Juan de Lurigancho
16	San Juan de Miraflores
17	San Luis
18	San Martin de Porres
19	San Miguel
20	Santiago de Surco
21	Surquillo
22	Villa el salvador
23	Villa Maria del triunfo
24	Callao

Fuente: Elaboración propia (2015)

La tabla 5 muestra la relación de los distritos de Lima Metropolitana y el Callao en donde la empresa posee ventas.

### 2.3.2. Muestra

La muestra a una porción de la población que se selecciona de acuerdo con reglas establecidas con la finalidad de obtener información acerca de la población que lo contiene (Córdova, 2003 pág. 2).



Por su parte Hernández citado en Castro (2011, pág. 69) determina que si se tiene una población inferior a cincuenta (50) elementos, la población y la muestra son de igual tamaño.

En la siguiente tabla se presenta la relación entre la población y su respectiva muestra:

**Tabla 6.** *Distribución de la muestra*

N°	Tipo de población	Tamaño de la población	Tamaño de la muestra
1	Productos ofertados por la empresa	42 productos	42 productos
2	Distritos objetivos de la empresa	24 distritos	24 distritos

Fuente: Elaboración propia (2015)

La tabla 6 muestra el tamaño de la población y de la muestra para cada tipo de población escogidos para el presente trabajo de investigación, la cual fue tomada de acuerdo con lo establecido por Castro (2011).

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

### 2.4.1. Técnicas

“Recolectar los datos implica elaborar un plan detallado de procedimientos que nos conduzcan a reunir datos con un propósito específico” (Hernández, y otros, 2010 pág. 198).

Por ese motivo, Del Cid, Méndez, y Sandoval (2007, pág. 94), define que es un conjunto estructurado de instrucciones relacionados con el objeto estudiado y con los recursos utilizables que conducen a generar información oportuna para una determinada investigación.

Carrasco (2005, pág. 274), señala que “las técnicas como herramientas procedimentales y estratégicas suponen un previo

conocimiento en cuanto a su utilidad y aplicación, de tal manera que seleccionarlas y elegir las resulte una tarea fácil al investigador”.

Para la presente tesis, se usaron las siguientes técnicas de recolección de datos:

#### **A. Entrevista**

Para Hernández, Baptista, y Fernández (2010, pág. 418), “esta se define como una reunión para conversar e intercambiar información entre una persona (el entrevistador) y otra (el entrevistado) u otras (entrevistados)”.

“La entrevista puede ser presentado en un esquema o estructura metódica secuencialmente organizado o simplemente a través de una hoja con preguntas, elaborado de acuerdo con las circunstancias y características del desenvolvimiento y desarrollo de la entrevista” (Carrasco, 2005 pág. 315).

#### **2.4.2. Instrumentos de recolección de datos**

Es aquel documento o formato donde se registra datos de las variables que el estudioso ha determinado para una investigación (Hernández, y otros, 2010 pág. 200).

Para el desarrollo de la presente tesis se utilizaron los siguientes instrumentos:

#### **A. Cuestionario**

Este instrumento se usó para recopilar información acerca de la problemática del presente estudio, el cual, a través de preguntas abiertas dirigidas hacia el Gerente de la Empresa, se pudo extraer datos para elaborar la problemática general de la empresa.

## B. Formato de revisión documental

Este instrumento se usó para recolectar datos de documentos históricos en la empresa, como las facturas y órdenes de compras.

A través de este formato, se pudieron medir los indicadores “Razón margen de contribución por territorios de ventas”.

En la siguiente tabla se muestra la relación entre los indicadores, la técnica y el instrumento a utilizar:

**Tabla 7.** *Relación entre indicador, técnica e instrumento de recolección de datos*

Indicador	Técnica	Instrumento	Fuente
Unidades de equilibrio por producto	Revisión documental	Formato de revisión documental	Reportes de compras y ventas por producto.
Razón margen de contribución	Revisión documental	Formato de revisión documental	Facturas

Fuente: Elaboración propia (2015)

La tabla 7 muestra la relación entre los indicadores del presente trabajo de investigación con la técnica de recolección de datos y su respectivo instrumento.

Las pruebas fueron realizadas en dos meses, en donde se recogieron los treinta y ocho (38) reportes de los productos proporcionados por el contador de la empresa y los treinta y tres (33) distritos de Lima estipulados anteriormente en el cálculo de la muestra.

## 2.5. Métodos de análisis de datos

Para Hernández, Baptista, y Fernández (2010, pág. 278), el análisis de los datos “se efectúa sobre la matriz de datos y se utiliza un programa computacional, tales como SPSS, Minitab, entre otros”.

Para analizar los datos de la presente tesis, se usó el software SPSS ® en su versión 20, debido a la variedad de funciones que ofrece para el presente trabajo de investigación.

### 2.5.1. Definición

- $I_a$  = Indicador del sistema actual
  - $I_d$  = Indicador del sistema propuesto
- A. Hipótesis nula:** Se denomina hipótesis nula ( $H_0$ ) a aquella que es aceptada temporalmente como verdadera y cuya validez será sometida a comprobación experimental (Córdova, 2003 pág. 418).
- B. Hipótesis alternativa:** Se denomina hipótesis alternativa ( $H_1$ ) a la hipótesis contraria a la hipótesis nula ( $H_0$ ) (Córdova, 2003 pág. 418).

### 2.5.2. Hipótesis estadísticas

- **Hipótesis  $H_1$ :** El sistema de información para la contabilidad administrativa reduce las unidades de equilibrio por producto de la empresa Corporación Arzapalo S.A.C.
  - **Hipótesis  $H_{10}$ :** El sistema de información para la contabilidad administrativa no reduce las unidades de equilibrio por producto de la empresa Corporación Arzapalo S.A.C.

$$H_{10}: I_a - I_d \leq 0 \approx I_a \leq I_d$$

- **Hipótesis  $H_{11}$ :** El sistema de información para la contabilidad administrativa reduce las unidades de equilibrio por producto de la empresa Corporación Arzapalo S.A.C.

$$H_{11}: I_a - I_d > 0 \approx I_a > I_d$$

- **Hipótesis H<sub>2</sub>:** El sistema información para la contabilidad administrativa incrementa la rentabilidad del producto por territorios de ventas de la empresa Corporación Arzapalo S.A.C.
  - **Hipótesis H<sub>20</sub>:** El sistema información para la contabilidad administrativa no incrementa la rentabilidad del producto por territorios de ventas de la empresa Corporación Arzapalo S.A.C.

$$\mathbf{H_{20}: I_{2a} - I_{2d} \leq 0 \approx I_{2a} \leq I_{2d}}$$

- **Hipótesis H<sub>21</sub>:** El sistema información para la contabilidad administrativa incrementa la rentabilidad del producto por territorios de ventas de la empresa Corporación Arzapalo S.A.C.

$$\mathbf{H_{21}: I_{2a} - I_{2d} > 0 \approx I_{2a} > I_{2d}}$$

#### A. Nivel de significancia

- $X = 5\%$  (*error*)
- *Nivel de confiabilidad* =  $1 - X = 0.95$

#### B. Estadística de prueba

##### a) Promedio

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

##### b) Varianza

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

##### c) Prueba T - Student

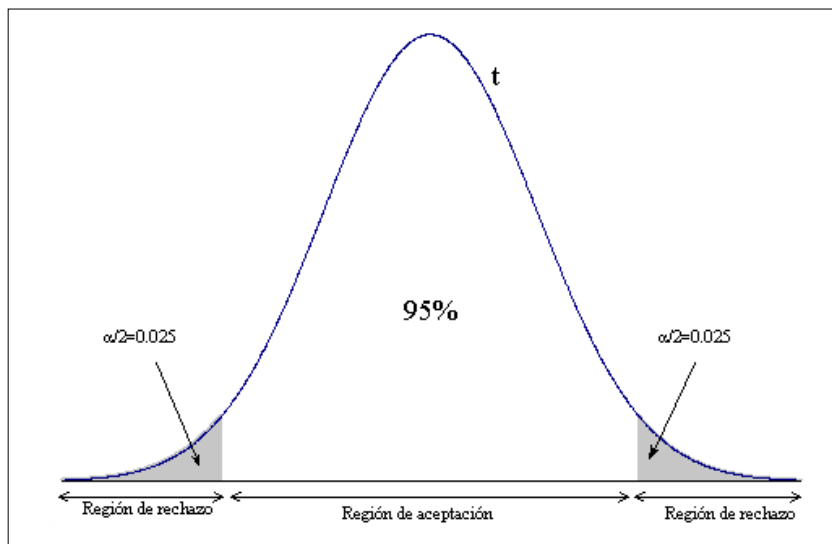
$$t = \frac{\bar{x}_1 - \bar{x}_2 - (\mu_1 - \mu_2)}{S_p \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}$$

Dónde:

$$S_p^2 = \frac{[(n_1 - 1)S_1^2 + (n_2 - 1)S_2^2]}{n_1 + n_2 - 2}$$

- $\bar{x}_1$  = Media muestra grupo 1
- $\bar{x}_2$  = Media muestra grupo 2
- $n_1$  = Numero de muestra grupo 1
- $n_2$  = Numero de muestra grupo 2
- $S_1^2$  = Varianza grupo 1
- $S_2^2$  = Varianza grupo 2

**Figura 20.** Gráfica de la distribución *t* – Student



Fuente: Recuperado de

[http://www.fisterra.com/mbe/investiga/t\\_student/t\\_student.asp](http://www.fisterra.com/mbe/investiga/t_student/t_student.asp)

La figura 20 muestra la gráfica de la distribución T – Student, en donde  $T_x$  es el valor tabular respecto al nivel de significancia. Entonces se concluye que, si  $T_x$  tabulado es menor que la  $T_x$  calculado, se acepta la hipótesis nula; en caso contrario se rechaza la hipótesis nula.

**d) Pruebas de normalidad**

Hernández, Fernández y Baptista (2010, pág. 276) señala lo siguiente:

“Todo análisis de datos cuantitativo debe seguir una secuencia estructurada la cual inicia en las pruebas de normalidad, existen principalmente dos pruebas de normalidad que regularmente son utilizadas para estudios estadísticos, la prueba de Shapiro – Wilk y la prueba de Kolmogorov – Smirnov, la principal diferencia entre ambos métodos es que el primero es utilizado para test con una muestra pequeña (menor a 50) y la segunda prueba cuando el tamaño de la muestra es exiguo (igual o mayor a 50). Si es mayor se convierte en una prueba demasiado exigente, que casi siempre lleva a la decisión de rechazar la hipótesis nula”.

- **Prueba de Kolmogorov – Smirnov**

“La prueba de Kolmogorov – Smirnov no necesita que los datos se encuentren agrupados” (Canavos, 1998, p. 368).

$$D = \max|F_n(x) - F_0(x)|$$

Dónde:

- $F_n(x)$  = función de la distribución muestral
- $F_0(x)$  = función teórica o correspondiente a la población normal

- **Prueba de Shapiro – Wilk**

Es una prueba de normalidad utilizado para muestras iguales o inferiores a 50 individuos (Mohd, y otros, 2011 pág. 25)

$$w = \frac{(\sum_{i=1}^n a_i y_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

Dónde:

- $y_i$  es el primer orden estadístico
- $\bar{y}$  es el valor de la muestra
- $a = (a_1, \dots, a_n) = \frac{m^T V^{-1}}{(m^T V^{-1} V^{-1} m)^{1/2}}$
- $m = (m_1, \dots, m_n)^T$  Valores esperados de las estadísticas de orden de variables aleatorias independientes e idénticamente distribuidas de la muestra de la distribución normal estándar y  $V$  es la matriz de covarianza de esas estadísticas de orden

Si el valor de “w” es demasiado pequeño, entonces se rechaza la hipótesis nula.

## 2.6. Aspectos éticos

En esta tesis, se protegió la identidad de los documentos e información proporcionada por la empresa Corporación Arzapalo S.A.C. Su uso y divulgación de documentos e información de índole empresarial y personal fueron gracias al consentimiento de cada una de las partes involucradas.

Por otro lado, los datos y resultados de la presente investigación no sufrieron manipulación y/o alteración alguna por ninguna de las partes comprometidas, cumpliendo así con los principios de transparencia y confiabilidad de la investigación.



## CAPÍTULO III: RESULTADOS

En el presente apartado, se presentan los resultados obtenidos de la tesis empleando los indicadores planteados a través de la implementación de un sistema de información para la contabilidad administrativa para la empresa Corporación Arzapalo S. A. C. Para ello, se usó el software SPSS en su versión 20.

Para el diseño pre – experimental se tuvo, en una primera etapa, datos cuando el sistema aún no se implementó (Pre – Test) y, en una segunda etapa, datos con el sistema ya en producción (Post – Test).

Una vez obtenido los datos, se procedieron a analizarlos de acuerdo con cada indicador y se efectuaron las pruebas para comprobar la hipótesis. Esto se realizó de la siguiente manera:

- Pruebas de normalidad
- Prueba de hipótesis
- Análisis comparativo

### 3.1. Pruebas de normalidad

Se empleó la prueba de Shapiro – Wilk. Esto se debió a que las muestras de la presente tesis son menores a 50 ( $n < 50$ ) (Mohd, y otros, 2011 pág. 25).

**Tabla 8.** *Pruebas de normalidad aplicada según la muestra obtenida*

N°	Indicador	Muestra	Prueba de normalidad
1	Unidades de equilibrio por producto	42 productos ofertados por la empresa	Shapiro - Wilk
2	Razón margen de contribución	24 distritos con mayor presencia de ventas	Shapiro - Wilk

Fuente: Elaboración propia (2015)

Como detalla la tabla 8, ambos indicadores poseen un tamaño de muestra menores a 50, por lo tanto, se aplicó la prueba de Shapiro – Wilk para determinar la normalidad de los datos.

Dichas pruebas se efectuaron ingresando los datos de cada indicador al software SPSS. Se hace énfasis que una muestra no normal se debe aplicar un análisis no paramétrico, mientras que en una muestra normal se aplica un análisis paramétrico.

A continuación, se mencionan las condiciones para aceptar que una muestra es normal o no:

- Si el sig. < 0.05, entonces la muestra adopta una distribución no normal y se aplica prueba no paramétrica.
- Si el sig. ≥ 0.05, la muestra adopta una distribución normal y se aplica la prueba paramétrica.

A continuación, se muestran los resultados arrojados por el software SPSS.

### 3.1.1. Test de Shapiro – Wilk para el primer indicador

En esta sección se describe la prueba de Shapiro – Wilk para el primer indicador “unidades de equilibrio por producto”.

**Tabla 9.** Resultado de la prueba de Shapiro – Wilk para el primer indicador

	Shapiro – Wilk		
	Estadístico	gl	Sig.
Valor de las unidades de equilibrio sin la aplicación web (Pre – Test)	0.397	42	0.000 (6.255005E-12)
Valor de las unidades de equilibrio con la aplicación web (Post – Test)	0.368	42	0.000 (3.139630E-12)
Diferencia (Pre – Test menos Post – Test)	0.968	42	0.288

Fuente: Elaboración propia (2015)

Como se observa en la tabla 9, el valor arrojado por el software SPSS (Sig.) es mayor a 0.05. Por lo tanto, se concluye que los datos siguen una distribución normal.

### 3.1.2. Test de Shapiro – Wilk para el segundo indicador

En esta sección se describe la prueba de Shapiro – Wilk para el segundo indicador “razón margen de contribución”.

**Tabla 10.** Resultados de la prueba de Shapiro – Wilk para el segundo indicador

	Shapiro – Wilk		
	Estadístico	gl	Sig.
Valor de la razón margen de contribución sin la aplicación web (Pre – Test)	0.950	24	0.277
Valor de la razón margen de contribución con la aplicación web (Post – Test)	0.874	24	0.006
Diferencia (Pre – Test menos Post – Test)	0.984	24	0.959

Fuente: Elaboración propia (2015)

Como se puede apreciar en la tabla 10, el valor arrojado por el software SPSS (Sig.) es mayor a 0.05. Por lo tanto, se concluye que los datos siguen una distribución normal.

## 3.2. Prueba de hipótesis

Luego de analizar las pruebas de normalidad a través del software SPSS, se pudo concluir que las muestras siguen una distribución normal. Por ello, se aplicó la prueba de T – Student debido a que las muestras son menores a 50 tal como lo muestra la siguiente tabla.

**Tabla 11.** Relación entre los indicadores y la prueba de hipótesis a utilizar

N°	Indicador	Tamaño de la muestra	Tipo de distribución	Prueba de hipótesis
1	Unidades de equilibrio por producto	42 productos ofertados por la empresa	Distribución normal	Prueba T
2	Razón margen de contribución	24 distritos con mayor presencia de ventas	Distribución normal	Prueba T

Fuente: Elaboración propia (2015)

La tabla 11 describe la relación entre el tamaño de la muestra, el tipo de distribución y la prueba de hipótesis a realizar para cada uno de los indicadores del presente trabajo de investigación.

### 3.2.1. Prueba de hipótesis para el primer indicador

En esta sección se describe la prueba de hipótesis para el primer indicador “unidades de equilibrio por producto”.

**Hipótesis específica N° 1:** El sistema de información para la contabilidad administrativa reduce las unidades de equilibrio por producto de la empresa Corporación Arzapalo S.A.C.

**a) Definición de variables:**

- **UEPP<sub>a</sub>:** Unidades de equilibrio por producto antes de implementar el sistema de información.
- **UEPP<sub>p</sub>:** Unidades de equilibrio por producto después de implementar el sistema de información.

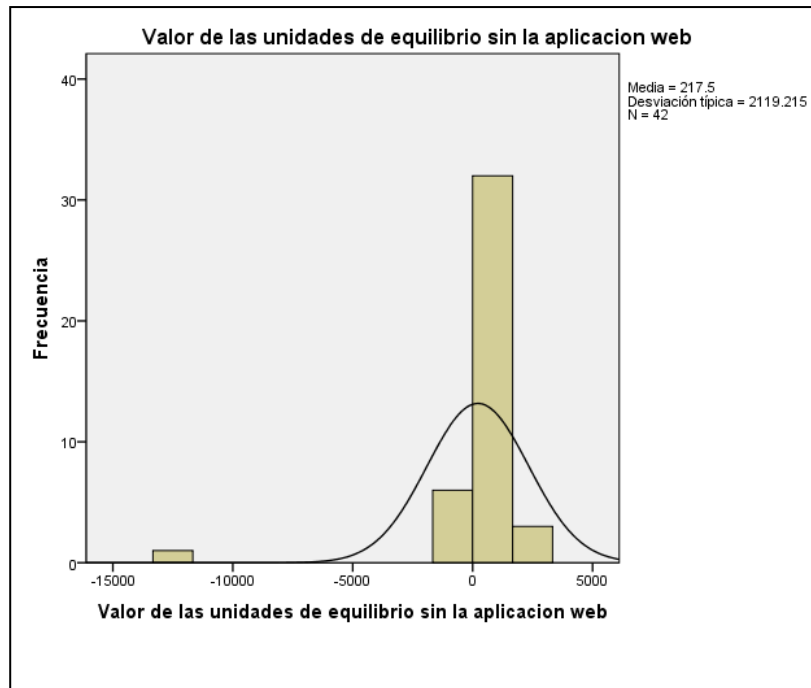
**b) Hipótesis nula (H1<sub>0</sub>):** El sistema de información para la contabilidad administrativa no reduce las unidades de equilibrio por producto de la empresa Corporación Arzapalo S.A.C.

$$UEPP_a \geq UEPP_p$$

**c) Hipótesis alterna (H1<sub>a</sub>):** El sistema de información para la contabilidad administrativa reduce las unidades de equilibrio por producto de la empresa Corporación Arzapalo S.A.C.

$$UEPP_a < UEPP_p$$

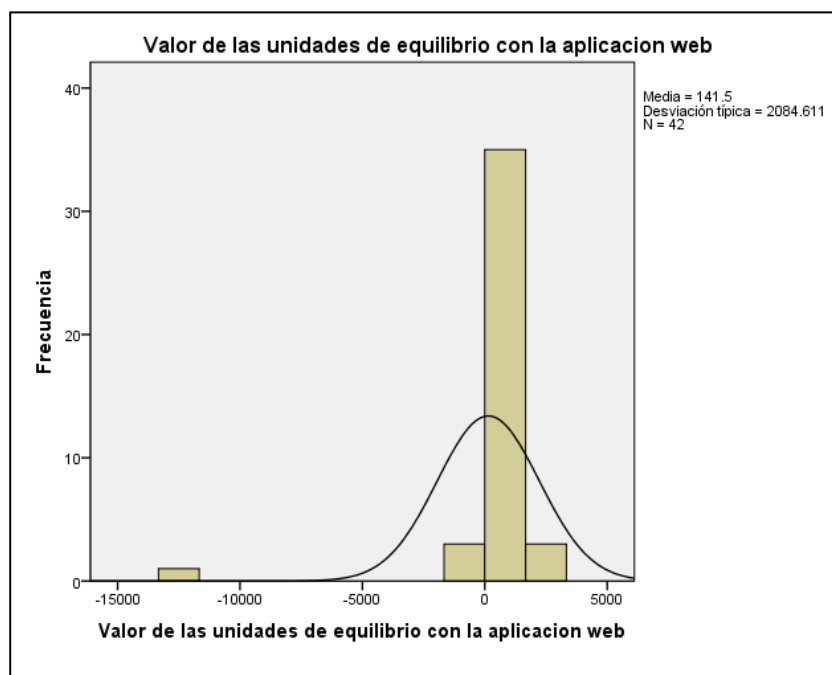
**Figura 21.** Diagrama de histograma para el indicador unidades de equilibrio por producto (Pre – Test)



**Fuente:** Elaboración propia (2015)

En la figura 21 se puede observar que la media de las unidades de equilibrio antes de implementar el sistema de información es de 217.5 unidades con una desviación estándar de 2,119.215 para 42 productos.

**Figura 22.** Diagrama de histograma para el indicador unidades de equilibrio por producto (Post – Test)



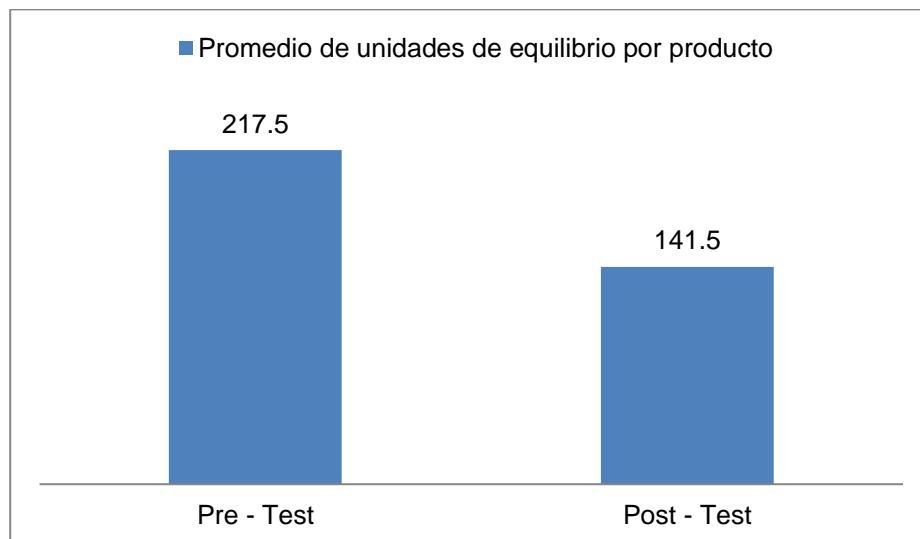
**Fuente:** Elaboración propia (2015)

En la figura 22 se puede observar que la media de las unidades de equilibrio después de implementar el sistema de información es de 141.5 unidades con una desviación estándar de 2,084.611 para 42 productos.

**Análisis comparativo**

En la siguiente figura se puede apreciar la media de las unidades de equilibrio por productos del Pre y Post test, que, al compararlos, se obtiene una reducción del 34.94%, es decir, 76 unidades.

**Figura 23.** Comparativa del Pre – Test y Post – Test para el indicador unidades de equilibrio por producto



Fuente: Elaboración propia (2015)

De la figura 23, se demuestra que el sistema de información para la contabilidad administrativa reduce las unidades de equilibrio por producto.

**Tabla 12.** Resultados de la prueba T – Student para el primer indicador

	95% Intervalo de confianza para la diferencia		t	gl	Sig. (bilateral)
	Inferior	Superior			
Valor de las unidades de equilibrio sin la aplicación web - Valor de las unidades de equilibrio con la aplicación web	46.793	105.207	5.255	41	0.000

Fuente: Elaboración propia (2015)

Como se puede observar en la tabla 12, el análisis arrojado por el software SPSS (sig.) es menor a 0.05, por lo que se rechaza la hipótesis nula y se acepta la hipótesis alterna con un 95% de

confianza. Además, el valor de T obtenido en el análisis es igual a 5.255 y mayor a 0.025, con lo que se ubica en la zona de rechazo de la hipótesis nula y se acepta la hipótesis alterna.

### 3.2.2. Prueba de hipótesis para el segundo indicador

En esta sección se describe la prueba de hipótesis para el segundo indicador “razón margen de contribución por territorios de ventas”.

**Hipótesis específica N° 2:** El sistema información para la contabilidad administrativa incrementa la rentabilidad del producto por territorios de ventas de la empresa Corporación Arzapalo S.A.C.

#### d) Definición de variables:

- **RMCTV<sub>a</sub>:** Razón margen de contribución por territorios de ventas antes de implementar el sistema de información.
- **RMCTV<sub>p</sub>:** Razón margen de contribución por territorios de ventas después de implementar el sistema de información.

e) **Hipótesis nula (H<sub>20</sub>):** El sistema información para la contabilidad administrativa no incrementa la rentabilidad del producto por territorios de ventas de la empresa Corporación Arzapalo S.A.C.

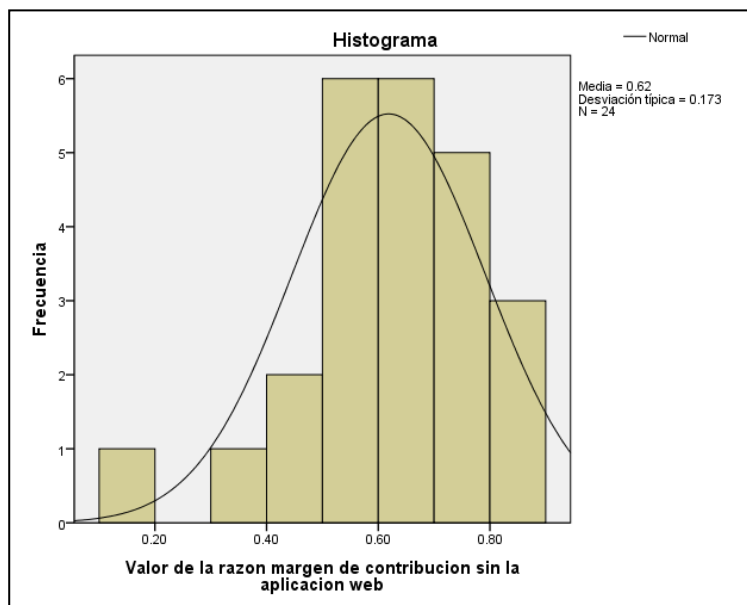
$$RMCTV_a \geq RMCTV_p$$

f) **Hipótesis alterna (H<sub>1a</sub>):** El sistema información para la contabilidad administrativa incrementa la rentabilidad del producto por territorios de ventas de la empresa Corporación Arzapalo S.A.C.

$$RMCTV_a < RMCTV_p$$



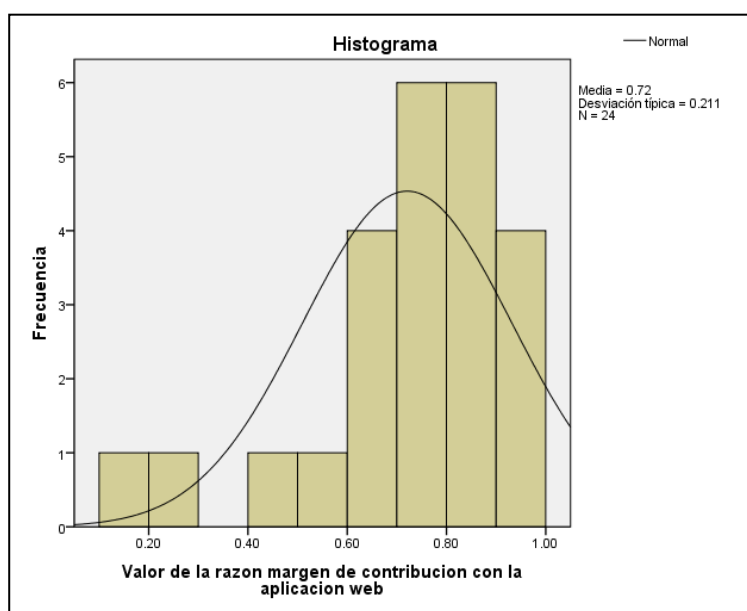
**Figura 24.** Diagrama de histograma para el indicador razón margen de contribución por territorios de ventas (Pre – Test)



**Fuente:** Elaboración propia (2015)

En la figura 24 se puede observar que la media de la razón margen de contribución por territorios de ventas del Pre – Test es de 0.62 con una desviación de 0.173 para una muestra de 24 distritos.

**Figura 25.** Diagrama de histograma para el indicador razón margen de contribución por territorios de ventas (Post – Test)



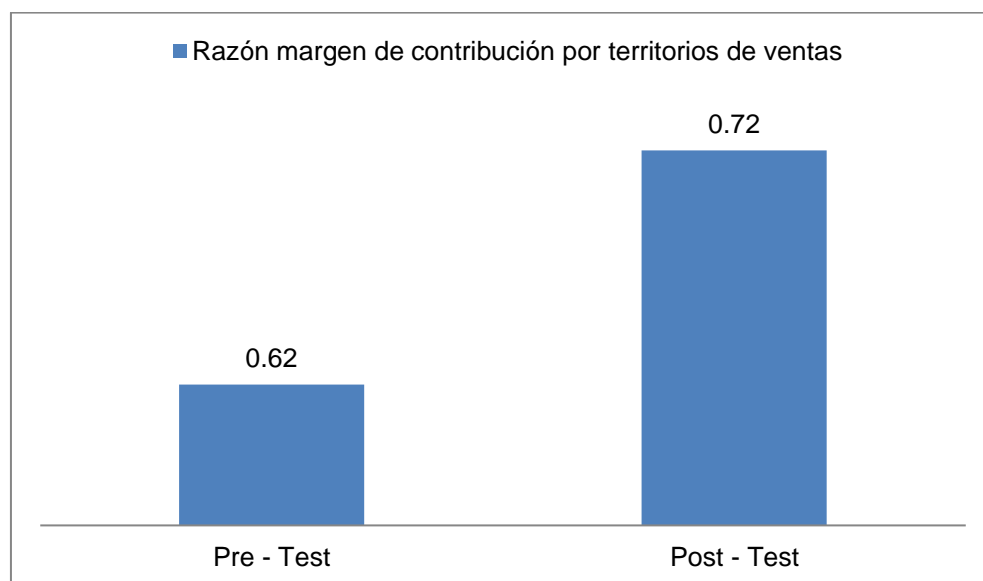
**Fuente:** Elaboración propia (2015)

En la figura 25 se puede observar que la media de la razón margen de contribución por territorios de ventas del Post – Test es de 0.72 con una desviación de 0.211 para una muestra de 24 distritos.

### Análisis comparativo

En la siguiente figura se puede apreciar la media de la razón margen de contribución del Pre y Post test, que, al compararlos, se obtiene un incremento del 16.12%, es decir, 10 puntos porcentuales.

**Figura 26.** Comparativa del Pre – Test y Post – Test para el indicador razón margen de contribución por territorios de ventas



**Fuente:** Elaboración propia (2015)

De la figura 26, se demuestra que el sistema de información para la contabilidad administrativa aumenta la razón margen de contribución por territorios de ventas.

**Tabla 13.** Resultados de la prueba *T – Student* para el segundo indicador

	95% Intervalo de confianza para la diferencia		t	gl	Sig. (bilateral)
	Inferior	Superior			
Valor de la razón margen de contribución sin la aplicación web - Valor de la razón margen de contribución con la aplicación web	-0.13371	-0.07046	-6.678	23	0.000 (8.204E-7)

Fuente: Elaboración propia (2015)

Como se puede observar en la tabla 13, el análisis arrojado por el software SPSS (sig.) es menor a 0.05, por lo que se rechaza la hipótesis nula y se acepta la hipótesis alterna con un 95% de confianza. Además, el valor de T obtenido en el análisis es igual a -6.678 y es mayor a -0.025, con lo que se ubica en la zona de rechazo de la hipótesis nula.

## CAPÍTULO IV: DISCUSIÓN

### 4.1. Indicador 1: Unidades de equilibrio por producto

De la presente tesis se obtiene que la media de las unidades de equilibrio por producto sin el sistema de información para la contabilidad administrativa asciende a 217.5 unidades, mientras que, con la implementación del sistema de información para la contabilidad administrativa, estas ascienden a 141.5 unidades, el cual representó una reducción del 34.94% unidades para llegar al equilibrio en la empresa Corporación Arzapalo S.A.C.

Por consiguiente, se obtuvo resultados opuestos a lo expresado por los investigadores Quinche y Quizhpe (2011), a través de su tesis “Análisis de la relación Costo – Volumen – Utilidad como base de planificación presupuestaria en la fábrica de embutidos La Escocesa de la ciudad de Loja”, los cuales arrojaron como resultado el valor de 50,322.72 libras y después necesitó 70,439.30 libras para alcanzar su punto de equilibrio. Esto representó un aumento del 39.97% con respecto al periodo 2011. Esto se debió a que el gerente de la fábrica propuso que la producción se dé a una escala más alta la cual conllevó al incremento de ventas y de esta manera pudo obtener un porcentaje más alto de utilidad.

### 4.2. Indicador 2: Razón margen de contribución por territorios de ventas

De la presente tesis se obtiene que el promedio de la razón margen de contribución por territorios de ventas sin el sistema de información para la contabilidad administrativa ascienden a 62%, mientras que, con la implementación del sistema de información para la contabilidad administrativa, esta asciende a 72%, el cual representó un aumento del 16.12%.

Por consiguiente, se lograron resultados similares a los obtenidos por Helfmann (2009), a través de su tesis “determinación de la rentabilidad,

costos y márgenes de los principales productos de una empresa de telecomunicaciones”, que obtuvo una rentabilidad promedio del 38.75 % en sus cuatro (04) principales productos, la cual implicó en la mejora de las políticas comerciales.

## CAPÍTULO V: CONCLUSIONES

En esta sección se muestran las conclusiones obtenidas en el desarrollo de la investigación:

1. Se concluye que el promedio de unidades para alcanzar el equilibrio se redujo en un 34.94% haciendo uso del sistema de información para la contabilidad administrativa. Esto indica que el sistema de información influye positivamente en la contabilidad administrativa reduciendo las unidades de equilibrio por producto.
2. Se concluye que la razón margen de contribución por territorios de ventas se incrementa en un 16.12%, lo que significa un aumento de 10 puntos porcentuales. Esto nos indica que el sistema de información implementado influye positivamente en la contabilidad administrativa de la empresa Corporación Arzapalo S. A. C. aumentando la razón margen de contribución por territorios de ventas.
3. A través de la presente investigación, se demuestra que el sistema de información para la contabilidad administrativa influye positivamente en la contabilidad administrativa de la empresa Corporación Arzapalo S.A.C. a través del análisis costo – volumen – utilidad que permitió determinar el rendimiento del producto por territorios de ventas a través del coste de los mismos, el cual a través de la razón margen de contribución por territorios de ventas, se pudo realizar ajustes en el precio de venta del producto y/o la comisión de ventas de los vendedores para obtener una utilidad mayor.
4. La implementación del sistema de información para la contabilidad administrativa logró agilizar el proceso de toma de decisiones debido a que la Alta Gerencia obtuvo la información proporcionada por el sistema en el momento oportuno y pudo elaborar estrategias óptimas para un mejor rendimiento de la empresa.

## **CAPÍTULO VI: RECOMENDACIONES**

A continuación, se describen las recomendaciones para futuras investigaciones:

1. Se propone incrementar el uso del sistema de información para la contabilidad administrativa para que abarque otros tipos de análisis, como el costeo por procesos y el costeo basado en actividades, la cual pudiera abarcar otras áreas que la empresa pudiera adoptar.
2. Se sugiere la implementación de objetivos de ventas a través de los indicadores utilidad objetivo del análisis costo – volumen – utilidad, el cual permitirá realizar una trazabilidad entre las ventas planificadas y las ventas reales para apoyar en el tema de predicciones que servirán también para la toma de decisiones.
3. Se recomienda utilizar técnicas de inteligencia artificial para poder combinar con el sistema de información desarrollado a fin de que pueda predecir resultados y formular tendencias que pudieran apoyar en la planificación, dirección y control del presupuesto.

---

## BIBLIOGRAFÍA

- Amaro, C y Valverde, J. 2007.** Metodologías ágiles. [En línea] 2007. [Citado el: 12 de 05 de 2015.]  
<http://www.sisman.utm.edu.ec/libros/FACULTAD%20DE%20CIENCIAS%20ZOOT%20C3%89CNICAS/CARRERA%20DE%20INGENIER%20C3%8DA%20EN%20INFORMATICA%20AGROPECUARIA/07/INGENIERIA%20DEL%20SOFTWARE%20I/METODOLOGIAS%20AGILES.pdf>.
- Arias, Fidias. 1999.** El proyecto de investigación. [En línea] 1999. [Citado el: 25 de Marzo de 2014.] <http://clasev.net/v2/mod/resource/view.php?id=2110>.
- Bernal, César. 2010.** *Metodología de la investigación*. 3. Bogotá : Pearson Education, 2010. ISBN 9789586991285.
- Cadenhead, Rogers. 2012.** *Programación Java 7*. Madrid : Anaya Multimedia, 2012. ISBN 9788441531789.
- Carrasco, S. 2005.** *Metodología de la investigación científica*. 1. Lima : Editorial San Marcos, 2005. ISBN 9972342425.
- Castelán, Leopoldo y Jorge, Ocharán. 2012.** Diseño de un almacén de datos basado en Data Warehouse Engineering Process y Hefesto. *Universidad Veracruzana*. [En línea] 2012. <https://www.uv.mx/mis/files/2012/11/Diseno-de-un-Almacen-de-datos.pdf>.
- Castro, Fernando. 2011.** *Cómo elaborar una tesis*. 2. México : Pearson Education, 2011.
- Castro, Fernando. 2003.** *El proyecto de investigación y su esquema de elaboración*. 2. Caracas : Uypar, 2003. ISBN 9806629000.
- Centrum Católica. 2011.** Factores que limitan el crecimiento de las Micro y Pequeñas Empresas en el Perú (Mypes). *Revista Strategia*. [En línea] 2011. <http://revistas.pucp.edu.pe/index.php/strategia/article/viewFile/4126/4094>.
- Cerda, Hugo. 1993.** *Los elementos de la investigación*. 2. Quito : El Búho, 1993. ISBN 9589023657.
- Cohen, Daniel y Asín, Enrique. 2000.** *Sistemas de información para los negocios. Un enfoque de toma de decisiones*. 3. México : McGraw Hill, 2000. ISBN: 9701026586.
- Córdova, Manuel. 2003.** *Estadística descriptiva e inferencial*. 5. Lima : Moshera, 2003. ISBN 9972813053.



**De Pablos, Carmen, y otros. 2013.** *Organización y transformación de los sistemas de información en la empresa. 2.* México : Alfaomega Editores, 2013. ISBN 9788473568142.

**Deemer, Pete, y otros. 2009.** The Scrum Primer. [En línea] 2009.  
[http://forja.softwarelibre.gob.ve/docman/view.php/78/575/scrumprimer\\_es.pdf](http://forja.softwarelibre.gob.ve/docman/view.php/78/575/scrumprimer_es.pdf).

**Del Cid, Alma, Méndez, Rosemary y Sandoval, Franco. 2007.** *Investigación. Fundamentos y métodos.* México : Pearson Education, 2007. ISBN 9789702610755.

**Elmasri, Ramez y Navathe, Shamkant. 2007.** *Fundamentos de sistemas de base de datos.* Madrid : Pearson Education, 2007. ISBN 9788478290857.

**Fondo Editorial UCV. 2017.** Universidad César Vallejo. [En línea] 2017. [Citado el: 18 de 5 de 2019.]  
[https://www.ucv.edu.pe/datafiles/FONDO%20EDITORIAL/Manual\\_ISO.pdf](https://www.ucv.edu.pe/datafiles/FONDO%20EDITORIAL/Manual_ISO.pdf).

**Gabriela, Morán y Darío, Alvarado. 2010.** *Métodos de investigación.* México : Pearson Education, 2010. ISBN 9786073219457.

**Garrison, Ray, Noreen, Eric y Brewer, Peter. 2007.** *Contabilidad administrativa.* 11. México : McGraw Hill, 2007. ISBN 9789701060384.

**Gartner. 2014.** Diccionario informático. *Sitio web de Gartner.* [En línea] 2014. [Citado el: 18 de Marzo de 2014.] <http://www.gartner.com>.

**Gilfillan, Ian. 2003.** *La biblia de MySQL. 1.* Madrid : Anaya Multimedia, 2003. ISBN 9788441515581.

**Guajardo, Gerardo y Andrade, Nora. 2008.** *Contabilidad financiera. 5.* México : McGraw Hill, 2008. ISBN 9789701066218.

**Hansen, Don y Mowen, Maryanne. 2007.** *Administración de costos. Contabilidad y control. 5.* Mexico : Cengage Learning, 2007. ISBN 9789706865830.

**Hernández, Roberto, Baptista, María del Pilar y Fernández, Carlos. 2010.** *Metodología de la investigación.* Quinta edición. México : McGraw Hill, 2010. ISBN 9786071502919.

**Herranz, Raúl. 2012.** Scrum Manager. [En línea] 2012.  
[http://www.scrummanager.net/files/scrum\\_manager\\_excelencia\\_del\\_codigo.pdf](http://www.scrummanager.net/files/scrum_manager_excelencia_del_codigo.pdf).

**Herranz, Raúl, y otros. 2011.** Scrum Manager. [En línea] 2011.  
[http://www.scrummanager.net/files/scrum\\_distribuido.pdf](http://www.scrummanager.net/files/scrum_distribuido.pdf).

**Holzner, Steven. 2011.** *Java 2.* Madrid : Anaya Multimedia, 2011. ISBN 9788441510371.

**Horngren, Charles, Datar, Srikant y Foster, George. 2007.** *Contabilidad de costos. Un enfoque gerencial*. 12. México : Pearson Education, 2007. ISBN 9789702607618.

**Horngren, Charles, Sundem, Gary y Stratton, William. 2007.** *Contabilidad administrativa*. 13. México : Pearson Education, 2007. ISBN 9789702606406.

**Inmon, Bill. 2005.** *Building the Data Warehouse*. 4. Indiana : Wiley Publishing, 2005. ISBN 9780764599446.

**Kendall, Kenneth y Kendall, Julie. 2005.** *Análisis y diseño de sistemas*. 6. México : Pearson Education, 2005. ISBN 970-26-0577-6.

**Kimball, Ralph y Ross, Margy. 2013.** *The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling*. 3. Indiana : Wiley Publishing, 2013. ISBN 9781118530801.

**Kofler, Michael. 2006.** *The Definitive Guide to MySQL* 5. New York : Apress, 2006. ISBN 9781590595350.

**Laer Statistics.** Testing for Normality using SPSS Statistics. [En línea] <https://statistics.laerd.com/spss-tutorials/testing-for-normality-using-spss-statistics.php>.

**Laudon, Kenneth y Laudon, Jane. 2008.** *Sistemas de información gerencial. Administración de la empresa digital*. 10. México : Pearson Education, 2008. ISBN 9789702611912.

**Lazo Palacios, Merlin. 2013.** Contabilidad gerencial. *Municipalidad Distrital de Crucero*. [En línea] 2013. <http://www.municrucero.gob.pe/Contabilidad%20de%20los%20Costos%20II%20-%20Merlin%20Lazo%20Palacios.pdf>.

**Lerma y Kirchner, Alejandro y Bárcena, Sergio. 2012.** *Planeación estratégica por áreas funcionales*. México : Alfaomega, 2012. ISBN 9786077074274.

**Levin, Richard y Rubín, David. 2004.** *Estadística para administración y economía*. 7. México : Pearson Education, 2004. ISBN 9702604974.

**Lluis, Josep. 2007.** Business intelligence: Competir con información. [En línea] 2007. [http://itemsweb.esade.edu/biblioteca/archivo/Business\\_Intelligence\\_competir\\_con\\_informacion.pdf](http://itemsweb.esade.edu/biblioteca/archivo/Business_Intelligence_competir_con_informacion.pdf).

**Mannino, Michael. 2007.** *Administración de base de datos. Diseño y desarrollo de aplicaciones*. 3. México : McGraw Hill, 2007. ISBN 9789701061091.

**Ministerio de la Producción. 2012.** Estudios económicos Produce. [En línea] Diciembre de 2012. [Citado el: 11 de Julio de 2014.]  
<http://ogeiee.produce.gob.pe/index.php/shortcode/oeo-documentos-publicaciones/publicaciones-anales/item/735-estadisticas-mype-2011>.

**Mohd, Nornadiah y Bee, Yap. 2011.** Power comparisons of Shapiro - Wilk, Kolmogorov - Smirnov, Lilliefors and Anderson - Darling tests. [En línea] 2011. [Citado el: 30 de 06 de 2015.] [https://noppa.aalto.fi/noppa/kurssi/beans-114.5501/materiaali/Beans-114\\_5501\\_normality\\_tests\\_comparison.pdf](https://noppa.aalto.fi/noppa/kurssi/beans-114.5501/materiaali/Beans-114_5501_normality_tests_comparison.pdf).

**Palacio, Juan. 2008.** Flexibilidad con Scrum. [En línea] 2008. [Citado el: 10 de 02 de 2015.] [http://pmi-panama.org/wp-content/uploads/2011/03/Flexibilidad\\_con\\_Scrum.pdf](http://pmi-panama.org/wp-content/uploads/2011/03/Flexibilidad_con_Scrum.pdf).

**Peña, A. 2006.** Ingeniería de software: Una guía para crear sistemas de información. [En línea] 2006.  
[http://www.wolnm.org/apa/articulos/Ingenieria\\_Software.pdf](http://www.wolnm.org/apa/articulos/Ingenieria_Software.pdf).

**Pérez, Óliver. 2011.** Cuatro enfoques metodológicos para el desarrollo de Software RUP – MSF – XP - SCRUM. *Uniminuto*. [En línea] 2011.  
<http://biblioteca.uniminuto.edu/ojs/index.php/Inventum/article/view/9/9>.

**Pressman, Roger. 2010.** *Ingeniería de software. Un enfoque práctico*. 6. México : McGraw Hill, 2010. 9786071503145.

*Programación en N capas*. **Moquillaza, Santiago, Hugo, Vega y Luis, Guerra. 2010.** 2, 2010, Revista de investigación de sistemas e informática, Vol. 7, pág. 120.

**Ramírez, Daniel. 2008.** *Contabilidad administrativa*. 8. México : McGraw Hill, 2008. ISBN 9789701066300.

**Ramos, Jesús, Ramos, Alicia y Montero, Fernando. 2006.** *Sistemas gestores de base de datos*. 1. Madrid : McGraw Hill, 2006. ISBN 8448148797.

**Regalado, Otto, y otros. 2009.** *Factores críticos de éxito en los centros comerciales de Lima Metropolitana y el Callao*. 1. Lima : Ediciones ESAN, 2009. ISBN 9789972622649.

**Ricardo, Catherine. 2009.** *Base de datos*. 1. México : McGraw Hill, 2009. ISBN 9789701072752.

**Scrum Manager. 2014.** Gestión de proyectos Scrum Manager. *Scrum Manager*. [En línea] 2014. [Citado el: 10 de 02 de 2015.]  
[http://www.scrummanager.net/files/sm\\_proyecto.pdf](http://www.scrummanager.net/files/sm_proyecto.pdf).

**Silberschatz, Abraham, Korth, Henry y Sudarshan, S. 2002.** *Fundamentos de base de datos*. Madrid : McGraw Hill, 2002. ISBN 8448136543.

**Sommerville, Ian. 2005.** *Ingeniería de software*. 7. México : Pearson Education, 2005. ISBN 8478290745.

**Triola, Mario. 2009.** *Estadística*. 10. México : Pearson Education, 2009. ISBN 9789702612872.

**Uribe, Ricardo. 2011.** *Costos para la toma de decisiones*. 1. Bogotá : McGraw Hill, 2011. ISBN 9789584104212.

**Walpole, Ronald, Myers, Raymond y Myers, Sharon. 1999.** *Probabilidad y estadística para ingenieros*. 6. México : Pearson Education, 1999. ISBN 9701702646.

**Warren, Carl, Reeve, James y Duchac, Jonathan. 2010.** *Contabilidad administrativa*. 10. México : CENGAGE Learning, 2010. ISBN 97803244812688.

**Yermanos, Estefanía y Correa, Luisa.** Contabilidad Administrativa. Un enfoque gerencia de costos. *Repositorio Universidad ICESI*. [En línea]  
[https://repository.icesi.edu.co/biblioteca\\_digital/bitstream/10906/66580/1/libro\\_contable\\_administrativo.pdf](https://repository.icesi.edu.co/biblioteca_digital/bitstream/10906/66580/1/libro_contable_administrativo.pdf).

**Zorrilla Castro, Unai, y otros. 2010.** *Guía de arquitectura N-Capas orientada al Dominio con .Net 4.0*. Madrid : Krasis Press, 2010. ISBN 8493669636.

### Anexo 1: Matriz de consistencia

Matriz de consistencia						
Problema	Objetivos	Hipótesis	VARIABLES	Dimensiones	Indicador	Metodología
<b>General</b>			<b>Independiente</b>			
¿De qué forma un sistema de información influye en la contabilidad administrativa de la empresa Corporación Arzapalo S. A. C?	Determinar la influencia de un sistema de información en la contabilidad administrativa de la empresa Corporación Arzapalo S.A.C	El sistema de información optimiza la contabilidad administrativa de la empresa Corporación Arzapalo S.A.C	Sistema de información			
<b>Específico</b>			<b>Dependiente</b>			
¿En qué medida un sistema de información para la contabilidad administrativa influye en las unidades de equilibrio por producto de la empresa Corporación Arzapalo S.A.C.?	Determinar la influencia de un sistema de información para la contabilidad administrativa en las unidades de equilibrio por producto de la empresa Corporación Arzapalo S.A.C.	El sistema de información para la contabilidad administrativa reduce las unidades de equilibrio por producto de la empresa Corporación Arzapalo S.A.C.	Contabilidad administrativa	Análisis costo – volumen – utilidad	Unidades de equilibrio por producto	
¿De qué forma un sistema información para la contabilidad administrativa influye en la rentabilidad del producto por territorios de ventas de la empresa Corporación Arzapalo S.A.C.?	Determinar la influencia de un sistema de información para la contabilidad administrativa la de rentabilidad del producto por territorios de ventas de la empresa Corporación Arzapalo S.A.C.	El sistema información para la contabilidad administrativa incrementa la rentabilidad del producto por territorios de ventas de la empresa Corporación Arzapalo S.A.C.		Costeo variable y análisis de segmentos de mercado	Razón margen de contribución por territorios de ventas	

**Anexo 2: Ficha de revisión documental**

**FICHA DE REVISIÓN DOCUMENTAL – PRE TEST**


<b>Investigador</b>	Luis Enrique Paraguay Arzapalo
<b>Título de la tesis</b>	Sistema de información para la contabilidad administrativa para la empresa Corporación Arzapalo S.A.C.
<b>Institución donde se investiga</b>	Corporación Arzapalo S.A.C
<b>Indicador</b>	Unidades de equilibrio por producto

N°	Periodo	Producto	Unidades de equilibrio
1	Jun-14	203241	245
2	Jun-14	203242	311
3	Jun-14	203243	589
4	Jun-14	203244	491
5	Jun-14	203245	245
6	Jun-14	203246	589
7	Jun-14	203247	589
8	Jun-14	203248	551
9	Jun-14	203350	917
10	Jun-14	203351	916
11	Jun-14	203352	458
12	Jun-14	203353	172
13	Jun-14	203354	137
14	Jun-14	203355	1,145
15	Jun-14	203356	1,056
16	Jun-14	203357	916
17	Jun-14	203361	2,385
18	Jun-14	203362	2,213
19	Jun-14	203363	-248
20	Jun-14	203364	1,225
21	Jun-14	203365	-12,511
22	Jun-14	203366	487
23	Jun-14	203367	-573
24	Jun-14	203368	-573
25	Jun-14	203369	2,632
26	Jun-14	203370	133
27	Jun-14	203371	152
28	Jun-14	204531	207
29	Jun-14	410214	51
30	Jun-14	410215	-10

CORPORACION ARZAPALO SAC  
 Carlos Arzapalo Maaaya  
 Gerente

31	Jun-14	521450	511
32	Jun-14	521451	213
33	Jun-14	521452	213
34	Jun-14	521453	213
35	Jun-14	521454	213
36	Jun-14	674365	405
37	Jun-14	674366	252
38	Jun-14	674367	-11
39	Jun-14	674368	611
40	Jun-14	674369	-25
41	Jun-14	674370	1,087
42	Jun-14	674371	556

**CORPORACION ARZAPALO SAC**



.....  
 Carlos Arzapalo Huauya  
 Gerente

FICHA DE REVISIÓN DOCUMENTAL – POST TEST


<b>Investigador</b>	Luis Enrique Paraguay Arzapalo
<b>Título de la tesis</b>	Sistema de información para la contabilidad administrativa para la empresa Corporación Arzapalo S.A.C.
<b>Institución donde se investiga</b>	Corporación Arzapalo S.A.C
<b>Indicador</b>	Unidades de equilibrio por producto

N°	Periodo	Producto	Unidades de equilibrio
1	Jun-15	203241	225
2	Jun-15	203242	225
3	Jun-15	203243	540
4	Jun-15	203244	450
5	Jun-15	203245	225
6	Jun-15	203246	540
7	Jun-15	203247	540
8	Jun-15	203248	433
9	Jun-15	203350	720
10	Jun-15	203351	720
11	Jun-15	203352	360
12	Jun-15	203353	135
13	Jun-15	203354	108
14	Jun-15	203355	900
15	Jun-15	203356	900
16	Jun-15	203357	720
17	Jun-15	203361	2156
18	Jun-15	203362	2157
19	Jun-15	203363	-195
20	Jun-15	203364	962
21	Jun-15	203365	-12480
22	Jun-15	203366	383
23	Jun-15	203367	-450
24	Jun-15	203368	-450
25	Jun-15	203369	2424
26	Jun-15	203370	0
27	Jun-15	203371	0
28	Jun-15	204531	0
29	Jun-15	410214	40
30	Jun-15	410215	40

CORPORACION ARZAPALO SAC  
 Carlos Arzapalo Huastiza  
 Gerente



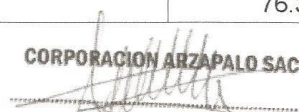
31	Jun-15	521450	432
32	Jun-15	521451	180
33	Jun-15	521452	180
34	Jun-15	521453	180
35	Jun-15	521454	180
36	Jun-15	674365	343
37	Jun-15	674366	213
38	Jun-15	674367	0
39	Jun-15	674368	517
40	Jun-15	674369	0
41	Jun-15	674370	920
42	Jun-15	674371	470

**CORPORACION ARZAPALO SAC**  
  
 .....  
 Carlos Arzapalo Huayn  
 Gerente

FICHA DE REVISIÓN DOCUMENTAL – PRE TEST

<b>Investigador</b>	Luis Enrique Paraguay Arzapalo
<b>Título de la tesis</b>	Sistema de información para la contabilidad administrativa para la empresa Corporación Arzapalo S.A.C.
<b>Institución donde se investiga</b>	Corporación Arzapalo S.A.C
<b>Indicador</b>	Razón margen de contribución

N°	Periodo	Distrito	Razón margen de contribución
1	Jun-14	Ancón	74.00%
2	Jun-14	Ate - Vitarte	69.20%
3	Jun-14	Breña	14.10%
4	Jun-14	Cercado de Lima	55.42%
5	Jun-14	Chorrillos	61.23%
6	Jun-14	Jesús María	58.10%
7	Jun-14	La Molina	47.28%
8	Jun-14	La victoria	88.34%
9	Jun-14	Lince	49.89%
10	Jun-14	Lurín	60.10%
11	Jun-14	Magdalena del mar	74.84%
12	Jun-14	Miraflores	66.16%
13	Jun-14	San Borja	86.01%
14	Jun-14	San Isidro	67.23%
15	Jun-14	San Juan de Lurigancho	74.65%
16	Jun-14	San Juan de Miraflores	41.92%
17	Jun-14	San Luis	60.15%
18	Jun-14	San Martín de Porres	86.92%
19	Jun-14	San Miguel	55.68%
20	Jun-14	Santiago de Surco	58.82%
21	Jun-14	Surquillo	51.15%
22	Jun-14	Villa el salvador	75.18%
23	Jun-14	Villa María del triunfo	33.71%
24	Jun-14	Callao	76.32%

CORPORACION ARZAPALO SAC  
  
 Carlos Arzapalo Huayta  
 Gerente

**FICHA DE REVISIÓN DOCUMENTAL – POST TEST**

<b>Investigador</b>	Luis Enrique Paraguay Arzapalo
<b>Título de la tesis</b>	Sistema de información para la contabilidad administrativa para la empresa Corporación Arzapalo S.A.C.
<b>Institución donde se investiga</b>	Corporación Arzapalo S.A.C
<b>Indicador</b>	Razón margen de contribución

N°	Periodo	Distrito	Razón margen de contribución
1	Jun-15	Ancón	92.27%
2	Jun-15	Ate - Vitarte	83.47%
3	Jun-15	Breña	12.58%
4	Jun-15	Cercado de Lima	74.15%
5	Jun-15	Chorrillos	70.35%
6	Jun-15	Jesús María	60.13%
7	Jun-15	La Molina	59.50%
8	Jun-15	La victoria	95.00%
9	Jun-15	Lince	54.38%
10	Jun-15	Lurín	87.08%
11	Jun-15	Magdalena del mar	89.10%
12	Jun-15	Miraflores	74.06%
13	Jun-15	San Borja	95.00%
14	Jun-15	San Isidro	78.74%
15	Jun-15	San Juan de Lurigancho	83.04%
16	Jun-15	San Juan de Miraflores	40.50%
17	Jun-15	San Luis	78.01%
18	Jun-15	San Martin de Porres	95.00%
19	Jun-15	San Miguel	70.86%
20	Jun-15	Santiago de Surco	78.25%
21	Jun-15	Surquillo	62.99%
22	Jun-15	Villa el salvador	81.10%
23	Jun-15	Villa María del triunfo	28.12%
24	Jun-15	Callao	87.27%

**CORPORACION ARZAPALO SAC**

*[Firma manuscrita]*  
 Carlos Arzapalo Huatya  
 Gerente

### Anexo 3: Evaluación de expertos para la metodología de desarrollo

**TABLA DE EVALUACIÓN DE EXPERTOS**

(Metodología de desarrollo)

Nombre y apellido del experto: ABO RAUL HILARIO ESPINOZA

Cargo: Docente de la Facultad de Sistemas

Institución donde labora: Universidad César Vallejo

Fecha: 23-11-13

**TESIS**

**SISTEMA DE INFORMACIÓN PARA LA CONTABILIDAD ADMINISTRATIVA PARA LA EMPRESA CORPORACIÓN ARZAPALO S.A.C.**

Mediante la tabla de evaluación de expertos, usted tiene la facultad de evaluar cada una de las preguntas a través de un puntaje. Asimismo, le exhortamos en la corrección de los ítems indicando sus sugerencias, con la finalidad de mejorar la coherencia de las preguntas.

ÍTEMS	Puntaje a colocar: <b>Muy Bueno = 4, Bueno = 3, Regular = 2, Malo = 1</b>			
	Preguntas - Criterios	RUP	SCRUM	MSF
1	La metodología permite una especificación de los roles.	3	3	2
2	La metodología permite que el cliente esté presente en todo el proceso de desarrollo del producto.	3	4	2
3	La metodología brinda resultados rápidos.	3	4	2
4	¿Qué metodología utiliza una documentación más ligera?	2	4	2
5	La metodología está enfocada hacia el producto	3	4	3
6	La metodología permite el trabajo muy dinámico con equipos con menos de 10 personas.	3	4	2
7	La metodología es más óptima para proyectos medianos y pequeños.	2	4	3
8	La metodología utiliza pocos artefactos	2	3	3
9	La metodología utiliza una cartera de actividades que son evaluados al final de cada iteración.	2	3	2
10	La metodología permite que el usuario cambie de requisitos constantemente.	2	3	2
11	La metodología permite reuniones diarias para revisar el estado del producto.	2	3	2
Total:		24	39	25

SUGERENCIAS: \_\_\_\_\_

Firma del experto: 



**TABLA DE EVALUACIÓN DE EXPERTOS**

(Metodología de desarrollo)

Nombre y apellido del experto: Huamani Huaranja Daniel

Cargo: Docente de la Facultad de Sistemas

Institución donde labora: Universidad César Vallejo

Fecha: 23/11/13

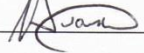
**TESIS**

SISTEMA DE INFORMACIÓN PARA LA CONTABILIDAD ADMINISTRATIVA PARA LA EMPRESA CORPORACIÓN ARZAPALO S.A.C.

Mediante la tabla de evaluación de expertos, usted tiene la facultad de evaluar cada una de las preguntas a través de un puntaje. Asimismo, le exhortamos en la corrección de los ítems indicando sus sugerencias, con la finalidad de mejorar la coherencia de las preguntas.

ÍTEMS	Puntaje a colocar: <b>Muy Bueno = 4, Bueno = 3, Regular = 2, Malo = 1</b>			
	Preguntas - Criterios	RUP	SCRUM	MSF
1	La metodología permite una especificación de los roles.	3	4	2
2	La metodología permite que el cliente esté presente en todo el proceso de desarrollo del producto.	2	3	1
3	La metodología brinda resultados rápidos.	2	3	1
4	¿Qué metodología utiliza una documentación más ligera?	2	3	1
5	La metodología está enfocada hacia el producto	2	3	2
6	La metodología permite el trabajo muy dinámico con equipos con menos de 10 personas.	2	3	1
7	La metodología es más óptima para proyectos medianos y pequeños.	2	3	2
8	La metodología utiliza pocos artefactos	2	3	2
9	La metodología utiliza una cartera de actividades que son evaluados al final de cada iteración.	2	3	2
10	La metodología permite que el usuario cambie de requisitos constantemente.	2	3	2
11	La metodología permite reuniones diarias para revisar el estado del producto.	2	3	1
Total:		23	34	17

SUGERENCIAS: \_\_\_\_\_

Firma del experto: 

**TABLA DE EVALUACIÓN DE EXPERTOS**

(Metodología de desarrollo)

Nombre y apellido del experto: Percy Omar Baldesi

Cargo: Docente de la Facultad de Sistemas

Institución donde labora: Universidad César Vallejo

Fecha: 22-11-13


**TESIS**

SISTEMA DE INFORMACIÓN PARA LA CONTABILIDAD ADMINISTRATIVA PARA LA EMPRESA CORPORACIÓN ARZAPALO S.A.C.

Mediante la tabla de evaluación de expertos, usted tiene la facultad de evaluar cada una de las preguntas a través de un puntaje. Asimismo, le exhortamos en la corrección de los ítems indicando sus sugerencias, con la finalidad de mejorar la coherencia de las preguntas.

ÍTEMS	Puntaje a colocar: <b>Muy Bueno = 4, Bueno = 3, Regular = 2, Malo = 1</b>			
	Preguntas - Criterios	RUP	SCRUM	MSF
1	La metodología permite una especificación de los roles.	2	3	2
2	La metodología permite que el cliente esté presente en todo el proceso de desarrollo del producto.	3	3	3
3	La metodología brinda resultados rápidos.	2	4	2
4	¿Qué metodología utiliza una documentación más ligera?	2	3	2
5	La metodología está enfocada hacia el producto	2	3	2
6	La metodología permite el trabajo muy dinámico con equipos con menos de 10 personas.	2	3	2
7	La metodología es más óptima para proyectos medianos y pequeños.	3	3	2
8	La metodología utiliza pocos artefactos	2	2	2
9	La metodología utiliza una cartera de actividades que son evaluados al final de cada iteración.	3	3	2
10	La metodología permite que el usuario cambie de requisitos constantemente.	2	3	2
11	La metodología permite reuniones diarias para revisar el estado del producto.	3	3	3
Total:		26	33	24

SUGERENCIAS: \_\_\_\_\_

Firma del experto: 

## Anexo 4: Entrevista al Gerente de Corporación Arzapalo S.A.C

### ENTREVISTA PERSONAL

<b>Entrevistador:</b>	Luis Paraguay Arzapalo
<b>Entrevistado:</b>	Carlos Arzapalo Huauya
<b>Cargo:</b>	Gerente
<b>Fecha:</b>	06/09/2013
<b>Lugar:</b>	Oficina Corporación Arzapalo S.A.C

**Objetivo:** La presente entrevista tiene como fin identificar la problemática de la empresa. Por ello, se pide responder a la presente entrevista con veracidad y sinceridad.

**1. ¿Actualmente como administra la contabilidad administrativa en su empresa?**

Actualmente esto se maneja con total desorden y sin la debida importancia del caso. Es todo en base a la intuición y por ello es que no tengo claro los costos de mi producto, principalmente mis unidades de equilibrio. Actualmente esos reportes lo tengo fuera de tiempo y en ocasiones ni se generan por el alto costo de generación de esos reportes en horas trabajadas. Cuando no los tengo, para hallar mi punto de equilibrio de mis productos hago una operación del costo agregándole la ganancia correspondiente.

**2. ¿Cree que los resultados son correctos?**

Efectivamente no. En los últimos meses he generado esos reportes y he contrastado con lo generado a mi intuición y son diferentes, varían regularmente y en algunos casos son mínimos pero ello no me asegura una correcta toma de decisiones.

**3. ¿Qué otro uso le da a la contabilidad administrativa?**

Lo más importante de la contabilidad administrativa es que me permite medir la rentabilidad de mi producto por distritos. Ese indicador para mi es de suma importancia porque con ello puedo conocer con datos cuantitativos cuan rentable es mi producto en Lima por ejemplo. Y no solo eso, me permite también conocer la rentabilidad de mis vendedores. Gracias a ello puedo planificar mejor mis estrategias de ventas como manejar las comisiones por ventas y sectorizar mis productos.

**4. ¿Hay otros indicadores que le son útiles?**

Según el rubro de mi empresa esos son los más importantes. Claro que para calcular el punto de equilibrio es necesario conocer mi costo variable unitario y el margen de seguridad. Otros indicadores que complementan son el margen de seguridad y el apalancamiento operativo.

**5. Por último, ¿Por qué no ha venido haciendo la contabilidad administrativa y que espera usted de la herramienta propuesta?**

No se ha venido haciendo constantemente por el costo de recursos humanos. He tenido que pagar horas extras para tener esos reportes. Lo que espero es poder tener una herramienta que me permita generar los reportes del costo variable unitario, margen de contribución unitario, el punto de equilibrio, el apalancamiento operativo, el margen de seguridad sobre todo el más importante, la rentabilidad del producto por territorios de ventas.

**CORPORACION ARZAPALO SAC**

Carlos Arzapalo Huauya  
Gerente

## Anexo 5: Carta de aprobación de la empresa



### CARTA DE ACEPTACIÓN DE SISTEMA DE INFORMACIÓN

Yo, **Carlos Alberto Arzapalo Huauya**, identificado con **DNI N° 09555827**, en calidad de **Gerente Administrador** de la empresa **Corporación Arzapalo S.A.C.**, doy por este medio la conformidad que el sistema de información "**Contad v. 1.0**" implementado en la empresa cumple satisfactoriamente con todos los requerimientos expuestos en las reuniones sostenidas con el **Sr. Luis Enrique Paraguay Arzapalo**, las cuales han sido validadas por los usuarios encargados en los siguientes periodos:


- Del 02 de marzo al 23 de junio de 2015 en desarrollo y pruebas.
- Del 06 de julio al 25 de julio de 2015 en producción.

Se ha validado que los procesos de contabilidad administrativa solicitados e implementados por el "**Contad v. 1.0**", así como las observaciones realizadas hasta el momento, han sido implementados satisfactoriamente.

Se expide el presente documento para los fines que se crean convenientes.

San Juan de Lurigancho, 31 de Julio de 2015

**CORPORACION ARZAPALO SAC**

  
.....  
Carlos Arzapalo Huauya  
Gerente

**CORPORACION ARZAPALO S.A.C.**  
P.J. VICTOR ANDRES BELAUDE NRO. 198 URB. CHACARILLA DE OTERO LIMA - LIMA - SAN JUAN DE LURIGANCHO



## Anexo 6: Desarrollo de la metodología

En esta sección se describe el desarrollo del sistema de información para la contabilidad administrativa siguiendo la metodología Scrum.

### 1. Personas y roles del proyecto

En esta sección se muestran las personas que participaron en el desarrollo del presente trabajo de investigación. De la misma manera, en esta sección se detalla los roles asignados a cada uno de los participantes.

**Tabla 14.** *Personas y roles del presente trabajo de investigación*

N°	Persona	Rol
1	Carlos Arzapalo	Dueño del producto
2	Luis Paraguay	Scrum máster
3	Sebastián Aguilar	Equipo de trabajo
4	Lorena Flores	Equipo de trabajo
5	Maribel Arana	Stakeholder

**Fuente:** Elaboración propia (2015)

La tabla 14 muestra los roles de las personas involucradas en el desarrollo del presente trabajo de investigación.

### 2. Historias de usuarios

En esta sección se procede a mostrar las historias de usuarios del presente trabajo de investigación.

<b>Historia de usuario</b>	
<b>Número:</b> 1	<b>Usuario:</b> Asistente
<b>Nombre de historia:</b> Registro de proveedores	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alta
<b>Programador responsable:</b> Lorena Flores	
<b>Descripción:</b> Deseo un formulario para registrar a los proveedores y que contengan su razón social, RUC, dirección, distrito, teléfono y correo electrónico. El código se genera automáticamente.	
<b>Validación:</b> El asistente ingresa los datos detallados en la descripción y luego presiona un botón para grabar el cual debe mostrar un mensaje de conformidad. Además, deberá validar primero si ya existe un proveedor con el mismo RUC. Si existe no se graba el registro y muestra un mensaje de error. Caso contrario, se graba muestra un mensaje de confirmación.	

<b>Historia de usuario</b>	
<b>Número:</b> 2	<b>Usuario:</b> Asistente
<b>Nombre de historia:</b> Búsqueda de proveedores	
<b>Prioridad:</b> Media	<b>Riesgo en el desarrollo:</b> Medio
<b>Programador responsable:</b> Sebastián Aguilar	
<b>Descripción:</b> Deseo un formulario para buscar los proveedores por su razón social. Esto es para poder editarlo o eliminarlo.	
<b>Validación:</b> La asistente debe ingresar la razón social del proveedor y luego al presionar un botón, este debe mostrar una lista de proveedores en una tabla con las opciones de eliminar y editar.	

Historia de usuario	
<b>Número:</b> 3	<b>Usuario:</b> Asistente
<b>Nombre de historia:</b> Actualizar datos del proveedor	
<b>Prioridad:</b> Baja	<b>Riesgo en el desarrollo:</b> Bajo
<b>Programador responsable:</b> Sebastián Aguilar	
<b>Descripción:</b> Deseo un formulario para actualizar los datos del proveedor. Los datos del proveedor a actualizar son la razón social, RUC, dirección, distrito, teléfono y correo electrónico. Deseo que por defecto se muestren los datos ya registrados del proveedor.	
<b>Validación:</b> La asistente deberá completar todos los datos que desee modificar y al presionar el botón de actualizar, deberá aparecer un mensaje de confirmación que valide si desea modificar el registro actual. Si es sí, se actualizarán los datos y mostrará un mensaje de confirmación de la transacción o el error generado.	

Historia de usuario	
<b>Número:</b> 4	<b>Usuario:</b> Asistente
<b>Nombre de historia:</b> Eliminar proveedor	
<b>Prioridad:</b> Baja	<b>Riesgo en el desarrollo:</b> Bajo
<b>Programador responsable:</b> Lorena Flores	
<b>Descripción:</b> Deseo un formulario en donde aparezcan los datos del proveedor que desee eliminar. Estos datos son la razón social, RUC, dirección, distrito, teléfono y correo electrónico del proveedor.	
<b>Validación:</b> La asistente visualizará un formulario en donde no permita la modificación de datos. Luego, verificará si es el proveedor que desea eliminar. Al presionar el botón de eliminar, deberá aparecer un mensaje de confirmación. Si es sí, se eliminará el proveedor y mostrará un mensaje confirmando la eliminación del proveedor o el error generado.	

<b>Historia de usuario</b>	
<b>Número:</b> 5	<b>Usuario:</b> Asistente
<b>Nombre de historia:</b> Búsqueda de productos	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Sebastián Aguilar	
<p><b>Descripción:</b>  Deseo un formulario que permita hacer la búsqueda de productos por nombre. Al presionar el botón buscar deberá aparecer una lista con los productos en donde coincidan el nombre con los siguientes datos: código, nombre, categoría, marca, stock mínimo, stock actual, precio y la fecha de la última actualización. Además, deberán aparecer las opciones de asignar proveedor y ver su contabilidad.</p>	
<p><b>Validación:</b>  La asistente visualizará todos los productos que coincidan con el nombre ingresado. En caso de no haber datos, mostrará un mensaje informativo o el error generado en la transacción.</p>	

<b>Historia de usuario</b>	
<b>Número:</b> 6	<b>Usuario:</b> Asistente
<b>Nombre de historia:</b> Registrar abastecimiento	
<b>Prioridad:</b> Media	<b>Riesgo en el desarrollo:</b> Medio
<b>Programador responsable:</b> Lorena Flores	
<p><b>Descripción:</b>  Deseo un formulario que permita registrar el abastecimiento de los productos. Para ello se necesitará escoger el proveedor, la cantidad mínima a pedir, el precio de venta por mayor y por menor. Por defecto ambos precios y la cantidad deben de ser cero.</p>	
<p><b>Validación:</b>  El asistente después de escoger el producto de la lista proporcionada por la historia N° 5, llenará todos los datos mostrados y presionará el botón guardar. Luego aparecerá un mensaje de confirmación, en donde si presiona la opción Sí, se procederá a guardar y se mostrará un mensaje de confirmación correcta del registro o el mensaje de error de la transacción.</p>	

<b>Historia de usuario</b>	
<b>Número:</b> 7	<b>Usuario:</b> Asistente
<b>Nombre de historia:</b> Buscar abastecimiento	
<b>Prioridad:</b> Media	<b>Riesgo en el desarrollo:</b> Medio
<b>Programador responsable:</b> Lorena Flores	
<p><b>Descripción:</b>                      Deseo un formulario que permita la búsqueda del abastecimiento de los productos. Para ello los buscaré por nombre del producto de la lista mostrada seleccionaré uno aparecerá una lista con todos los proveedores que abastecen el producto. La lista deberá mostrar el nombre del producto, el proveedor, la cantidad mínima, el precio por mayor y menor y las opciones de edición y eliminación del abastecimiento.</p>	
<p><b>Validación:</b>                      El asistente realizará una búsqueda aproximada del producto y presionará el botón buscar. Si hay datos mostrara la lista de productos, de la cual escogerá uno para ver el listado de abastecimientos. Si la búsqueda no obtiene datos mostrara un mensaje informativo que no se encontraron coincidencias o el error generado de la transacción.</p>	

<b>Historia de usuario</b>	
<b>Número:</b> 8	<b>Usuario:</b> Asistente
<b>Nombre de historia:</b> Editar abastecimiento	
<b>Prioridad:</b> Baja	<b>Riesgo en el desarrollo:</b> Bajo
<b>Programador responsable:</b> Lorena Flores	
<p><b>Descripción:</b>                      Deseo un formulario que me permita modificar los datos de un abastecimiento. Para ello, se debe visualizar los datos ya registrados del producto, proveedor, cantidad mínima a pedir, precio por mayor y al por menor. Los nombres del producto y del proveedor no se podrán modificar.</p>	
<p><b>Validación:</b>                      El asistente deberá cambiar los datos deseados del abastecimiento y luego procederá a presionar el botón actualizar los nuevos datos. Al presionar el botón, aparecerá un mensaje de confirmación. Si se elige la opción Sí, se actualizarán los datos y se mostrará un mensaje de confirmación o se mostrará el error de la transacción.</p>	

Historia de usuario	
<b>Número:</b> 9	<b>Usuario:</b> Asistente
<b>Nombre de historia:</b> Eliminar abastecimiento	
<b>Prioridad:</b> Baja	<b>Riesgo en el desarrollo:</b> Bajo
<b>Programador responsable:</b> Lorena Flores	
<b>Descripción:</b> Deseo un formulario que me permita visualizar los datos de un determinado abastecimiento para verificar los datos y luego poder eliminarlos al presionar un botón.	
<b>Validación:</b> El asistente visualizará los datos en solo lectura y luego de ello presionará el botón eliminar. Al presionarlo, se mostrará un mensaje de confirmación. Si se elige la opción Sí, se eliminará el registro y se mostrará un mensaje de confirmación o el error generado de la operación.	

Historia de usuario	
<b>Número:</b> 10	<b>Usuario:</b> Asistente
<b>Nombre de historia:</b> Registrar orden de compra	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Sebastián Aguilar	
<b>Descripción:</b> Deseo un formulario donde pueda registrar las órdenes de compras para los proveedores. El número de orden debe ser generado automáticamente y visualizado en el formulario. Los datos a ingresar son el proveedor, la fecha de emisión que por defecto mostrará la fecha actual, la fecha de vencimiento y las observaciones que pudiera haber. Al registrar la orden, deberá llevarme a otro formulario donde pueda registrar los productos de esta orden.	
<b>Validación:</b> El asistente ingresará los datos descritos en la descripción y al presionar el botón guardar, se debe mostrar un mensaje de confirmación de registro. Al seleccionar la opción afirmativa, se guardará los datos ingresados y se mostrará un mensaje de confirmación exitosa o el mensaje de error de la operación. Luego de ello, el usuario será redirigido al formulario de registro de productos para la presente orden de compra.	

<b>Historia de usuario</b>	
<b>Número:</b> 11	<b>Usuario:</b> Asistente
<b>Nombre de historia:</b> Registrar producto en la orden de compra	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Sebastián Aguilar	
<p><b>Descripción:</b>  Deseo un formulario donde pueda registrar los productos solicitados en una orden de compra. En ese formulario se seleccionará el producto, se ingresará la cantidad y el precio de compra acordado. Adicional a ello, se mostrará en una tabla abajo el código y nombre del producto, la cantidad a comprar, el precio, el subtotal de esa operación y la opción de eliminar dicho registro.</p>	
<p><b>Validación:</b>  El asistente ingresará los datos descritos en la descripción y al presionar el botón agregar, se registrará el producto en la base de datos y se visualizará en la tabla de detalle. Además, se mostrará un mensaje de confirmación exitosa o el mensaje de error de la operación.</p>	

<b>Historia de usuario</b>	
<b>Número:</b> 12	<b>Usuario:</b> Asistente
<b>Nombre de historia:</b> Eliminar producto de la orden de compra	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Lorena Flores	
<p><b>Descripción:</b>  Deseo poder eliminar un producto de la orden de compra. Para ello de la lista de productos que se ha registrado en la orden de compra debo seleccionar una opción de eliminar producto sin confirmación.</p>	
<p><b>Validación:</b>  El asistente seleccionará un producto de la lista de productos registrados con anterioridad. Presionará un botón y se eliminará de la lista y de la base de datos. Además, se mostrará un mensaje de confirmación exitosa o el mensaje de error de la operación.</p>	

<b>Historia de usuario</b>	
<b>Número:</b> 13	<b>Usuario:</b> Asistente
<b>Nombre de historia:</b> Buscar orden de compra	
<b>Prioridad:</b> Media	<b>Riesgo en el desarrollo:</b> Medio
<b>Programador responsable:</b> Lorena Flores	
<b>Descripción:</b> Deseo poder buscar una orden de compra por fecha para poder actualizar su observación y su estado (emitida, cancelada o anulada) y/o ver los productos registrados para una determinada orden de compra.	
<b>Validación:</b> El asistente seleccionará una fecha y luego presionará el botón buscar. Luego de ello visualizará la lista de órdenes de compra con las opciones de editar y ver detalle, o visualizará un mensaje informativo cuando no existan coincidencias o un mensaje de error producto de la operación.	

<b>Historia de usuario</b>	
<b>Número:</b> 14	<b>Usuario:</b> Asistente
<b>Nombre de historia:</b> Editar orden de compra	
<b>Prioridad:</b> Media	<b>Riesgo en el desarrollo:</b> Bajo
<b>Programador responsable:</b> Lorena Flores	
<b>Descripción:</b> Deseo un formulario para editar una orden de compra. Sólo debe permitir la modificación de la observación y el estado de la orden, y deben mostrarse la información que ya haya sido registrada.	
<b>Validación:</b> El asistente seleccionará una orden de compra, modificará los datos que crea necesario y presionará el botón actualizar. Cuando se presione el botón, se mostrará un mensaje de confirmación de la operación. Si se escoge la opción Sí, se actualizará la orden de compra y mostrará un mensaje de confirmación exitosa o el error generado en la operación.	



Historia de usuario	
<b>Número:</b> 15	<b>Usuario:</b> Administrador, Asistente
<b>Nombre de historia:</b> Formulario de ingreso al sistema	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Sebastián Aguilar	
<b>Descripción:</b> Deseo un formulario de acceso al sistema para evitar que usuarios no autorizados vean la información.	
<b>Validación:</b> Los usuarios ingresaran su usuario y contraseña y el sistema validará, de acuerdo con sus credenciales, si accede o no a la ventana principal. Si no tiene acceso al sistema se mostrará el mensaje respectivo; si no ha ingresado ningún dato, deberá aparecer un mensaje solicitando que ingrese ambos datos; y si ha digitado su usuario o clave incorrectamente, se mostrará el mensaje pertinente.	

Historia de usuario	
<b>Número:</b> 16	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Reporte de costo variable unitario por periodos	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Sebastián Aguilar	
<b>Descripción:</b> Deseo un formulario que permita generar un reporte en PDF del costo variable unitario de dos periodos diferentes.	
<b>Validación:</b> El usuario seleccionará dos periodos y presionará el botón generar reporte. Luego, deberá aparecer un reporte en formato PDF con la información solicitada (producto, unidades vendidas, costo total, costo variable unitario, costo fijo) agrupados por periodos.	

<b>Historia de usuario</b>	
<b>Número:</b> 17	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Reporte de costo variable unitario por producto	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Lorena Flores	
<b>Descripción:</b> Deseo un formulario que me permita generar un reporte en PDF que muestre un gráfico de barras del costo variable unitario de los últimos 6 meses.	
<b>Validación:</b> El usuario buscará el producto por nombre (no será búsqueda exacta) el cual mostrará una lista con los productos que coincidan con el criterio ingresado. Luego, el usuario al seleccionar el producto el sistema deberá mostrar un reporte en PDF con un gráfico en barras de la evolución del costo variable unitario en los últimos 6 meses. También deberá mostrar en una tabla la información de las unidades vendidas, el costo total, el costo variable unitario y el valor del costo fijo de los meses solicitados.	

<b>Historia de usuario</b>	
<b>Número:</b> 18	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Reporte del margen de contribución unitario por periodos	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Sebastián Aguilar	
<b>Descripción:</b> Deseo un reporte en PDF que me permita visualizar el margen de contribución mensual de todos los productos de los dos periodos seleccionados.	
<b>Validación:</b> El usuario seleccionará dos periodos diferentes y pulsará el botón "Generar reporte". Luego el sistema deberá mostrar el reporte en PDF con la información del producto, las unidades vendidas, los costos variables, el margen de contribución, los costos fijos y la utilidad de la operación, agrupados por meses.	

<b>Historia de usuario</b>	
<b>Número:</b> 19	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Reporte de evolución del margen de contribución unitario por producto	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Lorena Flores	
<b>Descripción:</b> Deseo un formulario que me permita generar un reporte en PDF que muestre un gráfico de línea la evolución del margen de contribución unitario de los últimos 6 meses.	
<b>Validación:</b> El usuario buscará el producto por nombre (no será búsqueda exacta) el cual mostrará una lista con los productos que coincidan con el criterio ingresado. Luego, el usuario al seleccionar el producto el sistema deberá mostrar un reporte en PDF con un gráfico en línea de la evolución del margen de contribución unitario en los últimos 6 meses. También deberá mostrar en una tabla la información correspondiente al margen de contribución.	

<b>Historia de usuario</b>	
<b>Número:</b> 20	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Reporte de punto de equilibrio por periodos	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Sebastián Aguilar	
<b>Descripción:</b> Deseo un reporte en PDF que me permita visualizar el punto de equilibrio por unidades y en monto de todos los productos en los dos periodos seleccionados y agrupados por periodos.	
<b>Validación:</b> El administrador seleccionará los 2 periodos a analizar y presionará el botón "Generar reporte". Luego, el sistema deberá mostrar un reporte en PDF con el producto, las unidades y el monto de equilibrio.	

<b>Historia de usuario</b>	
<b>Número:</b> 21	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Reporte de evolución del punto de equilibrio por producto	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Sebastián Aguilar	
<b>Descripción:</b> Deseo un formulario que me permita generar un reporte en PDF que muestre un gráfico de línea la evolución del punto de equilibrio de los últimos 6 meses de un determinado producto.	
<b>Validación:</b> El usuario buscará el producto por nombre (no será búsqueda exacta) el cual mostrará una lista con los productos que coincidan con el criterio ingresado. Luego, el usuario al seleccionar el producto el sistema deberá mostrar un reporte en PDF con un gráfico en línea de la evolución del punto de equilibrio en unidades y en montos de los últimos 6 meses. También deberá mostrar en una tabla la misma información brindada por el reporte de punto de equilibrio por periodos con la diferencia que esta información está orientada a un solo producto.	

<b>Historia de usuario</b>	
<b>Número:</b> 22	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Reporte de margen de seguridad por periodos	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Lorena Flores	
<b>Descripción:</b> Deseo un reporte en PDF que me permita visualizar el margen de seguridad de todos los productos de los periodos seleccionados.	
<b>Validación:</b> El administrador seleccionará los 2 periodos a analizar y presionará el botón "Generar reporte". Luego, el sistema deberá mostrar un reporte en PDF con información acerca del producto, las ventas, las ventas de equilibrio y el margen de seguridad agrupados por periodos.	

<b>Historia de usuario</b>	
<b>Número:</b> 23	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Reporte de margen de seguridad por producto	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Lorena Flores	
<b>Descripción:</b> Deseo un formulario que me permita generar un reporte en PDF que muestre un gráfico de barras la evolución del margen de seguridad y un gráfico de línea que muestre la evolución de las ventas de los últimos 6 meses para un determinado producto.	
<b>Validación:</b> El usuario buscará el producto por nombre (no será búsqueda exacta) el cual mostrará una lista con los productos que coincidan con el criterio ingresado. Luego, el usuario al seleccionar el producto el sistema deberá mostrar un reporte en PDF con un gráfico en barras la evolución del margen de seguridad y un gráfico de línea de las ventas del producto de los últimos 6 meses. También deberá mostrar en una tabla la misma información brindada por el reporte de margen de seguridad por periodos con la diferencia que esta información está orientada a un solo producto.	

<b>Historia de usuario</b>	
<b>Número:</b> 24	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Reporte de apalancamiento operativo por periodos	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Sebastián Aguilar	
<b>Descripción:</b> Deseo un reporte en PDF que me permita visualizar el apalancamiento operativo de todos los productos de los periodos seleccionados.	
<b>Validación:</b> El administrador seleccionará los 2 periodos a analizar y presionará el botón "Generar reporte". Luego, el sistema deberá mostrar un reporte en PDF con información acerca del producto y su apalancamiento operativo agrupado por periodos.	

<b>Historia de usuario</b>	
<b>Número:</b> 25	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Reporte de apalancamiento operativo por producto	
<b>Prioridad:</b> Alto	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Lorena Flores	
<b>Descripción:</b> Deseo un formulario que me permita generar un reporte en PDF que muestre un gráfico de línea la evolución del apalancamiento operativo de los últimos 6 meses para un determinado producto.	
<b>Validación:</b> El usuario buscará el producto por nombre (no será búsqueda exacta) el cual mostrará una lista con los productos que coincidan con el criterio ingresado. Luego, el usuario al seleccionar el producto el sistema deberá mostrar un reporte en PDF con un gráfico de línea del apalancamiento operativo del producto seleccionado de los últimos 6 meses. También deberá mostrar en una tabla la misma información brindada por el reporte de apalancamiento operativo por periodos con la diferencia que esta información está orientada a un solo producto.	

<b>Historia de usuario</b>	
<b>Número:</b> 26	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Gráfico costos vs ventas	
<b>Prioridad:</b> Media	<b>Riesgo en el desarrollo:</b> Medio
<b>Programador responsable:</b> Lorena Flores	
<b>Descripción:</b> Deseo un gráfico en línea que me muestre la comparación entre el monto de ventas y el monto de compra de los últimos 6 meses.	
<b>Validación:</b> El usuario ingresará al menú “Gráfico de costos y ventas”, el cual al cargar la página deberá mostrar un gráfico en línea con la información del monto de costos y montos de ventas de los últimos 6 meses.	

<b>Historia de usuario</b>	
<b>Número:</b> 27	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Reporte de rentabilidad por periodo	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Sebastián Aguilar	
<b>Descripción:</b> Deseo un reporte en PDF de rendimiento por territorios de ventas para un periodo en particular.	
<b>Validación:</b> El administrador seleccionará el periodo a analizar y dos distritos para comparar. Luego, al presionar el botón “Generar reporte” el sistema deberá mostrar un reporte en PDF con la información solicitada. También el reporte mostrará un gráfico circular sobre las ventas de cada distrito.	

<b>Historia de usuario</b>	
<b>Número:</b> 28	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Reporte de rentabilidad por distrito	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Lorena Flores	
<b>Descripción:</b> Deseo un reporte detallado en PDF sobre la rentabilidad de un distrito en particular y que también me muestre un gráfico de comparación entre las ventas, el margen de producción y el margen de contribución de los últimos 6 meses. A este reporte también se le añade un gráfico en línea sobre la evolución de la razón margen de contribución de los últimos 6 meses.	
<b>Validación:</b> El administrador seleccionará un distrito y presionará el botón “Generar reporte”. Luego, el sistema deberá mostrar el reporte en PDF con la información y gráficos solicitados para el distrito seleccionado.	

<b>Historia de usuario</b>	
<b>Número:</b> 29	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Reporte de rentabilidad por producto	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Sebastián Aguilar	
<b>Descripción:</b> Deseo un reporte detallado en PDF sobre la rentabilidad de un producto en particular por distritos. A este reporte también se le añada un gráfico en línea sobre la evolución de la razón margen de contribución de los últimos 6 meses.	
<b>Validación:</b> El usuario buscará el producto por nombre (no será búsqueda exacta) el cual mostrará una lista con los productos que coincidan con el criterio ingresado. Luego, el usuario al seleccionar el producto el sistema deberá mostrar un reporte en PDF con un gráfico de línea de la evolución de la razón margen de contribución del producto seleccionado de los últimos 6 meses.	

<b>Historia de usuario</b>	
<b>Número:</b> 30	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Reporte de rentabilidad por vendedor	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Sebastián Aguilar	
<b>Descripción:</b> Deseo un reporte detallado en PDF sobre la rentabilidad de un producto en particular por vendedores agrupados por últimos 6 meses en un determinado distrito.	
<b>Validación:</b> El usuario seleccionará un distrito y presionará el botón "Generar reporte". Luego, el sistema deberá mostrar un reporte en PDF con la información sobre la rentabilidad del vendedor. El reporte debe incluir un gráfico en pastel sobre las ventas de los vendedores.	



<b>Historia de usuario</b>	
<b>Número:</b> 31	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Visualización del sistema en Tablet y computadora	
<b>Prioridad:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Programador responsable:</b> Lorena Flores	
<b>Descripción:</b> Deseo que el sistema se pueda usar mediante una Tablet cuando me conecto por Wi – Fi.	
<b>Validación:</b> El administrador ingresará al sistema de la misma manera que ingresa por su computadora y este debe adaptarse al tamaño de la pantalla y mostrar las mismas opciones.	

### 3. Pila del producto

Tomando como base las historias de usuario mostrados en el paso anterior, se procedió a elaborar la pila del producto la cual consistió en ordenar las historias de usuario en base a su prioridad para luego agruparlos y asignarles la iteración que fueron desarrollados por el equipo de desarrollo.

Para ello, se separó el sistema en módulos las cuales se describen a continuación:

**Tabla 15.** *Módulos del sistema de información desarrollado*

<b>N°</b>	<b>Módulo</b>	<b>Abreviatura</b>	<b>Historia de usuario</b>
1	Operaciones	OPE	Del 1 al 14, 26
2	Reportes	REP	Del 16 al 25 y del 27 al 30
3	Seguridad	SEG	15

Fuente: Elaboración propia (2015)

La tabla 15 muestra los módulos y su abreviatura usados en el desarrollo del software y su relación con las historias de usuarios.

La siguiente tabla (tabla 16) muestra la pila del producto para el presente trabajo de investigación:

**Tabla 16. Pila del producto**

N°	N° Historia	Usuario	Módulo	Historia/Nombre	Prioridad	Riesgo	Estimación (Horas)	Iteración
1	1	AST	OPE	Registro de proveedores	Alta	Alto	12	1
2	5	AST	OPE	Búsqueda de productos	Alta	Alto	12	1
3	10	AST	OPE	Registrar orden de compra	Alta	Alto	12	1
4	11	AST	OPE	Registrar producto en la orden de compra	Alta	Alto	12	1
5	12	AST	OPE	Eliminar producto de la orden de compra	Alta	Alto	12	1
6	15	TODOS	SEG	Formulario de ingreso al sistema	Alta	Alto	18	1
7	16	ADM	REP	Reporte de costo variable unitario por periodos	Alta	Alto	30	1
8	17	ADM	REP	Reporte de costo variable unitario por producto	Alta	Alto	30	1
9	18	ADM	REP	Reporte del margen de contribución unitario por periodos	Alta	Alto	30	2
10	19	ADM	REP	Reporte de evolución del margen de contribución unitario por producto	Alta	Alto	30	2
11	20	ADM	REP	Reporte de punto de equilibrio por periodos	Alta	Alto	30	2
12	21	ADM	REP	Reporte de evolución del punto de equilibrio por producto	Alta	Alto	30	2
13	22	ADM	REP	Reporte de margen de seguridad por periodos	Alta	Alto	30	2
14	24	ADM	REP	Reporte de apalancamiento operativo por periodos	Alta	Alto	30	3

N°	N° Historia	Usuario	Módulo	Historia/Nombre	Prioridad	Riesgo	Estimación (Horas)	Iteración
15	25	ADM	REP	Reporte de apalancamiento operativo por producto	Alta	Alto	30	3
16	27	ADM	REP	Reporte de rentabilidad por periodo	Alta	Alto	30	3
17	28	ADM	REP	Reporte de rentabilidad por distrito	Alta	Alto	30	3
18	29	ADM	REP	Reporte de rentabilidad por producto	Alta	Alto	30	3
19	30	ADM	REP	Reporte de rentabilidad por vendedor	Alta	Alto	30	4
20	2	AST	OPE	Búsqueda de proveedores	Media	Medio	12	4
21	6	AST	OPE	Registrar abastecimiento	Media	Medio	12	4
22	7	AST	OPE	Buscar abastecimiento	Media	Medio	12	4
23	13	AST	OPE	Buscar orden de compra	Media	Medio	12	4
24	14	AST	OPE	Editar orden de compra	Media	Bajo	12	4
25	23	ADM	REP	Reporte de margen de seguridad por producto	Alta	Alto	30	4
26	26	ADM	OPE	Gráfico costos vs ventas	Media	Medio	18	4
27	3	AST	OPE	Actualizar datos del proveedor	Baja	Bajo	12	5
28	4	AST	OPE	Eliminar proveedor	Baja	Bajo	12	5
29	8	AST	OPE	Editar abastecimiento	Baja	Bajo	12	5
30	9	AST	OPE	Eliminar abastecimiento	Baja	Bajo	12	5
31	31	ADM	SEG	Visualización del sistema en Tablet y computadora	Alta	Alto	1	TODOS

Fuente: Elaboración propia (2015)

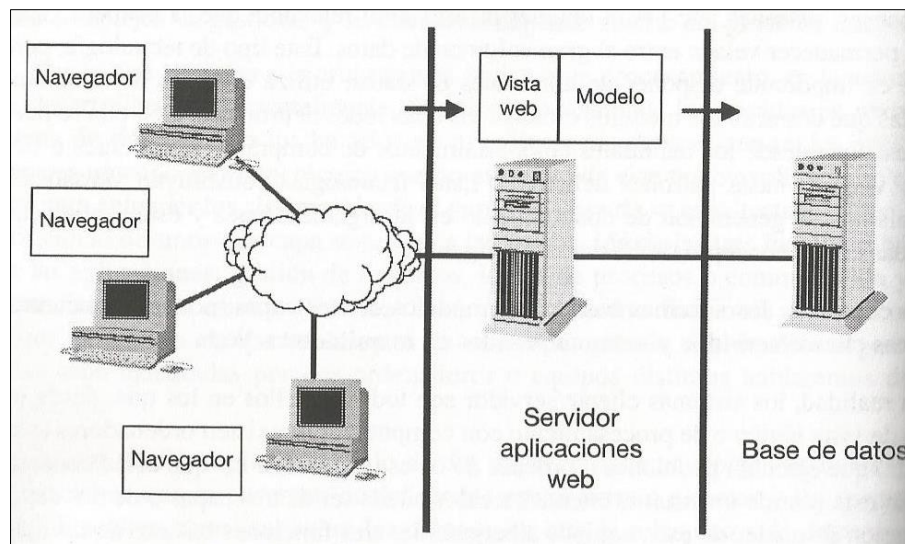
#### 4. Arquitectura de la solución

En esta sección, se describirá la arquitectura que se eligió y se desarrolló para el presente trabajo de investigación. Para ello, se planteó un estilo y patrón de diseño del sistema.

##### 4.1. Estilo arquitectónico

Para el estilo arquitectónico se escogió “Cliente-Servidor”, debido a que “es un modelo de aplicación distribuida (en dos o tres capas) en donde las tareas se reparten entre los proveedores de servicio, llamados servidores, y los consumidores de dichos servicios, llamados clientes” (De Pablos, y otros, 2013 pág. 163). Para el sistema desarrollado, se tiene un tipo de cliente (cliente web) y dos servidores (servidor web y de base de datos).

**Figura 27.** Esquema de una arquitectura web de 3 capas



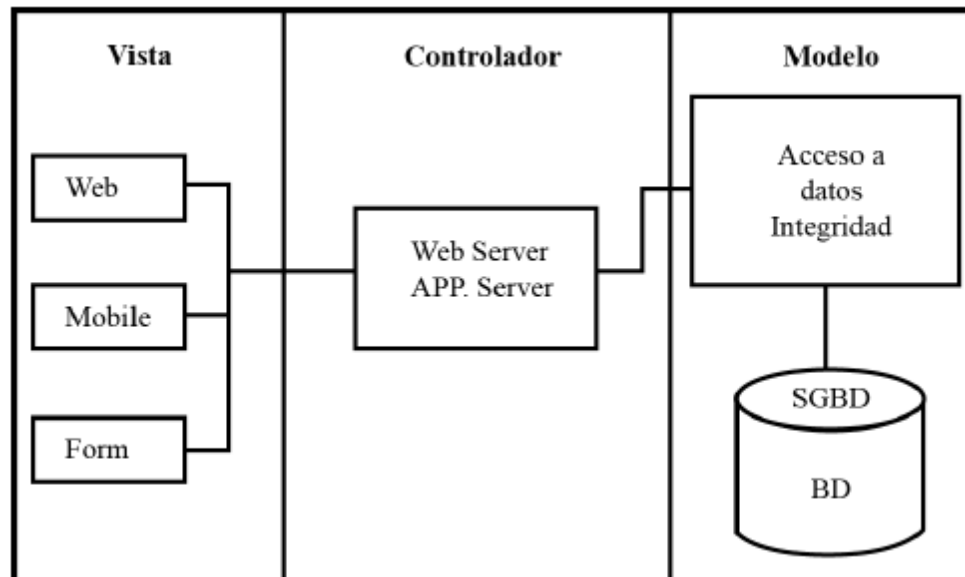
Fuente: De Pablos, López, Martín - Romo, & Medina (2013)

Como se aprecia en la figura 27, se observa la distribución de la arquitectura web en 3 capas, en donde la primera capa lo comprende el cliente (navegador web), la segunda el servidor de aplicaciones o servidor web y por último en la tercera capa la base de datos.

## 4.2. Patrón de diseño

Con respecto al patrón de diseño, se escogió el “Modelo - Vista - Controlador” (MVC). Esto debido a que posee una adecuada separación de capas el cual permite una independencia entre tecnologías y un mayor flujo de datos.

**Figura 28.** Patrón MCV asociado a la tecnología web



**Fuente:** Zorrilla Castro, De La Torre, Ramos Barroso, & Calvarro (2010)

La figura 28 muestra la estructura del patrón MVC con relación a la tecnología web. En ello se puede apreciar que la vista está asociada a una página web, un dispositivo móvil o un formulario, el controlador con el servidor web o un servidor de aplicaciones y el modelo con el acceso a los datos o una base de datos.

## 5. Desarrollo de las iteraciones

En esta sección, se procederá a desarrollar las iteraciones (sprint) del sistema de información propuesto. Para ello, se tomaron en cuenta las prioridades de las historias de usuarios y las horas estimadas, debido a que una iteración bajo la metodología Scrum se debe ubicar entre 02 y 04 semanas de desarrollo. A continuación, se detalla el desarrollo de las iteraciones.

### 5.1. Planificación del Sprint N° 1

La siguiente tabla muestra la primera iteración del sistema desarrollado:

**Tabla 17.** *Planificación de la primera iteración*

<b>Número Sprint:</b>	1		
<b>Fecha de inicio:</b>	02/03/2015	<b>Fecha fin:</b>	25/03/2015
<b>Días de duración:</b>	18	<b>Horas hombres estimadas:</b>	139
<b>Fecha de validación:</b>	26/03/2015	<b>Cantidad de historias:</b>	9
<b>N° Historia</b>	<b>Nombre de historia de usuario</b>	<b>Prioridad</b>	<b>Módulo</b>
31	Visualización del sistema en Tablet y computadora	Alta	SEG
1	Registro de proveedores	Alta	OPE
5	Búsqueda de productos	Alta	OPE
10	Registrar orden de compra	Alta	OPE
11	Registrar producto en la orden de compra	Alta	OPE
12	Eliminar producto de la orden de compra	Alta	OPE
15	Formulario de ingreso al sistema	Alta	SEG
16	Reporte de costo variable unitario por periodos	Alta	REP
17	Reporte de costo variable unitario por producto	Alta	REP

Fuente: Elaboración propia (2015)

La tabla 17 muestra la pila del producto del primer sprint. En ella se especifican las fechas de inicio y fin del sprint, así como la cantidad de horas/hombres estimadas, la duración en días, la fecha en cual el usuario valida el producto, las historias de usuarios a desarrollarse, el tipo de prioridad y el módulo al cual pertenece cada historia.

## 5.2. Pila del sprint N° 1

A continuación, la siguiente tabla detalla la pila del sprint en desarrollo.

**Tabla 18.** Pila del primer sprint

Sprint 1					
Inicio:	02/03/2015				
Duración:	139 horas				
Pila del sprint					
N° de historia	Nombre de la actividad	Tipo	Horas estimadas	Horas Incurridas	Responsable
31	Visualización del sistema en Tablet y PC	Req. No funcional	1	1	Lorena Flores
1	Registro de proveedores	Req. funcional	12	15	Lorena Flores
5	Búsqueda de productos	Req. funcional	12	10	Sebastián Aguilar
10	Registrar orden de compra	Req. funcional	12	10	Sebastián Aguilar
11	Registrar producto en la orden de compra	Req. funcional	12	12	Sebastián Aguilar
12	Eliminar producto de la orden de compra	Req. funcional	12	12	Lorena Flores
15	Formulario de ingreso al sistema	Req. funcional	18	15	Sebastián Aguilar
16	Reporte de costo variable unitario por periodos	Req. funcional	30	31	Sebastián Aguilar
17	Reporte de costo variable unitario por producto	Req. funcional	30	30	Lorena Flores
Total:			139	136	

Fuente: Elaboración propia (2015)

La tabla 18 muestra la pila del primer sprint. En ella se puede observar la relación de las historias con el tipo de requisito del sistema. También se observa las horas planificadas e incurridas y el responsable de la elaboración de cada historia.

### 5.3. Incremento del Sprint N° 1

En esta sección se muestran las partes o subsistemas entregados en la primera iteración completamente operativos.

#### 5.3.1. Registro de proveedores

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

##### A. Interfaz gráfica aprobado

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 29.** *Interfaz gráfica aprobada para el registro de proveedores*

The image shows a web form titled "Registrar proveedor". The form contains the following fields and controls:

- Razón social:** A text input field with the placeholder "Escribir texto".
- RUC:** A text input field with the placeholder "Escribir texto".
- Dirección:** A text input field with the placeholder "Escribir texto".
- Distrito:** A dropdown menu with the placeholder "Texto".
- Teléfono:** A text input field with the placeholder "Escribir texto".
- Correo:** A text input field with the placeholder "Escribir texto".
- Registrar:** A button with the text "Registrar".

**Fuente:** Elaboración propia (2015)

La figura 29 detalla la estructura del formulario para el registro del proveedor. En ella se puede observar los campos que van incluidos para el proveedor.



## B. Interfaz gráfica desarrollado

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 30.** *Interfaz gráfica desarrollada para el registro del proveedor*

**Fuente:** Elaboración propia (2015)

La figura 30 muestra la interfaz gráfica desarrollada para el registro del proveedor. En ella se puede observar que contiene los mismos campos mostrados en la interfaz gráfica aprobada por el cliente.

## C. Diseño de la tabla

En esta parte se muestra el diseño de la tabla elaborado para el proveedor.

**Figura 31.** *Diseño de la tabla proveedor.*

Nombre del campo	Tipo de dato
char_pvr_codigo	CHAR(6)
vchr_pvr_nombre	VARCHAR(80)
vchr_pvr_ruc	CHAR(11)
vchr_pvr_direccion	VARCHAR(80)
vchr_pvr_telefono	VARCHAR(11)
vchr_pvr_correo	VARCHAR(50)
dtim_pvr_registro	TIMESTAMP
char_dis_codigo	CHAR(4) (FK)

**Fuente:** Elaboración prueba (2015)

En la figura 31 se puede apreciar el diseño de la tabla proveedor. En ella se puede apreciar que contiene los campos mostrados en la interfaz gráfica.

#### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que registra el proveedor en la base de datos.

**Figura 32.** Procedimiento almacenado para el registro del proveedor

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_insertSupplier`(p_RazonSocial VARCHAR(80), p_Ruc CHAR(11), p_Direccion VARCHAR(80),
p_Distrito CHAR(4), p_Telefono VARCHAR(11), p_Correo VARCHAR(50))
COMMENT 'Procedimiento para registrar un proveedor'
BEGIN
-- declaramos una variable
DECLARE v_Codigo CHAR(6);
DECLARE v_Count INTEGER;
DECLARE k_FechaRegistro DATETIME DEFAULT NOW();
-- capturamos la cantidad de datos insertados
SET v_Count = (SELECT COUNT(char_pvrCodigo) FROM contad_tb_proveedor);
-- verificamos cuantos datos existen
IF v_Count = 0 THEN SET v_Codigo = 'PV0001';
ELSE
-- generamos el codigo
SET v_Codigo = (SELECT CONCAT('PV', RIGHT(MAX(RIGHT(char_pvrCodigo, 4) + 10001), 4)) FROM contad_tb_proveedor);
END IF;
-- insertamos el proveedor
INSERT INTO contad_tb_proveedor (char_pvrCodigo, vchr_pvrNombre, vchr_pvrRuc, vchr_pvrDireccion,
char_discodigo, vchr_pvrTelefono, vchr_pvrCorreo, dtim_pvrRegistro)
VALUES (v_Codigo, p_RazonSocial, p_Ruc, p_Direccion, p_Distrito, p_Telefono, p_Correo, k_FechaRegistro);
END;
```

**Fuente:** Elaboración propia (2015)

La figura 32 muestra la construcción del procedimiento almacenado para el registro del proveedor. En ella se ve que el código del proveedor es autogenerado como lo solicitó el usuario.

#### E. Programación en Java

En esta sección se muestra la programación en Java para el registro del proveedor.

**Figura 33.** Función en Java para el registro del proveedor

```
public boolean insertAll(Connection cn, TDbProveedor tProveedor)
    throws SQLException {

    //Preparamos la sentencia SQL
    String query = "{ call sp_insertSupplier(?, ?, ?, ?, ?, ?) }";

    //Instanciamos un objeto del tipo CallableStatement
    CallableStatement stm = null;

    try{

        //Conectamos el objeto stm con la sentencia sql
        stm = (CallableStatement) cn.prepareCall(query);

        //mandamos los atributos
        stm.setString("p_RazonSocial", tProveedor.getNombre());
        stm.setString("p_Ruc", tProveedor.getRuc());
        stm.setString("p_Direccion", tProveedor.getDireccion());
        stm.setString("p_Distrito", tProveedor.getDistrito());
        stm.setString("p_Telefono", tProveedor.getTelefono());
        stm.setString("p_Correo", tProveedor.getCorreo());

        //ejecutamos la sentencia
        stm.execute();

        //retornamos un valor verdadero
        return true;
    }
}
```

Fuente: Elaboración propia (2015)

La figura 33 muestra la programación desarrollada en Java que permite registrar un proveedor desde la aplicación web. En ella se puede observar la invocación del procedimiento almacenado construido.

### 5.3.2. Búsqueda de productos

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobado

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 36.** *Interfaz gráfica aprobada para la búsqueda de productos*

**Fuente:** Elaboración propia (2015)

La figura 34 detalla la estructura del formulario para la búsqueda del producto. En ella se puede observar que el resultado se debe mostrar en una tabla.

**B. Interfaz gráfica desarrollado**

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 35.** *Interfaz gráfica desarrollada para la búsqueda del producto*

Codigo	Nombre	Categoria	Marca	Stock mínimo	Stock actual	Precio	Última modificación	Acciones
674365	CINTA EMBALAJE 2 x 20Y	Cintas	Chino	1200	2361	0.58	2015-05-16 00:00:00.0	\$ 🗑
674366	CINTA EMBALAJE 2 x 90Y	Cintas	Chino	720	2808	2.29	2015-05-16 00:00:00.0	\$ 🗑
674367	CINTA MASKINGTAPE 1/2 x 40Y	Cintas	3M	900	1530	1.55	2015-05-16 00:00:00.0	\$ 🗑

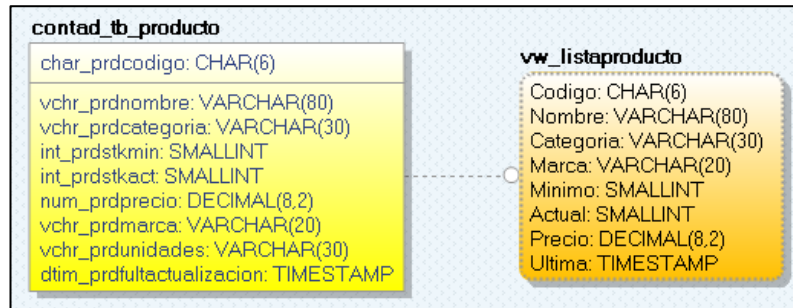
**Fuente:** Elaboración propia (2015)

La figura 35 muestra la interfaz gráfica desarrollada para la búsqueda de productos. En ella se puede observar que contiene la misma estructura mostrado en la interfaz gráfica aprobada.

### C. Diseño de la vista

En esta parte se muestra el diseño de la vista elaborado para la búsqueda de productos.

**Figura 36. Diseño de la vista lista producto**



**Fuente:** Elaboración prueba (2015)

En la figura 36 se puede apreciar el diseño de la vista de productos. En ella se puede apreciar los campos que contienen la vista y la tabla de donde se extrajeron los datos.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que selecciona la lista de productos conforme a un valor ingresado.

**Figura 37. Procedimiento almacenado para la búsqueda de productos**

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_getProductByName` (p_Producto VARCHAR(80))
COMMENT 'Procedimiento para extraer datos de un producto por su nombre'
BEGIN
    -- Seleccionamos los productos que cumplan con la condicion
    SELECT
        Codigo,
        Nombre,
        Categoria,
        Marca,
        Minimo,
        Actual,
        Precio,
        DATE(Ultima) AS Ultima
    FROM `vw_listaproducto`
    WHERE `Nombre` LIKE p_Producto;
END$$
```

**Fuente:** Elaboración propia (2015)

La figura 37 muestra la construcción del procedimiento almacenado que realiza la búsqueda del producto. En ella se

ve que el procedimiento devuelve valores aproximados de acuerdo con el criterio de búsqueda ingresado.

## E. Programación en Java

En esta sección se muestra la programación en Java para la búsqueda de productos.

**Figura 38.** *Función en Java para la búsqueda de productos*

```
public List<TDbProducto> getAll(Connection cn) throws SQLException {  
  
    //Creamos una variable que almacene los datos devueltos  
    List<TDbProducto> listaProducto = new ArrayList<TDbProducto>();  
  
    //Preparamos la sentencia SQL  
    String query = "{ call sp_getAllProducts()}";  
  
    //Instanciamos un objeto del tipo CallableStatement  
    CallableStatement stm = null;  
  
    //Instanciamos un objeto del tipo ResultSet  
    ResultSet rs = null;  
  
    try{  
  
        //Conectamos el objeto stm con la sentencia sql  
        stm = (CallableStatement) cn.prepareCall(query);  
  
        //Ejecutamos la sentencia  
        rs = stm.executeQuery(query);  
  
        //Capturamos todos los valores devueltos  
        while(rs.next()){  
  
            TDbProducto tProducto = new TDbProducto(  
                rs.getString("codigo"),
```

**Fuente:** Elaboración propia (2015)

La figura 38 muestra la programación desarrollada en Java que permite la búsqueda de productos desde la aplicación web. En ella se puede observar la invocación del procedimiento almacenado y el tipo de dato devuelto.

### 5.3.3. Registrar orden de compra

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobado

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 39.** *Interfaz gráfica aprobada para el registro de órdenes de compra*

**Fuente:** Elaboración propia (2015)

La figura 39 detalla la estructura del formulario para registro de órdenes de compra. En ella se puede observar los campos a considerar para el diseño de la tabla.

**B. Interfaz gráfica desarrollada**

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 40.** *Interfaz gráfica desarrollada para el registro de órdenes de compra*

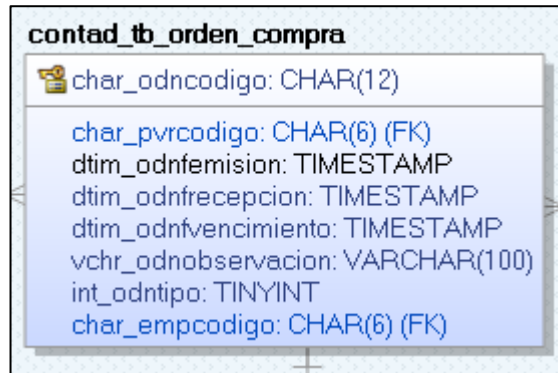
**Fuente:** Elaboración propia (2015)

La figura 40 muestra la interfaz gráfica desarrollada para el registro de órdenes de compra. En ella se puede observar que contiene la misma estructura mostrado en la interfaz gráfica aprobada.

### C. Diseño de la tabla

En esta parte se muestra el diseño de la tabla órdenes de compra.

**Figura 41. Diseño de la tabla orden de compra**



**Fuente:** Elaboración prueba (2015)

En la figura 41 se puede apreciar el diseño de la tabla orden de compra. En ella se puede apreciar los campos que contiene dicha tabla.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que selecciona la lista de productos conforme a un valor ingresado.

**Figura 42. Procedimiento almacenado para el registro de órdenes de compra**

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_insertOrden` (p_Proveedor CHAR(11), p_Empleado VARCHAR(30), p_Emision TIMESTAMP,
p_Vencimiento TIMESTAMP, p_Observacion VARCHAR(100))
COMMENT 'Procedimiento para insertar la cabecera de una orden de compra'
BEGIN
-- Declaramos las variables y constantes
DECLARE v_CodigoProveedor CHAR(6);
DECLARE v_NumOrden CHAR(12);
DECLARE v_Count INTEGER;
DECLARE k_TipoOrden SMALLINT DEFAULT 1;

-- capturamos la cantidad de datos insertados
SET v_Count = (SELECT COUNT('char_odncodigo') FROM 'contad_tb_orden_compra');

-- verificamos cuantos datos existen
IF v_Count = 0 THEN SET v_NumOrden = 'Nro_OC000001';
ELSE
-- generamos el codigo
SET v_NumOrden = (SELECT CONCAT('Nro_OC', RIGHT(MAX(RIGHT('char_odncodigo', 6) + 1000001), 6)) FROM 'contad_tb_orden_compra');
END IF;

-- Capturamos el codigo del proveedor
SELECT 'char_pvrcodigo' INTO v_CodigoProveedor FROM 'contad_tb_proveedor' WHERE 'vchr_pvruc' = p_Proveedor;

-- insertamos la cabecera
INSERT INTO 'contad_tb_orden_compra' ('char_odncodigo', 'char_pvrcodigo', 'char_empcodigo', 'dtim_odnfemision',
'dtim_odnfvencimiento', 'vchr_odnobservacion', 'int_odntipo')
VALUES (v_NumOrden, v_CodigoProveedor, p_Empleado, p_Emision, p_Vencimiento, p_Observacion, k_TipoOrden);
END;
```

**Fuente:** Elaboración propia (2015)



La figura 42 muestra la construcción del procedimiento almacenado que realiza el registro de una orden de compra. En ella se aprecia que el código de la orden es autogenerado.

## E. Programación en Java

En esta sección se muestra la programación en Java para el registro de una orden de compra.

**Figura 43.** *Función en Java para el registro de órdenes de compra*

```
public boolean insertAll(Connection cn, TDbOrdenCompra tOrden)
    throws SQLException {
    //Preparamos la sentencia SQL
    String query = "{ call sp_insertOrden(?, ?, ?, ?, ?) }";
    //Instanciamos un objeto del tipo CallableStatement
    CallableStatement stm = null;
    try{
        //Conectamos el objeto stm con la sentencia sql
        stm = (CallableStatement) cn.prepareCall(query);
        //mandamos los atributos
        stm.setString("p_Proveedor", tOrden.getProveedor());
        stm.setString("p_Empleado", tOrden.getEmpleado());
        stm.setString("p_Emision", tOrden.getEmision());
        stm.setString("p_Vencimiento", tOrden.getVencimiento());
        stm.setString("p_Observacion", tOrden.getObservacion());
        //ejecutamos la sentencia
        stm.execute();
        //retornamos un valor verdadero
        return true;
    }
```

**Fuente:** Elaboración propia (2015)

La figura 43 muestra la programación desarrollada en Java que permite el registro de órdenes de compras desde la aplicación web. En ella se puede observar la invocación del procedimiento almacenado.

### 5.3.4. Registrar producto en la orden de compra

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

### A. Interfaz gráfica aprobado

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 44.** *Interfaz gráfica aprobada para el registro de producto en la orden de compra*

Detalle de orden de compra

Detalle de la orden N° OC0001

Producto:  Cantidad:  Precio:

Columna 1	Columna 1	Columna 1	Columna 1
Texto	Texto	Texto	Texto
Texto	Texto	Texto	Texto
Texto	Texto	Texto	Texto

**Fuente:** Elaboración propia (2015)

La figura 44 detalla la estructura del formulario para registro de productos en la orden de compra. En ella se puede observar los campos a considerar para el diseño de la tabla.

### B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 45.** *Interfaz gráfica desarrollada para el registro de productos en órdenes de compra*

Registrar detalle de la orden N° OC0037

Producto:  Cantidad:  Precio:

Codigo	Producto	Cantidad	Precio	Total	Eliminar
203241	FOCO AHORRADOR 5W 2U	30	2.5	75.00	<input type="button" value="x"/>

**Fuente:** Elaboración propia (2015)

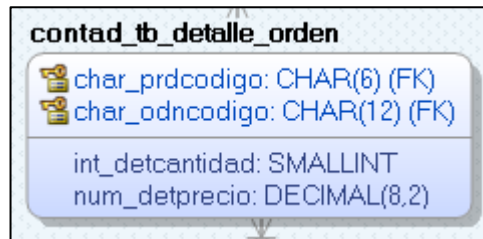
La figura 45 muestra la interfaz gráfica desarrollada para el registro de productos en la orden de compra. En ella se puede

observar que contiene la misma estructura mostrada en la interfaz gráfica aprobada.

### C. Diseño de la tabla

En esta parte se muestra el diseño de la tabla detalle de orden de compra.

**Figura 46.** *Diseño de la tabla detalle de orden de compra*



**Fuente:** Elaboración prueba (2015)

En la figura 46 se puede apreciar el diseño de la tabla detalle orden de compra. En ella se puede apreciar los campos que contiene dicha tabla.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que registra un producto en la orden de compra.

**Figura 47.** *Procedimiento almacenado para el registro de productos en órdenes de compra*

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_insertDetail`(p_NumOrdern CHAR(12), p_Producto CHAR(6),
p_Cantidad SMALLINT, p_Precio DECIMAL(8,2))
COMMENT 'Procedimiento que registra el detalle de una orden de compra'
BEGIN
-- Insertamos
INSERT INTO `contad_tb_detalle_orden` (`char_prdcodigo`, `char_odncodigo`, `int_detcantidad`, `num_detprecio`)
VALUES (p_Producto, p_NumOrdern, p_Cantidad, p_Precio);
END$$
```

**Fuente:** Elaboración propia (2015)

La figura 47 muestra la construcción del procedimiento almacenado que realiza el registro de productos en una orden de compra específica.

## E. Programación en Java

En esta sección se muestra la programación en Java para el registro de productos en una orden de compra específica.

**Figura 48.** *Función en Java para el registro de productos en órdenes de compra*

```
public boolean insertAll(Connection cn, TDbDetalleOrden tDetalle)
    throws SQLException {

    //Preparamos la sentencia SQL
    String query = "{ call sp_insertDetail(?, ?, ?, ?) }";

    //Instanciamos un objeto del tipo CallableStatement
    CallableStatement stm = null;

    try{

        //Conectamos el objeto stm con la sentencia sql
        stm = (CallableStatement) cn.prepareCall(query);

        //Enviamos los parametros
        stm.setString("p_NumOrdern", tDetalle.getNumorden());
        stm.setString("p_Producto", tDetalle.getProducto());
        stm.setInt("p_Cantidad", tDetalle.getCantidad());
        stm.setDouble("p_Precio", tDetalle.getPrecio());

        //ejecutamos la sentencia
        stm.execute();

        //retornamos un valor verdadero
        return true;
    }
}
```

**Fuente:** Elaboración propia (2015)

La figura 48 muestra la programación desarrollada en Java que permite el registro de productos en una determinada orden de compra. En ella se puede observar la invocación del procedimiento almacenado.

### 5.3.5. Eliminar producto de la orden de compra

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobado

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 49.** *Interfaz gráfica aprobada para eliminar producto en la orden de compra*

Detalle de orden de compra

Detalle de la orden N° OC0001

Producto:  Cantidad:  Precio:

Columna 1	Columna 1	Columna 1	Columna 1
Texto	Texto	Texto	Texto
Texto	Texto	Texto	Texto
Texto	Texto	Texto	Texto

**Fuente:** Elaboración propia (2015)

La figura 49 detalla la estructura del formulario para eliminar productos en la orden de compra.

**B. Interfaz gráfica desarrollado**

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 50.** *Interfaz gráfica desarrollada para eliminar productos en órdenes de compra*

Registrar detalle de la orden N° OC0037

Producto:  Cantidad:  Precio:

Codigo	Producto	Cantidad	Precio	Total	Eliminar
203241	FOCO AHORRADOR 5W 2U	30	2.5	75.00	<input type="button" value="✘"/>

**Fuente:** Elaboración propia (2015)

La figura 50 muestra la interfaz gráfica desarrollada para eliminar productos en órdenes de compra. En ella se puede observar que contiene la opción “eliminar” en la tabla donde se ubican los productos.

### C. Diseño de la tabla

Para esta historia de usuario, no se desarrolló una tabla o una vista.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que elimina un producto de la orden de compra.

**Figura 51.** *Procedimiento almacenado para eliminar productos en órdenes de compra*

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_deleteDetail`( p_Producto CHAR(6), p_NumOrden CHAR(6))
COMMENT 'Procedimiento que elimina el detalle de una orden de compra por producto'
BEGIN
  -- Eliminamos el detalle por producto
  DELETE FROM `contad_tb_detalle_orden`
  WHERE `char_prdcodigo` = p_Producto AND `char_odncodigo` = p_NumOrden;
END$$
```

**Fuente:** Elaboración propia (2015)

La figura 51 muestra la construcción del procedimiento almacenado que elimina un producto en una orden de compra específica.

### E. Programación en Java

En esta sección se muestra la programación en Java para el registro del proveedor.

**Figura 52.** Función en Java para eliminar un producto de una orden de compra

```
public boolean deleteAll(Connection cn, String producto, String numOrden)
    throws SQLException {

    //Preparamos la sentencia SQL
    String query = "{ call sp_deleteDetail(?, ?) }";

    //Instanciamos un objeto del tipo CallableStatement
    CallableStatement stm = null;

    try{

        //Conectamos el objeto stm con la sentencia sql
        stm = (CallableStatement) cn.prepareCall(query);

        //Enviamos los argumentos
        stm.setString("p_Producto", producto);
        stm.setString("p_NumOrden", numOrden);

        //Ejecutamos la sentencia
        stm.execute();

        //Retornamos el valor verdadero
        return true;
    }
```

Fuente: Elaboración propia (2015)

La figura 52 muestra la programación desarrollada en Java para eliminar un producto en una orden de compra determinada. En ella se puede observar la invocación del procedimiento almacenado.

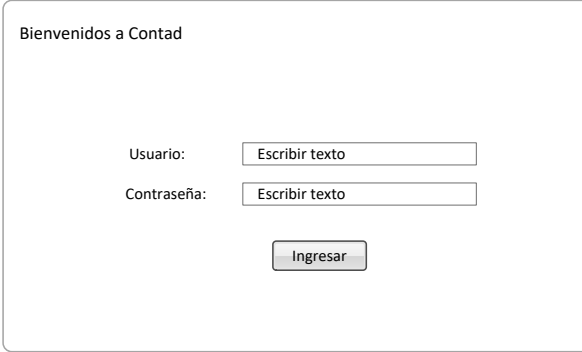
### 5.3.6. Formulario de ingreso al sistema

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobado

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 53.** *Interfaz gráfica aprobada para el ingreso al sistema*



Bienvendidos a Contad

Usuario:

Contraseña:

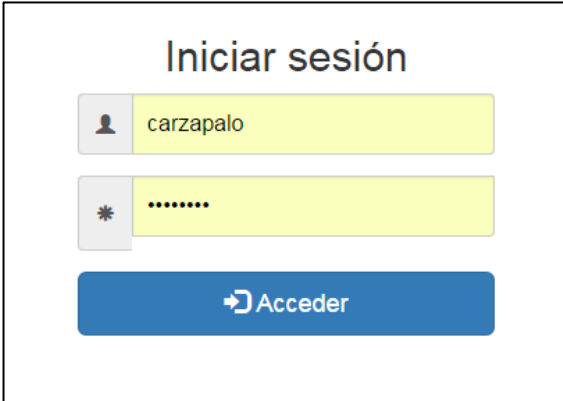
**Fuente:** Elaboración propia (2015)

La figura 53 detalla la estructura del formulario para el ingreso al sistema.

## **B. Interfaz gráfica desarrollada**

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 54.** *Interfaz gráfica desarrollada para el ingreso al sistema*



Iniciar sesión

**Fuente:** Elaboración propia (2015)

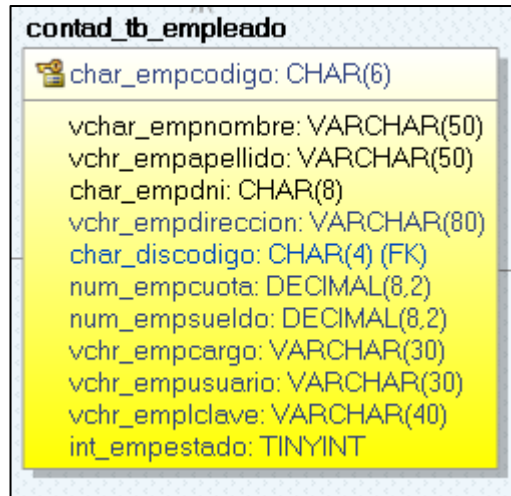
La figura 54 muestra la interfaz gráfica desarrollada para el ingreso al sistema.



### C. Diseño de la tabla

En esta parte se muestra el diseño de la tabla empleado.

**Figura 55.** *Diseño de la tabla empleado.*



contad_tb_empleado	
char_empcodigo	CHAR(6)
vchar_empnombre	VARCHAR(50)
vchr_empapellido	VARCHAR(50)
char_empdni	CHAR(8)
vchr_empdireccion	VARCHAR(80)
char_discodigo	CHAR(4) (FK)
num_empcuota	DECIMAL(8,2)
num_empsueldo	DECIMAL(8,2)
vchr_empcargo	VARCHAR(30)
vchr_empusuario	VARCHAR(30)
vchr_emplclave	VARCHAR(40)
int_empestado	TINYINT

**Fuente:** Elaboración propia (2015)

La figura 55 muestra el diseño de la tabla empleado. En ella se observa los campos involucrados que identifican a un empleado. Esta tabla a diferencia de las demás, procede desde la base de datos del sistema de ventas, por lo que, para la base de datos del presente sistema, se hizo un espejo de la tabla original, lo que en MySQL se le conoce como tablas federadas.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que valida el inicio de sesión de un empleado al sistema.

**Figura 56. Procedimiento almacenado para el acceso al sistema**

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_getEmployeeByUsername` (p_Usuario VARCHAR(30), p_Clave VARCHAR(32))
COMMENT 'Procedimiento para extraer los datos de un empleado por su usuario y clave'
BEGIN
    -- Validamos si el usuario existe y luego extraemos sus datos
    SELECT *
    FROM `vw_listaempleado`
    WHERE usuario = ( SELECT `vchr_empusuario`
                     FROM `contad_cb_empleado`
                     WHERE `vchr_empusuario` = p_Usuario AND `vchr_emplclave` = SHA1(p_Clave)
                     AND `vchr_empcargo` NOT IN ('Vendedor')));
END$$
```

**Fuente:** Elaboración propia (2015)

La figura 56 muestra la construcción del procedimiento almacenado que selecciona los datos de un empleado en donde coincidan con el usuario y clave ingresados y que además este usuario no posea el cargo de vendedor.

## E. Programación en Java

En esta sección se muestra la programación en Java para el registro del proveedor.

**Figura 57. Función en Java para validar el acceso al sistema**

```
public TdbEmpleado getByUsername(Connection cn, String username, String clave)
    throws SQLException {
    //creamos una variable
    TdbEmpleado tEmpleado = null;

    //Preparamos la sentencia SQL
    String query = "{ call sp_getEmployeeByUsername(?, ?) }";

    //Instanciamos un objeto del tipo CallableStatement
    CallableStatement stm = null;

    //Declaramos una variable del tipo resultset
    ResultSet rs = null;

    try{

        //Conectamos el objeto stm con la sentencia sql
        stm = (CallableStatement) cn.prepareCall(query);

        //mandamos los argumentos
        stm.setString("p_Usuario", username);
        stm.setString("p_Clave", clave);

        //ejecutamos la sentencia
        rs = stm.executeQuery();
    }
```

**Fuente:** Elaboración propia (2015)

La figura 57 muestra la programación desarrollada en Java para validar el acceso al sistema. En ella se puede observar la invocación del procedimiento almacenado.

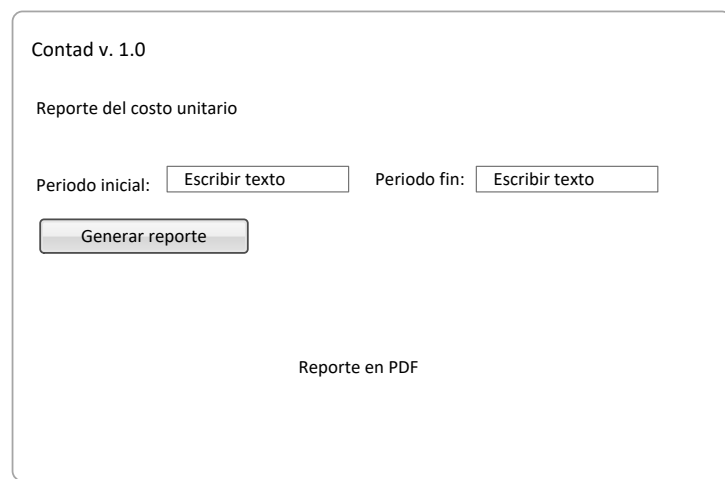
### 5.3.7. Reporte de costo variable unitario por periodos

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 58.** *Interfaz gráfica aprobada para generar el reporte del costo variable unitario por periodos*



Contad v. 1.0

Reporte del costo unitario

Periodo inicial:  Periodo fin:

[Reporte en PDF](#)

**Fuente:** Elaboración propia (2015)

La figura 58 detalla la estructura del formulario para generar el reporte del costo variable unitario por periodos. En ella se puede apreciar que se tienen que ingresar dos periodos (meses) para generar el reporte en PDF.

#### B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 59.** *Interfaz gráfica desarrollada para visualizar el detalle de costos de las facturas*

**Fuente:** Elaboración propia (2015)

La figura 59 muestra la interfaz gráfica desarrollada para generar el reporte del costo variable unitario por periodos. En ella se aprecia que el diseño coincide con lo aprobado por el cliente.

**C. Diseño de la tabla**

En esta parte se muestra el diseño de la tabla histórica que almacena el costo variable unitario.

**Figura 60.** *Diseño de la tabla histórica del costo variable unitario*

hsm_cvu_producto	
Periodo:	CHAR(7)
Producto:	CHAR(6)
UnidadesMinimas:	INTEGER
CostoTotalMinimo:	DECIMAL(8,2)
UnidadesMaximas:	DECIMAL(8,2)
CostoTotalMaximo:	DECIMAL(8,2)
CostoVariableUnitario:	DECIMAL(8,2)
CostosFijos:	DECIMAL(8,2)

**Fuente:** Elaboración propia (2015)

En la figura 60 se puede apreciar el diseño de la tabla que contiene todos los campos necesarios para generar el reporte del costo variable unitario por producto.

## D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que calcula el costo variable unitario y la almacena en la tabla correspondiente.

**Figura 61.** *Procedimiento almacenado que calcula el costo variable unitario por producto*

```
-- Capturamos los datos
SELECT fn_getMaxCantidad(v_Producto, v_Periodo) INTO v_MaxCantidad;
SELECT fn_getMinCantidad(v_Producto, v_Periodo) INTO v_MinCantidad;
SELECT fn_getMaxCost(v_Producto, v_Periodo) INTO v_MaxCostoTotal;
SELECT fn_getMinCost(v_Producto, v_Periodo) INTO v_MinCostoTotal;

IF v_MaxCantidad IS NULL THEN

    -- No se compro productos en el periodo
    SET v_MaxCantidad = 0;
END IF;

IF v_MinCantidad IS NULL THEN

    -- No se compro productos en el periodo
    SET v_MinCantidad = 0;
END IF;

-- Calculamos el Costo Variable Unitario
SET v_Cvu = (v_MaxCostoTotal - v_MinCostoTotal) / (v_MaxCantidad - v_MinCantidad);

-- Verificamos si es nulo
IF v_Cvu IS NULL THEN

    -- Lo colocamos como 0.0
    SET v_Cvu = 0.0;
END IF;

-- Calculamos el costo fijo
SET v_CostoFijo = v_MaxCostoTotal - (v_Cvu * v_MaxCantidad);
```

**Fuente:** Elaboración propia (2015)

La figura 61 muestra la construcción del procedimiento almacenado permite calcular el costo variable unitario de un determinado producto para un periodo en particular. En ello se puede observar que si en caso el producto no registra ventas o comprar en el periodo ingresado se asigna un valor igual a cero. De la misma manera, si el costo variable unitario arroja un valor nulo producto de que la cantidad máxima comprada ( $v\_MaxCantidad$ ) y la cantidad mínima comprada ( $v\_MinCantidad$ ) son iguales, se le asigna un valor de cero.

## E. Programación en Java

En esta sección se muestra la programación en Java para generar el reporte del costo variable unitario por periodos.

**Figura 62.** Procedimiento en Java para generar el reporte de costo variable unitario por periodos

```
private void reporteProductoCvu(HttpServletRequest request,
    HttpServletResponse response) {

    try{

        //Capturamos la conexion
        Connection conexion = AccesoDB.getConnection();

        //Establecemos la ruta del reporte
        String fileRepo = "/pe/corparzapalo/contad/app/appwebcontabilidad/reports/rpt_Costo";
        InputStream ioRepo = getClass().getResourceAsStream(fileRepo);

        //Creamos la lista de parametros
        Map<String, Object> parms = new HashMap<String, Object>();
        String periodoInicial = request.getParameter("pini");
        String periodoFinal = request.getParameter("pfin");
        parms.put("p_FecIni", periodoInicial);
        parms.put("p_FecFin", periodoFinal);

        //Creamos el reporte
        JasperReport jsRepo = JasperCompileManager.compileReport(ioRepo);
        byte[] bytes = JasperRunManager.runReportToPdf(jsRepo, parms, conexion);

        //Enviando el reporte
        response.setContentType("application/pdf");
        response.setContentLength(bytes.length);
        ServletOutputStream out = response.getOutputStream();
        out.write(bytes, 0, bytes.length);
        out.flush();
        out.close();
    }
}
```

**Fuente:** Elaboración propia (2015)

La figura 62 muestra la programación desarrollada en Java para generar el reporte del costo variable unitario del producto por periodos.

### 5.3.8. Reporte de costo variable unitario por producto

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 63.** *Interfaz gráfica aprobada para generar el reporte del costo variable unitario por producto*

The screenshot shows a web interface titled 'Contad v. 1.0'. Below the title is the heading 'Reporte del costo unitario'. There is a label 'Nombre del producto:' followed by a text input field containing 'Escribir texto' and a 'Buscar' button. Below this is a large rectangular area labeled 'Reporte en PDF'.

**Fuente:** Elaboración propia (2015)

La figura 63 detalla la estructura del formulario para generar el reporte del costo variable unitario por producto aprobada por el usuario. En ella se puede apreciar que se tienen que buscar el producto para luego generar el reporte.

## B. Interfaz gráfica desarrollado

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 64.** *Interfaz gráfica desarrollada para visualizar el detalle de costos de las facturas*

The screenshot shows a web interface titled 'Reporte de costo variable unitario'. It features two tabs: 'Mes actual' (selected) and 'Producto'. Below the tabs is the text 'Generar reporte de costo volumen unitario por producto'. There is a label 'Nombre del producto:' followed by a dropdown menu showing 'Ej. Cinta de embalaje' and a 'Buscar' button.

**Fuente:** Elaboración propia (2015)

La figura 64 muestra la interfaz gráfica desarrollada para generar el reporte del costo variable unitario por producto. En

ella se aprecia que el diseño coincide con lo aprobado por el cliente.

### C. Diseño de la tabla

Para este reporte se utilizó la misma tabla desarrollada para generar el reporte de costo variable unitario por periodos.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que selecciona la información sobre el costo variable unitario para un determinado producto.

**Figura 65.** *Procedimiento almacenado que selecciona el costo variable unitario por producto*

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_getCvuByProducto`(p_Producto CHAR(6))
  COMMENT 'Procedimiento que extrae datos del costo variable unitario por producto'
BEGIN
  -- Seleccionamos
  SELECT
    tmp.Periodo,
    tmp.`vchr_prdnombre` AS Producto,
    tmp.`unidades`,
    tmp.`costoTotal`,
    tmp.`cvu`,
    tmp.`costoFijo`
  FROM
    tmp_cvu_producto tmp INNER JOIN `contad_tb_producto` tmp ON tmp.`Producto` = tmp.`char_prdcodigo`
  WHERE
    tmp_producto = p_Producto
  LIMIT 6;
END$$
```

**Fuente:** Elaboración propia (2015)

La figura 65 muestra la construcción del procedimiento almacenado permite seleccionar el costo variable unitario para un determinado producto. En ella se puede apreciar que se seleccionan los 6 periodos solicitados por el cliente.

### E. Programación en Java

En esta sección se muestra la programación en Java para generar el reporte del costo variable unitario por producto.



**Figura 66.** Función en Java para generar el reporte del costo variable unitario por producto

```
private void reporteEvolucionProductoCvu(HttpServletRequest request,
    HttpServletResponse response) {

    try{

        //Capturamos la conexion
        Connection conexion = AccesoDB.getConnection();

        //Establecemos la ruta del reporte
        String fileRepo = "/pe/corparzapalo/contad/app/appwebcontabilidad/reports/rpt_Costo";
        InputStream ioRepo = getClass().getResourceAsStream(fileRepo);

        //Creamos la lista de parametros
        Map<String, Object> parms = new HashMap<String, Object>();
        String codigoProducto = request.getParameter("producto");
        parms.put("p_Producto", codigoProducto);

        //Creamos el reporte
        JasperReport jsRepo = JasperCompileManager.compileReport(ioRepo);
        byte[] bytes = JasperRunManager.runReportToPdf(jsRepo, parms, conexion);

        //Enviando el reporte
        response.setContentType("application/pdf");
        response.setContentLength(bytes.length);
        ServletOutputStream out = response.getOutputStream();
        out.write(bytes, 0, bytes.length);
        out.flush();
        out.close();
    }
}
```

**Fuente:** Elaboración propia (2015)

La figura 66 muestra la programación desarrollada en Java para generar el reporte del costo variable unitario por producto.

#### 5.4. Planificación del Sprint N° 2

La siguiente tabla muestra la segunda iteración del sistema desarrollado:

**Tabla 19.** Planificación de la segunda iteración

<b>Número Sprint:</b>	2		
<b>Fecha de inicio:</b>	27/03/2015	<b>Fecha fin:</b>	22/04/2015
<b>Días de duración:</b>	19	<b>Horas hombres estimadas:</b>	151
<b>Fecha de validación:</b>	23/04/2015	<b>Cantidad de historias:</b>	6
<b>N° Historia</b>	<b>Nombre de historia de usuario</b>	<b>Prioridad</b>	<b>Módulo</b>
31	Visualización del sistema en Tablet y computadora	Alta	SEG
18	Reporte de margen de contribución unitario por periodos	Alta	REP

N° Historia	Nombre de historia de usuario	Prioridad	Módulo
19	Reporte de evolución del margen de contribución unitario por producto	Alta	REP
20	Reporte de punto de equilibrio por periodos	Alta	REP
21	Reporte de evolución del punto de equilibrio por producto	Alta	REP
22	Reporte de margen de seguridad por periodos	Alta	REP

Fuente: Elaboración propia (2015)

La tabla 19 muestra la pila del producto del segundo sprint. En ella se especifican las fechas de inicio y fin del sprint, así como la cantidad de horas/hombres estimadas, la duración en días, la fecha en cual el usuario valida el producto, las historias de usuarios a desarrollarse, el tipo de prioridad y el módulo al cual pertenece cada historia.

### 5.5. Pila del sprint N° 2

En esta sección se muestra la pila del primer sprint. A continuación, la siguiente tabla detalla la pila del sprint en desarrollo.

**Tabla 20.** *Pila del segundo sprint*

Sprint 2					
Inicio:	27/03/2015				
Duración:	151 horas				
Pila del sprint					
N° de historia	Nombre de la actividad	Tipo	Horas estimadas	Horas Incurridas	Responsable
31	Visualización del sistema en Tablet y PC	Req. No funcional	1	1	Lorena Flores
18	Reporte del margen de contribución unitario por periodos	Req. funcional	30	36	Sebastián Aguilar
19	Reporte de evolución margen de contribución unitario por producto	Req. funcional	30	25	Lorena Flores

N° de historia	Nombre de la actividad	Tipo	Horas estimadas	Horas Incurridas	Responsable
20	Reporte de punto de equilibrio por periodos	Req. funcional	30	26	Sebastián Aguilar
21	Reporte de evolución del punto de equilibrio por producto	Req. funcional	30	30	Sebastián Aguilar
22	Reporte de margen de seguridad por periodos	Req. funcional	30	27	Lorena Flores
Total:			151	145	

Fuente: Elaboración propia (2015)

La tabla 20 muestra la pila del segundo sprint. En ella se puede observar la relación de las historias con el tipo de requisito del sistema. También se observa las horas planificadas e incurridas y el responsable de la elaboración de cada historia.

## 5.6. Incremento del Sprint N° 2

En esta sección se muestran las partes o subsistemas entregados en la segunda iteración completamente operativos.

### 5.6.1. Reporte del margen de contribución unitario por periodos

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 67.** *Interfaz gráfica aprobada para generar el reporte del margen de contribución unitario por periodos*

**Fuente:** Elaboración propia (2015)

La figura 67 detalla la estructura del formulario para generar el reporte del margen de contribución unitario por periodos. En ella se puede apreciar que se tienen que ingresar dos periodos (meses) para generar el reporte en PDF.

**B. Interfaz gráfica desarrollada**

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 68.** *Interfaz gráfica desarrollada para generar el reporte del margen de contribución unitario por periodos*

**Fuente:** Elaboración propia (2015)

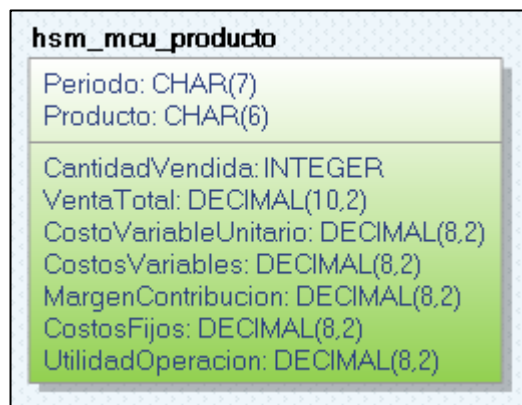
La figura 68 muestra la interfaz gráfica desarrollada para generar el reporte del margen de contribución unitario por

periodos. En ella se aprecia que el diseño coincide con lo aprobado por el cliente.

### C. Diseño de la tabla

En esta parte se muestra el diseño de la tabla histórica del margen de contribución unitario de los productos.

**Figura 69.** *Diseño de la tabla del margen de contribución unitario*



El diagrama muestra la estructura de la tabla **hsm\_mcu\_producto**. Los campos y sus tipos de datos son:

Nombre del campo	Tipo de dato
Periodo	CHAR(7)
Producto	CHAR(6)
CantidadVendida	INTEGER
VentaTotal	DECIMAL(10,2)
CostoVariableUnitario	DECIMAL(8,2)
CostosVariables	DECIMAL(8,2)
MargenContribucion	DECIMAL(8,2)
CostosFijos	DECIMAL(8,2)
UtilidadOperacion	DECIMAL(8,2)

**Fuente:** Elaboración propia (2015)

En la figura 69 se puede apreciar el diseño de la tabla que contiene todos los campos necesarios para generar el reporte del margen de contribución unitario por producto.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que calcula el margen de contribución unitario y la almacena en la tabla correspondiente.

**Figura 70.** Procedimiento almacenado que calcula el margen de contribución unitario por periodos

```

-- Seleccionamos la venta total
SELECT VentaTotal INTO v_Ventas
FROM vw_listaventaproductos
WHERE Producto = v_Producto AND Mes = p_Periodo;

-- Seleccionamos la cantidad vendida
SELECT CantidadVendida INTO v_Cantidad
FROM vw_listaventaproductos
WHERE Producto = v_Producto AND Mes = p_Periodo;

-- Seleccionamos el Costo variable unitario
SELECT CostoVariableUnitario INTO v_CostoVariableUnitario
FROM `hsm_cvu_producto`
WHERE `producto` = v_Producto AND `Periodo` = p_Periodo;

-- Calculamos los costos variables
SET v_CostosVariables = v_CostoVariableUnitario * v_Cantidad;

-- Calculamos el margen de contribucion
SET v_MargenContribucion = v_Ventas - v_CostosVariables;

-- Calculamos los costos fijos
SELECT CostosFijos INTO v_CostosFijos
FROM `cp_arzapalo`.`hsm_cvu_producto`
WHERE `producto` = v_Producto AND `Periodo` = p_Periodo;

-- Calculamos la utilidad
SET v_Utilidad = v_MargenContribucion - v_CostosFijos;

-- insertamos a la tabla temporal
INSERT INTO hsm_mcu_producto (`Periodo`, `Producto`, `CantidadVendida`,
                             `CostosVariables`, `MargenContribucion`,

```

**Fuente:** Elaboración propia (2015)

La figura 70 muestra la construcción del procedimiento almacenado permite calcular el margen de contribución unitario de un determinado producto para un periodo en particular y su posterior registro en la tabla correspondiente.

## E. Programación en Java

En esta sección se muestra la programación en Java para generar el reporte del margen de contribución unitario por periodos.

**Figura 71.** Procedimiento en Java para generar el reporte del margen de contribución unitario por periodos

```
private void reporteProductoMcu(HttpServletRequest request,
    HttpServletResponse response) {

    try{

        //Capturamos la conexion
        Connection conexion = AccesoDB.getConnection();

        //Establecemos la ruta del reporte
        String fileRepo = "/pe/corparzapalo/contad/app/appwebcontabilidad/reports/rpt_Margen
        InputStream ioRepo = getClass().getResourceAsStream(fileRepo);

        //Creamos la lista de parametros
        Map<String, Object> parms = new HashMap<String, Object>();
        String periodoInicial = request.getParameter("pini");
        String periodoFinal = request.getParameter("pfin");
        parms.put("p_FecIni", periodoInicial);
        parms.put("p_FecFin", periodoFinal);

        //Creamos el reporte
        JasperReport jsRepo = JasperCompileManager.compileReport(ioRepo);
        byte[] bytes = JasperRunManager.runReportToPdf(jsRepo, parms, conexion);

        //Enviando el reporte
        response.setContentType("application/pdf");
        response.setContentLength(bytes.length);
        ServletOutputStream out = response.getOutputStream();
        out.write(bytes, 0, bytes.length);
        out.flush();
        out.close();
    }
}
```

**Fuente:** Elaboración propia (2015)

La figura 71 muestra la programación desarrollada en Java para generar el reporte en PDF del margen de contribución unitario del producto por periodos.

### 5.6.2. Reporte de evolución del margen de contribución unitario por producto

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 72.** *Interfaz gráfica aprobada para generar el reporte de evolución del margen de contribución unitario por producto*

Contad v. 1.0

Reporte del margen de contribución unitario

Nombre del producto:

Reporte en PDF

**Fuente:** Elaboración propia (2015)

La figura 72 detalla la estructura del formulario para generar el reporte de evolución del margen de contribución unitario por producto. En ella se puede apreciar que se tienen que buscar el producto para luego generar el reporte.

## B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 73.** *Interfaz gráfica desarrollada para visualizar el reporte de evolución del margen de contribución unitario por producto*

Reporte del margen de contribución unitario

Mes actual Producto

Generar reporte de margen de contribución por producto

Nombre del producto:

**Fuente:** Elaboración propia (2015)



La figura 73 muestra la interfaz gráfica desarrollada para generar el reporte del margen de contribución por producto. En ella se aprecia que el diseño coincide con lo aprobado por el cliente.

### C. Diseño de la tabla

Para este reporte se utilizó la misma tabla desarrollada para generar el reporte de margen de contribución unitario por periodos.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que selecciona la información sobre el margen de contribución unitario para un determinado producto.

**Figura 74.** Procedimiento almacenado que selecciona el margen de contribución unitario por producto

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_getMcUByProducto`(p_Producto CHAR(6))
COMMENT 'Procedimiento que extrae datos del margen de contribucion unitario por producto'
BEGIN
  -- Seleccionamos los datos
  SELECT
    hsm.Periodo,
    tbp.`vchr_prdnombre` AS Producto,
    hsm.`CantidadVendida`,
    hsm.`VentaTotal`,
    hsm.`CostoVariableUnitario`,
    hsm.`CostosVariables`,
    hsm.`MargenContribucion`,
    hsm.`CostosFijos`,
    hsm.`UtilidadOperacion`
  FROM
    hsm_mc_u_producto hsm INNER JOIN `contad_tb_producto` tbp ON hsm.`Producto` = tbp.`char_prdcodigo`
  WHERE
    hsm.producto = p_Producto
  LIMIT 6;
END$$
```

**Fuente:** Elaboración propia (2015)

La figura 74 muestra la construcción del procedimiento almacenado permite seleccionar el margen de contribución unitario de un determinado producto para elaborar el reporte correspondiente.

## E. Programación en Java

En esta sección se muestra la programación en Java para generar el reporte de evolución del margen de contribución unitario por producto.

**Figura 75.** Procedimiento en Java para generar el reporte de evolución del margen de contribución unitario por producto

```
private void reporteEvolucionProductoMcu(HttpServletRequest request,
    HttpServletResponse response) {
    try{
        //Capturamos la conexion
        Connection conexion = AccesoDB.getConnection();

        //Establecemos la ruta del reporte
        String fileRepo = "/pe/corparzapalo/contad/app/appwebcontabilidad/reports/rpt_Margen
        InputStream ioRepo = getClass().getResourceAsStream(fileRepo);

        //Creamos la lista de parametros
        Map<String, Object> parms = new HashMap<String, Object>();
        String codigoProducto = request.getParameter("producto");
        parms.put("p_Producto", codigoProducto);

        //Creamos el reporte
        JasperReport jsRepo = JasperCompileManager.compileReport(ioRepo);
        byte[] bytes = JasperRunManager.runReportToPdf(jsRepo, parms, conexion);

        //Enviando el reporte
        response.setContentType("application/pdf");
        response.setContentLength(bytes.length);
        ServletOutputStream out = response.getOutputStream();
        out.write(bytes, 0, bytes.length);
        out.flush();
        out.close();
    }
}
```

**Fuente:** Elaboración propia (2015)

La figura 75 muestra la programación desarrollada en Java para generar el reporte de evolución del margen de contribución unitario por producto.

### 5.6.3. Reporte del punto de equilibrio por periodos

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 76.** *Interfaz gráfica aprobada para generar el reporte del punto de equilibrio por periodos*

The screenshot shows a web form titled 'Contad v. 1.0' with the subtitle 'Reporte de punto de equilibrio'. It contains two input fields: 'Periodo inicial:' and 'Periodo final:', each followed by a text box containing the placeholder 'Escribir texto'. Below these fields is a button labeled 'Generar reporte'. At the bottom of the form is a large rectangular area labeled 'Reporte en PDF'.

**Fuente:** Elaboración propia (2015)

La figura 76 detalla la estructura del formulario para generar el reporte del punto de equilibrio por periodos. En ella se puede apreciar que se tienen que ingresar dos periodos (meses) para generar el reporte en PDF.

## B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 77.** *Interfaz gráfica desarrollada para visualizar el reporte del punto de equilibrio por periodos*

The screenshot shows a web form titled 'Reporte de punto de equilibrio'. It features a navigation bar with 'Mes actual' and 'Producto'. Below the navigation bar is the text 'Generar reporte del punto de equilibrio por periodos'. The form includes two date pickers: 'Periodo inicial:' and 'Periodo final:', each followed by a text box containing the placeholder '..... de ----'. A 'Generar reporte' button is located to the right of the second date picker.

**Fuente:** Elaboración propia (2015)

La figura 77 muestra la interfaz gráfica desarrollada para generar el reporte del punto de equilibrio por periodos. En ella se aprecia que el diseño coincide con lo aprobado por el cliente.

### C. Diseño de la tabla

En esta parte se muestra el diseño de la tabla donde se almacenan las unidades y ventas de equilibrio del producto.

**Figura 78.** *Diseño de la tabla histórica del punto de equilibrio*

hsm_veq_producto	
Periodo:	CHAR(7)
Producto:	CHAR(6)
CostosFijos:	DECIMAL(8,2)
MargenContribucionUnitario:	DECIMAL(8,2)
UnidadesEquilibrio:	INTEGER
PrecioVentaUnitario:	DECIMAL(8,2)
RazonMargenContribucion:	DECIMAL(8,2)
MontoEquilibrio:	DECIMAL(10,2)

**Fuente:** Elaboración propia (2015)

En la figura 78 se puede apreciar el diseño de la tabla que contiene todos los campos necesarios para generar el reporte del punto de equilibrio por periodos.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que calcula el punto de equilibrio en unidades y en monto.

**Figura 79.** *Procedimiento almacenado que calcula el punto de equilibrio*

```

-- Capturamos los valores
SET v_CostosFijos = (SELECT fn_getCostoFijo(v_Producto, p_Periodo));
SET v_Mcu = (SELECT fn_getMargenContribucion(v_Producto, p_Periodo));
SET v_Rmc = (SELECT fn_getRazonMargenContribucion(v_Producto, p_Periodo));
SET v_PrecioUnitario = (SELECT fn_getPrecioVentaUnitario(p_Periodo, v_Producto));

-- Calculamos las ventas de equilibrio en unidades
SET v_VentasEquilibrioUnidades = (v_CostosFijos / v_Mcu);

-- Calculamos las ventas de equilibrio en monto
SET v_VentasEquilibrioMonto = (v_CostosFijos / v_Rmc);

-- Insertamos en la tabla
INSERT INTO hsm_veq_producto (Periodo, Producto, UnidadesEquilibrio, MontoEquilibrio,
                             CostosFijos, MargenContribucionUnitario, PrecioVentaUnitario,
                             RazonMargenContribucion)
VALUES(p_Periodo, v_Producto, v_VentasEquilibrioUnidades, v_VentasEquilibrioMonto, v_CostosFijos,
       v_Mcu, v_PrecioUnitario, v_Rmc);
    
```

**Fuente:** Elaboración propia (2015)

La figura 79 muestra la construcción del procedimiento almacenado permite calcular el punto de equilibrio en unidades vendidas y unidades monetarias de un determinado producto para un periodo en particular y su posterior registro en la tabla correspondiente.

## E. Programación en Java

En esta sección se muestra la programación en Java para generar el reporte del punto de equilibrio por periodos.

**Figura 80.** *Procedimiento en Java para generar el reporte en PDF de punto de equilibrio por periodos*

```
private void reporteProductoequ(HttpServletRequest request,
    HttpServletResponse response) {
    try{
        //Capturamos la conexion
        Connection conexion = AccesoDB.getConnection();

        //Establecemos la ruta del reporte
        String fileRepo = "/pe/corparzapalo/contad/app/appwebcontabilidad/reports/rpt_Punto
        InputStream ioRepo = getClass().getResourceAsStream(fileRepo);

        //Creamos la lista de parametros
        Map<String, Object> parms = new HashMap<String, Object>();
        String periodoInicial = request.getParameter("pini");
        String periodoFinal = request.getParameter("pfin");
        parms.put("p_FecIni", periodoInicial);
        parms.put("p_FecFin", periodoFinal);

        //Creamos el reporte
        JasperReport jsRepo = JasperCompileManager.compileReport(ioRepo);
        byte[] bytes = JasperRunManager.runReportToPdf(jsRepo, parms, conexion);

        //Enviando el reporte
        response.setContentType("application/pdf");
        response.setContentLength(bytes.length);
        ServletOutputStream out = response.getOutputStream();
        out.write(bytes, 0, bytes.length);
        out.flush();
        out.close();
    }
}
```

**Fuente:** Elaboración propia (2015)

La figura 80 muestra la programación desarrollada en Java para generar el reporte en PDF del punto de equilibrio por periodos.

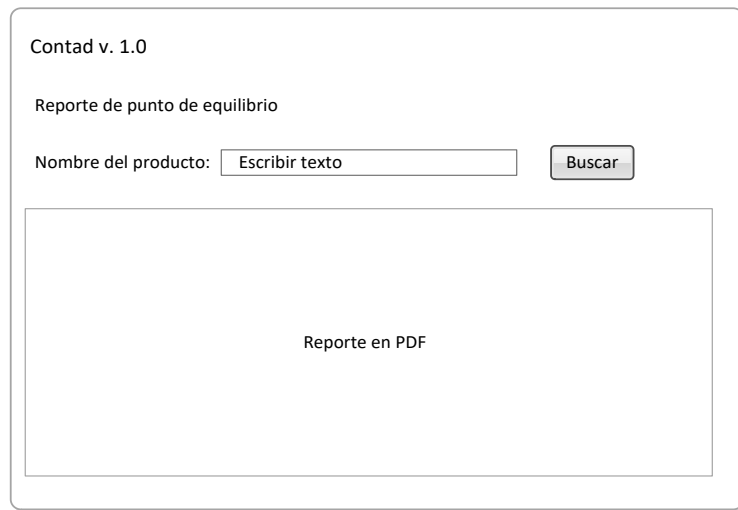
### 5.6.4. Reporte de evolución del punto de equilibrio por producto

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

## A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 81.** *Interfaz gráfica aprobada para generar el reporte de evolución del punto de equilibrio por producto*



Contad v. 1.0

Reporte de punto de equilibrio

Nombre del producto:

Reporte en PDF

**Fuente:** Elaboración propia (2015)

La figura 81 muestra la estructura del formulario para generar el reporte de evolución del punto de equilibrio por producto. En ella se puede apreciar que se tienen que buscar el producto para luego generar el reporte.

## B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 82.** Interfaz gráfica desarrollada para visualizar el reporte de evolución del punto de equilibrio por producto



**Fuente:** Elaboración propia (2015)

La figura 82 muestra la interfaz gráfica desarrollada para generar el reporte de evolución del punto de equilibrio por producto. En ella se aprecia que el diseño coincide con lo aprobado por el cliente.

**C. Diseño de la tabla**

Para este reporte se utilizó la misma tabla desarrollada para generar el reporte de punto de equilibrio por periodos.

**D. Programación en SQL**

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que selecciona la información sobre el punto de equilibrio para un determinado producto.

**Figura 83.** Procedimiento almacenado que selecciona el punto de equilibrio por producto

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_getEquByProducto`(p_Producto CHAR(6))
COMMENT 'Procedimiento que extrae el punto de equilibrio de un determinado producto'
BEGIN
    -- Seleccionamos los datos
    SELECT
        hsm.Periodo,
        tbp.vchr_prdnombre AS Producto,
        hsm.'CostosFijos',
        hsm.'MargenContribucionUnitario',
        hsm.'UnidadesEquilibrio',
        hsm.'PrecioVentaUnitario',
        hsm.'RazonMargenContribucion',
        hsm.'MontoEquilibrio'
    FROM
        hsm_veq_producto hsm INNER JOIN `contad_tb_producto` tbp ON hsm.'Producto' = tbp.'char_prdcodigo'
    WHERE
        hsm.Producto = p_Producto
    LIMIT 6;
```

**Fuente:** Elaboración propia (2015)

La figura 83 muestra la construcción del procedimiento almacenado permite seleccionar el punto de equilibrio en unidades y en monto para un determinado producto para elaborar el reporte correspondiente. También se puede observar que solo extrae información de los 6 últimos meses como lo solicitó el usuario.

## E. Programación en Java

En esta sección se muestra la programación en Java para generar el reporte de evolución del punto de equilibrio por producto.

**Figura 84.** *Procedimiento en Java para generar el reporte de evolución del punto de equilibrio por producto*

```
private void reporteEvolucionProductoEqu(HttpServletRequest request,
    HttpServletResponse response) {

    try{

        //Capturamos la conexion
        Connection conexion = AccesoDB.getConnection();

        //Establecemos la ruta del reporte
        String fileRepo = "/pe/corparzapalo/contad/app/appwebcontabilidad/reports/rpt_PuntoD
        InputStream ioRepo = getClass().getResourceAsStream(fileRepo);

        //Creamos la lista de parametros
        Map<String, Object> parms = new HashMap<String, Object>();
        String codigoProducto = request.getParameter("producto");
        parms.put("p_Producto", codigoProducto);

        //Creamos el reporte
        JasperReport jsRepo = JasperCompileManager.compileReport(ioRepo);
        byte[] bytes = JasperRunManager.runReportToPdf(jsRepo, parms, conexion);

        //Enviando el reporte
        response.setContentType("application/pdf");
        response.setContentLength(bytes.length);
        ServletOutputStream out = response.getOutputStream();
        out.write(bytes, 0, bytes.length);
        out.flush();
        out.close();
    }
}
```

**Fuente:** Elaboración propia (2015)

La figura 84 muestra la programación desarrollada en Java para generar el reporte en PDF de evolución del punto de equilibrio por producto.

### 5.6.5. Reporte de margen de seguridad por periodos

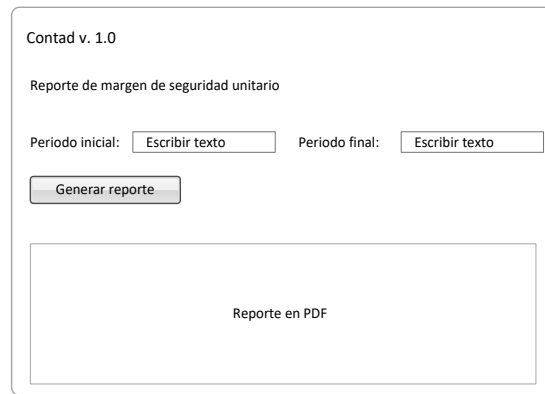
En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.



### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 85.** *Interfaz gráfica aprobada para generar el reporte de margen de seguridad por periodos*



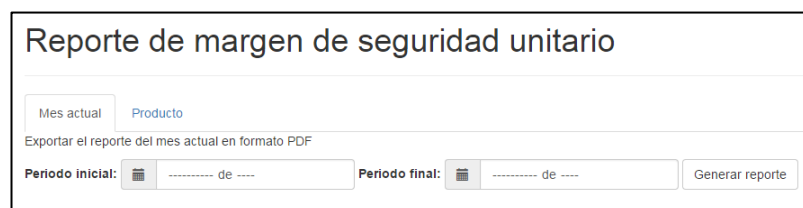
**Fuente:** Elaboración propia (2015)

La figura 85 detalla la estructura del formulario para generar el reporte de margen de seguridad por periodos. En ella se puede apreciar que se tienen que ingresar dos periodos (meses) para generar el reporte en PDF.

### B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 86.** *Interfaz gráfica desarrollada para visualizar el reporte de margen de seguridad por periodos*



**Fuente:** Elaboración propia (2015)

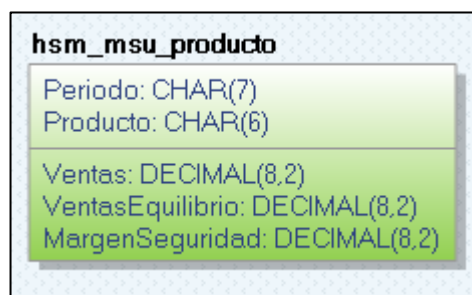
La figura 86 muestra la interfaz gráfica desarrollada para generar el reporte de margen de seguridad por periodos. En

ella se aprecia que el diseño coincide con lo aprobado por el cliente.

### C. Diseño de la tabla

En esta parte se muestra el diseño de la tabla donde se almacenan los datos necesarios para generar el reporte.

**Figura 87.** *Diseño de la tabla histórica del margen de seguridad*



El diagrama muestra el diseño de la tabla `hsm_msu_producto`. La tabla está representada como un rectángulo con un fondo verde claro y un borde negro. Dentro del rectángulo, se listan los campos de la tabla con sus respectivos tipos de datos y longitudes:

hsm_msu_producto	
Periodo:	CHAR(7)
Producto:	CHAR(6)
Ventas:	DECIMAL(8,2)
VentasEquilibrio:	DECIMAL(8,2)
MargenSeguridad:	DECIMAL(8,2)

**Fuente:** Elaboración propia (2015)

En la figura 87 se puede apreciar el diseño de la tabla que contiene todos los campos necesarios para generar el reporte de margen de seguridad por periodos.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que calcula el punto de equilibrio en unidades y en monto.

**Figura 88.** *Procedimiento almacenado que calcula el margen de seguridad*

```
-- Seleccionamos la venta total
SELECT VentaTotal INTO v_Ventas
FROM vw_listaventaproductos
WHERE Producto = v_Producto AND Mes = p_Periodo;

-- Seleccionamos las ventas de equilibrio
SELECT 'MontoEquilibrio' INTO v_Ventaseq
FROM 'hsm_veq_producto'
WHERE Producto = v_Producto AND 'Periodo' = p_Periodo;

-- Calculamos el margen de seguridad
SET v_MargenSeguridad = (v_Ventas - v_Ventaseq) / v_Ventas;

-- Insertamos en la tabla
INSERT INTO 'hsm_msu_producto' ('Periodo', 'Producto', 'Ventas',
                              |'VentasEquilibrio', 'MargenSeguridad')
VALUES(p_Periodo, v_Producto, v_Ventas, v_Ventaseq, v_MargenSeguridad);
```

**Fuente:** Elaboración propia (2015)

La figura 88 muestra la construcción del procedimiento almacenado permite calcular el margen de seguridad de un determinado producto para un periodo en particular y su posterior registro en la tabla correspondiente.

## E. Programación en Java

En esta sección se muestra la programación en Java para generar el reporte de margen de seguridad por periodos.

**Figura 89.** *Función en Java para generar el reporte en PDF de margen de seguridad por periodos*

```
private void reporteProductoMsu(HttpServletRequest request,
    HttpServletResponse response) {

    try{

        //Capturamos la conexion
        Connection conexion = AccesoDB.getConnection();

        //Establecemos la ruta del reporte
        String fileRepo = "/pe/corparzapalo/contad/app/appwebcontabilidad/reports/rpt_Margen";
        InputStream ioRepo = getClass().getResourceAsStream(fileRepo);

        //Creamos la lista de parametros
        Map<String, Object> parms = new HashMap<String, Object>();
        String periodoInicial = request.getParameter("pini");
        String periodoFinal = request.getParameter("pfin");
        parms.put("p_FecIni", periodoInicial);
        parms.put("p_FecFin", periodoFinal);

        //Creamos el reporte
        JasperReport jsRepo = JasperCompileManager.compileReport(ioRepo);
        byte[] bytes = JasperRunManager.runReportToPdf(jsRepo, parms, conexion);

        //Enviando el reporte
        response.setContentType("application/pdf");
        response.setContentLength(bytes.length);
        ServletOutputStream out = response.getOutputStream();
        out.write(bytes, 0, bytes.length);
        out.flush();
        out.close();
    }
}
```

**Fuente:** Elaboración propia (2015)

La figura 89 muestra la programación desarrollada en Java para generar el reporte en PDF del margen de seguridad por periodos.

### 5.7. Planificación del Sprint N° 3

La siguiente tabla muestra la tercera iteración del sistema desarrollado:

**Tabla 21.** *Planificación de la tercera iteración*

<b>Número Sprint:</b>	3		
<b>Fecha de inicio:</b>	24/04/2015	<b>Fecha fin:</b>	20/05/2015
<b>Días de duración:</b>	19	<b>Horas hombres estimadas:</b>	151
<b>Fecha de validación:</b>	21/05/2015	<b>Cantidad de historias:</b>	7
<b>N° Historia</b>	<b>Nombre de historia de usuario</b>	<b>Prioridad</b>	<b>Módulo</b>
31	Visualización del sistema en Tablet y computadora	Alta	SEG
23	Reporte de margen de seguridad por producto	Alta	REP
24	Reporte de apalancamiento operativo por periodos	Alta	REP
25	Reporte de apalancamiento operativo por producto	Alta	REP
27	Reporte de rentabilidad por periodo	Alta	REP
28	Reporte de rentabilidad por distrito	Alta	REP
29	Reporte de rentabilidad por producto	Alta	REP

Fuente: Elaboración propia (2015)

La tabla 21 muestra la pila del producto del tercer sprint. En ella se especifican las fechas de inicio y fin del sprint, así como la cantidad de horas/hombres estimadas, la duración en días, la fecha en cual el

usuario valida el producto, las historias de usuarios a desarrollarse, el tipo de prioridad y el módulo al cual pertenece cada historia.

### 5.8. Pila del sprint N° 3

En esta sección se muestra la pila del tercer sprint. A continuación, la siguiente tabla detalla la pila del sprint en desarrollo.

**Tabla 22. Pila del tercer sprint**

Sprint 3					
Inicio:		24/04/2015			
Duración:		151 horas			
Pila del sprint					
N° de historia	Nombre de la actividad	Tipo	Horas estimadas	Horas Incurridas	Responsable
31	Visualización del sistema en Tablet y computadora	Req. No funcional	1	1	Lorena Flores
23	Reporte de margen de seguridad por producto	Req. funcional	30	30	Lorena Flores
24	Reporte de apalancamiento operativo por periodos	Req. funcional	15	15	Sebastián Aguilar
25	Reporte de apalancamiento operativo por producto	Req. funcional	15	18	Lorena Flores
27	Reporte de rentabilidad por periodo	Req. funcional	30	35	Sebastián Aguilar
28	Reporte de rentabilidad por distrito.	Req. funcional	30	25	Lorena Flores
29	Reporte de rentabilidad por producto	Req. funcional	30	25	Sebastián Aguilar
Total:			151	149	

Fuente: Elaboración propia (2015)

La tabla 22 muestra la pila del primer sprint. En ella se puede observar la relación de las historias con el tipo de requisito del sistema. También se observa las horas planificadas e incurridas y el responsable de la elaboración de cada historia.

## 5.9. Incremento del Sprint N° 3

En esta sección se muestran las partes o subsistemas entregados en la tercera iteración completamente operativos.

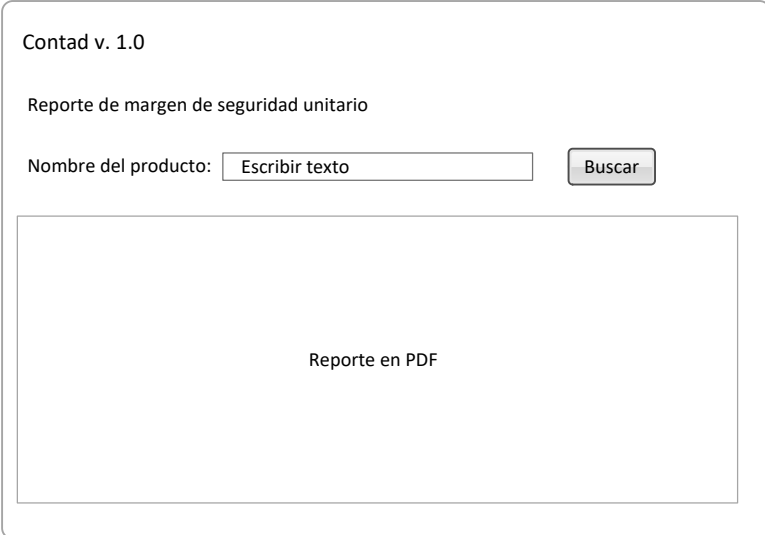
### 5.9.1. Reporte del margen de seguridad por producto

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 90.** *Interfaz gráfica aprobada para generar el reporte del margen de seguridad por producto*



Contad v. 1.0

Reporte de margen de seguridad unitario

Nombre del producto:

Reporte en PDF

**Fuente:** Elaboración propia (2015)

La figura 90 detalla la estructura del formulario para generar el reporte del margen de seguridad por producto.

En ella se puede apreciar que se tienen que buscar el producto para luego generar el reporte.

## B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 91.** *Interfaz gráfica desarrollada para generar el reporte de margen de seguridad por producto*



The screenshot shows a web form titled "Reporte de margen de seguridad unitario". It features two dropdown menus: "Mes actual" and "Producto". Below these is the instruction "Genera un reporte de Costo volumen unitario por producto". There is a text input field labeled "Nombre del producto:" with a small icon to its left, containing the text "Ej. Cinta de embalaje". To the right of this field is a "Buscar" button with a magnifying glass icon.

**Fuente:** Elaboración propia (2015)

La figura 91 muestra la interfaz gráfica desarrollada para generar el reporte del margen de seguridad por periodos. En ella se aprecia que el diseño coincide con lo aprobado por el usuario.

## C. Diseño de la tabla

Para este reporte se utilizó la misma tabla desarrollada para generar el reporte de margen de seguridad por periodos.

## D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que selecciona la información sobre el margen de seguridad unitario para un determinado producto.

**Figura 92.** Procedimiento almacenado que selecciona el margen de seguridad por producto

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_getMsuByProducto`(p_Producto CHAR(7))
COMMENT 'Procedimiento que lista el margen de seguridad de un determinado producto'
BEGIN
-- Seleccionamos los datos
SELECT
    hsm.Periodo,
    tbp.`vchr_prdnombre` AS Producto,
    hsm.`ventas`,
    hsm.`ventasEquilibrio`,
    hsm.`margenSeguridad`
FROM
    `hsm_msu_producto` hsm INNER JOIN `contad_tb_producto` tbp ON hsm.`Producto` = tbp.`char_prdcodigo`
WHERE
    hsm.Producto = p_Producto
LIMIT 6;
END$$
```

**Fuente:** Elaboración propia (2015)

La figura 92 muestra la construcción del procedimiento almacenado permite seleccionar el margen de seguridad de un determinado producto para generar el reporte correspondiente.

## E. Programación en Java

En esta sección se muestra la programación en Java para generar el reporte del margen de seguridad por producto.

**Figura 93.** Programación en Java para generar el reporte de margen de seguridad por producto

```
private void reporteEvolucionProductoMsu(HttpServletRequest request,
    HttpServletResponse response) {
    try{
        //Capturamos la conexion
        Connection conexion = AccesoDB.getConnection();

        //Establecemos la ruta del reporte
        String fileRepo = "/pe/corparzapalo/contad/app/appwebcontabilidad/reports/rpt_Margen";
        InputStream ioRepo = getClass().getResourceAsStream(fileRepo);

        //Creamos la lista de parametros
        Map<String, Object> parms = new HashMap<String, Object>();
        String codigoProducto = request.getParameter("producto");
        parms.put("p_Producto", codigoProducto);

        //Creamos el reporte
        JasperReport jsRepo = JasperCompileManager.compileReport(ioRepo);
        byte[] bytes = JasperRunManager.runReportToPdf(jsRepo, parms, conexion);

        //Enviando el reporte
        response.setContentType("application/pdf");
        response.setContentLength(bytes.length);
        ServletOutputStream out = response.getOutputStream();
        out.write(bytes, 0, bytes.length);
        out.flush();
        out.close();
    }
}
```

**Fuente:** Elaboración propia (2015)



La figura 93 muestra la programación desarrollada en Java para generar el reporte en PDF del margen de seguridad por producto.

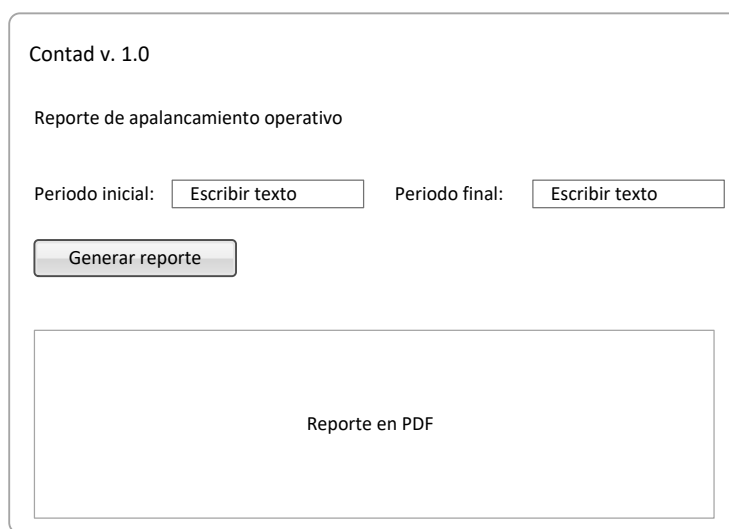
### 5.9.2. Reporte de apalancamiento operativo por periodos

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 94.** *Interfaz gráfica aprobada para generar el reporte de apalancamiento operativo por periodos*



Contad v. 1.0

Reporte de apalancamiento operativo

Periodo inicial:  Periodo final:

Reporte en PDF

**Fuente:** Elaboración propia (2015)

La figura 94 detalla la estructura del formulario para generar el reporte de apalancamiento operativo por periodos. En ella se puede apreciar que se tienen que ingresar dos periodos (meses) para generar el reporte en PDF.

### B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 95.** *Interfaz gráfica desarrollada para visualizar el reporte de apalancamiento operativo por periodos*

**Fuente:** Elaboración propia (2015)

La figura 95 muestra la interfaz gráfica desarrollada para generar el reporte de apalancamiento operativo por periodos. En ella se aprecia que el diseño coincide con lo aprobado por el usuario.

### C. Diseño de la tabla

En esta parte se muestra el diseño de la tabla donde se almacenan los datos necesarios para generar el reporte.

**Figura 96.** *Diseño de la tabla histórica del apalancamiento operativo*

hsm_aou_producto	
Periodo:	CHAR(7)
Producto:	CHAR(6)
MargenContribucionUnitario:	DECIMAL(8,2)
UtilidadOperacion:	DECIMAL(8,2)
ApalancamientoOperativo:	DECIMAL(8,2)

**Fuente:** Elaboración propia (2015)

En la figura 96 se puede apreciar el diseño de la tabla que contiene todos los campos necesarios para generar el reporte de apalancamiento operativo por periodos.

## D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que calcula el apalancamiento operativo.

**Figura 97.** *Procedimiento almacenado que calcula el apalancamiento operativo por periodos*

```
-- Seleccionamos el margen de contribucion unitario por producto
SET v_MargenContribucion = (SELECT MargenContribucion
                             FROM hsm_mcu_producto
                             WHERE `Producto` = v_Producto AND `Periodo` = p_Periodo);

-- Seleccionamos la utilidad
SET v_Utilidad = (SELECT `UtilidadOperacion`
                  FROM `hsm_mcu_producto`
                  WHERE `Producto` = v_Producto AND `Periodo` = p_Periodo);

-- Calculamos el apalancamiento operativo
SET v_Apalancamiento = v_MargenContribucion / v_Utilidad;

-- Insertamos los datos
INSERT INTO `hsm_aou_producto` (`Periodo`, `Producto`, `MargenContribucionUnitario`,
                                `UtilidadOperacion`, `ApalancamientoOperativo`)
VALUES(p_Periodo, v_Producto, v_MargenContribucion, v_Utilidad, v_Apalancamiento);
```

**Fuente:** Elaboración propia (2015)

La figura 97 muestra la construcción del procedimiento almacenado permite calcular el apalancamiento operativo de un determinado producto para un periodo en particular y su posterior registro en la tabla correspondiente.

## E. Programación en Java

En esta sección se muestra la programación en Java para generar el reporte de apalancamiento operativo por periodos.

**Figura 98.** *Función en Java para generar el reporte en PDF de apalancamiento operativo por periodos*

```
private void reporteProductoAou(HttpServletRequest request,
    HttpServletResponse response) {
    try{
        //Capturamos la conexion
        Connection conexion = AccesoDB.getConnection();

        //Establecemos la ruta del reporte
        String fileRepo = "/pe/corparzapalo/contad/app/appwebcontabilidad/reports/rpt_Apala
        InputStream ioRepo = getClass().getResourceAsStream(fileRepo);

        //Creamos la lista de parametros
        Map<String, Object> parms = new HashMap<String, Object>();
        String periodoInicial = request.getParameter("pini");
        String periodoFinal = request.getParameter("pfin");
        parms.put("p_FecIni", periodoInicial);
        parms.put("p_FecFin", periodoFinal);

        //Creamos el reporte
        JasperReport jsRepo = JasperCompileManager.compileReport(ioRepo);
        byte[] bytes = JasperRunManager.runReportToPdf(jsRepo, parms, conexion);

        //Enviando el reporte
        response.setContentType("application/pdf");
        response.setContentLength(bytes.length);
        ServletOutputStream out = response.getOutputStream();
        out.write(bytes, 0, bytes.length);
        out.flush();
        out.close();
    }
}
```

**Fuente:** Elaboración propia (2015)

La figura 98 muestra la programación desarrollada en Java para generar el reporte en PDF del apalancamiento operativo por periodos.

### 5.9.3. Reporte de apalancamiento operativo por producto

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 99.** *Interfaz gráfica aprobada para generar el reporte de apalancamiento operativo por producto*

Contad v. 1.0

Reporte de apalancamiento operativo

Nombre del producto:

Reporte en PDF

**Fuente:** Elaboración propia (2015)

La figura 99 detalla la estructura del formulario para generar el reporte de apalancamiento operativo por producto. En ella se puede apreciar que se tienen que buscar el producto para luego generar el reporte.

## B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 100.** *Interfaz gráfica desarrollada para visualizar el reporte de apalancamiento operativo por producto*

Reporte de apalancamiento operativo

Mes actual Producto

Generar el reporte del apalancamiento operativo por producto

Nombre del producto:

**Fuente:** Elaboración propia (2015)

La figura 100 muestra la interfaz gráfica desarrollada para generar el reporte del apalancamiento operativo por producto. En ella se aprecia que el diseño coincide con lo aprobado por el cliente.

### C. Diseño de la tabla

Para este reporte se utilizó la misma tabla desarrollada para generar el reporte de apalancamiento operativo por periodo.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que selecciona la información sobre el apalancamiento operativo para un determinado producto.

**Figura 101.** *Procedimiento almacenado que muestra el apalancamiento operativo de un producto*

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_getAopByProduct`(p_Producto CHAR(6))
COMMENT 'Procedimiento que selecciona el apalancamiento operativo por producto'
BEGIN
  -- Seleccionamos
  SELECT
    hsm.Periodo,
    ttp.`vchr_prdnombre` AS Producto,
    hsm.`MargenContribucionUnitario`,
    hsm.`UtilidadOperacion`,
    hsm.`ApalancamientoOperativo`
  FROM
    `hsm_aou_producto` hsm INNER JOIN `contad_tb_producto` ttp ON hsm.`Producto` = ttp.`char_prdcodigo`
  WHERE
    hsm.producto = p_Producto
  LIMIT 6;
END
```

**Fuente:** Elaboración propia (2015)

La figura 101 muestra la construcción del procedimiento almacenado permite mostrar el apalancamiento operativo para un determinado producto para elaborar el reporte correspondiente.

### E. Programación en Java

En esta sección se muestra la programación en Java para generar el reporte de evolución del margen de contribución unitario por producto.

**Figura 102.** *Función en Java para generar el reporte en PDF del apalancamiento operativo por producto*

```
private void reporteProductoAouPorProducto(HttpServletRequest request,
    HttpServletResponse response) {
    // TODO Auto-generated method stub
    try{

        //Capturamos la conexion
        Connection conexion = AccesoDB.getConnection();

        //Establecemos la ruta del reporte
        String fileRepo = "/pe/corparzapalo/contad/app/appwebcontabilidad/reports/rpt_Apala
        InputStream ioRepo = getClass().getResourceAsStream(fileRepo);

        //Creamos la lista de parametros
        Map<String, Object> parms = new HashMap<String, Object>();
        String codigoProducto = request.getParameter("producto");
        parms.put("p_Producto", codigoProducto);

        //Creamos el reporte
        JasperReport jsRepo = JasperCompileManager.compileReport(ioRepo);
        byte[] bytes = JasperRunManager.runReportToPdf(jsRepo, parms, conexion);

        //Enviando el reporte
        response.setContentType("application/pdf");
        response.setContentLength(bytes.length);
        ServletOutputStream out = response.getOutputStream();
        out.write(bytes, 0, bytes.length);
        out.flush();
        out.close();
    }
}
```

**Fuente:** Elaboración propia (2015)

La figura 102 muestra la programación desarrollada en Java para generar el reporte en PDF de apalancamiento por producto. En ella se puede apreciar la invocación del reporte.

#### 5.9.4. Reporte de rentabilidad por periodo

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

##### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 103.** *Interfaz gráfica aprobada para generar el reporte de rentabilidad por periodo*

Contad v. 1.0

Reportes de rentabilidad

Periodo:  Distrito 1:  Distrito 2:

Reporte en PDF

**Fuente:** Elaboración propia (2015)

La figura 103 detalla la estructura del formulario para generar el reporte de rentabilidad por periodo. En ella se puede apreciar que se tienen que seleccionar el periodo (mes) y seleccionar dos distritos para generar el reporte en PDF.

**B. Interfaz gráfica desarrollada**

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 104.** *Interfaz gráfica desarrollada para visualizar el reporte rentabilidad por periodos*

Reportes de rentabilidad

Periodo Distrito Producto

Generar el reporte de rentabilidad por periodos

Periodo:  Distrito 1:  Distrito 2:

**Fuente:** Elaboración propia (2015)


La figura 104 muestra la interfaz gráfica desarrollada para generar el reporte de rentabilidad por periodos. En ella se aprecia que el diseño coincide con lo aprobado por el usuario.



### C. Diseño de la tabla

En esta parte se muestra el diseño de la tabla donde se almacenan los datos necesarios para el reporte de rentabilidad.

**Figura 105.** *Diseño de la tabla histórica de la rentabilidad del producto*



hsm\_rentabilidad\_producto

Periodo: CHAR(7)
Distrito: CHAR(4)
Empleado: CHAR(6)
Producto: CHAR(6)
CantidadVendida: INTEGER
Subtotal: DECIMAL(8,2)
Comision: DECIMAL(8,2)
CostoVariableUnitario: DECIMAL(8,2)
CostoVentasVariable: DECIMAL(8,2)
MargenCompras: DECIMAL(8,2)
MargenContribucion: DECIMAL(8,2)
RazonMargenContribucion: DECIMAL(8,2)

**Fuente:** Elaboración propia (2015)

En la figura 105 se puede apreciar el diseño de la tabla histórica que contiene todos los datos necesarios para generar el reporte de rentabilidad por periodo.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que calcula los datos necesarios para generar el reporte de rentabilidad del producto.

**Figura 106. Procedimiento almacenado que genera el reporte de rentabilidad por periodos**

```
-- Capturamos los valores
SELECT CantidadVendida, Subtotal, Comision INTO v_CantidadVendida, v_Ventas, v_Comisiones
FROM vw_ventasporidistrito
WHERE Periodo = p_Periodo AND Distrito = v_Distrito
      AND Empleado = v_Empleado AND Producto = v_Producto;

-- Realizamos las operaciones
SET v_CostoVariableUnitario = (SELECT CostoVariableUnitario
                              FROM hsm_cvu_producto
                              WHERE PePeriodo = p_Periodo AND Producto = v_Producto);
SET v_CostoVentasVariable = v_CostoVariableUnitario * v_CantidadVendida;
SET v_MargenCompras = v_Ventas - v_CostoVentasVariable;
SET v_MargenContribucion = v_MargenCompras - v_Comisiones;
SET v_RazonMargenContribucion = v_MargenContribucion / v_Ventas;

-- Insertamos en la tabla historica
INSERT INTO hsm_rentabilidad_producto ('Periodo', 'Distrito', 'Empleado', 'Producto', 'CantidadVen
                                     'Subtotal', 'Comision', 'CostoVariableUnitario', 'CostoVen
                                     'MargenCompras', 'MargenContribucion', 'RazonMargenContrib
VALUES (p_Periodo, v_Distrito, v_Empleado, v_Producto,
       v_CantidadVendida, v_Ventas, v_Comisiones, v_CostoVariableUnitario, v_CostoVentasVariable,
       v_MargenCompras, v_MargenContribucion, v_RazonMargenContribucion);
```

**Fuente:** Elaboración propia (2015)

La figura 106 muestra la construcción del procedimiento almacenado permite calcular los datos necesarios para generar el reporte de rentabilidad por periodos de un determinado producto para un periodo en particular y su posterior registro en la tabla correspondiente.

### E. Programación en Java

En esta sección se muestra la programación en Java para generar el reporte de rentabilidad por periodos.

**Figura 107. Función en Java para generar el reporte en PDF de rentabilidad por periodos**

```
private void reporteRentabilidadDistrito(
    HttpServletRequest request, HttpServletResponse response) {
    // TODO Auto-generated method stub
    try{
        //Capturamos la conexión
        Connection conexion = AccesoDB.getConnection();

        //Establecemos la ruta del reporte
        String fileRepo = "/pe/corparzapalo/contad/app/appwebcontabilidad/reports/rpt_Rentab
        InputStream ioRepo = getClass().getResourceAsStream(fileRepo);

        //Creamos la lista de parametros
        Map<String, Object> parms = new HashMap<String, Object>();
        String periodo = request.getParameter("periodo");
        String distrito_1 = request.getParameter("distrito1");
        String distrito_2 = request.getParameter("distrito2");
        parms.put("p_Periodo", periodo);
        parms.put("p_Distrito_1", distrito_1);
        parms.put("p_Distrito_2", distrito_2);

        //Creamos el reporte
        JasperReport jsRepo = JasperCompileManager.compileReport(ioRepo);
        byte[] bytes = JasperRunManager.runReportToPdf(jsRepo, parms, conexion);

        //Enviando el reporte
        response.setContentType("application/pdf");
        response.setContentLength(bytes.length);
        ServletOutputStream out = response.getOutputStream();
        out.write(bytes, 0, bytes.length);
        out.flush();
    }
}
```

**Fuente:** Elaboración propia (2015)

La figura 107 muestra la programación desarrollada en Java para generar el reporte en PDF de rentabilidad por periodos. En ella se aprecia la invocación del reporte.

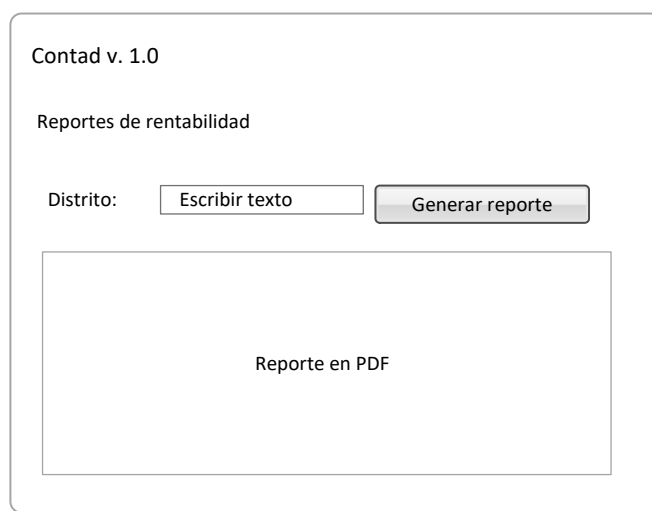
### 5.9.5. Reporte de rentabilidad por distrito

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 108.** *Interfaz gráfica aprobada para generar el reporte de rentabilidad por distrito*



The screenshot shows a web application window titled "Contad v. 1.0". Inside the window, there is a section labeled "Reportes de rentabilidad". Below this, there is a label "Distrito:" followed by a text input field containing the placeholder text "Escribir texto". To the right of the input field is a button labeled "Generar reporte". Below these elements is a large rectangular area labeled "Reporte en PDF", which is currently empty.

**Fuente:** Elaboración propia (2015)

La figura 108 detalla la estructura del formulario para generar el reporte de rentabilidad por distrito. En ella se puede apreciar que se tienen que seleccionar un distrito para luego generar el reporte.

#### B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 109.** *Interfaz gráfica desarrollada para visualizar el reporte de rentabilidad por distrito*



Reportes de rentabilidad

Periodo Distrito Producto

Generar el reporte de rentabilidad por territorios de ventas

Distrito:

Seleccione

Generar reporte

**Fuente:** Elaboración propia (2015)

La figura 109 muestra la interfaz gráfica desarrollada para generar el reporte de rentabilidad por distrito. En ella se aprecia que el diseño coincide con lo aprobado por el usuario.

### C. Diseño de la tabla

Para este reporte se utilizó la misma tabla desarrollada para generar el reporte rentabilidad por periodo.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que selecciona la información sobre el punto de equilibrio para un determinado producto.

**Figura 110.** Procedimiento almacenado que muestra los datos para el reporte de rentabilidad por distrito

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_getRentabilidadByDistrito`(p_Distrito CHAR(4))
COMMENT 'Procedimiento que muestra datos de la rentabilidad del producto por distrito'
BEGIN
  SELECT
    ham.`Periodo` AS Periodo,
    tbd.`vchr_disnombre` AS Distrito,
    SUM(ham.Subtotal) AS Ventas,
    SUM(ham.`MargenCompras`) AS MargenCompras,
    SUM(ham.Comision) AS GastosVentaVariables,
    ((SUM(ham.`MargenCompras`) - SUM(ham.Comision)) / SUM(ham.Subtotal)) AS MargenContribucion,
    ((SUM(ham.`MargenCompras`) - SUM(ham.Comision)) / SUM(ham.Subtotal)) AS RazonMargenContribucion
  FROM
    `contad_tb_distrito` tbd INNER JOIN `ham_rentabilidad_producto` ham ON tbd.`char_discodigo` = ham.`Distrito`
  WHERE
    ham.`Distrito` = p_Distrito
  GROUP BY
    Periodo, Distrito
  LIMIT
    6;
END##
```

Fuente: Elaboración propia (2015)

La figura 110 muestra la construcción del procedimiento almacenado permite generar el reporte de rentabilidad por distrito de los últimos 6 meses.

### E. Programación en Java

En esta sección se muestra la programación en Java para generar el reporte de rentabilidad por distrito.

**Figura 111.** Función en Java para generar el reporte de rentabilidad por distrito

```
private void reporteEvolucionRentabilidadDistrito(
    HttpServletRequest request, HttpServletResponse response) {
    // TODO Auto-generated method stub
    try{
        //Capturamos la conexion
        Connection conexion = AccesoDB.getConnection();

        //Establecemos la ruta del reporte
        String fileRepo = "/pe/corparzapalo/contad/app/appwebcontabilidad/reports/rpt_Rentab
        InputStream ioRepo = getClass().getResourceAsStream(fileRepo);

        //Creamos la lista de parametros
        Map<String, Object> parms = new HashMap<String, Object>();
        String codigoDistrito = request.getParameter("distrito");
        parms.put("p_Distrito", codigoDistrito);

        //Creamos el reporte
        JasperReport jsRepo = JasperCompileManager.compileReport(ioRepo);
        byte[] bytes = JasperRunManager.runReportToPdf(jsRepo, parms, conexion);

        //Enviando el reporte
        response.setContentType("application/pdf");
        response.setContentLength(bytes.length);
        ServletOutputStream out = response.getOutputStream();
        out.write(bytes, 0, bytes.length);
        out.flush();
        out.close();
    }
}
```

Fuente: Elaboración propia (2015)

La figura 111 muestra la programación desarrollada en Java para generar el reporte en PDF de rentabilidad por distrito.

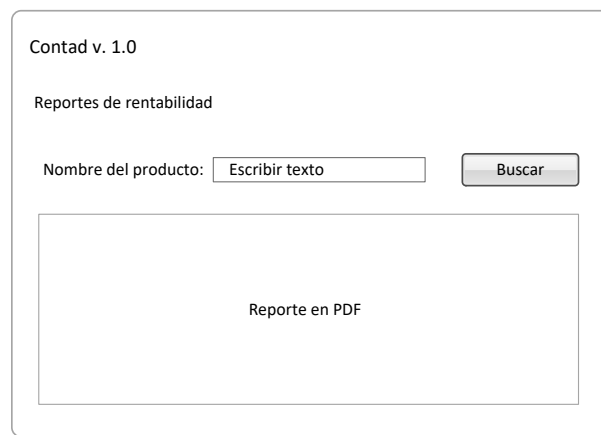
### 5.9.6. Reporte de rentabilidad por producto

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 112.** *Interfaz gráfica aprobada para generar el reporte de rentabilidad por producto*



The screenshot shows a web application window titled "Contad v. 1.0". Below the title, it says "Reportes de rentabilidad". There is a label "Nombre del producto:" followed by a text input field containing the placeholder "Escribir texto" and a "Buscar" button. Below this is a large rectangular area with the text "Reporte en PDF" centered inside it.

**Fuente:** Elaboración propia (2015)

La figura 112 detalla la estructura del formulario para generar el reporte de rentabilidad por producto. En ella se puede apreciar que se tiene que buscar un producto para luego generar el reporte.

#### B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 113.** *Interfaz gráfica desarrollada para visualizar el reporte de rentabilidad por producto*



**Fuente:** Elaboración propia (2015)

La figura 113 muestra la interfaz gráfica desarrollada para generar el reporte de rentabilidad por producto. En ella se aprecia que el diseño coincide con lo aprobado por el usuario.

**C. Diseño de la tabla**

Para este reporte se utilizó la misma tabla desarrollada para generar el reporte rentabilidad por producto.

**D. Programación en SQL**

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que selecciona la información sobre el reporte de rentabilidad por producto.

**Figura 114.** *Procedimiento almacenado que muestra datos para el reporte de rentabilidad por producto*

```

-- Consultamos
SELECT
    hsm.`Periodo` AS Periodo,
    tbd.`vchr_disnombre` AS Distrito,
    tbp.`vchr_prdnombre` AS Producto,
    SUM(hsm.`Subtotal`) AS Subtotal,
    SUM(hsm.`CostoVentasVariable`) AS CostoVentasVariable,
    SUM(hsm.`MargenCompras`) AS MargenCompras,
    SUM(hsm.`Comision`) AS Comision,
    SUM(hsm.`MargenContribucion`) AS MargenContribucion,
    (SUM(hsm.`MargenContribucion`) / SUM(hsm.`Subtotal`)) AS RazonMargenContribucion
FROM
    `cp_arzapalo`.`contad_tb_distrito` tbd INNER JOIN `cp_arzapalo`.`hsm_rentabilidad_producto` hsm
    ON tbd.`char_discodigo` = hsm.`Distrito`
    INNER JOIN `cp_arzapalo`.`contad_tb_producto` tbp ON hsm.`Producto` = tbp.`char_prdcodigo`
WHERE
    hsm.Producto = p_Producto
GROUP BY
    Periodo, Distrito, Producto
LIMIT
    6;
    
```

**Fuente:** Elaboración propia (2015)

La figura 114 muestra la construcción del procedimiento almacenado permite generar el reporte de rentabilidad por producto de los últimos 6 meses.

## E. Programación en Java

En esta sección se muestra la programación en Java para generar el reporte de rentabilidad por producto.

**Figura 115.** Procedimiento en Java para generar el reporte de rentabilidad por producto

```
private void reporteRentabilidadDistrito(
    HttpServletRequest request, HttpServletResponse response) {
    // TODO Auto-generated method stub
    try{

        //Capturamos la conexion
        Connection conexion = AccesoDB.getConnection();

        //Establecemos la ruta del reporte
        String fileRepo = "/pe/corparzapalo/contad/app/appwebcontabilidad/reports/rpt_Rentab
        InputStream ioRepo = getClass().getResourceAsStream(fileRepo);

        //Creamos la lista de parametros
        Map<String, Object> parms = new HashMap<String, Object>();
        String periodo = request.getParameter("periodo");
        String distrito_1 = request.getParameter("distrito1");
        String distrito_2 = request.getParameter("distrito2");
        parms.put("p_Periodo", periodo);
        parms.put("p_Distrito_1", distrito_1);
        parms.put("p_Distrito_1", distrito_2);

        //Creamos el reporte
        JasperReport jsRepo = JasperCompileManager.compileReport(ioRepo);
        byte[] bytes = JasperRunManager.runReportToPdf(jsRepo, parms, conexion);

        //Enviando el reporte
        response.setContentType("application/pdf");
        response.setContentLength(bytes.length);
        ServletOutputStream out = response.getOutputStream();
        out.write(bytes, 0, bytes.length);
        out.flush();
    }
}
```

**Fuente:** Elaboración propia (2015)

La figura 115 muestra la programación desarrollada en Java para generar el reporte en PDF de rentabilidad por producto. En ella se aprecia la invocación del reporte.



### 5.10. Planificación del Sprint N° 4

La siguiente tabla muestra la cuarta iteración del sistema desarrollado:

**Tabla 23.** *Planificación de la cuarta iteración*

<b>Número Sprint:</b>	4		
<b>Fecha de inicio:</b>	22/05/2015	<b>Fecha fin:</b>	10/06/2015
<b>Días de duración:</b>	14	<b>Horas hombres estimadas:</b>	109
<b>Fecha de validación:</b>	11/06/2015	<b>Cantidad de historias:</b>	8
<b>N° Historia</b>	<b>Nombre de historia de usuario</b>	<b>Prioridad</b>	<b>Módulo</b>
31	Visualización del sistema en Tablet y computadora	Alta	SEG
30	Reporte de rentabilidad por vendedor	Alta	OPE
2	Búsqueda de proveedores	Media	OPE
6	Registrar abastecimiento	Media	OPE
7	Buscar abastecimiento	Media	OPE
13	Buscar orden de compra	Media	OPE
14	Editar orden de compra	Media	OPE
26	Gráfico costos vs ventas	Media	OPE

Fuente: Elaboración propia (2015)

La tabla 23 muestra la pila del producto del cuarto sprint. En ella se especifican las fechas de inicio y fin del sprint, así como la cantidad de horas/hombres estimadas, la duración en días, la fecha en cual el usuario valida el producto, las historias de usuarios a desarrollarse, el tipo de prioridad y el módulo al cual pertenece cada historia.

### 5.11. Pila del sprint N° 4

En esta sección se muestra la pila del cuarto sprint. A continuación, la siguiente tabla detalla la pila del sprint en desarrollo.

**Tabla 24. Pila del cuarto sprint**

Sprint 4					
<b>Inicio:</b>		22/05/2015			
<b>Duración:</b>		109 horas			
Pila del sprint					
N° de historia	Nombre de la actividad	Tipo	Horas estimadas	Horas Incurridas	Responsable
31	Visualización del sistema en Tablet y computadora	Req. No funcional	1	1	Lorena Flores
30	Reporte de rentabilidad por vendedor	Req. funcional	30	35	Sebastián Aguilar
2	Búsqueda de proveedores	Req. funcional	12	15	Sebastián Aguilar
6	Registrar abastecimiento	Req. funcional	12	12	Lorena Flores
7	Buscar abastecimiento	Req. funcional	12	12	Lorena Flores
13	Buscar orden de compra	Req. funcional	12	10	Lorena Flores
14	Editar orden de compra	Req. funcional	12	12	Lorena Flores
26	Gráfico costos vs ventas	Req. funcional	12	12	Lorena Flores
Total:			109	109	

Fuente: Elaboración propia (2015)

La tabla 24 muestra la pila del cuarto sprint. En ella se puede observar la relación de las historias con el tipo de requisito del sistema. También se observa las horas planificadas e incurridas y el responsable de la elaboración de cada historia.

### 5.12. Incremento del Sprint N° 4

En esta sección se muestran las partes o subsistemas entregados en la segunda iteración completamente operativos.

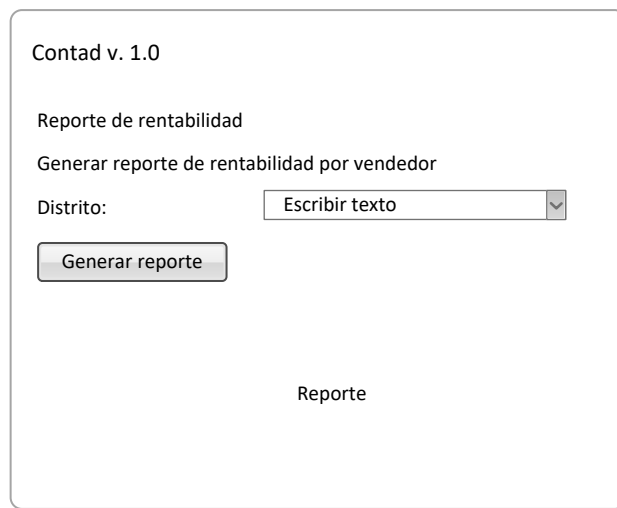
### 5.12.1. Reporte de rentabilidad por vendedor

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 116.** *Interfaz gráfica aprobada para generar el reporte de rentabilidad por vendedor*



**Fuente:** Elaboración propia (2015)

La figura 116 detalla la estructura del formulario para generar el reporte de rentabilidad por vendedor. En ella se puede apreciar que se tiene que seleccionar un distrito para luego generar el reporte.

#### B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 117.** *Interfaz gráfica desarrollada para visualizar el reporte de rentabilidad por vendedor*



**Fuente:** Elaboración propia (2015)

La figura 117 muestra la interfaz gráfica desarrollada para generar el reporte de rentabilidad por vendedor. En ella se aprecia que el diseño coincide con lo aprobado por el usuario.

**C. Diseño de la tabla**

Para este reporte se utilizó la misma tabla desarrollada para generar el reporte rentabilidad por producto.

**D. Programación en SQL**

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que selecciona la información sobre el reporte de rentabilidad por producto.

**Figura 118.** *Procedimiento almacenado que muestra datos para el reporte de rentabilidad por vendedor*

```

-- Consultamos
SELECT
    hsm.`Periodo` AS Periodo,
    tbd.`vchr_disnombre` AS Distrito,
    CONCAT(tbe.`vchar_emponombre`, ' ', tbe.`vchr_empapellido`) AS Vendedor,
    SUM(hsm.`Subtotal`) AS Subtotal,
    SUM(hsm.`CostoVentasVariable`) AS CostoVentasVariable,
    SUM(hsm.`MargenCompras`) AS MargenCompras,
    SUM(hsm.`Comision`) AS Comision,
    SUM(hsm.`MargenContribucion`) AS MargenContribucion,
    (SUM(hsm.`MargenContribucion`) / SUM(hsm.`Subtotal`)) AS RazonMargenContribucion
FROM
    `cp_arzapalo`.`contad_tb_distrito` tbd INNER JOIN `cp_arzapalo`.`hsm_rentabilidad_producto` hsm
    ON tbd.`char_discodigo` = hsm.`Distrito`
    INNER JOIN `cp_arzapalo`.`contad_tb_empleado` tbe ON hsm.`Empleado` = tbe.`char_empcodigo`
WHERE
    hsm.Distrito = p_Distrito
GROUP BY
    Periodo, Distrito, Vendedor
LIMIT
    6;
    
```

**Fuente:** Elaboración propia (2015)

La figura 118 muestra la construcción del procedimiento almacenado permite generar el reporte de rentabilidad por vendedor de los últimos 6 meses.

## E. Programación en Java

En esta sección se muestra la programación en Java para generar el reporte de rentabilidad por vendedor.

**Figura 119.** Procedimiento en Java para generar el reporte de rentabilidad por vendedor

```
private void reporteRentabilidadVendedor(HttpServletRequest request,
    HttpServletResponse response) {
    // TODO Auto-generated method stub
    try{

        //Capturamos la conexion
        Connection conexion = AccesoDB.getConnection();

        //Establecemos la ruta del reporte
        String fileRepo = "/pe/corparzapalo/contad/app/appwebcontabilidad/reports/rpt_Rentab
        InputStream ioRepo = getClass().getResourceAsStream(fileRepo);

        //Creamos la lista de parametros
        Map<String, Object> parms = new HashMap<String, Object>();
        String codigoDistrito = request.getParameter("distrito");
        parms.put("p_Distrito", codigoDistrito);

        //Creamos el reporte
        JasperReport jsRepo = JasperCompileManager.compileReport(ioRepo);
        byte[] bytes = JasperRunManager.runReportToPdf(jsRepo, parms, conexion);

        //Enviando el reporte
        response.setContentType("application/pdf");
        response.setContentLength(bytes.length);
        ServletOutputStream out = response.getOutputStream();
        out.write(bytes, 0, bytes.length);
        out.flush();
        out.close();
    }
}
```

**Fuente:** Elaboración propia (2015)

La figura 119 muestra la programación desarrollada en Java para generar el reporte en PDF de rentabilidad por producto. En ella se aprecia la invocación del reporte.

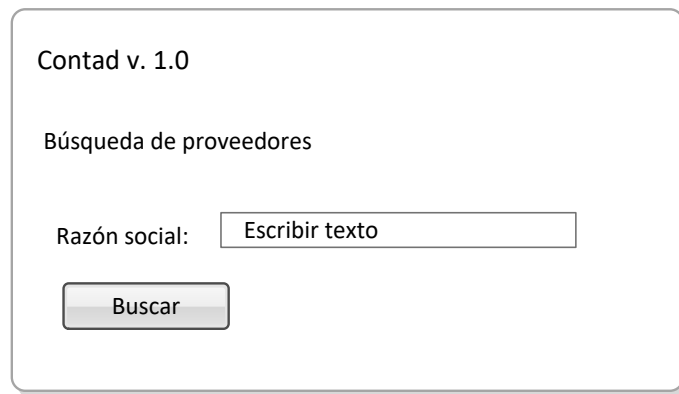
### 5.12.2. Búsqueda de proveedores

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 120.** *Interfaz gráfica aprobada para la búsqueda de proveedores*



Contad v. 1.0

Búsqueda de proveedores

Razón social:

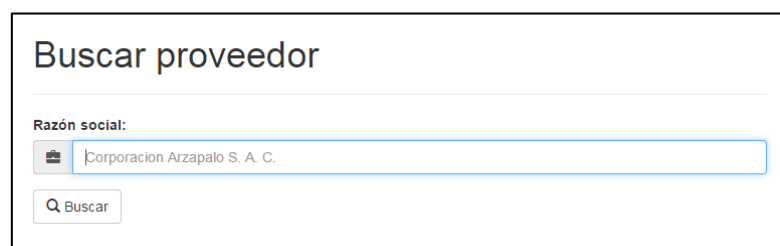
**Fuente:** Elaboración propia (2015)

La figura 120 detalla la estructura del formulario para la búsqueda de proveedores.

## B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 121.** *Interfaz gráfica desarrollada para la búsqueda de proveedores*



Buscar proveedor

Razón social:

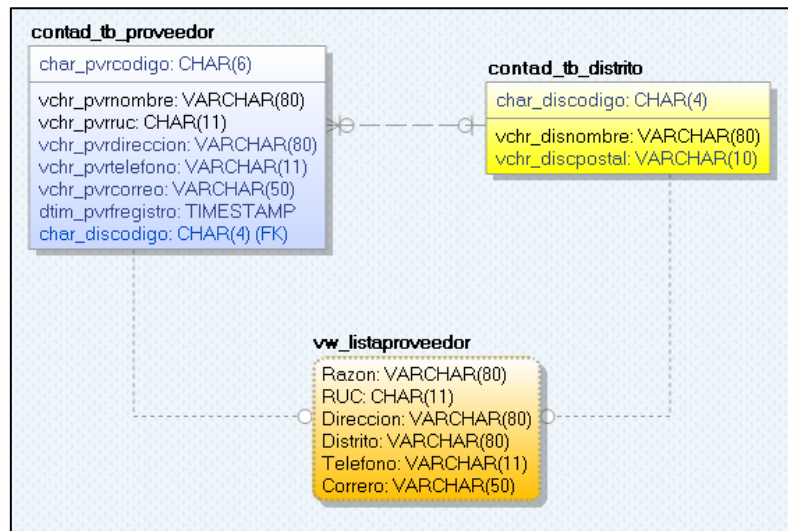
**Fuente:** Elaboración propia (2015)

La figura 121 muestra la interfaz gráfica desarrollada para realizar la búsqueda de proveedores. En ella se observa que se tiene que ingresar la razón social del proveedor.

## C. Diseño de la vista

En esta parte se muestra el diseño de la vista donde se extraen los datos para la búsqueda del proveedor.

**Figura 122. Diseño de la vista lista proveedor**



**Fuente:** Elaboración propia (2015)

En la figura 122 se puede apreciar el diseño de la vista que contiene todos los campos necesarios para realizar la búsqueda del proveedor. También se observa las relaciones de la vista con las tablas de la base de datos.

#### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que permite la búsqueda del proveedor por razón social.

**Figura 123. Procedimiento almacenado que permite la búsqueda del proveedor**

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_getSupplierByName`(p_NombreProveedor VARCHAR(80))
COMMENT 'Procedimiento para extraer datos de un proveedor por nombre'
BEGIN
    -- Seleccionamos los proveedores que cumplan con la condicion
    SELECT Razon, RUC, Direccion, Distrito, Telefono, Correo
    FROM `vw_listaproveedor`
    WHERE Razon LIKE p_NombreProveedor;
END$$
```

**Fuente:** Elaboración propia (2015)

La figura 123 muestra la construcción del procedimiento almacenado que permite realizar la búsqueda del proveedor por su razón social. Como se puede apreciar, no es una búsqueda exacta.

## E. Programación en Java

En esta sección se muestra la programación en Java para buscar un proveedor específico.

**Figura 124.** *Función en Java para la búsqueda de proveedores*

```
public List<TDbProveedor> getByNombreProveedor(Connection cn, String codigo)
    throws SQLException {

    //Creamos una variable de tipo lista para almacenar todos los proveedores
    List<TDbProveedor> lista = new ArrayList<TDbProveedor>();

    //Preparamos la sentencia SQL
    String query = "{ call sp_getSupplierByName(?) }";

    //Instanciamos un objeto del tipo CallableStatement
    CallableStatement stm = null;

    //Instanciamos un objeto del tipo ResultSet
    ResultSet rs = null;

    try{

        //Conectamos el objeto stm con la sentencia sql
        stm = (CallableStatement) cn.prepareCall(query);

        //mandamos los atributos
        stm.setString("p_NombreProveedor", codigo);

        //Ejecutamos el procedimiento
        rs = stm.executeQuery();
    }
}
```

**Fuente:** Elaboración propia (2015)

La figura 124 muestra la programación desarrollada en Java que permite la búsqueda de proveedores desde la aplicación web. En ella se observa la invocación del procedimiento almacenado.

### 5.12.3. Registrar abastecimiento

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.



**Figura 125.** *Interfaz gráfica aprobada para registrar el abastecimiento*

The screenshot shows a web form titled "Registrar abastecimiento". It contains the following fields:
 

- Código del producto:** A text input field with the placeholder "Escribir texto".
- Proveedor:** A dropdown menu with the placeholder "Texto".
- Cantidad mínima a pedir:** A text input field with the placeholder "Escribir texto".
- Precio por mayor:** A text input field with the placeholder "Texto".
- Precio por menor:** A text input field with the placeholder "Texto".
- Registrar:** A button at the bottom of the form.

**Fuente:** Elaboración propia (2015)

La figura 125 detalla la estructura del formulario para registrar el abastecimiento de los productos. En ella se observa los campos implicados en la operación.

**B. Interfaz gráfica desarrollada**

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 126.** *Interfaz gráfica desarrollada para registrar el abastecimiento*

The screenshot shows the developed version of the "Registrar abastecimiento" form. The fields are populated with the following data:
 

- Código del producto:** 674365
- Proveedor:** Seleccione
- Cantidad mínima a pedir:** # 0
- Precio por mayor:** S./ 0
- Precio por menor:** S./ 0
- Insertar:** A button at the bottom of the form.

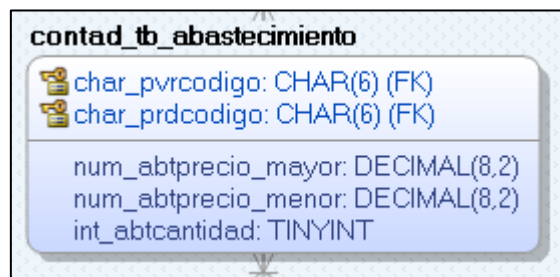
**Fuente:** Elaboración propia (2015)

La figura 126 muestra la interfaz gráfica desarrollada para registrar el abastecimiento del producto. En ella se aprecia que el diseño coincide con lo aprobado por el usuario.

### C. Diseño de la tabla

Para esta tarea, se diseñó la siguiente tabla:

**Figura 127.** *Diseño de la tabla abastecimiento*



**Fuente:** Elaboración propia (2015)

La figura 127 muestra el diseño de la tabla “abastecimiento”. En ella se puede observar los campos implicados con su respectivo tipo de dato.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que permite el registro del abastecimiento.

**Figura 128.** *Procedimiento almacenado que registra el abastecimiento de un producto.*

```

CREATE DEFINER='root'@'localhost' PROCEDURE `sp_insertProvision`(p_Proveedor CHAR(11), p_Producto CHAR(6), p_Cantidad TINYINT,
COMMENT 'Procedimiento para insertar datos de precio de abastecimiento'
p_PrecioMayor DECIMAL(8,2), p_PrecioMenor DECIMAL(8,2))
BEGIN
-- Declaramos una variable
DECLARE v_CodigoProveedor CHAR(6);
-- Iniciamos la transaccion
START TRANSACTION;
-- Capturamos el codigo del proveedor a traves del RUC
SELECT 'char_pvrcodigo' INTO v_CodigoProveedor FROM 'contad_tb_proveedor' WHERE 'vchr_pvruc' = p_Proveedor;
-- Insertamos los valores
INSERT INTO 'contad_tb_abastecimiento' ('char_pvrcodigo', 'char_prdcodigo', 'int_abtcantidad', 'num_abtprecio_mayor',
'num_abtprecio_menor')
VALUES (v_CodigoProveedor, p_Producto, p_Cantidad, p_PrecioMayor, p_PrecioMenor);
-- confirmamos
COMMIT;
END$$
    
```

**Fuente:** Elaboración propia (2015)

La figura 128 muestra la construcción del procedimiento almacenado que permita el registro del abastecimiento de un producto en particular en la base de datos.

## E. Programación en Java

En esta sección se muestra la programación en Java para registrar el abastecimiento de un producto.

**Figura 129.** *Función en Java para registrar el abastecimiento de un producto*

```
public boolean insertAll(Connection cn, TDbAbastecimiento tAbastecimiento)
    throws SQLException {

    //Preparamos la sentencia SQL
    String query = "{ call sp_insertProvision(?, ?, ?, ?, ?) }";

    //Instanciamos un objeto del tipo CallableStatement
    CallableStatement stm = null;

    try{

        //Conectamos el objeto stm con la sentencia sql
        stm = (CallableStatement) cn.prepareCall(query);

        //mandamos los atributos
        stm.setString("p_Proveedor", tAbastecimiento.getProveedor());
        stm.setString("p_Producto", tAbastecimiento.getProducto());
        stm.setInt("p_Cantidad", tAbastecimiento.getCantidad());
        stm.setDouble("p_PrecioMayor", tAbastecimiento.getMayor());
        stm.setDouble("p_PrecioMenor", tAbastecimiento.getMenor());

        //Ejecutamos la sentencia
        stm.execute();
    }
```

**Fuente:** Elaboración propia (2015)

La figura 129 muestra la programación desarrollada en Java que permite el registro de un abastecimiento para un producto en particular desde la aplicación web. En ella se puede observar la invocación del procedimiento almacenado.

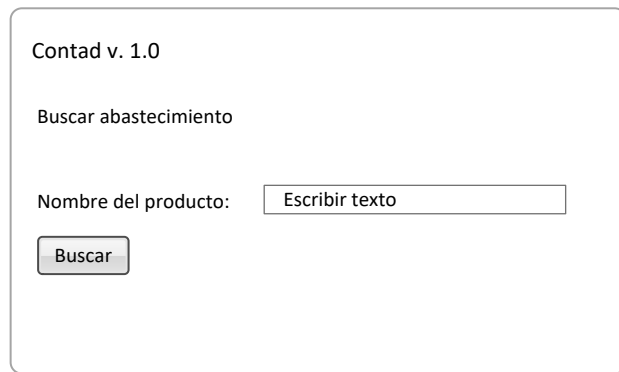
### 5.12.4. Buscar abastecimiento

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

## A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 130.** *Interfaz gráfica aprobada para buscar abastecimiento*



The screenshot shows a web form titled "Contad v. 1.0". Below the title is the heading "Buscar abastecimiento". There is a label "Nombre del producto:" followed by a text input field containing the placeholder text "Escribir texto". Below the input field is a button labeled "Buscar".

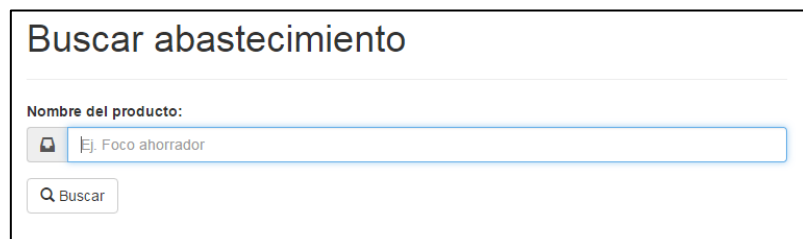
**Fuente:** Elaboración propia (2015)

La figura 130 muestra la estructura del formulario para buscar un abastecimiento. En ella se puede apreciar que la búsqueda se realiza mediante el nombre del producto.

## B. Interfaz gráfica desarrollado

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 131.** *Interfaz gráfica desarrollada para buscar un abastecimiento*



The screenshot shows a web form titled "Buscar abastecimiento". Below the title is a label "Nombre del producto:" followed by a text input field containing the example text "Ej. Foco ahorrador". Below the input field is a button labeled "Buscar" with a magnifying glass icon.

**Fuente:** Elaboración propia (2015)

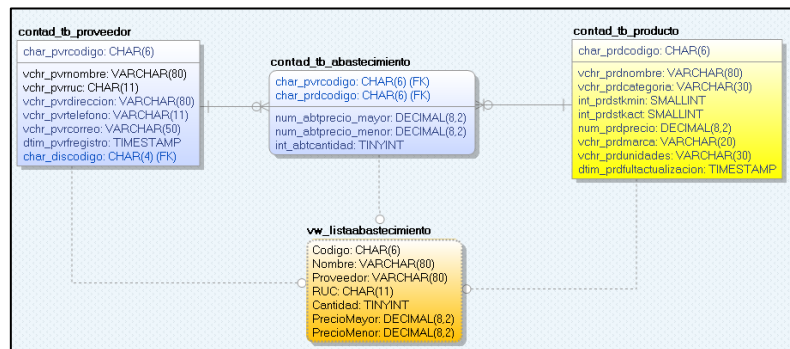
La figura 131 muestra la interfaz gráfica desarrollada para la búsqueda del abastecimiento. En ella se aprecia que el

diseño coincide con lo aprobado por el usuario y la búsqueda se realiza mediante el nombre del producto.

### C. Diseño de la tabla

Para la búsqueda de abastecimiento, se construyó una vista con la siguiente estructura:

**Figura 132.** *Diseño de la vista para la búsqueda de abastecimiento*



**Fuente:** Elaboración propia (2015)

La figura 132 muestra la estructura de la vista que permite realizar la búsqueda de un determinado abastecimiento. En ella se puede apreciar los campos que posee y su relación con las tablas de la base de datos.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que selecciona la información sobre el abastecimiento de un determinado producto.

**Figura 133.** Procedimiento almacenado que selecciona datos de abastecimiento de un producto en particular

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_getProvisionByProduct`(p_Producto CHAR(6))
COMMENT 'Procedimiento para extraer datos de un abastecimiento por producto'
BEGIN
-- seleccionamos
SELECT *
FROM `vw_listaabastecimiento`
WHERE `Codigo` = p_Producto;
END$$
DELIMITER ;
```

**Fuente:** Elaboración propia (2015)

La figura 133 muestra la construcción del procedimiento almacenado permite seleccionar los datos de la vista construida en el paso anterior para la búsqueda de un abastecimiento específico.

## E. Programación en Java

En esta sección se muestra la programación en Java para generar el reporte de rentabilidad por producto.

**Figura 134.** Función en Java para la búsqueda de abastecimiento

```
public List<TDbAbastecimiento> getByProducto(Connection conexion,
String codigo) throws SQLException {
//Creamos una variable de tipo lista para almacenar todos los abastecimiento
List<TDbAbastecimiento> lista = new ArrayList<TDbAbastecimiento>();
//Preparamos la sentencia SQL
String query = "{ call sp_getProvisionByProduct(?) }";
//Instanciamos un objeto del tipo CallableStatement
CallableStatement stm = null;
//Instanciamos un objeto del tipo ResultSet
ResultSet rs = null;
try{
//Conectamos el objeto stm con la sentencia sql
stm = (CallableStatement) conexion.prepareCall(query);
//mandamos los atributos
stm.setString("p_Producto", codigo);
//Ejecutamos el procedimiento
rs = stm.executeQuery();
}
```

**Fuente:** Elaboración propia (2015)

La figura 134 muestra la programación desarrollada en Java que permite la búsqueda de un abastecimiento mediante el código del producto desde la aplicación web.

En ella se puede observar la invocación del procedimiento almacenado.

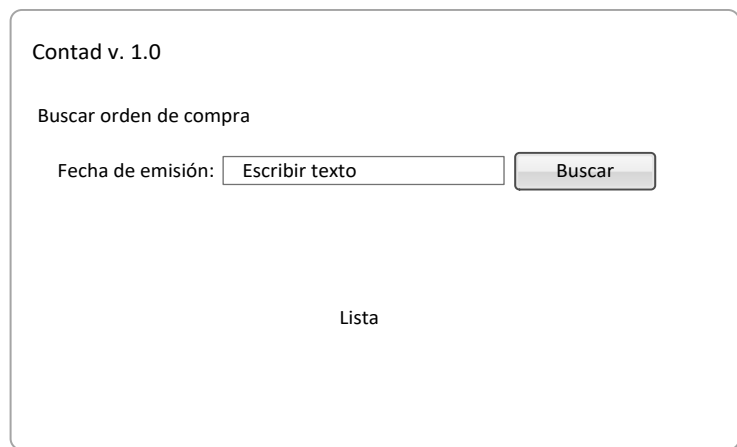
### 5.12.5. Buscar orden de compra

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 135.** *Interfaz gráfica aprobada para la búsqueda de una orden de compra*



The screenshot shows a web interface titled "Contad v. 1.0". Below the title, it says "Buscar orden de compra". There is a form with a label "Fecha de emisión:" followed by a text input field containing the placeholder "Escribir texto" and a "Buscar" button. Below the form, the word "Lista" is centered.

**Fuente:** Elaboración propia (2015)

La figura 135 muestra la estructura del formulario para buscar una orden de compra. En ella se puede apreciar que la búsqueda se realiza por la fecha de emisión.

#### B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 136.** Interfaz gráfica desarrollada para la búsqueda de una orden de compra



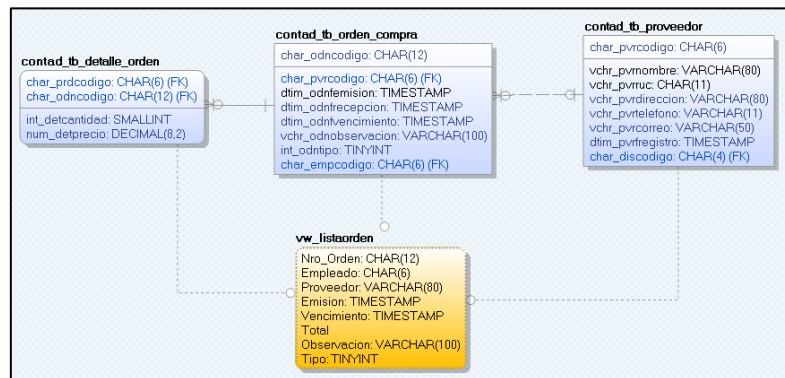
**Fuente:** Elaboración propia (2015)

La figura 136 muestra la interfaz gráfica desarrollada para la búsqueda de órdenes de compra. En ella se aprecia que el diseño coincide con lo aprobado por el usuario.

**C. Diseño de la vista**

La siguiente figura muestra la estructura de la vista utilizada para esta historia de usuario.

**Figura 137.** Diseño de la vista lista orden



**Fuente:** Elaboración propia (2015)

La figura 137 muestra el diseño de la vista y su relación con las tablas. En ella se puede observar los campos involucrados para la construcción de la vista.



## D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que selecciona los datos de una lista de órdenes de compra mediante la fecha de emisión.

**Figura 138.** Procedimiento almacenado que muestra los datos de una orden de compra en particular

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_getOrderByDate`(p_Fecha DATETIME)
COMMENT 'Procedimiento para extraer todas las OC por una determinada fecha'
BEGIN
    -- Seleccionamos las OC
    SELECT
        Nro_Orden,
        Empleado,
        Proveedor,
        DATE_FORMAT(Emission, '%d-%b-%Y') AS Emission,
        DATE_FORMAT(Vencimiento, '%d-%b-%Y') AS Vencimiento,
        Total,
        Observacion,
        Estado
    FROM vw_listaorden
    WHERE Emission = p_Fecha;
END$$
```

**Fuente:** Elaboración propia (2015)

La figura 138 muestra la construcción del procedimiento almacenado permite seleccionar los datos correspondientes de todas las órdenes de compras en donde coincidan con la fecha de emisión proporcionada.

## E. Programación en Java

En esta sección se muestra la programación en Java para la búsqueda de órdenes de compra.

**Figura 139.** *Función en Java para la búsqueda de órdenes de compra por fecha de emisión.*

```
public List<TDbOrdenCompra> getByFecha(Connection cn, String mes)
    throws SQLException {

    //Instanciamos un objeto
    List<TDbOrdenCompra> listaOrden = new ArrayList<TDbOrdenCompra>();

    //Preparamos la sentencia SQL
    String query = "{ call sp_getOrderByDate(?) }";

    //Instanciamos un objeto del tipo CallableStatement
    CallableStatement stm = null;

    //Declaramos una variable del tipo resultset
    ResultSet rs = null;

    try{

        //Conectamos el objeto stm con la sentencia sql
        stm = (CallableStatement) cn.prepareCall(query);

        //Mandamos los argumentos
        stm.setString("p_fecha", mes);

        //Ejecutamos la sentencia
        rs = stm.executeQuery();
    }
}
```

**Fuente:** Elaboración propia (2015)

La figura 139 muestra la programación desarrollada en Java que permite la búsqueda de órdenes de compra por fecha de emisión. En ella se puede observar la invocación del procedimiento almacenado.

#### 5.12.6. Editar orden de compra

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

##### A. Interfaz gráfica aprobado

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 133.** *Interfaz gráfica aprobada para editar una orden de compra*

**Fuente:** Elaboración propia (2015)

La figura 133 muestra la estructura del formulario para editar una orden de compra. En ella se puede apreciar los campos que se van a editar.

### B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 141.** *Interfaz gráfica desarrollada para la búsqueda de una orden de compra*

**Fuente:** Elaboración propia (2015)

La figura 141 muestra la interfaz gráfica desarrollada para la edición de órdenes de compra. En ella se aprecia que el diseño coincide con lo aprobado por el usuario.

### C. Diseño de la tabla

Para la presente historia, no se requirió la construcción de una tabla o una vista.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que permite la edición de órdenes de compra.

**Figura 142.** *Procedimiento almacenado que actualiza datos de una orden de compra*

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_updateOrder`(p_NumOrden CHAR(6), p_Observacion VARCHAR(100), p_Estado TINYINT)
COMMENT 'Procedimiento para actualizar datos de una OC'
BEGIN
    -- Actualizamo
    UPDATE `contad_tb_orden_compra`
    SET `vchr_odnobservacion` = p_Observacion, `int_odntipo` = p_Estado
    WHERE `char_odncodigo` = p_NumOrden;
END##
```

**Fuente:** Elaboración propia (2015)

La figura 142 muestra la construcción del procedimiento almacenado permite la edición órdenes de compras.

### E. Programación en Java

En esta sección se muestra la programación en Java para la edición de órdenes de compra.

**Figura 143.** Función en Java para la edición de órdenes de compra.

```
public boolean updateAll(Connection cn, TDbOrdenCompra tOrden)
    throws SQLException {

    //Preparamos la sentencia SQL
    String query = "{ call sp_updateOrder(?, ?, ?) }";

    //Instanciamos un objeto del tipo CallableStatement
    CallableStatement stm = null;

    try{

        //Conectamos el objeto stm con la sentencia sql
        stm = (CallableStatement) cn.prepareCall(query);

        //mandamos los atributos
        stm.setString("p_NumOrden", tOrden.getCodigo());
        stm.setString("p_Observacion", tOrden.getObservacion());
        stm.setString("p_Estado", tOrden.getTipo());

        //ejecutamos la sentencia
        stm.execute();

        //retornamos un valor verdadero
        return true;
    }
}
```

**Fuente:** Elaboración propia (2015)

La figura 143 muestra la programación desarrollada en Java que permite la edición de órdenes de compra. En ella se puede observar la invocación del procedimiento almacenado.

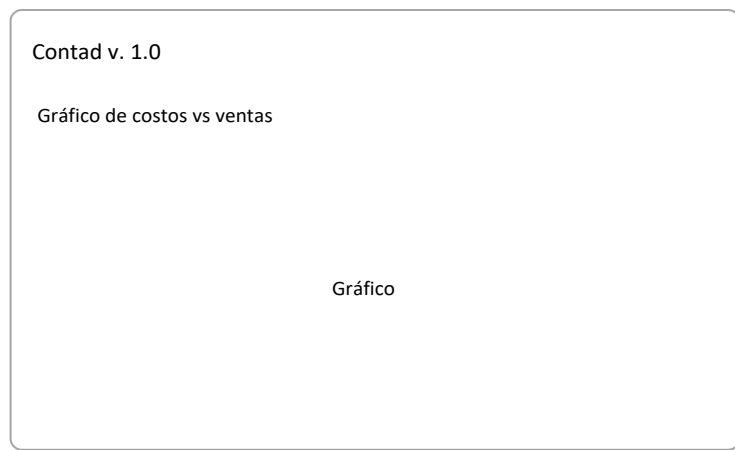
### 5.12.7. Gráfico de costos vs ventas

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 144.** *Interfaz gráfica aprobada el gráfico de costos vs ventas*



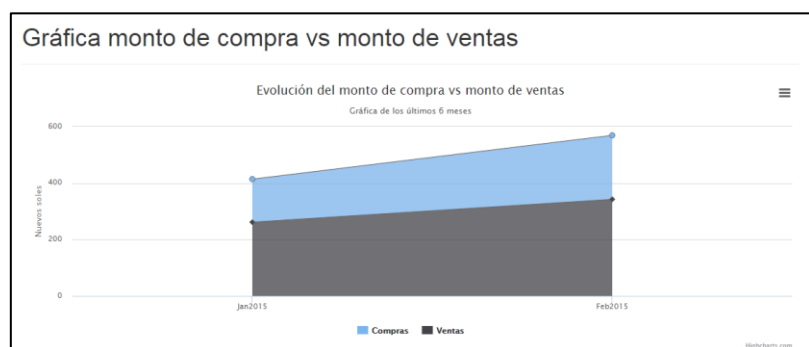
**Fuente:** Elaboración propia (2015)

La figura 144 muestra la estructura de la página en donde se visualizará el gráfico solicitado.

**B. Interfaz gráfica desarrollada**

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 145.** *Interfaz gráfica desarrollada para el gráfico de costos vs ventas*



**Fuente:** Elaboración propia (2015)

La figura 145 muestra la interfaz gráfica desarrollada para el gráfico de costos vs ventas. En ella se aprecia que el diseño coincide con lo aprobado por el usuario.

### C. Diseño de la tabla

Para la presente historia, no se requirió la construcción de una tabla o una vista.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que permite la generación del gráfico de costos vs ventas de los últimos 6 meses.

**Figura 146.** *Procedimiento almacenado que permite generar el gráfico de costos vs ventas*

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_getCostoVentas`()
COMMENT 'Procedimiento que devuelve la evolución de los costos y las ventas'
BEGIN
  -- Seleccionamos
  SELECT a.Mes, SUM(a.CostoTotalMensual) AS CostoTotalMensual, SUM(a.VentaTotalMensual) AS VentaTotalMensual FROM (
    (SELECT Mes, SUM(CostoTotal) AS CostoTotalMensual, 0 AS VentaTotalMensual FROM vw_listaacostosproductos GROUP BY Mes)
    UNION
    (SELECT Mes, 0 AS CostoTotalMensual, SUM(VentaTotal) AS VentaTotalMensual FROM vw_listaventaproduetos GROUP BY Mes)
  ) a
  GROUP BY a.Mes
  ORDER BY 1 DESC LIMIT 6;
END
```

**Fuente:** Elaboración propia (2015)

La figura 146 muestra la construcción del procedimiento almacenado permite generar el gráfico de costos – ventas de los últimos 6 meses.

### E. Programación en Java

En esta sección se muestra la programación en Java para generar el gráfico costos vs ventas.

**Figura 147.** *Función en Java para generar el gráfico costos vs ventas.*

```

public List<TDbAbastecimiento> getCostoVentas(Connection conexion)
    throws SQLException {

    //Creamos una variable que almacene los datos devueltos
    List<TDbAbastecimiento> listaCostoVentas = new ArrayList<TDbAbastecimiento>();

    //Preparamos la sentencia SQL
    String query = "{ call sp_getCostoVentas()}";

    //Instanciamos un objeto del tipo CallableStatement
    CallableStatement stm = null;

    try{

        //Conectamos el objeto stm con la sentencia sql
        stm = (CallableStatement) conexion.prepareCall(query);

        //Ejecutamos la sentencia
        ResultSet rs = stm.executeQuery(query);

        //Capturamos todos los valores devueltos
        while(rs.next()){

            TDbAbastecimiento tDbAbastecimiento = new TDbAbastecimiento(
                "",
                rs.getString("Mes"),
                rs.getDouble("CostoTotalMensual"),
                rs.getDouble("VentaTotalMensual"),
            );
        }
    }
}
    
```

**Fuente:** Elaboración propia (2015)

La figura 147 muestra la programación desarrollada en Java que permite generar el gráfico costos vs ventas. En ella se puede observar la invocación del procedimiento almacenado.

### 5.13. Planificación del Sprint N° 5

La siguiente tabla muestra la quinta iteración del sistema desarrollado:

**Tabla 25.** *Planificación de la quinta iteración*

<b>Número Sprint:</b>	5		
<b>Fecha de inicio:</b>	12/06/2015	<b>Fecha fin:</b>	22/06/2015
<b>Días de duración:</b>	7	<b>Horas hombres estimadas:</b>	49
<b>Fecha de validación:</b>	23/06/2015	<b>Cantidad de historias:</b>	5
<b>N° Historia</b>	<b>Nombre de historia de usuario</b>	<b>Prioridad</b>	<b>Módulo</b>
31	Visualización del sistema en Tablet y computadora	Alta	SEG



N° Historia	Nombre de historia de usuario	Prioridad	Módulo
3	Actualizar datos del proveedor	Baja	OPE
4	Eliminar proveedor	Baja	OPE
8	Editar abastecimiento	Baja	OPE
9	Eliminar abastecimiento	Baja	OPE

Fuente: Elaboración propia (2015)

La tabla 25 muestra la planificación del quinto sprint. En ella se especifican las fechas de inicio y fin del sprint, así como la cantidad de horas/hombres estimadas, la duración en días, la fecha en cual el usuario valida el producto, las historias de usuarios a desarrollarse, el tipo de prioridad y el módulo al cual pertenece cada historia.

#### 5.14. Pila del sprint N° 5

En esta sección se muestra la pila del quinto sprint. A continuación, la siguiente tabla detalla la pila del sprint en desarrollo.

**Tabla 26.** Pila del quinto sprint

Sprint 5					
Inicio:		12/06/2015			
Duración:		49 horas			
Pila del sprint					
N° de historia	Nombre de la actividad	Tipo	Horas estimadas	Horas Incurridas	Responsable
31	Visualización del sistema en Tablet y computadora	Req. No funcional	1	1	Lorena Flores
3	Actualizar datos del proveedor	Req. funcional	12	13	Sebastián Aguilar
4	Eliminar proveedor	Req. funcional	12	10	Lorena Flores
8	Editar abastecimiento	Req. funcional	12	12	Lorena Flores
9	Eliminar abastecimiento	Req. funcional	12	12	Lorena Flores
Total:			49	48	

Fuente: Elaboración propia (2015)

La tabla 26 muestra la pila del primer sprint. En ella se puede observar la relación de las historias con el tipo de requisito del sistema. También se observa las horas planificadas e incurridas y el responsable de la elaboración de cada historia.

## 5.15. Incremento del Sprint N° 5

En esta sección se muestran las partes o subsistemas entregados en la segunda iteración completamente operativos.

### 5.15.1. Actualizar datos del proveedor

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 148.** *Interfaz gráfica aprobada para editar el proveedor*

Contad v. 1.0

Editar proveedor

Razón social:  Distrito:

Ruc:  Teléfono:

Dirección:  Correo:

**Fuente:** Elaboración propia (2015)

La figura 148 detalla la estructura del formulario para actualizar los datos del proveedor. En ella se observa que se pueden editar todos los campos descritos.

## B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 149.** *Interfaz gráfica desarrollada para la edición del proveedor*

**Fuente:** Elaboración propia (2015)

La figura 149 muestra la interfaz gráfica desarrollada para editar el proveedor. En ella se aprecia que el diseño coincide con lo aprobado por el usuario.

## C. Diseño de la tabla

Para esta historia, no se requirió la construcción de una tabla o vista.

## D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que permite la edición del proveedor.

**Figura 150.** *Procedimiento almacenado que actualiza los datos de un proveedor*

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_updateSupplier`(p_RazonSocial VARCHAR(80), p_Ruc CHAR(11), p_Direccion VARCHAR(80),
p_Distrito CHAR(4), p_Telefono VARCHAR(11), p_Correo VARCHAR(50))
COMMENT 'Procedimiento para actualizar un proveedor mediante ruc'
BEGIN
-- actualizamos
UPDATE `contad_tb_proveedor`
SET `vchr_pvrnombre` = p_RazonSocial, `vchr_pvrdireccion` = p_Direccion, `vchr_pvrtelefono` = p_Telefono,
`vchr_pvrcorreo` = p_Correo, `char_discodigo` = p_Distrito
WHERE `vchr_pvruc` = p_Ruc;
END##
```

**Fuente:** Elaboración propia (2015)

La figura 150 muestra la construcción del procedimiento almacenado que permite la edición de un determinado proveedor.

## E. Programación en Java

En esta sección se muestra la programación en Java para actualizar los datos de un proveedor en particular.

**Figura 151.** *Función en Java para actualizar los datos de un proveedor*

```
public boolean updateAll(Connection cn, TDbProveedor tProveedor)
    throws SQLException {

    //Preparamos la sentencia SQL
    String query = "{ call sp_updateSupplier(?, ?, ?, ?, ?, ?) }";

    //Instanciamos un objeto del tipo CallableStatement
    CallableStatement stm = null;

    try{

        //Conectamos el objeto stm con la sentencia sql
        stm = (CallableStatement) cn.prepareCall(query);

        //mandamos los atributos
        stm.setString("p_RazonSocial", tProveedor.getNombre());
        stm.setString("p_Ruc", tProveedor.getRuc());
        stm.setString("p_Direccion", tProveedor.getDireccion());
        stm.setString("p_Distrito", tProveedor.getDistrito());
        stm.setString("p_Telefono", tProveedor.getTelefono());
        stm.setString("p_Correo", tProveedor.getCorreo());

        //ejecutamos la sentencia
        stm.execute();
    }
```

**Fuente:** Elaboración propia (2015)

La figura 151 muestra la programación desarrollada en Java que permite la edición del proveedor desde la aplicación web desarrollada. En ella se observa la invocación del procedimiento almacenado.

### 5.15.2. Eliminar proveedor

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

## A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 152.** *Interfaz gráfica aprobada para eliminar un proveedor*

The screenshot shows a web form titled "Contad v. 1.0" with the subtitle "Eliminar proveedor". The form contains six text input fields arranged in a 2x3 grid. The labels and their corresponding input fields are: "Razón social:" with "Escribir texto", "Distrito:" with "Escribir texto", "Ruc:" with "Escribir texto", "Teléfono:" with "Escribir texto", "Dirección:" with "Escribir texto", and "Correo:" with "Escribir texto". Below the grid is a single "Eliminar" button.

**Fuente:** Elaboración propia (2015)

La figura 152 detalla la estructura del formulario para eliminar un proveedor. En ella se puede apreciar que se tienen que visualizar los datos antes de proceder a eliminar el proveedor.

## B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 153.** *Interfaz gráfica desarrollada para eliminar un proveedor*

The screenshot shows a web form titled "Eliminar proveedor". It features six input fields in a 2x3 grid. The labels and their corresponding input fields are: "Razón social:" with a text input field, "Distrito:" with a dropdown menu showing "Seleccione", "RUC:" with a text input field, "Telefono:" with a text input field, "Direccion:" with a text input field, and "Correo:" with a text input field. Below the grid is a button labeled "x Eliminar".

**Fuente:** Elaboración propia (2015)

La figura 153 muestra la interfaz gráfica desarrollada eliminar un proveedor. En ella se aprecia que el diseño coincide con lo aprobado por el usuario.

### C. Diseño de la tabla

Para esta historia, no se requirió la construcción de una tabla o vista.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que permite la eliminación del proveedor.

**Figura 154.** *Procedimiento almacenado que elimina un proveedor*

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_deleteSupplier`(p_Ruc CHAR(11))
  COMMENT 'Procedimiento para eliminar un proveedor mediante su ruc'
]BEGIN
  -- eliminamos el proveedor por numero de ruc
  DELETE FROM `contad_tb_proveedor` WHERE `vchr_pvrruc` = p_Ruc;
  END$$
DELIMITER ;
```

**Fuente:** Elaboración propia (2015)

La figura 154 muestra la construcción del procedimiento almacenado permite eliminar un proveedor en particular mediante su RUC.

### E. Programación en Java

En esta sección se muestra la programación en Java que permite la eliminación de un proveedor específico.

**Figura 155.** Función en Java para eliminar un proveedor

```
public boolean deleteAll(Connection cn, String ruc)
    throws SQLException {

    //Preparamos la sentencia SQL
    String query = "{ call sp_deleteSupplier(?) }";

    //Instanciamos un objeto del tipo CallableStatement
    CallableStatement stm = null;

    try{

        //Conectamos el objeto stm con la sentencia sql
        stm = (CallableStatement) cn.prepareCall(query);

        //Enviamos los argumentos
        stm.setString("p_Ruc", ruc);

        //Ejecutamos la sentencia
        stm.execute();

        //Retornamos el valor verdadero
        return true;
    }
```

**Fuente:** Elaboración propia (2015)

La figura 155 muestra la programación desarrollada en Java que permita eliminar un proveedor en particular desde la aplicación web desarrollada. En ella se observa la invocación del procedimiento almacenado.

### 5.15.3. Editar abastecimiento

En esta sección se muestra el desarrollo empleado para la historia de usuario en mención.

#### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 156.** *Interfaz gráfica aprobada para editar un abastecimiento*

Contad v. 1.0

Editar abastecimiento para: Producto 1

Producto:  
Escribir texto

Proveedor:  
Escribir texto

Cantidad mínima a pedir:  
Escribir texto

Precio por mayor:  
Escribir texto

Precio por menor:  
Escribir texto

Editar

**Fuente:** Elaboración propia (2015)

La figura 156 muestra la estructura del formulario para editar un abastecimiento. En ella se puede apreciar los campos que solo se podrán editar (no sombreados).

## **B. Interfaz gráfica desarrollada**

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.



**Figura 157.** Interfaz gráfica desarrollada para editar un abastecimiento

Editar abastecimiento para

Producto:

Proveedor:

Cantidad mínima a pedir:  
# 0

Precio por mayor:  
S./ 0.0

Precio por menor:  
S./ 0.0

**Fuente:** Elaboración propia (2015)

La figura 157 muestra la interfaz gráfica desarrollada para la edición del abastecimiento. En ella se aprecia que el diseño coincide con lo aprobado por el usuario.

### C. Diseño de la tabla

Para esta historia, no se requirió la construcción de una tabla o vista.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que permite la actualización de un abastecimiento en particular.

**Figura 158.** Procedimiento almacenado que actualiza el abastecimiento de un producto en particular

```
-- Declaramos variables
DECLARE v_CodigoProducto CHAR(6);
DECLARE v_CodigoProveedor CHAR(6);

-- Capturamos el codigo del producto
SELECT `char_prdcodigo` INTO v_CodigoProducto FROM `contad_tb_producto` WHERE `vchr_prdnombre` = p_Producto;

-- Capturamos el codigo del proveedor
SELECT `char_pvrcodigo` INTO v_CodigoProveedor FROM `contad_tb_proveedor` WHERE `vchr_pvrnombre` = p_Proveedor;

-- Actualizamos la tabla abastecimiento
UPDATE `contad_tb_abastecimiento` SET `int_abtcantidad` = p_Cantidad, `num_abtprecio_mayor` = p_PrecioMayor,
`num_abtprecio_menor` = p_PrecioMenor
WHERE `char_pvrcodigo` = v_CodigoProveedor AND `char_prdcodigo` = v_CodigoProducto;
```

**Fuente:** Elaboración propia (2015)

La figura 158 muestra la construcción del procedimiento almacenado permite la actualización de un abastecimiento específico.

## E. Programación en Java

En esta sección se muestra la programación en Java para actualizar un abastecimiento en particular.

**Figura 159.** Función en Java para la actualización de un abastecimiento

```
public boolean updateAll(Connection cn, TDbAbastecimiento tAbastecimiento)
    throws SQLException {

    //Preparamos la sentencia SQL
    String query = "{ call sp_updateProvision(?, ?, ?, ?, ?) }";

    //Instanciamos un objeto del tipo CallableStatement
    CallableStatement stm = null;

    try{

        //Conectamos el objeto stm con la sentencia sql
        stm = (CallableStatement) cn.prepareCall(query);

        //mandamos los atributos
        stm.setString("p_Producto", tAbastecimiento.getProducto());
        stm.setString("p_Proveedor", tAbastecimiento.getProveedor());
        stm.setInt("p_Cantidad", tAbastecimiento.getCantidad());
        stm.setDouble("p_PrecioMayor", tAbastecimiento.getMayor());
        stm.setDouble("p_PrecioMenor", tAbastecimiento.getMenor());

        //ejecutamos la sentencia
        stm.execute();

        //retornamos un valor verdadero
        return true;
    }
```

**Fuente:** Elaboración propia (2015)

La figura 159 muestra la programación desarrollada en Java que permite la actualización de un abastecimiento

específico. En ella se puede observar la invocación del procedimiento almacenado.

#### 5.15.4. Eliminar abastecimiento

Se validó que no exista la función de eliminar el abastecimiento desde la aplicación.

##### A. Interfaz gráfica aprobada

En esta sección se muestra la interfaz gráfica aprobada por el usuario.

**Figura 160.** *Interfaz gráfica aprobada para la eliminación de abastecimientos*

Contad v. 1.0

Eliminar abastecimiento para el producto xxxxxxxxxxxx:

Producto:

Proveedor:

Cantidad mínima:

Precio por mayor:

Precio por menor:

**Fuente:** Elaboración propia (2015)

La figura 160 detalla la estructura del formulario para la eliminación de un abastecimiento en particular. En ella se observa que los datos son mostrados al usuario antes de su eliminación, las cuales tampoco se pueden editar.

##### B. Interfaz gráfica desarrollada

En esta sección se muestra la interfaz gráfica desarrollada por el equipo de desarrollo.

**Figura 161.** *Interfaz gráfica desarrollada para la eliminación de abastecimientos*

Eliminar abastecimiento para FOCO

**Producto:**  
FOCO AHORRADOR 21W 27W 3U

**Proveedor:**  
Comercial Bella Sur S.A.C.

**Cantidad mínima a pedir:**  
# 20

**Precio por mayor:**  
S./ 2.81

**Precio por menor:**  
S./ 3.3

**Fuente:** Elaboración propia (2015)

La figura 161 muestra la interfaz gráfica desarrollada para realizar la eliminación del abastecimiento de un producto en particular. En ella se observa que el diseño se asemeja con lo aprobado por el usuario.

### C. Diseño de la tabla

Para este requerimiento, no se construyó una tabla o vista.

### D. Programación en SQL

En esta sección se muestra el desarrollo del procedimiento almacenado en MySQL que permite la eliminación de abastecimientos.

**Figura 162.** Procedimiento almacenado que permite la eliminación de abastecimientos

```

CREATE DEFINER='root'@'localhost' PROCEDURE `sp_deleteProvision`(p_Producto VARCHAR(80), p_Proveedor VARCHAR(80))
COMMENT 'Procedimiento para eliminar un abastecimiento'
BEGIN
-- Declaramos variables
DECLARE v_Producto CHAR(6);
DECLARE v_Proveedor CHAR(6);
-- Capturamos el código del producto
SELECT `char_prdcodigo` INTO v_Producto FROM `contad_tb_producto` WHERE `vchr_prdnombre` = p_Producto;
-- Capturamos el código del proveedor
SELECT `char_pvrcodigo` INTO v_Proveedor FROM `contad_tb_proveedor` WHERE `vchr_pvrnombre` = p_Proveedor;
-- Borramos el registro
DELETE FROM `contad_tb_abastecimiento` WHERE `char_pvrcodigo` = v_Proveedor AND `char_prdcodigo` = v_Producto;
END$$

```

**Fuente:** Elaboración propia (2015)

La figura 162 muestra el desarrollo del procedimiento almacenado que permite la eliminación de un abastecimiento. En ella se puede observar que se necesita ingresar el nombre del producto y el nombre del proveedor e internamente el procedimiento se encarga de buscar el código para su posterior eliminación de la base de datos.

## E. Programación en Java

En esta sección se muestra la programación en Java que permite la eliminación del abastecimiento mediante la aplicación web desarrollada.

**Figura 163.** Función en Java que permite la eliminación de abastecimientos

```


public boolean deleteAll(Connection cn, String proveedor, String producto)
throws SQLException {
//Preparamos la sentencia SQL
String query = "{ call sp_deleteProvision(?, ?) }";
//Instanciamos un objeto del tipo CallableStatement
CallableStatement stm = null;
try{
//Conectamos el objeto stm con la sentencia sql
stm = (CallableStatement) cn.prepareCall(query);
//Enviamos los argumentos
stm.setString("p_Producto", producto);
stm.setString("p_Proveedor", proveedor);
//Ejecutamos la sentencia
stm.execute();
}
}

```

**Fuente:** Elaboración propia (2015)

La figura 163 muestra la programación desarrollada en Java que permite la eliminación de un abastecimiento en particular desde la aplicación web construida. En ella se puede apreciar la invocación del procedimiento almacenado y los parámetros que esta recibe para realizar la operación.

**Anexo 7: Relación de costos para elaborar reportes de contabilidad administrativa.**

Resumen de planilla para generación de reportes de Contabilidad Administrativa									
<b>Periodo Junio 2014</b>									
N°	Nombre del empleado	Tipo contrato	Tarea	Horas laborables	Sueldo	Horas extras	Precio x HE	Total HE	Total
1	Angela Muñoz Valverde	Interno	Reportes Contabilidad Administrativa	8	\$ 2,500.00	16	\$ 15.50	\$ 248.00	\$ 2,748.00
2	Sonia Torres Baca	Interno	Reportes Contabilidad Administrativa	8	\$ 1,500.00	16	\$ 7.50	\$ 120.00	\$ 1,620.00
3	Maribel Arana Beltrán	Externo	Reportes Contabilidad Administrativa	-	\$ 4,200.00	-	-	-	\$ 4,200.00
<b>Sub - Total:</b>					\$ 8,200.00	32	\$ 23.00	\$ 368.00	\$ 8,568.00
<b>Periodo Junio 2015</b>									
N°	Nombre del empleado	Tipo contrato	Tarea	Horas laborables	Sueldo	Horas extras	Precio x HE	Total HE	Total
1	Angela Muñoz Valverde	Interno	Reportes Contabilidad Administrativa	8	\$ 2,500.00	1	\$ 15.50	\$ 15.50	\$ 2,515.50
2	Maribel Arana Beltrán	Externo	Reportes Contabilidad Administrativa	-	\$ 4,200.00	-	-	-	\$ 4,200.00
<b>Sub - Total:</b>					\$ 6,700.00	1	\$ 15.50	\$ 15.50	\$ 6,715.50
<b>Reduccion:</b>									<b>22%</b>
 <p>CORPORACION ARZAPALO SAC Carlos Arzapalo Huayta Gerente</p>									

**Anexo 8: Acta de inicio del proyecto**

ACTA DE REUNIÓN		
REF: Inicio del proyecto	FECHA: 21/02/2015	Página 1 de 2
Hora inicio: 10:00 am	Hora Fin: 10:35 am	
Empresa: Corporación Arzapalo S. A. C.		
<b>ASISTENTES A LA REUNIÓN:</b>		
1. Carlos Arzapalo	3. Sebastián Aguilar	
2. Luis Paraguay		
ASUNTOS TRATADOS		
Luis Paraguay se presenta ante el cliente y presenta también al equipo de desarrollo de software.		
Carlos Arzapalo se presenta y disculpa la ausencia de la contadora. Otorga el número de contacto de la contadora para consultas y pedidos de información necesaria para el desarrollo del sistema.		
El cliente expone su demanda y sus expectativas acerca del sistema a desarrollar. Menciona que los módulos desarrollados estén orientados a los productos en forma unitaria y en forma mensual.		
Luis Paraguay pregunta sobre los módulos que va a poseer el sistema. Muestra al cliente una lista de los temas tratados en la contabilidad administrativa.		
Carlos Arzapalo recibe la lista y comienza a decir cuáles serán los temas a desarrollar para el software. Los temas a tratar son el comportamiento de costo, en análisis costo – volumen – utilidad y menciona el análisis de mercado como el más importante. Pregunta la forma de desarrollar estos módulos.		
Sebastián Aguilar toma la palabra y menciona que el desarrollo se hará conforme a lo solicitado por el usuario. Añade también que los usuarios serán parte del proyecto como stakeholders para disminuir la resistencia al cambio y realizar un diseño amigable y personalizado.		
El cliente acepta la propuesta y hace referencia que estas reuniones se hagan los días lunes o martes en donde hay baja venta y en horas de la mañana para no interferir con las tareas de los colaboradores.		
Los presentes definen los roles.		
Los presentes dan por inicio el proyecto.		



<b>REF:</b> Inicio del proyecto	<b>FECHA:</b> 21/02/2015	<b>Página 2 de 2</b>
<b>ACUERDOS</b>		<b>Responsable</b>
Coordinar con la contadora para obtener información necesaria para la construcción del software		Carlos Arzapalo
Autorizar y comunicar la participación de los colaboradores de la empresa en la construcción del software.		Carlos Arzapalo
Dar seguimiento a los roles establecidos para el desarrollo del software		Luis Paraguay
<b>PROXIMA REUNIÓN</b>		
<b>ASUNTOS A TRATAR</b>		<b>Fecha</b>
Revisar las historias de usuarios y definir las prioridades.		25/02/2015
Definir la fecha de inicio del desarrollo del sistema de información.		25/02/2015

**CORPORACION ARZAPALO SAC**  
  
 .....  
 Carlos Arzapalo Huayua  
 Gerente

**Anexo 9: Acta de revisión de historias de usuarios**

ACTA DE REUNIÓN		
REF: Revisión de historias	FECHA: 25/02/2015	Página 1 de 2
Hora inicio: 10:30 am	Hora Fin: 12:12 am	
Empresa: Corporación Arzapalo S. A. C.		
<b>ASISTENTES A LA REUNIÓN:</b>		
1. Carlos Arzapalo Huauya	3. Ángela Muñoz	
2. Luis Paraguay Arzapalo	4. Sebastián Aguilar	
ASUNTOS TRATADOS		
Luis Paraguay inicia la reunión presentando todas las historias de usuarios recopiladas en el transcurso de la semana. Presenta 35 historias de usuarios al cliente.		
Carlos Arzapalo elimina 5 historias de usuarios por considerar no relevantes. Considera que esos reportes se pueden extraer desde el sistema de ventas que tienen.		
Carlos Arzapalo también añade un requerimiento. El desea que el sistema se pueda ver desde su Tablet Samsung.		
Sebastián Aguilar toma la palabra y menciona que los requerimientos tomados solo funcionan dentro de la empresa y no tiene salida a internet debido a que su servidor no soporta esa función.		
Carlos Arzapalo menciona que está conversando con su proveedor para que el servidor tenga acceso a internet y que esto será para octubre cuando renueve el contrato de mantenimiento y que necesita este requerimiento ahora para no tener que incurrir en gastos de adaptación.		
Ángela Muñoz toma la palabra y menciona que ella es la encargada de revisar los productos que llegan en almacén y en ese lugar no posee una computadora y usa la Tablet para esa tarea.		
Luis Paraguay toma nota de este requerimiento y procede a agregarlo como historia de usuario.		
Carlos y Luis proceden a otorgar prioridades a las 31 historias de usuarios establecidas.		
Carlos Arzapalo aprueba las historias de usuarios y autoriza a Ángela Muñoz a apoyar en el desarrollo del sistema.		

<b>REF:</b> Revisión de historias	<b>FECHA:</b> 25/02/2015	<b>Página 2 de 2</b>
Carlos Arzapalo firma el acta		
<b>ACUERDOS</b>		<b>Responsable</b>
Realizar el seguimiento de la construcción del software según lo establecido en las historias de usuarios.		Luis Paraguay
Coordinar cualquier cambio en los requerimientos con el jefe del proyecto.		Carlos Arzapalo
Apoyar en la construcción y definir el diseño		Ángela Muñoz
Apoyar y aprobar los diseños de prototipos		Carlos Arzapalo, Ángela Muñoz
<b>PROXIMA REUNIÓN</b>		
<b>ASUNTOS A TRATAR</b>		<b>Fecha</b>
Revisar la planificación del primer sprint		28/02/2015
Definir los entregables del primer sprint		28/02/2015

  
**CORPORACION ARZAPALO SAC**  
 .....  
 Carlos Arzapalo Huauya  
 Gerente

**Anexo 10: Acta de inicio del primer sprint**

<b>ACTA DE REUNIÓN</b>		
<b>REF:</b> Inicio del 1° Sprint	<b>FECHA:</b> 28/02/2015	<b>Página 1 de 2</b>
<b>Hora inicio:</b> 10:00 am	<b>Hora Fin:</b> 10:25 am	
<b>Empresa:</b> Corporación Arzapalo S. A. C.		
<b>ASISTENTES A LA REUNIÓN:</b>		
<ol style="list-style-type: none"> <li>1. Carlos Arzapalo Huauya</li> <li>2. Luis Paraguay Arzapalo</li> </ol>		
<b>ASUNTOS TRATADOS</b>		
Luis Paraguay realiza un checklist de las historias de usuarios a realizar en la primera iteración.		
Carlos Arzapalo toma la palabra y pregunta si se puede tener el producto antes de la fecha planificada.		
Luis Paraguay indica que los tiempos estimados se elaboraron de acuerdo a la disponibilidad de la asistente contable y explica la metodología de trabajo que se efectuará a lo largo del desarrollo del sistema de información.		
Carlos Arzapalo entiende lo mencionado por Luis Paraguay y aprueba la estimación del sprint y la pila de sprint en el siguiente orden:		
<ol style="list-style-type: none"> <li>1. Registro de proveedores</li> <li>2. Búsqueda de productos</li> <li>3. Registrar orden de compra</li> <li>4. Registrar producto en la orden de compra</li> <li>5. Eliminar producto en la orden de compra</li> <li>6. Reporte de costo variable unitario por producto</li> <li>7. Reporte de costo variable unitario por periodos</li> </ol>		
Carlos Arzapalo pregunta sobre la visualización en la Tablet		
Luis Paraguay menciona que ese requerimiento está consignado en todas las iteraciones para que el equipo de desarrollo lo tenga en cuenta al momento de la construcción de todos los módulos.		
Carlos Arzapalo entrega el diseño de los reportes en formato Excel.		
Carlos Arzapalo confirma la aceptación de la planificación y de las historias.		
Ambos fijan como fecha de inicio de la iteración para el 02 de marzo del presente año y fijan como fecha de presentación del producto para el día 26 de marzo.		


<b>REF:</b> Inicio del 1° Sprint	<b>FECHA:</b> 28/02/2015	<b>Página 2 de 2</b>
<b>ACUERDOS</b>		<b>Responsable</b>
Cumplir con las fechas establecidas y realizar el seguimiento del proyecto		Luis Paraguay
Facilitar información para el normal desarrollo del sistema.		Carlos Arzapalo
Mantener comunicación constante para que el desarrollo del sistema sea exitoso.		Carlos Arzapalo, Luis Paraguay
<b>PROXIMA REUNIÓN</b>		
<b>ASUNTOS A TRATAR</b>		<b>Fecha</b>
Revisar el producto del sprint		26/03/2015
Aprobar el producto y otorgar el pase de la siguiente fase		26/03/2015

**CORPORACION ARZAPALO SAC**

*Carlos Arzapalo Huayra*  
Gerente




**Anexo 11: Acta de fin del primer sprint**

ACTA DE REUNIÓN		
REF: Fin del 1° Sprint	FECHA: 26/03/2015	Página 1 de 1
Hora inicio: 9:00 am	Hora Fin: 10:22 am	
Empresa: Corporación Arzapalo S. A. C.		
<b>ASISTENTES A LA REUNIÓN:</b>		
1. Carlos Arzapalo Huauya	3. Ángela Muñoz	
2. Luis Paraguay Arzapalo		
ASUNTOS TRATADOS		
Luis Paraguay muestra el producto del primer sprint. Hace un contraste entre las historias de usuarios del primer sprint, el diseño aprobado por el usuario y el producto entregado.		
Carlos Arzapalo y Ángela Muñoz revisan el diseño y el producto terminado.		
Carlos Arzapalo pide que se le muestre el producto en una Tablet.		
Luis Paraguay hace la simulación en su computadora y se comprueba que el diseño se adapta al tamaño de la pantalla.		
Carlos Arzapalo aprueba el producto entregado.		
Luis Paraguay menciona el caso del costo variable unitario, el cual que el redondeo hace que el resultado sea distinto cuando se ingresa el monto mínimo y el monto máximo.		
Carlos Arzapalo indica que tiene en cuenta ello y que no hay problema. Revisa el cálculo del costo variable.		
Ambos dan por culminado el primer sprint.		
Ambos acuerdan que el despliegue del software comience a las 6 pm del día de hoy		
ACUERDOS		Responsable
Autorizar el despliegue del software.		Carlos Arzapalo
Dar seguimiento al despliegue para que no impacte en los sistemas actuales.		Luis Paraguay
PROXIMA REUNIÓN		
ASUNTOS A TRATAR		Fecha
Planificar y aprobar el segundo sprint		26/03/2015
CORPORACION ARZAPALO SAC  Carlos Arzapalo Huauya Gerente		

**Anexo 12: Acta de inicio del segundo sprint**

<b>ACTA DE REUNIÓN</b>		
<b>REF:</b> Inicio del 2° Sprint	<b>FECHA:</b> 26/03/2015	<b>Página 1 de 2</b>
<b>Hora inicio:</b> 11:30 am	<b>Hora Fin:</b> 12:05 am	
<b>Empresa:</b> Corporación Arzapalo S. A. C.		
<b>ASISTENTES A LA REUNIÓN:</b>		
1. Carlos Arzapalo Huauya 2. Luis Paraguay Arzapalo		
<b>ASUNTOS TRATADOS</b>		
Luis Paraguay realiza un checklist de las historias de usuarios a realizar en la segunda iteración.		
Carlos Arzapalo revisa las historias de usuarios y revisa la validación de cada historia de usuario.		
Luis Paraguay indica que el desarrollo también se hará el diseño adaptable.		
Carlos Arzapalo aprueba la estimación del sprint y la pila de sprint. Para ello indica el siguiente orden:		
1. Reporte del margen de contribución unitario por periodos 2. Reporte de evolución del margen de contribución unitario por productos 3. Reporte del punto de equilibrio por periodos 4. Reporte de evolución del punto de equilibrio por producto. 5. Reporte de margen de seguridad por periodos		
Carlos Arzapalo pregunta sobre la historia que trata sobre la visualización del producto en la Tablet.		
Luis Paraguay le recuerda que ese requerimiento estará presente en todo el desarrollo del sistema para que el desarrollador tenga en cuenta que lo desarrollado debe ser adaptable a una Tablet.		
Carlos Arzapalo entiende el procedimiento y acepta la planificación del segundo sprint.		
Carlos Arzapalo entrega el diseño de los reportes e indica que para el reporte de evolución del punto de equilibrio incluya el gráfico del monto de equilibrio también.		
Luis Paraguay actualiza la historia de usuario.		
Ambos fijan como fecha de inicio de la iteración para el 27 de marzo del presente año y fijan como fecha de presentación del producto para el día 23 de abril.		


<b>REF:</b> Inicio del 2° Sprint	<b>FECHA:</b> 26/03/2015	<b>Página 2 de 2</b>
<b>ACUERDOS</b>		<b>Responsable</b>
Cumplir con las fechas establecidas y realizar el seguimiento del proyecto		Luis Paraguay
Facilitar información para el normal desarrollo del sistema.		Carlos Arzapalo
Mantener comunicación constante para que el desarrollo del sistema sea exitoso.		Carlos Arzapalo, Luis Paraguay
<b>PROXIMA REUNIÓN</b>		
<b>ASUNTOS A TRATAR</b>		<b>Fecha</b>
Revisar el producto del sprint		23/04/2015
Aprobar el producto y autorizar el pase a producción		23/04/2015

**CORPORACION ARZAPALO SAC**  
  
 .....  
 Carlos Arzapalo Huauya  
 Gerente



**Anexo 13: Acta de finalización del segundo sprint**


ACTA DE REUNIÓN		
REF: Fin del 2° Sprint	FECHA: 23/04/2015	Página 1 de 1
Hora inicio: 7:10 pm	Hora Fin: 7:42 pm	
Empresa: Corporación Arzapalo S. A. C.		
<b>ASISTENTES A LA REUNIÓN:</b>		
1. Carlos Arzapalo Huauya 2. Luis Paraguay Arzapalo		
ASUNTOS TRATADOS		
Luis Paraguay muestra el producto del segundo sprint y contrasta las historias de usuarios, el diseño aprobado por el usuario y el producto entregado.		
Carlos Arzapalo revisa el cálculo del margen de contribución, el punto de equilibrio y el margen de seguridad.		
Luis Paraguay le indica que el precio del producto se tomó del promedio del mes como le indicó.		
Carlos Arzapalo indica que el cálculo es correcto y que le muestre la visualización del producto en la Tablet.		
Luis Paraguay realiza la simulación de la vista en Tablet.		
Carlos Arzapalo aprueba el producto y el diseño de los reportes.		
Ambos dan por culminado el segundo sprint.		
Ambos acuerdan que el despliegue del software comience el sábado 25 de abril		
ACUERDOS		Responsable
Autorizar el despliegue del software.		Carlos Arzapalo
Dar seguimiento al despliegue para que no impacte en los sistemas actuales.		Luis Paraguay
PROXIMA REUNIÓN		
ASUNTOS A TRATAR		Fecha
Planificar y aprobar el tercer sprint		23/04/2015

**CORPORACION ARZAPALO SAC**  
  
 Carlos Arzapalo Huauya  
 Gerente

**Anexo 14: Acta de inicio del tercer sprint**

ACTA DE REUNIÓN		
<b>REF:</b> Inicio del 3° Sprint	<b>FECHA:</b> 23/04/2015	<b>Página 1 de 2</b>
<b>Hora inicio:</b> 7:45 pm	<b>Hora Fin:</b> 8:15 pm	
<b>Empresa:</b> Corporación Arzapalo S. A. C.		
<b>ASISTENTES A LA REUNIÓN:</b>		
<ol style="list-style-type: none"> <li>1. Carlos Arzapalo Huauya</li> <li>2. Luis Paraguay Arzapalo</li> </ol>		
ASUNTOS TRATADOS		
Luis Paraguay realiza un checklist de las historias de usuarios a realizar en la tercera iteración.		
Carlos Arzapalo revisa las historias de usuarios y revisa la validación de cada historia.		
Luis Paraguay indica que el desarrollo también se hará el diseño adaptable.		
Carlos Arzapalo mantiene las historias de usuario. No realiza modificación		
Carlos Arzapalo entrega el diseño de los reportes en formato Excel.		
Carlos Arzapalo aprueba la estimación del sprint y la pila de sprint. Para ello indica el siguiente orden:		
<ol style="list-style-type: none"> <li>1. Reporte de margen de seguridad por producto.</li> <li>2. Reporte de apalancamiento operativo por periodos.</li> <li>3. Reporte de apalancamiento operativo por producto.</li> <li>4. Reporte de rentabilidad por periodo</li> <li>5. Reporte de rentabilidad por distrito</li> <li>6. Reporte de rentabilidad por producto</li> </ol>		
Carlos Arzapalo aprueba la planificación		
Ambos fijan como fecha de inicio de la iteración para el 24 de abril del presente año y fijan como fecha de presentación del producto para el día 21 de mayo.		

<b>REF:</b> Inicio del 3° Sprint	<b>FECHA:</b> 23/04/2015	<b>Página 2 de 2</b>
<b>ACUERDOS</b>		<b>Responsable</b>
Cumplir con las fechas establecidas y realizar el seguimiento del proyecto		Luis Paraguay
Facilitar información para el normal desarrollo del sistema.		Carlos Arzapalo
Mantener comunicación constante para que el desarrollo del sistema sea exitoso.		Carlos Arzapalo, Luis Paraguay
<b>PROXIMA REUNIÓN</b>		
<b>ASUNTOS A TRATAR</b>		<b>Fecha</b>
Revisar el producto del sprint		21/05/2015
Aprobar el producto y otorgar el pase de la siguiente fase		21/05/2015

  
**CORPORACION ARZAPALO SAC**  
 .....  
 Carlos Arzapalo Huauya  
 Gerente

**Anexo 15: Acta de finalización del tercer sprint**

ACTA DE REUNIÓN		
REF: Fin del 3° Sprint	FECHA: 21/05/2015	Página 1 de 1
Hora inicio: 6:40 pm	Hora Fin: 7:35 pm	
Empresa: Corporación Arzapalo S. A. C.		
<b>ASISTENTES A LA REUNIÓN:</b>		
1. Carlos Arzapalo	3. Ángela Muñoz	
2. Luis Paraguay		
ASUNTOS TRATADOS		
Luis Paraguay muestra el producto del tercer sprint y contrasta las historias de usuarios, el diseño aprobado por el usuario y el producto entregado.		
Carlos Arzapalo y Ángela Muñoz revisan el cálculo de margen de seguridad, del apalancamiento operativo.		
Luis Paraguay indica que hay valores negativos en algunos casos.		
Carlos Arzapalo revisa el cálculo y confirma que cuando existen valores negativos se tratan de pérdidas.		
Carlos Arzapalo indica que todos los cálculos son correctos. Procede a revisar el diseño adaptable.		
Luis Paraguay realiza la simulación de la vista en Tablet.		
Carlos Arzapalo aprueba el entregable de la tercera iteración.		
Los participantes dan por culminado el tercer sprint.		
Carlos Arzapalo y Luis Paraguay acuerdan que el despliegue del software se realice el sábado 23 de mayo		
ACUERDOS		Responsable
Autorizar el despliegue del software.		Carlos Arzapalo
Dar seguimiento al despliegue para que no impacte en los sistemas actuales.		Luis Paraguay
PROXIMA REUNIÓN		
ASUNTOS A TRATAR		Fecha
Planificar y aprobar el cuarto sprint		21/05/2015

**CORPORACION ARZAPALO SAC**  
  
 Carlos Arzapalo Hualpa  
 Gerente




**Anexo 16: Acta de inicio del cuarto sprint**

<b>ACTA DE REUNIÓN</b>		
<b>REF:</b> Inicio del 4° Sprint	<b>FECHA:</b> 21/05/2015	<b>Página 1 de 2</b>
<b>Hora inicio:</b> 8:00 pm	<b>Hora Fin:</b> 8:25 pm	
<b>Empresa:</b> Corporación Arzapalo S. A. C.		
<b>ASISTENTES A LA REUNIÓN:</b>		
1. Carlos Arzapalo Huauya	3. Ángela Muñoz	
2. Luis Paraguay Arzapalo		
<b>ASUNTOS TRATADOS</b>		
Luis Paraguay realiza un checklist de las historias de usuarios a realizar en la cuarta iteración.		
Carlos Arzapalo revisa las historias de usuarios y revisa la validación de cada historia.		
Ángela Muñoz indica que el diseño de los formularios se haga de la misma forma que los formularios realizados anteriormente para ahorrar tiempo.		
Luis Paraguay acepta la propuesta pero que de todas maneras el sábado se reunirán con el equipo de desarrollo para dar la conformidad de los prototipos.		
Ángela Muñoz acepta la reunión.		
Con respecto al reporte de rentabilidad del vendedor, Luis Paraguay muestra un ejemplo del reporte.		
Carlos Arzapalo lo revisa y realiza la modificación al reporte. Esta modificación es que se agrupen por periodos, que se añada un gráfico en pastel de las ventas de cada vendedor y que muestre información de los 6 últimos meses.		
Luis Paraguay actualiza la historia de usuario.		
Carlos Arzapalo indica que el gráfico de costos y ventas sea solo informativo y que no se visualice como PDF.		

<b>REF:</b> Inicio del 4° Sprint	<b>FECHA:</b> 21/05/2015	<b>Página 2 de 2</b>
<p>Carlos Arzapalo y Ángela Muñoz aprueban la estimación del sprint y la pila de sprint.                  Para ello indica el siguiente orden:</p> <ol style="list-style-type: none"> <li>1. Reporte de rentabilidad por vendedor</li> <li>2. Búsqueda de proveedores</li> <li>3. Registrar abastecimiento</li> <li>4. Buscar abastecimiento</li> <li>5. Buscar orden de compra</li> <li>6. Editar orden de compra</li> <li>7. Gráfico de costos vs ventas</li> </ol>		
<p>Carlos Arzapalo aprueba la planificación</p>		
<p>Ambos fijan como fecha de inicio de la iteración para el 22 de mayo del presente año y fijan como fecha de presentación del producto para el día 11 de junio.</p>		
<b>ACUERDOS</b>		<b>Responsable</b>
Cumplir con las fechas establecidas y realizar el seguimiento del proyecto		Luis Paraguay
Facilitar información para el normal desarrollo del sistema.		Carlos Arzapalo, Ángela Muñoz
Mantener comunicación constante para que el desarrollo del sistema sea exitoso.		Carlos Arzapalo, Luis Paraguay
<b>PROXIMA REUNIÓN</b>		
<b>ASUNTOS A TRATAR</b>		<b>Fecha</b>
Revisar el producto del sprint		11/06/2015
Aprobar el producto y otorgar el pase de la siguiente fase		11/06/2015

  
**CORPORACION ARZAPALO SAC**  
 .....  
 Carlos Arzapalo Huauya  
 Gerente

**Anexo 17: Acta de finalización del cuarto sprint**

ACTA DE REUNIÓN		
REF: Fin del 4° Sprint	FECHA: 11/06/2015	Página 1 de 1
Hora inicio: 7:30 pm	Hora Fin: 8:15 pm	
Empresa: Corporación Arzapalo S. A. C.		
<b>ASISTENTES A LA REUNIÓN:</b>		
1. Carlos Arzapalo	3. Ángela Muñoz	
2. Luis Paraguay		
ASUNTOS TRATADOS		
Luis Paraguay muestra el producto del cuarto sprint y contrasta las historias de usuarios, el diseño aprobado por el usuario y el producto entregado. Hace referencia al reporte modificado por Carlos Arzapalo al inicio del sprint.		
Carlos Arzapalo revisa el cambio que propuso.		
Carlos Arzapalo revisa detalladamente el reporte de rentabilidad por vendedor.		
Carlos Arzapalo visualiza valores idénticos. Hace los cálculos aisladamente y corrobora el resultado obtenido por el sistema.		
Carlos Arzapalo indica que todos los cálculos son correctos. Procede a revisar el diseño adaptable.		
Luis Paraguay realiza la simulación de la vista en Tablet.		
Ángela Muñoz hace las pruebas necesarias de los módulos que ella utiliza.		
Ángela Muñoz y Carlos Arzapalo aprueban el entregable de la cuarta iteración.		
Los participantes dan por culminado el cuarto sprint.		
Carlos Arzapalo y Luis Paraguay acuerdan que el despliegue del software se realice el sábado 13 de junio.		
ACUERDOS		Responsable
Autorizar el despliegue del software.		Carlos Arzapalo
Dar seguimiento al despliegue para que no impacte en los sistemas actuales.		Luis Paraguay
PROXIMA REUNIÓN		
ASUNTOS A TRATAR		Fecha
Planificar y aprobar el quinto sprint		11/06/2015
 CORPORACION ARZAPALO SAC Carlos Arzapalo Huayta Gerente		

**Anexo 18: Acta de inicio del quinto sprint**

<b>ACTA DE REUNIÓN</b>		
<b>REF:</b> Inicio del 5° Sprint	<b>FECHA:</b> 11/06/2015	<b>Página 1 de 2</b>
<b>Hora inicio:</b> 8:20 pm	<b>Hora Fin:</b> 8:50 pm	
<b>Empresa:</b> Corporación Arzapalo S. A. C.		
<b>ASISTENTES A LA REUNIÓN:</b>		
1. Carlos Arzapalo	3. Ángela Muñoz	
2. Luis Paraguay		
<b>ASUNTOS TRATADOS</b>		
Luis Paraguay realiza un checklist de las historias de usuarios a realizar en la tercera iteración.		
Carlos Arzapalo revisa las historias de usuarios y revisa la validación de cada historia.		
Ángela Muñoz toma la palabra e indica que se mantenga el diseño para los demás formularios y que le consulte cualquier inconveniente en el diseño para poder cambiarlo.		
Carlos Arzapalo y Ángela Muñoz revisan las validaciones		
Carlos Arzapalo aprueba la estimación del sprint y la pila de sprint. Para ello indica el siguiente orden:		
<ol style="list-style-type: none"> <li>1. Actualizar datos del proveedor</li> <li>2. Eliminar proveedor</li> <li>3. Editar abastecimiento</li> <li>4. Eliminar abastecimiento</li> </ol>		
Carlos Arzapalo aprueba la planificación		
Carlos Arzapalo y Luis Paraguay fijan como fecha de inicio de la iteración para el 12 de junio del presente año y fijan como fecha de presentación del producto para el día 23 de junio.		
<b>ACUERDOS</b>		<b>Responsable</b>
Cumplir con las fechas establecidas y realizar el seguimiento del proyecto		Luis Paraguay
Facilitar información para el normal desarrollo del sistema.		Carlos Arzapalo
Mantener comunicación constante para que el desarrollo del sistema sea exitoso.		Carlos Arzapalo, Luis Paraguay



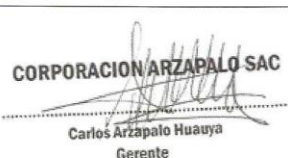
<b>REF:</b> Inicio del 5° Sprint	<b>FECHA:</b> 11/06/2015	<b>Página 2 de 2</b>
<b>PROXIMA REUNIÓN</b>		
<b>ASUNTOS A TRATAR</b>		<b>Fecha</b>
Revisar el producto del sprint		23/06/2015
Aprobar el producto y otorgar el pase de la siguiente fase		23/06/2015

**CORPORACION ARZAPALO SAC**  
  
 .....  
 Carlos Arzapalo Huauya  
 Gerente

**Anexo 19: Acta de fin del quinto sprint**

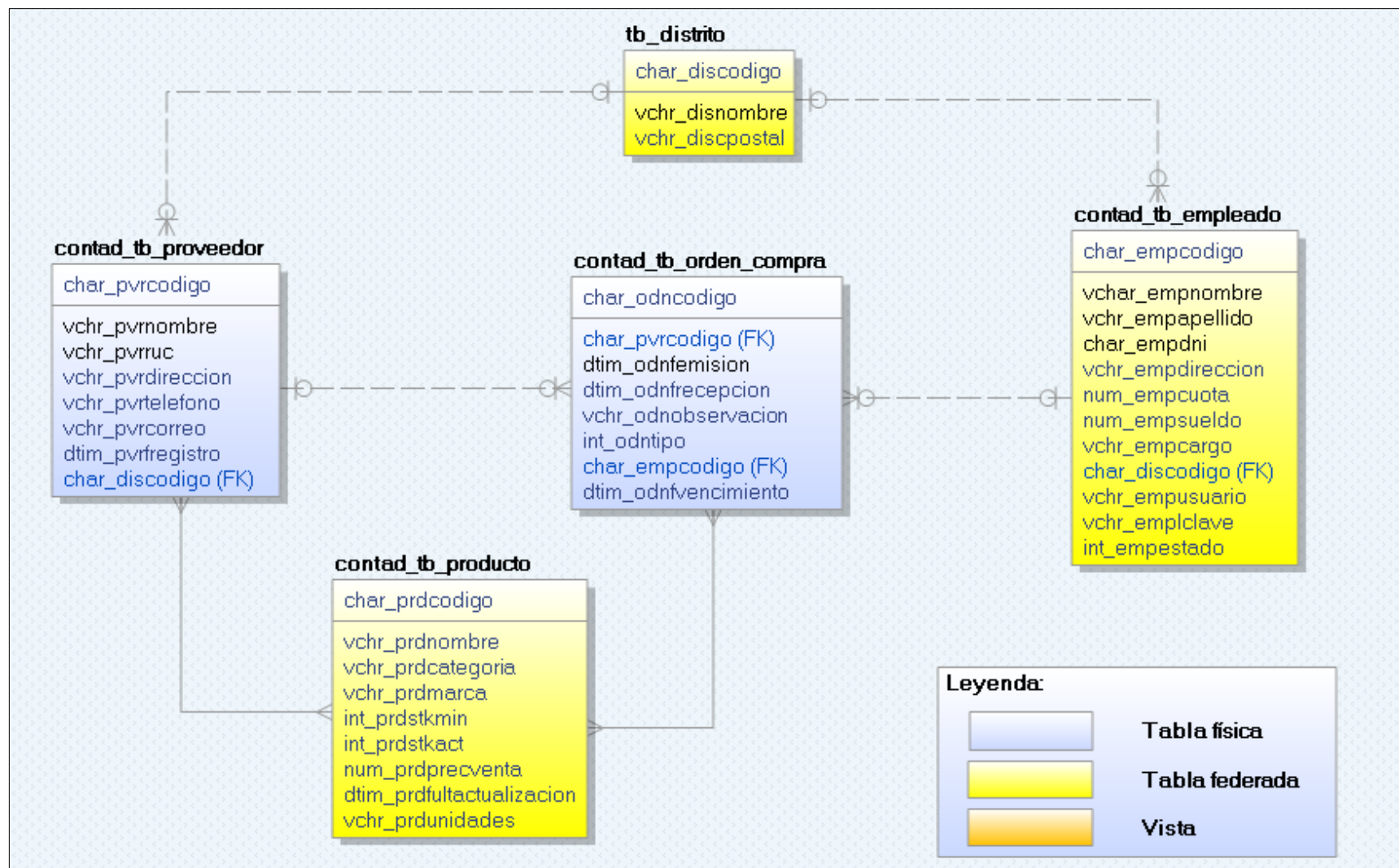
ACTA DE REUNIÓN		
REF: Fin del 5° Sprint	FECHA: 23/06/2015	Página 1 de 1
Hora inicio: 7:30 pm	Hora Fin: 8:40 pm	
Empresa: Corporación Arzapalo S. A. C.		
<b>ASISTENTES A LA REUNIÓN:</b>		
1. Carlos Arzapalo	3. Ángela Muñoz	
2. Luis Paraguay		
ASUNTOS TRATADOS		
Luis Paraguay muestra el producto del quinto sprint y contrasta las historias de usuarios, el diseño aprobado por el usuario y el producto entregado.		
Ángela Muñoz revisa y valida la funcionalidad entregada.		
Ángela Muñoz realiza las pruebas correspondientes.		
Carlos Arzapalo consulta sobre la visualización en Tablet.		
Luis Paraguay realiza la simulación de la vista en Tablet.		
Ángela Muñoz y Carlos Arzapalo aprueban el entregable de la quinta iteración.		
Los participantes dan por culminado el quinto sprint.		
Carlos Arzapalo y Luis Paraguay acuerdan que el despliegue del software se realice el sábado 27 de junio.		
ACUERDOS		Responsable
Autorizar el despliegue del software.		Carlos Arzapalo
Dar seguimiento al despliegue para que no impacte en los sistemas actuales.		Luis Paraguay
PROXIMA REUNIÓN		
ASUNTOS A TRATAR		Fecha
Firmar el acta de culminación del proyecto.		27/06/2015

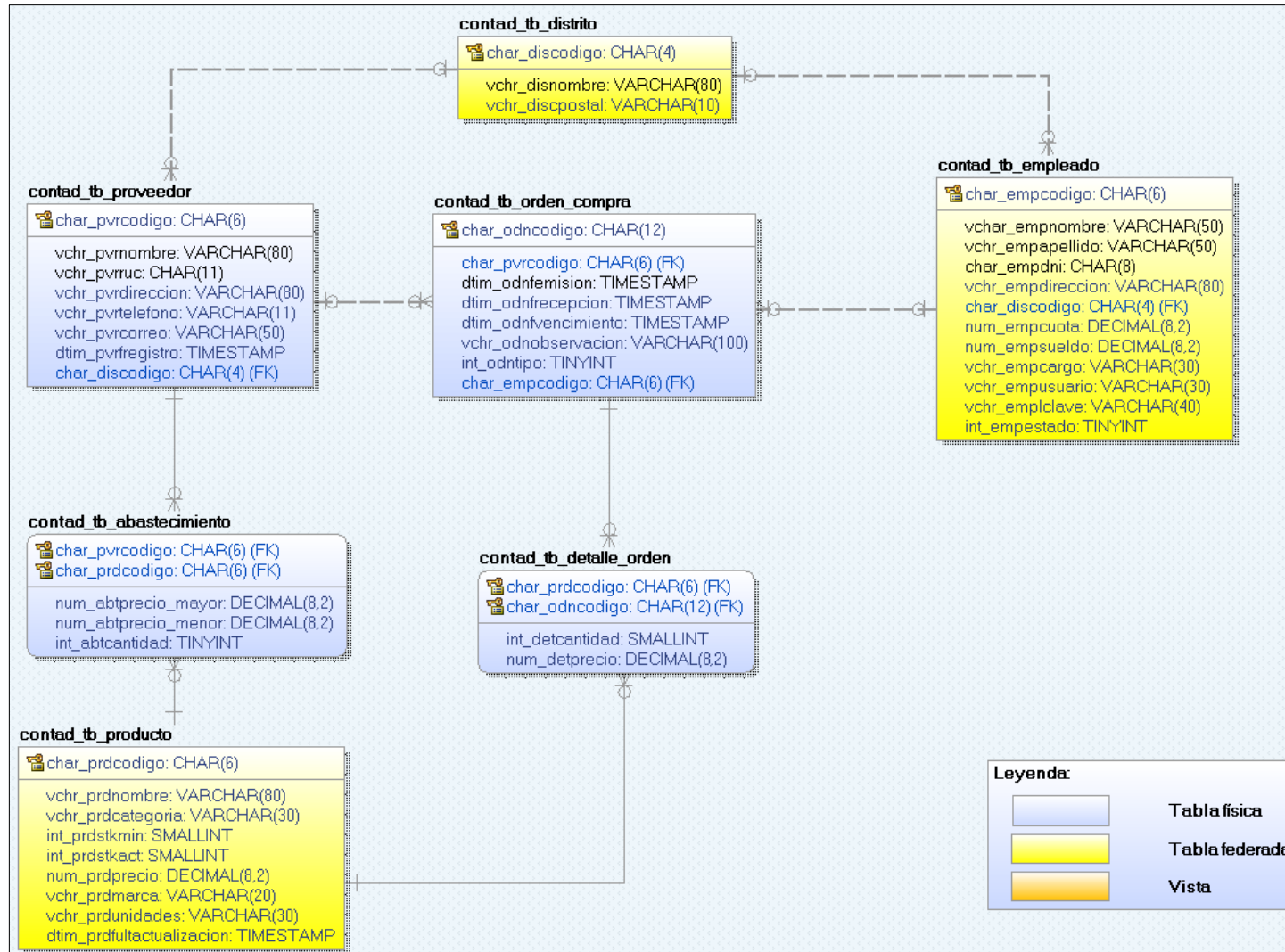


**CORPORACION ARZAPALO SAC**  
 Carlos Arzapalo Huauya  
 Gerente

**Anexo 20: Diseño lógico de la base de datos**



**Anexo 21: Diseño físico de la base de datos**



	<b>ACTA DE APROBACIÓN DE ORIGINALIDAD DE TESIS</b>	Código : F06-PP-PR-02.02 Versión : 09 Fecha : 23-03-2018 Página : 1 de 1
---	--	---

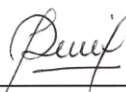
Yo, Mg. Orleans Moisés Gálvez Tapia, docente de la Facultad de Ingeniería y Escuela Profesional de Ingeniería de Sistemas de la Universidad César Vallejo Lima Norte, revisor de la tesis titulada:

“SISTEMA DE INFORMACIÓN PARA LA CONTABILIDAD ADMINISTRATIVA PARA LA EMPRESA CORPORACIÓN ARZAPALO S.A.C”

del estudiante PARAGUAY ARZAPALO LUIS ENRIQUE, constato que la investigación tiene un índice de similitud del 26% verificable en el reporte de originalidad del programa Turnitin.

El suscrito analizó dicho reporte y concluyó que cada una de las coincidencias detectadas no constituyen plagio. A mi leal saber y entender la tesis cumple con todas las normas para el uso de citas y referencias establecidas por la Universidad Cesar Vallejo.

Los Olivos, 30 de 05 del 2019.



Mg. Orleans Moisés Gálvez Tapia

Docente Asesor de Tesis

DNI: 16 798332

Feedback Studio - Google Chrome  
 https://ev.turnitin.com/app/carta/es/?student\_user=1&lang=es&o=1134594217&s=&u=1087842401

feedback studio Luis Enrique PARAGUAY ARZAPALO Tesis: Paraguay\_ALE



**UNIVERSIDAD CÉSAR VALLEJO**

**FACULTAD DE INGENIERÍA**

**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**

**“SISTEMA DE INFORMACIÓN PARA LA CONTABILIDAD ADMINISTRATIVA PARA LA EMPRESA CORPORACIÓN ARZAPALO S.A.C”**

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

**Resumen de coincidencias**

**26 %**

Se están viendo fuentes estándar

Ver fuentes en inglés (Beta)

Coincidencias

1	Entregado a Universida... <small>Trabajo del estudiante</small>	5 %	>
2	repositorio.ucv.edu.pe <small>Fuente de Internet</small>	4 %	>
3	docplayer.es <small>Fuente de Internet</small>	1 %	>
4	www.scribd.com <small>Fuente de Internet</small>	1 %	>
5	repositorio.unh.edu.pe <small>Fuente de Internet</small>	1 %	>
6	www.slideshare.net <small>Fuente de Internet</small>	1 %	>

Página: 1 de 80    Número de palabras: 14399    Text-only Report | High Resolution    Activado





UNIVERSIDAD CÉSAR VALLEJO

Centro de Recursos para el Aprendizaje y la Investigación (CRAI)  
 “César Acuña Peralta”

## FORMULARIO DE AUTORIZACIÓN PARA LA PUBLICACIÓN ELECTRÓNICA DEL TRABAJO DE INVESTIGACIÓN O LA TESIS

### 1. DATOS PERSONALES

Apellidos y Nombres: (solo los datos del que autoriza)

..... Paraguay Arzapalo, Luis Enrique .....  
 D.N.I. : 46671622 .....  
 Domicilio : Av. B. H. D. U. 9. Urb. Las Terrazas de Carapungillo, Lingarcho .....  
 Teléfono : Fijo : ..... Móvil : 997.837.099 .....  
 E-mail : luis.paraguay@icloud.com .....

### 2. IDENTIFICACIÓN DE LA TESIS

Modalidad:

- Trabajo de Investigación de Pregrado  
 Tesis de Pregrado

Facultad : Ingeniería .....  
 Escuela : Ingeniería de sistemas .....  
 Carrera : Ingeniería de sistemas .....  
 Grado  Título  
 ..... Ingeniero de sistemas .....

Tesis de Post Grado

- Maestría  Doctorado  
 Grado : .....  
 Mención : .....

### 3. DATOS DE LA TESIS

Autor (es) Apellidos y Nombres:

..... Paraguay Arzapalo, Luis Enrique .....

Título del trabajo de investigación o de la tesis:

..... Sistema de información para la contabilidad administrativa .....  
 ..... para la empresa Corporación Arzapalo S.A.C. ....

Año de publicación : 2019 .....

### 4. AUTORIZACIÓN DE PUBLICACIÓN DE LA TESIS EN VERSIÓN ELECTRÓNICA:

A través del presente documento,

- Si autorizo a publicar en texto completo mi trabajo de investigación o tesis.  
 No autorizo a publicar en texto completo mi trabajo de investigación o tesis.

Firma :  .....

Fecha : 03/06/19 .....



## UNIVERSIDAD CÉSAR VALLEJO

### AUTORIZACIÓN DE LA VERSIÓN FINAL DEL TRABAJO DE INVESTIGACIÓN

CONSTE POR EL PRESENTE EL VISTO BUENO QUE OTORGA EL ENCARGADO DE INVESTIGACIÓN DE:

LA ESCUELA ACADÉMICO PROFESIONAL DE INGENIERÍA DE SISTEMAS

A LA VERSIÓN FINAL DE INVESTIGACIÓN QUE PRESENTA:

PARAGUAY ARZAPALO, LUIS ENRIQUE

INFORME TITULADO:

SISTEMA DE INFORMACIÓN PARA LA CONTABILIDAD ADMINISTRATIVA PARA LA EMPRESA CORPORACIÓN ARZAPALO S.A.C.

PARA OBTENER EL TÍTULO O GRADO DE:

**INGENIERO DE SISTEMAS**

SUSTENTACIÓN EN FECHA: ..... 18 de agosto de 2015 .....

NOTA O MENCIÓN: ..... 15 .....



FIRMA DEL ENCARGADO DE INVESTIGACIÓN