



**UNIVERSIDAD CÉSAR VALLEJO**

**FACULTAD DE INGENIERÍA Y ARQUITECTURA**

**ESCUELA PROFESIONAL INGENIERIA DE SISTEMAS**

**Sistema web para la gestión de incidencias en el Colegio**

**Suizo Pestalozzi – Miraflores**

TESIS PARA OBTENER EL TÍTULO PROFESIONAL DE:

Ingeniero de Sistemas

**AUTORES:**

Alvarado Flores, Carlos Alberto (ORCID: 0000-0001-6382-9834)

Dorador Diaz, Luis Felipe (ORCID: 0000-0002-8465-0718)

**ASESOR:**

Dr. Chávez Pinillos, Frey Elmer (ORCID: 0000-0003-3785-5259)

**LÍNEA DE INVESTIGACIÓN:**

Sistemas de Información y Comunicación

**LIMA - PERÚ**

**2020**

## **Dedicatoria**

La presente tesis la dedicó a mi Madre, Esposa e Hijos que fueron mi motor y pilar fundamental en todo este tiempo de carrera y por supuesto a Dios por haberme permitido llegar hasta este punto y darme su Bendición.

Dedicó el presente proyecto de investigación a mis Padres, Hermana, Cuñado, R.F.S, por el apoyo que me han dado en este tiempo y también a Dios al no dejar que caiga en mi voluntad de llegar hasta este punto.

## **Agradecimiento**

El presente trabajo de tesis a ti Dios por bendecirnos la vida, por guiarnos a lo largo de nuestra existencia, ser el apoyo y fortaleza.

A la UNIVERSIDAD CESAR VALLEJO por darme la oportunidad de estudiar y ser un profesional.

Alvarado Flores Carlos Alberto

El presente trabajo de tesis primeramente me gustaría agradecerte a ti Dios por bendecirme para llegar hasta donde he llegado, porque hiciste realidad este sueño anhelado.

A la UNIVERSIDAD CESAR VALLEJO por darme la oportunidad de estudiar y ser un profesional.

Dorador Diaz Luis Felipe

## Índice de contenidos

Carátula.....	i
Dedicatoria .....	ii
Agradecimiento .....	ii
Índice de contenidos.....	iv
Índice de tablas .....	v
Índice de figuras .....	vii
Resumen.....	vii
Abstract.....	viii
I. INTRODUCCIÓN .....	1
II. MARCO TEÓRICO.....	8
III. MÉTODO.....	28
3.1. Tipo y diseño de la investigación .....	28
3.1.1. Tipo de investigación .....	28
3.1.2. Diseño de investigación .....	28
3.2. Variables y Operacionalización .....	29
3.2.1. Variables.....	29
3.3. Población, muestra y muestreo.....	32
3.4. Técnicas e instrumentos de recolección de datos.....	33
3.5. Procedimientos .....	35
3.6. Método de análisis de datos.....	36
3.7. Aspectos Éticos.....	39
IV. RESULTADOS.....	39
4.1. Analisis Descriptivo .....	39
4.2. Analisis Inferencial .....	41
4.3. Prueba de Hipótesis.....	46
IV. DISCUSIÓN .....	51
IV. CONCLUSIONES .....	52
IV. RECOMENDACIONES.....	53
REFERENCIAS.....	54
ANEXOS .....	62

## Índice de tablas

Tabla N° 1. Diferencia entre metodologías de desarrollo de software .....	17
Tabla N° 2. Ventajas y Desventajas Sistema Web .....	21
Tabla N° 3. Operacionalización de Variables e Indicadores .....	31
Tabla N° 4. Determinar la Población.....	32
Tabla N° 5. Determinar la Muestra.....	33
Tabla N° 6. Validación de Instrumentos.....	35
Tabla N° 7. Estadísticos Descriptivos de RULI antes y después de la implementar el Sistema Web.....	41
Tabla N° 8. Estadísticos Descriptivos de RULI antes y después de la implementar el Sistema Web.....	42
Tabla N° 9. Prueba de Normalidad de RRI antes y después de la implementar el Sistema Web .....	44
Tabla N° 10. Prueba de Normalidad de RULI antes y después de la implementar el Sistema Web.....	44
Tabla N° 11. Prueba de T-Student para el Ratio de Resolución de Incidencias en la Gestión de Incidencias antes y después de implementar el Sistema Web ....	49
Tabla N° 12. Prueba de T-Student para el Ratio de Utilización Laboral en Incidencias en la Gestión de Incidencias antes y después de implementar el Sistema Web .....	52

## Índice de figuras

Figura N° 1. Diagrama de Cantidad de Incidencias por Áreas del Colegio Pestalozzi en 1 mes .....	3
Figura N° 2. Diagrama de Ratio de Utilización Laboral de 1 mes de 2020.....	4
Figura N° 3. Diagrama de Ishikawa de las Incidencias en el Colegio Pestalozzi ..	5
Figura N° 4. Metodología Ágil SCRUM .....	15
Figura N° 5. Tablero Kanban.....	17
Figura N° 6. Modelo Vista Controlador.....	21
Figura N° 7. Gestión de incidencias según ITIL V3.....	22
Figura N° 8. Región de aceptación y región de Rechazo para una hipótesis bilateral o unilateral .....	39
Figura N° 9. Porcentaje de Ratio de Resolución de Incidencias antes y después de Implementar el Sistema Web .....	40
Figura N° 10. Porcentaje de Ratio de Utilización Laboral en Incidencias antes y después de Implementar el Sistema Web .....	41
Figura N° 11. Prueba de normalidad de Ratio de Resolución de Incidencias antes de implementar el Sistema Web.....	43
Figura N° 12. Prueba de normalidad de Ratio de Resolución de Incidencias después de implementar el Sistema Web .....	43
Figura N° 13. Prueba de normalidad de Ratio de Utilización Laboral en Incidencias antes de implementar el Sistema Web .....	45
Figura N° 14. Prueba de normalidad de Ratio de Utilización Laboral en Incidencias después de implementar el Sistema Web .....	45
Figura N° 15. Ratio de Resolución de Incidencias antes y después de Implementar el Sistema Web .....	47
Figura N° 16. Prueba T-Student - Ratio de Resolución de Incidencias .....	48
Figura N° 17. Ratio de Utilización Laboral en Incidencias antes y después de Implementar el Sistema Web .....	49
Figura N° 18. Prueba T-Student - Ratio de Utilización Laboral en Incidencias ...	50

## Resumen

La presente investigación se centró en dar respuesta al problema de investigación relativo, al impacto que tiene la implementación de un sistema web en la gestión de incidencias en el área de soporte técnico del Colegio Suizo Pestalozzi – Miraflores. El objetivo de la investigación estuvo orientado a determinar la influencia del Sistema Web para la gestión de incidencias en el área de soporte técnico del Colegio Suizo Pestalozzi - Miraflores.

La metodología SCRUM se utilizó para el desarrollo del sistema web porque se adapta mejor a los requisitos y a las fases del proyecto y, por lo tanto, es rápida en tiempos de resolución de incidentes. Esto genera una solución más acelerada y empleó un diseño pre experimental basado en los elementos de Pre y Post Test.

Para el desarrollo del proyecto se hizo uso de dos indicadores, los cuales estuvieron conformados por el ratio de resolución de incidencias y el ratio de utilización laboral en incidencias, teniendo como población un total de 21 fichas de registros para los dos indicadores, de acuerdo con las hojas de observación realizadas como una prueba previa de la investigación. Asimismo, para la obtención de los datos se utilizaron los registros correspondientes a la atención del software de hoja de cálculo, obtenidos del soporte técnico.

Las conclusiones determinaron que el sistema web influyó en la optimización del proceso de gestión de incidentes, al ofrecer soluciones rápidas y eficientes a las necesidades del usuario.

**Palabras clave:** Sistema Web, Gestión de Incidencias, Scrum.

## **Abstract**

This research focused on responding to the research problem related to the impact that the implementation of a web system has on the management of incidents in the technical support area of Colegio Suizo Pestalozzi –Miraflores. The objective of the research was aimed at determining the influence of the Web System for incident management in the technical support area of the Swiss College Pestalozzi - Miraflores.

The SCRUM methodology was used for the development of the web system because it is better adapted to the requirements and the phases of the project and, therefore, it is fast in times of incident resolution. This generates a more accelerated solution and used a pre-experimental design based on the Pre and Post Test elements.

For the development of the project, two indicators were used, which were made up of the ratio of resolution of incidents and the ratio of employment use in incidents, having as a population a total of 21 records for the two indicators, according to the observation sheets carried out as a preliminary test of the investigation. Likewise, to obtain the data, the records corresponding to the attention of the spreadsheet software, obtained from technical support, were used. The conclusions determined that the web system influenced the optimization of the incident management process, by offering fast and efficient solutions to the user's needs.

**Keywords:** Web System, Incidence Management, Scrum



## **I. INTRODUCCIÓN**

Gómez (2014, párr. 1) señala que a nivel mundial los incidentes en la Web vienen siendo cada vez más recurrentes, debido fundamentalmente al crecimiento exponencial de las redes y las telecomunicaciones, así como también, por los aportes que se viene generando en el entorno de las tecnologías de la información y las comunicaciones (TIC). En este sentido, al no desarrollarse una adecuada gestión, es probable que se produzca un incremento en el tiempo y los costos dentro de las organizaciones.

Actualmente, es innegable la dependencia de las TIC por parte de ciertos grupos, lo que incrementa las incidencias e impulsan a la creación de sistemas eficaces que brinden soporte a dichos incidentes; intentando así, ser más competitivos para garantizar la durabilidad del mercado.

Eurydice (2000, pág169), señala que a nivel mundial se ha producido un incremento en la implementación de las TIC en diversas áreas, incluyendo las pertenecientes al ambiente educativo; aumentando así, la ocurrencia de incidentes relativos a la correcta funcionalidad de las mismas. A nivel nacional, diversas organizaciones estiman que el crecimiento de los medios tecnológicos es considerable, lo cual es percibido por el sector comercial como un hecho positivo.

En este mismo sentido, el aumento de la cantidad de estudiantes se ha hecho notable, teniendo un reto en relación a los desafíos en términos de la atención de los usuarios de equipos informáticos.

A partir de estos planteamientos, la presente investigación fue realizada en el Colegio Pestalozzi, institución de carácter privado que se encuentra ubicada en la Av. Ricardo Palma, N° 1450, Urb. La Aurora, Miraflores. Cabe destacar, que este instituto educativo cuenta con una trayectoria de más de 77 años en el sector educativo, abarcando las diversas etapas de la educación, desde el Pre kínder, Kínder, Primaria y Secundaria, inclusive con entrada al Bachillerato Internacional. Dicha institución recibe aportes materiales y académicos del Gobierno Central de Suiza y del Distrito Suizo del Cantón de Turgovia. Con el propósito de materializar su visión, tienen la intención de proporcionar al alumno

una educación integral que les permita desarrollar sus habilidades cognitivas, sociales y físicas, basados en un entorno donde el alumno pueda avanzar progresivamente en siete habilidades: autoconciencia y autocontrol para el éxito de la comunicación progresa el pensamiento crítico de la creatividad, la procedencia de la visión global y la capacidad de comunicación, así como el manejo de varios idiomas como el alemán, francés, inglés y español. Por otro lado, se encuentra establecido el monitoreo y la evaluación a cargo de instituciones reconocidas que son externas a la escuela Pestalozzi, para garantizar el éxito académico.

La institución cuenta con 185 equipos, de los cuales, 55 están repartidos en las aulas de primaria, secundaria, sala de profesores (Ver Anexo 01), también unos 42 equipos en las áreas de dirección, oficinas de administración, psicología, secretaría, deporte, TIC, enfermería, materiales e impresiones; de esta manera, tanto profesores como el personal, cuentan con usuario, lo que les permite ingresar a los servicios integrados en la red y hacer uso de registros, registro de calificaciones y asistencias, información personal de estudiantes y sus representantes, así como de temas médicos (Ver Anexo 02).

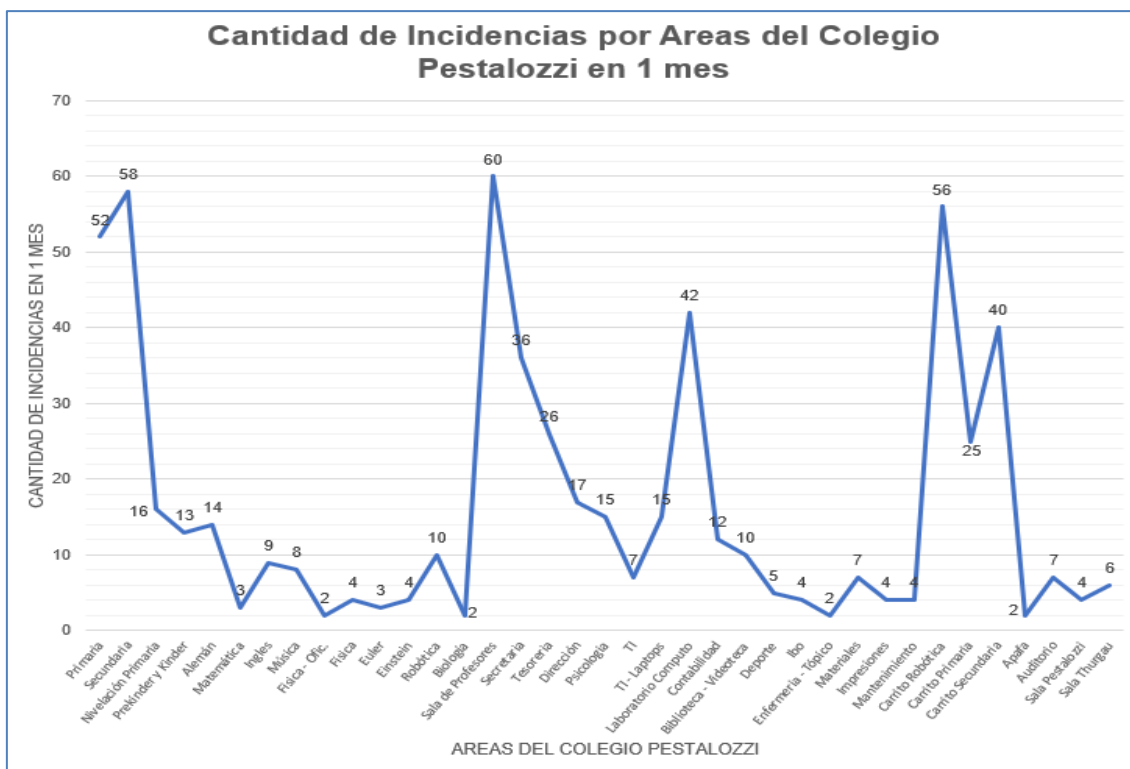


Figura N° 1. Diagrama de Cantidad de Incidencias por Áreas del Colegio Pestalozzi en 1 mes

Por otra parte, los docentes reciben soporte para el uso de los equipos de computación y celulares, configuración de emails, inconvenientes con cuentas de usuario, desbloqueo de contraseñas, dificultades con los dispositivos durante la clase, complicaciones con la carga de notas y asistencias, trabas en la proyección, problemas con el audio, reproducción de archivos multimedia, dificultades en cambio de formato de archivos, grabado y edición de discos, requerimiento de proporción de matrícula por aula y segmentos escolares, conexión a la señal wifi, instalación de aplicativos para su uso durante clases, facilitación de equipos portátiles a profesores y estudiantes, apoyo técnico en el auditorio para actividades de la institución con padres familia, capacitación y reuniones de profesores, apoyo en biblioteca para el uso de equipos por parte de los alumnos, apoyo técnico en laboratorios de computación debido a problemas con el software de control Net Support (Ver Anexo 05).

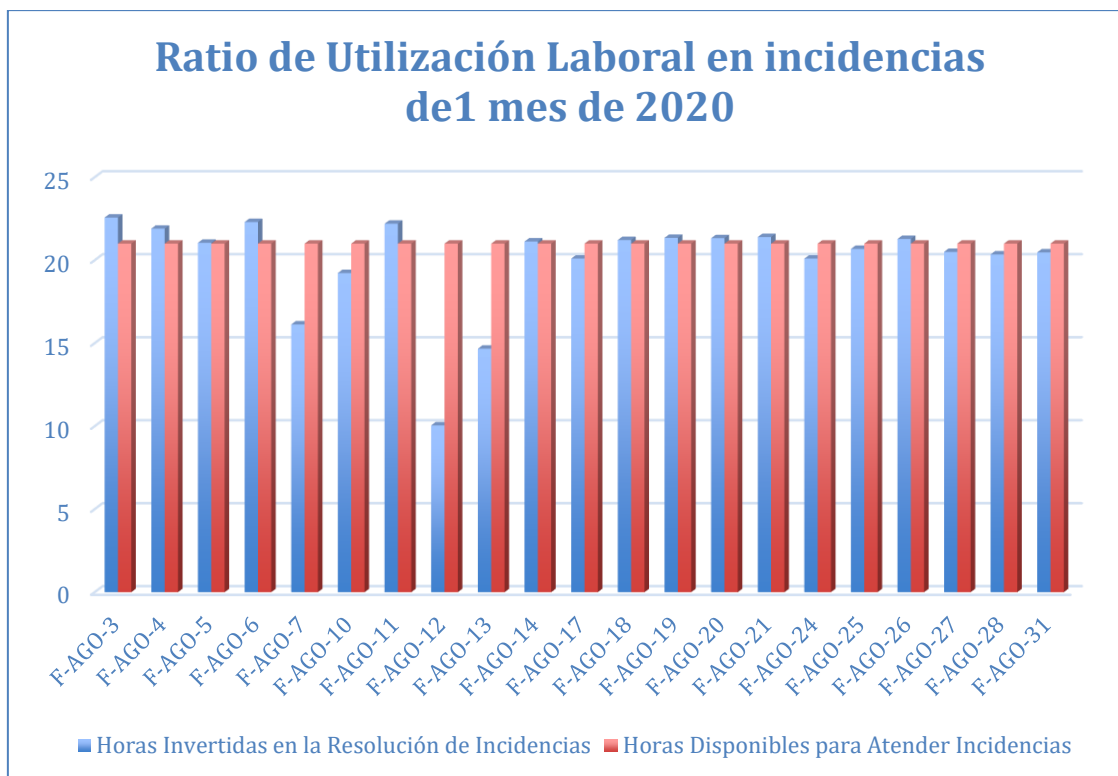


Figura N° 2. Diagrama de Ratio de Utilización Laboral de 1 mes de 2020

Las actividades del departamento de Soporte TI describen las funciones como colaboradores de la Institución, en el archivo MOF (Ver Anexo 04).

En este sentido, la manera como se reporta actualmente un requerimiento de asistencia del departamento TI en la institución, para la atención de las incidencias, incluye las siguientes alternativas:

- Email corporativo.
- Llamado telefónico al departamento de TI.
- Llamado vía celular al personal de TI.
- Empleando mensajes de WhatsApp a los colaboradores de TI.
- Solicitando a un estudiante que avise al departamento de TI sobre el problema.
- Se registran los requerimientos atendidos y los que se encuentran por atender, los archivos de registro los hace manualmente cada usuario del área.
- El personal de soporte TI intenta establecer un procedimiento estándar para la atención en su área, a pesar de ello, es normal que el profesor envíe mensaje o llame al personal al que considere de mayor confianza.

En este proceso, las incidencias se registran de forma manual en un libro (archivo) de Excel; sin embargo, dado el volumen de asistencias llevadas a cabo, es normal que el personal de soporte en TI, no realice efectivamente todos los registros de la asistencia brindada al personal (Ver Anexo 03).

Es así que, el área de TI carece de información estadística, donde se especifique el volumen de incidencias, el número de asistencia por usuario solicitó, el porcentaje de fallas de equipos, el porcentaje de fallas de aplicaciones con licencia, el porcentaje de fallas de aplicaciones de servicios propios de la institución, entre otros indicadores.

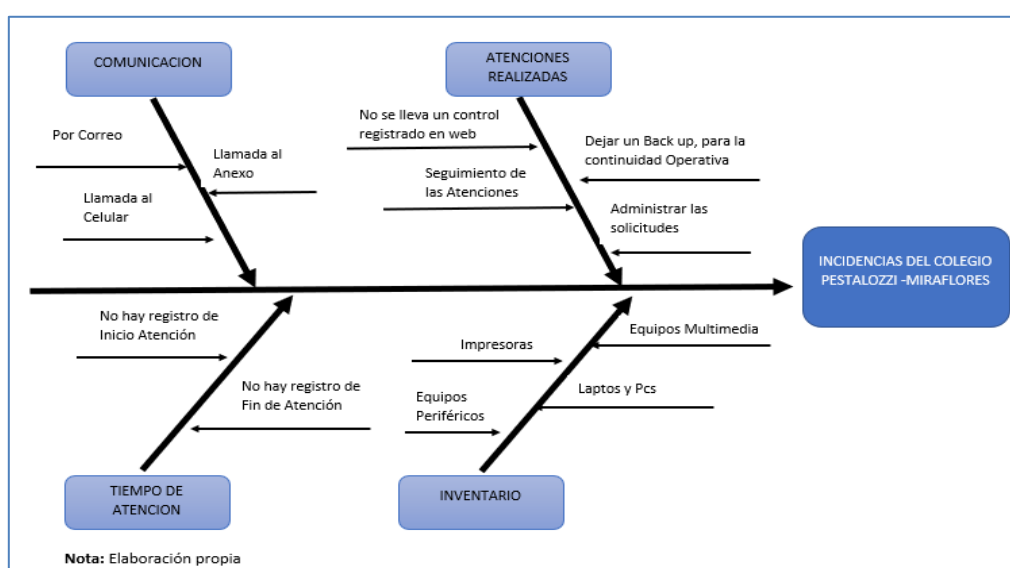


Figura N° 3. Diagrama de Ishikawa de las Incidencias en el Colegio Pestalozzi

Es así, como bajo este contexto, se implementó la herramienta Web para gestionar las Incidencias en el Colegio Suizo Pestalozzi – Miraflores, orientado a la optimización y la automatización de los procesos; interfaz web que permitiría a los usuarios tramitar sus requerimientos para la revisión de los equipos, de esta manera quedará registrado en una base de datos.

Esta interfaz web permitirá al área de TI disponer de una plataforma con toda la información de las incidencias reportadas por áreas, especificando el número de requerimientos, gráficas de reportes por día, tiempo en el cual se logró resolver el requerimiento, índice de incidencias solventadas, número de incidencias no resueltas, además de las que demandan de atención externa por parte de la marca del equipo.

De acuerdo a los planteamientos realizados acerca de las circunstancias del Colegio Pestalozzi, fue identificado el respectivo **problema general**: ¿Cómo influye un sistema web en la gestión de incidencias en el Colegio Suizo Pestalozzi - Miraflores? De igual modo, se han considerado los respectivos **problemas específicos**: **PE1)** ¿En qué medida el uso de un Sistema Web influye en el ratio de resolución de Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi - Miraflores? **PE2)** ¿En qué medida el uso de un Sistema Web influye en el ratio de utilización laboral en Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi - Miraflores? En consecuencia, se plantea el **objetivo general**: Determinar la influencia del Sistema Web para la Gestión de Incidencias en el Colegio Suizo Pestalozzi - Miraflores y como **objetivos específicos**: **OE1)** Determinar la influencia del Sistema Web en el ratio de resolución de Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi – Miraflores. **OE2)** Determinar la influencia del Sistema Web en el ratio de utilización laboral en Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi - Miraflores.

En función de los objetivos expuestos, se define la siguiente **Hipótesis general**: El Sistema Web mejorará la gestión de incidencias en el Colegio Suizo Pestalozzi - Miraflores y como **Hipótesis específicas**: **HE 1)** El Sistema web aumenta el ratio de resolución de Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi - Miraflores. **HE 2)** El Sistema web reduce el ratio de utilización laboral en Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi - Miraflores.

Según el criterio de la **Justificación social** y conforme a lo expuesto por Valdez (2006, pág. 151), esta época actual se caracteriza por un redimensionamiento de la información, por lo que las comunidades más competitivas serán precisamente aquellas que se desarrollen en un contexto que permita el acceso a la información; especialmente, con el apoyo de las TIC, por lo que en su conjunto representan un elemento fundamental de cualquier sociedad en la era actual.

Por tanto, al implementar la propuesta en la institución, se logró tener a disposición un conjunto de información clave, como cuadros estadísticos en

donde se muestran datos sobre las incidencias, en el cual se muestran los estados de los equipos a nivel general, los lapsos de respuesta ante los diversos requerimientos. Asimismo, es necesario destacar que el aumento de equipos produjo el incremento de las incidencias, lo cual dificultó desarrollar la gestión de los mismos, por ello, la implementación del Sistema permitirá incrementar la eficiencia del personal de TI, ante los usuarios que lo requieran.

Respecto a la **Justificación Institucional**, según el criterio de Laudon, K y Laudon, J. (2012, pág. 129), el estudio se realizó en el departamento de Sistemas del Colegio Suizo Pestalozzi-Miraflores, implementando acciones para el adecuado trabajo de la gestión de incidencias, lo cual se evidenciaría con las mejoras dentro de este proceso, reflejado en un incremento de la velocidad de la información.

En cuanto a la **Justificación Tecnológica**, es necesario destacar que la implementación del saber científico en la resolución de las necesidades humanas específicas, es la base de lo que se conoce como tecnología; incluso, el desarrollo de la humanidad ha sido gracias a la vinculación entre el pensamiento racional y la vida cotidiana y el esfuerzo constante para desarrollar las actividades diarias con los recursos disponibles. Es así como el individuo modifica su entorno con la ayuda de la tecnología y, también, ésta despliega su poder para transformar al hombre (Gil, Rosas y Olmeda, 2014, pág. 36).

## II. MARCO TEÓRICO

El desarrollo del estudio se apoyó en **antecedentes nacionales**, como la investigación de Balladares (2018), titulada “Sistema web para la gestión de incidencias en la empresa Business Soft S.R.L.”, cuyo propósito consistió en la de aplicación de técnicas Web en la Gestión de efecto en consistió en resolver el poder de una Técnica web en la Gestión de efecto en la Empresa Businesssoft S.R.L, empleando procedimientos SCRUM, donde fueron analizados ciertos indicadores, entre estos, el grado de incidencia y de reincidencia. La población estuvo conformada por los 25 reportes obtenidos de los dos indicadores, conforme a los datos de información efectuada de un pre-test de indagación concretada; para la recolección de la información fueron empleados los registros de atención del software. De esta manera, se determinó que la instauración del Sistema web va a intervenir en el desarrollo de la gestión de incidencias, concurriendo efectivamente al trámite de las solicitudes de los usuarios de forma rápida y eficaz.

También, Sandoval (2018) en su investigación sobre un “Sistema web para la gestión de incidencias en la institución educativa Innova Schools Sede Los Olivos”, y cuyo objetivo radicaba en establecer la influencia de un sistema web para el procedimiento de incidentes en la organización educativa Sede de Innova Schools Los Olivos. El estudio utilizó una metodología SCRUM, visto que era apropiado para las obligaciones y etapas del plan; igualmente, por ayudar a superar la resistencia al cambio. La aplicación de la herramienta web conllevó al aumento de la proporción de cantidades resueltas en el primer nivel de 70.10%



al 90.05%, y la proporción de incidentes reabiertos se redujo del 26.66% al 10.15%, por lo que estas cifras permitieron determinar que la herramienta mejoró la tramitación de los incidentes.

Mientras que López (2018), llevo a cabo el estudio denominado “Sistema web para la gestión de incidentes por parte de Gaval Soluciones S.A.C.”, donde este proceso fue identificado como uno de los aspectos claves de la automatización. La investigación tenía la intención de examinar la incidencia de una herramienta para la gestión de incidentes en Gaval Soluciones S.A.C. La metodología que se usó en la elaboración de la herramienta web estuvo sustentada en el procedimiento SCRUM. Para el desarrollo de la herramienta Web se empleó la programación PHP y el gestor de bases de datos PostgreSQL. El estudio resultó ser de tipo aplicado, con un diseño preexperimental. La población estuvo compuesta por los 20 informes de incidentes correspondientes a los dos indicadores implementados, los cuales estuvieron relacionados al nivel de incidencias atendidas y al grado de reincidencia. De esta manera se concluyó que la herramienta web mejoraría la gestión de incidentes y sería más rápido y eficiente para atender los requisitos del usuario.

Seguidamente, el proyecto de investigación de Chipulina (2018) titulado “Sistema Web para la Gestión de Incidencias en la Empresa Consulit S.A.C.”. El estudio fue realizado para determinar la influencia que tiene la aplicación de una herramienta web en la gestión de incidencia en la empresa consulit S.A.C, abarcando la estimación del ratio de resolución de incidencias y el ratio de incidencias reabiertas como indicadores explicativos del trabajo. La metodología implementada en el desarrollo del sistema estuvo basada en la SCRUM, apropiada en este tipo de proyectos; asimismo se usó el lenguaje PHP, con una base de datos María DB. Los resultados evidenciaron que antes de la implementación del sistema web; el ratio de resolución de incidencias en el pre-test registró una media 44.67% y el ratio de incidencias reabiertas en el pre-test arrojó una media de 73.09%, mientras que posterior a la instauración de dicho sistema; el ratio de resolución de incidencias en el pre-test registró una media alcanzó una media de 84.77% y el ratio de incidencias reabiertas en el pre-test

arrojó una media de 63.80%, con lo cual se obtuvo un incremento de 40% para el primero de los índices y una disminución de 63.80% para el segundo.

Adicionalmente, Vargas (2018) en el desarrollo del estudio denominado un “Sistema web para la gestión de incidencias en la empresa New Global SAC”,. El propósito del estudio consistía en determinar la influencia de un sistema web de gestión de incidentes en la empresa New GlobalSAC. En el desarrollo del sistema web fue empleado el método SCRUM, por ser el de mejor acoplamiento a los requerimientos y fases del proyecto, así como la rapidez para solventar incidencias, generando soluciones más expeditas. Se aplicó el diseño de preexperimental y el método cuantitativo, con una población de 299 incidentes reportados por usuarios de acuerdo a 24 fichas de registro, mientras que la muestra se basó en 168 incidentes, empleando un muestreo aleatorio de probabilidad simple. En cuanto a las técnicas de recolección de información utilizadas, se tiene al registro con una herramienta de tipo formulario de registro, el cual fue validado por expertos. La implementación del sistema web permitió que el índice de servicio experimentara un incremento de 40,48% al 66,07%. Las conclusiones determinaron que la herramienta web influye en la optimización de la gestión de incidencias.

Asimismo, Barrantes (2018), en su estudio “Sistema web para la gestión de incidencias informáticas en la empresa South Express Cargo Perú S.A.C.”. Como consecuencia de la pequeña organización y el control con que se maneja la atención en las incidencias informáticas, el trabajo de los usuarios se ve seriamente afectado, incluso la continuidad de la empresa está en riesgo. El propósito del estudio fue determinar cómo un sistema Web podría afectar la gestión de incidencias para los equipos informáticos de una empresa como South Express S.A.C. En la elaboración del sistema informático se empleó el método SCRUM, porque es el método flexible que prioriza los resultados sobre la documentación y sobre todo porque involucra mucho al usuario en el proceso desde la etapa inicial. El diseño del estudio fue pre-experimental y el método fue cuantitativo, con una población de 180 solicitudes de incidentes y muestra delimitada a 63 solicitudes estratificadas por días en 24 registros, usando un muestreo probabilístico aleatorio simple, mientras que la técnica empleada se

basó en el registro y el instrumento fue el formulario de registro, debidamente validado por expertos. La implementación de la propuesta consiguió mejorar la atención de incidencias del 64,81% al 86,88%, de igual forma la proporción de incidencias informáticas aumentó del 44,60% al 100%. Los resultados anteriores permiten concluir que propuesta mejoró el manejo de incidencias informáticas de equipos para los equipos informáticos de la empresa South Express Cargo Perú S.A.C.

Por otra parte, se dispone del trabajo desarrollado por Torres (2018) con el título “Aplicación web para la gestión de incidencias en el soporte de Tecnología de Información a los clientes internos de la compañía PRONACA S.A.”, el cual se enfocó en proponer un sistema web para la gestión, el procesamiento y respuesta a las incidencias notificadas por los clientes en la empresa PRONACA S.A. El desarrollo del sistema estuvo sustentado en plataformas LAMP (Linux, Apache, Mysql, PHP) presentado por medio de una aplicación web de nombre PronacaDesk para la Gestión de incidencias en la entidad. La metodología implementada para el desarrollo de la aplicación estuvo a cargo del método de cascada iterativo, proposición que se justifica, porque incorpora un prototipo funcional, el cual ofrece a los usuarios una revisión rápida de las posibles incidencias y soluciones que se pudieran presentar para los casos de uso planteados. En conclusión, se diseñó un sistema web denominado PronacaDesk, el cual fue desarrollado con PHP Generator for MySQL Professional, con la intención de ofrecer a PRONACA un servicio de soporte al usuario completamente apegado a las mejores prácticas para la gestión de incidencias y que pudiera contar con tecnología de punta a nivel de procesos.

En este mismo contexto, se presentan los **antecedentes internacionales** que dan soporte al presente estudio, Cortez (2019) en su estudio titulado “Implementación de un proceso de gestión de incidentes caso práctico Empresa de Agua Potable y Alcantarillado EAPA San Mateo”, señala como propósito principal implementar un proceso de administración de incidentes de los servicios del Departamento de Sistemas de la Compañía de Agua Potable EAPA de Esmeraldas, a través de un software pre-existente en el mercado. Por ello, fue empleada una metodología con base en una investigación descriptiva, la cual

hizo posible una exploración intensiva acerca de la temática planteada, especificando sus propiedades y aspectos más resaltantes; asimismo, se usó el método cuantitativo, aplicando encuestas, las cuales se tabularon y presentaron en tablas para realizar el análisis. En cuanto a las conclusiones del estudio, se evidenció que la herramienta propuesta, requirió de mejoras tecnológicas y organizacionales, específicamente en el departamento de TI, adoptando un mecanismo preciso para la notificación de deterioros; asimismo, se recomendó capacitar a los equipos de trabajo para incrementar el nivel de especialización ITIL, para lograr un entorno en el que las buenas prácticas sean aplicadas y prestar una esmerada atención a los usuarios.

Del mismo modo, Ricaurte (2018) en su estudio titulado “Sistema web para la gestión de incidencias y eventos en activos informáticos, biomédicos e infraestructura para Perfect Body Medical Center”. El objetivo principal de esta investigación radicó en desarrollar un sistema web para el seguimiento y manejo de incidencias en activos a la empresa Perfect Body Medical Center. Para la elaboración de este estudio se utilizaron técnicas de recolección de registro, sustentados en hoja de anotaciones para cada uno de los activos, esta actividad fue llevada por cada uno de los empleados que tenían recurso a su disposición. Los resultados de esta investigación permitieron obtener un sistema web capaz de mejorar los lapsos de contestación y por otra parte que pueda controlar y rastrear los medios de propiedad de IPS, con lo cual se beneficiaría a las distintas áreas de la empresa en el soporte de recursos de cálculo biomédico y de infraestructura.

Asimismo, Alfonso (2016) en su investigación titulada “Desarrollo de un Sistema Web Orientado a una Mesa de Servicio para el Registro, Gestión y Control de Incidencias Técnicas”, en la cual se propone un estudio, diseño y mejora de los instrumentos web, orientadas a ofrecer soporte al trabajo normalizado, de tal manera que este procedimiento se llegue a desarrollar e implementar centralizadamente en cualquier organización, teniendo la finalidad de ahorrar tiempos en el procedimientos de registro, servicio y búsqueda de aspectos técnicos, trámites y requerimientos generados por los clientes del procedimiento. Al respecto, se realizó una indagación acerca de los prototipos

explicativos, para así facilitar las solicitudes técnicas de la administración de IT. Las técnicas de recolección de información estuvieron dadas por la observación y el registro, sustentados en anotaciones que se realizaron de la revisión de la información contenida en documentos, encuestas, diagramas de funciones, etc., para determinar los datos más importantes referentes a los desarrollos de la interfaz de usuario, así como del banco de datos relacional. Las conclusiones permitieron visualizar los efectos en el procesamiento, disposición, consolidación, estudio y presentación de los datos referentes a los incidentes del sistema web, así como las solicitudes atendidas en la mesa de servicio.

Asimismo, Mona y Uribe (2015), en su estudio “Sistema de gestión de incidentes de seguridad informática para Corvette”, presenta una herramienta para manejar sucesos de protección de la información que permite el registro, la categorización, la notificación, la escalada y la respuesta rápida a eventos que ocurrieron o que pueden ocurrir en la empresa, y cuenta como un punto de partida para el estándar ISO / IEC 27035, además de una estrategia de avance que admita el respaldo de la herramienta de gestión de incidentes de seguridad. De igual forma, se plantea un mecanismo para categorizar y ponderar los incidentes de seguridad informática, clasificando los mismos según su tipo o medio, como incidentes de ingeniería humana y de procedimiento, y según su origen, como incidentes privados o externos, otorgándoles una prioridad de acuerdo al impacto producido, para así ubicarlos y gestionar de forma efectiva. Las conclusiones del estudio plantean la instauración de un mecanismo de gestión de incidentes de seguridad certificada en ITIL, que pueda optimizar los procesos en la empresa.

Por último, Riera y Zambrano (2015) en su investigación titulada “Propuesta de Procesos para la Fase de Operación de los Servicios de Tecnología, para Bio Agencia”, presentada con el objetivo de permitir a la empresa llevar a cabo sus operaciones de una manera más competente y eficiente, ahorrando tiempo y recursos. Para este propósito, se utilizó la "Biblioteca de Infraestructura de Tecnología de la Información (ITIL)" junto con ISO 9000 e ISO 20000 como elementos optimizadores del rendimiento de los procesos operativos de los servicios de TI. Entre las conclusiones del estudio, se evidencia que la propuesta puede mejorar el desempeño, ya que el prototipo realizado lograría optimizar la

eficiencia de las actividades del proceso, lo que permitirá al gerente verificar la evolución de las actividades, y de esta forma alcanzar los objetivos propuestos.

En términos de lenguaje de programación, se utilizará **PHP**, definida por Cobo (2005, pág. 23) como un lenguaje de programación interpretado por el servidor que sale del flujo llamado Open Source (código abierto), caracterizado por ser potente, versátil, robusto y tener la capacidad de entender. Además de tener un personaje multiplataforma. Además, se utilizará el administrador de base de datos **MySQL**, el cual Arias (2015, pág. 19) describe como un administrador manipulable o abierto, con muchas funciones para personas nuevas y más práctico que otros sistemas con propiedades similares.

En cuanto a la **Metodología Scrum**, SCRUMstudy (2017, pág. 2 ) afirma que Scrum es un método simple y flexible que dura entre 1 y 8 semanas y que hacer un cambio no perjudica los otros procesos porque no es secuencial; asimismo, menciona las siguientes fases: **a) Inicio:** al progreso, las actividades del proyecto son definidas por el equipo Scrum y Scrum Master; **b) planificación y evaluación:** historias de usuario están definidas y sus estimaciones también definen tareas; **c) implementación:** Las entregas de Sprint se presentan a través de las reuniones diarias del equipo Scrum; **d) revisión y reflexión:** Las entregas se presentan al propietario del producto en reuniones de revisión de sprint; **e) lanzamiento:** este es el proceso final donde se realizan todas las entregas al usuario y documentan todas las lecciones aprendidas.

También Castelo (2019, pág. 45), señala que es la herramienta más empleada para desarrollar software; se basa en la gestión de proyectos, especialmente, los de plazos de entrega breves. En SCRUM, el trabajo se organiza en bloques temporales o Sprint (que pueden variar de una a cuatro semanas). Operando en pequeños grupos de 5-9 personas, aunque pueden convertirse en equipos más grandes.

**Sprint:** es la unidad básica de trabajo. Dura entre una y cuatro semanas. Su función es fraccionar el proyecto en unidades pequeñas y viables para que se pueda realizar un seguimiento del progreso y poder entregar software utilizable al cliente en todo momento. Todos los Sprint comienzan con una reunión en la que se define el **Sprint Backlog**.

**Roles es Scrum:** hay 3 roles los cuales son, **Product Owner**, siendo el garante fundamental de la gestión de proyecto.

**Scrum master:** responsable de la mayor eficiencia de la aplicación de Scrum.

**Equipo de desarrollo:** son los responsables de consignar los parámetros determinados para cada uno de los diferentes Sprint.

**Documentos:** destaca el **Product backlog** - que suministra una descripción general del proyecto. A partir de lo cual se definen los Sprint. Todos pueden contribuir con su finalidad, aunque, únicamente el Product Owner tiene la capacidad de ejecutar las transformaciones. **Sprint backlog** → Documento que detalla las actividades a ejecutar en un breve lapso de tiempo.

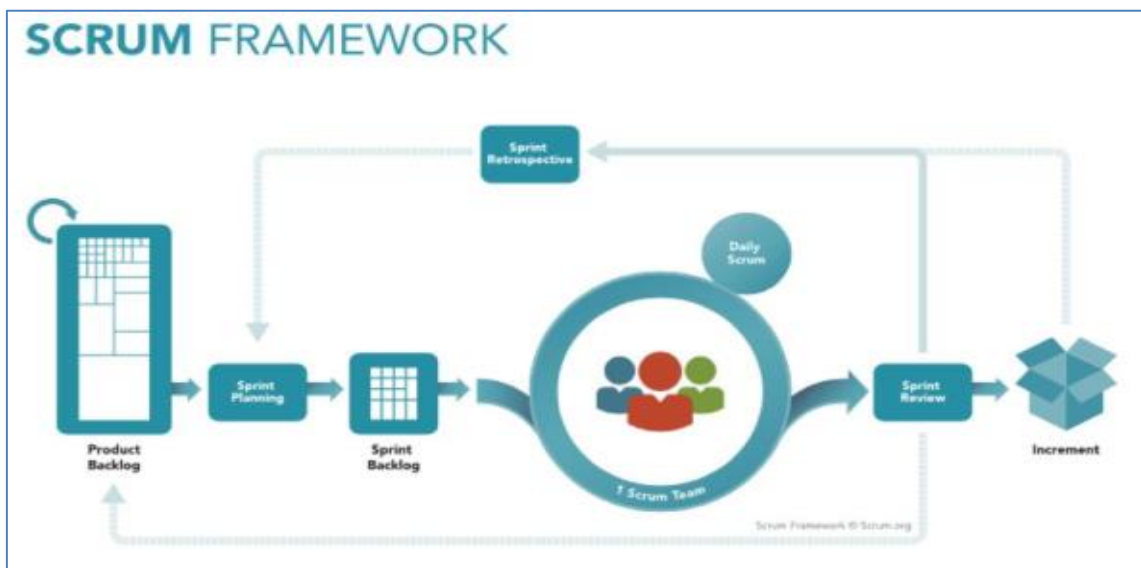


Figura N° 4. Metodología Ágil SCRUM

Así también, Díaz (2014, pág. 45) indica que la **Metodología XP** es un método simple para procesos de software y se prepara en poco tiempo. Las tareas más avanzadas se le muestran al cliente, pues todos los cambios se pueden hacer de inmediato, incluso el proyecto terminado. Esto consta de varias fases que son planificación de exploración, iteración, mantenimiento de producción y muerte del proyecto.

En la **Metodología Kanban**, por Project Management Institute (2011, pág.3), indica que el método Kanban se usa y aplica en diversos ambientes y admite al cliente un flujo continuo de trabajo y valor. El método Kanban es menos

prescriptivo que algunos enfoques ágiles y, por lo tanto, menos disruptivo. para comenzar a implementar, ya que es el método original de "comenzar donde está". Las organizaciones pueden comenzar a aplicar los métodos Kanban con relativa facilidad y progresar hacia la implementación completa del método si eso es lo que consideran necesario o apropiado.

A diferencia de la mayoría de los enfoques ágiles, el Método Kanban no prescribe el uso de iteraciones de timeboxed. Las iteraciones se pueden usar dentro del Método Kanban, pero el principio de extraer elementos individuales a través del proceso continuamente y limitar el trabajo en progreso para optimizar el flujo siempre debe permanecer intacto. El Método Kanban puede usarse mejor cuando un equipo u organización necesita las siguientes condiciones:

- Flexibilidad: Los equipos generalmente no están sujetos a cajas de tiempo y trabajarán en el elemento de mayor prioridad en la flexibilidad.
- Centrarse en la entrega continua: Los equipos se centran en el flujo de trabajo a través del sistema hasta su finalización y no comenzar un nuevo trabajo hasta que se complete el trabajo en progreso.
- Mayor productividad y calidad: La productividad y la calidad aumentan al limitar el trabajo en progreso.
- Eficiencia incrementada. Verificando cada tarea para actividades de valor añadido o sin valor añadido y eliminando estas últimas.
- Enfoque de los miembros del equipo. El trabajo en progreso limitado le permite al equipo concentrarse en el trabajo actual.
- Variabilidad en la carga de trabajo. Cuando hay imprevisibilidad en la forma en que llega el trabajo, y se hace imposible para los equipos hacer compromisos predecibles; incluso por cortos períodos de tiempo.
- Reducción de residuos. La transparencia hace que los residuos sean visibles para que puedan eliminarse. El Método Kanban se deriva de los principios de pensamiento lean.



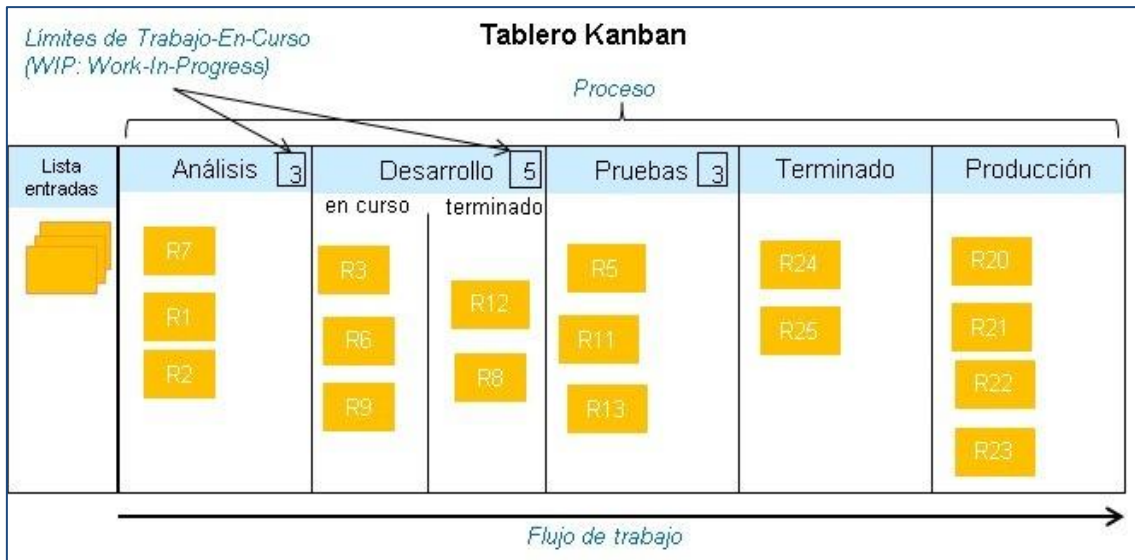


Figura N° 5. Tablero Kanban

En la tabla 1 se comparan los métodos Kanban, XP y Scrum.

Tabla N° 1. Diferencia entre metodologías de desarrollo de software

Criterio	Metodología		
	Scrum	XP	Kanban
<b>Tipo de Iteraciones</b>	Iteraciones de plazo fijo	Iteraciones de plazo variable	Iteraciones plazo fijo o variable
<b>Roles - Facilitador</b>	Scrum Master	Coach, Big Boss	N/A
<b>Roles – Administrador Requerimientos</b>	Product Owner	Cliente	N/A
<b>Roles – Equipo Proyecto</b>	Equipo de Desarrollo	Programador, Tester	N/A
<b>Equipos</b>	Multifuncional	Especializados	Especializados o Multifuncional
<b>Practicas / Reglas</b>	9	12	3
<b>Limitación Work In Progress</b>	Limitación por iteración	Limitación por iteración	Limitación por estado
<b>Incorporación de Tareas</b>	No es posible hasta finalizar el sprint	No es posible hasta terminar la iteración	Es posible, en tanto exista capacidad
<b>Seguimiento de tareas</b>	Gráfico Burn-down	Velocity	Tablero Kanban
<b>Estimación</b>	Obligatoria	Obligatoria	Opcional

Fuente: Elaboración propia

Asimismo, se presenta a continuación, las comparaciones entre 2 metodologías:

**Kanban vs Scrum:** La metodología Scrum supera en reglas a Kanban, lo que impide definir roles y equipos, además, no dispone de una ordenación consecuente de reunión. Kanban no establece interacciones, pero es posible insertarlas si se requieren. Los métodos de verificación del proceso Kanban lo hacen perfecto para dispositivos en el sitio que trabajan dentro de un marco temporal de objetos susceptible a modificaciones constantes (por ejemplo, Kanban es empleado a menudo por los grupos de apoyo). Pero el equipo Scrum a menudo adopta el tablero Kanban en forma de entablado de tareas y se usa para rastrear el progreso durante un sprint. El término de la regla para Kanban Work in Progress también es apropiado para dispositivos con capacidades limitadas o si se requiere la contribución de cada individuo para cada artículo. Esto puede, por ejemplo, aplicarse a un equipo de comunicación dentro de una empresa de gran tamaño. No obstante, Scrum restringe la carga de trabajo para cada sprint, la carga de tarea está determinada por la apreciación sobre el tamaño de cada historia (en puntos) y acordada por el grupo de Scrum en cada reunión de programación. Por otro lado, un equipo Kanban indaga sobre la duración del ciclo y perfecciona los lapsos de entrega, para que sean lo más pequeños y previsibles posible, un equipo Scrum quiere mejorar su rendimiento mediante carreras cortas consecutivas y mejorar los "puntos de velocidad" del equipo (completado en una estimación de golpe corto). Esto ciertamente hace que Scrum sea más adecuado para escalar: ciertamente, es más sencillo y previsible, lo que generalmente es reconfortante para empresas de gran tamaño.

**Scrum vs XP:** los grupos y las reuniones están prácticamente congelados en Scrum, en tanto, lo relativo a cómo se hace efectivamente o el deber, lo hacen los mismos equipos. XP dispone de un clan de destrezas esenciales que pueden tener valoraciones muy altas para el novel ágil. De esta manera, Scrum es una herramienta que se preocupa fundamental por el rendimiento, en tanto, XP se centra en la ingeniería. Aunque el beneficio que las prácticas de XP aportan es apodíctico, numerosas empresas que usan Scrum están adoptando el cambio y la refactorización impulsados por pares, dado que las prácticas de mejora de la casta aceleran el recurso de tirada y / o reducen la apremio de retornar. Además,

de las iteraciones más cortas, otros utensilios destacados que marcan diferencia de XP con respecto a Scrum son: los dispositivos de XP actúan en un estricto nivel de preferencia, en tanto, Scrum no se ocupa precisamente de todos los elementos de orden preferencia, mientras que en los dispositivos de sprint XP es posible incorporar elementos de trabajo a una iteración y modificar componentes de tamaño similar (mientras que no inicien) en caso de que el cliente resuelve una prelación distinta. En función de las semejanzas, el factor del cliente en XP es similar a la del propietario del artículo en Scrum, dado que permiten anotar registros de usuarios. Las prioriza y asimismo se encuentran dispuestas para los desarrolladores, sin embargo, no se encuentran efectivamente definidas.

- Scrum y XP requieren una reunión permanente todos los días.
- Si bien ambos enfatizan la importancia de la colocación, solo XP crea deal-breakerie.

**Conclusión:** Los tres métodos siguen los ideales contemplados en el manifiesto Ágil, cuyo propósito es ofrecer a los clientes el máximo valor en el período utilizable. Los contrastes entre ambos son el efecto de tratar de conservar principios ágiles en ambientes absolutamente distintos. Kanban es un procedimiento de gran ventaja para equipos con retrasos continuos en los elementos para acrecentar la eficacia al limitar la cuantía de trabajo en progreso teniendo en cuenta los roles y asignaciones propias. Scrum es conveniente para dispositivos que ocasionalmente o constantemente pasan trabajan juntos en un proyecto o producto, brindando mucho más a la estructura que apoya a los equipos a incrementar la productividad mediante la comunicación y planificación constante, al tiempo que brinda la posibilidad de elegir cómo se deben diseñar las soluciones. XP ofrece otro grado de innovación, centrado en la calidad, al persistir en un grupo de acciones elementales de ingeniería que permiten mantener el código limpio y el software estable.

El presente estudio estuvo basado en el desarrollo de un “Sistema Web para la Gestión de Incidencias en el Colegio Suizo Pestalozzi – Miraflores” bajo la metodología Scrum, visto que con su uso es posible especificar y figurar la secuencia del sistema implementado. De igual forma, contribuye a determinar

los diversos tipos de usuarios de acuerdo a sus roles, responsabilidad y desarrollo y al mismo tiempo va a permitir identificar anticipadamente inseguridades y los niveles de feedback y reúso de elementos. El ambiente que hace posible controlar SCRUM es necesario ajustarlo al contexto de las empresas que servirán para desarrollar el programa.

**SCRUM** se define como una estructura donde las personas pueden manejar problemas de adaptación complejos mientras son productivos y creativos para ofrecer productos finales de alto valor. Scrum también contiene varios elementos que lo hacen sencillo y fácil de entender (Goncalves 2020, pág. 2).

La evolución del SW dará paso a la GI en el Colegio Suizo Pestalozzi - Miraflores, permitiendo a todos los miembros de la institución, involucrarse con el departamento de TI, al involucrarlos con la centralización de las incidencias de forma digital, con lo que se podrá dar estadísticas de solicitudes o requerimientos, para proponer mejora en la atención y el servicio de soporte.

Ahora bien, con respecto a los referentes teóricos vinculados a la **variable independiente, Sistema Web** en términos de Berzal, Cortijo y Cubero (2016, pág. 230), señalan que se basan en aplicaciones cuya interfaz varía de acuerdo con los archivos y textos de sus respectivos protocolos.

En tanto, Taniar y Rayahu (2016, pág. 21) señalan que es una red hipermedia repartida en internet, que dispone de diversos contenidos, como, por ejemplo, videos, imágenes, sonidos de texto y un lenguaje específico. Es así, como un sistema basado en la web ofrece la propiedad de una vía única y exploraciones livianas.

Por otra parte, la **Arquitectura de un Sistema Web**, según Vara y otros (2014, pág. 96), representan modelos más consecutivos e instaurados al momento de edificar aplicaciones, no solo ofrece grado web sino que además provee aplicaciones standalone. Tal es el caso, de la arquitectura basada en un esquema Modelo-Vista-Controlador (MVC), el cual busca descomponer una aplicación en 3 elementos primordiales: el modelo, la vista, y el controlador.

- **Modelo:** constituyen las partes de la aplicación que implementan la lógica para un dominio específico; en modelos pequeños es considerado una división conceptual más que una división física.
- **Vista:** son los componentes que despliegan la interfaz de usuario (UI-User Interface), dicha interfaz es construida por el consenso que se produce en el modelo de datos y en las condiciones de la aplicación web es elaborado por el grupo de páginas web que presentan y recogen la información del cliente.
- **Controlador:** representan los elementos que manejan la relación con el cliente, trabajan con el modelo y eligen cuál va a ser la vista idónea a desplegar para comprobar la información.

El modelo (MVC) producirá aplicaciones que intentarán separar los distintos puntos de la aplicación, es este caso constituidos por la lógica de entradas, la lógica de comercio y la lógica de la interfaz; estableciendo un acoplamiento de pequeñas dimensiones entre la totalidad de las partes.

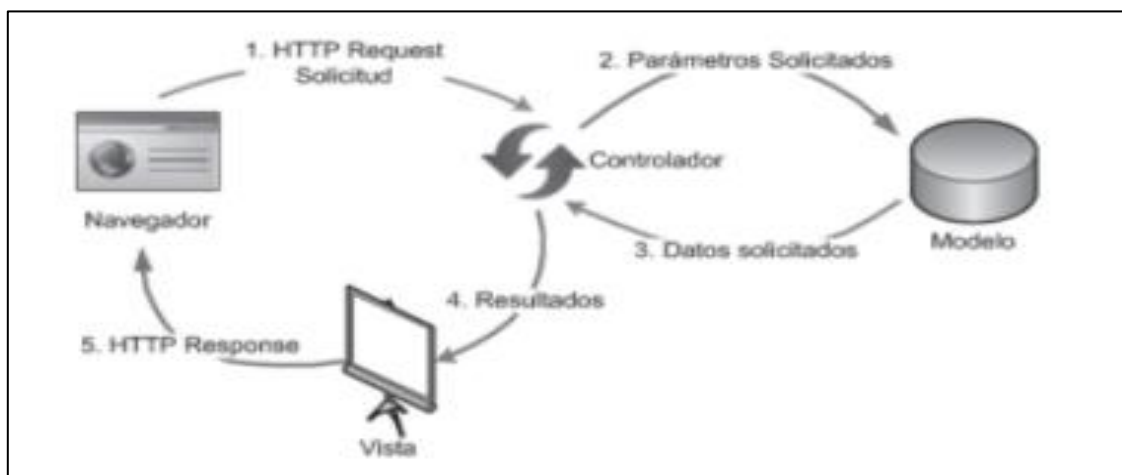


Figura N° 6. Modelo Vista Controlador (MVC)

De esta manera, un Sistema Web tiene ventajas y desventajas, como se menciona en la tabla 2.

Tabla N° 2. Ventajas y Desventajas Sistema Web

VENTAJAS	DESVENTAJAS
Ingreso total al mecanismo en software y hardware.	Diversos mecanismos de capacidades de lenguaje para los diversos entornos de funcionamiento.
Excelente usabilidad para el cliente.	Suelen resultar costosos de elaborar.
La aplicación se actualiza constantemente	El carácter del usuario no se puede reutilizar entre los distintos entornos.
Varios usuarios al mismo tiempo.	Ausencia de parámetros

Fuente: Elaboración propia

De esta forma, para tener un sustento teórico del estudio, se muestran referentes teóricos de la **Variable Dependiente Gestión de Incidencia**, Castelo (2019) expresa que la finalidad de la gestión de incidentes es restaurar el funcionamiento adecuado del servicio en el menor tiempo. En este sentido, una incidencia es toda perturbación que impida la calidad o el servicio acordado; mientras que el proceso de gestión de incidentes cumple un orden tal como se muestra a continuación.

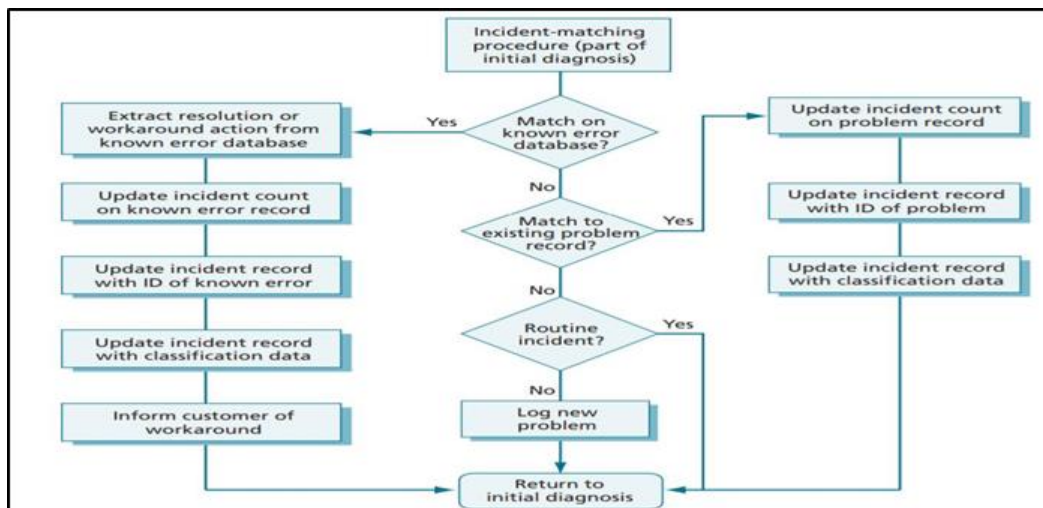


Figura N° 7. Gestión de incidencias según ITIL V3

Asimismo, Steinberg (2013, pág. 88) destaca que la gestión de incidentes abarca todo tipo de incidentes; como, por ejemplo, interrupciones, dudas o consultas de los clientes (usualmente mediante notificaciones realizadas vía

llamadas telefónicas al centro de atención), fallas observadas por el equipo técnico o las que son detectadas de manera automática por los dispositivos de control permanente. **ITIL** define un incidente como un tiempo de inactividad no planificado que produce una disminución en los estándares de calidad de los servicios de TI, al igual que la falla de un elemento de programación, sin que llegue a afectar el servicio. El propósito básico de esta gestión es el inmediato retorno a la normalidad y reducir los efectos sobre las operaciones comerciales. La gestión de incidentes incluye todos los eventos que pueden obstaculizar las actividades, abarcando reportes de los usuarios directamente al Centro de Atención o mediante sistemas automáticos que existan; el equipo especializado puede reportar o registrar incidentes, sin embargo, esto no significa que todos los reportes sean incidentes.

De esta manera, los incidentes y las solicitudes de servicio se informan al centro de atención al cliente, a pesar de no ser lo mismo, ya que las solicitudes de servicio no implican una interrupción del servicio, son requerimientos de apoyo, consignación, datos, asesoramiento o solicitud de documentos por los clientes. El valor de esta gestión radica en:

- La posibilidad de solventar los reportes, lo que representa menos interrupciones en el negocio y mayor tiempo de servicio disponible.
- La capacidad de unificar las actividades del área TI con las preferencias del negocio, ya que se logran establecer las prelacións del cliente y asignar recursos estratégicamente.
- La posibilidad de reconocer las posibles mejoras del servicio.

La gestión de incidentes tiene efectos resaltantes en el negocio, por lo que es posible identificar sus aportes con mayor facilidad que en otras áreas de operaciones de servicio. Esto lo convierte en uno de los procesos previamente implementados en proyectos de gestión de servicios, en donde se deben considerar los siguientes aspectos:

- Plazos: deben definirse plazos para cada una de las etapas y elementos como fines de los Acuerdos de Nivel Operativo (OLA) y los contratos de apoyo.
- Modelos de incidentes: es una manera para identificar las diversas actividades que son requeridas para ejecutar adecuadamente un proceso

(específicamente, la atención los incidentes). Esto significa que las incidencias normales, se atienden adecuadamente y en los lapsos definidos.

- Incidencias graves: éstas requieren de un procedimiento diferente y de menor duración y mayor validez. Es necesario definir qué sigue siendo un incidente importante y describir todo el sistema de prioridad para los incidentes.

A veces, un incidente grave se confunde con un problema, pero un incidente siempre será un incidente, puede aumentar su impacto o su prioridad, pero de ningún modo se convertirá en un problema. Un problema es el posible trasfondo de uno o varios incidentes y siempre será una entidad separada. Un incidente no se gestiona hasta que se sabe que existe. Esto se conoce como identificación de incidentes. Desde una perspectiva comercial, la práctica generalmente aceptada es esperar a que un usuario conozca las implicaciones de un incidente y se comunique con el centro de servicio al cliente. De tal forma, que para que pueda activarse el proceso de gestión de incidentes, es necesario que la empresa supervise los aspectos claves que permitan la visualización temprana de los errores reales o potenciales. En muchas oportunidades, los problemas se pueden solucionar antes de que afecten a los usuarios.

Para un mejor procedimiento debe registrarse toda la información de los incidentes, incluida la fecha y hora de los mismo. Esto se conoce como registro de incidentes y se aplica tanto a los incidentes que se reciben mediante el Centro de servicio al usuario, como a los detectados en forma automática por un sistema de monitoreo; esto permitirá tener un registro histórico completo, inclusive si el incidente es transferido a otros grupos de apoyo, toda la información que necesitan estará disponible para ellos. Se debe registrar al menos lo siguiente:

- Número de referencia único.
- Categoría del incidente.
- Nombre / identificador del usuario que registró el incidente.
- Descripción de los síntomas.
- Actividades desarrolladas para solventar el incidente.

Asimismo, es necesario que se utilicen pautas de codificación de incidentes apropiados en la fundamentación de los diversos requerimientos; lo cual será importante luego, para el análisis sobre los tipos y frecuencia de los incidentes,



información necesaria en la gestión de incidentes, de distribuidores y otras diligencias de gestión de servicios de TI.

Ahora bien, al reasignarse un problema, los datos disponibles pueden no estar completos o no ser correctos, por tanto, es conveniente consultar la clasificación de incidencias y actualizarla mientras se cierra la convocatoria. A continuación, se muestra un ejemplo de un problema categorizado: software, aplicación, paquete financiero y sistema de ventas. Al mismo tiempo, existe otro elemento primordial en lo que se refiere al registro de incidentes, y radica precisamente en la identificación con un código de prelación adecuado, ya que el personal de soporte verifica dicho código para establecer la forma de tratar el incidente.

En general, la prioridad de un incidente puede determinarse por la urgencia (qué tan rápido la empresa necesita solventar) y su impacto (número de usuarios que son afectados). Cuando un usuario notifica un incidente al centro de servicio al usuario, el técnico debe procurar hacer un registro con la mayor cantidad de síntomas del incidente como sea posible como diagnóstico inicial, también debe intentar determinar qué salió mal y cómo podría corregirse; en este sentido, los scripts de diagnóstico y la información de errores conocidos son muy útiles en este contexto; de ser posible, el técnico resolverá el incidente de forma inmediata y cerrará el caso, sino es posible, se deberá escalar la incidencia.

En este orden de ideas, las etapas del progreso de control de Incidentes o etapas de un incidente de ITIL V3 son: **1) Localización:** es la fase inicial, donde se identifica el impacto, por lo que debe ubicarse rápidamente, para evitar consecuencias negativas para la empresa; **2) Registro de Incidencia:** cada incidencia se debe registrar de forma separada, este registro deberá incluir: código del solicitante, categoría, premura, prioridad, lapso de la gestión, persona u organización que atiende la incidencia, información de la incidencia, labores desarrolladas para solventar la incidencia, lapso e información de la resolución y del cierre del servicio; **3) Categorización:** esta etapa representa el modelo de la incidencia, el cual se especifica para detallar el flujo dentro de la empresa y la necesidad de pronta resolución. Se debe indicar también el personal asignado para atender la incidencia. Generar medidas concretas para reducir los efectos perjudiciales en la empresa; **4) Priorización:** Usualmente, el detalle de la

incidencia va a señalar la manera en que la misma puede resolverse. Por tanto, es sumamente importante que el instrumento de gestión permita registrar con anticipación en base a reglas, por lo que resulta importante que el equipo de soporte se familiarice con los parámetros que permitan priorizar convenientemente; **5) Diagnóstico Inicial:** Es la fase donde se desarrolla la revisión para corregir y restaurar de inmediato la incidencia y para fortalecer las actividades de servicio; **6) Escalado:** En este paso, la práctica del técnico encargado, se utiliza para evaluar la solución o para asignar esta incidencia al nivel de atención debido a la gravedad de los escenarios que se consideran progresivos. Básicamente hay dos tipos de escalados: **a) Operativo:** cuando los trabajadores de base se ven incapacitados para tratar las incidencias, la derivan al especialista adecuado; **b) Jerárquico:** Probablemente se den condiciones (incidencias graves o críticas) que requieren la notificación al responsable del servicio; **7.) Investigación y Diagnóstico:** Se buscan las fallas en los sistemas que requieren revisión profunda; realizando actividades como la definición de las actividades que debe ejecutar el operador, establecer las posibles causas de la incidencia, verificar si la incidencia requiere de un cambio, consultar a especialistas internos de acuerdo al incidente planteado; **8) Resolución:** Durante este período, se da la posibilidad que actualmente se ha estimado para el incidente registrado. **9) Cierre:** Al terminar la incidencia, se debe verificar si el cierre del mismo fue organizado, si fue categorizado, si se ejecutó el registro adecuado, si es una incidencia repetida, si existe un reclamo, en ocasiones, es posible aplicar una encuesta al cliente.

En las **Dimensiones**, se encuentra el **registro**, según Steinberg (2013, p 89), todos los incidentes deben registrarse con toda la información, incluida la fecha y la hora. Esto se conoce como registro de incidentes y afectará a ambos incidentes recibidos a través del Centro de Atención, ya que aquellos que son reconocidos automáticamente con un sistema de monitoreo de eventos.

**Indicador: Ratio de Resolución de Incidencias (RRI):** Índice correspondiente al Ratio de Resolución de Incidencias, y cuya fórmula de cálculo es:

$$RRI = \frac{C}{A}$$

Donde:

**RRI:** Ratio de resolución de Incidencias.

**C:** Número de incidencias resuelta cumpliendo el SLA.

**A:** Número Total de Incidentes

Según Steinberg (2013, pág. 88), en la segunda **Dimensión**, la **resolución y recuperación** indica que, al determinarse una solución factible, se debe implementar y probar; precisamente, de eso se trata la resolución.

**Indicador: Ratio de Utilización Laboral en Incidencias (RULI):** es el cociente de las horas invertidas en la resolución de incidencias entre las horas disponibles para atender incidencias (y la fórmula utilizada para calcularlo).

$$RULI = \frac{H}{G}$$

Dónde:

**RULI:** Ratio de Utilización Laboral en Incidencias.

**H:** Horas Invertidas en la Resolución de Incidencias.

**G:** Horas Disponibles para atender incidencias.

**Solución y recuperación de incidencias:** De acuerdo con ITIL (2011 pág. 287), una vez que se ha identificado una solución, debe ser aprobada, ya que le beneficiará resolver y proporcionar respuestas. Usar una base de datos de conocimiento ayudará. **Informe de Incidencias:** Según ITIL (2011, pág. 287), es preciso reconocer y tener el detalle de todos los incidentes resueltos y en proceso, este detalle nos ayuda a reconocer los puntos de ataque para la solución rápida de las mismas.

Seguidamente, también sobre la variable dependiente, se define las **escalas de Tiempo**: que, según García (2016, pág.43) desde el SLA representan los lapsos en los cuales se deben determinar y reparar las incidencias; conforme a ello, se requiere la aplicación de herramientas de gestión para automatizar las escalas de este ciclo; asimismo, para controlar las alarmas y escalados de la réplica /solución de las incidencias comprendido en los lapsos establecidos.

Finalmente, se debe mencionar el **Sistema Web**, el cual, según Báez (2012, pág. 96), estas aplicaciones web son generalmente desarrollados bajo un entorno de S.O instalados en un servidor web, asemejándose a un sitio de Internet. Mientras que, la **Gestión de Incidencia**, según Guzmán (2017, parr. 2) permite coordinar las incidencias de manera más eficaz y eficiente posible, ya que, cada incidencia produce retrasos en el servicio.

### III. MÉTODO

#### 3.1. Tipo y diseño de la investigación

##### 3.1.1. Tipo de investigación

Sobre este particular, Baena (2017, pág. 18) señala con respecto al **Tipo de Investigación** que se considera aplicada, pues tiene como propósito el abordaje de una problemática para establecer soluciones. Esta investigación puede generar hechos nuevos, al proyectar de manera adecuada esta investigación, de forma tal que se pueda tener confianza en los hechos presentados, por lo que la nueva información tendría utilidad para la teoría.

La presente tesis permitió plantear la relación existente entre la implementación del sistema web y la gestión de incidencias, de forma tal; que fue factible solucionar la problemática identificada, estudiando los hechos con el objeto de evidenciar las Hipótesis sustentadas a través de los indicativos formulados.

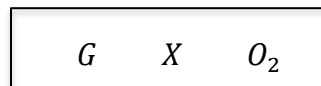
##### 3.1.2. Diseño de investigación

Según Baena (2017, pág. 18), **la investigación experimental** requiere el manejo de la variable experimental no probada en un ambiente

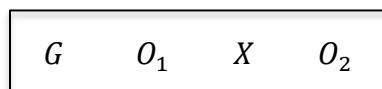
estrictamente controladas para describir cómo o por qué causa un evento o situación conjunta. En estos diseños, el investigador no solo está en una posición práctica para desarrollar el experimento, también tiene conocimiento sobre la naturaleza del fenómeno que estudia. En cuanto al **enfoque de investigación**, corresponde al cuantitativo, en el cual se usan valores numéricos para medir las variables en forma precisa y se enfoca en la comprobación de Hipótesis.

Para Hernández (2014, pág. 141), se basa en aplicar un estímulo o procedimiento a un conjunto y, luego, emplear una medición de una o más variables para contemplar cual es el grado del conjunto en éstas. El diseño de un solo conjunto cuyo nivel de control es mínimo principalmente es eficaz como un primer acercamiento al problema de indagación en la verdad. El plan tiene la siguiente estructura:

- Proyecto preexperimental con pos- test solamente:



- Proyecto preexperimental con pre test – pos test:



$G$ : Grupo Experimental

$O_1$ : Previo de emplear el método web

$O_2$ : Luego de emplear el método web

$X$ : Proceso, incentivo o estado práctico (Método Web)

El diagrama anterior refleja las características de este tipo de diseño expuestas por Ávila (2006).

Para este proyecto de investigación, se utilizará el diseño preexperimental, por lo tanto, los indicadores se medirán en dos fases (pre test, pos test), cotejando los efectos que se producen entre estos dos tipos.

### 3.2. Variables y Operacionalización

#### 3.2.1. Variables

Estas variables mencionadas son:

### **Variable Independiente (VI)**

Sistema Web: se refiere a aquellas, en las cuales los usuarios acceden bajo un servidor web (Andreu 2011, pág.152).

### **Variable Dependiente (VD)**

Gestión de incidencias: comprende recibir, procesar, administrar y resolver incidencias mediante el escalado funcional, dando soporte de segundo nivel y niveles posteriores. Este concepto, engloba el mantenimiento de categorías (OGC, 2010, pág. 60).

A continuación, se presenta la explicación detallada de las variables e indicadores, con sus respectivos instrumentos y sus respectivas fórmulas, como se evidencia en la tabla 3.

Tabla N° 3. Operacionalización de Variables e Indicadores

Variables de Estudio	Definición Conceptual	Definición Operacional	Dimensiones	Indicadores	Ítem /Formula	Escala de Medición
Sistema Web	“Estas se pasan mediante aplicaciones, cuya interfaz se especifica conforme con cada uno de los archivos y textos de todos sus protocolos”. Berzar, Cortijo y Cubero (2016).					
Gestión de Incidencias	“El objetivo central del proceso de gestión de incidentes es retornar a la situación normal lo más rápido posible y reducir al mínimo el impacto en los procesos comerciales” Steinberg (2013)	La Variable Gestión de Incidencia será medido con fichas de registro que tiene 2 indicadores.	Registro (Steinberg ,2013)	Ratio de Resolución de Incidencias	$RRI = \frac{C}{A}$ <p><b>RRI:</b> Ratio de Resolución de Incidencias.  <b>C:</b> Número de incidencias resuelta cumpliendo el SLA.  <b>A:</b> Número Total de Incidentes</p>	Razón
			Resolución y Recuperación (Steinberg 2013)	Ratio de Utilización Laboral en Incidencias.	$RULI = \frac{H}{G}$ <p><b>RULI:</b> Ratio de Utilización Laboral en Incidencias.  <b>H:</b> Horas Invertidas en la Resolución de Incidencias.  <b>G:</b> Horas Disponibles para atender incidencias.</p>	Razón

Fuente: Elaboración Propia.

### 3.3. Población, muestra y muestreo

**Poblacion**, para Kendall y Kendall (2011, pág. 132 ), la población son los elementos seleccionados que se examinarán en detalle, cuyo análisis permitirá responder a las preguntas de investigación.

En el colegio Pestalozzi, para realizar la gestión de incidentes tuvo como objeto los incidentes informados por los distintos profesores a través de llamadas, mensajes de texto, correo y presencialmente, utilizando fichas de registros. Para este desarrollo de tesis, la población a estimar será el número de registros realizados en un mes. Es por eso que se tomará el mes de agosto para la población en un tiempo de 21 días de lunes a viernes.

*Tabla N° 4. Determinar la Población*

INDICADOR	POBLACION	TIPO DE LA POBLACION	TIEMPO/MES
Ratio de resolución de Incidencias	21	Ficha de Registro	21 dias/Agosto
Ratio de Utilización Laboral en incidencias			

**Fuente:** Elaboración propia.

**Muestra:** Kendall y Kendall (2011, pág. 133) indican que si los sujetos o elementos de la población observaran el mundo de la misma manera, o si todos los documentos en una población tuviera exactamente la misma información que cualquier otro, un tamaño de muestra sería suficiente. Al no ser así, es imprescindible estimar una muestra mayor, pero menor que la población.

Por otro lado, Hernández, Fernández y Baptista (2014, pag 141) consideran a la muestra como un subgrupo, perteneciente a un conjunto total definido con características similares, llamado población; cuando la población no supera los 50 individuos o elementos, la población será la muestra. En consecuencia, la



muestra estará determinada por la misma cantidad de elementos que componen la población.

*Tabla N° 5. Determinar la Muestra*

INDICADOR	POBLACION	MUESTRA	TIPO DE LA POBLACION	TIEMPO/MES
Ratio de resolución de Incidencias	21	21	Ficha de Registro	21 días/Agosto
Ratio de Utilización Laboral en incidencias				

**Fuente:** Elaboración propia.

**Muestreo:** Kendall y Kendall (2011 Pág. 134) indica que en la muestra, son seleccionados sistemáticamente los elementos característicos de una población; al revisarse los elementos seleccionados, se estima que el análisis ofrecerá información útil respecto a la población general.

Para la presente Tesis no se llega a utilizar técnicas de muestreo, debido a que se utilizaron todos los elementos que forman parte de la Población.

### **3.4. Técnicas e instrumentos de recolección de datos**

Según Kendall y Kendall (2011, pag 136) mientras el analista de sistemas labora para entender a los usuarios, la empresa y su departamento, requieren una revisión de los datos "duros" que suministrarán los datos, ya que los mismos no se encuentran disponibles en ningún otro medio de recopilación; dichos datos información relevante sobre la organización, donde se encuentra y hacia dónde va, según sus propios trabajadores; por ello, el trabajador debe revisar entonces los datos cuantitativos recabados.

**Técnicas:** Kendall y Kendall (2011, pág. 136) explican que dentro de las organizaciones existe gran cantidad de documentos cuantitativos disponibles para interpretación, como, por ejemplo, informes para tomar acciones, reportes de desempeño, registros y diversidad de formatos; documentos que pueden tener un objeto y ser dirigidos a usuarios específicos.

**Instrumentos de recolección de datos.** Para Hernandez et al. (2014, pág. 201) en todas las investigaciones cuantitativas se utiliza un instrumento para la medición de las variables, lo cual será efectivo cuando el instrumento representa efectivamente a las variables que se desean medir, en caso contrario, la medición es débil, y la investigación carece de bases sólidas. Todo instrumento debe cumplir con tres exigencias fundamentales: confiabilidad, validez y objetividad.

**Validez:** Para Hernandez y otros (2014, pág. 201), la validez se refiere al grado que un instrumento estima efectivamente la variable que se desea medir.

Los 3 aspectos de la validez del instrumento son:

- **Validez de Criterio:** determina la validez que posee un instrumento de medición al cotejar sus resultados con los de algún juicio externo que intente estimar lo mismo (Hernandez y otros 2014, pág. 202).
- **Validez de Constructo:** posiblemente la más relevante dentro del concepto de validez, sobre todo lo que engloba desde una perspectiva científica. Al respecto, está referido al grado en que una herramienta estima y caracteriza una concepción teórica. A dicha validez le corresponde en particular, el sentido del instrumento que está estimando y elementos que se requieren para medirlo (Hernandez y otros 2014, pág. 203).
- **Validez de Contenido:** se menciona al nivel en que una herramienta muestra un dominio particular de contenido de lo que se estima. De tal forma, una herramienta de medición necesita incorporar prácticamente a todos los componentes del dominio de contenido de las variables que van a ser estimadas (Hernandez y otros 2014, pág. 201).

Los instrumentos designados en esta Tesis son fichas de registro, los cuales son validados según los criterios de 3 expertos, quienes tomaron la determinación sobre la calidad de los instrumentos. (Tabla N° 6)

Tabla N° 6. Validación de Instrumentos

Experto	Ficha de Ratio de resolución de Incidencias	Ficha de Ratio de Utilización Laboral de Incidencias
Dr. Frey Elmer Chávez Pinillos	80%	80%
Mg. Aníbal Antonio Galarreta Velarde	100%	80%
Dra. Yesenia Del Rosario Vásquez Valencia	70%	70%

Fuente: Elaboración Propia.

**Confiabilidad:** Según Hernandez y otros (2014, pág. 200) es el grado en el cual el uso repetido del instrumento arroja los mismos resultados.

### 3.5. Procedimientos

Procedimiento para la realizar la presente investigación se tomó en cuenta las siguientes pautas:

- Inicialmente se realizó la búsqueda de tesis y trabajos que tengan relación con el tema a desarrollarse donde se establecen las variables, indicadores e dimensiones que se plasman en la matriz de consistencia.
- Se define el marco teórico donde se define las variables, y se implementa la metodología del estudio, esto incluye; el diseño de investigación, población, muestra y técnicas de recolección de datos.

- Se realizó la validación del instrumento por expertos, seguido de la prueba de confiabilidad utilizando en método de test - retest.
- Teniendo claro que para la muestra se las 21 Fichas de Registro en el mes de agosto, estas fueron revisadas cada día, como registro de Pre-Test.
- Finalmente se producirán 21 fichas de registro para el Post Test, logrando establecer la influencia que tiene el Sistema Web para Gestión de incidencias, toda esta información se lleva al software SPSS para el análisis de datos.

### 3.6. Método de análisis de datos

Gómez (2013) establece que la T-Student representa una prueba empleada en la evaluación de los valores de la población, empleando datos de la muestra y así estimar la probabilidad de pertenecer a dos medios en la misma población o en grupos distintivos; esta prueba se usa en muestras inferiores a 30. Esto es cuantitativo, siendo preexperimental, pudiendo desarrollarse un análisis estadístico para probar la Hipótesis.

#### Definición de Variables

$I_a$ : Indicador del Sistema actual para la Gestión de Incidencias en el Colegio Suizo Pestalozzi – Miraflores sin el Sistema web.

$I_d$ : Indicador del Sistema propuesto para la Gestión de Incidencias en el Colegio Suizo Pestalozzi – Miraflores con el Sistema web.

#### Hipótesis Estadística

- $H_1$ : El Sistema web aumenta el ratio de resolución de Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi - Miraflores.

#### Definición de Indicador: Ratio de Resolución de Incidencias

$RRI_a$ : Ratio de Resolución de Incidencias antes de manipular el sistema web.

$RRI_d$ : Ratio de Resolución de Incidencias después de manipular el sistema web.

**Hipótesis  $H1_0$ :** El Sistema Web no aumentará el ratio de resolución de Incidencias en el Colegio Suizo Pestalozzi – Miraflores.

$$H1_0: RRI_a \geq RRI_d$$

**Hipótesis  $H1_a$ :** El Sistema Web aumenta el ratio de resolución de Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi – Miraflores.

$$H1_a: RRI_a < RRI_d$$

- **$H_2$ :** El Sistema Web reduce el ratio de utilización laboral en Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi – Miraflores.

**Definición de Indicador:** Ratio de Utilización Laboral en Incidencias.

**$RULI_a$ :** Ratio de Utilización Laboral en Incidencias antes de manipular el sistema web.

**$RULI_d$ :** Ratio de Utilización Laboral en Incidencias después de manipular el sistema web.

**Hipótesis  $H2_0$ :** El Sistema Web no reduce el ratio de utilización laboral en Incidencias en el Colegio Suizo Pestalozzi – Miraflores.

$$H2_0: RULI_a \geq RULI_d$$

**Hipótesis  $H2_a$ :** El Sistema Web reduce el ratio de utilización laboral en Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi – Miraflores.

$$H2_a: RULI_a > RULI_d$$

### **Nivel de significancia**

Para esta investigación se considera lo siguiente:

**Nivel de Significancia**  $X = 0.05$  (5% error)

**Nivel de confianza**  $= (1-X) = 0.95 = 95\%$

En función de la tabla de distribución normal (Anexo 16), se obtiene el valor de  $Z_x = -2.08$  (cola izquierda) o  $Z_x = 2.08$  (cola derecha) para un nivel de significancia  $X = 0.05$  y una Hipótesis unilateral.

### **Estadístico de Prueba**

Para evaluar la muestra ( $n = 21$  fichas de registro de incidencias diarias), se utilizará la prueba t-Student o la prueba Wilcoxon, al ser inferior a 30.

Fórmula estadística – matemática para prueba T – Student:

$$t = \frac{X_1 - X_2}{\sigma \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}$$

Donde:

$n_1$  y  $n_2$  = Tamaño de las muestras

$X_1$  y  $X_2$  = Media de las muestras

$\sigma$  = Desviación Estándar

### **Región de Rechazo**

La zona de rechazo de la cola derecha es  $t = t_x$

Donde  $t_x$  es tal que:  $P [t > t_x] = 0.05$

Donde  $t_x$  = **Valor Tabular**

Así, la región de rechazo es  $t > t_x$

La zona de rechazo de la cola izquierda es  $t = t_x$

Donde  $t_x$  es tal que:  $P [t < t_x] = 0.05$

Donde  $t_x$  = **Valor Tabular**

Así, la región de rechazo es  $t < t_x$

### **Análisis de Resultados**

Los datos fueron sometidos a la prueba t-Student o prueba de rangos con signo de Wilcoxon, para verificar la hipótesis y precisar si se rechaza o se acepta la hipótesis nula.

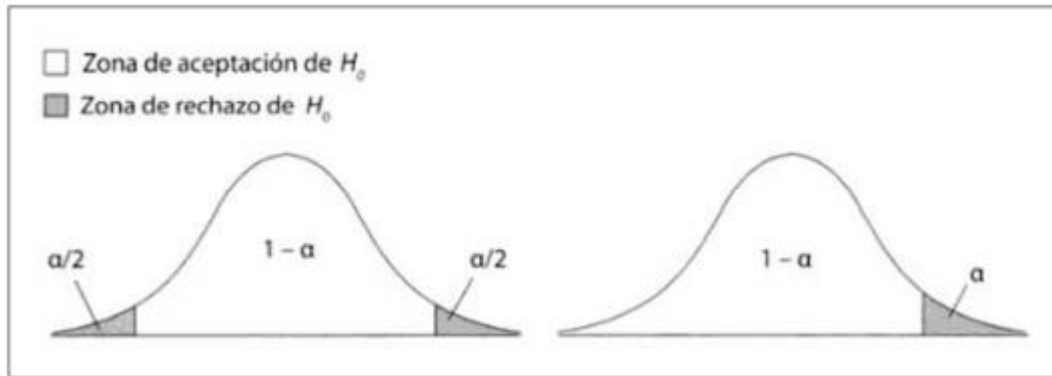


Figura N° 8. Región de aceptación y región de Rechazo para una Hipótesis bilateral o unilateral

En la figura 10, se visualizan las zonas de aceptación o rechazo de la hipótesis nula, para Hipótesis bilateral (ambas colas) o unilateral (una cola).

### 3.7. Aspectos Éticos

Este trabajo se lleva a cabo bajo estrictos principios éticos, que permiten asegurar resultados seguros, que contribuyan significativamente al desarrollo posterior de la gestión de incidentes en la empresa.

Para la preparación del trabajo, se solicitó la aprobación de la empresa (Colegio Suizo Pestalozzi - Miraflores), dando acceso a todos los datos requeridos y necesarios. La información obtenida de la empresa se tomará en privado porque sirvió para este estudio con la debida consideración a la integridad de la misma sin divulgarla, para así impedir la propagación y modificación de los datos de los clientes. El autor también empleó los datos durante el tiempo requerido para completar el estudio.

**Beneficencia:** este principio es un concepto de la ética de la investigación, que contempla el esfuerzo que deben hacer los investigadores por lograr bienestar de los participantes en un ensayo clínico u otra investigación.

**No maleficencia:** principio que se entiende como la obligación de no perjudicar a los participantes de la investigación, tratando de maximizar todos los beneficios posibles y anticipar las medidas necesarias para reducir los posibles riesgos.

**Autonomía:** En este llamado principio moral, el respeto por las personas también se basa en el hecho de que las personas tienen la capacidad de realizar

sus propias acciones como personas, es decir, determinar sus propias propiedades de forma independiente.

**Justicia:** este el principio se relaciona con los parámetros de elección de los partícipes en una investigación, en dos niveles: individual y social.

#### IV. RESULTADOS

En el presente estudio se realizó un pre-test de la situación actual del Colegio Suizo Pestalozzi – Miraflores (Ver Anexo 8 y 11), y posterior a la implementación de la propuesta en el colegio se realizó el post-test (Ver Anexo 14 y 15) para comprobar las Hipótesis previstas en la investigación.

- **Análisis Descriptivo**

Los resultados que se obtuvieron en el estudio se presentan a continuación:

**Indicador 1: Ratio de Resolución de Incidencias (RRI)**

El Ratio de Resolución de Incidencias (RRI) obtenido en el pre-test y en el post-test proyectan los resultados estadísticos de acuerdo a la siguiente Tabla:

Tabla 10: Estadísticos Descriptivos de RRI antes y después de la implementar el Sistema Web

Estadísticos Descriptivos					
	N	Mínimo	Máximo	Media	Desviación Estándar
RRI_Pretest	21	0.64	0.91	0.8157	0.07152
RRI_Postest	21	0.93	1.00	0.9714	0.02175
N válido (por lista)	21				

Fuente: Elaboración Propia

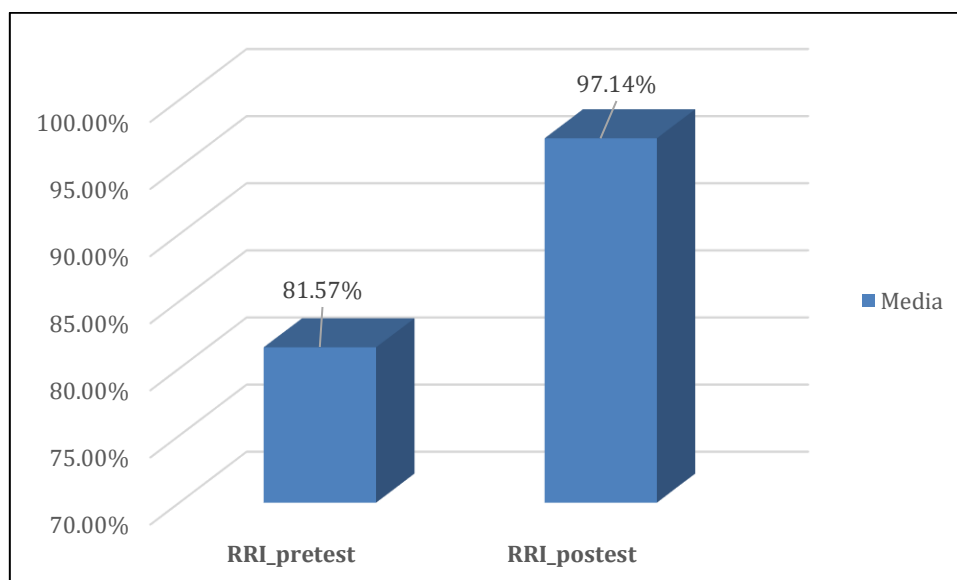
Con respecto al Ratio de Resolución de Incidencias para la gestión de incidencias, el pre-test arrojó un valor de 0.82, y para el post-test fue de 0.97. Como se evidencia en la Tabla 10, lo que revela un cambio a partir de la implementación del sistema Web; por otra parte, el Ratio de Resolución de Incidencias fue 0.64 antes y 0.93 luego de haberse implementado la propuesta.



En relación a la dispersión de datos, en el pre-test fue de 0.071 y en el post-test 0.021.

En la Figura 11, se presenta el antes y luego de la implementación del Sistema Web. Se visualiza que para el Ratio presentó un valor de 81.57% en el pre-test y en el post-test fue de 97.14%; lo que significa que con la implementación del Sistema Web se mejoró 15.57% la atención de incidencias.

Figura 11: Porcentaje de Ratio de Resolución de Incidencias antes y después de Implementar el Sistema Web



Fuente: Elaboración Propia

### Indicador 2: Ratio de Utilización Laboral en incidencias (RULI)

El Ratio de Utilización Laboral en incidencias (RULI) obtenidos en el pre-test y post-test proyectan los datos que se muestran a continuación:

Tabla 11: Estadísticos Descriptivos de RULI antes y después de la implementar el Sistema Web

Estadísticos Descriptivos					
	N	Mínimo	Máximo	Media	Desviación Estándar
RULI_Pretest	21	0.9	1.07	0.9986	0.04041
RULI_Postest	21	0.78	1.01	0.8957	0.05016
N válido (por lista)	21				

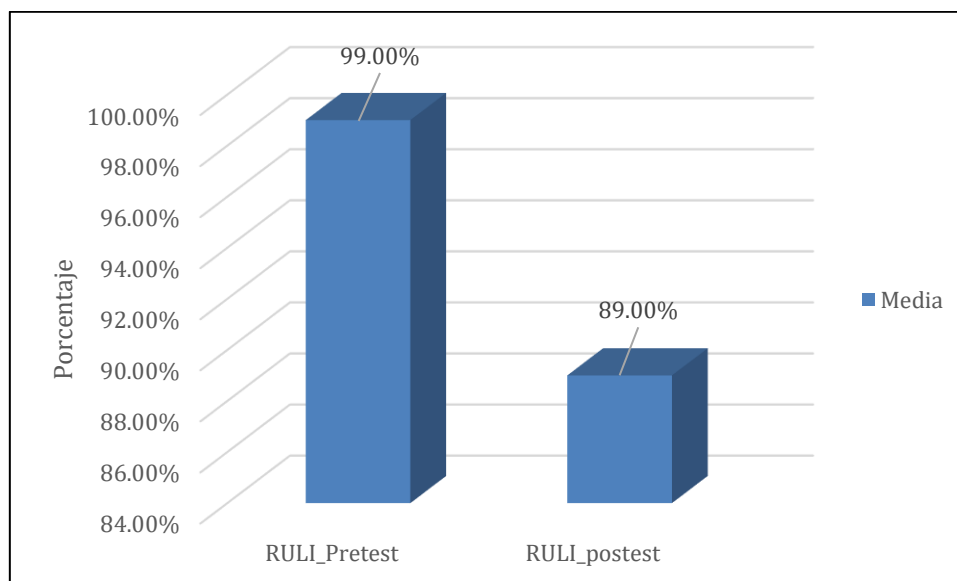
Fuente: Elaboración Propia

Para el Ratio de Utilización Laboral en incidencias, el pre-test reflejó un valor de 0.99, y el post-test fue de 0.89. Esto quiere revela diferencias con la implementación del sistema Web; en tanto, el Ratio de Utilización Laboral en incidencias mínimo se ubicó en 0.9 antes y 0.78 después de implementarse la propuesta.

Ahora bien, la dispersión del Ratio de Utilización Laboral en incidencias, en el pre-test la variación fue de 0.04 y en el post-test 0.05.

En la Figura 12, se interpreta en porcentajes la implementación del Sistema Web. Se visualiza un valor de 99% en el Ratio de Utilización Laboral en incidencias en el pre-test y en el post-test fue 89%, lo que significa que la implementación del Sistema Web permitió 10% de reducción en la utilización laboral de incidencias.

Figura 12: Porcentaje de Ratio de Utilización Laboral en Incidencias antes y después de Implementar el Sistema Web



Fuente: Elaboración Propia

- **Análisis Inferencial**

A continuación, se presenta el análisis de los resultados:

**Prueba de Normalidad**

La prueba de normalidad para el Ratio de resolución de incidencias y el Ratio de utilización laboral de incidencias, fue desarrollado aplicando el método Shapiro-Wilk, ya que la muestra fue inferior a 50 elementos, tal como lo indica Hernández

y otros (2014, p. 376). El análisis se desarrolló mediante el programa SPSS, con una confiabilidad de 95%, tomando en cuenta los siguientes parámetros:

Si:

Sig. < 0.05, entonces la distribución se comporta de forma no normal.

Sig.  $\geq$  0.05, entonces la distribución se comporta de forma normal.

Dónde:

Sig.: P-valor o nivel crítico del contraste.

### Indicador 1: Ratio de Resolución de Incidencias (RRI)

Para la selección de la prueba de hipótesis, los datos se sometieron a comprobación, concretamente, si el Ratio de Resolución de Incidencias se distribuye normalmente.

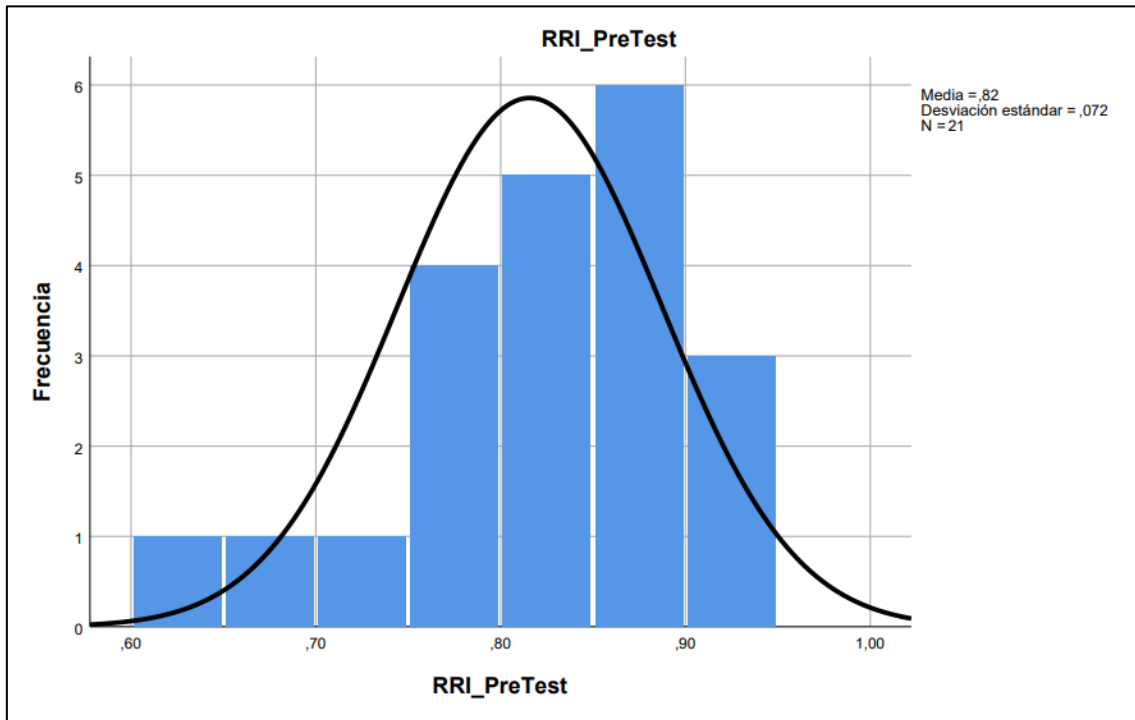
Tabla 12: Prueba de Normalidad de RRI antes y después de la implementar el Sistema Web

Pruebas de Normalidad			
	Shapiro - Wilk		
	Estadístico	gl	Sig.
RRI_Prestest	0.932	21	0.149
RRI_Posttest	0.912	21	0.061

Fuente: Elaboración propia

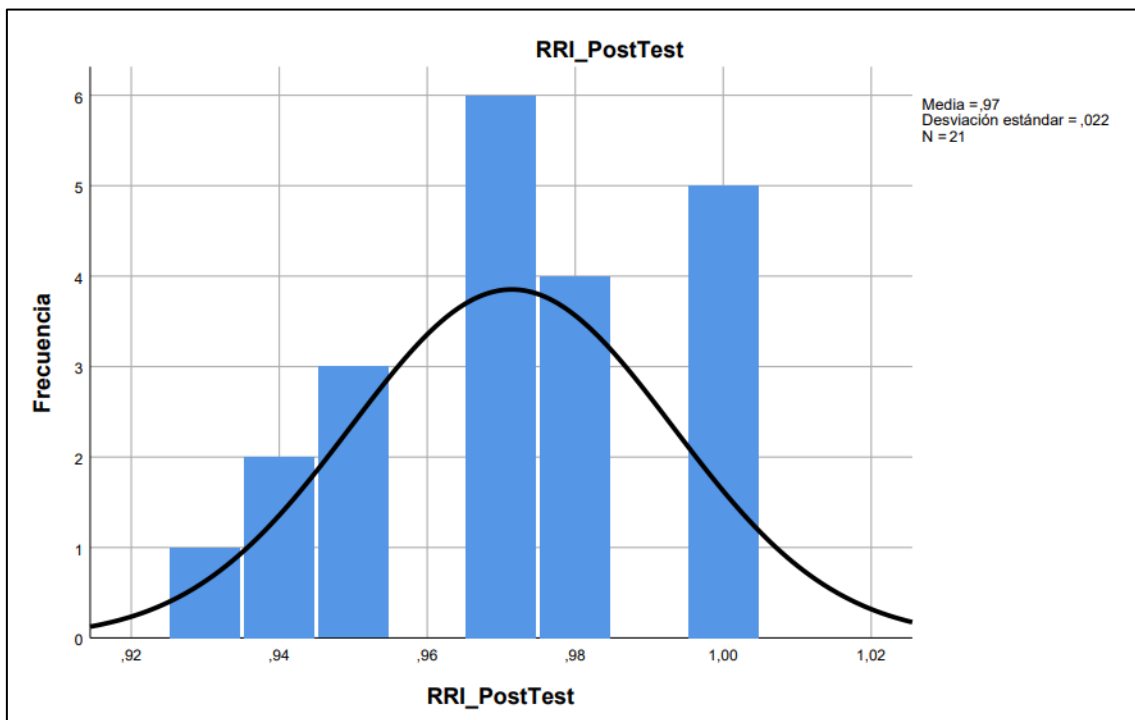
Tal como se presenta en la Tabla 12, los valores obtenidos con la prueba revelan que el Sig. del Ratio de Resolución de Incidencias en la gestión de incidencias para el Pre-Test, resultó siendo de 0.149, valor superior a 0.05; por lo que, el ratio evidenció una distribución normal. Mientras que, los valores obtenidos en la el Post-Test revelan que el Sig. del ratio fue de 0.061, superior a 0.05, lo que revela que el ratio se distribuye de forma normal. Para evidenciar de esta forma, una distribución normal de ambos datos (ver Figuras 13 y 14).

Figura 13: Prueba de normalidad de Ratio de Resolución de Incidencias antes de implementar el Sistema Web



Fuente: Elaboración Propia

Figura 14: Prueba de normalidad de Ratio de Resolución de Incidencias después de implementar el Sistema Web



Fuente: Elaboración Propia

## Indicador 2: Ratio de Utilización Laboral en incidencias (RULI)

Para la selección de la prueba de hipótesis, se comprobó la distribución de los valores, específicamente si el RULI tenía una distribución normal.

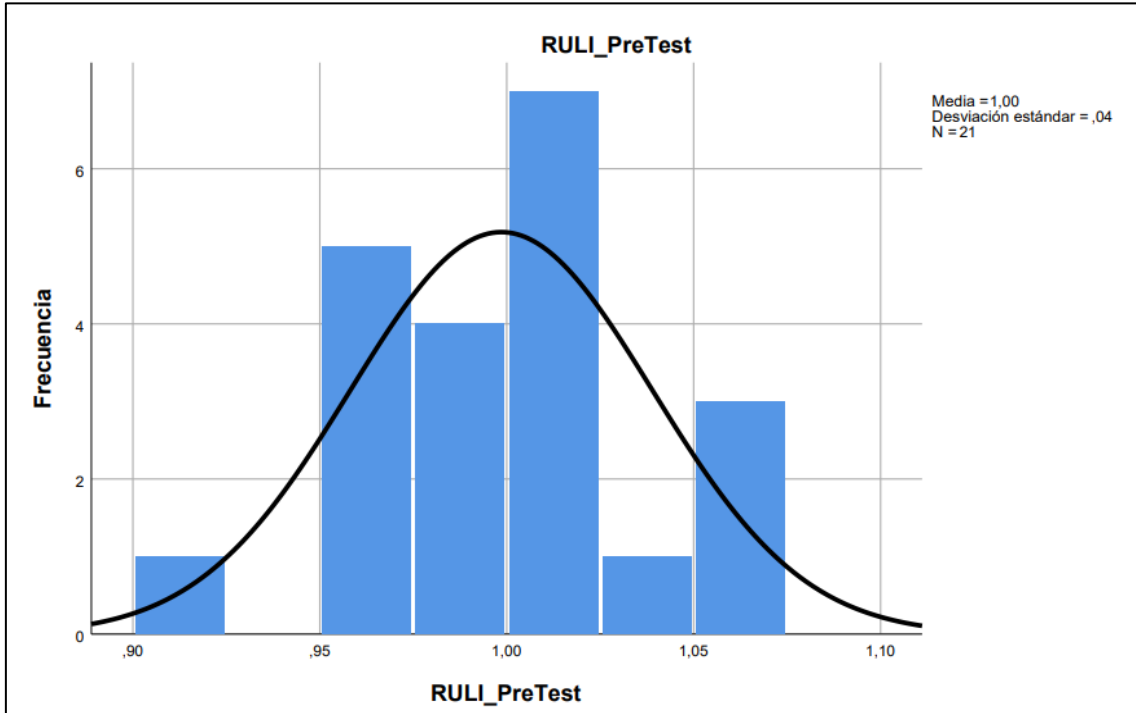
Tabla 13: Prueba de Normalidad de RULI antes y después de la implementar el Sistema Web

Pruebas de Normalidad			
	Shapiro - Wilk		
	Estadístico	gl	Sig.
RULI_pretest	0.951	21	0.354
RULI_postest	0.959	21	0.496

Fuente: Elaboración propia

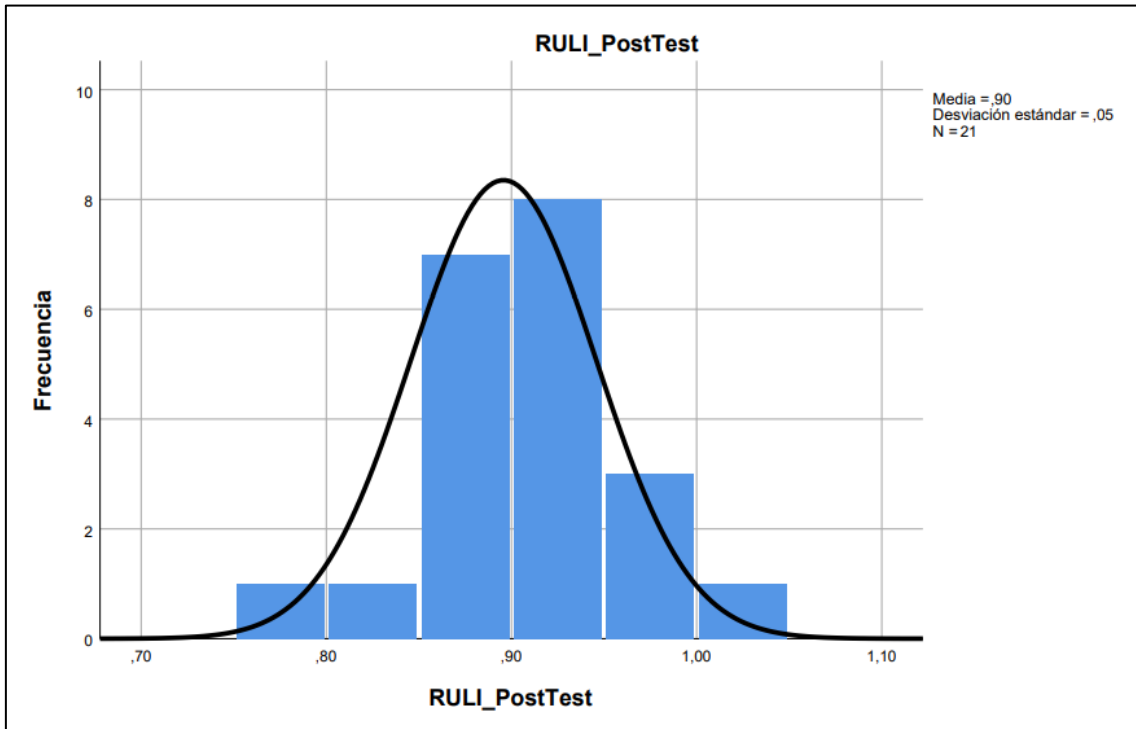
De esta manera se puede observar según la prueba que el Sig. del RULI en el Pre-Test fue de 0.354, por encima de 0.05, lo que indica que dicho ratio tiene una distribución normal. En tanto, los datos del Post-Test revelan que el Sig. del ratio se ubicó en 0.496, también superior a 0.05, revelando así que dicho ratio tiene de igual forma, una distribución normal. De esta manera, los resultados indican que ambos datos presentan una distribución normal (ver Figuras 15 y 16).

Figura 15: Prueba de normalidad de Ratio de Utilización Laboral en Incidencias antes de implementar el Sistema Web



Fuente: Elaboración Propia

Figura 16: Prueba de normalidad de Ratio de Utilización Laboral en Incidencias después de implementar el Sistema Web



Fuente: Elaboración Propia

- **Prueba de Hipótesis**

**Hipótesis de Investigación 1:**

- $H_1$ : El Sistema web aumenta el ratio de resolución de Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi - Miraflores.

**Hipótesis Estadística**

**Definición de Indicador:** Ratio de Resolución de Incidencias

$RRI_a$ : Ratio de Resolución de Incidencias antes de manipular el sistema web.

$RRI_d$ : Ratio de Resolución de Incidencias después de manipular el sistema web.

**Hipótesis  $H1_0$ :** El Sistema Web no aumentará el ratio de resolución de Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi – Miraflores.

$$H1_0: RRI_a \geq RRI_d$$

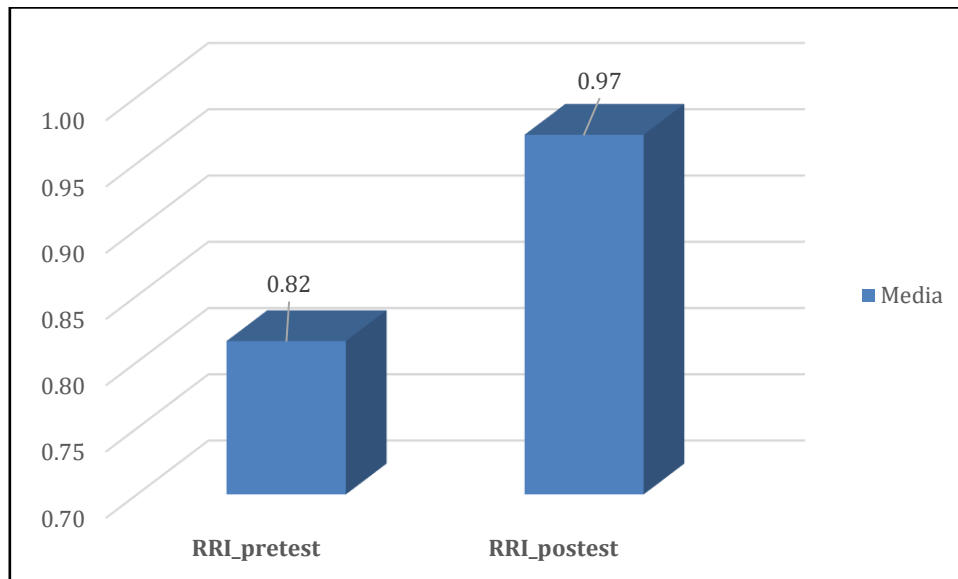
El indicador sin el Sistema Web es mejor que el indicador con el Sistema Web.

**Hipótesis  $H1_a$ :** El Sistema Web aumenta la ratio de resolución de Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi – Miraflores.

$$H1_a: RRI_a < RRI_d$$

El indicador sin el Sistema Web no es mejor que el indicador con el Sistema Web.

Figura 17: Ratio de Resolución de Incidencias antes y después de Implementar el Sistema Web



Fuente: Elaboración Propia

Observando la Figura 17, se evidencia que existe un aumento en el Ratio de Resolución de Incidencias, verificable al contrastar las medias correspondientes, que evidencia aumento de 0.15, es decir, 15.57% de progreso.

En relación a los valores obtenidos del contraste de hipótesis se utilizó la Prueba T-Student, ya que los datos obtenidos (Pre-Test y Post-Test) evidenciaron una distribución normal. Asimismo, T contraste es de -10.126, claramente inferior a -2.08 (Ver tabla 14).

Tabla 14: Prueba de T-Student para el Ratio de Resolución de Incidencias en la Gestión de Incidencias antes y después de implementar el Sistema Web

Pruebas de Muestras emparejadas					
		Diferencias emparejadas	t	gl	Sig. (bilateral)
		Media			
Par 1	RRI_Prestest - RRI_Postest	-0.15571	-10.126	20	0.000

Fuente: Elaboración Propia

De esta forma, se refuta la Hipótesis nula, aceptando la Hipótesis alterna, con un nivel de confianza de 95%. Además, el valor T se ubicó en la zona de rechazo (ver Figura 18); en tal sentido, el Sistema Web aumenta el ratio de resolución de Incidencias en la institución objeto de estudio.



### Aplicando la Formula T-Student:

$$t_c = \frac{x - \mu}{S/\sqrt{n}}$$
$$t_c = \frac{0.8157 - 0.9714}{0.07047/\sqrt{21}}$$
$$t_c = \frac{0.8157 - 0.9714}{0.07047/4.58}$$
$$t_c = \frac{-0.15571}{0.01538}$$
$$t_c = -10.126$$

Figura 18: Prueba T-Student - Ratio de Resolución de Incidencias



Fuente: Elaboración Propia

### Hipótesis de Investigación 2:

- $H_2$ : El Sistema Web reduce el ratio de utilización laboral en Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi – Miraflores.

**Definición de Indicador:** Ratio de Utilización Laboral en Incidencias.

$RULI_a$ : Ratio de Utilización Laboral en Incidencias antes de manipular el sistema web.

$RULI_d$ : Ratio de Utilización Laboral en Incidencias después de manipular el sistema web.

**Hipótesis  $H_{2_0}$ :** El Sistema Web no reduce el ratio de utilización laboral en Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi – Miraflores.

$$H_{2_0}: RULIa \geq RULId$$

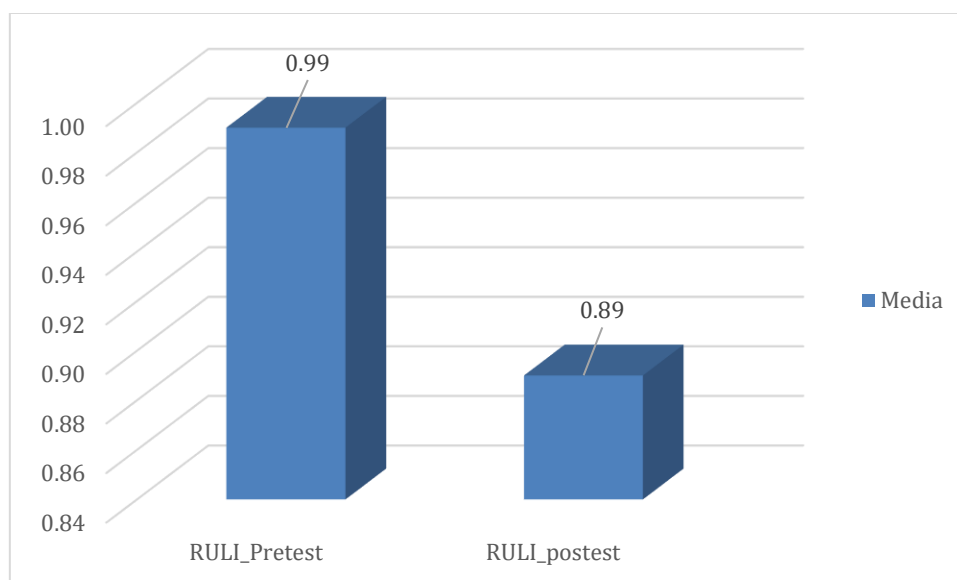
El indicador sin el Sistema Web es mejor que el indicador con el Sistema Web.

**Hipótesis  $H_{2_a}$ :** El Sistema Web reduce el ratio de utilización laboral en Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi – Miraflores.

$$H_{2_a}: RULIa > RULId$$

El indicador sin el Sistema Web no es mejor que el indicador con el Sistema Web.

Figura 19: Ratio de Utilización Laboral en Incidencias antes y después de Implementar el Sistema Web



Fuente: Elaboración Propia

De esta forma, se determina un aumento en el RULI, evidenciado al contrastar las medias correspondientes, descendiendo 0.10, lo que equivale a 10% de mejora.

En relación a los datos del contraste de Hipótesis se utilizó T-Student, ya que los valores (Pre-Test y Post-Test) evidenciaron una distribución normal. El T contraste fue 9.387, notablemente superior a 2.08 (Ver tabla 15).

Tabla 15: Prueba de T-Student para el Ratio de Utilización Laboral en Incidencias en la Gestión de Incidencias antes y después de implementar el Sistema Web

Pruebas de Muestras emparejadas					
		Diferencias emparejadas	t	gl	Sig. (bilateral)
		Media			
Par 1	RULI_Pretest - RULI_Posttest	0.10286	9.387	20	0.000

Fuente: Elaboración Propia

En consecuencia, se refuta la hipótesis nula, admitiendo con un 95% de confianza la hipótesis alterna. Asimismo, el valor T se ubicó en la zona de rechazo (ver Figura 20). Es por ello que, se evidenció que el Sistema Web reduce el ratio de utilización laboral en Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi – Miraflores.

#### Aplicando la Formula T-Student:

$$t_c = \frac{x - \mu}{S/\sqrt{n}}$$

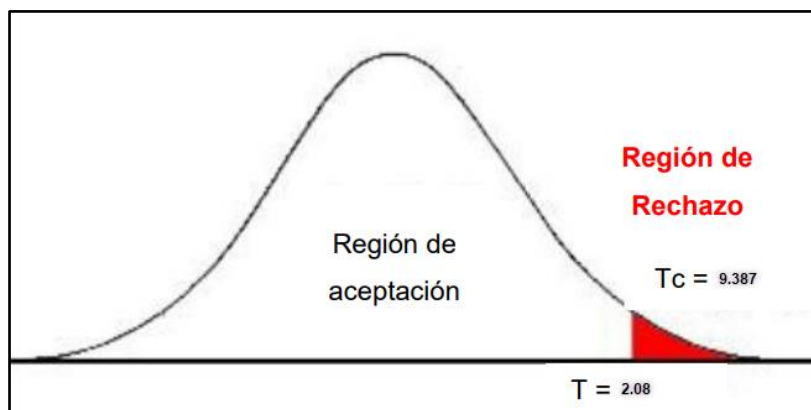
$$t_c = \frac{0.9986 - 0.8957}{0.05021/\sqrt{21}}$$

$$t_c = \frac{0.9986 - 0.8957}{0.05021/4.58}$$

$$t_c = \frac{0.1029}{0.01096}$$

$$t_c = 9.387$$

Figura 20: Prueba T-Student - Ratio de Utilización Laboral en Incidencias



Fuente: Elaboración Propia

## V. DISCUSIÓN

En función de los datos obtenidos con el estudio, se realizó una comparación entre el Ratio de Resolución de Incidencias y el Ratio de Utilización Laboral de Incidencias.

En este sentido, se comprobó que el sistema web ha aumentado la atención de incidencias resueltas, pasando del 82% al 97%, significando un incremento de 15%, mejorando la situación planteada de la atención de incidencias. De la misma forma, Sandoval (2018), en su investigación concluyó que la aplicación del Sistema Web optimiza el proceso de gestión de incidentes de la Institución Educativa Innovaschools Sede Los Olivos. Asimismo, permitió que el departamento de sistemas tenga mejores servicios para resolver incidencias, en este caso el porcentaje de incidencias resueltas en el primer nivel experimentaron un incremento, al pasar de 70.10% a un 90.05%, en tanto que el Ratio de Utilización Laboral de Incidencias sufrió una disminución al alcanzar en primera instancia un 99% y posteriormente registro 89%, es decir, una reducción de 10 %. Del mismo modo, Sandoval (2018), en su investigación concluyó que la aplicación del sistema Web produjo una serie de mejoras a la gestión de incidencias, permitiendo al departamento de sistemas evitar que los incidentes vuelvan a ocurrir, teniéndose una disminución porcentual de las incidencias reabiertas, al pasar de un 26.66% a un 10.15%.

Al igual que los resultados de Sandoval (2018), la presente investigación demuestra la utilidad de implementar el sistema web dentro de los procesos de la empresa, ya que el mismo proporciona información accesible y confiable, aunado al hecho de comprobar que el sistema Web mejora la gestión de incidentes en el Colegio Suizo Pestalozzi, al ver incrementar el ratio de resolución de incidencias en 15%, y disminuir el ratio de utilización laboral en Incidencias en un 10%.

## VI. CONCLUSIONES

1. Como conclusión del estudio, se destaca que antes de realizar la implementación del Sistema Web, el Ratio de Resolución de Incidencias para la gestión de incidencias, registró un 82%; entre tanto que posterior a la implementación de dicho sistema alcanzó un 97%; observándose de esta forma un aumento de 15% en la gestión. Con lo que, se comprueba que el Sistema Web incrementa el Ratio de Resolución de Incidencias para la Gestión de Incidencias en el Colegio Suizo Pestalozzi – Miraflores.
2. En lo que respecta al Ratio de Utilización Laboral en incidencias para la gestión de incidencias en la institución, se tiene que el mismo registró un 99% antes de la implementación del sistema, y luego de la implementación del mismo, obtuvo un 89%, produciéndose una disminución del 10% para este último indicador.
3. En un plano más general, se concluye que el diseño e implementación del Sistema Web mejoró para la Gestión de Incidencias en el Colegio Suizo Pestalozzi – Miraflores. De esta manera se comprobó que el planteamiento de las Hipótesis fue aceptado con un porcentaje de confiabilidad del 95% y que su instalación en la empresa fue exitosa y óptima.

## VII. RECOMENDACIONES

Considerando todos los resultados obtenidos y las conclusiones que se desprendieron de esta investigación, se recomienda lo siguiente:

- Para futuras estudios que guarden similitud al presente, es conveniente emplear el ratio de resolución de incidencias con propósito de optimizar el registro y resolución de las fallas en la gestión de incidencias, todo esto con el propósito de mejorar la atención al cliente.
- Se recomienda en próximos desarrollos de tesis que guarden similitud a éste se debe tomar en cuenta el indicador utilizado; ratio de utilización laboral en incidencias teniendo como fin reducir el tiempo que toma resolver las incidencias, de esta manera permite disponer de tiempo para atender otras incidencias, haciendo un buen uso del tiempo laboral disponible, esto permitirá un mejor desenvolvimiento del personal.
- Se sugiere y recomienda la implementación de la presente investigación de tesis en Instituciones o empresas para contribuir en la obtención de mejores resultados en gestión de incidencias, de esta forma ayuda a tener un canal de atención en donde se registra todas las incidencias manteniendo una gestión de quien genera la solicitud hasta brindar la solución y cierre de la atención. Con lo cual, el mejoramiento del ratio de resolución de incidencias va a incidir en la reducción en el ratio de reutilización laboral de incidencias logrando tener un desempeño eficaz.

## REFERENCIAS

- ALFONSO ARANA, E.A., 2016. Repositorio Universidad de Guayaquil: Desarrollo de un sistema web orientado a una mesa de servicio para el registro, gestión y control de incidencias técnicas. *Universidad de Guayaquil. Facultad de Ingeniería Industrial* [en línea]. [Consulta: 27 mayo 2020]. Disponible en: <http://repositorio.ug.edu.ec/handle/redug/18820>.
- ANDREU, J., 2011. *Gestión de servidores web (Servicios en red)* [en línea]. S.I.: Editorial Editex. Ciclos Formativos. ISBN 9788490030882. Disponible en: [https://books.google.com.pe/books?id=Gc\\_TAAwAAQBAJ](https://books.google.com.pe/books?id=Gc_TAAwAAQBAJ).
- BAENA, G. y BAENA, P., (2017. Metodología de la investigación Grupo Editorial Patria Sistema de aprendizaje en línea Metodología de la investigación [en línea]. S.I.: s.n. [Consulta: 19 diciembre 2020]. ISBN 978-607-744-752-8. Disponible en: [http://www.biblioteca.cij.gob.mx/Archivos/Materiales\\_de\\_consulta/Drogas\\_de\\_Abuso/Articulos/metodologia de la investigacion.pdf](http://www.biblioteca.cij.gob.mx/Archivos/Materiales_de_consulta/Drogas_de_Abuso/Articulos/metodologia%20de%20la%20investigacion.pdf).
- BALLADARES DE LA CRUZ, D.S., (2018. Sistema web para la gestión de incidencias en la empresa Businesssoft S.R.L. Repositorio Institucional - UCV [en línea]. [Consulta: 29 noviembre 2020]. Disponible en: <http://repositorio.ucv.edu.pe/handle/20.500.12692/39109>.
- BARRANCO DE AREBA, J., 2001. *Metodología del análisis estructurado de sistemas* [en línea]. S.I.: Universidad Pontificia Comillas (Publicaciones). Libros de Texto. ISBN 9788484680437. Disponible en: <https://books.google.com.pe/books?id=PUqxsNVaQC8C>.
- BARRANTES LI ELGUERA, C.M., (2018. Sistema web para la gestión de incidencias informáticas en la empresa South Express Cargo Perú S.A.C. Repositorio Institucional - UCV [en línea]. [Consulta: 29 noviembre 2020]. Disponible en: [http://repositorio.ucv.edu.pe/bitstream/handle/20.500.12692/32439/Vargas\\_FTMD.pdf?sequence=1&isAllowed=y](http://repositorio.ucv.edu.pe/bitstream/handle/20.500.12692/32439/Vargas_FTMD.pdf?sequence=1&isAllowed=y).

- BERZAL, F., CORTIJO, F.J. y CUBERO, J.C., [sin fecha]. Desarrollo Profesional de Aplicaciones Web con ASP.NET. [en línea]. S.I.: [Consulta: 16 diciembre 2020]. Disponible en: <https://elvex.ugr.es/decsai/csharp/pdf/web/web-book-a4.pdf>.
- BURNS, N. y GROVE, S.K., 2004. *Investigacion En Enfermeria* [en línea]. S.I.: Harcourt Brace de Espana, S.A. ISBN 9788481747201. Disponible en: <https://books.google.com.pe/books?id=5UNB9ZknC84C>.
- CALVO GARCÍA, Á.L., 2015. *Gestión de redes telemáticas. IFCT0410* [en línea]. S.I.: IC Editorial. Certificado de profesionalidad. ISBN 9788416433438. Disponible en: <https://books.google.com.pe/books?id=dDPTCgAAQBAJ>.
- CARVAJAL PALOMARES, F., 2017. *Manual. Gestión de servicios en el sistema informático (Transversal: MF0490\_3). Certificados de profesionalidad* [en línea]. S.I.: Editorial CEP, S.L. Manual módulo formativo. ISBN 9788468179308. Disponible en: <https://books.google.com.pe/books?id=ccU-DwAAQBAJ>.
- CHIPULINA PUELLES, L., (2018. Sistema web para la gestión de incidencias en la empresa Consulit S.A.C. Universidad César Vallejo [en línea]. S.I.: Universidad César Vallejo. [Consulta: 29 noviembre 2020]. Disponible en: <http://repositorio.ucv.edu.pe/handle/20.500.12692/21275>.
- COBO, Á., 2005. *PHP y MySQL: Tecnología para el desarrollo de aplicaciones web*. [en línea]. S.I.: Editorial Díaz de Santos, S.A. ISBN 9788479787066. Disponible en: <https://books.google.com.pe/books?id=zMK3GOMOpQ4C>.
- COMMERCE, O. of G., 2010. *Operación del servicio* [en línea]. S.I.: Stationery Office. ISBN 9780113311507. Disponible en: <https://books.google.com.pe/books?id=htb2mp3A2WAC>.



- CORTEZ QUEZADA, M., 2019. Repositorio Digital PUCESE: Implementación de un proceso de gestión de incidentes caso práctico Empresa de Agua Potable y Alcantarillado EAPA San Mateo. *Ecuador - PUCESE - Escuela de Sistemas y Computación* [en línea]. [Consulta: 27 mayo 2020]. Disponible en: <https://repositorio.pucese.edu.ec/handle/123456789/1769>.
- DE LA PEÑA CALVO, N., 2015. *UF1643 - Gestión y control de los sistemas de Información* [en línea]. S.I.: Nuevos Negocios en la Red. ISBN 9788416360956. Disponible en: <https://books.google.com.pe/books?id=6cJWDwAAQBAJ>.
- DURÁN HERAS, M.Á. y ROGERO GARCÍA, J., 2009. *La investigación sobre el uso del tiempo* [en línea]. S.I.: Centro de Investigaciones Sociológicas. Cuadernos Metodológicos. ISBN 9788474764819. Disponible en: <https://books.google.com.pe/books?id=mW7dt3Am7cMC>.
- ELIZONDO ALMAGUER, A., 2018. *La mesa de ayuda: El lado humano de TI* [en línea]. S.I.: Editorial Digital UNID. ISBN 9786079460112. Disponible en: <https://books.google.com.pe/books?id=atRJDwAAQBAJ>.
- EURYDICE, 1999. *Las cifras clave de la educación en Europa 1999-2000, Volumen 4* [en línea]. Ministerio. S.I.: Office for Official Publications of the European Communities. [Consulta: 27 mayo 2020]. ISBN 9282885372. Disponible en: [https://books.google.com.pe/books?id=AwsOCQAAQBAJ&dq=adquisicion+de+equipos+mantenimiento+informaticos&source=gbs\\_navlinks\\_s](https://books.google.com.pe/books?id=AwsOCQAAQBAJ&dq=adquisicion+de+equipos+mantenimiento+informaticos&source=gbs_navlinks_s).
- GAMBOA CRUZADO, J. y RODRIGUEZ SILVA, R., 2015. Desarrollo de un sistema web para el proceso de gestión de incidencias en la empresa Inversiones Tobal S.A.C. - Boticas Inkasalud. *Universidad Autónoma del Perú*. S.I.: Universidad Autónoma del Perú.

- GIL CASTRO, V.E., ROSAS ARMENTA, X. y OLMEDA QUIROZ, M., 2014. *Fundamentos de las tecnologías de Información: viviendo en una sociedad tecnológica. Volumen I* [en línea]. S.l.: Editorial Digital del Tecnológico de Monterrey. ISBN 9786075012797. Disponible en: <https://books.google.com.pe/books?id=vUJuDQAAQBAJ>.
- GIL PASCUAL, J.A., 2016. TÉCNICAS E INSTRUMENTOS PARA LA RECOGIDA DE INFORMACIÓN [en línea]. S.l.: UNED. ISBN 9788436271287. Disponible en: [https://books.google.com.pe/books?id=ANrkDAAAQBAJ&dq=3.4.%09Técnicas+e+instrumentos+de+recolección+de+datos,+validez+y+confiabilidad&authuser=1&source=gbs\\_navlinks\\_s](https://books.google.com.pe/books?id=ANrkDAAAQBAJ&dq=3.4.%09Técnicas+e+instrumentos+de+recolección+de+datos,+validez+y+confiabilidad&authuser=1&source=gbs_navlinks_s).
- GÓMEZ BEAS, D., 2014. *Resolución de incidencias en redes telemáticas (UF1881)*. [en línea]. S.l.: IC Editorial. [Consulta: 27 mayo 2020]. ISBN 9788416433421. Disponible en: <https://books.google.com.pe/books?id=mTPTCgAAQBAJ&printsec=frontcover&hl=es#v=onepage&q&f=false>.
- HERNANDEZ SAMPIERI, R., FERNANDEZ COLLADO, C. y BAPTISTA, M. del P., 2014. *Metodología de la Investigación* (6ta edición). S.l.: s.n. ISBN 9781456223960.
- HUAMAN VALENCIA, H., 2005. *MANUAL DE TÉCNICAS DE INVESTIGACION Conceptos y Aplicaciones*. Ipladees S.A.C [en línea]. [Consulta: 8 julio 2020]. Disponible en: [https://books.google.com.pe/books?id=OEHABAAAQBAJ&vq=confiabilidad&dq=Técnicas+recolección+de+datos+confiabilidad&hl=es&source=gbs\\_navlinks\\_s](https://books.google.com.pe/books?id=OEHABAAAQBAJ&vq=confiabilidad&dq=Técnicas+recolección+de+datos+confiabilidad&hl=es&source=gbs_navlinks_s).
- LAUDON, K.C. y LAUDON, J.P., (2012. *Sistemas de Información Gerencial*, 12va Edición [en línea]. PEARSON ED. México: s.n. [Consulta: 16 diciembre 2020]. ISBN 978-607-32-0949-6. Disponible en:

<https://juanantonioleonlopez.files.wordpress.com/2017/08/sistemas-de-informacic3b3n-gerencial-12va-edicic3b3n-kenneth-c-laudon.pdf>.

- LERMA GONZÁLEZ, H.D., 2009. Metodología de la investigación: propuesta, anteproyecto y proyecto [en línea]. 4a Ed. Bogotá: s.n. [Consulta: 8 julio 2020]. ISBN 978-958-648-602-6. Disponible en: [http://roa.ult.edu.cu/bitstream/123456789/3244/1/METODOLOGIA DE LA INVESTIGACION PROPUESTA ANTEPROYECTO Y PROYECTO.pdf](http://roa.ult.edu.cu/bitstream/123456789/3244/1/METODOLOGIA_DE_LA_INVESTIGACION_PROPUESTA_ANTEPROYECTO_Y_PROYECTO.pdf).
- LOPEZ POMA, C., (2018. Sistema web para la gestión de incidencias de la empresa Gaval Soluciones S.A.C. Repositorio Institucional - UCV [en línea]. [Consulta: 29 noviembre 2020]. Disponible en: <http://repositorio.ucv.edu.pe/handle/20.500.12692/39782>.
- LUJÁN-MORA, S., 2001. RUA: Programación en Internet: clientes web. *Universidad de Alicante. Departamento de Lenguajes y Sistemas Informáticos* [en línea]. [Consulta: 27 mayo 2020]. Disponible en: <http://rua.ua.es/dspace/handle/10045/16994>.
- MARTÍNEZ MEDIANO, C. y GALÁN GONZÁLEZ, A., 2014. TÉCNICAS E INSTRUMENTOS DE RECOGIDA Y ANÁLISIS DE DATOS. Universidad Nacional de educación a Distancia [en línea]. [Consulta: 8 julio 2020]. Disponible en: <https://books.google.com.pe/books?id=iiTHAAQBAJ&printsec=frontcover&hl=es#v=onepage&q&f=false>.
- MESA VARA, J.M., 2014. Desarrollo web en entorno servidor (GRADO SUPERIOR) - Juan Manuel Vara Mesa - Google Libros. Grupo Editorial RA-MA [en línea]. [Consulta: 16 diciembre 2020]. Disponible en: [https://books.google.com.pe/books?id=Zl-fDwAAQBAJ&dq=modelo+vista+controlador+para+sistemas+web&hl=es&source=gbs\\_navlinks\\_s](https://books.google.com.pe/books?id=Zl-fDwAAQBAJ&dq=modelo+vista+controlador+para+sistemas+web&hl=es&source=gbs_navlinks_s).
- MOGUEL, E.A.R., 2005. *Metodología de la Investigación* [en línea]. S.I.:

Universidad Juárez Autónoma de Tabasco. Colección Héctor Merino Rodríguez. ISBN 9789685748667. Disponible en: <https://books.google.com.pe/books?id=r4yrEW9Jhe0C>.

- MONÁ CARDONA, L.D. y URIBE SERNA, A.E., 2015. SISTEMA DE GESTIÓN DE INCIDENTES DE SEGURIDAD INFORMÁTICA PARA CORBETA LYDA DURLEY MONA CARDONA. *Biblioteca USB Medellín (San Benito):CD-3626t* [en línea]. S.l.: Ingenierías. [Consulta: 27 mayo 2020]. Disponible en: <http://hdl.handle.net/10819/3932>.
- MOREJÓN RIVERA, R., CÁMARA, F.A., JIMÉNEZ, D.E. y DÍAZ, S.H., 2016. SISDAM: APLICACIÓN WEB PARA EL PROCESAMIENTO DE DATOS SEGÚN UN DISEÑO AUMENTADO MODIFICADO. [en línea]. [Consulta: 27 mayo 2020]. Disponible en: <https://www.redalyc.org/articulo.oa?id=193246976017>.
- NAMAUFOROOSH, M.N., 2005. *Metodología de la investigación* [en línea]. S.l.: Limusa. Area ciencias sociales. ISBN 9789681855178. Disponible en: <https://books.google.com.pe/books?id=ZEJ7-0hmvhwC>.
- ORELLANA GOMERO, E.L., 2017. Descripción: "Sistema informático para el proceso de gestión de incidencias en la empresa INSECORP SAC". [en línea]. [Consulta: 27 mayo 2020]. Disponible en: [https://alicia.concytec.gob.pe/vufind/Record/UCVV\\_accd0091e7b8377bf68220afc6af741d](https://alicia.concytec.gob.pe/vufind/Record/UCVV_accd0091e7b8377bf68220afc6af741d).
- ORTEGA, VEGA Y ZEÑA (2012). Metodología de la investigación.
- PERES, W. y HILBERT, M.R., 2009. La sociedad de la información en América Latina y el Caribe: desarrollo de las tecnologías y tecnologías para el desarrollo [en línea]. S.l.: Naciones Unidas, CEPAL. Libros de la CEPAL. ISBN 9789213231777. Disponible en: <https://books.google.com.pe/books?id=4FlqzQEACAAJ>.

- RICAURTE ARGÜELLES, B.J., 2018. Repositorio Digital Universidad del Magdalena: Sistema web para la gestión de incidencias y eventos en activos informáticos, biomédicos e infraestructura para Perfect Body Medical Center. [en línea]. [Consulta: 27 mayo 2020]. Disponible en: <http://repositorio.unimagdalena.edu.co/jspui/handle/123456789/3876>.
- RIERA BONILLA, S.C. y ZAMBRANO RENDÓN, A.D., 2014. Repositorio de la Universidad de Fuerzas Armadas ESPE: Artículo Científico - Propuesta de procesos para la fase de operación de los servicios de tecnología fundamentado en ITIL, para Bio Agencia de Comunicación. [en línea]. S.l.: [Consulta: 27 mayo 2020]. Disponible en: <http://repositorio.espe.edu.ec/handle/21000/8636>.
- RODRÍGUEZ MOGUEL, E.A., 2005. *Metodología de la Investigación* [en línea]. S.l.: Universidad Juárez Autónoma de Tabasco. Colección Héctor Merino Rodríguez. ISBN 9789685748667. Disponible en: <https://books.google.com.pe/books?id=r4yrEW9Jhe0C>.
- SANDOVAL VIVIANI, R.J., 2018. Sistema web para la gestión de incidencias en la institución educativa Innovaschools sede Los Olivos. [en línea]. S.l.: [Consulta: 27 mayo 2020]. Disponible en: <http://repositorio.ucv.edu.pe/handle/20.500.12692/17905>.
- SCRUMSTUDY, 2013. Una guía completa para la entrega de proyectos utilizando Scrum Una guía para el CONOCIMIENTO DE SCRUM (GUÍA SBOKTM) 2013 Edición A Guide to the SCRUM BODY OF KNOWLEDGE (SBOKTM Guide) [en línea]. S.l.: s.n. [Consulta: 19 diciembre 2020]. ISBN 9780989925204. Disponible en: [https://www.tenstep.ec/portal/images/pdfs/Suscripciones\\_TenStep/Silver/SCRUMstudy\\_GUIA\\_SBOK\\_espanol.pdf](https://www.tenstep.ec/portal/images/pdfs/Suscripciones_TenStep/Silver/SCRUMstudy_GUIA_SBOK_espanol.pdf).
- SOMMERVILLE, I., 2011. *INGENIERÍA DE SOFTWARE Novena edición*

[en línea]. S.l.: s.n. [Consulta: 27 mayo 2020]. ISBN 9780137035151. Disponible en: <https://baixardoc.com/documents/ingenieria-de-software-ian-somerville-9-edicion--5d191c3256fdf>.

- STEINBERG, R., 2013. Measuring ITSM. Estados Unidos: Trafford Publishing, ISBN: 97814907119450.
- TORRES MENESES, E.R., 2018. Aplicación web para la gestión de incidencias en el soporte de T.I. a los clientes internos de la compañía Pronaca | Semantic Scholar. [en línea]. [Consulta: 27 mayo 2020]. Disponible en: <https://www.semanticscholar.org/paper/Aplicación-web-para-la-gestión-de-incidencias-en-el-Meneses-Rubén/617a9bc9f0a2725623367b2b6cff63bf4b8fe84>.
- VALDEZ SALS, B., 2006. Tecnología en la UABC. En: UABC (ed.) [en línea]. S.l.: Universidad Autónomas de Baja California. Serie Conmemortiva 50 aniversario UABC. Disponible en: <https://books.google.com.pe/books?id=dPtJ1xuC3c0C&printsec=frontcover#v=onepage&q&f=false>.
- VARGAS FATAMA, M., (2018. Sistema web para la gestión de incidencias en la empresa New Global SAC. Repositorio Institucional - UCV [en línea]. [Consulta: 29 noviembre 2020]. Disponible en: <http://repositorio.ucv.edu.pe/handle/20.500.12692/32439>.

# ANEXOS

## ANEXO 01

### CERTIFICADO DE VALIDEZ DE CONTENIDO DEL INSTRUMENTO QUE MIDE LA VARIABLE GESTIÓN DE INCIDENCIAS

N°	DIMENSIONES / Items	Pertinencia <sup>1</sup>		Relevancia <sup>2</sup>		Claridad <sup>3</sup>		Superencias
		Si	No	Si	No	Si	No	
1	Registro Ratio de Resolución de Incidencias $RRI = \frac{C}{A}$ RRI: Ratio de resolución de Incidencias. C: Número de incidencias resuelta cumpliendo el SLA. A: Número Total de Incidentes	X		X		X		
	Resolución y Recuperación	Si	No	Si	No	Si	No	
2	Ratio de Utilización Laboral en Incidencias $RUL = \frac{H}{G}$ RUL: Ratio de Utilización Laboral de Incidencias. H: Horas Invertidas en la Resolución de Incidencias. G: Horas Disponibles para atender incidencias.	X		X		X		

Observaciones (precisar si hay suficiencia): Existe suficiencia. \_

Opinión de aplicabilidad:    Aplicable [X]    Aplicable después de corregir [ ]    No aplicable [ ]

Apellidos y nombres del juez validador. Dr./ Mg: Dr. Frey Chávez Pinillos

DNI: 40074326

Especialidad del validador: Metodólogo

Lima 25 de septiembre del 2020

- <sup>1</sup>Pertinencia: El ítem corresponde al concepto teórico formulado.  
<sup>2</sup>Relevancia: El ítem es apropiado para representar el componente o dimensión específica del constructo  
<sup>3</sup>Claridad: Se entiende sin dificultad alguna el enunciado del ítem, es conciso, exacto y directo

Nota: Suficiencia, se dice suficiencia cuando los ítems planteados son suficientes para medir la dimensión



-----  
 Firma del Experto Informante



## CERTIFICADO DE VALIDEZ DE CONTENIDO DEL INSTRUMENTO QUE MIDE LA VARIABLE GESTIÓN DE INCIDENCIAS

N°	DIMENSIONES / Items	Pertinencia <sup>1</sup>		Relevancia <sup>2</sup>		Claridad <sup>3</sup>		Sugerencias
		Si	No	Si	No	Si	No	
1	Registro  Ratio de Resolución de Incidencias  $RRI = \frac{C}{A}$  <b>RRI:</b> Ratio de resolución de Incidencias. <b>C:</b> Número de incidencias resuelta cumpliendo el SLA. <b>A:</b> Número Total de Incidentes	X		X		X		
2	Resolución y Recuperación  Ratio de Utilización Laboral en Incidencias  $RULI = \frac{H}{G}$  <b>RULI:</b> Ratio de Utilización Laboral de Incidencias. <b>H:</b> Horas Invertidas en la Resolución de Incidencias. <b>G:</b> Horas Disponibles para atender incidencias.	X		X		X		

Observaciones (precisar si hay suficiencia): \_\_\_\_\_

Opinión de aplicabilidad: Aplicable []    Aplicable después de corregir [ ]    No aplicable [ ]

Apellidos y nombres del juez validador. Dr/ Mg: Yesenia Del Rosario Vásquez Valencia    DNI: 40352590

Especialidad del validador:..... ING DE SISTEMAS/ DOCTORA EN EDUCACIÓN .....

<sup>1</sup>Pertinencia: El ítem corresponde al concepto teórico formulado.<sup>2</sup>Relevancia: El ítem es apropiado para representar al componente o dimensión específica del constructo<sup>3</sup>Claridad: Se entiende sin dificultad alguna el enunciado del ítem, es conciso, exacto y directo

Nota: Suficiencia, se dice suficiencia cuando los ítems planteados son suficientes para medir la dimensión

...29....de...junio.. del 2020..

Firma del Experto Informante.



## CERTIFICADO DE VALIDEZ DE CONTENIDO DEL INSTRUMENTO QUE MIDE LA VARIABLE GESTIÓN DE INCIDENCIAS

N°	DIMENSIONES / Items	Pertinencia <sup>1</sup>		Relevancia <sup>2</sup>		Claridad <sup>3</sup>		Sugerencias
		Si	No	Si	No	Si	No	
1	Registro  Ratio de Resolución de Incidencias  $RRI = \frac{C}{A}$  <b>RRI:</b> Ratio de resolución de Incidencias. <b>C:</b> Número de incidencias resuelta cumpliendo el SLA. <b>A:</b> Número Total de Incidentes	X		X		X		Es conforme.
2	Resolución y Recuperación  Ratio de Utilización Laboral en Incidencias  $RULI = \frac{H}{G}$  <b>RULI:</b> Ratio de Utilización Laboral de Incidencias. <b>H:</b> Horas Invertidas en la Resolución de Incidencias. <b>G:</b> Horas Disponibles para atender incidencias.	X		X		X		Es conforme, podría utilizarse otro indicador también como el tiempo promedio en resolver una incidencia.

Observaciones (precisar si hay suficiencia): \_\_\_\_\_ Es conforme

Opinión de aplicabilidad: Aplicable []    Aplicable después de corregir [ ]    No aplicable [ ]

Apellidos y nombres del juez validador. Dr/ Mg: ..... Galarreta Velarde Anibal Antonio .....    DNI:....10831569.....

Especialidad del validador:... Magister en Finanzas y Maestro en Dirección de Tecnologías de la Información .....

<sup>1</sup>Pertinencia: El ítem corresponde al concepto teórico formulado.<sup>2</sup>Relevancia: El ítem es apropiado para representar al componente o dimensión específica del constructo<sup>3</sup>Claridad: Se entiende sin dificultad alguna el enunciado del ítem, es conciso, exacto y directo

Nota: Suficiencia, se dice suficiencia cuando los ítems planteados son suficientes para medir la dimensión

...1... de ...noviembre..... del 2020.....

Firma del Experto Informante.



## ANEXO 02

### Áreas de Colegio Pestalozzi

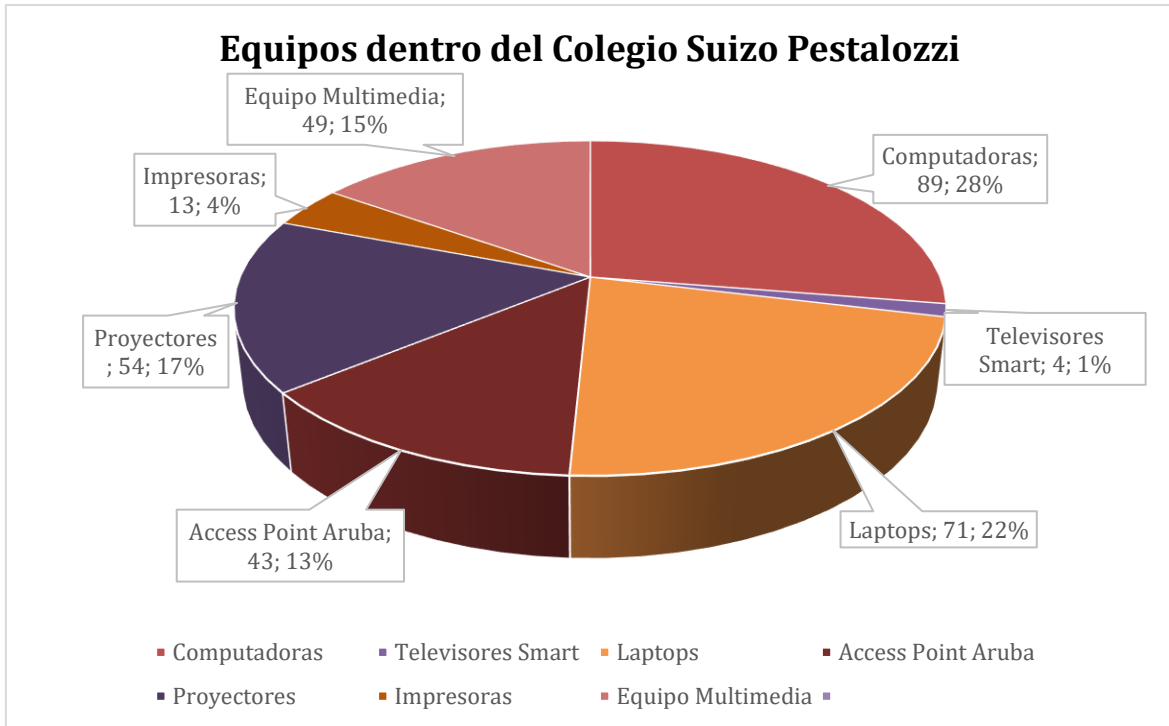


Laboratorio de cómputo



Aulas de Nivel Secundario

### ANEXO 03



## ANEXO 04


### Registro de incidencias actualmente

ID	Categoría	Hora Creación	Hora de Cierre	Departamento	prioridad	Nombre del solicitante	Agente	Fuente	Estado	Asunto
2		3/08/2020 10:41	3/08/2020 11:35		Alta	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	Chat	Cerrado	Lenguaje 1
3		3/08/2020 10:52	3/08/2020 11:55		Media	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	Teléfono	Cerrado	Lenguaje 1
4		3/08/2020 11:03	3/08/2020 12:35		Alta	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	Portal	Cerrado	Audio Zoom
5		3/08/2020 11:14	3/08/2020 12:15		Media	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	Portal	Cerrado	Calendario Teams
6		3/08/2020 11:25	3/08/2020 11:35		Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	Teams	Cerrado	Etiquetas freshservice
7		3/08/2020 12:30	3/08/2020 12:40		Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	Teams	Cerrado	Consulta Zoom
8		3/08/2020 12:47	3/08/2020 13:07		Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	Teams	Cerrado	Microsoft Teams
9		3/08/2020 13:04	3/08/2020 13:34		Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	Teléfono	Cerrado	Creación cuenta Zoom
10		3/08/2020 13:21	3/08/2020 14:01		Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	Teléfono	Cerrado	Acceso a link video de gimnasia
11		3/08/2020 13:38	3/08/2020 14:28		Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	Teléfono	Cerrado	Activación licencia zoom
12	Hardware	3/08/2020 13:55	3/08/2020 14:55	Tesorería	Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	Teléfono	Cerrado	No imprime
13		3/08/2020 14:12	3/08/2020 15:22	Secretaría	Alta	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	Teams	Cerrado	Impresora Secretaría - Reporte Ricoh N° caso 02395861
14	Software	3/08/2020 14:29	3/08/2020 15:49	Coordinac	Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	Teléfono	Cerrado	Zoom Secundaria
15	Other	3/08/2020 14:46	3/08/2020 16:16		Media	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	Zoom Alemán materno 4toB
16		3/08/2020 15:03	3/08/2020 16:43	Coordinac	Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	Correo Matemáticas 2
17	Other	3/08/2020 15:20	3/08/2020 17:10	Coordinac	Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	Actualización de horario Lenguaje y Personal Social
18		3/08/2020 15:37	3/08/2020 17:37	Coordinac	Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	Zoom Lenguaje 4toB
19	Other	3/08/2020 15:54	3/08/2020 18:04	Coordinac	Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	Compartir video por el día del padre
20	Other	3/08/2020 16:11	3/08/2020 18:31	Coordinac	Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	No le llegan los correos al gmail
21	Other	3/08/2020 16:28	3/08/2020 18:58		Alta	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	Zoom Alemán materno 4toB
22	Other	3/08/2020 16:45	3/08/2020 19:25	Coordinac	Alta	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	No me puedo conectar a Zoom
23		3/08/2020 17:02	3/08/2020 19:52		Media	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	Teléfono	Cerrado	Archivos libro de Educare
24		3/08/2020 17:19	3/08/2020 20:19	Coordinac	Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	WhatsApp	Abierto	Pintar pizarra en zoom
25		3/08/2020 17:36	3/08/2020 20:46	Coordinac	Baja	Jenny Leslie Mendoza Contreras	Jenny Leslie Mendoza Contre	Teams	Cerrado	Horario zoom 1A
26	Other	3/08/2020 17:53	3/08/2020 21:13	Coordinac	Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	Office 365
27		3/08/2020 18:10	3/08/2020 21:40	Coordinac	Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	Cómo conservar correos eliminados del correo de tareas
28		4/08/2020 10:41	4/08/2020 11:35	Coordinac	Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	Buzón 2do grado lleno
29	Other	4/08/2020 10:52	4/08/2020 11:55	Coordinac	Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	Cuál es el protocolo de la contraseña de zoom
30		4/08/2020 11:03	4/08/2020 12:15	Coordinac	Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	No puedo ingresar a la página del colegio
31		4/08/2020 11:14	4/08/2020 12:35	Coordinac	Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	Buzón de Personal Social 2do, ya superó el límite de almacenamiento
32	Other	4/08/2020 11:25	4/08/2020 12:55		Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	Usuario temporal Dra. Rivero
33	Software	4/08/2020 11:36	4/08/2020 13:15	Psicoped	Baja	Jenny Leslie Mendoza Contreras	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	Asignar licencia zoom a Laura S.
34	Software	4/08/2020 11:47	4/08/2020 13:35	Psicoped	Baja	Jenny Leslie Mendoza Contreras	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	Verificación licencia pro de zoom
35	Software	4/08/2020 11:58	4/08/2020 13:55	Psicoped	Baja	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	Teams	Cerrado	Breve capacitación Teams
36	Software	4/08/2020 12:09	4/08/2020 14:15	Coordinac	Baja	Jenny Leslie Mendoza Contreras	Jenny Leslie Mendoza Contre	WhatsApp	Cerrado	Asignación de licencia zoom
37		4/08/2020 12:20	4/08/2020 14:35		Baja	Comunicados - Colegio Suizo Pest	Jenny Leslie Mendoza Contre	Correo	Cerrado	RE: PROBLEMAS CON LAS NOTIFICACIONES - SISTEMAS
38	Other	4/08/2020 12:31	4/08/2020 14:55	Secretaría	Baja	Comunicados - Colegio Suizo Pest	Jenny Leslie Mendoza Contre	Correo	Cerrado	El usuario y contraseña nos sale incorrecto - mathekoniq
39	Software	4/08/2020 12:42	4/08/2020 15:15	Coordinac	Alta	Profesor Pestalozzi	Jenny Leslie Mendoza Contre	Correo	Cerrado	Activar licencia zoom.

AGOSTO 2020

## ANEXO 05

### Manual de Organización y Funciones (MOF)

	<b>Funciones del Técnico de Soporte</b>	Jefatura de TI
		Código: PSTLZZ-IT-F002
		Ver. / Rev.: V. 02 R. 01

**1. Objetivo**

Establecer las funciones aplicables sobre los procesos del Técnico de Soporte.

**2. Alcance**

Aplica a todas las operaciones del Técnico de Soporte.

**3. Identificación**

Nombre del puesto	Técnico de Soporte
Área a la que pertenece	Jefatura de Tecnologías de Información
Cargo del Jefe Directo	Jefe de Tecnologías de Información
Número de Plazas	1

**4. Definiciones**

4.1. **ACP:** siglas para Asociación Colegio Pestalozzi.

4.2. **Administrar:** tomando como referencia las etapas de la administración de operaciones productivas, involucra responsabilidades de planeamiento, organización, dirección o ejecución y control (D'Alessio, 2015).

4.3. **Aplicaciones ofimáticas homologadas:** comprenden las diferentes aplicaciones instaladas en las computadoras personales, laptops y notebooks de ACP que cumplen con los requerimientos técnicos de la Jefatura de Tecnologías de Información.

4.4. **Equipos informáticos:** comprenden teléfonos IP, computadoras personales, laptops, notebooks, impresoras y escáneres.

4.5. **SLA:** Acuerdo de Nivel de Servicios (SLA, por sus siglas en inglés).

4.6. **TI:** siglas para Tecnologías de Información.

**5. Relaciones**

5.1. **De Dependencia / Reporta a:** Jefe de TI.

5.2. **De Autoridad / Le Reportan:** Ninguna.

5.3. **De Coordinación**

- Internamente:** Con las áreas pedagógicas, administrativas, de mantenimiento e infraestructuras de TI para el cumplimiento de sus funciones.
- Externamente:** Con los proveedores de servicios de tecnologías de información, ya sea por contratos de soporte vigentes o nuevos requerimientos de productos o servicios para el cumplimiento de sus funciones.

Elaborado por:	Revisado por:	Aprobado por:
Ricardo Estrada	Carlos Alvarado	Urs Steiner
		Fecha: 27.11.2019

ESTE DOCUMENTO ES PROPIEDAD DE ASOCIACIÓN COLEGIO PESTALOZZI Página 3 de 5



## Funciones del Técnico de Soporte

Jefatura de TI

Código: PSTLZZ-IT-F002

Ver. / Rev.: V. 02 R. 01

### 6. Descripción de funciones y responsabilidades

#### 6.1. Funciones generales

1. Asegurar el buen estado, el correcto funcionamiento y la continuidad operativa de las computadoras personales, laptops, notebooks, teléfonos IP, impresoras, escáneres, proyectores de video, equipos multimedia y otros periféricos de ACP.
2. Asegurar el buen estado, el correcto funcionamiento y la continuidad operativa de las instalaciones eléctricas, las instalaciones de la red de datos y las conexiones de audio y video en los diferentes ambientes de ACP: salones de clase, laboratorios, biblioteca, auditorio y salas de reuniones entre otros.
3. Mantener actualizado el inventario de los siguientes equipos informáticos: computadoras personales, laptops, notebooks, teléfono IP, impresoras, escáneres, proyectores de video, equipos multimedia y otros periféricos.
4. Elaborar la planificación quincenal, en coordinación con el Jefe de TI, de las tareas relacionadas a asegurar la continuidad y mejora de los servicios de soporte técnico.
5. Ejecutar los planes de despliegue de equipos informáticos por asignación o cambio.
6. Otras funciones asignadas por el Jefe de TI.

#### 6.2. Funciones específicas

##### 6.2.1. Funciones relacionadas a los servicios de soporte técnico

1. Atender los requerimientos de soporte técnico de los usuarios de las áreas pedagógicas y administrativas de ACP, relacionados al uso de sistemas operativos y aplicaciones ofimáticas homologadas, y equipos informáticos asignados por la Jefatura de TI.
2. Atender los requerimientos de soporte técnico de los estudiantes de ACP, relacionados al uso de sistemas operativos y aplicaciones ofimáticas homologadas, y equipos informáticos asignados por la Jefatura de TI en los salones de clase, laboratorios, talleres y biblioteca.
3. Reportar los problemas con equipos informáticos - que se encuentren en garantía - a los diferentes proveedores de hardware de ACP.
4. Realizar el seguimiento de los casos reportados a los diferentes proveedores de hardware de ACP hasta completar su resolución.
5. Administrar las solicitudes de copia de material educativo en CD o DVD de los usuarios de las áreas pedagógicas de ACP.
6. Administrar las solicitudes de impresión a color de los usuarios de las áreas pedagógicas y administrativa de ACP.
7. Administrar las solicitudes de instalación de laptops y notebooks - en el auditorio, Sala Pestalozzi, Sala Thurgau y eventualmente en los salones de clase y talleres - para las presentaciones en las diferentes actividades escolares de ACP.

Elaborado por:	Revisado por:	Aprobado por:
Ricardo Estrada	Carlos Alvarado	Urs Steiner
		Fecha: 27.11.2019



## Funciones del Técnico de Soporte

Jefatura de TI

Código: PSTLZZ-IT-F002

Ver. / Rev.: V. 02 R. 01

8. Administrar las solicitudes de préstamo de laptops y notebooks – realizadas por el personal docente – para ser utilizadas por los estudiantes de ACP en los salones de clase.
9. Llevar el control y registro diario de los requerimientos de soporte técnico y de las tareas asignadas por el Jefe de TI.

### 6.2.2. Funciones relacionadas a la seguridad de la información

1. Administrar y monitorear la solución de antivirus a través de la consola de McAfee ePO.
2. Ejecutar las tareas necesarias para asegurar que McAfee Endpoint Security se encuentre instalado y actualizado en todas las computadoras personales, laptops y notebooks de ACP.
3. Administrar, monitorear y asegurar el despliegue e instalación de las actualizaciones del sistema operativo y aplicaciones ofimáticas instaladas en todas las computadoras personales, laptops y notebooks de ACP.
4. Velar por la seguridad e integridad de los datos e información que se almacenan y viajan a través de la infraestructura y servicios de red, y equipos informáticos de ACP.
5. Identificar las brechas de seguridad de la infraestructura y servicios de red, y equipos informáticos de ACP, así como también los ataques a los que se puedan ver expuestos.
6. Proponer e implementar, en coordinación con el Jefe de TI, las mejoras necesarias para prevenir y/o mitigar ataques a la infraestructura y servicios de red, y equipos informáticos de ACP.
7. Proponer e implementar, en coordinación con el Jefe de TI, acciones, y planes de capacitación y entrenamiento de usuarios, o acciones específicas, que permitan evangelizarlos en temas de seguridad informática y buen uso de los equipos informáticos.

### 6.2.3. Funciones relacionadas al proceso de mejora continua

1. Diseñar y proponer soluciones de TI que mejoren los servicios de soporte y experiencia de usuario, alineadas a los objetivos de la organización y que generen valor e impacto positivo en nuestra comunidad de usuarios.
2. Evaluar y proponer soluciones que hagan uso intensivo de tecnologías en busca de mejorar la eficiencia operativa del área.
3. Elaborar o actualizar la documentación relacionada a las tareas de soporte técnico: listas de comprobación, manuales de instalación y resolución de problemas y manuales de usuario, entre otros.

Elaborado por:	Revisado por:	Aprobado por:
Ricardo Estrada	Carlos Alvarado	Urs Steiner
		Fecha: 27.11.2019

	<b>Funciones del Técnico de Soporte</b>	Jefatura de TI
		Código: PSTLZZ-IT-F002
		Ver. / Rev.: V. 02 R. 01

#### 6.2.4. Funciones relacionadas a las tareas de mantenimiento preventivo y correctivo

1. Planificar y supervisar, en coordinación con el Jefe de TI, los mantenimientos anuales preventivos y correctivos de las computadoras personales, laptops, notebooks, teléfonos IP, impresoras, escáneres, proyectores de video, equipos multimedia y otros periféricos.
2. Ejecutar eventualmente tareas de mantenimiento correctivo sobre las computadoras personales, laptops, notebooks, teléfonos IP, impresoras, escáneres, proyectores de video, equipos multimedia y otros periféricos.
3. Actualizar el registro mensual de horas de uso de los proyectores de video.

#### Referencias

D'Alessio, F. (2015). Administración de las operaciones productivas. Un enfoque en procesos para la gerencia, Lima, Perú: Pearson Educación del Perú S.A.

#### Revisión y aprobación

Ver. / Rev.	Fecha	Elaborado por	Revisado por	Aprobado por
V. 01 R. 01	12.12.2018	Ricardo Estrada	Carlos Alvarado	Urs Steiner
V. 02 R. 01	27.11.2019	Ricardo Estrada	Carlos Alvarado	Urs Steiner

#### Ediciones

Ver. / Rev.	Fecha	Descripción de las actualizaciones / modificaciones
V. 01 R. 01	27.08.2017	Primera versión.
V. 02 R. 01	27.11.2019	Cambio en la estructura y actualización de funciones.

Elaborado por:	Revisado por:	Aprobado por:
Ricardo Estrada	Carlos Alvarado	Urs Steiner
		Fecha: 27.11.2019

ANEXO 06

Tabla de Áreas y Tipos de Incidencias

	Correo Exchange	Web Gateway McAfee	Mail Gateway McAfee	Firewall WatchGuard	Desbloqueo de cuentas	Configuración de celulares	Préstamo de laptops	Unidades de red del Dominio	Descarga de archivos	Problemas con archivos	Problemas con Equipos y Periféricos	Programa Heinrich	Canvas - Pestalozzi	Programa SAP	Programa de Biblioteca y Videoteca	Impresiones	Escaneos	Equipos de Proyección	Páginas de internet	Microsoft office	Detección de virus	Actualización de programas contables	Actualizaciones de sistema	Actualizaciones de antivirus	Internet por cable	Internet por Wifi	Asistencia con Audiovisuales	
Primaria	x	x	x	x	x	x	x	x	x	x	x					x	x	x	x	x			x	x	x	x	x	
Secundaria	x	x	x	x	x	x	x	x	x	x	x		x			x	x	x	x	x	x			x	x	x	x	x
Nivelación Primaria					x	x	x	x	x	x									x	x	x			x	x	x	x	x
Prekinder y Kinder	x				x	x				x	x	x				x	x			x						x	x	
Alemán	x				x	x	x	x			x								x						x	x	x	x
Matemática				x	x	x	x					x				x			x	x	x			x	x			x
Inglés	x	x	x	x	x	x	x		x	x	x													x	x			x
Música	x	x	x	x	x		x		x	x	x					x		x	x	x	x	x		x	x			x
Física - Ofic.	x					x	x		x	x		x	x				x	x		x	x	x		x	x			
Física	x	x	x	x	x	x						x	x															
Euler		x	x	x	x	x			x	x		x	x															x
Einstein		x	x	x	x	x	x		x	x		x					x	x	x	x	x	x		x	x			x
Robótica	x				x	x	x			x			x				x	x	x	x	x	x		x	x			x
Biología	x			x	x	x	x					x	x				x	x	x	x	x	x		x	x			x
Sala de Profesores	x	x				x	x	x	x			x	x				x	x	x	x	x	x		x	x	x		
Secretaría	x	x				x	x	x	x	x	x						x	x	x	x	x	x		x	x			
Tesorería					x	x	x	x	x	x	x			x			x	x	x	x	x	x		x	x			
Dirección	x	x			x	x	x	x	x	x	x	x	x				x	x	x	x	x	x		x	x			
Psicología	x	x			x	x	x	x	x	x	x						x	x	x	x	x	x		x	x	x		
TI	x	x				x	x					x	x	x	x									x	x			x
TI - Laptops				x	x	x			x	x			x															x
Laboratorio Computo	x	x			x							x	x				x		x	x	x	x		x	x	x		x
Contabilidad	x	x			x															x	x	x	x	x	x	x		
Biblioteca - Videoteca	x	x			x				x	x	x	x	x		x	x					x	x	x	x	x			
Deporte	x	x			x	x						x	x															
Ibo	x	x			x	x						x	x				x		x	x	x	x		x	x	x		
Enfermería - Tópico	x	x			x	x						x	x															
Materiales	x	x			x							x	x															
Impresiones	x	x			x							x	x															
Mantenimiento	x	x			x	x	x			x	x																	
Carrito Robótica					x	x							x															x
Carrito Primaria					x	x											x											x
Carrito Secundaria					x	x							x				x											x
Apafa	x				x																							
Auditorio							x	x					x															x
Sala Pestalozzi							x	x																				x
Sala Thurgau							x	x																				x

Fuente: Elaboración propia.



## ANEXO 07: MATRIZ DE CONSISTENCIA

PROBLEMAS	OBJETIVOS	HIPOTESIS	VARIABLES	DIMENSION	INDICADOR	METODOLOGIA
<b>Principal</b>	<b>General</b>	<b>General</b>	<b>Independiente</b>			
¿Cómo influye un sistema web en la gestión de incidencias en el área de soporte técnico del Colegio Suizo Pestalozzi - Miraflores?	Determinar la influencia del Sistema Web para la gestión de incidencias en el área de soporte técnico del Colegio Suizo Pestalozzi - Miraflores.	El Sistema Web mejora la gestión de incidencias en el área de soporte del Colegio Suizo Pestalozzi - Miraflores.	Sistema Web			<b>Tipo de Estudio:</b> El tipo de estudio del trabajo de investigación es Aplicada.
<b>Secundario</b>	<b>Específico</b>	<b>Específico</b>	<b>Dependiente</b>			<b>Diseño de la Investigación:</b> Experimental
P1: ¿En qué medida el uso de un Sistema Web influye en el ratio de resolución de Incidencias registradas para la gestión de incidencias en el Colegio Suizo Pestalozzi - Miraflores?	O1: Determinar la influencia del sistema web en el ratio de resolución de Incidencias registradas para la gestión de incidencias en el Colegio Suizo Pestalozzi - Miraflores.	H1: El Sistema web aumenta el ratio de resolución de Incidencias registradas para la gestión de incidencias en el Colegio Suizo Pestalozzi - Miraflores.	Gestión de Incidencias	Registro (Steinberg, 2013)	Ratio de Resolución de Incidencias (Steinberg, 2013)	<b>Población:</b> La población estará constituida de 21 Fichas de Registro del Colegio Suizo Pestalozzi - Miraflores
P2: ¿En qué medida el uso de un Sistema Web influye en el ratio de utilización laboral en Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi - Miraflores?	O2: Determinar la influencia del sistema web en el ratio de utilización laboral en Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi - Miraflores.	H2: El Sistema web reduce el ratio de utilización laboral en Incidencias para la gestión de incidencias en el Colegio Suizo Pestalozzi - Miraflores.		Resolución y Recuperación (Steinberg, 2013)	Ratio de Utilización Laboral en incidencias (Steinberg, 2013)	<b>Muestra:</b> La muestra de este proyecto es de 21 Fichas de Registro.

Fuente: Elaboración Propia.

**ANEXO 08:**

**Ficha de registro del indicador: Ratio de Resolución de Incidencias**

<b>FICHA DE REGISTRO</b>					
<b>Tipo de Prueba</b>		Pre -Test			
<b>investigador</b>		Dorador Diaz Luis Felipe			
		Alvarado Flores Carlos Alberto			
<b>Empresa investigada</b>		Colegio Suizo Pestalozzi			
<b>Dirección</b>		Av. Ricardo Palma 1450 Urb. La Aurora - Miraflores			
<b>Indicador</b>		Ratio de Resolución de Incidencias			
<b>Periodo</b>		Ago-20			
<b>Variable</b>	<b>Indicador</b>		<b>Medida</b>	<b>Instrumento</b>	<b>Formula</b>
Gestión de Incidencias	Ratio de Resolución de Incidencias		Unidades	Ficha de Registro	$RRI = \frac{C}{A}$
<b>Ítems</b>	<b>Ficha de Registro</b>	<b>N° Resueltas cumpliendo los SLA (C)</b>	<b>N° Total de Incidencias (A)</b>	<b>Ratio de Resolución de Incidencias (RRI)</b>	
1	F-AGO-3	22	26	0.85	
2	F-AGO-4	19	24	0.79	
3	F-AGO-5	22	31	0.71	
4	F-AGO-6	17	21	0.81	
5	F-AGO-7	16	20	0.80	
6	F-AGO-10	19	23	0.83	
7	F-AGO-11	32	36	0.89	
8	F-AGO-12	15	19	0.79	
9	F-AGO-13	13	17	0.76	
10	F-AGO-14	19	23	0.83	
11	F-AGO-17	16	25	0.64	
12	F-AGO-18	25	32	0.78	
13	F-AGO-19	33	38	0.87	
14	F-AGO-20	36	40	0.90	
15	F-AGO-21	37	41	0.90	
16	F-AGO-24	33	39	0.85	
17	F-AGO-25	26	30	0.87	
18	F-AGO-26	29	32	0.91	
19	F-AGO-27	29	34	0.85	
20	F-AGO-28	25	31	0.81	
21	F-AGO-31	27	39	0.69	

**ANEXO 09:**

**Ficha de registro del indicador: Ratio de Utilización Laboral en incidencias**

<b>FICHA DE REGISTRO</b>					
<b>Tipo de Prueba</b>		Pre -Test			
<b>investigador</b>		Dorador Diaz Luis Felipe			
		Alvarado Flores Carlos Alberto			
<b>Empresa investigada</b>		Colegio Suizo Pestalozzi			
<b>Dirección</b>		Av. Ricardo Palma 1450 Urb. La Aurora - Miraflores			
<b>Indicador</b>		Ratio de Utilización Laboral en incidencias			
<b>Periodo</b>		1 mes			
<b>Variable</b>	<b>Indicador</b>		<b>Medida</b>	<b>Instrumento</b>	<b>Formula</b>
Gestión de Incidencias	Ratio de Utilización Laboral en incidencias		Unidades	Ficha de Registro	$RULI = \frac{H}{G}$
<b>Ítems</b>	<b>Ficha de Registro</b>	<b>Horas Invertidas en la Resolución de Incidencias (H)</b>	<b>Horas Disponibles para Atender Incidencias (G)</b>	<b>Ratio de Utilización Laboral en incidencias (RULI)</b>	
1	F-AGO-3	22.57	21	1.07	
2	F-AGO-4	21.9	21	1.04	
3	F-AGO-5	21.05	21	1.00	
4	F-AGO-6	22.3	21	1.06	
5	F-AGO-7	20.37	21	0.97	
6	F-AGO-10	20.58	21	0.98	
7	F-AGO-11	22.2	21	1.06	
8	F-AGO-12	20.58	21	0.98	
9	F-AGO-13	18.9	21	0.90	
10	F-AGO-14	21.13	21	1.01	
11	F-AGO-17	20.1	21	0.96	
12	F-AGO-18	21.22	21	1.01	
13	F-AGO-19	21.35	21	1.02	
14	F-AGO-20	21.33	21	1.02	
15	F-AGO-21	21.4	21	1.02	
16	F-AGO-24	20.1	21	0.96	
17	F-AGO-25	20.68	21	0.98	
18	F-AGO-26	21.28	21	1.01	
19	F-AGO-27	20.5	21	0.98	
20	F-AGO-28	20.35	21	0.97	
21	F-AGO-31	20.47	21	0.97	

**ANEXO 10:**

**Ficha de registro del indicador: Ratio de Resolución de Incidencias**

<b>FICHA DE REGISTRO</b>					
<b>Tipo de Prueba</b>		Post-Test			
<b>investigador</b>		Dorador Diaz Luis Felipe			
		Alvarado Flores Carlos Alberto			
<b>Empresa investigada</b>		Colegio Suizo Pestalozzi			
<b>Dirección</b>		Av. Ricardo Palma 1450 Urb. La Aurora - Miraflores			
<b>Indicador</b>		Ratio de Resolución de Incidencias			
<b>Periodo</b>		Octubre - Noviembre			
<b>Variable</b>		<b>Indicador</b>	<b>Medida</b>	<b>Instrumento</b>	<b>Formula</b>
Gestión de Incidencias		Ratio de Resolución de Incidencias	Unidades	Ficha de Registro	$RRI = \frac{C}{A}$
<b>Ítems</b>	<b>Ficha de Registro</b>	<b>N° Incidencias Resueltas cumpliendo los SLA (C)</b>		<b>N° Total de Incidencias (A)</b>	<b>Ratio de Resolución de Incidencias (RRI)</b>
1	F-OCT-14	30		32	0.94
2	F-OCT-15	27		29	0.93
3	F-OCT-16	32		33	0.97
4	F-OCT-19	24		24	1.00
5	F-OCT-20	20		21	0.95
6	F-OCT-21	36		36	1.00
7	F-OCT-22	37		38	0.97
8	F-OCT-23	21		22	0.95
9	F-OCT-26	18		19	0.95
10	F-OCT-27	25		25	1.00
11	F-OCT-28	32		33	0.97
12	F-OCT-29	38		39	0.97
13	F-OCT-30	40		41	0.98
14	F-NOV-2	42		43	0.98
15	F-NOV-3	40		40	1.00
16	F-NOV-4	41		42	0.98
17	F-NOV-5	32		32	1.00
18	F-NOV-6	33		34	0.97
19	F-NOV-9	34		36	0.94
20	F-NOV-10	36		37	0.97
21	F-NOV-11	40		41	0.98


**ANEXO 11:**

**Ficha de registro del indicador: Ratio de Resolución de Incidencias**

<b>FICHA DE REGISTRO</b>					
<b>Tipo de Prueba</b>		Post -Test			
<b>investigador</b>		Dorador Díaz Luis Felipe			
		Alvarado Flores Carlos Alberto			
<b>Empresa investigada</b>		Colegio Suizo Pestalozzi			
<b>Dirección</b>		Av. Ricardo Palma 1450 Urb. La Aurora - Miraflores			
<b>Indicador</b>		Ratio de Utilización Laboral en incidencias			
<b>Periodo</b>		Octubre - Noviembre			
<b>Variable</b>		<b>Indicador</b>	<b>Medida</b>	<b>Instrumento</b>	<b>Formula</b>
Gestión de Incidencias		Ratio de Utilización Laboral en incidencias	Unidades	Ficha de Registro	$RULI = \frac{H}{G}$
<b>Ítems</b>	<b>Ficha de Registro</b>	<b>Horas Invertidas en la Resolución de Incidencias (H)</b>	<b>Horas Disponibles para Atender Incidencias (G)</b>	<b>Ratio de Utilización Laboral en incidencias (RUL)</b>	
1	F-OCT-14	19.3	21	0.92	
2	F-OCT-15	18.54	21	0.88	
3	F-OCT-16	19	21	0.90	
4	F-OCT-19	18.24	21	0.87	
5	F-OCT-20	16.35	21	0.78	
6	F-OCT-21	19.06	21	0.91	
7	F-OCT-22	19	21	0.90	
8	F-OCT-23	18	21	0.86	
9	F-OCT-26	17	21	0.81	
10	F-OCT-27	18.12	21	0.86	
11	F-OCT-28	18	21	0.86	
12	F-OCT-29	19.18	21	0.91	
13	F-OCT-30	20	21	0.95	
14	F-NOV-2	21.18	21	1.01	
15	F-NOV-3	19.09	21	0.91	
16	F-NOV-4	19.3	21	0.92	
17	F-NOV-5	20	21	0.95	
18	F-NOV-6	19	21	0.90	
19	F-NOV-9	19.06	21	0.91	
20	F-NOV-10	18.21	21	0.87	
21	F-NOV-11	19.1	21	0.91	

## ANEXO 12: Tabla de la Distribución Normal Estándar

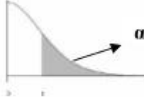
Tabla de valores críticos de la distribución t de Student



**Niveles de Significancia DOS COLA**

	0.500	0.250	0.200	0.100	0.050	0.025	0.020	0.010	0.005
1	1.00	2.41	3.08	6.31	12.71	25.45	31.82	63.66	127.32
2	0.82	1.60	1.89	2.92	4.30	6.21	6.96	9.92	14.09
3	0.76	1.42	1.64	2.35	3.18	4.18	4.54	5.84	7.45
4	0.74	1.34	1.53	2.13	2.78	3.50	3.75	4.60	5.60
5	0.73	1.30	1.48	2.02	2.57	3.16	3.36	4.03	4.77
6	0.72	1.27	1.44	1.94	2.45	2.97	3.14	3.71	4.32
7	0.71	1.25	1.41	1.89	2.36	2.84	3.00	3.50	4.03
8	0.71	1.24	1.40	1.86	2.31	2.75	2.90	3.36	3.83
9	0.70	1.23	1.38	1.83	2.26	2.69	2.82	3.25	3.69
10	0.70	1.22	1.37	1.81	2.23	2.63	2.76	3.17	3.58
11	0.70	1.21	1.36	1.80	2.20	2.59	2.72	3.11	3.50
12	0.70	1.21	1.36	1.78	2.18	2.56	2.68	3.05	3.43
13	0.69	1.20	1.35	1.77	2.16	2.53	2.65	3.01	3.37
14	0.69	1.20	1.35	1.76	2.14	2.51	2.62	2.98	3.33
15	0.69	1.20	1.34	1.75	2.13	2.49	2.60	2.95	3.29
16	0.69	1.19	1.34	1.75	2.12	2.47	2.58	2.92	3.25
17	0.69	1.19	1.33	1.74	2.11	2.46	2.57	2.90	3.22
18	0.69	1.19	1.33	1.73	2.10	2.45	2.55	2.88	3.20
19	0.69	1.19	1.33	1.73	2.09	2.43	2.54	2.86	3.17
20	0.69	1.18	1.33	1.72	2.09	2.42	2.53	2.85	3.15
21	0.69	1.18	1.32	1.72	2.08	2.41	2.52	2.83	3.14
22	0.69	1.18	1.32	1.72	2.07	2.41	2.51	2.82	3.12
23	0.69	1.18	1.32	1.71	2.07	2.40	2.50	2.81	3.10
24	0.68	1.18	1.32	1.71	2.06	2.39	2.49	2.80	3.09
25	0.68	1.18	1.32	1.71	2.06	2.38	2.49	2.79	3.08
26	0.68	1.18	1.31	1.71	2.06	2.38	2.48	2.78	3.07
27	0.68	1.18	1.31	1.70	2.05	2.37	2.47	2.77	3.06
28	0.68	1.17	1.31	1.70	2.05	2.37	2.47	2.76	3.05
29	0.68	1.17	1.31	1.70	2.05	2.36	2.46	2.76	3.04
30	0.68	1.17	1.31	1.70	2.04	2.36	2.46	2.75	3.03
31	0.68	1.17	1.31	1.70	2.04	2.36	2.45	2.74	3.02
32	0.68	1.17	1.31	1.69	2.04	2.35	2.45	2.74	3.01
33	0.68	1.17	1.31	1.69	2.03	2.35	2.44	2.73	3.01
34	0.68	1.17	1.31	1.69	2.03	2.35	2.44	2.73	3.00
35	0.68	1.17	1.31	1.69	2.03	2.34	2.44	2.72	3.00
36	0.68	1.17	1.31	1.69	2.03	2.34	2.43	2.72	2.99
37	0.68	1.17	1.30	1.69	2.03	2.34	2.43	2.72	2.99
38	0.68	1.17	1.30	1.69	2.02	2.33	2.43	2.71	2.98
39	0.68	1.17	1.30	1.68	2.02	2.33	2.43	2.71	2.98
40	0.68	1.17	1.30	1.68	2.02	2.33	2.42	2.70	2.97
	0.250	0.125	0.100	0.050	0.025	0.013	0.010	0.005	0.003

**Niveles de Significancia UNA COLA**



## **DOCUMENTACIÓN DEL SISTEMA**

## Índice

DESARROLLO DE METODOLOGÍA: .....	86
Introducción.....	86
Alcance .....	86
Valores de Trabajo .....	86
Personas y Roles del Proyecto .....	87
Matriz de Impacto.....	87
Product Backlog .....	88
Reunión de Planificación de Sprint (Sprint Planning Meeting) .....	90
Cronograma de Sprint .....	93
1. SPRINT 1:.....	96
1.1. HISTORIAS DE USUARIOS.....	96
1.2. SPRINT BACKLOG .....	103
1.3. TRABAJO DE DESARROLLO DE SPRINT.....	104
1.4. PROTOTIPOS DE DISEÑO DEL SPRINT 1 .....	108
1.5. CODIFICACIÓN DEL SPRINT 1 .....	113
1.6. IMPLEMENTACIÓN DEL SPRINT 1 .....	138
1.7. REVISIÓN DEL SPRINT 1 .....	142
1.8. RETROSPECTIVA DEL SPRINT 1 .....	144
2. SPRINT 2:.....	146
2.1. HISTORIAS DE USUARIOS.....	147
2.2. SPRINT BACKLOG .....	148
2.3. TRABAJO DE DESARROLLO DE SPRINT.....	149
2.4. PROTOTIPOS DE DISEÑO DEL SPRINT 2 .....	153
2.5. CODIFICACIÓN DEL SPRINT 2 .....	155
2.6. IMPLEMENTACIÓN DEL SPRINT 2 .....	167
2.7. REVISION DE SPRINT 2 .....	169
2.8. RETROSPECTIVA DE SPRINT 2 .....	170
3. SPRINT 3:.....	173
3.1. HISTORIAS DE USUARIOS.....	173
3.2. SPRINT BACKLOG .....	175
3.3. TRABAJO DE DESARROLLO DE SPRINT.....	176
3.4. PROTOTIPOS DE DISEÑO DEL SPRINT 3 .....	179
3.5. CODIFICACIÓN DEL SPRINT 3 .....	181
3.6. IMPLEMENTACIÓN DEL SPRINT 3 .....	193
3.7. REVISION DE SPRINT 3 .....	195
3.8. RETROSPECTIVA DE SPRINT 3 .....	196



## Índice de Tablas

Tabla 1: Personas y Roles del Proyecto.....	87
Tabla 2: Tabla de Impacto .....	87
Tabla 3: Product Backlog .....	90
Tabla 4: Sprint Planning Meeting.....	93
Tabla 5: Planificación de Sprint.....	95
Tabla 6: Historia de Usuario 01.....	97
Tabla 7: Historia de Usuario 02.....	98
Tabla 8: Historia de Usuario 03.....	99
Tabla 9: Historia de Usuario 04.....	99
Tabla 10: Historia de Usuario 05 .....	100
Tabla 11: Historia de Usuario 06 .....	101
Tabla 12: Historia de Usuario 07 .....	102
Tabla 13: Historia de Usuario 08.....	103
Tabla 14: Historia de Usuario 09.....	103
Tabla 15: Sprint Backlog de Sprint 1 .....	104
Tabla 16: Resumen del Sprint 1.....	143
Tabla 17: Retrospectiva del Sprint 1 .....	144
Tabla 18: Historia de Usuario 10.....	147
Tabla 19: Historia de Usuario 11.....	148
Tabla 20: Historia de Usuario 12.....	148
Tabla 21: Product Backlog Sprint 2 .....	149
Tabla 22: Resumen del Sprint 2.....	170
Tabla 23: Retrospectiva del Sprint 2 .....	170
Tabla 24: Historia de Usuario N°13.....	173
Tabla 25: Historia de Usuario N°14 .....	174
Tabla 26: Historia de Usuario N°15 .....	174
Tabla 27: Historia de Usuario N°16 .....	175
Tabla 28: Product Backlog Sprint 3 .....	176
Tabla 29: Resumen del Sprint 3.....	195
Tabla 30: Retrospectiva del Sprint 3 .....	196

## Índice de Figuras

Ilustración 1: Caso de Uso de Usuario – Administrador.....	105
Ilustración 2: Diagrama Entidad – Relación de Sprint 1.....	105
Ilustración 3: Modelo Lógico de Sprint 1.....	106
Ilustración 4: Modelo Físico de Sprint 1.....	107
Ilustración 5: Login del Sistema.....	108
Ilustración 6: Mantenimiento Área.....	109
Ilustración 7: Mantenimiento Cargo.....	109
Ilustración 8: Mantenimiento Categoría.....	110
Ilustración 9: Mantenimiento Subcategoría.....	111
Ilustración 10: Mantenimiento Nivel.....	111
Ilustración 11: Mantenimiento Prioridad.....	112
Ilustración 12: Mantenimiento Perfil.....	112
Ilustración 13: Mantenimiento Usuarios.....	113
Ilustración 14: Controlador para Login de Usuarios.....	114
Ilustración 15: Vista (HTML), llamados de Librerías del Login de Usuarios.....	114
Ilustración 16: Vista en Blade (HTML) del Login de Usuarios.....	115
Ilustración 17: Controlador de Vistas de Mantenimiento de Área.....	115
Ilustración 18: Vista en Blade (HTML) de Listado de Área.....	116
Ilustración 19: Modelo de Listado de Área.....	116
Ilustración 20: Vista de nueva New Área.....	116
Ilustración 21: Modelo de registro Área.....	117
Ilustración 22: Modelo getArea.....	117
Ilustración 23: Vista Blade editar Área.....	117
Ilustración 24: Modelo actualización Área.....	117
Ilustración 25: Modelo de Inactivar/Activar Área.....	118
Ilustración 26: Controlador de Vistas de Mantenimiento de Cargo.....	118
Ilustración 27: Vista en Blade (HTML) de Listado de Cargo.....	119
Ilustración 28: Modelo de Listado de Cargo.....	119
Ilustración 29: Vista de nueva New Área.....	119
Ilustración 30: Modelo de registro Cargo.....	120
Ilustración 31: Modelo de getCargo.....	120
Ilustración 32: Vista Blade editar Cargo.....	120
Ilustración 33: Modelo actualización Cargo.....	120
Ilustración 34: Modelo de Inactivar/Activar Cargo.....	121
Ilustración 35: Controlador de Vistas de Mantenimiento de Categoría.....	121
Ilustración 36: Vista en Blade (HTML) de Listado de Categoría.....	121
Ilustración 37: Modelo de Listado de Categoría.....	122
Ilustración 38: Vista de nueva New Categoría.....	122
Ilustración 39: Modelo de registro Categoría.....	122
Ilustración 40: Modelo de get Categoría.....	122
Ilustración 41: Vista Blade editar Categoría.....	123
Ilustración 42: Modelo actualización Categoría.....	123
Ilustración 43: Modelo de Inactivar/Activar Categoría.....	123
Ilustración 44: Controlador de Vistas de Mantenimiento de Subcategoría.....	124
Ilustración 45: Vista en Blade (HTML) de Listado de Subcategoría.....	124
Ilustración 46: Modelo de Listado de Subcategoría.....	125

Ilustración 47: Vista de nueva New Subcategoría.....	125
Ilustración 48: Modelo de registro Subcategoría .....	125
Ilustración 49: Modelo de get Subcategoría.....	126
Ilustración 50: Vista Blade editar Subcategoría.....	126
Ilustración 51: Modelo actualización Subcategoría.....	126
Ilustración 52: Modelo de Inactivar/Activar Subcategoría .....	126
Ilustración 53: Controlador de Vistas de Mantenimiento de Nivel .....	127
Ilustración 54: Vista en Blade (HTML) de Listado de Nivel .....	127
Ilustración 55: Modelo de Listado de Nivel.....	127
Ilustración 56: Vista de nueva New Nivel .....	128
Ilustración 57: Modelo de registro Nivel.....	128
Ilustración 58: Modelo de get Nivel.....	128
Ilustración 59: Vista Blade editar Nivel.....	128
Ilustración 60: Modelo actualización Nivel.....	129
Ilustración 61: Modelo de Inactivar/Activar Nivel.....	129
Ilustración 62: Controlador de Vistas de Mantenimiento de Prioridad .....	129
Ilustración 63: Vista en Blade (HTML) de Listado de Prioridad .....	130
Ilustración 64: Modelo de Listado de Prioridad.....	130
Ilustración 65: Vista de nueva New Prioridad .....	130
Ilustración 66: Modelo de registro Prioridad.....	131
Ilustración 67: Modelo de getPrioridad.....	131
Ilustración 68: Vista Blade editar Prioridad .....	131
Ilustración 69: Modelo actualización Prioridad .....	131
Ilustración 70: Modelo de Inactivar/Activar Prioridad.....	132
Ilustración 71: Controlador de Vistas de Mantenimiento de Perfil .....	132
Ilustración 72: Vista en Blade (HTML) de Listado de Perfil .....	132
Ilustración 73: Modelo de Listado de Perfil .....	133
Ilustración 74: Vista de nueva New Perfil .....	133
Ilustración 75: Modelo de registro Perfil.....	133
Ilustración 76: Modelo de getPerfil .....	133
Ilustración 77: Vista Blade editar Perfil .....	134
Ilustración 78: Modelo actualización Perfil .....	134
Ilustración 79: Modelo de Inactivar/Activar Perfil.....	134
Ilustración 80: Controlador de Vistas de Mantenimiento de Usuarios .....	135
Ilustración 81: Vista en Blade (HTML) de Listado de Usuarios.....	135
Ilustración 82: Modelo de Listado de Usuarios.....	136
Ilustración 83: Vista de nueva New Usuarios .....	136
Ilustración 84: Modelo de registro Usuarios.....	136
Ilustración 85: Modelo de get Usuarios.....	137
Ilustración 86: Vista Blade editar Usuarios.....	137
Ilustración 87: Modelo actualización Usuarios.....	137
Ilustración 88: Modelo de Inactivar/Activar Usuarios.....	138
Ilustración 89: Interfaz del Login de Usuarios.....	138
Ilustración 90: Interfaz de Área .....	139
Ilustración 91: Interfaz de Cargos .....	139
Ilustración 92: Interfaz de Categorías .....	140
Ilustración 93: Interfaz de Sub-Categorías.....	140

Ilustración 94: Interfaz de Nivel.....	141
Ilustración 95: Interfaz de Prioridad.....	141
Ilustración 96: Interfaz de Perfiles.....	142
Ilustración 97: Interfaz de Usuarios.....	142
Ilustración 98: Acta de Reunión de Sprint 1.....	146
Ilustración 99: Caso de Uso de Solicitante Sprint 2.....	150
Ilustración 100: Caso de Uso de Soporte Sprint 2.....	150
Ilustración 101: Caso de Uso de Administrador Sprint 2.....	151
Ilustración 102: Diagrama Entidad – Relación de Sprint 2.....	151
Ilustración 103: Modelo Lógico de Sprint 2.....	152
Ilustración 104: Modelo Físico de la BD Sprint 2.....	153
Ilustración 105: Registro de Solicitud de Tickets.....	154
Ilustración 106: Visualización de Bandeja de Tickets.....	154
Ilustración 107: Seguimiento de Tickets.....	155
Ilustración 108: Controlador de Vistas de Nuevo Ticket.....	155
Ilustración 109: Vista en Blade I (HTML) de Nuevo Ticket.....	156
Ilustración 110: Vista en Blade II (HTML) de Nuevo Ticket.....	156
Ilustración 111: Modelo de la vista de registro de Ticket.....	157
Ilustración 112: Controlador de guardado de Ticket.....	157
Ilustración 113: Modelo I de la Función de registro de Ticket.....	157
Ilustración 114: Modelo II de la Función de registro de Ticket.....	158
Ilustración 115: Controlador de Vistas de Gestión de Ticket.....	158
Ilustración 116: Vista de Bandeja de Ticket.....	159
Ilustración 117: Modelo de Función de Listado de Ticket.....	159
Ilustración 118: Vista I del módulo de Gestión de Ticket.....	160
Ilustración 119: Vista II del módulo de Gestión de Ticket.....	160
Ilustración 120: Vista III del módulo de Gestión de Ticket.....	161
Ilustración 121: Vista IV del módulo de Gestión de Ticket.....	161
Ilustración 122: Vista V del módulo de Gestión de Ticket.....	162
Ilustración 123: Vista VI del módulo de Gestión de Ticket.....	162
Ilustración 124: Modelo de actualización del Nombre del Tema del Ticket.....	163
Ilustración 125: Modelo de asignación de Usuario de Soporte.....	163
Ilustración 126: Modelo de vista de Gestión de Ticket.....	164
Ilustración 127: Modelo de vista II de Gestión de Ticket.....	164
Ilustración 128: Modelo de vista III de Gestión de Ticket.....	165
Ilustración 129: Modelo de Función I de agregar comentario en la Gestión de Ticket.....	165
Ilustración 130: Modelo de Función II de agregar comentario en la Gestión de Ticket.....	166
Ilustración 131: Modelo de Función I de actualización de estado del Ticket.....	166
Ilustración 132: Modelo de Función II de cambio de estado del Ticket.....	167
Ilustración 133: Interfaz de Creación de Nuevo Ticket.....	167
Ilustración 134: Interfaz de la Bandeja de Tickets.....	168
Ilustración 135: Interfaz del desarrollo de la atención del Ticket.....	168
Ilustración 136: Interfaz del proceso de Cerrado de un Ticket.....	169
Ilustración 137: Acta de Reunión de Sprint 2.....	172
Ilustración 138: Caso de Uso de Solicitante Sprint 3.....	176
Ilustración 139: Caso de Uso de Soporte Sprint 3.....	177

Ilustración 140: Caso de Uso de Administrador Sprint 3 .....	177
Ilustración 141: Diagrama Entidad – Relación de Sprint 3 .....	178
Ilustración 142: Modelo Lógico de Sprint 3 .....	178
Ilustración 143: Modelo Físico de la BD .....	179
Ilustración 144: Reporte de Indicador Ratio de Resolución de Incidencias .....	180
Ilustración 145: Reporte de Indicador Ratio de Utilización Laboral de Incidencias .....	180
Ilustración 146: Reporte de Incidencias .....	181
Ilustración 147: Gráficas de incidencias .....	181
Ilustración 148: Controlador de Vista de Reportes .....	182
Ilustración 149: Modelo de la Función de Reportes RRI .....	182
Ilustración 150: Vista del Reporte RRI .....	183
Ilustración 151: Modelo de Función de Reporte RULI .....	183
Ilustración 152: Vista del Reporte RULI .....	184
Ilustración 153: Modelo de Función de Reportes de Incidencias .....	184
Ilustración 154: Vista del Reporte de Incidencias .....	185
Ilustración 155: Vista de la generación y búsqueda de reportes .....	185
Ilustración 156: Vista de Dashboards .....	186
Ilustración 157: Controlador de Dashboards .....	186
Ilustración 158: Modelo I dashboard .....	186
Ilustración 159: Modelo II dashboard .....	187
Ilustración 160: Modelo III Dashboard .....	188
Ilustración 161: Modelo IV Dashboard .....	188
Ilustración 162: Modelo V Dashboard .....	189
Ilustración 163: Modelo VI dashboard .....	189
Ilustración 164: Script I construcción Dashboards .....	190
Ilustración 165: Script II construcción Dashboards .....	191
Ilustración 166: Script III construcción Dashboards .....	191
Ilustración 167: Script IV construcción Dashboards .....	192
Ilustración 168: Script V construcción Dashboards .....	192
Ilustración 169: Script VI construcción Dashboards .....	193
Ilustración 170: Interfaz de Ratio de Resolución de Incidencias .....	193
Ilustración 171: Interfaz de Ratio de Utilización Laboral de Incidencias .....	194
Ilustración 172: Interfaz de Reporte de Incidencias .....	194
Ilustración 173: Interfaz de Visualización de gráficas de incidencias .....	195
Ilustración 174: Acta de Reunión de Sprint 3 .....	197

## **DESARROLLO DE METODOLOGÍA:**

Desarrollo del Sistema Web para la Gestión de Incidencias en el Colegio Suizo Pestalozzi – Miraflores.

### **Introducción**

El presente documento redacta la metodología que se implementará en el desarrollo del Sistema Web para la Gestión de Incidencias en el Colegio Suizo Pestalozzi – Miraflores. Esta Metodología será SCRUM y estará dividida en 3 Sprints.

La metodología SCRUM en la presente tesis se realizará mediante entregables funcionales e incrementales que se irán realizando periódicamente por cada sprint el cual tendrá una duración de 2 a 4 semanas. La metodología llevará a cabo ciertas pautas como parte de su planificación estas servirán como guía y no serán consideradas un reglamento.

### **Alcance**

Teniendo como base lo analizado en los objetivos específicos, el Sistema Web para la Gestión de Incidencias en el Colegio Suizo Pestalozzi – Miraflores deberá cumplir objetivos con mayor prioridad.

- Desarrollar un Sistema Web que permita llevar un control de registro de todas las incidencias ingresadas.
- Desarrollar un Sistema Web que permita visualizar el estado de las incidencias.
- El Sistema Web debe permitir generar reportes mediante gráficos o listados sobre los indicadores expuestos.
- Desarrollar el Sistema Web de manera intuitiva e interactiva de modo que el usuario (docentes / administrador) tenga empatía desde la primera interacción con el sistema.

### **Valores de Trabajo**

Los valores que deben ser practicados por todos los miembros involucrados en el desarrollo y que hacen posible que la metodología SCRUM tenga éxito son:

- Autonomía del equipo.
- Respeto en el equipo.
- Responsabilidad y autodisciplina.
- Foco en la tarea.
- Información, transparencia y visibilidad.

### Personas y Roles del Proyecto

Persona	Contacto	Rol
Dorador Diaz, Luis Felipe	<a href="mailto:luisfelipe210714@gmail.com">luisfelipe210714@gmail.com</a>	Analista /Programador
Alvarado Flores, Carlos Alberto	<a href="mailto:carlos_mat_net@hotmail.com">carlos_mat_net@hotmail.com</a>	Analista /Programador
Estrada Mejía Ricardo, Ernesto	<a href="mailto:restrada@pestalozzi.edu.pe">restrada@pestalozzi.edu.pe</a>	Administrador
Huaranga Gabriel, Jhon Williams	<a href="mailto:jwhuarangag@gmail.com">jwhuarangag@gmail.com</a>	Scrum Master

*Tabla 7: Personas y Roles del Proyecto*

### Matriz de Impacto

Como se observa en la tabla N°2, el matriz de impacto de prioridades donde se detalla por cada usuario el nivel de prioridad

Prioridad	
Muy Alta	1
Alta	2
Media	3
Baja	4
Muy Baja	5

*Tabla 8: Tabla de Impacto*

## Product Backlog

La pila del producto (Product Backlog) que se muestra a continuación describe los requerimientos funcionales, debidamente especificados con su número de historia, prioridad y tiempo estimado, se muestra los requerimientos desde el punto de vista del Product Owner previa coordinación con el Scrum master el desarrollo del listado por prioridad de avance en la implementación y entrega en los futuros Sprint que se vayan a realizar para la investigación.

En la tabla ##, se visualiza el product backlog

Requerimiento Funcionales	Historias	Prioridad	Responsable
<b>RF01:</b> El sistema debe tener una pantalla de inicio de sesión para que puedan ingresar el personal de la empresa	01	Muy Alta	Carlos Alvarado / Luis Dorador
<b>RF02:</b> El sistema deberá permitir realizar el registro, actualización, desactivación y listado de áreas.	02	Muy Alta	Carlos Alvarado / Luis Dorador
<b>RF03:</b> El sistema deberá permitir realizar el registro, actualización, desactivación y listado de Cargo.	03	Muy Alta	Carlos Alvarado / Luis Dorador



<b>RF04:</b> El sistema deberá permitir realizar el registro, actualización, desactivación y listado de Categoría.	04	Muy Alta	Carlos Alvarado / Luis Dorador
<b>RF05:</b> El sistema deberá permitir realizar el registro, actualización, desactivación y listado de Sub Categoría.	05	Muy Alta	Carlos Alvarado / Luis Dorador
<b>RF06:</b> El sistema deberá permitir realizar el registro, actualización, desactivación y listado de Nivel.	06	Muy Alta	Carlos Alvarado / Luis Dorador
<b>RF07:</b> El sistema deberá permitir realizar el registro, actualización, desactivación y listado de Prioridad.	07	Muy Alta	Carlos Alvarado / Luis Dorador
<b>RF08:</b> El sistema deberá permitir realizar el registro, actualización, desactivación y listado de Perfil.	08	Muy Alta	Carlos Alvarado / Luis Dorador
<b>RF09:</b> El sistema deberá permitir realizar el registro, actualización, desactivación y listado de Usuarios.	09	Muy Alta	Carlos Alvarado / Luis Dorador
<b>RF10:</b> El sistema debe permitir realizar la generación de Tickets de incidencias a los docentes y personal de Soporte	10	Alta	Carlos Alvarado / Luis Dorador
<b>RF11:</b> El sistema permitirá realizar el seguimiento a la incidencia generada desde la creación hasta el cierre	11	Alta	Carlos Alvarado / Luis Dorador
<b>RF12:</b> El sistema permitirá visualizar la bandeja con los tickets generados por usuario	12	Alta	Carlos Alvarado / Luis Dorador
<b>RF13:</b> El sistema debe permitir generar el reporte que permita	13	Alta	Carlos Alvarado / Luis Dorador

evidenciar el ratio de resolución de incidencias

<b>RF14:</b> El sistema debe permitir generar el reporte que permita evidenciar el ratio de utilización laboral de incidencias	14	Alta	Carlos Alvarado / Luis Dorador
<b>RF15:</b> El sistema debe permitir visualizar el reporte general de incidencias	15	Alta	Carlos Alvarado / Luis Dorador
<b>RF16:</b> El sistema debe permitir visualizar gráficos de incidencias reportadas en los últimos días.	16	Alta	Carlos Alvarado / Luis Dorador

*Tabla 9: Product Backlog*

## **Reunión de Planificación de Sprint (Sprint Planning Meeting)**

En la reunión de planificación de Sprint se determinó el trabajo a realizar en cada Sprint. El plan se elaboró mediante el trabajo colaborativo del Equipo SCRUM. En la Reunión de Planificación de Sprint se determinaron dos puntos, el primero refiere a lo que se entregará en cada Sprint (Entregables por Sprint) y en el segundo punto se detalla el plan de trabajo necesario para cumplir el objetivo de cada Sprint, ello se ve en el Plan de Trabajo.

Sprint	Requerimiento Funcionales	Historias	Prioridad	Estimación
	<b>RF01:</b> El sistema debe tener una pantalla de inicio de sesión para que puedan ingresar el personal de la empresa	01	Muy Alta	3
	<b>RF02:</b> El sistema deberá permitir realizar el registro, actualización, desactivación y listado de áreas.	02	Muy Alta	1
	<b>RF03:</b> El sistema deberá permitir realizar el registro, actualización, desactivación y listado de Cargo.	03	Muy Alta	1
<b>Sprint 1</b>	<b>RF04:</b> El sistema deberá permitir realizar el registro, actualización, desactivación y listado de Categoría.	04	Muy Alta	1
	<b>RF05:</b> El sistema deberá permitir realizar el registro, actualización, desactivación y listado de Sub Categoría.	05	Muy Alta	1
	<b>RF06:</b> El sistema deberá permitir realizar el registro, actualización, desactivación y listado de Nivel.	06	Muy Alta	1
	<b>RF07:</b> El sistema deberá permitir realizar el registro,	07	Muy Alta	1

actualización, desactivación y listado de Prioridad.

**RF08:** El sistema deberá permitir realizar el registro, actualización, desactivación y listado de Perfil.

08      Muy Alta      1

**RF09:** El sistema deberá permitir realizar el registro, actualización, desactivación y listado de Usuarios.

09      Muy Alta      1

**RF10:** El sistema debe permitir realizar la generación de Tickets de incidencias a los docentes y personal de Soporte

10      Alta      3

**Sprint  
2**

**RF11:** El sistema permitirá realizar el seguimiento a la incidencia generada desde la creación hasta el cierre

11      Alta      3

**RF12:** El sistema permitirá visualizar la bandeja con los tickets generados por usuario

12      Alta      3

**RF13:** El sistema debe permitir generar el reporte que permita evidenciar el ratio de resolución de incidencias

13      Alta      3

**Sprint  
3**

**RF14:** El sistema debe permitir generar el reporte que permita evidenciar el ratio de utilización laboral de incidencias

14      Alta      3

**RF15:** El sistema debe permitir visualizar el reporte general de incidencias

15      Alta      3

**RF16:** El sistema debe permitir visualizar gráficos de incidencias reportadas en los últimos días. 16 Alta 4

*Tabla 10: Sprint Planning Meeting*

**Cronograma de Sprint**

NOMBRE DE TAREA	DURACIÓN	COMIENZO	FIN
<b>Sistema Web para la Gestión de Incidencias en el Colegio Suizo Pestalozzi – Miraflores</b>	<b>44 días</b>	<b>Lun 03/08/2020</b>	<b>Jue 01/10/2020</b>
<b>1. SPRINT 1</b>	<b>13 días</b>	<b>Lun 03/08/2020</b>	<b>Jue 20/08/2020</b>
<b>1.1 TAREAS</b>	<b>11 días</b>	<b>Mie 05/08/2020</b>	<b>Jue 20/08/2020</b>
Login de Usuarios del Sistema			
Mantenimiento de Área			
Mantenimiento de Cargo			
Mantenimiento de Categoría			
Mantenimiento de Subcategoría			
Mantenimiento de Nivel			
Mantenimiento de Prioridad			
Mantenimiento Perfil			
Mantenimiento de Usuarios			
1.2 ANALIS	2 días	Lun 03/08/2020	Mar 04/08/2020
1.3 DISEÑO DE SPRINT 1	1 día	Mie 05/08/2020	Mie 05/08/2020
1.4 CODIFICACION	9 días	Jue 06/08/2020	Mar 18/08/2020

1.5 IMPLEMENTACION	1 día	Mie 19/08/2020	Mie 19/08/2020
<b>2. SPRINT 2</b>	<b>19 días</b>	<b>Jue 20/08 /2020</b>	<b>Mar 15/09 /2020</b>
<b>2.1 TAREAS</b>	<b>17 días</b>	Lun 24/08/2020	Mar 15/09/2020
Registro de Solicitud de Tickets			
Visualización de bandeja de Tickets			
Seguimiento de Tickets			
2.2 ANALISIS	2 días	Jue 20/08/2020	Vie 21/08/2020
2.3 DISEÑO DE SPRINT 2	1 día	Lun 24/08/2020	Lun 24/08/2020
2.4 CODIFICACION	15 días	Mar 25/08/2020	Lun 14/09/2020
2.5 IMPLEMENTACION	1 día	Mar 15/09/2020	Mar 15/09/2020
<b>3. PRINT 3</b>	<b>12 días</b>	<b>Mie 16/09/2020</b>	<b>Jue 1/10/2020</b>
<b>3.1 TAREAS</b>	<b>11 días</b>	Jue 17/09/2020	Jue 01/10/2020
Reporte de Indicador de Ratio de Resolución de Incidencias			
Reporte de Indicador de Ratio de Utilización Laboral en Incidencias			
Reporte de Incidencias			
Visualizar Gráficas de incidencias			
3.2 ANALISIS	1 día	Mie 16/09/2020	Mie 16/09/2020
3.3 DISEÑO DE SPRINT 3	1 día	Jue 17/09/2020	Jue 17/09/2020
3.4 CODIFICACION	9 días	Vie 18/09/2020	Mie 30/09/2020

3.5 IMPLEMENTACION

1 día

Jue 01/10/2020

Jue 01/10/2020

**Cierre de Proyecto**

*Tabla 11: Planificación de Sprint*

## SPRINT BACKLOG (Pila de Sprints)

### 1. SPRINT 1:

#### 1.1. HISTORIAS DE USUARIOS

Según Gallardo (2016), señaló el historial del usuario: Encuentre la forma más rápida de definir requisitos o características Busca desarrollar el enfoque del sistema de red, De forma funcional y finalmente entregada al usuario.

HISTORIA DE USUARIO		PRIORIDAD	Estimado
<b>Numero:</b> 01	<b>Usuario:</b> Administrador / Soporte/ Solicitante	Muy Alta	1 días
<b>Nombre de la Historia:</b> Login de Usuarios en el Sistema			



**Programador Responsable:** Carlos Alvarado / Luis Dorador

**Descripción:** El acceso al Sistema Web se va desarrollar mediante perfiles de Administrador, Soporte y Solicitante (Profesores y Administrativos). El acceso m brindado a cada perfil de usuario al Sistema se dará mediante el ingreso de un usuario y la contraseña que serán almacenados en la base de datos del proyecto.

Como Probarlo:

-El ingreso al sistema dependerá del perfil del usuario.

- El Administrador tendrá acceso general al sistema.
- El personal de Soporte tendrá acceso a la Bandeja de Incidencias, generar reportes y generación de tickets de incidencia.
- El solicitante solo tendrá acceso a generar incidencias y visualización de Tickets generados.

-El acceso al sistema será correcto si las credenciales ingresadas son las correctas.

-Cada perfil de usuario contará con su propio panel y las opciones que puede realizar cada tipo de usuario.

*Tabla 12: Historia de Usuario 01  
Fuente: Elaboración Propio*

HISTORIA DE USUARIO		PRIORIDAD	Estimado
<b>Numero:</b> 02	<b>Usuario:</b> Administrador / Soporte	Muy Alta	1 día
<b>Nombre de la Historia:</b> Mantenimiento de Área			
<b>Programador Responsable:</b> Carlos Alvarado / Luis Dorador			
<b>Descripción:</b> Es importante realizar un mantenimiento de Áreas para identificar el área donde se genera la incidencia. El administrador del Sistema / Soporte puede agregar, editar y colocar en inactivo el Áreas.			
<b>Como Probarlo:</b> - El Administrador / Soporte ingresa al sistema.			

- Ir al menú Administración, luego seleccionar la opción Área.
- Se visualiza el listado de Áreas que fueron ingresados, se podrá ver los botones de Visualizar, Editar, Inactivar.
  - El botón Visualizar, permitirá ver ventana emergente con los datos del Área.
  - El botón Editar, permitirá modificar el nombre del Área.
  - El botón Inactivar, permitirá cambiar a estado inactivo. Es decir el Área no será visible en otras opciones del sistema.
- Se visualizará el botón Agregar en la parte superior derecha, que permitirá visualizar ventana emergente para ingresar registrar nuevas Áreas.

*Tabla 13: Historia de Usuario 02  
Fuente: Elaboración Propio*

HISTORIA DE USUARIO		PRIORIDAD	Estimado
<b>Numero:</b> 03	<b>Usuario:</b> Administrador / Soporte	Muy Alta	1 días
<b>Nombre de la Historia:</b> Mantenimiento de Cargo			
<b>Programador Responsable:</b> Carlos Alvarado / Luis Dorador			
<b>Descripción:</b> Es importante realizar un mantenimiento de los Cargos del personal del Colegio Suizo Pestalozzi, ya que permitirá conocer el cargo que tendrá el usuario que acceda al sistema. El administrador del Sistema / Soporte podrá agregar, editar e inactivar el cargo.			
<b>Como Probarlo:</b> <ul style="list-style-type: none"> <li>- El Administrador / Soporte ingresa al sistema.</li> <li>- Ir al menú Administración, luego seleccionar la opción Cargo.</li> <li>- Se visualizará el listado de Cargos que fueron ingresados, se podrá ver los botones de Visualizar, Editar, Inactivar.           <ul style="list-style-type: none"> <li>• El botón Visualizar, permitirá ver ventana emergente con los datos del cargo.</li> <li>• El botón Editar, permitirá modificar el nombre de cargo.</li> <li>• El botón Inactivar, permitirá cambiar a estado inactivo. Es decir, el cargo no será visible en otras opciones del sistema.</li> </ul> </li> </ul>			

- Se visualizará el botón Agregar en la parte superior derecha, que permitirá visualizar ventana emergente para ingresar registrar nuevos cargos.

*Tabla 14: Historia de Usuario 03*

*Fuente: Elaboración Propio*

HISTORIA DE USUARIO		PRIORIDAD	Estimado
<b>Numero:</b> 04	<b>Usuario:</b> Administrador / Soporte	Muy Alta	1 día
<b>Nombre de la Historia:</b> Mantenimiento de Categoría			
<b>Programador Responsable:</b> Carlos Alvarado / Luis Dorador			
<b>Descripción:</b> Es importante realizar un mantenimiento de Categoría para saber qué tema está relacionado el Incidente. (Hardware, Software, Infraestructura y Otros). El administrador del Sistema / Soporte podrá agregar, editar e inactivar la categoría			
<b>Como Probarlo:</b>			
<ul style="list-style-type: none"> <li>- El Administrador / Soporte ingresa al sistema.</li> <li>- Ir al menú Administración, luego seleccionar la opción Categoría.</li> <li>- Se visualiza el listado de Categorías que fueron ingresados, se podrá ver los botones de Visualizar, Editar, Inactivar. <ul style="list-style-type: none"> <li>• El botón Visualizar, permitirá ver ventana emergente con los datos de la categoría.</li> <li>• El botón Editar, permitirá modificar el nombre de la categoría.</li> <li>• El botón Inactivar, permitirá cambiar a estado inactivo. Es decir la categoría no será visible en otras opciones del sistema.</li> </ul> </li> <li>- Se visualizará el botón Agregar en la parte superior derecha, que permitirá visualizar ventana emergente para ingresar registrar nuevas categorías.</li> </ul>			

*Tabla 15: Historia de Usuario 04*

*Fuente: Elaboración Propio*

HISTORIA DE USUARIO		PRIORIDAD	Estimado
<b>Numero:</b> 05	<b>Usuario:</b> Administrador / Soporte	Muy Alta	1 día
<b>Nombre de la Historia:</b> Mantenimiento de Sub-Categoría			
<b>Programador Responsable:</b> Carlos Alvarado / Luis Dorador			
<b>Descripción:</b> Es importante realizar un mantenimiento de la Sub - Categoría para saber un detalle más específico a lo que estará relacionado el Incidente. El administrador del Sistema/ Soporte puede agregar, editar e inactivar la sub -categoría			
<b>Como Probarlo:</b>			
<ul style="list-style-type: none"> <li>- El Administrador / Soporte ingresa al sistema.</li> <li>- Ir al menú Administración, luego seleccionar la opción Sub-categoría.</li> <li>- Se visualiza el listado de Sub-Categorías que fueron ingresados, se podrá ver los botones de Visualizar, Editar, Inactivar. <ul style="list-style-type: none"> <li>• El botón Visualizar, permitirá ver ventana emergente con los datos de la sub-categoría.</li> <li>• El botón Editar, permitirá modificar el nombre de la sub-categoría.</li> <li>• El botón Inactivar, permitirá cambiar a estado inactivo. Es decir la sub-categoría no será visible en otras opciones del sistema.</li> </ul> </li> <li>- Se visualizará el botón Agregar en la parte superior derecha, que permitirá visualizar ventana emergente para ingresar registrar nuevas sub-categorías.</li> </ul>			

*Tabla 16: Historia de Usuario 05  
Fuente: Elaboración Propio*

HISTORIA DE USUARIO		PRIORIDAD	Estimado
<b>Numero:</b> 06	<b>Usuario:</b> Administrador / Soporte	Muy Alta	1 día
<b>Nombre de la Historia:</b> Mantenimiento de Nivel			
<b>Programador Responsable:</b> Carlos Alvarado / Luis Dorador			

<p><b>Descripción:</b> Es importante realizar un mantenimiento del Nivel, para conocer el nivel de conocimiento para la atención de incidencias. El administrador del Sistema/ Soporte puede agregar, editar e inactivar el nivel.</p>
<p><b>Como Probarlo:</b></p> <ul style="list-style-type: none"> <li>- El Administrador / Soporte ingresa al sistema.</li> <li>- Ir al menú Administración, luego seleccionar la opción Nivel.</li> <li>- Se visualiza el listado de Nivel que fueron ingresados, se podrá ver los botones de Visualizar, Editar, Inactivar. <ul style="list-style-type: none"> <li>• El botón Visualizar, permitirá ver ventana emergente con los datos del Nivel.</li> <li>• El botón Editar, permitirá modificar el nombre del Nivel.</li> <li>• El botón Inactivar, permitirá cambiar a estado inactivo. Es decir, el Nivel no será visible en otras opciones del sistema.</li> </ul> </li> <li>- Se visualizará el botón Agregar en la parte superior derecha, que permitirá visualizar ventana emergente para ingresar registrar nuevos niveles.</li> </ul>

*Tabla 17: Historia de Usuario 06  
Fuente: Elaboración Propio*

HISTORIA DE USUARIO		PRIORIDAD	Estimado
<b>Numero:</b> 07	<b>Usuario:</b> Administrador / Soporte	Muy Alta	1 día
<b>Nombre de la Historia:</b> Mantenimiento de Prioridad			
<b>Programador Responsable:</b> Carlos Alvarado / Luis Dorador			
<b>Descripción:</b> Es importante realizar un mantenimiento de Prioridad del Incidente (Alta, Media y baja) para conocer la prioridad de la Incidencia. El administrador del Sistema / Soporte puede agregar, editar y colocar en inactivo la prioridad.			
<b>Como Probarlo:</b>			
<ul style="list-style-type: none"> <li>- El Administrador / Soporte ingresa al sistema.</li> <li>- Ir al menú Administración, luego seleccionar la opción Prioridad.</li> </ul>			

- Se visualiza el listado de Prioridad que fueron ingresados, se podrá ver los botones de Visualizar, Editar, Inactivar.
  - El botón Visualizar, permitirá ver ventana emergente con los datos de la Prioridad.
  - El botón Editar, permitirá modificar el nombre de la Prioridad.
  - El botón Inactivar, permitirá cambiar a estado inactivo. Es decir la Prioridad no será visible en otras opciones del sistema.
- Se visualizará el botón Agregar en la parte superior derecha, que permitirá visualizar ventana emergente para ingresar registrar nuevas Prioridades.

*Tabla 18: Historia de Usuario 07  
Fuente: Elaboración Propio*

<b>HISTORIA DE USUARIO</b>		<b>PRIORIDAD</b>	<b>Estimado</b>
<b>Numero:</b> 08	<b>Usuario:</b> Administrador / Soporte	Muy Alta	1 día
<b>Nombre de la Historia:</b> Mantenimiento de Perfil			
<b>Programador Responsable:</b> Carlos Alvarado / Luis Dorador			
<b>Descripción:</b> Es importante realizar un mantenimiento del Perfil (Administrador, Soporte, Solicitante). El administrador del Sistema / Soporte puede agregar, editar y colocar en inactivo el nivel.			
<b>Como Probarlo:</b>			
<ul style="list-style-type: none"> <li>- El Administrador / Soporte ingresa al sistema.</li> <li>- Ir al menú Administración, luego seleccionar la opción Perfil.</li> <li>- Se visualiza el listado de Perfil que fueron ingresados, se podrá ver los botones de Visualizar, Editar, Inactivar.               <ul style="list-style-type: none"> <li>• El botón Visualizar, permitirá ver ventana emergente con los datos del Perfil.</li> <li>• El botón Editar, permitirá modificar el nombre del Perfil.</li> <li>• El botón Inactivar, permitirá cambiar a estado inactivo. Es decir el Perfil no será visible en otras opciones del sistema.</li> </ul> </li> <li>- Se visualizará el botón Agregar en la parte superior derecha, que permitirá visualizar ventana emergente para ingresar registrar nuevos perfiles.</li> </ul>			

Tabla 19: Historia de Usuario 08  
Fuente: Elaboración Propio

HISTORIA DE USUARIO		PRIORIDAD	Estimado
<b>Numero:</b> 09	<b>Usuario:</b> Administrador / Soporte	Muy Alta	1 día
<b>Nombre de la Historia:</b> Mantenimiento de Usuarios			
<b>Programador Responsable:</b> Carlos Alvarado / Luis Dorador			
<b>Descripción:</b> Es importante realizar un mantenimiento de Usuarios, desde donde se va reflejar datos del usuario a solicitar como los que atienden, incluyendo al Administrador. El administrador del Sistema / Soporte puede agregar, editar y colocar en inactivo los usuarios.			
<b>Como Probarlo:</b>			
<ul style="list-style-type: none"> <li>- El Administrador / Soporte ingresa al sistema.</li> <li>- Ir al menú Administración, luego seleccionar la opción Usuarios.</li> <li>- Se visualiza el listado de Usuarios que fueron ingresados, se podrá ver los botones de Visualizar, Editar, Inactivar. <ul style="list-style-type: none"> <li>• El botón Visualizar, permitirá ver ventana emergente con los datos del Usuario.</li> <li>• El botón Editar, permitirá modificar el nombre del Usuario.</li> <li>• El botón Inactivar, permitirá cambiar a estado inactivo. Es decir el Usuario no será visible en otras opciones del sistema.</li> </ul> </li> <li>- Se visualizará el botón Agregar en la parte superior derecha, que permitirá visualizar ventana emergente para ingresar registrar nuevos Usuarios.</li> </ul>			

Tabla 20: Historia de Usuario 09  
Fuente: Elaboración Propio

## 1.2. SPRINT BACKLOG

Según Schwaber K. y Sutherland J. (2015, p. 16), Sprint Backlog o The List of Sprint pendiente es el conjunto de elementos seleccionados para Sprint en la lista de productos y el plan para cumplir con los requisitos y lograr Esprintar como meta. La acumulación de Sprint es la predicción del equipo de desarrollo de qué

función estará en el siguiente incremento y el trabajo requerido para entregar la función en el incremento. "llevar a cabo".

Sprint	RF	H.U.	Prioridad	T. Estimado	Responsables
Sprint 1	1	Login de Usuarios del Sistema	Muy Alta	8	Carlos Alvarado / Luis Dorador
	2	Mantenimiento de Área	Muy Alta	8	Carlos Alvarado / Luis Dorador
	3	Mantenimiento de Cargo	Muy Alta	8	Carlos Alvarado / Luis Dorador
	4	Mantenimiento de Categoría	Muy Alta	8	Carlos Alvarado / Luis Dorador
	5	Mantenimiento de Sub-Categoría	Muy Alta	8	Carlos Alvarado / Luis Dorador
	6	Mantenimiento de Nivel	Muy Alta	8	Carlos Alvarado / Luis Dorador
	7	Mantenimiento de Prioridad	Muy Alta	8	Carlos Alvarado / Luis Dorador
	8	Mantenimiento de Perfil	Muy Alta	8	Carlos Alvarado / Luis Dorador
	9	Mantenimiento de Usuarios	Muy Alta	8	Carlos Alvarado / Luis Dorador

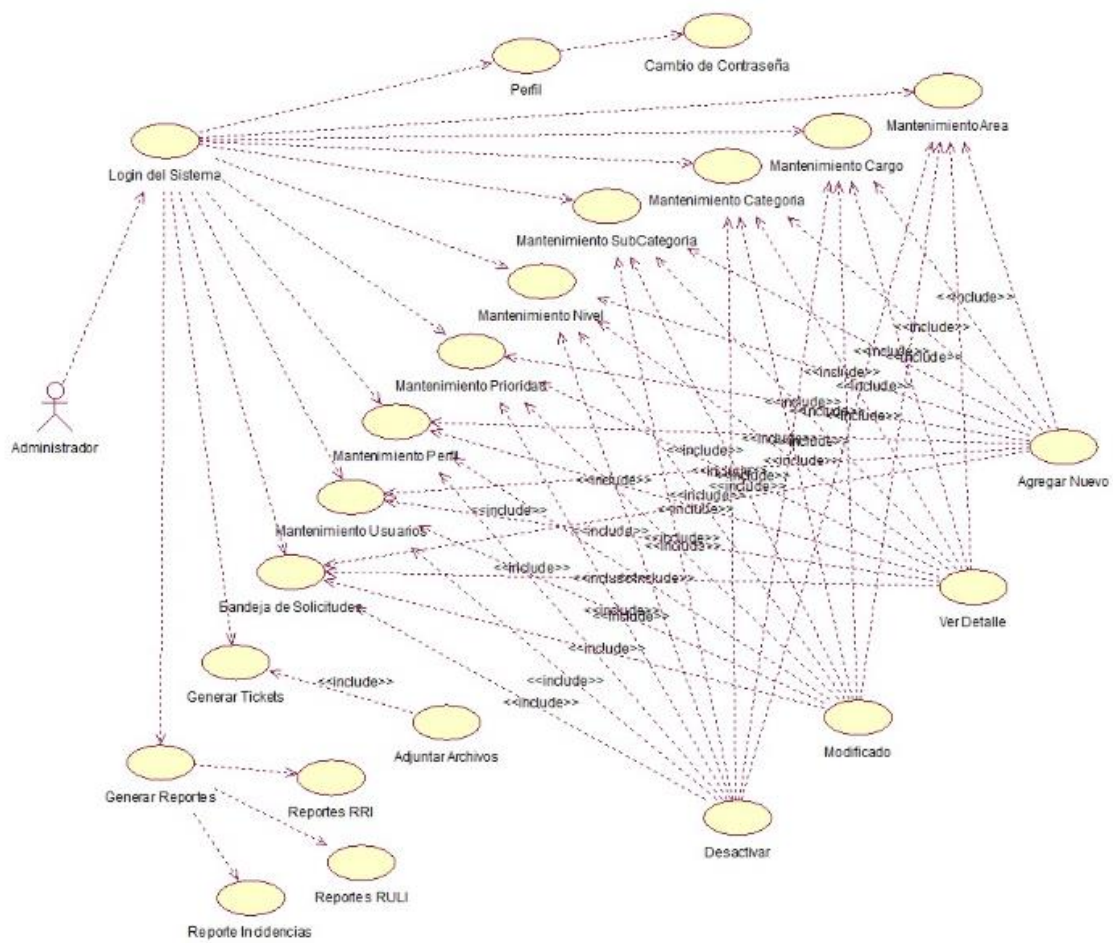
*Tabla 21: Sprint Backlog de Sprint 1*  
Fuente: Elaboración Propio

### 1.3. TRABAJO DE DESARROLLO DE SPRINT

#### 1.3.1. Análisis del Sprint 1

Antes de iniciar la fase de maquetación o diseño, es necesario comprender y Saber exactamente lo que debe hacer el sistema; es decir, analizar Corresponde a la comprensión de historias de usuarios para saber qué se necesita realmente.

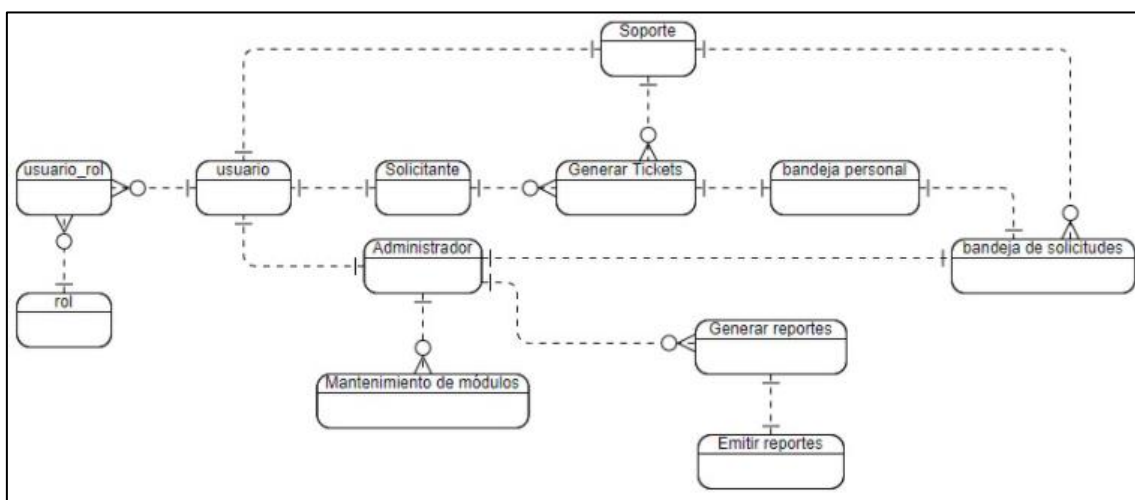




*Ilustración 1: Caso de Uso de Usuario – Administrador  
Fuente: Elaboración Propia*

### 1.3.2. Diseño de Sprint 1

#### a. Diagrama Entidad – Relación de Sprint 1



*Ilustración 2: Diagrama Entidad – Relación de Sprint 1*

Fuente: Elaboración Propia

## b. Modelo Lógico.

Es un prototipo de una base de datos real, por lo que los usuarios pueden verificar los resultados del diseño. De esta forma, podemos verificar el modelo antes de iniciar la construcción física de la base de datos. Este modelo solo se desarrolla en base a Sprint 1 para satisfacer las necesidades del cliente.

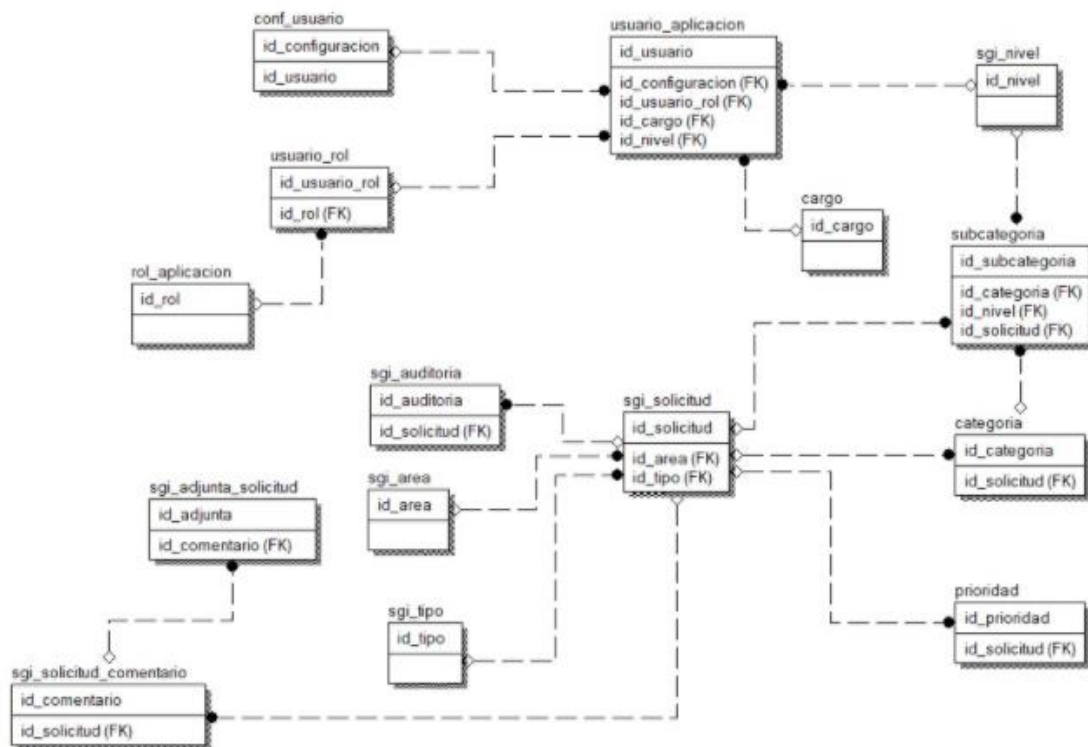
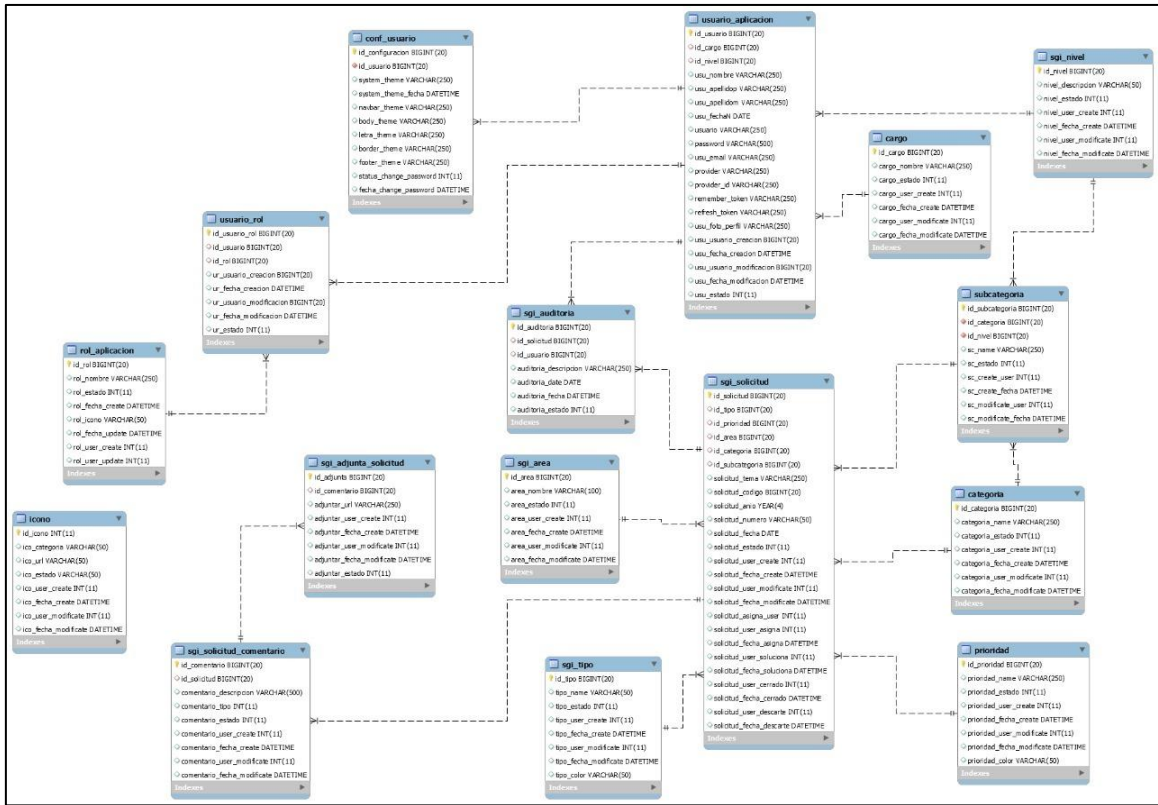


Ilustración 3: Modelo Lógico de Sprint 1  
Fuente: Elaboración Propia

## c. Modelo Físico

Detalla las relaciones básicas, la estructura de almacenamiento y los métodos de acceso que se utilizarán para acceder a los datos de manera eficiente. El diseño de esta relación se debe a una comprensión detallada de todas las funciones proporcionadas por el Sprint 1 (ver Figura XX)

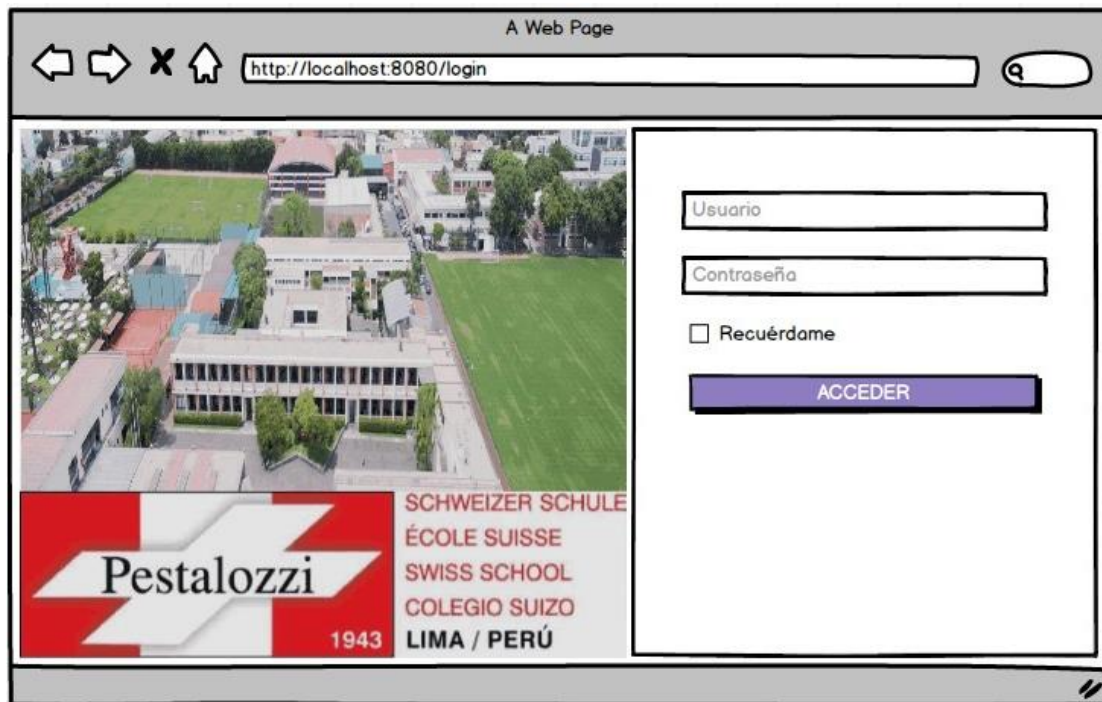


*Ilustración 4: Modelo Físico de Sprint 1*  
Fuente: Elaboración Propia

## 1.4. PROTOTIPOS DE DISEÑO DEL SPRINT 1

### a. Login de Usuarios del Sistema

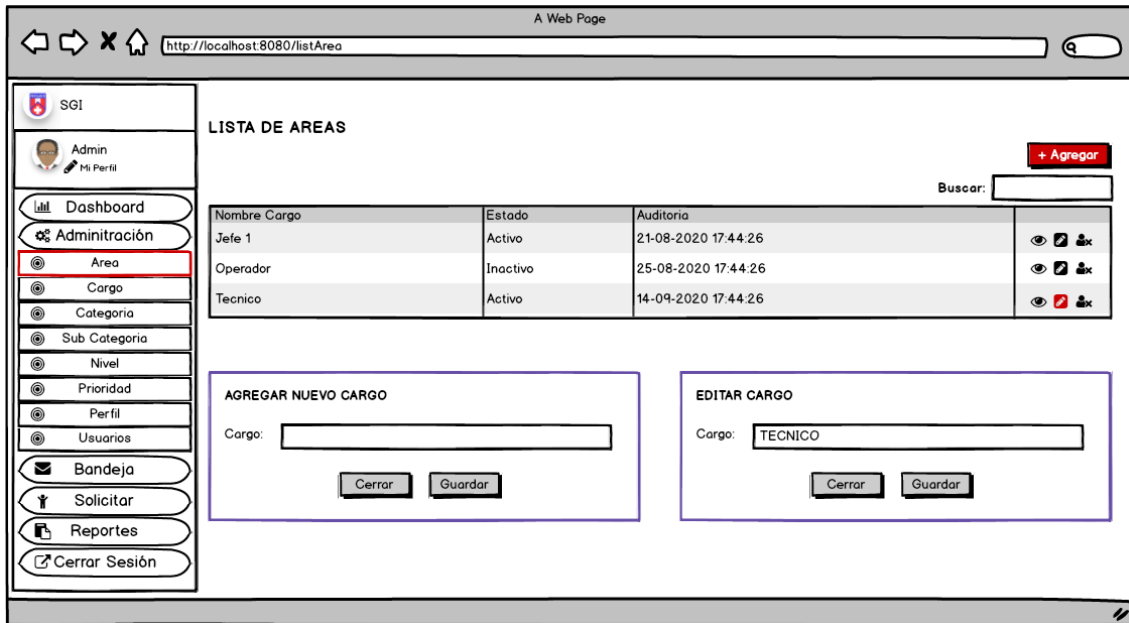
Se presenta un prototipo relacionado con el sistema de inicio de sesión de usuarios. El acceso al sistema se desarrolla en base a tres roles de usuario (ADMINISTRADOR, SOPORTE, SOLICITANTE). Debe ingresar el nombre de usuario y las credenciales de contraseña para ingresar a la plataforma.



*Ilustración 5: Login del Sistema  
Fuente: Elaboración Propia*

### b. Interfaz de Panel Administración - Mantenimiento Área

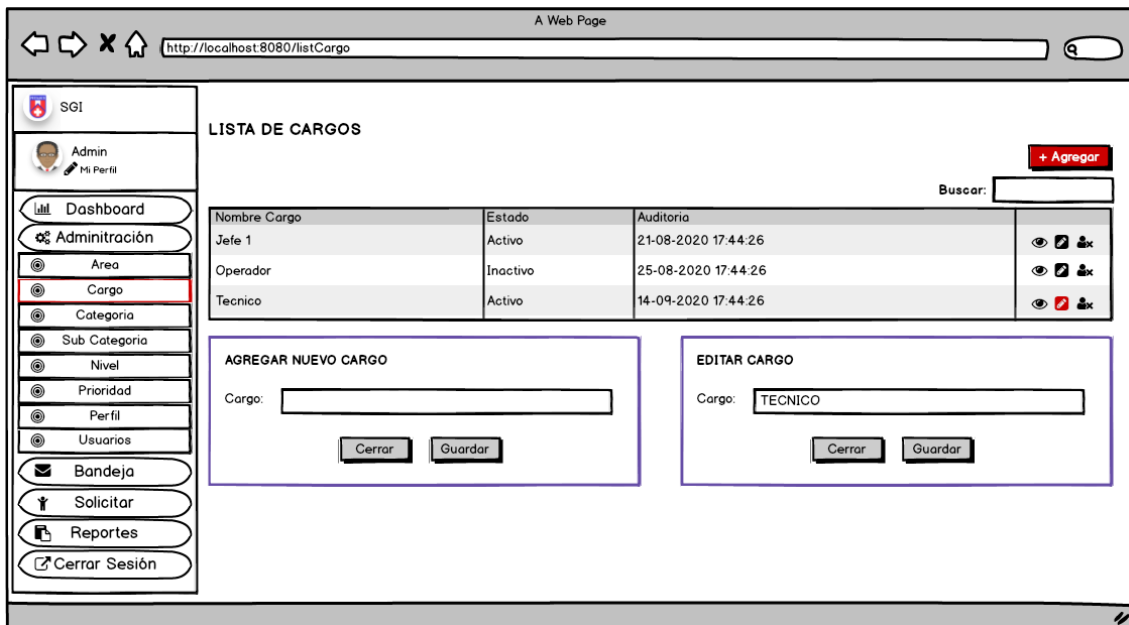
El prototipo detalla el panel de Administración, como primera pestaña tenemos el Área. En el presente caso el Administrador /Soporte pueden visualizar, agregar, editar y colocar inactivo el Área, que estará disponible al crear los usuarios con perfil de Soporte.



*Ilustración 6: Mantenimiento Área  
Fuente: Elaboración Propia*

### c. Interfaz de Panel Administración - Mantenimiento Cargo

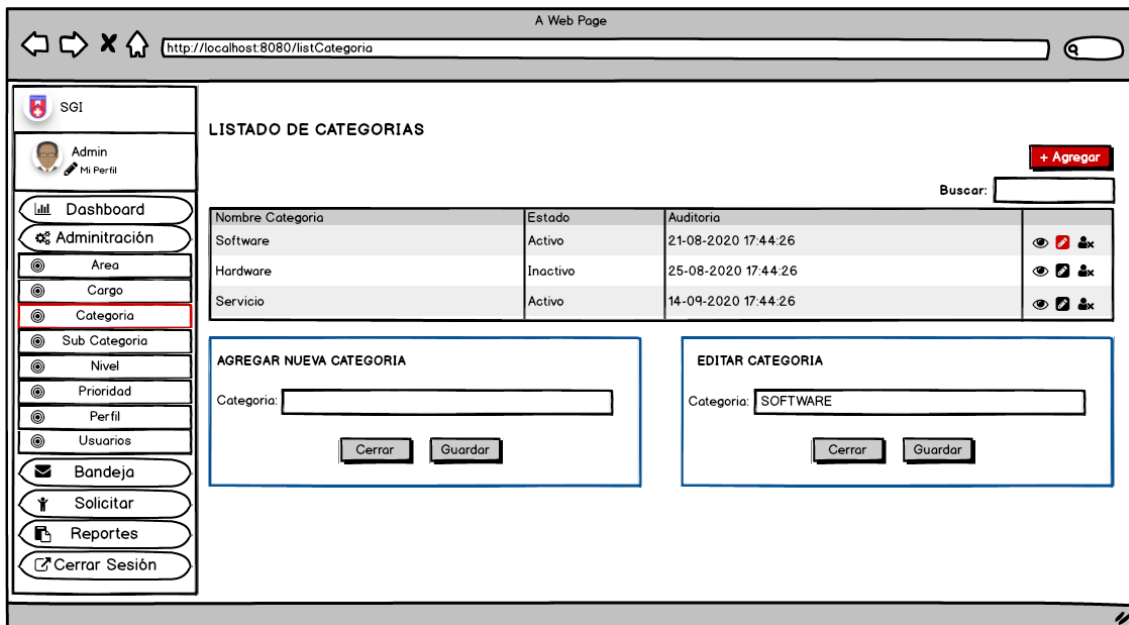
El prototipo detalla el panel de Administración, como segunda pestaña el cargo, en él se agregará los nuevos cargos que serán asignados en la creación de los usuarios.



*Ilustración 7: Mantenimiento Cargo  
Fuente: Elaboración Propia*

#### d. Interfaz de Panel Administración - Mantenimiento Categoría

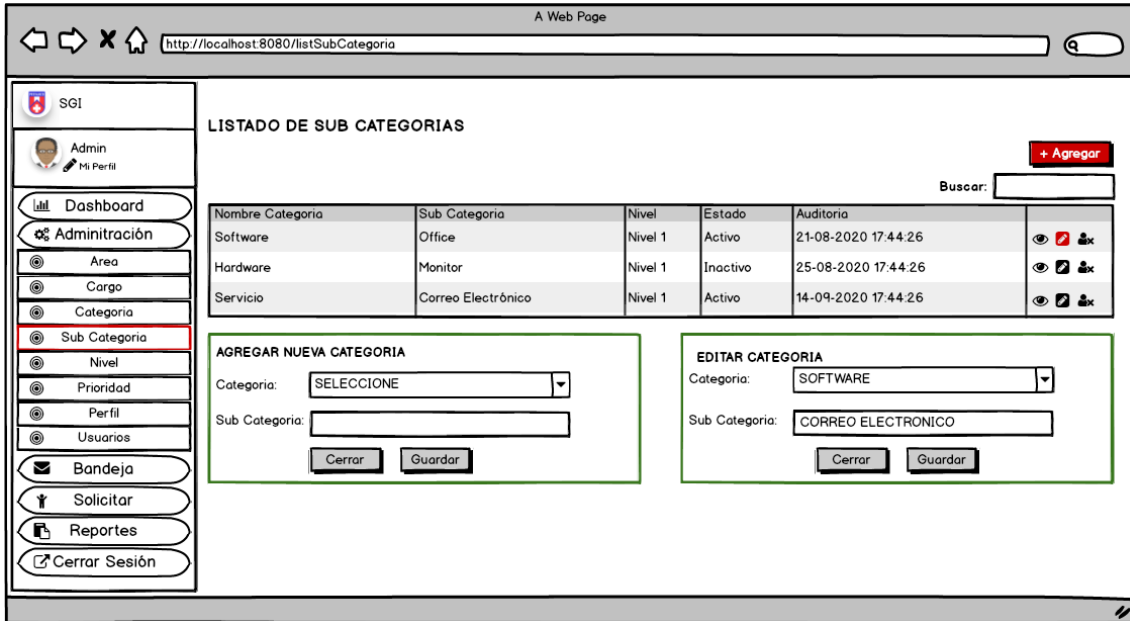
El prototipo detalla el panel de Administración, como tercera pestaña tenemos la Categoría, en él se agregará las categorías para atención de manera general.



*Ilustración 8: Mantenimiento Categoría  
Fuente: Elaboración Propia*

#### e. Interfaz de Panel Administración - Mantenimiento Sub-Categoría

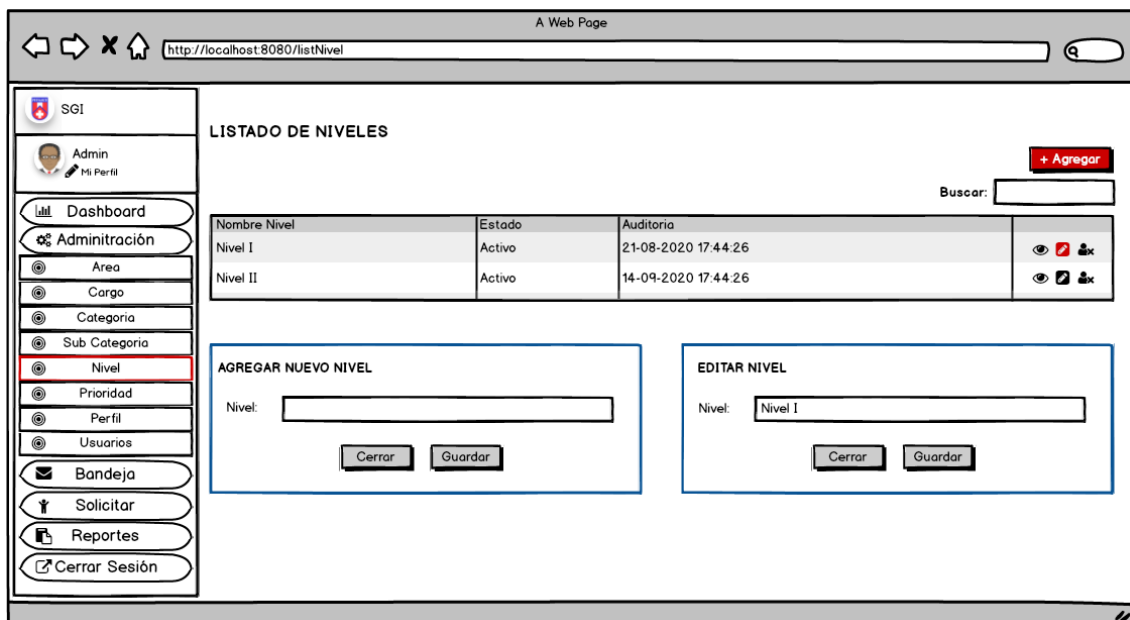
El prototipo detalla el panel de Administración, como cuarta pestaña tenemos la Subcategoría por cada categoría, también se contará para agregar nuevas subcategorías. En el presente caso el Administrador /Soporte pueden visualizar, agregar, editar y colocar inactivo el campo.



*Ilustración 9: Mantenimiento Subcategoría  
Fuente: Elaboración Propia*

#### f. Interfaz de Panel Administración - Mantenimiento Nivel

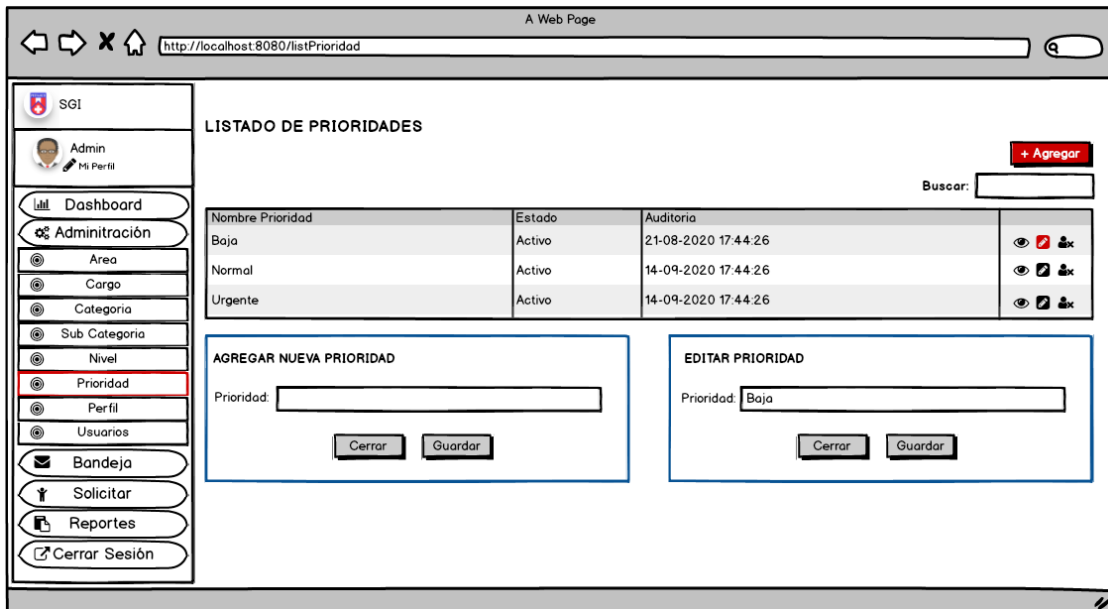
El prototipo detalla el panel de Administración, como quinta pestaña tenemos el Nivel. En el presente caso el Administrador /Soporte pueden visualizar, agregar, editar y colocar inactivo el nivel, que estará disponible al crear los usuarios con perfil de Soporte.



*Ilustración 10: Mantenimiento Nivel  
Fuente: Elaboración Propia*

#### g. Interfaz de Panel Administración -Mantenimiento Prioridad

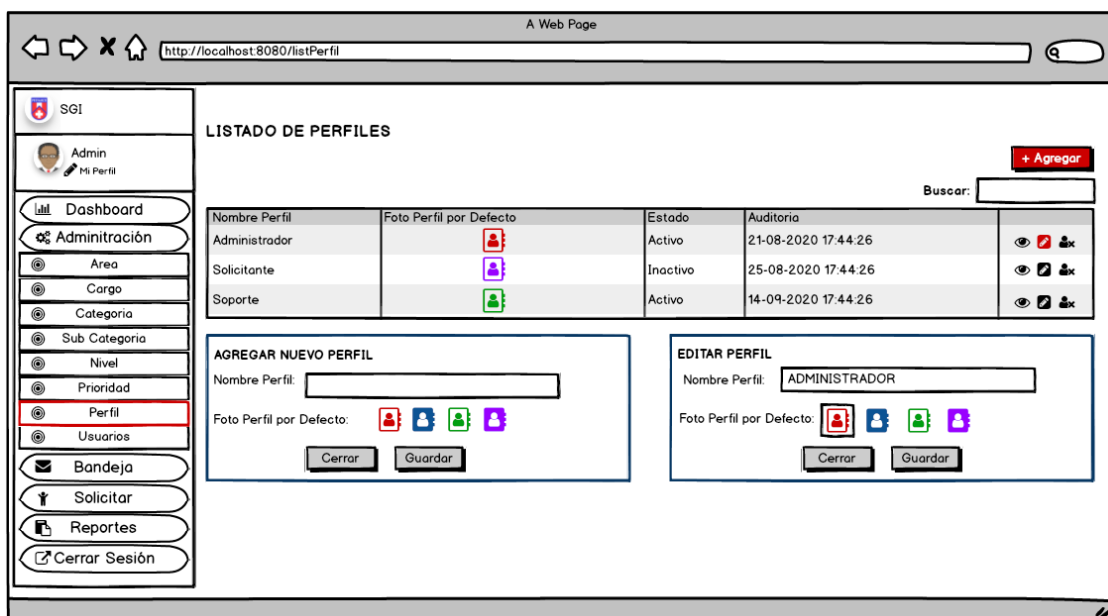
El prototipo detalla el panel de Administración, como sexta pestaña tenemos la Prioridad. En el presente caso el Administrador /Soporte pueden visualizar, agregar, editar y colocar inactivo la prioridad, que estará disponible al solicitar un nuevo Ticket de Incidencia.



*Ilustración 11: Mantenimiento Prioridad  
Fuente: Elaboración Propia*

## h. Interfaz de Panel Administración - Mantenimiento Perfil

El prototipo detalla el panel de Administración, como séptima pestaña tenemos el Perfil. En el presente caso el Administrador /Soporte pueden visualizar, agregar, editar y colocar inactivo el perfil.



*Ilustración 12: Mantenimiento Perfil*



Fuente: Elaboración Propia

## i. Interfaz de Panel Administración - Mantenimiento Usuarios

El prototipo detalla el panel de Administración, como novena pestaña tenemos a Usuarios. En el presente caso el Administrador/Soporte pueden visualizar, agregar, editar y colocar inactivo el usuario.

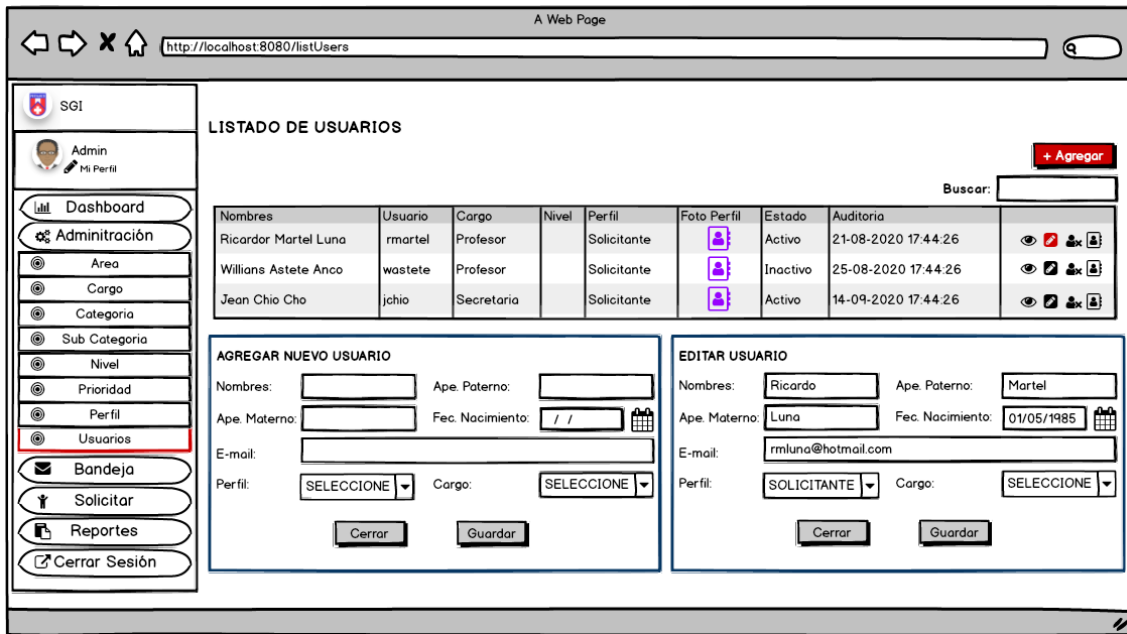


Ilustración 13: Mantenimiento Usuarios  
Fuente: Elaboración Propia

## 1.5. CODIFICACIÓN DEL SPRINT 1

### a. Login de Usuarios.

En la vista, el login de sesión de estos tres tipos de usuarios se realiza a través de una dependencia llamada Spring Security en el marco Spring. El cual configura los perfiles que tendrán cada usuario y las restricciones en las ventanas a las que cada usuario puede acceder y almacenar en la base de datos usando sus perfiles asignados. Después de la verificación, si es correcto, podrá usar su personaje para ingresar al panel correspondiente, de lo contrario verá errores mediante alertas como el siguiente mensaje "DATOS INCORRECTOS"

```

1 <?php
2
3 namespace App\Http\Controllers\Auth;
4
5 use App\Http\Controllers\Controller;
6 use App\Providers\RouteServiceProvider;
7 use Illuminate\Foundation\Auth\AuthenticatesUsers;
8
9 class LoginController extends Controller
10 {
11
12     use AuthenticatesUsers;
13     protected $redirectTo = RouteServiceProvider::HOME;
14
15     /**
16      * Create a new controller instance.
17      *
18      * @return void
19      */
20     public function __construct(){
21         $this->middleware('guest')->except('logout');
22     }
23
24     public function username(){
25         return 'usuario';
26     }
27 }
28
29
30

```

*Ilustración 14: Controlador para Login de Usuarios  
Fuente Elaboración Propia*

```

1 <!DOCTYPE html>
2 <html lang="{{ str_replace('_', '-', app()->getLocale()) }}">
3 <head>
4     <title{{ config('app.name', 'Laravel') }}</title>
5     <meta charset="utf-8">
6     <meta name="viewport" content="width=device-width, initial-scale=1">
7     <meta http-equiv="x-ua-compatible" content="ie=edge">
8     <meta name="csrf-token" content="{{ csrf_token() }}">
9
10    <link rel="stylesheet" type="text/css" href="{{ asset('plugins/bootstrap/css/bootstrap.min.css') }}">
11    <link rel="stylesheet" href="{{ asset('plugins/fontawesome-free/css/all.min.css') }}">
12    <link rel="stylesheet" type="text/css" href="{{ asset('fonts/Linearicons-Free-v1.0.0/icon-font.min.css') }}">
13    <link rel="stylesheet" type="text/css" href="{{ asset('plugins/animate/animate.css') }}">
14    <link rel="stylesheet" type="text/css" href="{{ asset('plugins/css-hamburgers/hamburgers.min.css') }}">
15    <link rel="stylesheet" type="text/css" href="{{ asset('plugins/animation/css/animation.min.css') }}">
16    <link rel="stylesheet" type="text/css" href="{{ asset('plugins/select2/css/select2.min.css') }}">
17    <link rel="stylesheet" type="text/css" href="{{ asset('plugins/daterangepicker/daterangepicker.css') }}">
18    <link rel="stylesheet" type="text/css" href="{{ asset('plugins/toastr/toastr.css') }}">
19    <link rel="stylesheet" type="text/css" href="{{ asset('css/ut11.css') }}">
20    <link rel="stylesheet" type="text/css" href="{{ asset('css/main.css') }}">
21 </head>
22 <body style="background-color: #666666;">
23
24 <div class="limiter">
25     <div class="container-login100">
26         <div class="wrap-login100">
27             <form class="login100-form validate-form" method="POST" action="{{ route('login') }}">
28                 @csrf
29                 <span class="login100-form-title p-b-43">
30                     Iniciar Sesión para Continuar
31                 </span>
32
33                 <div class="wrap-input100 validate-input">
34                     <input class="input100 @error('usuario') is-invalid @enderror" type="text" id="usuario" name="usuario" value="{{ old('usuario') }}" required autocomplete="usuario"
35                         autofocus>
36                     <span class="focus-input100"></span>
37                     <span class="label-input100">{{ __('Usuario') }}</span>
38                 </div>
39
40

```

*Ilustración 15: Vista (HTML), Llamados de Librerías del Login de Usuarios  
Fuente Elaboración Propia*

```

login.blade.php
<div class="limiter">
  <div class="container-login100">
    <div class="wrap-login100">
      <form class="login100-form validate-form" method="POST" action="{{ route('login') }}">
        @csrf
        <span class="login100-form-title p-b-43">
          Iniciar Sesión para Continuar
        </span>
        <div class="wrap-input100 validate-input">
          <input class="input100 @error('usuario') is-invalid @enderror" type="text" id="usuario" name="usuario" value="{{ old('usuario') }}" required autocomplete="usuario" autofocus>
          <span class="focus-input100"></span>
          <span class="label-input100">{{ __('Usuario') }}</span>
        </div>
        <div class="wrap-input100 validate-input" data-validate="Password is required">
          <input class="input100 @error('password') is-invalid @enderror" type="password" id="password" name="password" required autocomplete="current-password">
          <span class="focus-input100"></span>
          <span class="label-input100">{{ __('Password') }}</span>
        </div>
        <div class="flex-ab-m w-full p-t-3 p-b-32">
          <div class="contact100-form-checkbox">
            <input class="input-checkbox100" id="remember" type="checkbox" name="remember" {{ old('remember') ? 'checked' : '' }}>
            <label class="label-checkbox100" for="remember">
              {{ __('Recuérdame') }}
            </label>
          </div>
        </div>
        <div class="container-login100-form-btn">
          <button class="login100-form-btn" type="submit">
            ACCEDER
          </button>
        </div>
        </form>
        <div class="login100-more" style="background-image: url('img/logo-login.png');">
        </div>
      </div>
    </div>
  </div>

```

*Ilustración 16: Vista en Blade (HTML) del Login de Usuarios  
Fuente Elaboración Propia*

## **b. Mantenimiento Área**

En la vista de Mantenimiento de Área, el administrador/soporte, podrá visualizar el Área donde puede suceder la incidencia. El administrador /Soporte podrán agregar, editar, y colocar en inactivo un Área según se necesiten para las atenciones.

```

1 | /php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use App\Models\Area;
7
8 class AreaController extends Controller{
9
10 public function __construct(){
11     $this->middleware('auth');
12 }
13
14 public function listArea(){
15     $area = Area::getListArea();
16     return view('setting.area.listArea',compact('area'));
17 }
18
19 public function areaNew(){
20     return view('setting.area.newArea');
21 }
22
23 public function viewArea(){
24     $area = Area::getArea();
25     return view('setting.area.viewArea',compact('area'));
26 }
27
28 public function editArea(){
29     $area = Area::getArea();
30     return view('setting.area.editArea',compact('area'));
31 }
32 }
33
34 }

```

*Ilustración 17: Controlador de Vistas de Mantenimiento de Área  
Fuente Elaboración Propia*

```

ListArea.blade.php
<div class="row">
<div class="col-md-12">
<div class="card ({ Auth:user()->getCookie()->body_items.' ' .Auth:user()->getCookie()->border_theme })">
<div class="card-header ({ Auth:user()->getCookie()->border_theme })">
<h3 class="card-title">Áreas</h3>
<div class="card-body">
<div class="card-body">
<div class="row">
<div class="col text-right">
<button type="button" class="btn btn-tool" data-card-widget="collapse" data-toggle="tooltip" title="Collapse">
<span class="fas fa-minus"/></button>
</div>
</div>
<div class="card-body">
<div class="row">
<div class="col text-right">
<button type="button" class="btn btn-success btn-sm text-white sb-3" id="btnAddArea" data-toggle="modal" data-target="#modalArea"><i class="fas fa-plus"/></i> Agregar</div>
</div>
</div>
<table class="table table-striped projects display nowrap" id="tblArea" width="100%">
<thead>
<tr>
<th style="width: 15%;></th>
<th style="width: 40%;>Nombre Área</th>
<th style="width: 10%;></th>
<th style="width: 20%;></th>
<th style="width: 25%;></th>
</tr>
</thead>
<tbody>
<tr>
<td style="width: 15%;></td>
<td style="width: 40%;></td>
<td style="width: 10%;></td>
<td style="width: 20%;></td>
<td style="width: 25%;></td>
</tr>
</tbody>
</table>
</div>
</div>
</div>

```

**Ilustración 18:** Vista en Blade (HTML) de Listado de Área  
Fuente Elaboración Propia

```

public static function getListArea(){
$area = Area::select('sgi_area.*', 'u.usu_email as correo_creador', 'u.usu_nombre as nombre_creador', 'u.usu_apellido as ap_creador', 'u.usu_apellido as am_creador', 'ua.usu_email as correo_actualizador', 'ua.usu_nombre as nombre_actualizador', 'ua.usu_apellido as ap_actualizador', 'ua.usu_apellido as am_actualizador')
->join('usuario aplicacion as u', 'u.id_usuario', 'sgi_area.area_user_create')
->leftJoin('usuario aplicacion as ua', 'ua.id_usuario', 'sgi_area.area_user_modificate')
->get();

foreach ($area as $key => $value) {
if($value->area_estado == 1){
$value->nombre_estado = 'ACTIVO';
$value->color_estado = 'success';
}else{
$value->nombre_estado = 'INACTIVO';
$value->color_estado = 'danger';
}

if(empty($value->area_user_modificate)){
$value->nombre_auditoria = $value->nombre_creador.' '.$value->ap_creador.' '.$value->am_creador;
$value->fecha_auditoria = date("d-m-Y", strtotime($value->area_fecha_create));
$value->hora_auditoria = date("H:i:s", strtotime($value->area_fecha_create));
}else{
$value->nombre_auditoria = $value->nombre_actualizador.' '.$value->ap_actualizador.' '.$value->am_actualizador;
$value->fecha_auditoria = date("d-m-Y", strtotime($value->area_fecha_modificate));
$value->hora_auditoria = date("H:i:s", strtotime($value->area_fecha_modificate));
}
}
return $area;
}

```

**Ilustración 19:** Modelo de Listado de Área  
Fuente Elaboración Propia

```

1 <form id="addArea">
2 <meta type="hidden" name="csrf-token" content="{ { csrf_token() } }">
3 <div class="row">
4 <div class="col-xs-12 col-sm-12">
5 <div class="form-group">
6 <label>Área: <span class="text-danger">*</span></label>
7 <input type="text" onkeypress="return /[a-z-ñ ]/.test(event.key)" class="form-control input-sm req" id="area_nombre" name="area_nombre">
8 <input type="hidden" maxlength="150" class="form-control req" id="area_user_create" name="area_user_create" value="{ { Auth:user()->id_usuario } }">
9 </div>
10 </div>
11 </div>
12 </form>
13 <script>
14
15 </script>

```

**Ilustración 20:** Vista de nueva New Área  
Fuente Elaboración Propia

```

52 public static function areaSave(){
53     try {
54         $area_existe = Area::where('area_nombre',strtoupper(request()->area_nombre))->first();
55         if(empty($area_existe)){
56             $area = new Area();
57             $area->area_user_create = request()->area_user_create;
58             $area->area_fecha_create = now();
59             $area->area_nombre = strtoupper(request()->area_nombre);
60             $area->area_estado = 1;
61             $area->save();
62             return array("error"=>0,"mensaje"=>"Se agregó exitosamente el Área ".$area->area_nombre,"tipo"=>"success","title"=>"¡Bien!");
63         }else{
64             return array("error"=>1,"mensaje"=>"El Área ".strtoupper(request()->area_nombre)." que intenta registrar ya existe","tipo"=>"warning","title"=>"¡Oops!");
65         }
66     } catch (\Exception $e) {
67         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
68     }
69 }

```

*Ilustración 21: Modelo de registro Área  
Fuente Elaboración Propia*

```

42 public static function getArea(){
43     $area = Area::where('id_area',request()->id_area)->first();
44     return $area;
45 }

```

*Ilustración 22: Modelo getArea  
Fuente Elaboración Propia*

```

1 <form id="editArea">
2 <meta type="hidden" name="csrf-token" content="{{ csrf_token() }}">
3 <div class="row">
4 <div class="col-xs-12 col-sm-12">
5 <div class="form-group">
6 <label>Área: <span class="text-danger">*</span></label>
7 <input type="text" onkeypress="return /[a-z-ñ ]/i.test(event.key)" class="form-control input-sm req" id="area_nombre" name="area_nombre" value="{{ $area->area_nombre }}">
8 <input type="hidden" maxlength="150" class="form-control req" id="area_user_modificate" name="area_user_modificate" value="{{ Auth::user()->id_usuario }}">
9 <input type="hidden" maxlength="150" class="form-control req" id="id_area" name="id_area" value="{{ $area->id_area }}">
10 </div>
11 </div>
12 </div>
13 </form>
14

```

*Ilustración 23: Vista Blade editar Área  
Fuente Elaboración Propia*

```

71 public static function areaUpdate(){
72     try {
73         $area_existe = Area::where('area_nombre',strtoupper(request()->area_nombre))->where('id_area','!=',request()->id_area)->first();
74         if(empty($area_existe)){
75             $area = Area::find(request()->id_area);
76             $area->area_user_modificate = request()->area_user_modificate;
77             $area->area_fecha_modificate = now();
78             $area->area_nombre = strtoupper(request()->area_nombre);
79             $area->save();
80             return array("error"=>0,"mensaje"=>"Se modificó exitosamente el Área ".$area->area_nombre,"tipo"=>"success","title"=>"¡Bien!");
81         }else{
82             return array("error"=>1,"mensaje"=>"El Área ".strtoupper(request()->area_nombre)." que intenta registrar ya existe","tipo"=>"warning","title"=>"¡Oops!");
83         }
84     } catch (\Exception $e) {
85         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
86     }
87 }

```

*Ilustración 24: Modelo actualización Área  
Fuente Elaboración Propia*

```

80 public static function changeStateArea(){
81     try {
82         $area = Area::find(request()->id_area);
83         if($area->area_estado == 1){
84             $area->area_estado = 0;
85             $mensaje = 'Se inactivo exitosamente el Área ' . $area->area_nombre;
86         }else{
87             $area->area_estado = 1;
88             $mensaje = 'Se activo exitosamente el Área ' . $area->area_nombre;
89         }
90         $area->area_user_modificate = request()->id_user;
91         $area->area_fecha_modificate = now();
92         $area->save();
93         return array("error"=>0,"mensaje"=>$mensaje,"tipo"=>"success", "title"=>"¡Bien!");
94     } catch (\Exception $e) {
95         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente", "tipo"=>"error", "titulo"=>"¡Oops!");
96     }
97 }

```

*Ilustración 25: Modelo de Inactivar/Activar Área  
Fuente Elaboración Propia*

### **c. Mantenimiento de Cargo**

En la vista de Mantenimiento Cargo, el administrador/soporte podrá visualizar los cargos existentes. El administrador /Soporte podrán agregar, editar, y colocar en inactivo el cargo según las indicaciones de la Institución.

```

1  /?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use App\Models\Cargo;
7
8  class JobController extends Controller{
9
10     public function __construct(){
11         $this->middleware('auth');
12     }
13
14     public function listJob(){
15         $cargo = Cargo::getListCargo();
16         return view('setting.cargo.listCargo',compact('cargo'));
17     }
18
19     public function jobNew(){
20         return view('setting.cargo.newCargo');
21     }
22
23     public function viewJob(){
24         $cargo = Cargo::getCargo();
25         return view('setting.cargo.viewCargo',compact('cargo'));
26     }
27
28     public function editJob(){
29         $cargo = Cargo::getCargo();
30         return view('setting.cargo.editCargo',compact('cargo'));
31     }
32 }
33 }

```

*Ilustración 26: Controlador de Vistas de Mantenimiento de Cargo  
Fuente Elaboración Propia*

```

45 <table class="table table-striped projects display nowrap" id="tblCargo" width="100%">
46 <thead>
47 <tr>
48 <th style="width: 15%"></th>
49 <th style="width: 65%">Nombre Cargo</th>
50 <th style="width: 10%" class="text-center">Estado</th>
51 <th style="width: 20%" class="text-right">Auditoria</th>
52 <th style="width: 20%"></th>
53 </tr>
54 </thead>
55 <tbody>
56 @foreach ($cargo as $key => $value)
57 <tr>
58 <td></td>
59 <td>{{ $value->cargo_nombre }}</td>
60 <td class="text-center"><span class="badge badge-{{ $value->color_estado }}">{{ $value->nombre_estado }}</span></td>
61 <td class="text-right">
62 <small>
63 <div style="display: flex; justify-content: space-between; font-size: 0.8em; margin: 0; padding: 0 5px;">
64 <span>{{ $value->fecha_auditoria }}</span>
65 <span>{{ $value->hora_auditoria }}</span>
66 </div>
67 <small>
68 <div style="display: flex; justify-content: space-between; font-size: 0.8em; margin: 0; padding: 0 5px;">
69 <span>{{ $value->nombre_auditoria }}</span>
70 </div>
71 </small>
72 </td>
73 <td class="project-actions text-right">
74 <div style="display: flex; justify-content: space-between; font-size: 0.8em; margin: 0; padding: 0 5px;">
75 <span><a href="#" class="btn btn-primary btn-sm text-white btn-visualizar" title="Visualizar" data-id="{{ $value->id_cargo }}"></span>
76 <span><a href="#" class="fas fa-eye"></span>
77 </div>
78 <div style="display: flex; justify-content: space-between; font-size: 0.8em; margin: 0; padding: 0 5px;">
79 <span><a href="#" class="btn btn-info btn-sm text-white btn-edit" title="Editar" data-id="{{ $value->id_cargo }}"></span>
80 <span><a href="#" class="fas fa-pencil-alt"></span>
81 </div>
82 <div style="display: flex; justify-content: space-between; font-size: 0.8em; margin: 0; padding: 0 5px;">
83 <span><a href="#" class="btn btn-{{ $value->cargo_estado == 0?'success':'danger' }}" btn-sm text-white btn-cargar" title="{{ ($value->cargo_estado == 0?'Activar':'Inactivar' }}" data-id="{{ $value->id_cargo }}"></span>
84 <span><a href="#" class="fas fa-exchange-alt"></span>
85 </div>
86 </td>
87 </tr>
88 @endforeach
89 </tbody>
90 </table>

```

*Ilustración 27: Vista en Blade (HTML) de Listado de Cargo  
Fuente Elaboración Propia*

```

14 public static function getListCargo(){
15     $cargo = Cargo::select('cargo.', 'u.usu_email as correo_creador', 'u.usu_nombre as nombre_creador', 'u.usu_apellido_p as ap_creador', 'u.usu_apellido_m as am_creador', 'ua.usu_email as correo_actualizador', 'ua.usu_nombre as nombre_actualizador', 'ua.usu_apellido_p as ap_actualizador', 'ua.usu_apellido_m as am_actualizador')
16     ->join('usuario_aplicacion as u', 'u.id_usuario', 'cargo.cargo_user_create')
17     ->leftJoin('usuario_aplicacion as ua', 'ua.id_usuario', 'cargo.cargo_user_modifycate')
18     ->get();
19
20     foreach ($cargo as $key => $value) {
21         if($value->cargo_estado == 1){
22             $value->nombre_estado = 'ACTIVO';
23             $value->color_estado = 'success';
24         }else{
25             $value->nombre_estado = 'INACTIVO';
26             $value->color_estado = 'danger';
27         }
28
29         if(empty($value->cargo_user_modifycate)){
30             $value->nombre_auditoria = $value->nombre_creador.' '.$value->ap_creador.' '.$value->am_creador;
31             $value->fecha_auditoria = date("d-m-Y", strtotime($value->cargo_fecha_create));
32             $value->hora_auditoria = date("H:i:s", strtotime($value->cargo_fecha_create));
33         }else{
34             $value->nombre_auditoria = $value->nombre_actualizador.' '.$value->ap_actualizador.' '.$value->am_actualizador;
35             $value->fecha_auditoria = date("d-m-Y", strtotime($value->cargo_fecha_modifycate));
36             $value->hora_auditoria = date("H:i:s", strtotime($value->cargo_fecha_modifycate));
37         }
38     }
39     return $cargo;
40 }

```

*Ilustración 28: Modelo de Listado de Cargo  
Fuente Elaboración Propia*

```

1 <form id="addCargo">
2 <meta type="hidden" name="csrf-token" content="{{ csrf_token() }}">
3 <div class="row">
4 <div class="col-xs-12 col-sm-12">
5 <div class="form-group">
6 <label>Cargo: <span class="text-danger">*</span></label>
7 <input type="text" onkeypress="return /[a-z-ñ ]/i.test(event.key)" class="form-control input-sm req" id="cargo_nombre" name="cargo_nombre">
8 <input type="hidden" maxLength="150" class="form-control req" id="cargo_user_create" name="cargo_user_create" value="{{ Auth::user()->id_usuario }}">
9 </div>
10 </div>
11 </div>
12 </form>

```

*Ilustración 29: Vista de nueva New Área  
Fuente Elaboración Propia*

```

43 public static function cargoSave(){
44     try {
45         $cargo_existe = Cargo::where('cargo_nombre',strtoupper(request()->cargo_nombre))->first();
46         if(empty($cargo_existe)){
47             $cargo = new Cargo();
48             $cargo->cargo_user_create = request()->cargo_user_create;
49             $cargo->cargo_fecha_create = now();
50             $cargo->cargo_nombre = strtoupper(request()->cargo_nombre);
51             $cargo->cargo_estado = 1;
52             $cargo->save();
53             return array("error"=>0,"mensaje"=>"Se agregó exitosamente el Cargo ".$cargo->cargo_nombre,"tipo"=>"success","title"=>"¡Bien!");
54         }else{
55             return array("error"=>1,"mensaje"=>"El Cargo ".$cargo->cargo_nombre." que intenta registrar ya existe","tipo"=>"warning","title"=>"¡Oops!");
56         }
57     } catch (\Exception $e) {
58         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
59     }
60 }

```

*Ilustración 30: Modelo de registro Cargo  
Fuente Elaboración Propia*

```

99 public static function getCargo(){
100     $cargo = Cargo::where('id_cargo',request()->id_cargo)->first();
101     return $cargo;
102 }

```

*Ilustración 31: Modelo de getCargo  
Fuente Elaboración Propia*

```

1 <form id="editCargo">
2 <meta type="hidden" name="csrf-token" content="{{ csrf_token() }}">
3 <div class="row">
4 <div class="col-xs-12 col-sm-12">
5 <div class="form-group">
6 <label>Cargo: <span class="text-danger">*</span></label>
7 <input type="text" onkeypress="return /[a-z-â ]/i.test(event.key)" class="form-control input-sm req" id="cargo_nombre" name="cargo_nombre" value="{{ $cargo->cargo_nombre }}">
8 <input type="hidden" maxlength="150" class="form-control req" id="cargo_user_modificate" name="cargo_user_modificate" value="{{ Auth::user()->id_usuario }}">
9 <input type="hidden" maxlength="150" class="form-control req" id="id_cargo" name="id_cargo" value="{{ $cargo->id_cargo }}">
10 </div>
11 </div>
12 </div>
13 </form>

```

*Ilustración 32: Vista Blade editar Cargo  
Fuente Elaboración Propia*

```

62 public static function cargoUpdate(){
63     try {
64         $cargo_existe = Cargo::where('cargo_nombre',strtoupper(request()->cargo_nombre))->where('id_cargo','!=',request()->id_cargo)->first();
65         if(empty($cargo_existe)){
66             $cargo = Cargo::find(request()->id_cargo);
67             $cargo->cargo_user_modificate = request()->cargo_user_modificate;
68             $cargo->cargo_fecha_modificate = now();
69             $cargo->cargo_nombre = strtoupper(request()->cargo_nombre);
70             $cargo->save();
71             return array("error"=>0,"mensaje"=>"Se modificó exitosamente el Cargo ".$cargo->cargo_nombre,"tipo"=>"success","title"=>"¡Bien!");
72         }else{
73             return array("error"=>1,"mensaje"=>"El Cargo ".$cargo->cargo_nombre." que intenta registrar ya existe","tipo"=>"warning","title"=>"¡Oops!");
74         }
75     } catch (\Exception $e) {
76         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
77     }
78 }

```

*Ilustración 33: Modelo actualización Cargo  
Fuente Elaboración Propia*



```

80 public static function changeStateJob(){
81     try {
82         $cargo = Cargo::find(request()->id_cargo);
83         if($cargo->cargo_estado == 1){
84             $cargo->cargo_estado = 0;
85             $mensaje = 'Se inactivo exitosamente el Cargo '.$cargo->cargo_nombre;
86         }else{
87             $cargo->cargo_estado = 1;
88             $mensaje = 'Se activo exitosamente el Cargo '.$cargo->cargo_nombre;
89         }
90         $cargo->cargo_user_modificate = request()->id_user;
91         $cargo->cargo_fecha_modificate = now();
92         $cargo->save();
93         return array("error"=>0,"mensaje"=>$mensaje,"tipo"=>"success","title"=>"¡Bien!");
94     } catch (\Exception $e) {
95         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
96     }
97 }

```

*Ilustración 34: Modelo de Inactivar/Activar Cargo  
Fuente Elaboración Propia*

#### **d. Mantenimiento de Categoría**

En la vista de Mantenimiento Categoría, el administrador/soporte, podrá visualizar las categorías existentes. El administrador /Soporte podrán agregar, editar, y colocar en inactivo las categorías según se necesiten para las atenciones.

```

14 public function listCategoria(){
15     $categoria = Categoria::getListCategoria();
16     return view('setting_categoria.listCategoria',compact('categoria'));
17 }
18
19 public function categoriaNew(){
20     return view('setting_categoria.newCategoria');
21 }
22
23 public function viewCategoria(){
24     $categoria = Categoria::getCategoria();
25     return view('setting_categoria.viewCategoria',compact('categoria'));
26 }
27
28 public function editCategoria(){
29     $categoria = Categoria::getCategoria();
30     return view('setting_categoria.editOldCategoria',compact('categoria'));
31 }
32 }

```

*Ilustración 35: Controlador de Vistas de Mantenimiento de Categoría  
Fuente Elaboración Propia*

```

33 <table class="table table-striped projects display nowrap" id="tblcategoria" width="100%">
34     <thead>
35         <tr>
36             <th style="width: 15%">#</th>
37             <th style="width: 40%">Nombre Categoría</th>
38             <th style="width: 30%" class="text-center">Estado</th>
39             <th style="width: 20%" class="text-right">Auditoria</th>
40         </tr>
41     </thead>
42     <tbody>
43         @foreach ($categoria as $key => $value)
44             <tr>
45                 <td>#</td>
46                 <td>{{ $value->categoria_nombre }}</td>
47                 <td class="text-center"><span class="badge badge-{{ $value->color_estado }}">{{ $value->nombre_estado }}</span></td>
48                 <td class="text-right">
49                     <small>
50                         <br>
51                         <small>{{ $value->fecha_auditoria }}</small>
52                         <br>
53                         <small>{{ $value->hora_auditoria }}</small>
54                         <br>
55                         <small>{{ $value->nombre_auditoria }}</small>
56                     </small>
57                 </td>
58                 <td class="project-actions text-right">
59                     <a class="btn btn-primary btn-sm text-white btn-visualize" title="Visualizar" data-id="{{ $value->id_categoria }}"></a>
60                     <a class="fas fa-pencil-alt"></a>
61                     <a class="fas fa-info btn-sm text-white btn-edit" title="Editar" data-id="{{ $value->id_categoria }}"></a>
62                     <a class="fas fa-exchange-alt"></a>
63                     <a class="btn btn-{{ ($value->categoria_estado == 0)?'success':'danger' }} btn-sm text-white btn-change" title="{{ ($value->categoria_estado == 0)?'Activar':'Inactivar' }}" data-id="{{ $value->id_categoria }}" data-user="{{ Auth::user()->id_usuario }}"></a>
64                 </td>
65             </tr>
66         @endforeach
67     </tbody>
68 </table>

```

*Ilustración 36: Vista en Blade (HTML) de Listado de Categoría  
Fuente Elaboración Propia*

```

34 public static function getListCategoria(){
35     $categoria = Categoria::select('categoria.*','u.usu_email as correo_creador','u.usu_nombre as nombre_creador','u.usu_apellido as ap_creador','u.usu_email as
36     correo_actualizador','ua.usu_nombre as nombre_actualizador','ua.usu_apellido as ap_actualizador','ua.usu_email as am_creador','ua.usu_email as
37     correo_actualizador','ua.usu_nombre as nombre_actualizador','ua.usu_apellido as ap_actualizador','ua.usu_email as am_creador','ua.usu_email as
38     correo_actualizador')
39     ->join('usuario aplicacion as u','u.id_usuario','categoria.categoria_user_create')
40     ->leftJoin('usuario aplicacion as ua','ua.id_usuario','categoria.categoria_user_modificate')
41     ->get();
42
43     foreach($categoria as $key => $value){
44         if($value->categoria_estado == 1){
45             $value->nombre_estado = 'ACTIVO';
46             $value->color_estado = 'success';
47         }else{
48             $value->nombre_estado = 'INACTIVO';
49             $value->color_estado = 'danger';
50         }
51
52         if(empty($value->categoria_user_modificate)){
53             $value->nombre_auditoria = $value->nombre_creador." ".$value->ap_creador." ".$value->am_creador;
54             $value->fecha_auditoria = date("d-m-Y",strtotime($value->categoria_fecha_create));
55             $value->hora_auditoria = date("H:i:s",strtotime($value->categoria_fecha_create));
56         }else{
57             $value->nombre_auditoria = $value->nombre_actualizador." ".$value->ap_actualizador." ".$value->am_actualizador;
58             $value->fecha_auditoria = date("d-m-Y",strtotime($value->categoria_fecha_modificate));
59             $value->hora_auditoria = date("H:i:s",strtotime($value->categoria_fecha_modificate));
60         }
61     }
62
63     return $categoria;
64 }

```

*Ilustración 37: Modelo de Listado de Categoría  
Fuente Elaboración Propia*

```

1 <form id="addCategoria">
2 <meta type="hidden" name="csrf-token" content="{{ csrf_token() }}">
3 <div class="row">
4 <div class="col-xs-12 col-sm-12">
5 <div class="form-group">
6 <label>Categoría: <span class="text-danger">*</span></label>
7 <input type="text" onkeypress="return /[a-z-ñ ]/i.test(event.key)" class="form-control input-sm req" id="categoria_name" name="categoria_name">
8 <input type="hidden" maxLength="150" class="form-control req" id="categoria_user_create" name="categoria_user_create" value="{{ Auth::user()->id_usuario }}">
9 </div>
10 </div>
11 </div>
12 </form>

```

*Ilustración 38: Vista de nueva New Categoría  
Fuente Elaboración Propia*

```

40 public static function categoriaSave(){
41     try {
42         $categoria_existe = Categoria::where('categoria_name',strtoupper(request()->categoria_name))->first();
43         if(empty($categoria_existe)){
44             $categoria = new Categoria();
45             $categoria->categoria_user_create = request()->categoria_user_create;
46             $categoria->categoria_fecha_create = now();
47             $categoria->categoria_name = strtoupper(request()->categoria_name);
48             $categoria->categoria_estado = 1;
49             $categoria->save();
50             return array("error"=>0,"mensaje"=>"Se agregó exitosamente la Categoría ".$categoria->categoria_name,"tipo"=>"success","title"=>"¡Bien!");
51         }else{
52             return array("error"=>1,"mensaje"=>"La Categoría ".strtoupper(request()->categoria_name)." que intenta registrar ya existe","tipo"=>"warning","title"=>"¡Oops!");
53         }
54     } catch (\Exception $e) {
55         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
56     }
57 }

```

*Ilustración 39: Modelo de registro Categoría  
Fuente Elaboración Propia*

```

43 public static function getCategory(){
44     $categoria = Categoria::where('id_categoria',request()->id_categoria)->first();
45     return $categoria;
46 }

```

*Ilustración 40: Modelo de get Categoría  
Fuente Elaboración Propia*

```

1 <form id="editCategoria">
2 <meta type="hidden" name="csrf-token" content="{{ csrf_token() }}">
3 <div class="row">
4 <div class="col-xs-12 col-sm-12">
5 <div class="form-group">
6 <label>Categoria: <span class="text-danger">*</span></label>
7 <input type="text" onkeypress="return /[a-z-ñ ]/.test(event.key)" class="form-control input-sm req" id="categoria_name" value="{{ $categoria->categoria_name }}" name="categoria_name" />
8 <input type="hidden" maxLength="150" class="form-control req" id="categoria_user_modificate" name="categoria_user_modificate" value="{{ Auth::user()->id_usuario }}">
9 <input type="hidden" maxLength="150" class="form-control req" id="id_categoria" name="id_categoria" value="{{ $categoria->id_categoria }}">
10 </div>
11 </div>
12 </div>
13 </form>

```

*Ilustración 41: Vista Blade editar Categoría  
Fuente Elaboración Propia*

```

67 public static function categoriaUpdate(){
68     try {
69         $categoria_existe = Categoria::where('categoria_name',strtoupper(request()->categoria_name))->where('id_categoria','!=',request()->id_categoria)->first();
70         if(empty($categoria_existe)){
71             $categoria = Categoria::find(request()->id_categoria);
72             $categoria->categoria_user_modificate = request()->categoria_user_modificate;
73             $categoria->categoria_fecha_modificate = now();
74             $categoria->categoria_name = strtoupper(request()->categoria_name);
75             $categoria->save();
76             return array("error"=>0,"mensaje"=>"Se actualizó exitosamente la Categoría ".$categoria->categoria_name,"tipo"=>"success","title"=>"¡Bien!");
77         }else{
78             return array("error"=>1,"mensaje"=>"La Categoría ".strtoupper(request()->categoria_name)." que intenta registrar ya existe","tipo"=>"warning","title"=>"¡Oops!");
79         }
80     } catch (\Exception $e) {
81         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
82     }
83 }

```

*Ilustración 42: Modelo actualización Categoría  
Fuente Elaboración Propia*

```

85 public static function changeStateCategoria(){
86     try {
87         $categoria = Categoria::find(request()->id_categoria);
88         if($categoria->categoria_estado == 1){
89             $categoria->categoria_estado = 0;
90             $mensaje = 'Se inactivo exitosamente la Categoría '.$categoria->categoria_name;
91         }else{
92             $categoria->categoria_estado = 1;
93             $mensaje = 'Se activo exitosamente la Categoría '.$categoria->categoria_name;
94         }
95         $categoria->categoria_user_modificate = request()->id_usuario;
96         $categoria->categoria_fecha_modificate = now();
97         $categoria->save();
98         return array("error"=>0,"mensaje"=>$mensaje,"tipo"=>"success","title"=>"¡Bien!");
99     } catch (\Exception $e) {
100         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
101     }
102 }

```

*Ilustración 43: Modelo de Inactivar/Activar Categoría  
Fuente Elaboración Propia*

## f. Mantenimiento de Subcategoría

En la vista de Mantenimiento Subcategoría, el administrador/soporte, podrá visualizar las Subcategorías existentes. El administrador /Soporte podrán agregar, editar, y colocar en inactivo las subcategorías según se necesiten para las atenciones.

```

public function listSubCategoria(){
    $subcategoria = Categoria::listSubCategoria();
    if(empty($subcategoria->subcategoria)){
        return redirect('/listCategoria');
    }
    return view('setting.subcategoria.listSubCategoria',compact('subcategoria'));
}

public function categoriaSubNew(){
    $categoria = Categoria::where('categoria_estado',1)
        ->orderBy('id_categoria','asc')
        ->get();

    $nivel = Nivel::where('nivel_estado',1)
        ->orderBy('nivel_descripcion','asc')
        ->get();

    return view('setting.subcategoria.newSubCategoria',compact('categoria','nivel'));
}

public function editSubCategoria(){
    $subcategoria = Categoria::getSubCategoria();
    $categoria = Categoria::where('categoria_estado',1)
        ->orderBy('id_categoria','asc')
        ->get();

    $nivel = Nivel::where('nivel_estado',1)
        ->orderBy('nivel_descripcion','asc')
        ->get();

    return view('setting.subcategoria.editSubCategoria',compact('subcategoria','categoria','nivel'));
}

public function viewSubCategoria(){
    $subcategoria = Categoria::getSubCategoria();
    $categoria = Categoria::where('categoria_estado',1)
        ->orderBy('id_categoria','asc')
        ->get();

    $nivel = Nivel::where('nivel_estado',1)
        ->orderBy('nivel_descripcion','asc')
        ->get();

    return view('setting.subcategoria.viewSubCategoria',compact('subcategoria','categoria','nivel'));
}

```

**Ilustración 44: Controlador de Vistas de Mantenimiento de Subcategoría**  
Fuente Elaboración Propia

```

<table class="table table-striped projects display nowrap" id="tblSubCategoria" width="100%">
  <thead>
    <tr>
      <th style="width: 15%;>#</th>
      <th style="width: 30%;>Categoria</th>
      <th style="width: 30%;>Sub Categoria</th>
      <th style="width: 10%;>Nivel</th>
      <th style="width: 10%; class="text-center">Estado</th>
      <th style="width: 20%; class="text-right">Auditoria</th>
      <th style="width: 20%;></th>
    </tr>
  </thead>
  <tbody>
    <#foreach ($subcategoria->subcategoria as $key => $value)>
      <tr>
        <td#</td>
        <td>{{ $value->categoria_name }}</td>
        <td>{{ $value->sub_categoria_name }}</td>
        <td>{{ $value->nivel_descripcion }}</td>
        <td class="text-center"><span class="badge badge-{{ $value->color_estado }}">{{ $value->nombre_estado }}</span></td>
        <td class="text-right">
          <small>
            <div>{{ $value->fecha_auditoria }}</div>
            <div>{{ $value->hora_auditoria }}</div>
            <div>
              <div>{{ $value->nombre_auditoria }}</div>
            </div>
          </small>
        </td>
        <td class="project-actions text-right">
          <a class="btn btn-primary btn-sm text-white btn-visualize" title="Visualizar" data-id="{{ $value->id_subcategoria }}">
            <i class="fas fa-eye"></i>
          </a>
          <a class="btn btn-info btn-sm text-white btn-edit" title="Editar" data-id="{{ $value->id_subcategoria }}">
            <i class="fas fa-pencil-alt"></i>
          </a>
          <a class="btn btn-{{ ($value->sc_estado == 0)?'success':'danger' }}" btn-sm text-white btn-change" title="{{ ($value->sc_estado == 0)?'Activar':'Inactivar' }}" data-id="{{ $value->id_subcategoria }}">
            <i class="fas fa-exchange-alt"></i>
          </a>
        </td>
      </tr>
    </#foreach>
  </tbody>
</table>

```

**Ilustración 45: Vista en Blade (HTML) de Listado de Subcategoría**  
Fuente Elaboración Propia

```

294 public static function listSubCategoria(){
295     $subcategoria = new SubCategoria();
296     $subcategoria->categoria = Categoria::where('categoria_estado',1)->get();
297     $subcategoria->subcategoria = SubCategoria::select('subcategoria.*','u.usu_email as correo_creador','u.usu_nombre as nombre_creador','u.usu_apellido as ap_creador','u.usu_apellido as am_creador','u.usu_email as
298     correo_actualizador','u.usu_nombre as nombre_actualizador','u.usu_apellido as ap_actualizador','u.usu_apellido as am_actualizador','c.categoria_name','n.nivel_description')
299     ->join('categoria as c','c.id_categoria','subcategoria.id_categoria')
300     ->join('usuario_aplicacion as u','u.id_usuario','subcategoria.sc_create_user')
301     ->join('scf_nivel as n','n.id_nivel','subcategoria.id_nivel')
302     ->leftJoin('usuario_aplicacion as ua','ua.id_usuario','subcategoria.sc_modificate_user')
303     ->get();
304
305     foreach($subcategoria->subcategoria as $key => $value){
306         if($value->sc_estado == 1){
307             $value->nombre_estado = 'ACTIVO';
308             $value->color_estado = 'success';
309         }else{
310             $value->nombre_estado = 'INACTIVO';
311             $value->color_estado = 'danger';
312         }
313
314         if(empty($value->sc_modificate_user)){
315             $value->nombre_auditoria = $value->nombre_creador.' '.$value->ap_creador.' '.$value->am_creador;
316             $value->fecha_auditoria = date("d-m-Y",strtotime($value->sc_create_fecha));
317             $value->hora_auditoria = date("H:i:s",strtotime($value->sc_create_fecha));
318         }else{
319             $value->nombre_auditoria = $value->nombre_actualizador.' '.$value->ap_actualizador.' '.$value->am_actualizador;
320             $value->fecha_auditoria = date("d-m-Y",strtotime($value->sc_modificate_fecha));
321             $value->hora_auditoria = date("H:i:s",strtotime($value->sc_modificate_fecha));
322         }
323     }
324     return $subcategoria;
325 }

```

**Ilustración 46: Modelo de Listado de Subcategoría**  
Fuente Elaboración Propia

```

1 <form id="addSubCategoria">
2 <meta type="hidden" name="csrf-token" content="{{ csrf_token }}">
3 <div class="row">
4 <div class="col-xs-12 col-sm-12">
5 <div class="form-group">
6 <label>Categoría: <span class="text-danger">*</span></label>
7 <select class="form-control req" id="id_categoria" name="id_categoria">
8 <option value="0">SELECCIONE</option>
9 @foreach ($categoria as $key => $value)
10 <option value="{{ $value->id_categoria }}">{{ $value->categoria_name }}</option>
11 @endforeach
12 </select>
13 </div>
14 </div>
15 <div class="col-xs-12 col-sm-12">
16 <div class="form-group">
17 <label>Sub Categoría: <span class="text-danger">*</span></label>
18 <input type="text" onkeypress="return /[a-z-ñ ]/.test(event.key)" class="form-control input-sm req" id="sc_name" name="sc_name">
19 <input type="hidden" maxlength="150" class="form-control req" id="sc_create_user" name="sc_create_user" value="{{ Auth::user()->id_usuario }}">
20 </div>
21 </div>
22 <div class="col-xs-12 col-sm-12">
23 <div class="form-group">
24 <label>Nivel: <span class="text-danger">*</span></label>
25 <select type="text" class="form-control input-sm req" id="id_nivel" name="id_nivel">
26 <option value="0">SELECCIONE</option>
27 @foreach ($nivel as $key => $value)
28 <option value="{{ $value->id_nivel }}">{{ $value->nivel_descripcion }}</option>
29 @endforeach
30 </select>
31 </div>
32 </div>
33 </div>
34 </form>

```

**Ilustración 47: Vista de nueva New Subcategoría**  
Fuente Elaboración Propia

```

136 public static function subcategoriaSave(){
137     try {
138         $subcategoria_existe = SubCategoria::where('sc_name',strtoupper(request()->sc_name))->first();
139         if(empty($subcategoria_existe)){
140             $subcategoria = new SubCategoria();
141             $subcategoria->id_categoria = request()->id_categoria;
142             $subcategoria->id_nivel = request()->id_nivel;
143             $subcategoria->sc_create_user = request()->sc_create_user;
144             $subcategoria->sc_create_fecha = now();
145             $subcategoria->sc_name = strtoupper(request()->sc_name);
146             $subcategoria->sc_estado = 1;
147             $subcategoria->save();
148             return array("error"=>0,"mensaje"=>"Se agregó exitosamente la Sub Categoría ".$subcategoria->sc_name,"tipo"=>"success","title"=>"¡Bien!");
149         }else{
150             return array("error"=>1,"mensaje"=>"La Sub Categoría ".strtoupper(request()->sc_name)." que intenta registrar ya existe","tipo"=>"warning","title"=>"¡Oops!");
151         }
152     } catch (\Exception $e) {
153         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
154     }
155 }

```

**Ilustración 48: Modelo de registro Subcategoría**  
Fuente Elaboración Propia

```

176
177 public static function getSubCategoria(){
178     $subcategoria = SubCategoria::where('id_subcategoria',request()->id_subcategoria)->first();
179     return $subcategoria;
180 }
181

```

**Ilustración 49: Modelo de get Subcategoría**  
Fuente Elaboración Propia

```

1 <form id="editSubCategoria">
2     <meta type="hidden" name="csrf-token" content="{{ csrf_token }}">
3     <div class="row">
4         <div class="col-xs-12 col-sm-12">
5             <div class="form-group">
6                 <label>Categoría: <span class="text-danger">*</span></label>
7                 <select class="form-control req" id="id_categoria" name="id_categoria">
8                     <option value="0">SELECCIONE</option>
9                     @foreach ($categorias as $key => $value)
10                        <option value="{{ $value->id_categoria }}" {{ ($subcategoria->id_categoria == $value->id_categoria) ? 'selected' : '' }}>{{ $value->categoria_name }}</option>
11                    @endforeach
12                </select>
13            </div>
14        </div>
15        <div class="col-xs-12 col-sm-12">
16            <div class="form-group">
17                <label>Sub Categoría: <span class="text-danger">*</span></label>
18                <input type="text" onkeypress="return /[a-zñ ]/.test(event.key)" class="form-control input-sm req" id="sc_name" value="{{ $subcategoria->sc_name }}" name="sc_name">
19                <input type="hidden" maxlength="150" class="form-control req" id="sc_modificate_user" name="sc_modificate_user" value="{{ Auth::user()->id_usuario }}">
20                <input type="hidden" maxlength="150" class="form-control req" id="id_subcategoria" name="id_subcategoria" value="{{ $subcategoria->id_subcategoria }}">
21            </div>
22        </div>
23        <div class="col-xs-12 col-sm-12">
24            <div class="form-group">
25                <label>Nivel: <span class="text-danger">*</span></label>
26                <select type="text" class="form-control input-sm req" id="id_nivel" name="id_nivel">
27                    <option value="0">SELECCIONE</option>
28                    @foreach ($niveles as $key => $value)
29                        <option value="{{ $value->id_nivel }}" {{ ($subcategoria->id_nivel == $value->id_nivel) ? 'selected' : '' }}>{{ $value->nivel_descripcion }}</option>
30                    @endforeach
31                </select>
32            </div>
33        </div>
34    </div>
35 </form>

```

**Ilustración 50: Vista Blade editar Subcategoría**  
Fuente Elaboración Propia

```

157 public static function subcategoriaUpdate(){
158     try {
159         $subcategoria_existe = SubCategoria::where('sc_name',strtoupper(request()->sc_name))->where('id_subcategoria','!=',request()->id_subcategoria)->first();
160         if(empty($subcategoria_existe)){
161             $subcategoria = SubCategoria::find(request()->id_subcategoria);
162             $subcategoria->id_categoria = request()->id_categoria;
163             $subcategoria->id_nivel = request()->id_nivel;
164             $subcategoria->sc_modificate_user = request()->sc_modificate_user;
165             $subcategoria->sc_modificate_fecha = now();
166             $subcategoria->sc_name = strtoupper(request()->sc_name);
167             $subcategoria->save();
168             return array("error"=>0,"mensaje"=>"Se actualizó exitosamente la Sub Categoría ". $subcategoria->sc_name,"tipo"=>"success","title"=>"¡Bien!");
169         }else{
170             return array("error"=>1,"mensaje"=>"La Sub Categoría ".strtoupper(request()->sc_name)." que intenta registrar ya existe","tipo"=>"warning","title"=>"¡Oops!");
171         }
172     } catch (\Exception $e) {
173         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
174     }
175 }

```

**Ilustración 51: Modelo actualización Subcategoría**  
Fuente Elaboración Propia

```

182 public static function changeStateSubCategoria(){
183     try {
184         $subcategoria = SubCategoria::find(request()->id_subcategoria);
185         if($subcategoria->sc_estado == 1){
186             $subcategoria->sc_estado = 0;
187             $mensaje = "Se inactivo exitosamente la Sub Categoría ". $subcategoria->sc_name;
188         }else{
189             $subcategoria->sc_estado = 1;
190             $mensaje = "Se activo exitosamente la Sub Categoría ". $subcategoria->sc_name;
191         }
192         $subcategoria->sc_modificate_user = request()->id_user;
193         $subcategoria->sc_modificate_fecha = now();
194         $subcategoria->save();
195         return array("error"=>0,"mensaje"=>$mensaje,"tipo"=>"success","title"=>"¡Bien!");
196     } catch (\Exception $e) {
197         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
198     }
199 }

```

**Ilustración 52: Modelo de Inactivar/Activar Subcategoría**  
Fuente Elaboración Propia

## g. Mantenimiento de Nivel

En la vista de Mantenimiento de Nivel, el administrador/soporte, podrá visualizar el nivel existente. El administrador /Soporte podrán agregar, editar, y colocar en inactivo el nivel según se necesiten para las atenciones.

```
1 | /php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use App\Models\Nivel;
7
8 class NivelController extends Controller{
9
10 public function __construct(){
11     $this->middleware('auth');
12 }
13
14 public function listNivel(){
15     $nivel = Nivel::getListNivel();
16     return view('setting-nivel.listNivel',compact('nivel'));
17 }
18
19 public function nivelNew(){
20     return view('setting-nivel.newNivel');
21 }
22
23 public function viewNivel(){
24     $nivel = Nivel::getNivel();
25     return view('setting-nivel.viewNivel',compact('nivel'));
26 }
27
28 public function editNivel(){
29     $nivel = Nivel::getNivel();
30     return view('setting-nivel.editOldNivel',compact('nivel'));
31 }
32
33 }
```

Ilustración 53: Controlador de Vistas de Mantenimiento de Nivel  
Fuente Elaboración Propia

```
1 <table class="table table-striped projects display nowrap" id="tblNivel" width="100%">
2 <thead>
3 <tr>
4 <th style="width: 15%;></th>
5 <th style="width: 45%;>Nombre Nivel</th>
6 <th style="width: 15%;>Estado</th>
7 <th style="width: 20%;>Auditoria</th>
8 </tr>
9 </thead>
10 <tbody>
11 @foreach ($nivel as $key => $value)
12 <tr>
13 <td style="width: 15%;></td>
14 <td style="width: 45%;>{{ $value->nombre_nivel }}</td>
15 <td style="width: 15%;>{{ $value->nombre_estado }}</td>
16 <td style="width: 20%;>{{ $value->auditoria }}</td>
17 </tr>
18 </tbody>
19 </table>
```

Ilustración 54: Vista en Blade (HTML) de Listado de Nivel  
Fuente Elaboración Propia

```
1 public static function getListNivel(){
2     $nivel = Nivel::select('sgl_nivel','u.usu_email as correo_creador','u.usu_nombre as nombre_creador','u.usu_apellido as ap_creador','u.usu_apellido as am_creador','u.usu_email as correo_actualizador','u.usu_nombre as nombre_actualizador','u.usu_apellido as ap_actualizador','u.usu_apellido as am_actualizador')
3     ->join('usuario_aplicacion as u','u.id_usuario','sgl_nivel.nivel_user_create')
4     ->leftJoin('usuario_aplicacion as ua','ua.id_usuario','sgl_nivel.nivel_user_modifycate')
5     ->get();
6
7     foreach ($nivel as $key => $value) {
8         if($value->nivel_estado == 1){
9             $value->nombre_estado = 'ACTIVO';
10            $value->nombre_estado = 'success';
11        }else{
12            $value->nombre_estado = 'INACTIVO';
13            $value->nombre_estado = 'danger';
14        }
15    }
16
17    if(empty($value->nivel_user_modifycate)){
18        $value->nombre_auditoria = $value->nombre_creador.' '.$value->ap_creador.' '.$value->am_creador;
19        $value->fecha_auditoria = date("d-m-Y",strtotime($value->nivel_fecha_create));
20        $value->hora_auditoria = date("H:i:s",strtotime($value->nivel_fecha_create));
21    }else{
22        $value->nombre_auditoria = $value->nombre_actualizador.' '.$value->ap_actualizador.' '.$value->am_actualizador;
23        $value->fecha_auditoria = date("d-m-Y",strtotime($value->nivel_fecha_modifycate));
24        $value->hora_auditoria = date("H:i:s",strtotime($value->nivel_fecha_modifycate));
25    }
26 }
27
28 return $nivel;
29 }
```

Ilustración 55: Modelo de Listado de Nivel  
Fuente Elaboración Propia

```

1 <form id="addNivel">
2 <meta type="hidden" name="csrf-token" content="{{ csrf_token() }}">
3 <div class="row">
4 <div class="col-xs-12 col-sm-12">
5 <div class="form-group">
6 <label>Nivel: <span class="text-danger">*</span></label>
7 <input type="text" onkeypress="return /[a-z-ñ ]/i.test(event.key)" class="form-control input-sm req" id="nivel_descripcion" name="nivel_descripcion">
8 <input type="hidden" maxLength="150" class="form-control req" id="nivel_user_create" name="nivel_user_create" value="{{ Auth::user()->id_usuario }}">
9 </div>
10 </div>
11 </div>
12 </form>
13

```

*Ilustración 56: Vista de nueva New Nivel  
Fuente Elaboración Propia*

```

42 public static function NivelSave(){
43     try {
44         $nivel_existe = Nivel::where('nivel_descripcion',strtoupper(request()->nivel_descripcion))->first();
45         if(empty($nivel_existe)){
46             $nivel = new Nivel();
47             $nivel->nivel_user_create = request()->nivel_user_create;
48             $nivel->nivel_fecha_create = now();
49             $nivel->nivel_descripcion = strtoupper(request()->nivel_descripcion);
50             $nivel->nivel_estado = 1;
51             $nivel->save();
52             return array("error"=>0,"mensaje"=>"Se agregó exitosamente el Nivel ".$nivel->nivel_descripcion,"tipo"=>"success","title"=>"¡Bien!");
53         }else{
54             return array("error"=>1,"mensaje"=>"El Nivel ".strtoupper(request()->nivel_descripcion)." que intenta registrar ya existe","tipo"=>"warning","title"=>"¡Oops!");
55         }
56     } catch (\Exception $e) {
57         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
58     }
59 }

```

*Ilustración 57: Modelo de registro Nivel  
Fuente Elaboración Propia*

```

98 public static function getNivel(){
99     $nivel = Nivel::where('id_nivel',request()->id_nivel)->first();
100     return $nivel;
101 }

```

*Ilustración 58: Modelo de get Nivel  
Fuente Elaboración Propia*

```

1 <form id="editNivel">
2 <meta type="hidden" name="csrf-token" content="{{ csrf_token() }}">
3 <div class="row">
4 <div class="col-xs-12 col-sm-12">
5 <div class="form-group">
6 <label>Nivel: <span class="text-danger">*</span></label>
7 <input type="text" onkeypress="return /[a-z-ñ ]/i.test(event.key)" class="form-control input-sm req" id="nivel_descripcion" value="{{ $nivel->nivel_descripcion }}" name="nivel_descripcion" />
8 <input type="hidden" maxLength="150" class="form-control req" id="nivel_user_modifycate" name="nivel_user_modifycate" value="{{ Auth::user()->id_usuario }}">
9 <input type="hidden" maxLength="150" class="form-control req" id="id_nivel" name="id_nivel" value="{{ $nivel->id_nivel }}">
10 </div>
11 </div>
12 </div>
13 </form>

```

*Ilustración 59: Vista Blade editar Nivel  
Fuente Elaboración Propia*



```

61 > public static function NivelUpdate(){
62 >     try {
63 >         $nivel_existe = Nivel::where('nivel_descripcion',strtoupper(request()->nivel_descripcion))->where('id_nivel','=',request()->id_nivel)->first();
64 >         if(empty($nivel_existe)){
65 >             $nivel = Nivel::find(request()->id_nivel);
66 >             $nivel->nivel_user_modificate = request()->nivel_user_modificate;
67 >             $nivel->nivel_fecha_modificate = now();
68 >             $nivel->nivel_descripcion = strtoupper(request()->nivel_descripcion);
69 >             $nivel->save();
70 >             return array("error"=>0,"mensaje"=>"Se modificó exitosamente el Nivel ".$nivel->nivel_descripcion,"tipo"=>"success","title"=>"¡Bien!");
71 >         }else{
72 >             return array("error"=>1,"mensaje"=>"El Nivel ".$nivel->nivel_descripcion." que intenta registrar ya existe","tipo"=>"warning","title"=>"¡Oops!");
73 >         }
74 >     } catch (\Exception $e) {
75 >         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
76 >     }
77 > }

```

*Ilustración 60: Modelo actualización Nivel  
Fuente Elaboración Propia*

```

78 > public static function changeStateNivel(){
79 >     try {
80 >         $nivel = Nivel::find(request()->id_nivel);
81 >         if($nivel->nivel_estado == 1){
82 >             $nivel->nivel_estado = 0;
83 >             $mensaje = 'Se inactivo exitosamente el Nivel '.$nivel->nivel_descripcion;
84 >         }else{
85 >             $nivel->nivel_estado = 1;
86 >             $mensaje = 'Se activo exitosamente el Nivel '.$nivel->nivel_descripcion;
87 >         }
88 >         $nivel->nivel_user_modificate = request()->id_user;
89 >         $nivel->nivel_fecha_modificate = now();
90 >         $nivel->save();
91 >         return array("error"=>0,"mensaje"=>$mensaje,"tipo"=>"success","title"=>"¡Bien!");
92 >     } catch (\Exception $e) {
93 >         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
94 >     }
95 > }
96 > }
97 > }

```

*Ilustración 61: Modelo de Inactivar/Activar Nivel  
Fuente Elaboración Propia*

## h. Mantenimiento Prioridad

En la vista de Mantenimiento de Área, el administrador/soporte, podrá visualizar la prioridad que se selecciona al registrar la incidencia. El administrador /Soporte podrán agregar, editar, y colocar en inactivo prioridad según se necesiten para las atenciones.

```

1 <?php
2 namespace App\Http\Controllers;
3 use Illuminate\Http\Request;
4 use App\Models\Prioridad;
5
6 class PrioridadController extends Controller{
7
8     public function __construct(){
9         $this->middleware("auth");
10    }
11
12    public function listPrioridad(){
13        $prioridad = Prioridad::getListPrioridad();
14        return view('setting.prioridad.listPrioridad',compact('prioridad'));
15    }
16
17    public function prioridadNew(){
18        return view('setting.prioridad.newPrioridad');
19    }
20
21    public function viewPrioridad(){
22        $prioridad = Prioridad::getPrioridadX();
23        return view('setting.prioridad.viewPrioridad',compact('prioridad'));
24    }
25
26    public function editPrioridad(){
27        $prioridad = Prioridad::getPrioridadX();
28        return view('setting.prioridad.editOldPrioridad',compact('prioridad'));
29    }
30 }

```

*Ilustración 62: Controlador de Vistas de Mantenimiento de Prioridad  
Fuente Elaboración Propia*

```

45 <table class="table table-striped projects display nowrap" id="tblPrioridad" width="100%">
46 <thead>
47 <tr>
48 <th style="width: 15%"></th>
49 <th style="width: 40%">Nombre Prioridad</th>
50 <th style="width: 10%" class="text-center">Estado</th>
51 <th style="width: 20%" class="text-right">Auditoria</th>
52 <th style="width: 20%"></th>
53 </tr>
54 </thead>
55 <tbody>
56 @foreach ($prioridad as $key => $value)
57 <tr>
58 <td></td>
59 <td><div>{{ $value->prioridad_name }}</div></td>
60 <td class="text-center"><span class="badge badge-{{ $value->color_estado }}">{{ $value->nombre_estado }}</span></td>
61 &td class="text-right">
62 <small>
63 <div>{{ $value->fecha_auditoria }}</div>
64 <div>{{ $value->hora_auditoria }}</div>
65 <div>
66 <div>{{ $value->nombre_auditoria }}
67 </div>
68 </div>
69 <td class="project-actions text-right">
70 <div class="btn btn-primary btn-sm text-white btn-visualiza" title="Visualizar" data-id="{{ $value->id_prioridad }}">
71 <div class="fas fa-eye"></div>
72 </div>
73 <div class="btn btn-info btn-sm text-white btn-edit" title="Editar" data-id="{{ $value->id_prioridad }}">
74 <div class="fas fa-pencil-alt"></div>
75 </div>
76 <div class="btn btn-{{ ($value->prioridad_estado == 0)?'success':'danger' }}" btn-sm text-white btn-change" title="{{ ($value->prioridad_estado == 0)?'Activar':'Inactivar' }}" data-id="{{ $value->id_prioridad }}" data-user="{{ Auth::user()->id_usuario }}">
77 <div class="fas fa-exchange-alt"></div>
78 </div>
79 </div>
80 </tr>
81 @endforeach
82 </tbody>
83 </table>

```

**Ilustración 63: Vista en Blade (HTML) de Listado de Prioridad**  
Fuente Elaboración Propia

```

21 public static function getListPrioridad(){
22 $prioridad = Prioridad::select('prioridad.', 'u.usu_email as correo_creador', 'u.usu_nombre as nombre_creador', 'u.usu_apellido as ap_creador', 'u.usu_email as
23 correo_actualizador', 'ua.usu_nombre as nombre_actualizador', 'ua.usu_apellido as ap_actualizador', 'ua.usu_email as am_actualizador')
24 ->join('usuario_aplicacion as u', 'u.id_usuario', 'prioridad.prioridad_user_create')
25 ->leftJoin('usuario_aplicacion as ua', 'ua.id_usuario', 'prioridad.prioridad_user_modificate')
26 ->get();
27
28 foreach ($prioridad as $key => $value) {
29 if($value->prioridad_estado == 1){
30 $value->nombre_estado = 'ACTIVO';
31 $value->color_estado = 'success';
32 }else{
33 $value->nombre_estado = 'INACTIVO';
34 $value->color_estado = 'danger';
35 }
36
37 if(empty($value->prioridad_user_modificate)){
38 $value->nombre_auditoria = $value->nombre_creador.' '.$value->ap_creador.' '.$value->am_creador;
39 $value->fecha_auditoria = date("d-e-Y", strtotime($value->prioridad_fecha_create));
40 $value->hora_auditoria = date("H:i:s", strtotime($value->prioridad_fecha_create));
41 }else{
42 $value->nombre_auditoria = $value->nombre_actualizador.' '.$value->ap_actualizador.' '.$value->am_actualizador;
43 $value->fecha_auditoria = date("d-e-Y", strtotime($value->prioridad_fecha_modificate));
44 $value->hora_auditoria = date("H:i:s", strtotime($value->prioridad_fecha_modificate));
45 }
46 }
47 return $prioridad;
48 }

```

**Ilustración 64: Modelo de Listado de Prioridad**  
Fuente Elaboración Propia

```

1 <form id="addPrioridad">
2 <meta type="hidden" name="csrf-token" content="{{ csrf_token() }}">
3 <div class="row">
4 <div class="col-xs-12 col-sm-12">
5 <div class="form-group">
6 <label>Prioridad: <span class="text-danger">*</span></label>
7 <input type="text" onkeypress="return /[a-z-ñ ]/i.test(event.key)" class="form-control input-sm req" id="prioridad_name" name="prioridad_name">
8 <input type="hidden" maxlength="150" class="form-control req" id="prioridad_user_create" name="prioridad_user_create" value="{{ Auth::user()->id_usuario }}">
9 </div>
10 </div>
11 </div>
12 </form>

```

**Ilustración 65: Vista de nueva New Prioridad**  
Fuente Elaboración Propia

```

49 public static function PrioridadSave(){
50     try {
51         $prioridad_existe = Prioridad::where('prioridad_name',strtoupper(request()->prioridad_name))->first();
52         if(empty($prioridad_existe)){
53             $prioridad = new Nivel();
54             $prioridad->prioridad_user_create = request()->prioridad_user_create;
55             $prioridad->prioridad_fecha_create = now();
56             $prioridad->prioridad_name = strtoupper(request()->prioridad_name);
57             $prioridad->prioridad_estado = 1;
58             $prioridad->save();
59             return array("error"=>0,"mensaje"=>"Se agregó exitosamente la Prioridad ".$prioridad->prioridad_name,"tipo"=>"success","title"=>"¡Bien!");
60         }else{
61             return array("error"=>1,"mensaje"=>"La Prioridad ".$prioridad->prioridad_name." que intenta registrar ya existe","tipo"=>"warning","title"=>"¡Oops!");
62         }
63     } catch (\Exception $e) {
64         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
65     }
66 }
67

```

*Ilustración 66: Modelo de registro Prioridad  
Fuente Elaboración Propia*

```

185 public static function getPrioridadX(){
186     $prioridad = Prioridad::where('id_prioridad',request()->id_prioridad)->first();
187     return $prioridad;
188 }

```

*Ilustración 67: Modelo de getPrioridad  
Fuente Elaboración Propia*

```

1 <form id="editPrioridad">
2 <meta type="hidden" name="csrf-token" content="{{ csrf_token() }}">
3 <div class="row">
4 <div class="col-xs-12 col-sm-12">
5 <div class="form-group">
6 <label>Prioridad: <span class="text-danger">*</span></label>
7 <input type="text" onkeypress="return /[a-z-ñ ]/.test(event.key)" class="form-control input-sm req" id="prioridad_name" value="{{ $prioridad->prioridad_name }}" name="prioridad_name" />
8 <input type="hidden" maxlength="150" class="form-control req" id="prioridad_user_modificate" name="prioridad_user_modificate" value="{{ Auth::user()->id_usuario }}">
9 <input type="hidden" maxlength="150" class="form-control req" id="id_prioridad" name="id_prioridad" value="{{ $prioridad->id_prioridad }}">
10 </div>
11 </div>
12 </div>
13 </form>

```

*Ilustración 68: Vista Blade editar Prioridad  
Fuente Elaboración Propia*

```

67
68 public static function PrioridadUpdate(){
69     try {
70         $prioridad_existe = Prioridad::where('prioridad_name',strtoupper(request()->prioridad_name))->where('id_prioridad','!=',request()->id_prioridad)->first();
71         if(empty($prioridad_existe)){
72             $prioridad = Prioridad::find(request()->id_prioridad);
73             $prioridad->prioridad_user_modificate = request()->prioridad_user_modificate;
74             $prioridad->prioridad_fecha_modificate = now();
75             $prioridad->prioridad_name = strtoupper(request()->prioridad_name);
76             $prioridad->save();
77             return array("error"=>0,"mensaje"=>"Se modificó exitosamente la Prioridad ".$prioridad->prioridad_name,"tipo"=>"success","title"=>"¡Bien!");
78         }else{
79             return array("error"=>1,"mensaje"=>"La Prioridad ".$prioridad->prioridad_name." que intenta registrar ya existe","tipo"=>"warning","title"=>"¡Oops!");
80         }
81     } catch (\Exception $e) {
82         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
83     }
84 }
85

```

*Ilustración 69: Modelo actualización Prioridad  
Fuente Elaboración Propia*

```

85
86     public static function changeStatePrioridad(){
87         try {
88             $prioridad = Prioridad::find(request()->id_prioridad);
89             if($prioridad->prioridad_estado == 1){
90                 $prioridad->prioridad_estado = 0;
91                 $mensaje = 'Se inactivo exitosamente la Prioridad '.$prioridad->prioridad_name;
92             }else{
93                 $prioridad->prioridad_estado = 1;
94                 $mensaje = 'Se activo exitosamente la Prioridad '.$prioridad->prioridad_name;
95             }
96             $prioridad->prioridad_user_modificate = request()->id_user;
97             $prioridad->prioridad_fecha_modificate = now();
98             $prioridad->save();
99             return array("error"=>0,"mensaje"=>$mensaje,"tipo"=>"success","title"=>"¡Bien!");
100         } catch (\Exception $e) {
101             return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
102         }
103     }
104 }

```

*Ilustración 70: Modelo de Inactivar/Activar Prioridad  
Fuente Elaboración Propia*

### **i. Mantenimiento Perfil**

En la vista de Mantenimiento de Perfil, el administrador/soporte, podrá visualizar el perfil existente. El administrador /Soporte podrán agregar, editar, y colocar en inactivo el perfil según se necesiten para las atenciones.

```

1 namespace App\Http\Controllers;
2
3 use Illuminate\Http\Request;
4 use App\Models\Rol;
5 use App\Models\Icono;
6
7 class RolController extends Controller{
8
9     public function __construct(){
10         $this->middleware('auth');
11     }
12
13     public function listRol(){
14         $list_rol = Rol::getListRol();
15         return view('setting.role.listRole',compact('list_rol'));
16     }
17
18     public function addRole(){
19         $list_icons = Icono::getListIcono();
20         return view('setting.role.newRole',compact('list_icons'));
21     }
22
23     public function viewRole(){
24         $rol = Rol::visualizeRol();
25         return view('setting.role.viewRole',compact('rol'));
26     }
27
28     public function editRole(){
29         $rol = Rol::viewRole();
30         return view('setting.role.editRole',compact('rol'));
31     }
32 }

```

*Ilustración 71: Controlador de Vistas de Mantenimiento de Perfil  
Fuente Elaboración Propia*

```

34 <table class="table table-striped projects display nowrap" id="tblRole" width="100%">
35     <thead>
36         <tr>
37             <th style="width: 10%;>ID</th>
38             <th style="width: 20%;>Nombre Perfil</th>
39             <th style="width: 30%;>Foto Perfil por Defector</th>
40             <th style="width: 10%;>class="text-center">Estado</th>
41             <th style="width: 30%;>Auditoria</th>
42         </tr>
43     </thead>
44     <tbody>
45         <tr>
46             <td colspan="5">
47                 <div class="table">
48                     <table border="1">
49                         <thead>
50                             <tr>
51                                 <th style="width: 10%;>ID</th>
52                                 <th style="width: 20%;>Nombre Perfil</th>
53                                 <th style="width: 30%;>Foto Perfil por Defector</th>
54                                 <th style="width: 10%;>class="text-center">Estado</th>
55                                 <th style="width: 30%;>Auditoria</th>
56                             </tr>
57                         </thead>
58                         <tbody>
59                             <tr>
60                                 <td>{{ $rol->id_rol }}</td>
61                                 <td>{{ $rol->nombre_rol }}</td>
62                                 <td>{{ $rol->foto_rol }}</td>
63                                 <td>{{ $rol->estado_rol }}</td>
64                                 <td>{{ $rol->auditoria_rol }}</td>
65                             </tr>
66                         </tbody>
67                     </table>
68                 </div>
69             </td>
70         </tr>
71     </tbody>
72 </table>
73
74 <div class="project-state">
75     <span class="badge badge-{{ $rol->color_estado }}">{{ $rol->nombre_estado }}</span>
76 </div>
77
78 <div class="project-progress text-right">
79     <div class="progress">
80         <div style="width: {{ $rol->progreso }}%;></div>
81     </div>
82     <div class="text-right">
83         <span class="font-weight-bold">{{ $rol->progreso }}%</span>
84     </div>
85 </div>
86
87 <div class="project-actions text-right">
88     <button type="button" class="btn btn-primary btn-sm text-white btn-visualize" title="Visualizar" data-id="{{ $rol->id_rol }}"></button>
89     <button type="button" class="btn btn-info btn-sm text-white btn-edit" title="Editar" data-id="{{ $rol->id_rol }}"></button>
90     <button type="button" class="btn btn-danger btn-sm text-white btn-change" title="{{ $rol->estado == 0?'Activar':'Inactivar' }}" data-id="{{ $rol->id_rol }}"></button>
91 </div>
92
93 </tbody>
94 </table>

```

*Ilustración 72: Vista en Blade (HTML) de Listado de Perfil  
Fuente Elaboración Propia*

```

45 public static function getListRol(){
46     $list_rol = Rol::select('rol_aplicacion.*','u.usu_email as correo_creador','u.usu_nombre as nombre_creador','u.usu_apellido as ap_creador','u.usu_apellido as am_creador','ua.usu_email as
47     correo_actualizador','ua.usu_nombre as nombre_actualizador','ua.usu_apellido as ap_actualizador','ua.usu_apellido as am_actualizador')
48     ->join('usuario_aplicacion as u','u.id_usuario','rol_aplicacion.rol_user_create')
49     ->leftJoin('usuario_aplicacion as ua','ua.id_usuario','rol_aplicacion.rol_user_update')
50     ->get();
51
52     foreach($list_rol as $key => $value){
53         if($value->rol_estado == 1){
54             $value->nombre_estado = 'ACTIVO';
55             $value->color_estado = 'success';
56         }else{
57             $value->nombre_estado = 'INACTIVO';
58             $value->color_estado = 'danger';
59         }
60
61         if(empty($value->rol_user_update)){
62             $value->nombre_auditoria = $value->nombre_creador.' '.$value->ap_creador.' '.$value->am_creador;
63             $value->fecha_auditoria = date("d-m-Y",strtotime($value->rol_fecha_create));
64             $value->hora_auditoria = date("H:i:s",strtotime($value->rol_fecha_create));
65         }else{
66             $value->nombre_auditoria = $value->nombre_actualizador.' '.$value->ap_actualizador.' '.$value->am_actualizador;
67             $value->fecha_auditoria = date("d-m-Y",strtotime($value->rol_fecha_update));
68             $value->hora_auditoria = date("H:i:s",strtotime($value->rol_fecha_update));
69         }
70     }
71
72     return $list_rol;
73 }

```

*Ilustración 73: Modelo de Listado de Perfil  
Fuente Elaboración Propia*

```

1 <form id="addRole">
2 <meta type="hidden" name="csrf-token" content="{{ csrf_token }}">
3 <div class="row">
4 <div class="col-xs-12 col-sm-12">
5 <div class="form-group">
6 <label>Nombre Perfil: <span class="text-danger">*</span></label>
7 <input type="text" onkeypress="return /[a-z-À ]/i.test(event.key)" class="form-control input-sm req" id="rol_nombre" name="rol_nombre">
8 <input type="hidden" maxlength="150" class="form-control req" id="rol_user_create" name="rol_user_create" value="{{ Auth::user()->id_usuario }}">
9 </div>
10 </div>
11 <div class="col-xs-12 col-sm-12">
12 <div class="form-group">
13 <label>Foto Perfil por Defecto: <span class="text-danger">*</span></label>
14 <div class="row">
15 @foreach ($list_rol as $key => $value)
16 <div class="col-sm-2">
17 <a href="#">{{ $value->en_uso == 0?$value->items_href.' '.$value->items_toggle.' '.$value->items_title.' '.$value->items_gallery:'' }}>
18 en_uso == 0?'':style="filter:grayscale(100%);webkit-filter:grayscale(100%);" }} class="img-fluid mb-2" alt="white sample"/>
19 </a>
20 </div>
21 @endforeach
22 </div>
23 <input type="hidden" onkeypress="return /[a-z-À ]/i.test(event.key)" class="form-control input-sm req" id="rol_icono" name="rol_icono">
24 </div>
25 </div>
26 </div>
27 </form>
28 </form>

```

*Ilustración 74: Vista de nueva New Perfil  
Fuente Elaboración Propia*

```

44 public static function roleSave(){
45     try {
46         $rol_existe = Rol::where('rol_nombre',strtoupper(request()->rol_nombre))->first();
47         if(empty($rol_existe)){
48             $rol = new Rol();
49             $rol->rol_user_create = request()->rol_user_create;
50             $rol->rol_fecha_create = now();
51             $rol->rol_nombre = strtoupper(request()->rol_nombre);
52             $rol->rol_icono = request()->rol_icono;
53             $rol->rol_estado = 1;
54             $rol->save();
55             return array("error"=>0,"mensaje"=>"Se agregó exitosamente el Perfil ".$rol->rol_nombre,"tipo"=>"success","title"=>"¡Bien!");
56         }else{
57             return array("error"=>1,"mensaje"=>"El Perfil ".$rol->rol_nombre." que intenta registrar ya existe","tipo"=>"warning","title"=>"¡Oops!");
58         }
59     } catch (\Exception $e) {
60         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","title"=>"¡Oops!");
61     }
62 }

```

*Ilustración 75: Modelo de registro Perfil  
Fuente Elaboración Propia*

```

106 public static function visualizeRol(){
107     $rol = Rol::where('id_rol',request()->id_rol)->first();
108     return $rol;
109 }
110

```

*Ilustración 76: Modelo de getPerfil  
Fuente Elaboración Propia*

```

<form id="updateRole">
  <meta type="hidden" name="csrf-token" content="{{ csrf_token() }}">
  <div class="row">
    <div class="col-xs-12 col-sm-12">
      <div class="form-group">
        <label>Nombre Perfil: <span class="text-danger">*</span></label>
        <input type="text" onkeypress="return /[a-z-0-9 ]:/i.test(event.key)" class="form-control input-sm req" value="{{ $rol->rol_nombre }}" id="rol_nombre" name="rol_nombre">
        <input type="hidden" maxlength="150" class="form-control req" id="id_rol" name="id_rol" value="{{ $rol->rol_id_rol }}">
        <input type="hidden" maxlength="150" class="form-control req" id="rol_user_create" name="rol_user_create" value="{{ Auth::user()->id_usuario }}">
      </div>
    </div>
    <div class="col-xs-12 col-sm-12">
      <div class="form-group">
        <label>foto Perfil por Defecto: <span class="text-danger">*</span></label>
        <div class="row">
          @foreach ($rol->iconos as $key => $value)
            <div class="col-sm-2">
              <pre style="display:none">
                {{ print_r($value->sen_uso) }}
              <br>
              {{ print_r($rol->rol_icono) }}
            </pre>
            <a href="#">
              sen_uso == 1 && $value->ico_url == $rol->rol_icono)?$value->item_href.' '.$value->item_suggin.' '.$value->item_title.' '.$value->item_gallery:'' }}
              sen_uso == 0 || $value->ico_url == $rol->rol_icono)?'':style="filter:grayscale(100%);-webkit-filter:grayscale(100%);" }} class="img-fluid mb-2" {{ ($value->ico_url == $rol->rol_icono)?'border border-dark':'' }} alt="white sample"/>
            </a>
          </div>
        @endforeach
      </div>
      <input type="hidden" onkeypress="return /[a-z-0-9 ]:/i.test(event.key)" value="{{ $rol->rol_icono }}" class="form-control input-sm req" id="rol_icono" name="rol_icono">
    </div>
  </div>
</form>

```

*Ilustración 77: Vista Blade editar Perfil  
Fuente Elaboración Propia*

```

64 public static function updateRole(){
65     try {
66         $rol_existe = Rol::where('rol_nombre',strtoupper(request()->rol_nombre))->where('id_rol','!=',request()->id_rol)->first();
67         if(empty($rol_existe)){
68             $rol = Rol::find(request()->id_rol);
69             $rol->rol_user_update = request()->rol_user_create;
70             $rol->rol_fecha_update = now();
71             $rol->rol_nombre = strtoupper(request()->rol_nombre);
72             $rol->rol_icono = request()->rol_icono;
73             $rol->rol_estado = 1;
74             $rol->save();
75
76             $usuarios = User::where('id_rol',request()->id_rol)
77                 ->join('usuario_rol as r','r.id_usuario','usuario_aplicacion.id_usuario')
78                 ->update(array('usu_foto_perfil' => request()->rol_icono));
79
80             return array("error"=>0,"mensaje"=>"Se modificó exitosamente el Perfil ".$rol->rol_nombre,"tipo"=>"success","title"=>"¡Bien!");
81         }else{
82             return array("error"=>1,"mensaje"=>"El nombre del Perfil ".$rol->rol_nombre." que intenta modificar ya existe","tipo"=>"warning","title"=>"¡Oops!");
83         }
84     } catch (\Exception $e) {
85         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
86     }
87 }

```

*Ilustración 78: Modelo actualización Perfil  
Fuente Elaboración Propia*

```

89 public static function changeRole(){
90     try {
91         $rol = Rol::find(request()->id_rol);
92         if($rol->rol_estado == 1){
93             $rol->rol_estado = 0;
94             $mensaje = 'Se inactivo exitosamente el Perfil '.$rol->rol_nombre;
95         }else{
96             $rol->rol_estado = 1;
97             $mensaje = 'Se activo exitosamente el Perfil '.$rol->rol_nombre;
98         }
99         $rol->save();
100         return array("error"=>0,"mensaje"=>$mensaje,"tipo"=>"success","title"=>"¡Bien!");
101     } catch (\Exception $e) {
102         return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
103     }
104 }

```

*Ilustración 79: Modelo de Inactivar/Activar Perfil  
Fuente Elaboración Propia*

## j. Mantenimiento de Usuarios

En la vista de Mantenimiento Usuario, el administrador/soporte, rellena el formulario que se completará con los datos del usuario, asignando el perfil y

cargo correspondiente. El administrador /Soporte podrán agregar, editar, y colocar en inactivo el usuario según las indicaciones de la Institución.

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use App\User;
7 use App\Models\Rol;
8 use App\Models\Cargo;
9 use App\Models\Nivel;
10
11 class UserController extends Controller{
12
13     public function __construct(){
14         $this->middleware('auth');
15     }
16
17     public function prueba(){
18         return request()->all();
19     }
20
21     public function getListUsers(){
22         $list_user = User::getList();
23         return view('setting.user-listUser',compact('list_user'));
24     }
25
26     public function newUser(){
27         $list_rol = Rol::where('rol_estado',1)->get();
28         $list_cargo = Cargo::where('cargo_estado',1)->get();
29         $nivel = Nivel::where('nivel_estado',1)
30             ->orderBy('nivel_descripcion','asc')
31             ->get();
32         return view('setting.user.newUser',compact('list_rol','list_cargo','nivel'));
33     }
34
35     public function viewUser(){
36         $user = User::getUserView();
37         return view('setting.user.viewUser',compact('user'));
38     }
39
40     public function editUser(){
41         $user = User::getUserView();
42         return view('setting.user.editUser',compact('user'));
43     }
44 }
45 }

```

*Ilustración 80: Controlador de Vistas de Mantenimiento de Usuarios  
Fuente Elaboración Propia*

```

<table class="table table-striped projects display nowrap" id="tbl01" width="100%">
| # | Nombre | Usuario | Cargo | Nivel | Foto Perfil | Estado | Auditoria |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Nombre  Usuario  Cargo  Nivel  Foto Perfil  Estado  Auditoria | | | | | | | |
| Nombre  Usuario  Cargo  Nivel  Foto Perfil  Estado  Auditoria |

```

*Ilustración 81: Vista en Blade (HTML) de Listado de Usuarios  
Fuente Elaboración Propia*

```

public static function getUsuarios()
{
    $user = User::select('usuario_aplicacion.*', 'u.usu_email as correo_creador', 'u.usu_nombre as nombre_creador', 'u.usu_spellidop as ap_creador', 'u.usu_spellidop as am_creador', 'us.usu_email as correo_actualizador', 'us.usu_nombre as nombre_actualizador', 'us.usu_spellidop as ap_actualizador', 'us.usu_spellidop as am_actualizador', 'ur.*', 'ra.*', 'c.cargo_nombre', 'u.nivel_descripcion')
    ->join('usuario_aplicacion as ua', 'ua.id_usuario', 'usuario_aplicacion.usu_usuario_creation')
    ->join('usuario_rol as ur', 'ur.id_usuario', 'usuario_aplicacion.usu_usuario')
    ->join('rol_aplicacion as ra', 'ra.id_rol', 'ur.id_rol')
    ->join('cargo as c', 'c.id_cargo', 'usuario_aplicacion.id_cargo')
    ->leftJoin('ul.nivel as u', 'u.id_nivel', 'usuario_aplicacion.id_nivel')
    ->leftJoin('usuario_aplicacion as ua', 'ua.id_usuario', 'usuario_aplicacion.usu_usuario_modificacion')
    ->get();

    foreach($user as $key => $value){
        if(empty($value->nivel_descripcion)){
            $value->nivel_descripcion = '-';
        }

        $value->nombre_usuario = $value->usu_nombre . ' ' . $value->usu_apellido . ' ' . $value->usu_apellido;

        if($value->usu_estado == 1){
            $value->nombre_estado = 'ACTIVO';
        } else {
            $value->nombre_estado = 'INACTIVO';
        }
        $value->color_estado = 'danger';
    }

    if(empty($value->usu_usuario_modificacion)){
        $value->nombre_auditoria = $value->nombre_creador . ' ' . $value->ap_creador . ' ' . $value->am_creador;
        $value->fecha_auditoria = date("d-m-Y", strtotime($value->usu_fecha_creation));
        $value->hora_auditoria = date("H:i:s", strtotime($value->usu_fecha_creation));
    } else {
        $value->nombre_auditoria = $value->nombre_actualizador . ' ' . $value->ap_actualizador . ' ' . $value->am_actualizador;
        $value->fecha_auditoria = date("d-m-Y", strtotime($value->usu_fecha_modificacion));
        $value->hora_auditoria = date("H:i:s", strtotime($value->usu_fecha_modificacion));
    }
}

return $user;
}
}

```

**Ilustración 82: Modelo de Listado de Usuarios**  
Fuente Elaboración Propia

```

<table border="1" class="table">
| Formulario de nuevo usuario | |
| --- | --- |
| <div class="form-group"> <input type="text" class="form-control" value="" id="usu_nombre" name="usu_nombre"> | |
| <input type="text" class="form-control" value="" id="usu_apellido" name="usu_apellido"> | |
| <input type="text" class="form-control" value="" id="usu_email" name="usu_email"> | |
| <input type="text" class="form-control" value="" id="usu_fecha" name="usu_fecha"> | |
| <input type="text" class="form-control" value="" id="usu_rol" name="usu_rol"> | |
| <input type="text" class="form-control" value="" id="usu_cargo" name="usu_cargo"> | |
| <input type="text" class="form-control" value="" id="usu_nivel" name="usu_nivel"> | |

```

**Ilustración 83: Vista de nueva New Usuarios**  
Fuente Elaboración Propia

```

public static function userSave()
{
    try {
        $user = User::create(['usu_nombre' => $request->usu_nombre,
            'usu_apellido' => $request->usu_apellido,
            'usu_email' => $request->usu_email,
            'usu_fecha' => $request->usu_fecha,
            'usu_estado' => 1,
            'usu_usuario_creation' => now()]);

        if($request->usu_usuario_modificacion){
            $rol = Rol::where('id_rol', $request->usu_rol)->first();
            $cargo = Cargo::where('id_cargo', $request->usu_cargo)->first();
            $nivel = Nivel::where('id_nivel', $request->usu_nivel)->first();
            $user->usu_usuario_modificacion = $request->usu_usuario_modificacion;
            $user->usu_usuario_creation = $request->usu_usuario_creation;
            $user->usu_usuario_modificacion = $request->usu_usuario_modificacion;
            $user->usu_usuario_modificacion = $request->usu_usuario_modificacion;
            $user->usu_usuario_modificacion = $request->usu_usuario_modificacion;
        }

        $user->usu_usuario_modificacion = $request->usu_usuario_modificacion;
        $user->usu_usuario_modificacion = $request->usu_usuario_modificacion;
        $user->usu_usuario_modificacion = $request->usu_usuario_modificacion;
        $user->usu_usuario_modificacion = $request->usu_usuario_modificacion;

        return array('mensaje' => "Se registro exitosamente el usuario.", 'tipo' => "success", 'titulo' => "Exito!");
    } catch (Exception $e) {
        return array('mensaje' => "El usuario que intenta registrar ya existe.", 'tipo' => "warning", 'titulo' => "Ooops!");
    }
}
}

```

**Ilustración 84: Modelo de registro Usuarios**  
Fuente Elaboración Propia



```
174
175 public static function getUserView()
176 {
177     $user = new Usuario();
178     $user->usuario = User::join('usuario rol as r','r.id_usuario','usuario.aplicacion.id_usuario')
179     ->join('cargo as c','c.id_cargo','usuario.aplicacion.id_cargo')
180     ->where('usuario.aplicacion.id_usuario',request()->id_usuario)
181     ->first();
182     $user->list_rol = Rol::where('rol_estado',1)->get();
183     $user->list_cargo = Cargo::where('cargo_estado',1)->get();
184     $user->nivel = Nivel::where('nivel_estado',1)
185     ->orderBy('nivel_descripcion','asc')
186     ->get();
187     return $user;
188 }
```

Ilustración 85: Modelo de get Usuarios  
Fuente Elaboración Propia

```
140 class `editusuario`
141 {
142     `name`="carv-taken" content="({ user-taken() }})"
143     `class`="form"
144     `div class="col-sm-12 col-md-6"
145     `class="form-group"
146     `labelNombre: copy class="text-danger"/`<span/>/label)
147     `input type="text" autocomplete="off" /s-a-B /f.tact(omit_key)" class="form-control input-sm req" id="usu_nombre" name="usu_nombre" value="{{ $user->usuario->usu_nombre }}"`
148     `input type="hidden" maxlength="100" class="form-control req" id="usu_usuario_modificacion" name="usu_usuario_modificacion" value="1"
149     `input type="hidden" maxlength="100" class="form-control req" id="id_usuario" name="id_usuario" value="{{ $user->usuario->id_usuario }}"`
150     `endif`
151     `div class="col-sm-12 col-md-6"
152     `class="form-group"
153     `labelApellido: copy class="text-danger"/`<span/>/label)
154     `input type="text" autocomplete="off" /s-a-B /f.tact(omit_key)" class="form-control input-sm req" id="usu_apellido" name="usu_apellido" value="{{ $user->usuario->usu_apellido }}"`
155     `endif`
156     `div class="col-sm-12 col-md-6"
157     `class="form-group"
158     `labelApellido: copy class="text-danger"/`<span/>/label)
159     `input type="text" autocomplete="off" /s-a-B /f.tact(omit_key)" class="form-control input-sm req" id="usu_apellido" name="usu_apellido" value="{{ $user->usuario->usu_apellido }}"`
160     `endif`
161     `div class="col-sm-12 col-md-6"
162     `class="form-group"
163     `labelFecha: copy class="text-danger"/`<span/>/label)
164     `input type="date" class="form-control input-sm req" id="usu_fecha" name="usu_fecha" value="{{ $user->usuario->usu_fecha }}"`
165     `endif`
166     `div class="col-sm-12 col-md-12"
167     `class="form-group"
168     `labelE-Mail: copy class="text-danger"/`<span/>/label)
169     `input type="email" class="form-control input-sm req" id="usu_email" name="usu_email" value="{{ $user->usuario->usu_email }}"`
170     `endif`
171     `div class="col-sm-12 col-md-6"
172     `class="form-group"
173     `labelPerfil: copy class="text-danger"/`<span/>/label)
174     `select class="form-control req" id="id_rol" name="id_rol"
175     `option value="0"></option/>/option)
176     `foreach ($user->list_rol as $key => $value)
177     `option value="{{ $value->id_rol }}" {{ $value->id_rol==$user->usuario->id_rol ? "selected" : "" }} {{ $value->rol_nombre }}</option`
178     `endforeach`
179     `endif`
180     `div class="col-sm-12 col-md-6"
181     `class="form-group"
182     `labelCargo: copy class="text-danger"/`<span/>/label)
183     `select class="form-control req" id="id_cargo" name="id_cargo"
184     `option value="0"></option/>/option)
185     `foreach ($user->list_cargo as $key => $value)
186     `option value="{{ $value->id_cargo }}" {{ $value->id_cargo==$user->usuario->id_cargo ? "selected" : "" }} {{ $value->cargo_nombre }}</option`
187     `endforeach`
188     `endif`
189     `div class="col-sm-12 col-md-6" id="div-nivel"
190     `class="form-group"
191     `labelNivel: copy class="text-danger"/`<span/>/label)
192     `select class="form-control" id="id_nivel" name="id_nivel"
193     `option value="0"></option/>/option)
194     `foreach ($user->list_nivel as $key => $value)
195     `option value="{{ $value->id_nivel }}" {{ $value->id_nivel==$user->usuario->id_nivel ? "selected" : "" }} {{ $value->nivel_descripcion }}</option`
196     `endforeach`
197     `endif`
198     `endif`
199     `endif`
200     `endif`
201     `endif`
202 }
```

Ilustración 86: Vista Blade editar Usuarios  
Fuente Elaboración Propia

```
137
138 public static function updateUser()
139 {
140     $user_in_existe = User::where('usu_nombre', request()->usu_nombre)
141     ->where('usu_apellido', request()->usu_apellido)
142     ->where('usu_usuario_modificacion', request()->usu_usuario_modificacion)
143     ->where('id_usuario', request()->id_usuario)
144     ->first();
145     if(empty($user_in_existe)){
146         $rol_id = Rol::where('rol_estado',1)->first();
147         $user = User::where('id_usuario', request()->id_usuario)->first();
148         $user->usu_nombre = trim(request()->usu_nombre);
149         $user->usu_apellido = trim(request()->usu_apellido);
150         $user->usu_usuario_modificacion = trim(request()->usu_usuario_modificacion);
151         $user->usu_fecha = request()->usu_fecha;
152         $user->id_cargo = request()->id_cargo;
153         if(request()->id_rol == null){
154             $user->id_nivel = request()->id_nivel;
155         }
156         $user->id_nivel = null;
157     }
158     $user_in_existe = User::where('usu_nombre', request()->usu_nombre)
159     ->where('id_usuario', request()->id_usuario)
160     ->first();
161     if(empty($user_in_existe) || $user_in_existe->usu_nombre != 'ADMIN'){
162         $user_in_existe = User::where('usu_nombre', request()->usu_nombre)
163         ->where('id_usuario', request()->id_usuario)
164         ->first();
165     }
166     $user->usu_nombre = request()->usu_nombre;
167     $user->usu_apellido = request()->usu_apellido;
168     $user->usu_usuario_modificacion = request()->usu_usuario_modificacion;
169     $user->usu_fecha_modificacion = request()->usu_fecha_modificacion;
170     $user->usu_fecha = request()->usu_fecha;
171     $user->usu_fecha_perfil = request()->usu_fecha_perfil;
172     $user->save();
173     $nombre = request()->usu_nombre." ".$user->usu_apellido." ".$user->usu_apellido;
174     $rol = Usuario::where('id_usuario', request()->id_usuario)->first();
175     $rol->id_usuario = request()->id_usuario;
176     $rol->id_rol = request()->id_rol;
177     $rol->usu_usuario_modificacion = request()->usu_usuario_modificacion;
178     $rol->usu_fecha_modificacion = request()->usu_fecha_modificacion;
179     $rol->save();
180     return array("error"=>"0","mensaje"=>"Se actualizó exitosamente el Usuario ". $nombre, "tipo"=>"success","titulo"=>"¡Bien!");
181     }
182     }
183     return array("error"=>"1","mensaje"=>"El usuario que intenta modificar ya existe", "tipo"=>"warning","titulo"=>"¡Opa!");
184     }
185     catch (Exception $e) {
186     return array("error"=>"1","mensaje"=>"Existe un problema de conexión, inténtalo nuevamente", "tipo"=>"error","titulo"=>"¡Opa!");
187     }
188 }
```

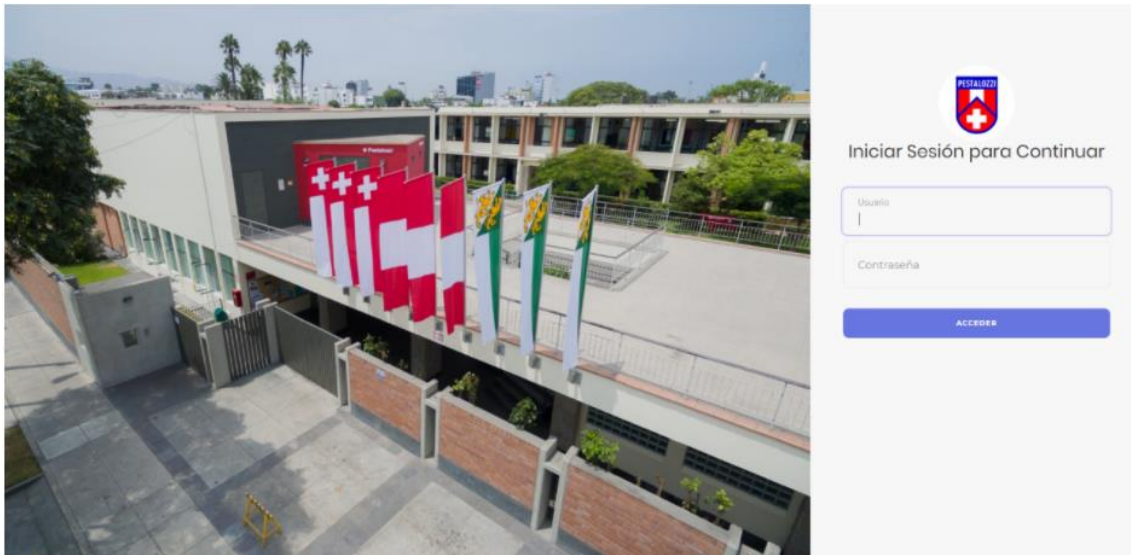
Ilustración 87: Modelo actualización Usuarios  
Fuente Elaboración Propia

```
201 public static function changeUser(){
202     try {
203         $user = User::find(request()->id_usuario);
204         $nombre = $user->usu_nombre.' '.$user->usu_apellido.' '.$user->usu_apellido;
205         if($user->usu_estado == 1){
206             $user->usu_estado = 0;
207             $mensaje = 'Se Inactivo exitosamente al Usuario '.$nombre;
208         }else{
209             $user->usu_estado = 1;
210             $mensaje = 'Se activo exitosamente al Usuario '.$nombre;
211         }
212         $user->usu_usuario_modificacion = request()->id_usuario;
213         $user->usu_fecha_modificacion = now();
214         $user->save();
215         return array("error">0,"mensaje">$mensaje,"tipo">"success","titulo">"Bien!");
216     } catch (Exception $e) {
217         return array("error">1,"mensaje">"Existe un problema de conexión, inténtalo nuevamente","tipo">"error","titulo">"¡Ups!");
218     }
219 }
```

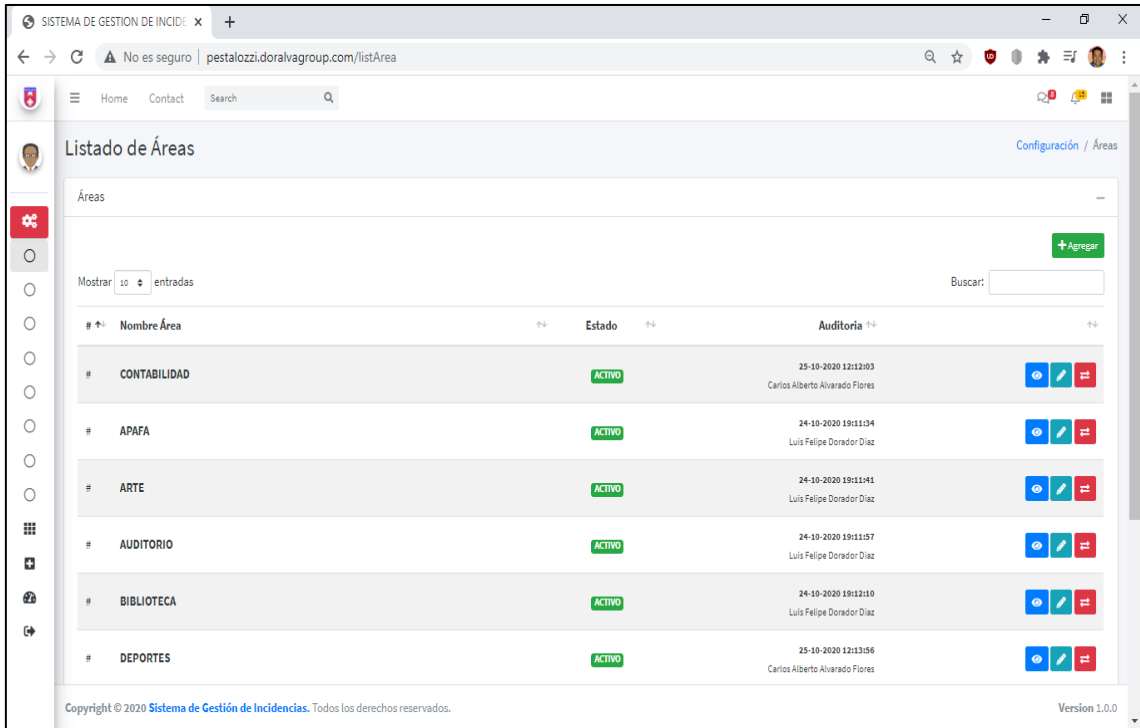
*Ilustración 88: Modelo de Inactivar/Activar Usuarios  
Fuente Elaboración Propia*

### 1.6. IMPLEMENTACIÓN DEL SPRINT 1

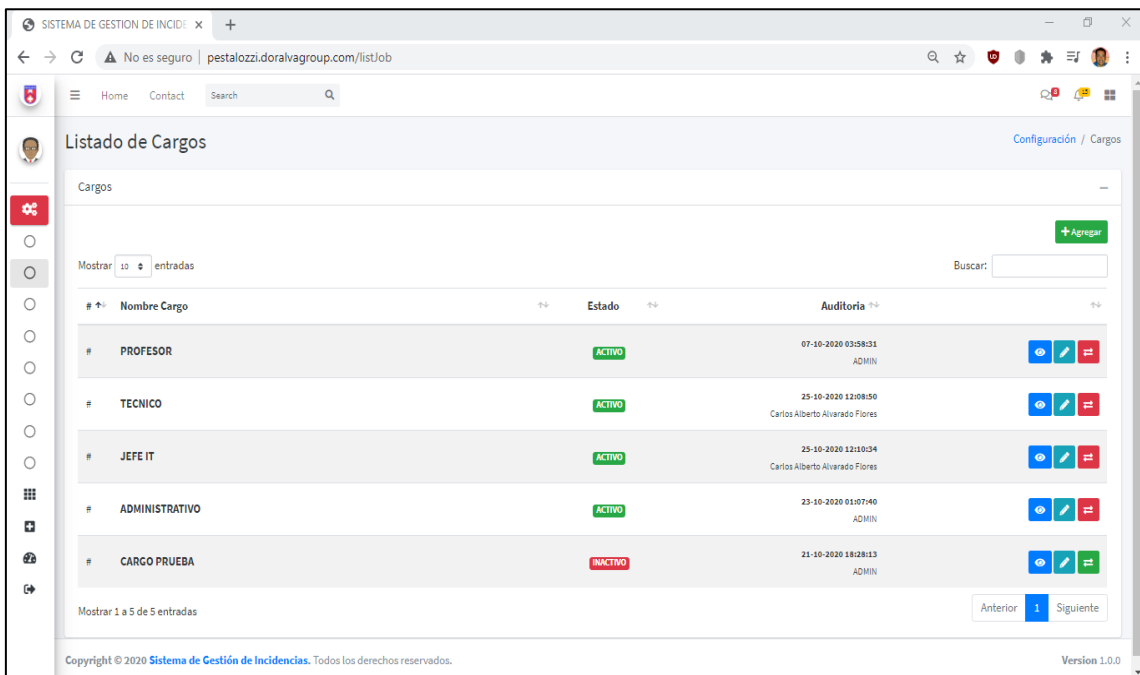
Una vez concluidas las etapas de desarrollo del sistema web (análisis, diseño y código), se pasa a la presentación final de las vistas y funcionalidades del sprint 1.



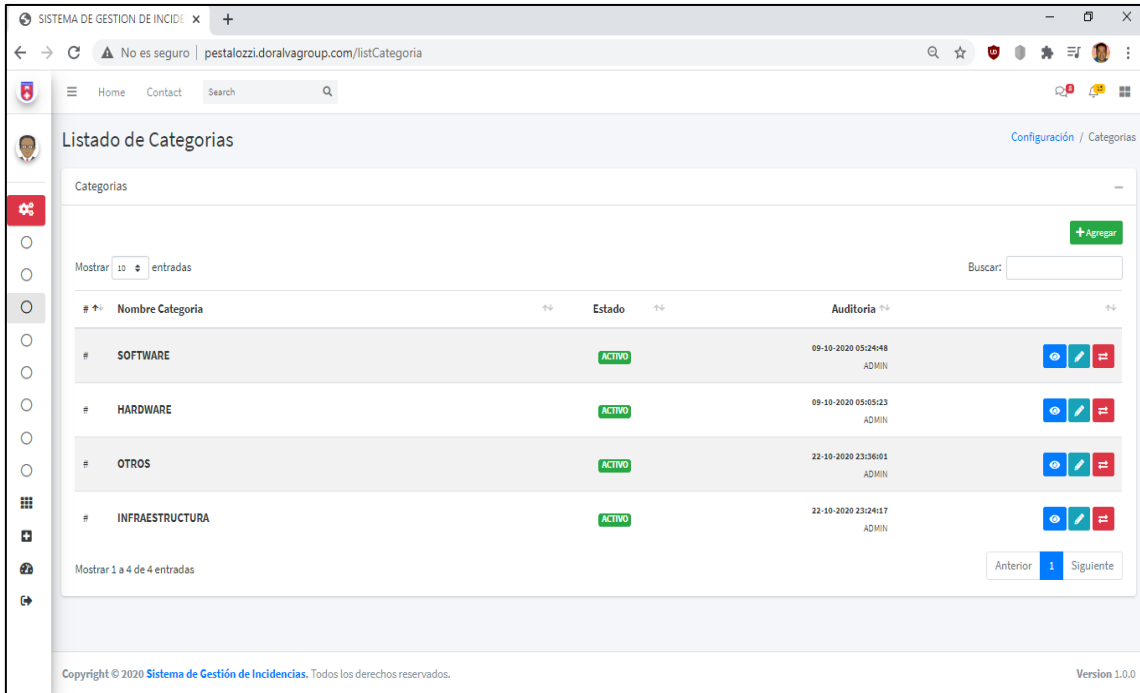
*Ilustración 89: Interfaz del Login de Usuarios  
Fuente: Elaboración Propia*



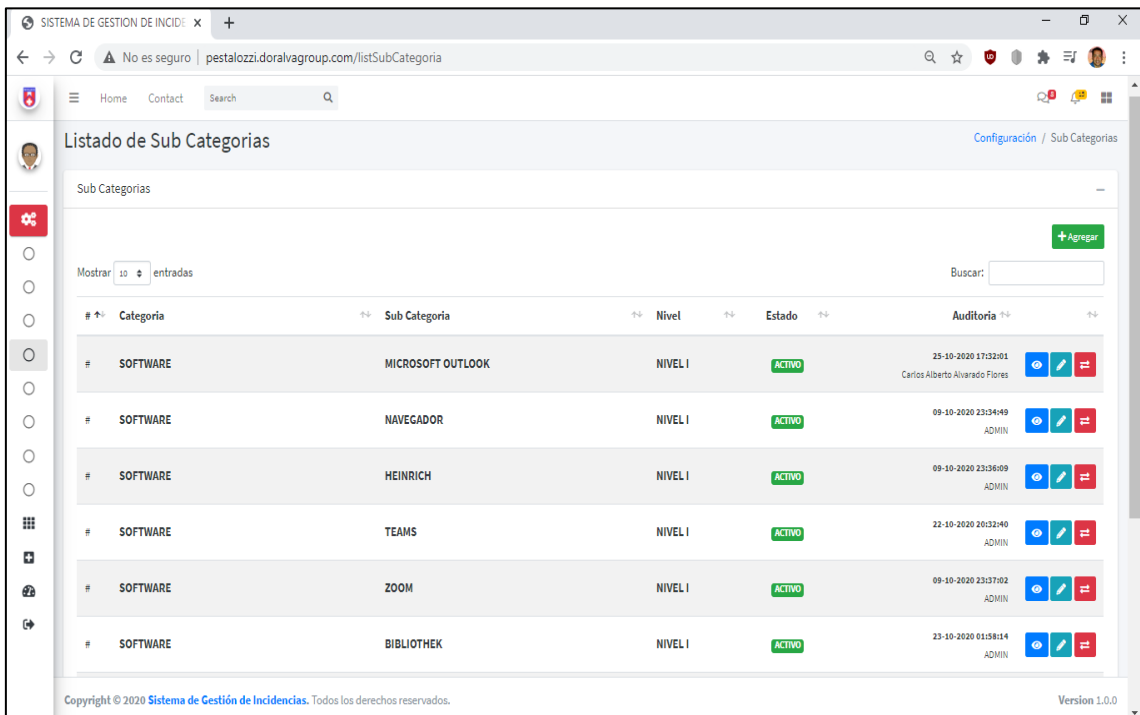
*Ilustración 90: Interfaz de Área*  
Fuente: Elaboración Propia



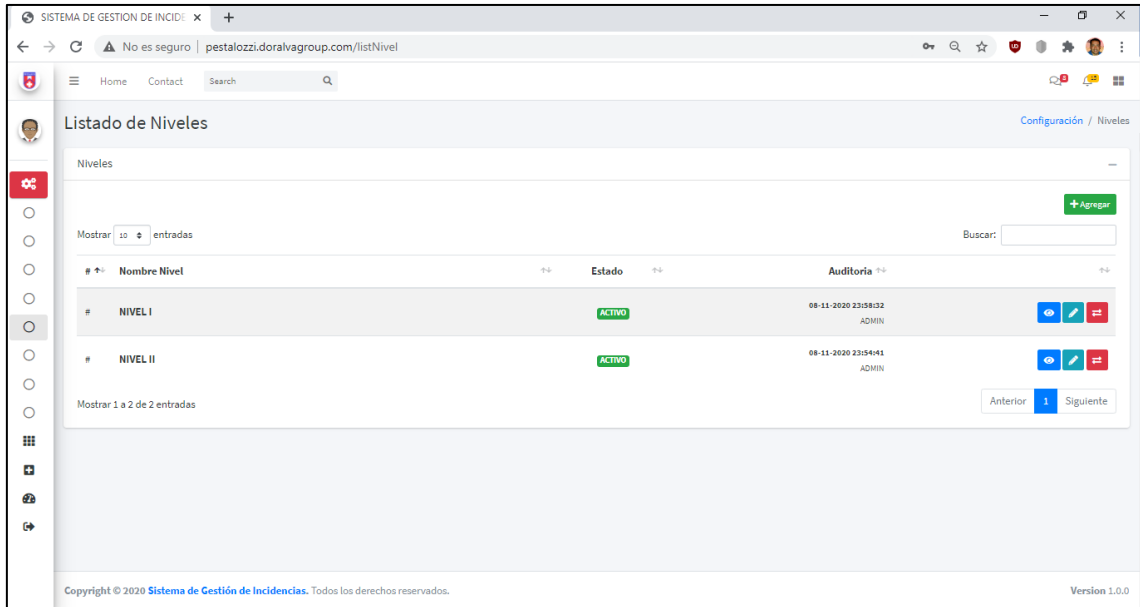
*Ilustración 91: Interfaz de Cargos*  
Fuente: Elaboración Propia



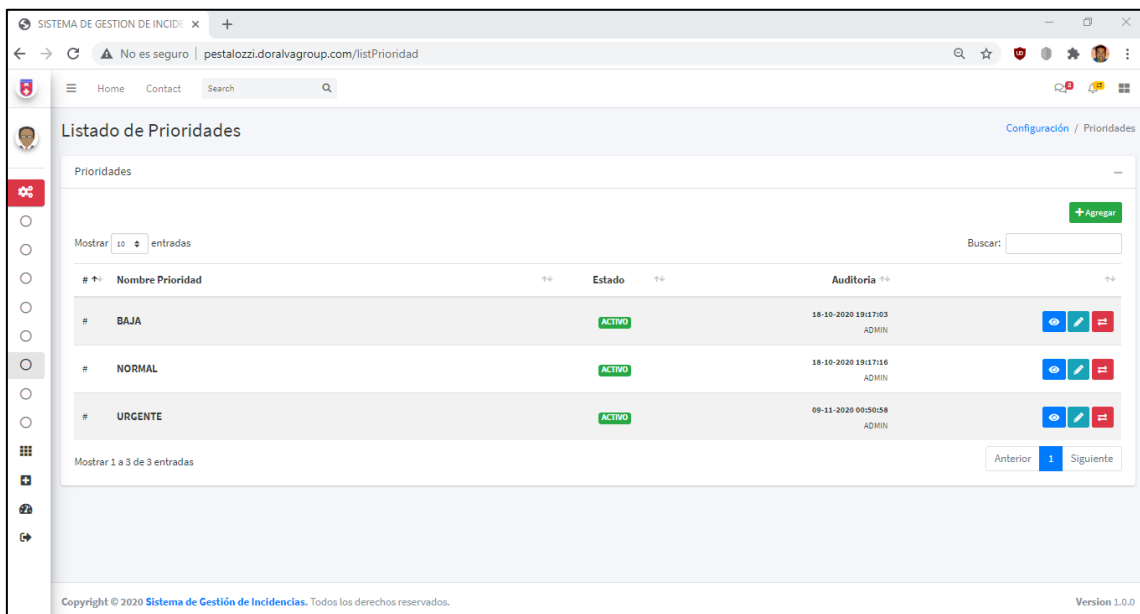
*Ilustración 92: Interfaz de Categorías  
Fuente: Elaboración Propia*



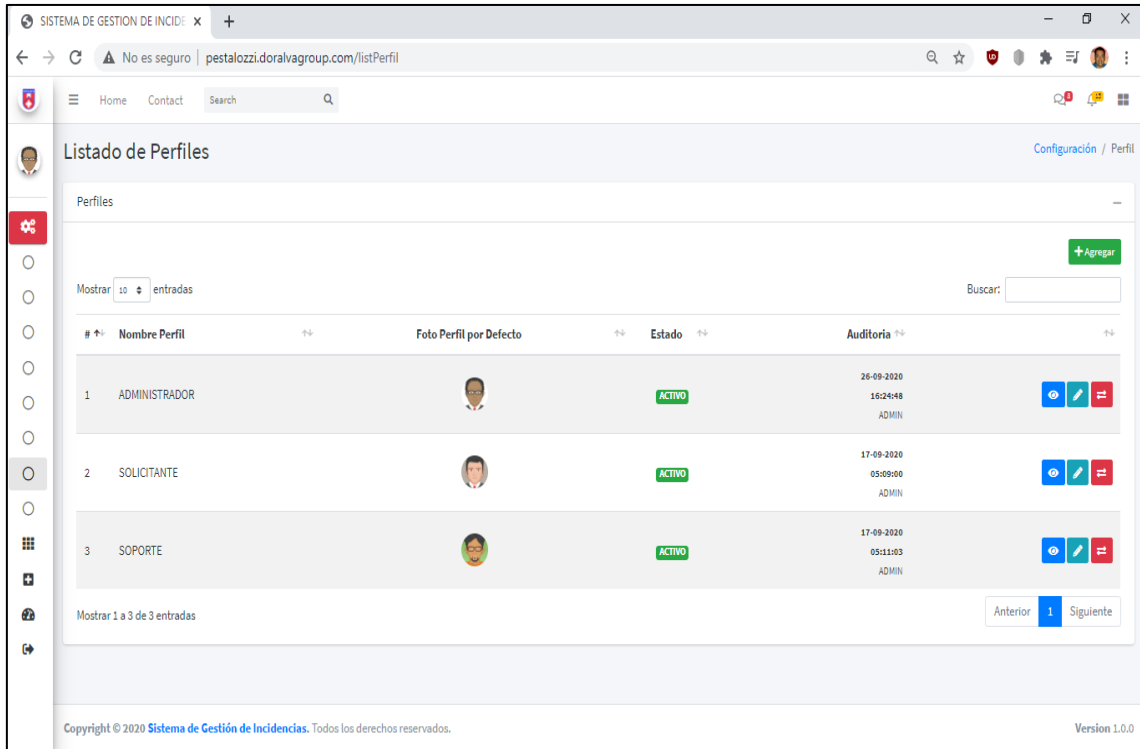
*Ilustración 93: Interfaz de Sub-Categorías  
Fuente: Elaboración Propia*



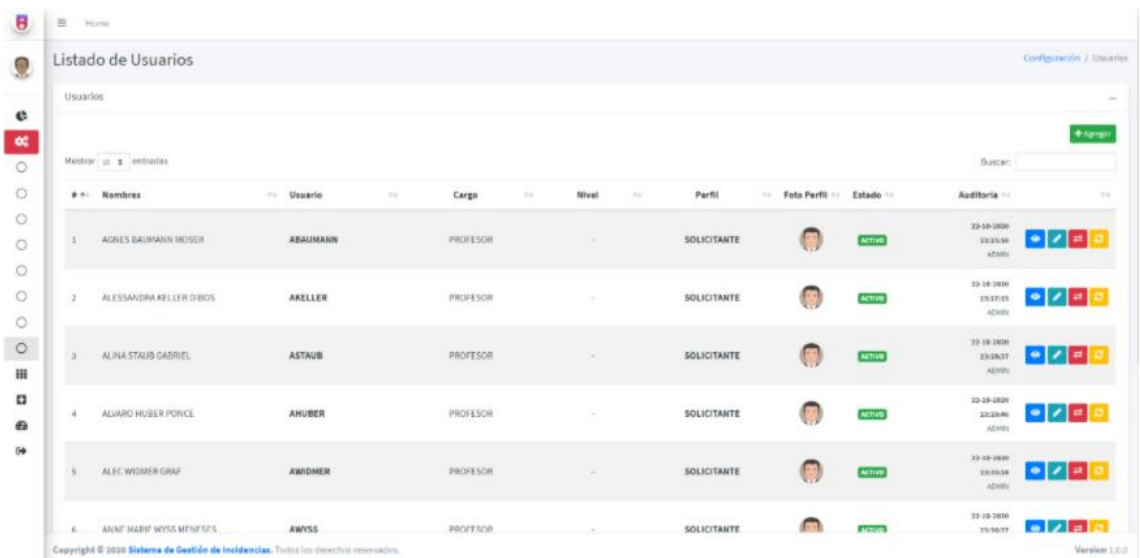
*Ilustración 94: Interfaz de Nivel*  
Fuente: Elaboración Propia



*Ilustración 95: Interfaz de Prioridad*  
Fuente: Elaboración Propia



*Ilustración 96: Interfaz de Perfiles  
Fuente: Elaboración Propia*



*Ilustración 97: Interfaz de Usuarios  
Fuente: Elaboración Propia*

## 1.7. REVISIÓN DEL SPRINT 1

El desarrollo de las historias de usuario es supervisado de acuerdo con la Tabla N° 16 donde se detalla el seguimiento de las horas trabajadas de acuerdo con los días programados para finalizar y cumplir con el objetivo del Sprint 1.

Tareas	Horas Iniciales	Dia 1	Dia 2	Dia 3	Dia 4	Dia 5	Dia 6	Dia 7	Dia 8	Dia 9	Total de Horas
Login de Usuarios al Sistema	8	8	0	0	0	0	0	0	0	0	8
Mantenimiento Área	8	0	8	0	0	0	0	0	0	0	8
Mantenimiento Cargo	8	0	0	8	0	0	0	0	0	0	8
Mantenimiento Categoría	8	0	0	0	6	0	0	0	0	0	6
Mantenimiento Sub-Categoría	8	0	0	0	0	8	0	0	0	0	8
Mantenimiento Nivel	8	0	0	0	0	0	6	0	0	0	6
Mantenimiento Prioridad	8	0	0	0	0	0	0	6	0	0	6
Mantenimiento Perfil	8	0	0	0	0	0	0	0	10	0	10
Mantenimiento Usuarios	8	0	0	0	0	0	0	0	0	10	10

Horas Restantes - Estimado	72	8	8	8	8	8	8	8	8	8	8
Horas Restantes - Actual	70	8	8	8	6	8	6	6	6	10	10

*Tabla 22: Resumen del Sprint 1  
Fuente: Elaboración Propia*

De acuerdo a la tabla N° 16, se puede observar que se cumplió con el desarrollo del primer sprint, más aún que las horas cumplidas para cada tarea se realizaron en menos tiempo de acuerdo a las horas iniciales estimadas para su desarrollo.

### 1.8. RETROSPECTIVA DEL SPRINT 1

Según Schwaber K. y Sutherland J. (2016, p.14), El equipo de SCRUM puede aprovechar esta oportunidad para realizar una auto inspección y desarrollar un plan de mejora para resolver en el próximo Sprint. Aunque se pueden realizar mejoras en cualquier momento, la Retrospectiva de Sprint llevó a cabo un evento especial para este propósito, enfocándose en inspecciones y ajustes. Tiene lugar después de la revisión de Sprint y antes de la próxima reunión de planificación de Sprint. De esta forma, en este estudio se evaluará la calidad de los resultados obtenidos en cada Sprint en la Tabla N° 17.

¿Qué salió bien en la presentación del Sprint 1?	¿Qué no salió bien en la presentación del Sprint1?	¿Qué vamos a implementar en la próxima presentación?
El sistema de inicio de sesión de usuario proporciona una buena seguridad, porque la contraseña registrada por el usuario está directamente encriptada en la base de datos y tiene derechos de acceso. Además de la verificación en el registro de usuarios, también se verifica la existencia del usuario registrado en la base de datos.	En el mantenimiento de Usuarios, se presentaron problemas para ordenar los campos del formulario debido a que tiene relaciona con algunos mantenimientos	En la próxima presentación se dará a conocer el desarrollo de registro de cada Ticket creado, casos que fueron cerrados, y los que aún están pendientes para finalmente exportar los registros de las atenciones realizadas en un determinado periodo.

*Tabla 23: Retrospectiva del Sprint 1  
Fuente: Elaboración Propia*





## Acta de Reunión de Sprint 1

### Acta de Reunión

Fecha: 12 de setiembre ,2020

En la presente reunión se presentaron los prototipados sobre el sprint 1, acerca de los diseños, de vistas, los roles o perfiles de los usuarios. Esto para que puedan tener una idea del funcionamiento de idea para el Sistema Web.

En dicha reunión se describieron ciertos requerimientos adicionales para los prototipos sobre el Diseño N° 1 en adelante y posteriormente para que impacte en el desarrollo. Luego de esto se procede a la codificación teniendo como guía los prototipos presentados, los requerimientos adicionales solicitados, y las herramientas para poder trabajar e iniciar.

Se acordó también durante la reunión, que el administrador / soporte tendrían que visualizar la mayoría de pestañas dentro del panel, para tener un cotejo de como va realizándose el proceso de las incidencias.

Luego de todo lo presentado al administrador de la empresa, accedió a su conformidad e inicio inmediato del desarrollo del Sistema Web.

Finalmente, cabe decir que estarán a la espera del desarrollo del Sprint 1, e inicio de pruebas unitarias con data ingresada por personal del Colegio.



Ricardo Estrada Mejía  
Jefe de Tecnologías de Información

The image shows a handwritten signature in blue ink over a circular stamp. The stamp contains the text 'Asociación Colegio Pestalozzi' around the top edge, 'JEFE DE TI' in the center, and 'Ricardo Estrada Mejía' and 'Jefe de Tecnologías de Información' around the bottom edge.

*Ilustración 98: Acta de Reunión de Sprint 1*

## 2. SPRINT 2:

## 2.1. HISTORIAS DE USUARIOS

HISTORIA DE USUARIO		PRIORIDAD	Estimado
<b>Numero:</b> 10	<b>Usuario:</b> Administrador / Soporte / Solicitante	Alta	5 días
<b>Nombre de la Historia:</b> Registro de Solicitud de Tickets			
<b>Programador Responsable:</b> Carlos Alvarado / Luis Dorador			
<b>Descripción:</b> El Administrador / Solicitante / Soporte puede gestionar (Solicitar) las Solicitud de un nuevo incidente realizado por cualquier personal Administrativo de la Institución. El administrador /Soporte puede visualizar el Ticket generado.			
<b>Como Probarlo:</b>			
<ul style="list-style-type: none"> <li>- El Administrador / Soporte / Solicitante ingresa al Sistema.</li> <li>- Selecciona la opción de menú Solicitar.</li> <li>- Visualiza formulario para registrar Ticket</li> <li>- Rellena todos los campos requeridos y le da en Guardar.</li> <li>- Su Ticket automáticamente llega a la Bandeja del área de TI (soporte) para su atención.</li> </ul>			

*Tabla 24: Historia de Usuario 10  
Fuente: Elaboración Propio*

HISTORIA DE USUARIO		PRIORIDAD	Estimado
<b>Numero:</b> 11	<b>Usuario:</b> Administrador / Soporte / Solicitante	Alta	5 días
<b>Nombre de la Historia:</b> Visualización de bandeja de Tickets.			
<b>Programador Responsable:</b> Carlos Alvarado / Luis Dorador			

<p><b>Descripción:</b> El Sistema permitirá visualizar la bandeja de tickets mediante un archivo Excel el detalle de todos los Tickets creados en el rango de fechas que se seleccione.</p>
<p><b>Como Probarlo:</b></p> <ul style="list-style-type: none"> <li>- El Administrador / Soporte / Solicitante ingresa al Sistema.</li> <li>- Selecciona la opción de menú Bandeja</li> <li>- Se visualizará los tickets generados por perfiles.</li> <li>- Administrador / Soporte comienza con la atención de los tickets según su categoría</li> <li>- El Solicitante comienza con la seguimiento de los tickets según su categoría</li> </ul>

*Tabla 25: Historia de Usuario 11  
Fuente: Elaboración Propio*

HISTORIA DE USUARIO		PRIORIDAD	Estimado
<b>Numero:</b> 12	<b>Usuario:</b> Administrador / Soporte	Alta	5 días
<b>Nombre de la Historia:</b> Seguimiento de Tickets			
<b>Programador Responsable:</b> Carlos Alvarado / Luis Dorador			
<b>Descripción:</b> El administrador /Soporte revisa en la Bandeja los Tickets generados para poder dar Inicio a la atención.			
<b>Como Probarlo:</b>			
<ul style="list-style-type: none"> <li>- El Administrador / Soporte / Solicitante ingresa al Sistema.</li> <li>- Selecciona la opción de menú Bandeja</li> <li>- El Administrador / Soporte revisa por estado cuales son los más prioritarios.</li> <li>- El Administrador / Soporte inicia con la atención seguida del Progreso, dar con lo Solucionado hasta llegar a cerrar el Ticket.</li> <li>- El solicitante visualiza el estado e ingresa comentarios en su ticket.</li> </ul>			

*Tabla 26: Historia de Usuario 12  
Fuente: Elaboración Propio*

## 2.2. SPRINT BACKLOG

En el siguiente sprint tiene como historia a realizar el formulario de generación de Ticket y la bandeja donde se visualizarán para poder hacer seguimiento a los tickets el cual se aprecia en la tabla a continuación.

Sprint	RF	H.U.	Prioridad	T. Estimado	Responsables
Sprint 2	10	Registro de Solicitud de Ticket	Alta	40	Carlos Alvarado / Luis Dorador
	11	Visualización de bandeja de Tickets	Alta	40	Carlos Alvarado / Luis Dorador
	12	Seguimiento de Tickets	Alta	40	Carlos Alvarado / Luis Dorador

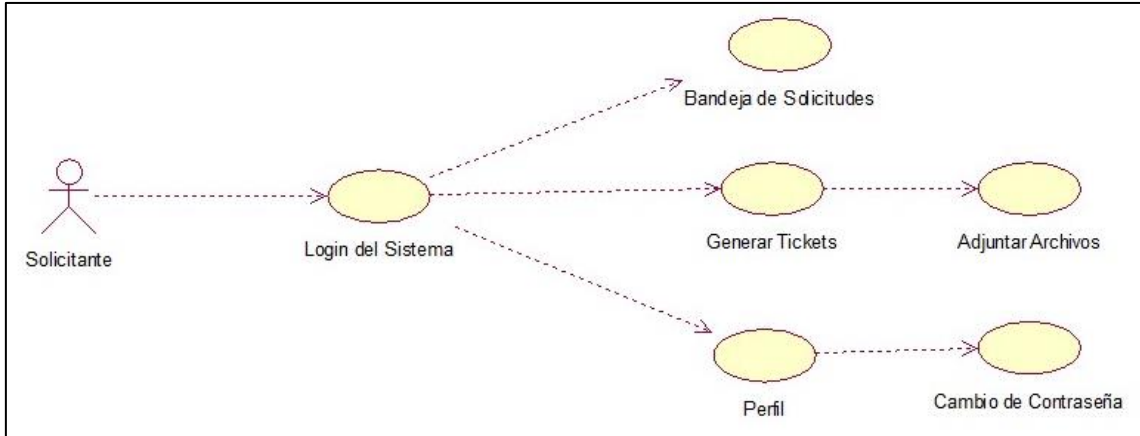
*Tabla 27: Product Backlog Sprint 2  
Fuente: Elaboración Propia*

### 2.3. TRABAJO DE DESARROLLO DE SPRINT

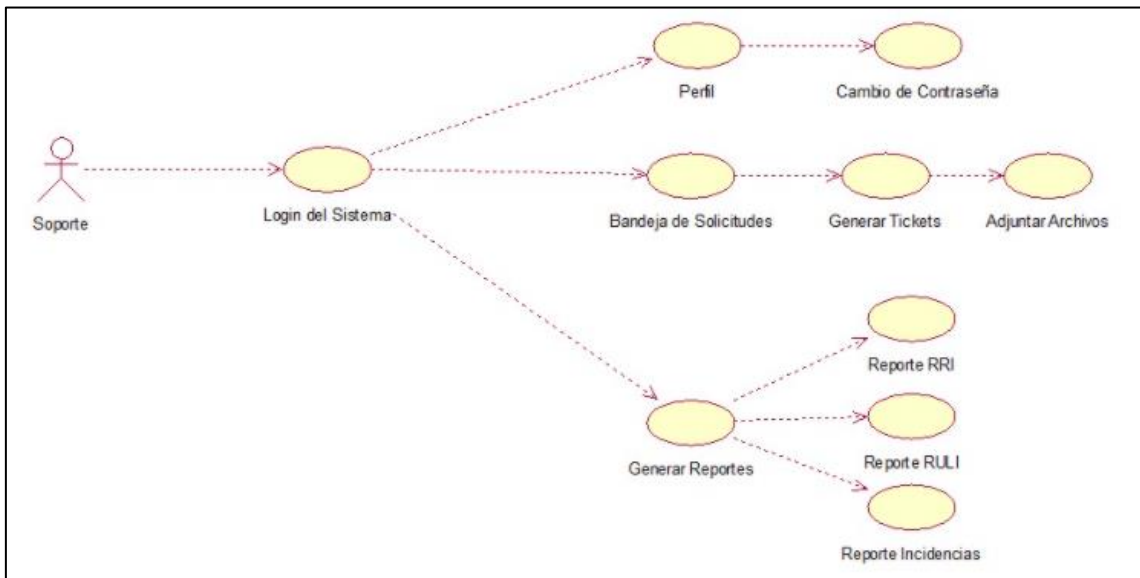
#### 2.3.1. Análisis del Sprint 2

Para el sprint 2 se analiza el funcionamiento del sistema en base al módulo de solicitar un nuevo Ticket. Teniendo como interpretación lo siguiente:

- El usuario con el Perfil de Solicitante, es el encargado de crear un nuevo Ticket para el área de TI, donde tendrá que rellenar todos los campos que se soliciten.
- El administrador / Soporte va poder visualizar en la pestaña de Bandeja, los nuevos Tickets creados para poder llegar a atenderlos.
- Luego de ser atendidos los Tickets se procede a dar como solucionado y cerrado respectivamente.



*Ilustración 99: Caso de Uso de Solicitante Sprint 2  
Fuente: Elaboración Propia*



*Ilustración 100: Caso de Uso de Soporte Sprint 2  
Fuente: Elaboración Propia*

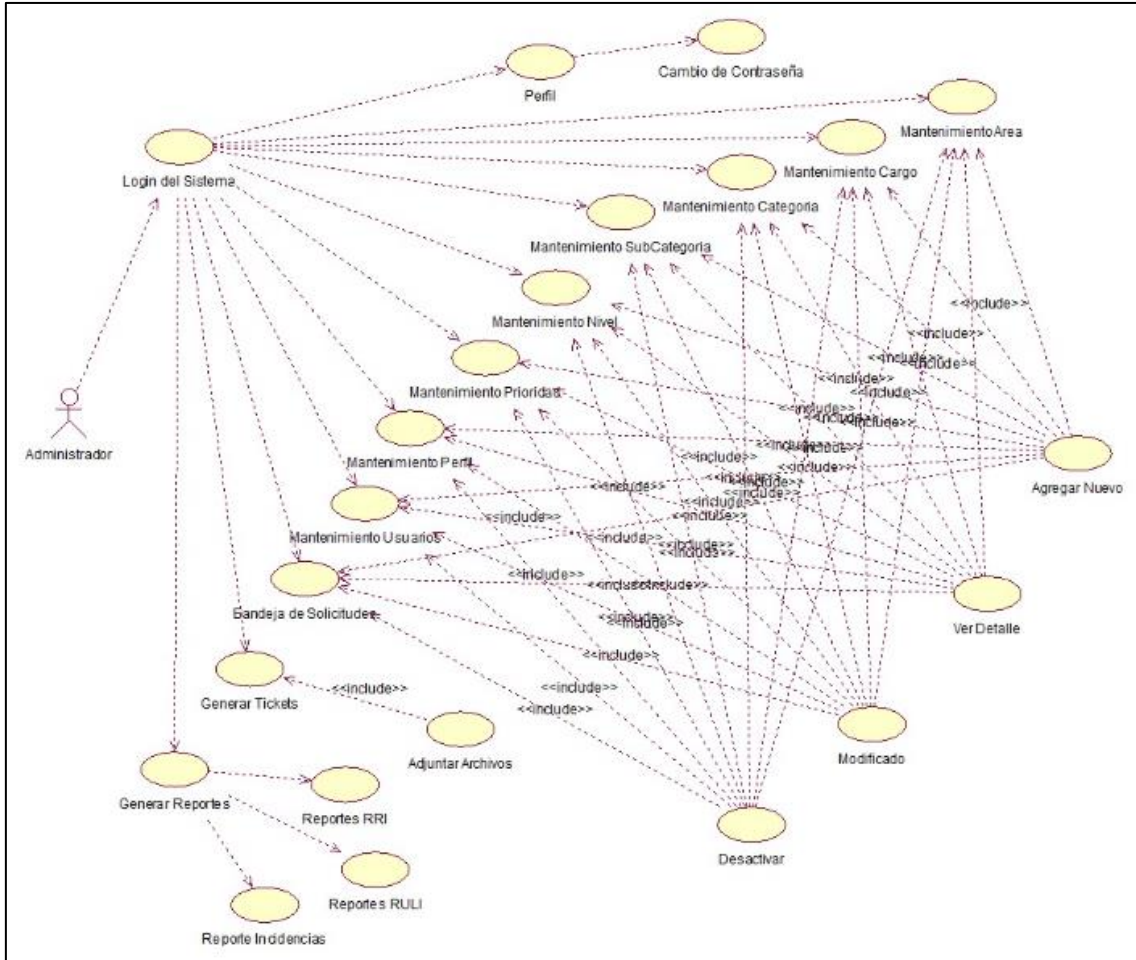


Ilustración 101: Caso de Uso de Administrador Sprint 2  
Fuente: Elaboración Propia

### 2.3.2. Diseño del Sprint 2

#### a. Diagrama Entidad – Relación de Sprint 2

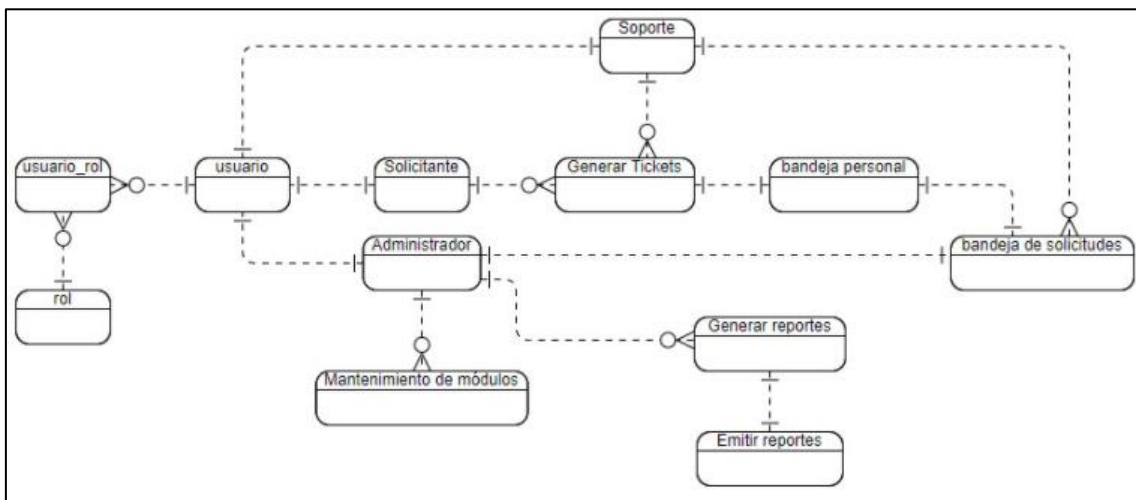
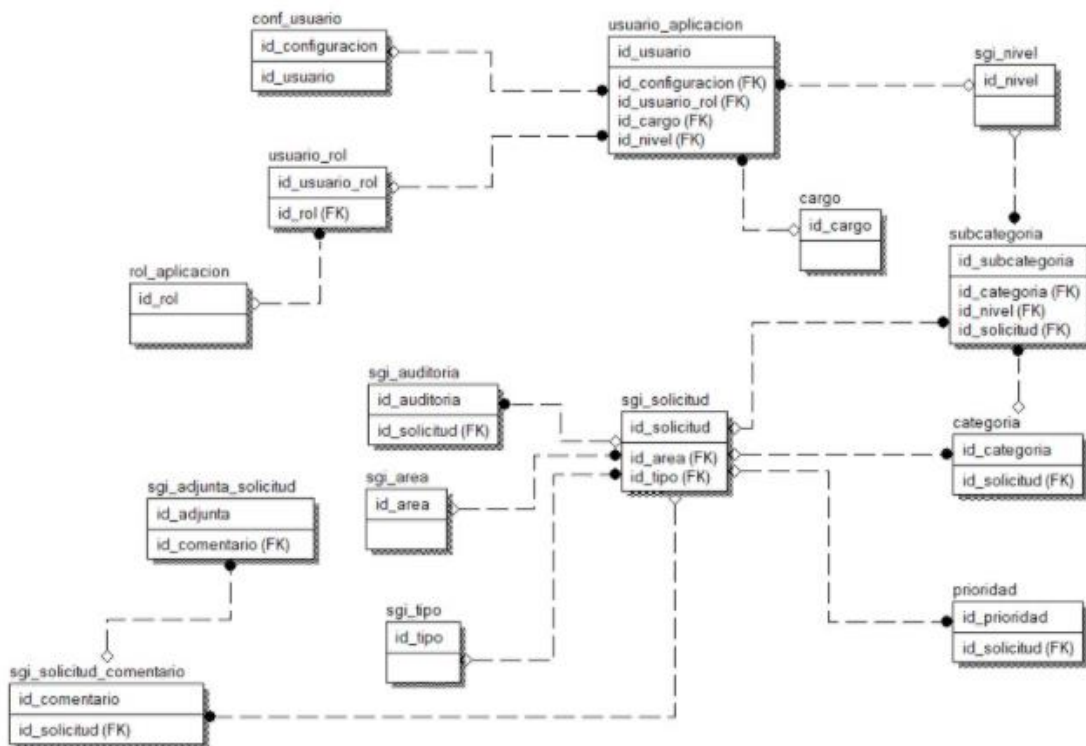


Ilustración 102: Diagrama Entidad – Relación de Sprint 2  
Fuente: Elaboración Propia

## b. Modelo Lógico

Es un prototipo de una base de datos real, por lo que los usuarios pueden verificar los resultados del diseño. De esta forma, podemos verificar el modelo antes de iniciar la construcción física de la base de datos. Este modelo solo se desarrolla en base a Sprint 3 para satisfacer las necesidades del cliente.

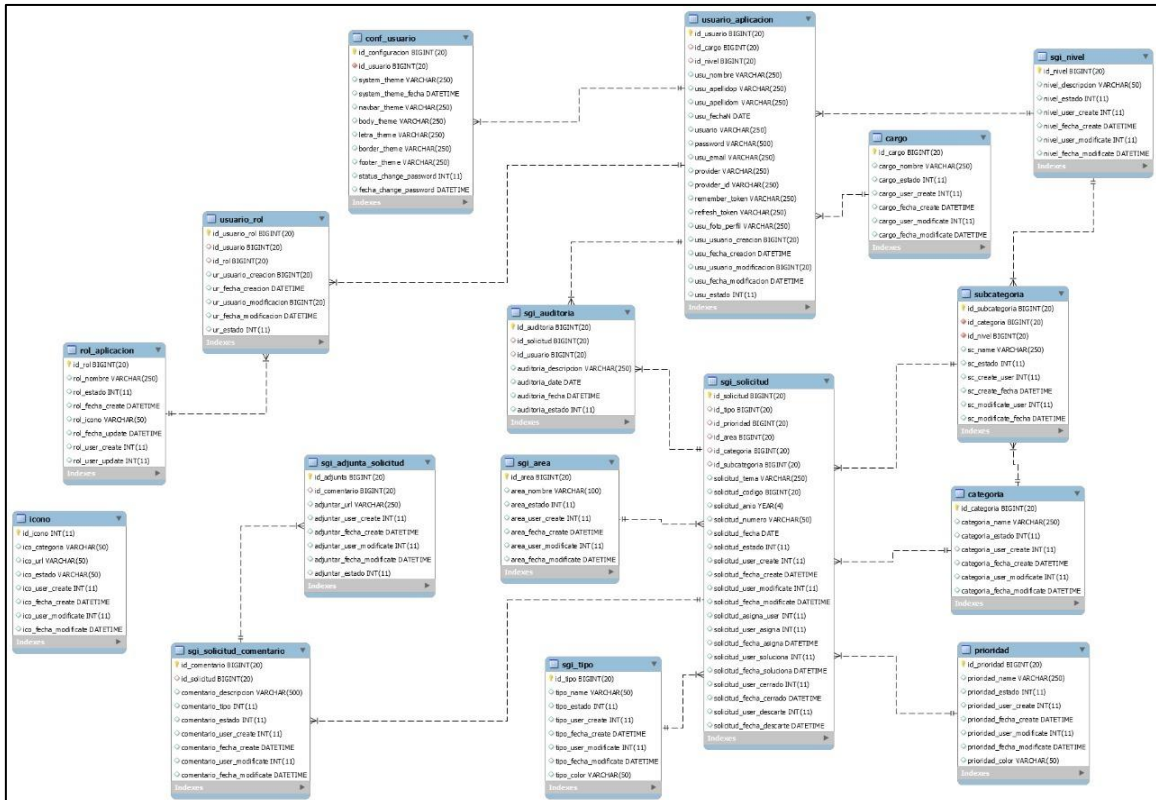


*Ilustración 103: Modelo Lógico de Sprint 2  
Fuente: Elaboración Propia*

## c. Modelo Físico

Detalla las relaciones básicas, la estructura de almacenamiento y los métodos de acceso que se utilizarán para acceder a los datos de manera eficiente. El diseño de esta relación se debe a una comprensión detallada de todas las funciones proporcionadas por el Sprint 2 (ver Figura XX)





*Ilustración 104: Modelo Físico de la BD Sprint 2  
Fuente: Elaboración Propia*

## 2.4. PROTOTIPOS DE DISEÑO DEL SPRINT 2

### a. Registro de Solicitud de Ticket

Se visualiza el prototipo donde los encargados de Soporte y Solicitantes podrán generar los tickets de incidencias para que posteriormente puedan dar el seguimiento

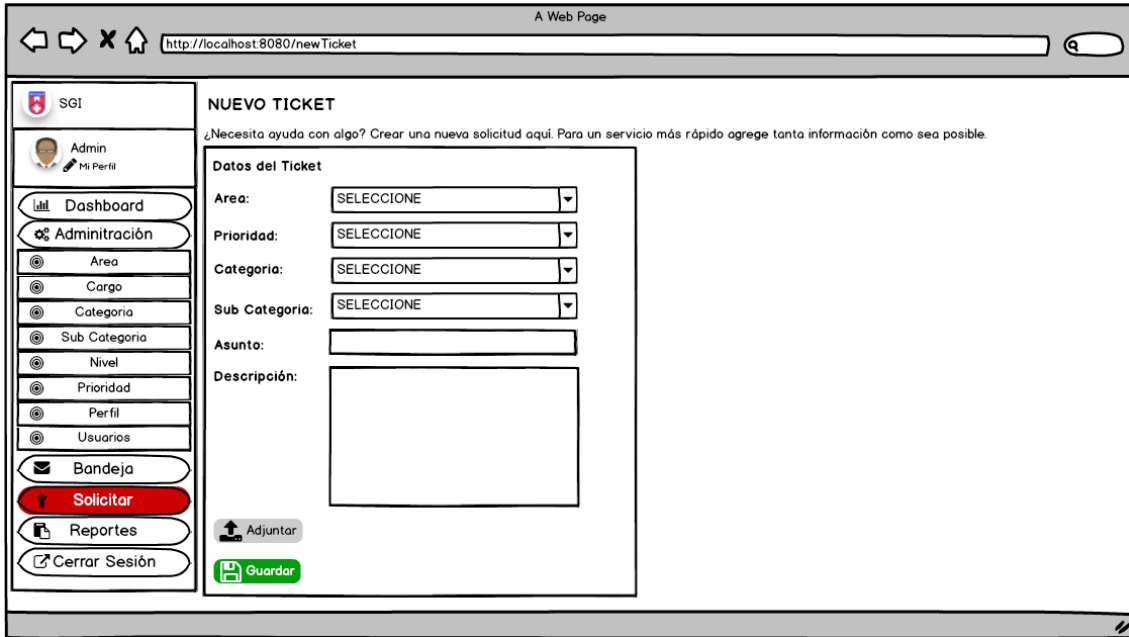


Ilustración 105: Registro de Solicitud de Tickets  
Fuente Elaboración Propia

## b. Visualización de Bandeja de Tickets

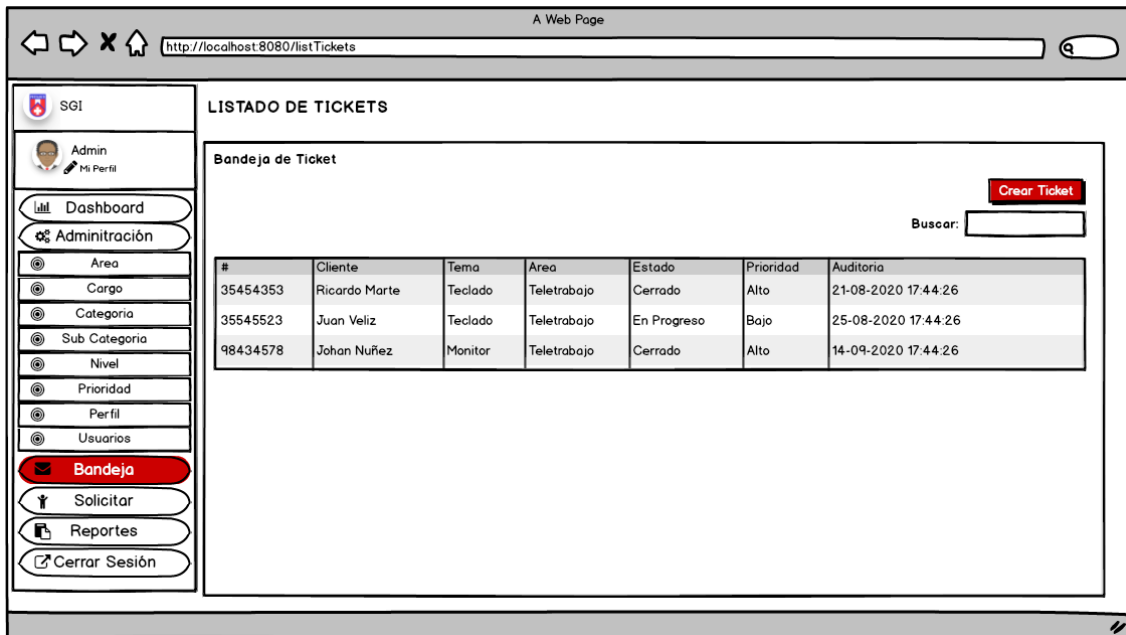
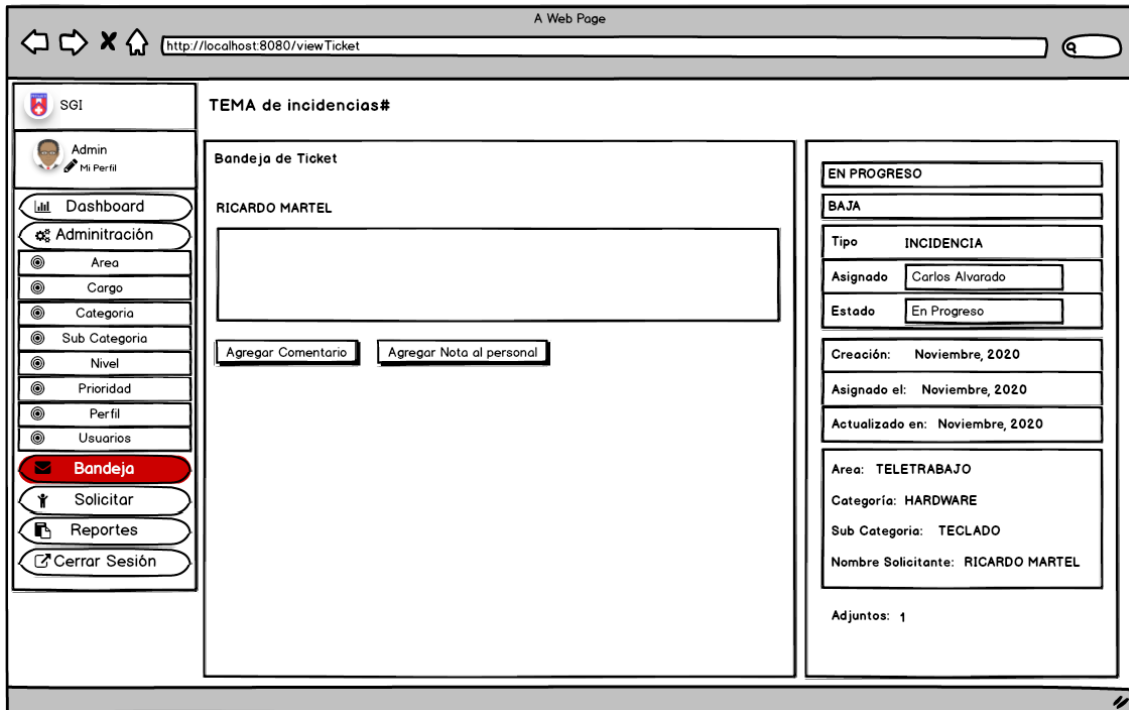


Ilustración 106: Visualización de Bandeja de Tickets  
Fuente: Elaboración Propia

## c. Seguimiento de Tickets



*Ilustración 107: Seguimiento de Tickets  
Fuente: Elaboración Propia*

## 2.5. CODIFICACIÓN DEL SPRINT 2

### a. Registro de Solicitud de Ticket

En la vista de “Nuevo Ticket” el solicitante crear su solicitud en un formulario llenando ciertos datos necesarios para su registro.

```

19
20 public function newTicket(){
21     $solicitud = Ticket::newTicket();
22     return view('tickets.newTicket',compact('solicitud'));
23 }
24

```

*Ilustración 108: Controlador de Vistas de Nuevo Ticket  
Fuente: Elaboración Propia*

```

42 <div class="row">
43 <form enctype="multipart/form-data" id="formSolicitud" method="POST">
44 <div class="col-md-6 mr-1">
45 <div class="form-group">
46 <label>Área: <a title="Este campo es obligatorio" class="text-sm text-warning"><i class="fas fa-info-circle"></i> </a></label>
47 <input type="hidden" class="form-control input-sm req" readonly name="id_tipo" id="id_tipo" value="{{ $solicitud->tipo[0]->id_tipo }}">
48 <select class="form-control input-sm req" id="id_area" name="id_area">
49 <option value="0">Seleccione</option>
50 @foreach($solicitud->areas as $key => $value)
51 <option value="{{ $value->id_area }}">{{ $value->area_nombre }}</option>
52 @endforeach
53 </select>
54 </div>
55 </div>
56 <div class="col-md-6 mr-1">
57 <div class="form-group">
58 <label>Prioridad: <a title="Este campo es obligatorio" class="text-sm text-warning"><i class="fas fa-info-circle"></i> </a></label>
59 <select class="form-control input-sm req" id="id_prioridad" name="id_prioridad">
60 <option value="0">Seleccione</option>
61 @foreach($solicitud->prioridades as $key => $value)
62 <option value="{{ $value->id_prioridad }}">{{ $value->prioridad_name }}</option>
63 @endforeach
64 </select>
65 </div>
66 </div>
67 <div class="col-md-6 mr-1">
68 <div class="form-group">
69 <label>Categoría: <a title="Este campo es obligatorio" class="text-sm text-warning"><i class="fas fa-info-circle"></i> </a></label>
70 <select class="form-control input-sm req" id="id_categoria" name="id_categoria">
71 <option value="0">Seleccione</option>
72 @foreach($solicitud->categorias as $key => $value)
73 <option value="{{ $value->id_categoria }}">{{ $value->categoria_name }}</option>
74 @endforeach
75 </select>
76 </div>
77 </div>
78 <div class="col-md-6">
79 <div class="form-group">
80 <label>Sub categoría: <a title="Este campo es obligatorio" class="text-sm text-warning"><i class="fas fa-info-circle"></i> </a></label>
81 <select class="form-control input-sm req" id="id_subcategoria" name="id_subcategoria">
82 <option value="0">Seleccione</option>
83 </select>
84 </div>

```

*Ilustración 109: Vista en Blade I (HTML) de Nuevo Ticket  
Fuente: Elaboración Propia*

```

81 <select class="form-control input-sm req" id="id_subcategoria" name="id_subcategoria">
82 <option value="0">Seleccione</option>
83 </select>
84 </div>
85 </div>
86 <div class="col-md-12">
87 <div class="form-group">
88 <label>Asunto: <a title="Este campo es obligatorio" class="text-sm text-warning"><i class="fas fa-info-circle"></i> </a></label>
89 <input type="text" class="form-control input-sm req" name="solicitud_tema" id="solicitud_tema" maxlength="250">
90 <input type="hidden" class="form-control input-sm" name="id_usuario" id="id_usuario" value="{{ Auth::user()->id_usuario }}">
91 </div>
92 </div>
93 <div class="col-md-12">
94 <div class="form-group">
95 <label>Descripción: <a title="Este campo es obligatorio" class="text-sm text-warning"><i class="fas fa-info-circle"></i> </a></label>
96 <textarea class="form-control input-sm req" name="comentario_descripcion" id="comentario_descripcion" rows="8" cols="500"></textarea>
97 </div>
98 </div>
99 <div class="col-md-6 mr-1">
100 <div class="form-group">
101 <a class="btn btn-sm btn-default text-dark" id="btnAdjuntar" ><i class="fas fa-upload"></i> Adjuntar</a><input type="hidden" name="tipo" id="tipo" value="0">
102 </div>
103 </div>
104 <div class="col-md-12" id="div-adjuntar" hidden>
105 <div class="form-group alert alert-info">
106 <span class="text-sm">Tamaño Máximo 2MB:</span>
107 @for($i=0;$i<3;$i++)
108 <input class="form-control input-sm col-md-6 mr-1 border-0 bg-info" type="file" id="file_adjuntar[{{ $i }}" name="file_adjuntar[]" accept="image/png, image/jpeg,
109 image/jpg">
110 @endfor
111 </div>
112 </form>
113 </div>
114 <div class="row">
115 <div class="col-md-3">
116 <a class="btn btn-sm btn-success text-white" id="saveSolicitud"><i class="fas fa-save"></i> GUARDAR</a>
117 </div>
118 </div>
119 </div>

```

*Ilustración 110: Vista en Blade II (HTML) de Nuevo Ticket  
Fuente: Elaboración Propia*

```

84
85 public static function newTicket(){
86     $solicitud = new Ticket();
87     $solicitud->categorias = Categoria::getCategoriaActivate();
88     $solicitud->subcategorias = SubCategoria::getSCActivate();
89     $solicitud->tipo = Tipo::getTipos();
90     $solicitud->prioridades = Prioridad::getPrioridad();
91     $solicitud->areas = Area::getAreas();
92     return $solicitud;
93 }
94

```

*Ilustración 111: Modelo de la vista de registro de Ticket  
Fuente: Elaboración Propia*

```

11
12 public function saveSolicitud(){
13     $ticket = Ticket::saveSolicitud();
14     return $ticket;
15 }
16

```

*Ilustración 112: Controlador de guardado de Ticket  
Fuente: Elaboración Propia*

```

95 public static function saveSolicitud(){
96     try {
97         $codigo_max = Ticket::select('solicitud_codigo')
98             ->where('solicitud_fecha',date('Y-m-d'))
99             ->orderBy('solicitud_codigo','desc')
100             ->first();
101
102         $user = User::find(request()->id_usuario);
103
104         if(!$codigo_max){
105             $codigo_max = new Ticket();
106             $codigo_max->solicitud_codigo = 0;
107         }
108
109         $solicitud = new Ticket();
110         $solicitud->id_tipo = request()->id_tipo;
111         $solicitud->id_prioridad = request()->id_prioridad;
112         $solicitud->id_categoria = request()->id_categoria;
113         $solicitud->id_subcategoria = request()->id_subcategoria;
114         $solicitud->id_area = request()->id_area;
115         $solicitud->solicitud_tema = mb_strtoupper(request()->solicitud_tema);
116         $solicitud->solicitud_user_create = request()->id_usuario;
117         $solicitud->solicitud_fecha_create = now();
118         $solicitud->solicitud_estado = 1;
119         $solicitud->solicitud_anio = date('Y');
120         $solicitud->solicitud_fecha = date('Y-m-d');
121         $solicitud->solicitud_codigo = ($codigo_max->solicitud_codigo + 1);
122         if(strlen($solicitud->solicitud_codigo) == 1){
123             $solicitud->solicitud_numero = date('Y').date('m').date('d').'0'.$solicitud->solicitud_codigo;
124         }else if(strlen($solicitud->solicitud_codigo) == 2){
125             $solicitud->solicitud_numero = date('Y').date('m').date('d').$solicitud->solicitud_codigo;
126         }
127
128         $solicitud->save();
129
130         $comentario = new Comentario();
131         $comentario->id_solicitud = $solicitud->id_solicitud;
132         $comentario->comentario_descripcion = request()->comentario_descripcion;
133         $comentario->comentario_tipo = 1;
134         $comentario->comentario_estado = 1;

```

*Ilustración 113: Modelo I de la Función de registro de Ticket  
Fuente: Elaboración Propia*

```

132     $comentario->comentario_id_estructura = request()->comentario_id_estructura;
133     $comentario->comentario_tipo = 1;
134     $comentario->comentario_estado = 1;
135     $comentario->comentario_user_create = request()->id_usuario;
136     $comentario->comentario_fecha_create = now();
137     $comentario->save();
138
139     $doc = request()->file('file_adjuntar');
140     if(!empty($doc)){
141         foreach($doc as $photo){
142             $extension = pathinfo($photo->getClientOriginalName(),PATHINFO_EXTENSION);
143             $nombre_now = strtotime(date('Y-m-d H:i:s'));
144             $nombre = $nombre_now.'.'.$extension;
145
146             \Storage::disk('public')->put('solicitud/'.$solicitud->solicitud_numero.'/'.$nombre, \File::get($photo));
147
148             $adjunto = new Adjunto();
149             $adjunto->id_comentario = $comentario->id_comentario;
150             $adjunto->adjuntar_url = 'storage/solicitud/'.$solicitud->solicitud_numero.'/'.$nombre;
151             $adjunto->adjuntar_user_create = request()->id_usuario;
152             $adjunto->adjuntar_fecha_create = now();
153             $adjunto->adjuntar_estado = 1;
154             $adjunto->save();
155         }
156     }
157
158     $auditoria = new Auditoria();
159     $auditoria->id_solicitud = $solicitud->id_solicitud;
160     $auditoria->id_usuario = request()->id_usuario;
161     $auditoria->auditoria_descripcion = 'El usuario '.$user->usu_nombre.' '.$user->usu_apellido.' '.$user->usu_apellido.' ha registrado la solicitud '.$solicitud->
162     >solicitud_numero;
163     $auditoria->auditoria_date = date('Y-m-d');
164     $auditoria->auditoria_fecha = now();
165     $auditoria->auditoria_estado = 1;
166     $auditoria->save();
167
168     return array("error">0,"mensaje">"Se Registró exitosamente el Ticket ".$solicitud->solicitud_numero,"tipo">"success","titulo">"¡Éxito!");
169 } catch (\Exception $e) {
170     return array("error">1,"mensaje">"Existe un problema de conexión, inténtelo nuevamente","tipo">"error","titulo">"¡Oops!");
171 }
172 }

```

*Ilustración 114: Modelo II de la Función de registro de Ticket  
Fuente: Elaboración Propia*

## **b. Visualización de Bandeja de Ticket**

```

1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use App\Models\Ticket;
7
8
9  class TicketController extends Controller{
10
11     public function __construct(){
12         $this->middleware('auth');
13     }
14
15     public function listTickets(){
16         $list_ticket = Ticket::getList();
17         return view('bandeja.listado',compact('list_ticket'));
18     }
19
20     public function viewTicket($numero){
21         $solicitud = Ticket::getTicket($numero);
22         return view('tickets.viewTicket',compact('solicitud'));
23     }
24 }

```

*Ilustración 115: Controlador de Vistas de Gestión de Ticket  
Fuente: Elaboración Propia*

```

43 <table class="table table-striped projects display nowrap" id="tblRole" width="100%">
44 <thead>
45 <tr>
46 <th style="width: 1%">#</th>
47 <th style="width: 20%">Cliente</th>
48 <th style="width: 30%" class="text-left">Tema</th>
49 <th style="width: 30%" class="text-left">Area</th>
50 <th style="width: 8%" class="text-left">Estado</th>
51 <!-- <th style="width: 8%" class="text-left">Tipo</th> -->
52 <th style="width: 8%" class="text-left">Prioridad</th>
53 <th style="width: 15%" class="text-right">Auditoria</th>
54 </tr>
55 </thead>
56 <tbody>
57 @foreach ($list_ticket as $key => $value)
58 <tr>
59 <td><b>#</b>{{ $value->solicitud_numero }}</td>
60 <td><span class="badge bg-primary badge-status" title="PRIORIDAD: {{ $value->cliente }}"><label class="h6">{{ $value->cliente }}</label></span>
61 <td><a href="{{ route('viewTicket', ['numero'=>$value->solicitud_numero]) }}">{{ $value->solicitud_tema }}</a></td>
62 <td class="text-left">{{ $value->area_nombre }}</td>
63 <td class="text-left"><span class="badge {{ $value->color_estado }}" badge-status="ESTADO: {{ $value->nombre_estado }}"><label class="h6">{{ $value->nombre_estado }}</label></span> {{ $value->nombre_estado }}</td>
64 <!-- <td class="text-left"><span class="badge {{ $value->tipo_color }}" badge-status="TIPO: {{ $value->tipo }}"><label class="h6">{{ $value->tipo }}</label></span> {{ $value->tipo }}</td> -->
65 <td class="text-left"><span class="badge {{ $value->prioridad_color }}" badge-status="PRIORIDAD: {{ $value->prioridad }}"><label class="h6">{{ $value->prioridad }}</label></span> {{ $value->prioridad }}</td>
66 <td class="text-right"><small><b>{{ $value->solicitud_fecha_create_dia }}</b><br>{{ $value->solicitud_fecha_create_hora }}</b><br>{{ $value->nombre_creador.' '.$value->apellidop_creador.' '.$value->apellidom_creador }}</small></td>
67 </tr>
68 @endforeach
69 </tbody>
70 </table>

```

*Ilustración 116: Vista de Bandeja de Ticket  
Fuente: Elaboración Propia*

```

58 public static function getIist(){
59 $list_ticket = Ticket::select('sgi_solicitud.*','u.usu_nombre as nombre_creador','u.usu_apellidop as apellidop_creador','u.usu_apellidom as
60 apellidom_creador','u.usuario as usuario_creador','u.usu_email as email_creador','sc.sc_name as subcategoria','c.categoria_name as categoria','p.prioridad_name
61 as prioridad','t.tipo_name as tipo','t.tipo_color','p.prioridad_color','a.area_nombre')
62 ->join('subcategoria as sc','sc.id_subcategoria','sgi_solicitud.id_subcategoria')
63 ->join('categoria as c','c.id_categoria','sgi_solicitud.id_categoria')
64 ->join('prioridad as p','p.id_prioridad','sgi_solicitud.id_prioridad')
65 ->join('sgi_area as a','a.id_area','sgi_solicitud.id_area')
66 ->join('sgi_tipo as t','t.id_tipo','sgi_solicitud.id_tipo')
67 ->join('usuario_aplicacion as u','u.id_usuario','sgi_solicitud.solicitud_user_create')
68 ->orderBy('solicitud_fecha_create','desc')
69 ->get();
70
71 foreach($list_ticket as $key => $value){
72 $value->nombre_estado = Ticket::getEstado($value->solicitud_estado);
73 $value->color_estado = Ticket::getColorEstado($value->solicitud_estado);
74 $value->cliente = $value->nombre_creador.' '.$value->apellidop_creador.' '.$value->apellidom_creador;
75 $value->solicitud_fecha_create_dia = date_format(date_create($value->solicitud_fecha_create),'d-m-Y');
76 $value->solicitud_fecha_create_hora = date_format(date_create($value->solicitud_fecha_create),'H:i:s');
77 }
78
79 if(Auth::user()->rol[0]->id_rol == env('ID_SOLICITANTE')){
80 $list_ticket = $list_ticket->where('solicitud_user_create',Auth::user()->id_usuario);
81 }
82
83 return $list_ticket;
84 }

```

*Ilustración 117: Modelo de Función de Listado de Ticket  
Fuente: Elaboración Propia*

```

35 <div class="row">
36   @foreach($solicitud->comentarios as $key => $value)
37     @if(in_array(Auth::user()->rol[0]->id_rol,$value->comentario_permisos))
38       <div class="col-md-1 mt-2">
39         <div class="form-group">
40           <span class="border rounded border-success p-2 bg-gradient-success">{{ (empty($value->usuario_apellido[0]))?$value->usuario_nombre[0]:$value->
41             *
42             usuario_nombre[0].$value->usuario_apellido[0] }}</span>
43         </div>
44         <div class="col-md-11 mt-1">
45           <div class="form-group">
46             <strong>{{ $value->comentario_usuario }}</strong> - <span>{{ $value->comentario_correo }}</span>
47           </div>
48           <div class="col-md-12 mb-1">
49             <textarea class="form-control input-sm" readonly name="name" rows="4" cols="500">{{ $value->comentario_descripcion }}</textarea>
50           </div>
51           @foreach($value->imagenes as $k => $val)
52             <div class="col-md-12 mt-2 mb-2 text-center mb-1">
53               solicitud_numero.'-'.($key+1) }}">
54             </div>
55           @endforeach
56         @endif
57       @endforeach
58     @if($solicitud->solicitud_estado == 1 || $solicitud->solicitud_estado == 2)
59       <div class="col-md-1 mt-2">
60         <div class="form-group">
61           <span class="border rounded border-primary p-2 bg-gradient-primary">{{ (empty(Auth::user()->usu_apellido[0]))?Auth::user()->
62             *
63             usu_nombre[0]:Auth::user()->usu_nombre[0].Auth::user()->usu_apellido[0] }}</span>
64         </div>
65         <div class="col-md-11 mt-2">
66           <div class="form-group">
67             <strong>{{ Auth::user()->getFullNameAttribute() }}</strong> - <span>{{ Auth::user()->usu_email }}</span>
68           </div>
69         </div>

```

*Ilustración 118: Vista I del módulo de Gestión de Ticket  
Fuente: Elaboración Propia*

```

69 </div class="col-md-12 mt-2">
70 <a id="addComentario" class="btn btn-sm btn-info text-white"> Agregar Comentario</a>
71 @if(Auth::user()->rol[0]->id_rol == env('ID_ADMINISTRADOR'))
72 <a id="addNota" class="btn btn-sm btn-secondary text-white"> Agregar Nota al Personal</a>
73 @endif
74 <form enctype="multipart/form-data" id="formComentario" method="POST">
75 <textarea id="comentario_descripcion" hidden class="form-control input-sm mb-2" name="comentario_descripcion" rows="4" cols="500"
76   placeholder="Mensaje..."></textarea>
77 <div class="form-group alert alert-info" hidden id="div-Adjuntar">
78   <span class="text-sm">Tamaño Máximo 2MB:</span>
79   @for($i=0;$i<3;$i++)
80     <input class="form-control input-sm col-md-6 mr-1 border-0 bg-info" type="file" id="file_adjuntar[{{ $i }}" name="file_adjuntar[]" accept="image/
81     png, image/jpeg, image/jpg">
82   @endfor
83 </div>
84 <input type="hidden" name="comentario_tipo" id="comentario_tipo">
85 <input type="hidden" name="comentario_user_create" id="comentario_user_create" value="{{ Auth::user()->id_usuario }}">
86 <a id="cancelComentario" hidden class="btn btn-sm btn-secondary text-white mt-1"> Cancelar</a>
87 <a id="uptComentario" hidden class="btn btn-sm btn-primary text-white mt-1 addComentario" data-tipo="2"> Agregar</a>
88 @if(in_array(Auth::user()->rol[0]->id_rol,array(env('ID_ADMINISTRADOR'),env('ID_SOPORTE'))))
89 <a id="uptFinComentario" hidden class="btn btn-sm btn-success text-white mt-1 addComentario" data-tipo="3"> Agregar y Solucionar</a>
90 @endif
91 </form>
92 </div>
93 @endif
94 </div>
95 </div>

```

*Ilustración 119: Vista II del módulo de Gestión de Ticket  
Fuente: Elaboración Propia*



```

99 <div class="col-md-4">
100 <div class="card card-default {{ Auth::user()->getConf()->body_theme.' '.Auth::user()->getConf()->border_theme }}">
101 <div class="card-header">
102 <div class="row align-items-end">
103 <div class="col-md-6">
104 <span class="badge bg-secondary"><label class="h6 m-1">Ticket #{{ $solicitud->solicitud_numero }}</label></span>
105 </div>
106 </div>
107 </div>
108 <div class="card-body">
109 <div class="row">
110 <div class="col-md-12">
111 <table class="table table-bordered">
112 <tbody>
113 <tr>
114 <td><span class="badge {{ $solicitud->color_estado }} badge-status" title="ESTADO: {{ $solicitud->nombre_estado }}"><label class="h6">{{
+ $solicitud->nombre_estado[0] }}</label></span> {{ $solicitud->nombre_estado}}</td>
115 </tr>
116 <tr>
117 <td><span class="badge {{ $solicitud->prioridad_color }} badge-status" title="ESTADO: {{ $solicitud->prioridad }}"><label class="h6">{{
+ $solicitud->prioridad[0] }}</label></span> {{ $solicitud->prioridad}}</td>
118 </tr>
119 </tbody>
120 </table>
121 </div>
122 <div class="col-md-12">
123 <table class="table table-bordered">
124 <tbody>
125 <tr>
126 <td>
127 <div class="row">
128 <div class="col-md-4">
129 <label class="m-1"><i class="fas fa-bookmark"></i> Tipo</label>
130 </div>
131 <div class="col-md-8">
132 <p class="m-1">{{ $solicitud->tipo}}</p>
133 </div>
134 </div>

```

*Ilustración 120: Vista III del módulo de Gestión de Ticket  
Fuente: Elaboración Propia*

```

139 <div class="row">
140 <div class="col-md-4">
141 <label class="m-1"><i class="fas fa-user-tag"></i> Asignado</label>
142 </div>
143 <div class="col-md-8">
144 <select class="form-control input-sm" {{ (in_array(Auth::user()->rol[0]->id_rol,array(env('ID_ADMINISTRADOR'),env('ID_SOPORTE')) ||
+ in_array($solicitud->solicitud_estado,array(3,4)))?'disabled':'') }} name="id_asignado" id="id_asignado">
145 <option value="0">SELECCIONE</option>
146 @foreach($solicitud->usuarios as $key => $value)
147 <option value="{{ $value->id_usuario }}" {{ ($value->id_usuario == $solicitud->solicitud_asigna_user)?'selected':''}}>{{
+ mb_strtoupper($value->usu_nombre.' '.$value->usu_apellidoop.' '.$value->usu_apellidoom) }}</option>
148 @endforeach
149 </select>
150 </div>
151 </div>
152 </td>
153 </tr>
154 <tr>
155 <td>
156 <div class="row">
157 <div class="col-md-4">
158 <label class="m-1"><i class="fas fa-asterisk"></i> Estado</label>
159 </div>
160 <div class="col-md-8">
161 <select class="form-control input-sm" {{ ((Auth::user()->rol[0]->id_rol == env('ID_SOLICITANTE')) || in_array($solicitud->
+ solicitud_estado,array(4,5)))?'disabled':'') }} name="id_resolucion" id="id_resolucion">
162 <option value="0">SELECCIONE</option>
163 <option value="0" {{ ($solicitud->solicitud_estado == 1)?'selected':''}}>ABIERTO</option>
164 <option value="0" {{ ($solicitud->solicitud_estado == 2)?'selected':''}}>EN PROGRESO</option>
165 <option value="1" {{ ($solicitud->solicitud_estado == 3)?'selected':''}}>SOLUCIONADO</option>
166 <option value="3" {{ ($solicitud->solicitud_estado == 4)?'selected':''}}>CERRADO</option>
167 </select>
168 </div>
169 </div>
170 </td>
171 </tr>
172 </tbody>
173 </table>

```

*Ilustración 121: Vista IV del módulo de Gestión de Ticket  
Fuente: Elaboración Propia*

```

175 <div class="col-md-12">
176   <table class="table table-bordered">
177     <tbody>
178       <tr>
179         <td>
180           <div class="row">
181             <div class="col-md-4">
182               <label class="m-1"><i class="far fa-calendar-alt"></i> Creación</label>
183             </div>
184             <div class="col-md-8">
185               <p class="m-1">{{ $solicitud->mes_creacion }}</p>
186             </div>
187           </div>
188         </td>
189       </tr>
190       <tr>
191         <td>
192           <div class="row">
193             <div class="col-md-4">
194               <label class="m-1"><i class="far fa-calendar-alt"></i> Asignado el</label>
195             </div>
196             <div class="col-md-8">
197               <p class="m-1">{{ $solicitud->mes_asignacion }}</p>
198             </div>
199           </div>
200         </td>
201       </tr>
202       <tr>
203         <td>
204           <div class="row">
205             <div class="col-md-4">
206               <label class="m-1" style="white-space: nowrap;"><i class="far fa-calendar-alt"></i> Actualizado en</label>
207             </div>
208             <div class="col-md-8">
209               <p class="m-1">{{ $solicitud->mes_modificate }}</p>
210             </div>
211           </div>
212         </td>
213       </tr>
214     </tbody>
215   </table>
216 </div>

```

*Ilustración 122: Vista V del módulo de Gestión de Ticket  
Fuente: Elaboración Propia*

```

218 <table class="table table-bordered">
219   <tbody>
220     <tr>
221       <td>
222         <label>Área</label>
223         <p>{{ $solicitud->area_nombre }}</p>
224         <label>Categoría</label>
225         <p>{{ $solicitud->categoria }}</p>
226         <label>Sub Categoría</label>
227         <p>{{ $solicitud->subcategoria }}</p>
228         <label>Nombre Solicitante</label>
229         <p>{{ $solicitud->nombre_creador.' '.$solicitud->apellidop_creador.' '.$solicitud->apellidom_creador }}</p>
230       </td>
231     </tr>
232   </tbody>
233 </table>
234 </div>
235 <div class="col-md-12">
236   <table class="table table-bordered">
237     <tbody>
238       <tr>
239         <td>
240           <div class="row">
241             <div class="col-md-4">
242               <label class="m-1"><i class="fas fa-paperclip"></i> Adjuntos</label>
243             </div>
244             <div class="col-md-8">
245               <span class="badge badge-secondary">{{ $solicitud->contador }}</span>
246             </div>
247           </div>
248         </td>
249       </tr>
250     </tbody>
251   </table>
252 </div>
253 </div>
254 </div>

```

*Ilustración 123: Vista VI del módulo de Gestión de Ticket  
Fuente: Elaboración Propia*

### c. Seguimiento de Ticket

```
173 public static function updateTema(){
174     try {
175         $ticket = Ticket::find(request()->id_solicitud);
176         if($ticket->solicitud_tema == request()->solicitud_tema){
177             return array("error">1,"mensaje">"El Tema que intenta actualizar es el mismo por el cual intenta
178             + modificar.", "tipo">"warning", "titulo">"¡Alerta!", "solicitud_tema">=$ticket->solicitud_tema);
179         }else{
180             $ticket->solicitud_tema = mb_strtoupper(request()->solicitud_tema);
181             $ticket->solicitud_user_modificate = request()->id_usuario;
182             $ticket->solicitud_fecha_modificate = now();
183             $ticket->save();
184
185             $user = User::find(request()->id_usuario);
186
187             $auditoria = new Auditoria();
188             $auditoria->id_solicitud = $ticket->id_solicitud;
189             $auditoria->id_usuario = request()->id_usuario;
190             $auditoria->auditoria_descripcion = "El usuario '$user->usu_nombre.' '$user->usu_apellido.' '$user->usu_apellido.' ha actualizado el tema de la
191             + solicitud '$ticket->solicitud_numero.' a '$request()->solicitud_tema;
192             $auditoria->auditoria_date = date('Y-m-d');
193             $auditoria->auditoria_fecha = now();
194             $auditoria->auditoria_estado = 2;
195             $auditoria->save();
196
197             return array("error">0,"mensaje">"Se Actualizó exitosamente el Tema del Ticket ", $ticket-
198             + >solicitud_numero, "tipo">"success", "titulo">"¡Éxito!", "solicitud_tema">=$ticket->solicitud_tema);
199         }
200     } catch (\Exception $e) {
201         return array("error">1,"mensaje">"Existe un problema de conexión, inténtelo nuevamente", "tipo">"error", "titulo">"¡oops!");
202     }
203 }
```

*Ilustración 124: Modelo de actualización del Nombre del Tema del Ticket  
Fuente: Elaboración Propia*

```
202 public static function updateAsigna(){
203     try {
204         $ticket = Ticket::find(request()->id_solicitud);
205         if($ticket->solicitud_tema == request()->solicitud_tema){
206             return array("error">1,"mensaje">"El Usuario que intenta asignar ya esta asignado a esta
207             + solicitud.", "tipo">"warning", "titulo">"¡Alerta!", "solicitud_tema">=$ticket->solicitud_tema);
208         }else{
209             $ticket->solicitud_asigna_user = request()->id_asignado;
210             $ticket->solicitud_user_asigna = request()->id_usuario;
211             $ticket->solicitud_fecha_asigna = now();
212             $ticket->solicitud_estado = 2;
213             $ticket->save();
214
215             $user = User::find(request()->id_usuario);
216             $usuario = User::find(request()->id_asignado);
217
218             $auditoria = new Auditoria();
219             $auditoria->id_solicitud = $ticket->id_solicitud;
220             $auditoria->id_usuario = request()->id_usuario;
221             $auditoria->auditoria_descripcion = "El usuario '$user->usu_nombre.' '$user->usu_apellido.' '$user->usu_apellido.' ha asignado a '$usuario-
222             + >usu_nombre.' '$usuario->usu_apellido.' '$usuario->usu_apellido.' a la solicitud '$ticket->solicitud_numero;
223             $auditoria->auditoria_date = date('Y-m-d');
224             $auditoria->auditoria_fecha = now();
225             $auditoria->auditoria_estado = 3;
226             $auditoria->save();
227
228             return array("error">0,"mensaje">"Se Asignó exitosamente al usuario '$usuario->usu_nombre.' '$usuario->usu_apellido.' '$usuario->usu_apellido.' a la
229             + solicitud '$ticket->solicitud_numero, "tipo">"success", "titulo">"¡Éxito!", "solicitud_tema">=$ticket->solicitud_tema);
230         }
231     } catch (\Exception $e) {
232         return array("error">1,"mensaje">"Existe un problema de conexión, inténtelo nuevamente", "tipo">"error", "titulo">"¡oops!");
233     }
234 }
```

*Ilustración 125: Modelo de asignación de Usuario de Soporte  
Fuente: Elaboración Propia*



```

291     if(empty($value->comentario_user_modificate)){
292         $value->usuario_apellido = $value->apellido_creador;
293         $value->usuario_apellido = $value->apellido_creador;
294         $value->usuario_nombre = $value->nombre_creador;
295         $value->comentario_usuario = $value->nombre_creador.' '.$value->apellido_creador.' '.$value->apellido_creador;
296         $value->comentario_correo = $value->email_creador;
297     }else{
298         $value->comentario_usuario = $value->nombre_actualizador.' '.$value->apellido_actualizador.' '.$value->apellido_actualizador;
299         $value->comentario_correo = $value->email_actualizador;
300         $value->usuario_apellido = $value->apellido_actualizador;
301         $value->usuario_apellido = $value->apellido_actualizador;
302         $value->usuario_nombre = $value->nombre_actualizador;
303     }
304     $value->imagenes = Adjuntar::where('id_comentario',$value->id_comentario)
305         ->where('adjuntar_estado',1)
306         ->orderby('adjuntar_fecha_create','asc')
307         ->get();
308     $count = ($count + count($value->imagenes));
309 }
310 $ticket->contador = $count;
311
312 return $ticket;
313 }

```

*Ilustración 128: Modelo de vista III de Gestión de Ticket  
Fuente: Elaboración Propia*

```

314
315 public static function addComentario(){
316     try {
317         $user = User::find(request()->comentario_user_create);
318         $solicitud = Ticket::find(request()->id_solicitud);
319
320         $comentario = new Comentario();
321         $comentario->id_solicitud = request()->id_solicitud;
322         $comentario->comentario_descripcion = request()->comentario_descripcion;
323         $comentario->comentario_tipo = request()->comentario_tipo;
324         $comentario->comentario_estado = 1;
325         $comentario->comentario_user_create = request()->comentario_user_create;
326         $comentario->comentario_fecha_create = now();
327         $comentario->save();
328
329         $solicitud->solicitud_user_modificate = request()->comentario_user_create;
330         $solicitud->solicitud_fecha_modificate = now();
331         if($comentario->comentario_tipo == 3){
332             $solicitud->solicitud_user_soluciona = request()->comentario_user_create;
333             $solicitud->solicitud_fecha_soluciona = now();
334             $solicitud->solicitud_estado = 3;
335         }
336         $solicitud->save();
337
338         $doc = request()->file('file_adjuntar');
339         if(!empty($doc)){
340             foreach($doc as $photo){
341                 $extension = pathinfo($photo->getClientOriginalName(),PATHINFO_EXTENSION);
342                 $nombre_now = strtotime(date("Y-m-d H:i:s"));
343                 $nombre = $nombre_now.'.'.$extension;
344
345                 \Storage::disk('public')->put('solicitud/'.$solicitud->solicitud_numero.'/'.$nombre, \File::get($photo));
346
347                 $adjunto = new Adjuntar();
348                 $adjunto->id_comentario = $comentario->id_comentario;
349                 $adjunto->adjuntar_url = 'storage/solicitud/'.$solicitud->solicitud_numero.'/'.$nombre;
350                 $adjunto->adjuntar_user_create = request()->comentario_user_create;
351                 $adjunto->adjuntar_fecha_create = now();

```

*Ilustración 129: Modelo de Función I de agregar comentario en la Gestión de Ticket  
Fuente: Elaboración Propia*

```

352     $adjunto->adjuntar_estado = 1;
353     $adjunto->save();
354 }
355 }
356
357 if($comentario->comentario_tipo == 3){
358     $auditoria = new Auditoria();
359     $auditoria->id_solicitud = $solicitud->id_solicitud;
360     $auditoria->id_usuario = request()->comentario_user_create;
361     $auditoria->auditoria_descripcion = 'El usuario '.$user->usu_nombre.' '.$user->usu_apellidoop.' '.$user->usu_apellidoop.' ha respondido y ha dado como
    +
362     Solucionado la solicitud '.$solicitud->solicitud_numero;
363     $auditoria->auditoria_date = date('Y-m-d');
364     $auditoria->auditoria_fecha = now();
365     $auditoria->auditoria_estado = 6;
366     $auditoria->save();
367     $mensaje = "Se registró exitosamente el comentario y se dio por solucionado exitosamente la Solicitud ".$solicitud->solicitud_numero;
368 }else{
369     $auditoria = new Auditoria();
370     $auditoria->id_solicitud = $solicitud->id_solicitud;
371     $auditoria->id_usuario = request()->comentario_user_create;
372     $auditoria->auditoria_date = date('Y-m-d');
373     $auditoria->auditoria_fecha = now();
374     if($comentario->comentario_tipo == 4){
375         $auditoria->auditoria_descripcion = 'El usuario '.$user->usu_nombre.' '.$user->usu_apellidoop.' '.$user->usu_apellidoop.' ha agregado una nota al personal en
    +
376         la solicitud '.$solicitud->solicitud_numero;
377         $auditoria->auditoria_estado = 5;
378         $mensaje = "Se registró exitosamente la Nota al Personal en la Solicitud ".$solicitud->solicitud_numero;
379     }else{
380         $auditoria->auditoria_descripcion = 'El usuario '.$user->usu_nombre.' '.$user->usu_apellidoop.' '.$user->usu_apellidoop.' ha respondido a la solicitud
    +
381         '.$solicitud->solicitud_numero;
382         $auditoria->auditoria_estado = 4;
383         $mensaje = "Se registró exitosamente el comentario en la Solicitud ".$solicitud->solicitud_numero;
384     }
385     $auditoria->save();
386 }
387
388 return array("error"=>0,"mensaje"=>$mensaje,"tipo"=>"success","titulo"=>"¡Éxito!");
389 } catch (Exception $e) {
390     return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente","tipo"=>"error","titulo"=>"¡Oops!");
391 }

```

*Ilustración 130: Modelo de Función II de agregar comentario en la Gestión de Ticket*  
Fuente: Elaboración Propia

```

391     public static function updateStatus(){
392         //id_resolucion = 1 -> SOLUCIONADO
393         //id_resolucion = 2 -> SOLUCIONADO Y CERRADO
394         //id_resolucion = 3 -> CERRADO
395         try{
396             $user = User::find(request()->id_usuario);
397             $solicitud = Ticket::find(request()->id_solicitud);
398
399             $auditoria = new Auditoria();
400             $auditoria->id_solicitud = $solicitud->id_solicitud;
401             $auditoria->id_usuario = request()->id_usuario;
402             $auditoria->auditoria_date = date('Y-m-d');
403             $auditoria->auditoria_fecha = now();
404
405             if(request()->id_resolucion == 1){
406                 $solicitud->solicitud_user_soluciona = request()->id_usuario;
407                 $solicitud->solicitud_fecha_soluciona = now();
408                 $solicitud->solicitud_estado = 3;
409
410                 $auditoria->auditoria_descripcion = 'El usuario '.$user->usu_nombre.' '.$user->usu_apellidoop.' '.$user->usu_apellidoop.' ha marcado como solucionada la
    +
411                 solicitud '.$solicitud->solicitud_numero;
412                 $auditoria->auditoria_estado = 6;
413                 $mensaje = "Se Marcó como Solucionada la Solicitud ".$solicitud->solicitud_numero.' exitosamente.';
414             }else if(request()->id_resolucion == 2){
415                 $solicitud->solicitud_user_soluciona = request()->id_usuario;
416                 $solicitud->solicitud_fecha_soluciona = now();
417                 $solicitud->solicitud_user_cerrado = request()->id_usuario;
418                 $solicitud->solicitud_fecha_cerrado = now();
419                 $solicitud->solicitud_estado = 4;
420
421                 $auditoria->auditoria_descripcion = 'El usuario '.$user->usu_nombre.' '.$user->usu_apellidoop.' '.$user->usu_apellidoop.' ha marcado como solucionada y cerrada
    +
422                 la solicitud '.$solicitud->solicitud_numero;
423                 $auditoria->auditoria_estado = 7;
424                 $mensaje = "Se Marcó como Solucionada y Cerrada la Solicitud ".$solicitud->solicitud_numero.' exitosamente.';
425             }
426         }

```

*Ilustración 131: Modelo de Función I de actualización de estado del Ticket*  
Fuente: Elaboración Propia

```

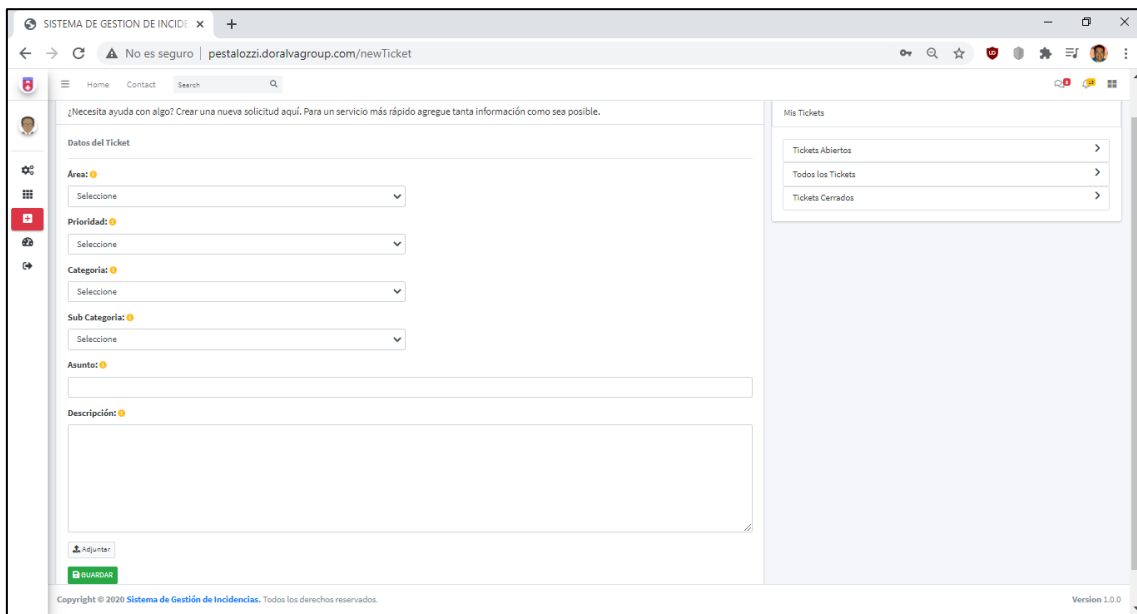
423 }else if(request()->id_resolucion == 3){
424     if(in_array($solicitud->solicitud_estado,array(1,2))){
425         return array("error"=>1,"mensaje"=>"Esta solicitud debe estar en estado Solucionado para continuar.", "tipo"=>"warning", "titulo"=>"¡Alerta!");
426     }else{
427         $solicitud->solicitud_user_cerrado = request()->id_usuario;
428         $solicitud->solicitud_fecha_cerrado = now();
429         $solicitud->solicitud_estado = 4;
430
431         $auditoria->auditoria_descripcion = 'El usuario ' . $user->usu_nombre . ' ' . $user->usu_apellido . ' ' . $user->usu_apellido . ' ha marcado como cerrada la
432         * solicitud ' . $solicitud->solicitud_numero;
433         $auditoria->auditoria_estado = 8;
434         $mensaje = "Se Cerro la Solicitud " . $solicitud->solicitud_numero . " exitosamente.";
435     }
436     $solicitud->save();
437
438     $auditoria->save();
439     return array("error"=>0,"mensaje"=>$mensaje, "tipo"=>"success", "titulo"=>"¡Éxito!");
440 } catch (\Exception $e) {
441     return array("error"=>1,"mensaje"=>"Existe un problema de conexión, inténtelo nuevamente", "tipo"=>"error", "titulo"=>"¡Oops!");
442 }
443 }

```

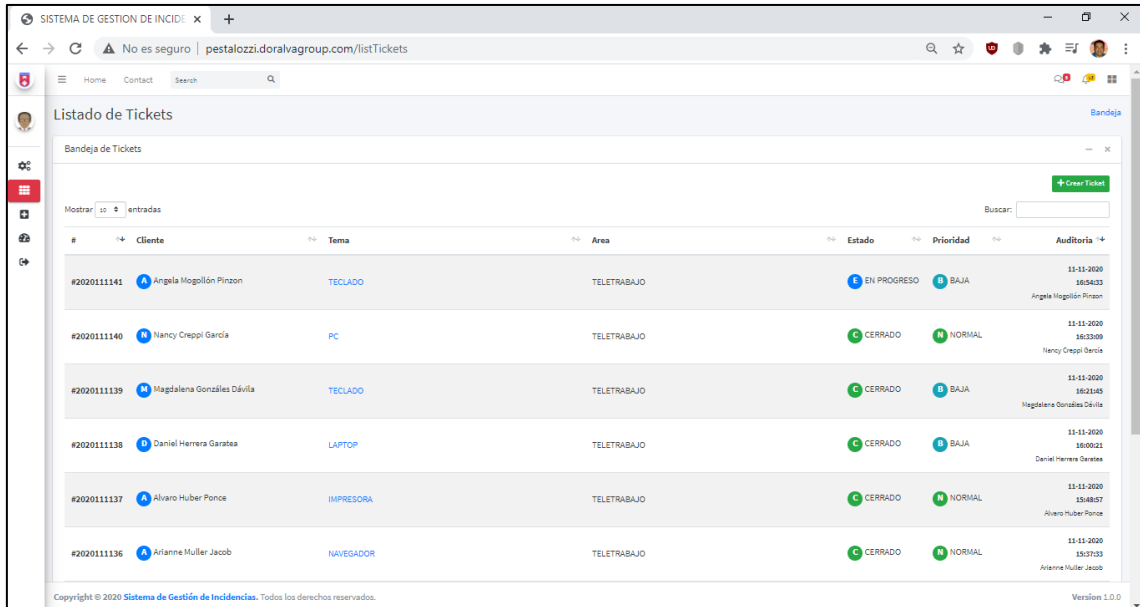
*Ilustración 132: Modelo de Función II de cambio de estado del Ticket  
Fuente: Elaboración Propia*

## 2.6. IMPLEMENTACIÓN DEL SPRINT 2

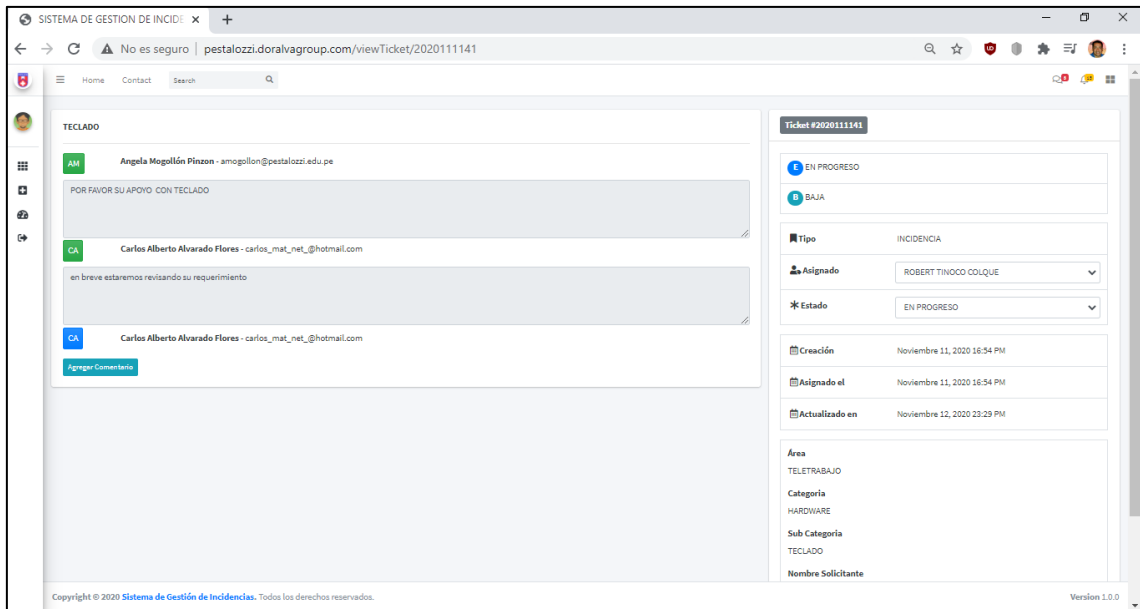
Una vez concluidas las etapas de desarrollo del sistema web (análisis, diseño y código), se pasa a la presentación final de las vistas y funcionalidades del sprint 2.



*Ilustración 133: Interfaz de Creación de Nuevo Ticket  
Fuente: Elaboración Propia*

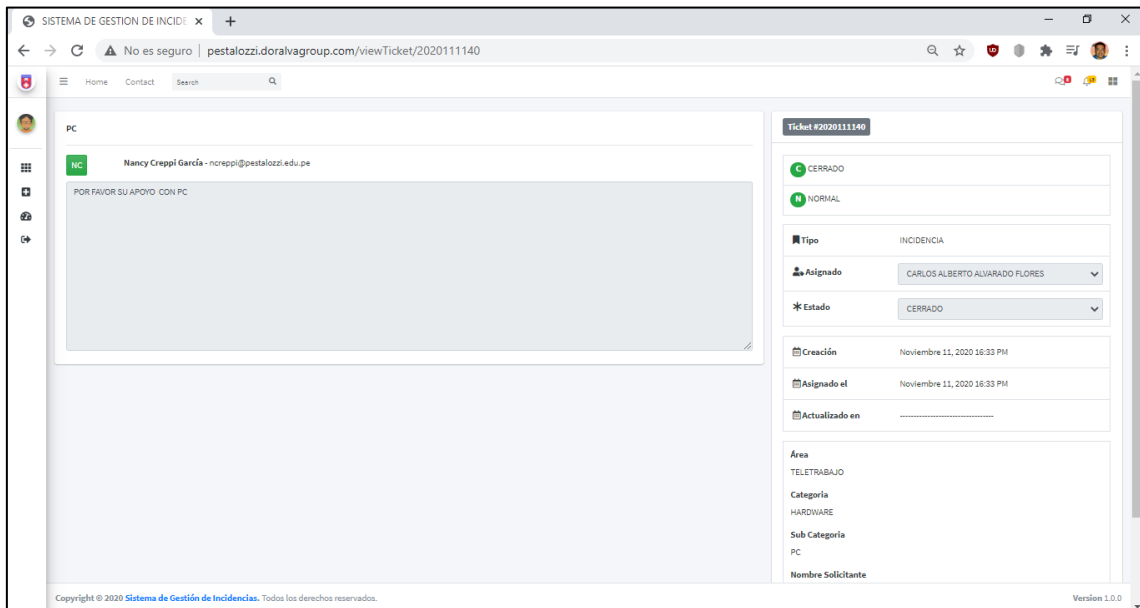


*Ilustración 134: Interfaz de la Bandeja de Tickets  
Fuente: Elaboración Propia*



*Ilustración 135: Interfaz del desarrollo de la atención del Ticket  
Fuente: Elaboración Propia*





*Ilustración 136: Interfaz del proceso de Cerrado de un Ticket  
Fuente: Elaboración Propia*

## 2.7. REVISION DE SPRINT 2

El fin que posee el Sprint Review es el presentar al Product Owner el producto pactado según lo establecido en el cronograma, así como también evaluar el desempeño de los involucrados.

El desarrollo de las historias de usuario es supervisado de acuerdo a la Tabla N° 22 donde se detalle el seguimiento de las horas trabajadas de acuerdo a los días programados para finalizar y cumplir con el objetivo del Sprint 2.

Tareas	Horas Iniciales	Dia 1-3	Dia 4-6	Dia 7-9	Dia 10-12	Dia 13-15	Total de Horas
Registro de Solicitud de Ticket	40	24	14	8	0	0	38
Visualización de Bandeja	40	9	8	24	10	3	34
Seguimiento de Tickets	40	7	7	0	16	16	32

Horas Restantes - Estimado	120	40	40	40
Horas Restantes - Actual	104	38	34	32

*Tabla 28: Resumen del Sprint 2*

Fuente: Elaboración Propia

De acuerdo a la tabla N° 22, se puede observar que se cumplió con el desarrollo del segundo sprint, más aún que las horas cumplidas para cada tarea se realizaron en menos tiempo de acuerdo a las horas iniciales estimadas para su desarrollo.

## 2.8. RETROSPECTIVA DE SPRINT 2

¿Qué salió bien en la presentación del Sprint 2?	¿Qué no salió bien en la presentación del Sprint 2?	¿Qué vamos a implementar en la próxima presentación?
--	---	--

El Sistema permite la creación de los Ticket de atención como también adjuntar imágenes del inconveniente presentado, para que se pueda revisar con mayor precisión la incidencia.	En la gestión de Seguimiento de Ticket, no se visualizaba la actualización que mostraba el otro usuario de soporte al ir actualizando la incidencia.	En la próxima presentación se mostrará mediante gráficos la visualización de los reportes de los indicadores de estudio y incidencias en general
--	--	--

*Tabla 29: Retrospectiva del Sprint 2*

Fuente: Elaboración Propia



## Acta de Reunión de Sprint 2

### Acta de Reunión

**Fecha:** 17 de setiembre ,2020

En la presente reunión se presentaron los prototipados sobre el sprint 2, acerca de los diseños, de vistas, los roles o perfiles de los usuarios. Esto para que puedan tener una idea del funcionamiento de idea para el Sistema Web.

En dicha reunión se describieron ciertos requerimientos adicionales para los prototipos sobre el Diseño N° 2 en adelante y posteriormente para que impacte en el desarrollo. Luego de esto se procede a la codificación teniendo como guía los prototipos presentados, los requerimientos adicionales solicitados, y las herramientas para poder trabajar e iniciar.

Se acordó también durante la reunión, que el administrador / soporte tendrían que visualizar la mayoría de pestañas dentro del panel, para tener un cotejo de como va realizándose el proceso de las incidencias.

Luego de todo lo presentado al administrador de la empresa, accedió a su conformidad e inicio inmediato del desarrollo del Sistema Web.

Finalmente, cabe decir que estarán a la espera del desarrollo del Sprint 2, e inicio de pruebas unitarias con data ingresada por personal del Colegio.

  
Ricarzo Estrada Méndez  
Jefe de Tecnologías de Información



*Ilustración 137: Acta de Reunión de Sprint 2*

### 3. SPRINT 3:

#### 3.1. HISTORIAS DE USUARIOS

HISTORIA DE USUARIO		PRIORIDAD	Estimado
<b>Numero:</b> 13	<b>Usuario:</b> Administrador / Soporte	Alta	2.5 días
<b>Nombre de la Historia:</b> Reporte de Indicador Ratio de Resolución de Incidencias			
<b>Programador Responsable:</b> Carlos Alvarado / Luis Dorador			
<b>Descripción:</b> Luego de haberse generados los Tickets de Incidencias y haber llegado al proceso de solución, esto generará reportes por indicador, este es el caso del indicador Ratio de Resolución de Incidencias donde mediante un gráfico estadístico se visualiza en un rango de 21 días las atenciones realizadas cumpliendo los SLA frente al total de incidencias realizadas.			
<b>Como Probarlo:</b>			
<ul style="list-style-type: none"> <li>- El usuario ingresa al sistema como rol administrador /Soporte.</li> <li>- Selecciona menú Reporte luego en Ratio de Resolución de Incidencias.</li> <li>- Ingresa fechas para obtener el reporte</li> <li>- Visualiza cuadro con datos solicitados</li> </ul>			

*Tabla 30: Historia de Usuario N°13  
Fuente: Elaboración Propio*

HISTORIA DE USUARIO		PRIORIDAD	Estimado
<b>Numero:</b> 14	<b>Usuario:</b> Administrador / Soporte	Alta	2.5 días
<b>Nombre de la Historia:</b> Reporte de Indicador Ratio de Utilización Laboral de Incidencias			
<b>Programador Responsable:</b> Carlos Alvarado / Luis Dorador			

<p><b>Descripción:</b> Luego de haberse generados los Tickets de Incidencias y haber llegado al proceso de cierre, esto generará reportes por indicador, este es el caso del indicador Ratio de Utilización Laboral de Incidencias donde mediante un gráfico estadístico se visualiza en un rango de 21 días las horas Invertidas de Resolución de Incidencia frente a las horas disponibles para atender Incidencias.</p>
<p><b>Como Probarlo:</b></p> <ul style="list-style-type: none"> <li>- El usuario ingresa al sistema como rol administrador /Soporte.</li> <li>- Selecciona menú Reporte luego en Ratio de Utilización Laboral de Incidencias</li> <li>- Ingresas fechas para obtener el reporte</li> <li>- Visualiza cuadro con datos solicitados</li> </ul>

*Tabla 31: Historia de Usuario N°14  
Fuente: Elaboración Propio*

HISTORIA DE USUARIO		PRIORIDAD	Estimado
<b>Numero:</b> 15	<b>Usuario:</b> Administrador / Soporte	Alta	2 días
<b>Nombre de la Historia:</b> Reporte de Incidencias			
<b>Programador Responsable:</b> Carlos Alvarado / Luis Dorador			
<b>Descripción:</b> Luego de haberse generados los Tickets de Incidencias y llegar a los estados de abiertos, en proceso, solución y cerrado en la Bandeja se procede a obtener los reportes de todas las Incidencias.			
<b>Como Probarlo:</b>			
<ul style="list-style-type: none"> <li>- El usuario ingresa al sistema como rol administrador /Soporte.</li> <li>- Selecciona menú Reporte opción Reporte de Incidencias.</li> <li>- Ingresas fechas para obtener el reporte</li> <li>- Visualiza cuadro con datos solicitados</li> </ul>			

*Tabla 32: Historia de Usuario N°15  
Fuente: Elaboración Propio.*

HISTORIA DE USUARIO		PRIORIDAD	Estimado
<b>Numero:</b> 16	<b>Usuario:</b> Administrador / Soporte	Alta	2 días
<b>Nombre de la Historia:</b> Visualización de gráficos de incidencias			
<b>Programador Responsable:</b> Carlos Alvarado / Luis Dorador			
<b>Descripción:</b> Se podrá visualizar los gráficos detallando las categorías de las incidencias			
<b>Como Probarlo:</b>			
<ul style="list-style-type: none"> <li>- El usuario ingresa al sistema como rol administrador /Soporte.</li> <li>- Seleccionar menú Dashboard</li> <li>- Visualiza gráfico de detalles de las incidencias reciente</li> </ul>			

*Tabla 33: Historia de Usuario N°16  
Fuente: Elaboración Propio.*

### 3.2. SPRINT BACKLOG

En el siguiente sprint tiene como historia a realizar el formulario de generación de Reportes donde se visualizarán los detalles de las incidencias.

Sprint	RF	H.U.	Prioridad	T. Estimado	Responsables
Sprint 3	13	Reporte de Indicador Ratio de Resolución de Incidencias	Alta	22 horas	Carlos Alvarado / Luis Dorador
	14	Reporte de Indicador Ratio de Utilización Laboral de Incidencias	Alta	22 horas	Carlos Alvarado / Luis Dorador
	15	Reporte de Incidencias	Alta	14 horas	Carlos Alvarado / Luis Dorador

16	Visualizar gráficas de incidencias	Alta	14 horas	Carlos Alvarado / Luis Dorador
----	------------------------------------	------	----------	--------------------------------

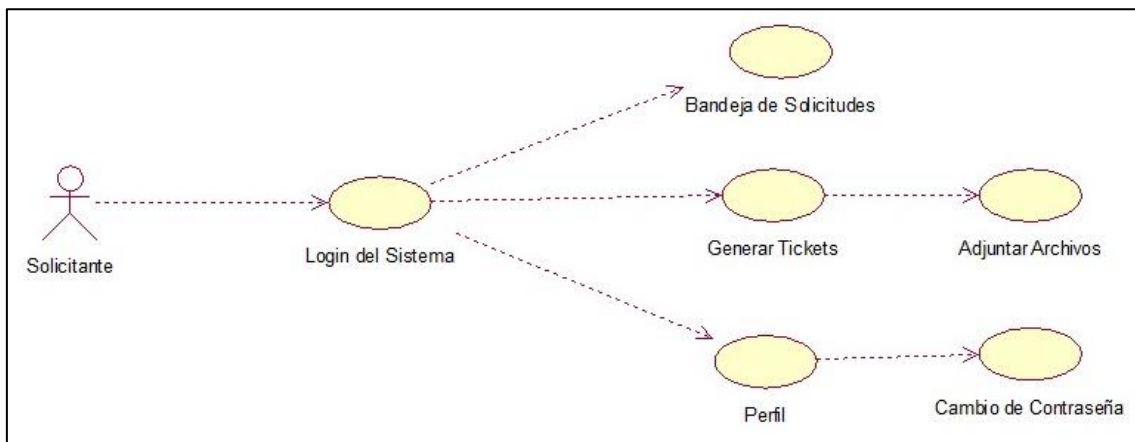
*Tabla 34: Product Backlog Sprint 3  
Fuente: Elaboración Propia*

### 3.3. TRABAJO DE DESARROLLO DE SPRINT

#### 3.3.1. Análisis del Sprint 3

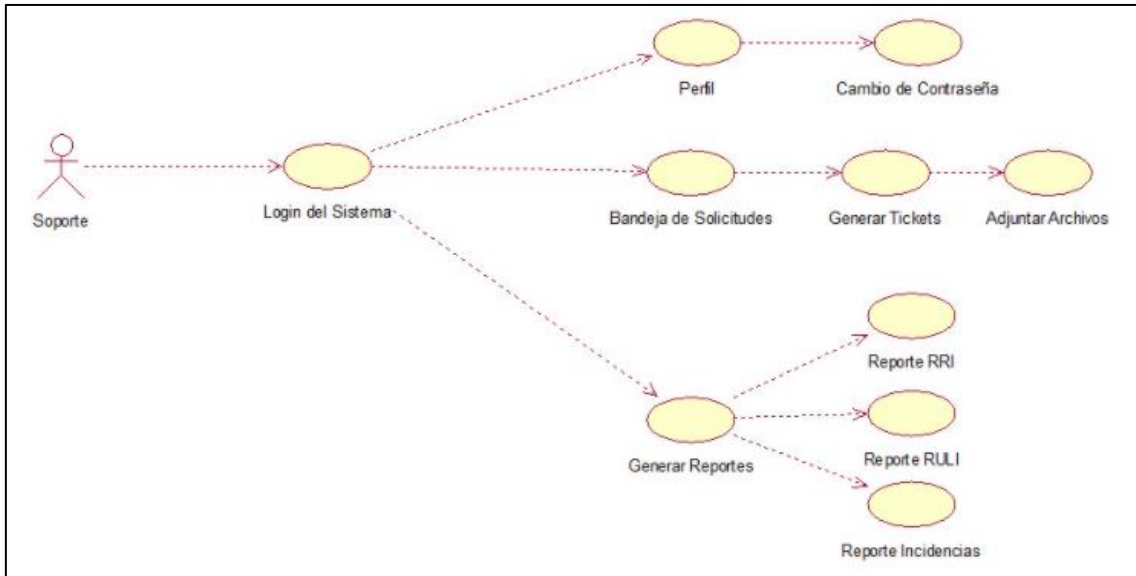
Para el sprint 3 se analiza el funcionamiento del sistema en base al módulo de reportes de indicadores: ratio de resolución de incidencias y ratio de utilización laboral de incidencias y reporte de Incidencias. Teniendo como interpretación lo siguiente:

- El administrador / Soporte genera gráficos donde se visualiza la cantidad de incidentes atendidos durante un cierto tiempo determinado.
- El administrador / Soporte puede generar reportes de incidentes.
- El administrador / Soporte puede descargar la cantidad de incidentes registrados.

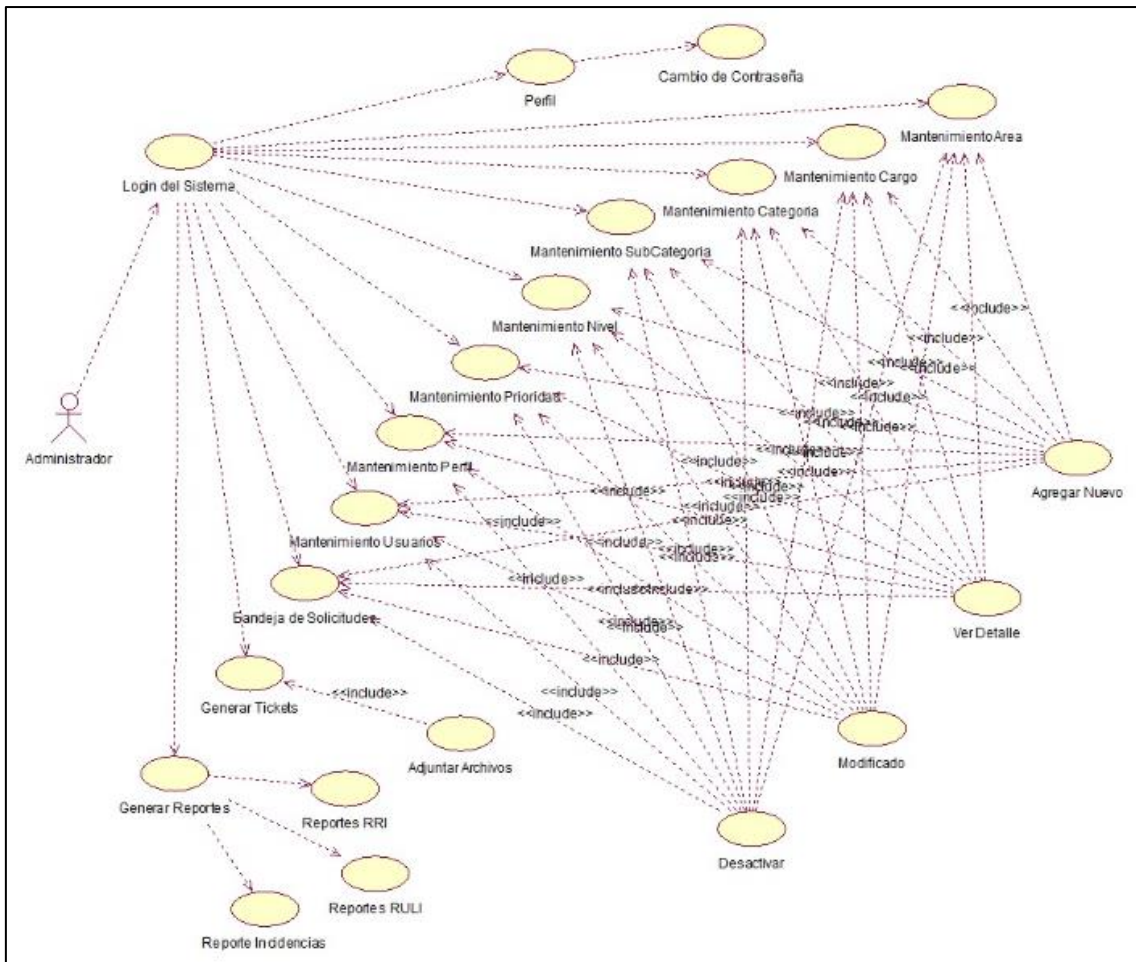


*Ilustración 138: Caso de Uso de Solicitante Sprint 3  
Fuente: Elaboración Propia*





*Ilustración 139: Caso de Uso de Soporte Sprint 3  
Fuente: Elaboración Propia*



*Ilustración 140: Caso de Uso de Administrador Sprint 3  
Fuente: Elaboración Propia*

### 3.3.2. Diseño del Sprint 3

### a. Diagrama Entidad – Relación de Sprint 3

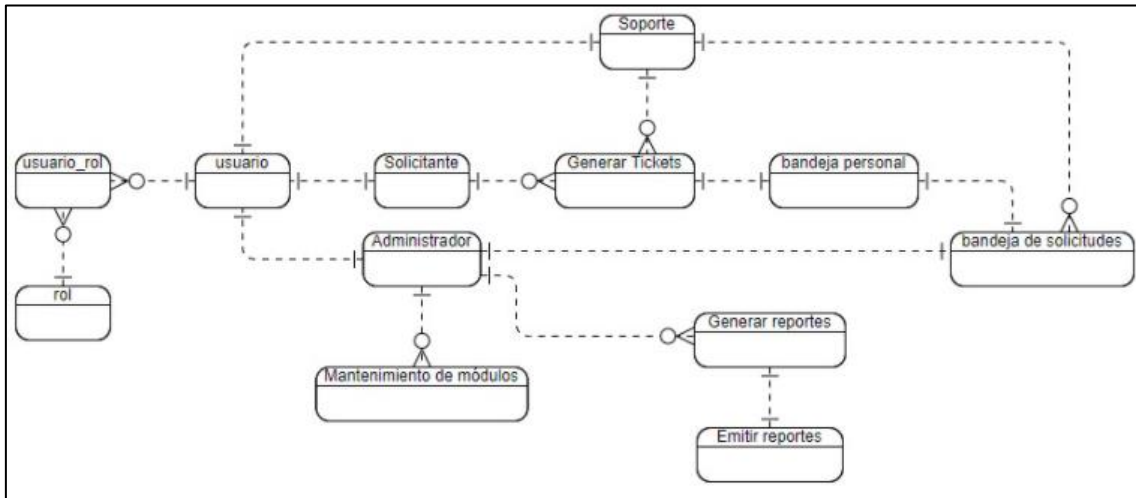


Ilustración 141: Diagrama Entidad – Relación de Sprint 3  
Fuente: Elaboración Propia

### b. Modelo Lógico.

Es un prototipo de una base de datos real, por lo que los usuarios pueden verificar los resultados del diseño. De esta forma, podemos verificar el modelo antes de iniciar la construcción física de la base de datos. Este modelo solo se desarrolla en base a Sprint 3 para satisfacer las necesidades del cliente.

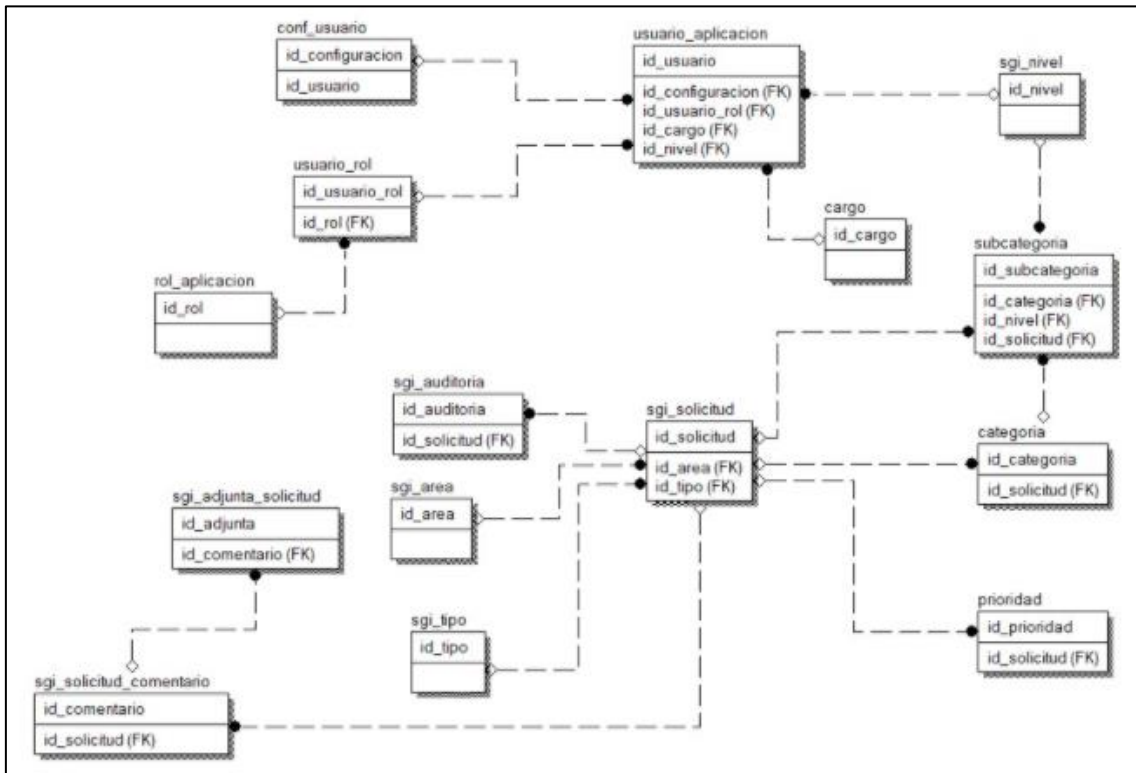


Ilustración 142: Modelo Lógico de Sprint 3

Fuente: Elaboración Propia

### c. Modelo Físico

Detalla las relaciones básicas, la estructura de almacenamiento y los métodos de acceso que se utilizarán para acceder a los datos de manera eficiente. El diseño de esta relación se debe a una comprensión detallada de todas las funciones proporcionadas por el Sprint 3 (ver Figura XX)

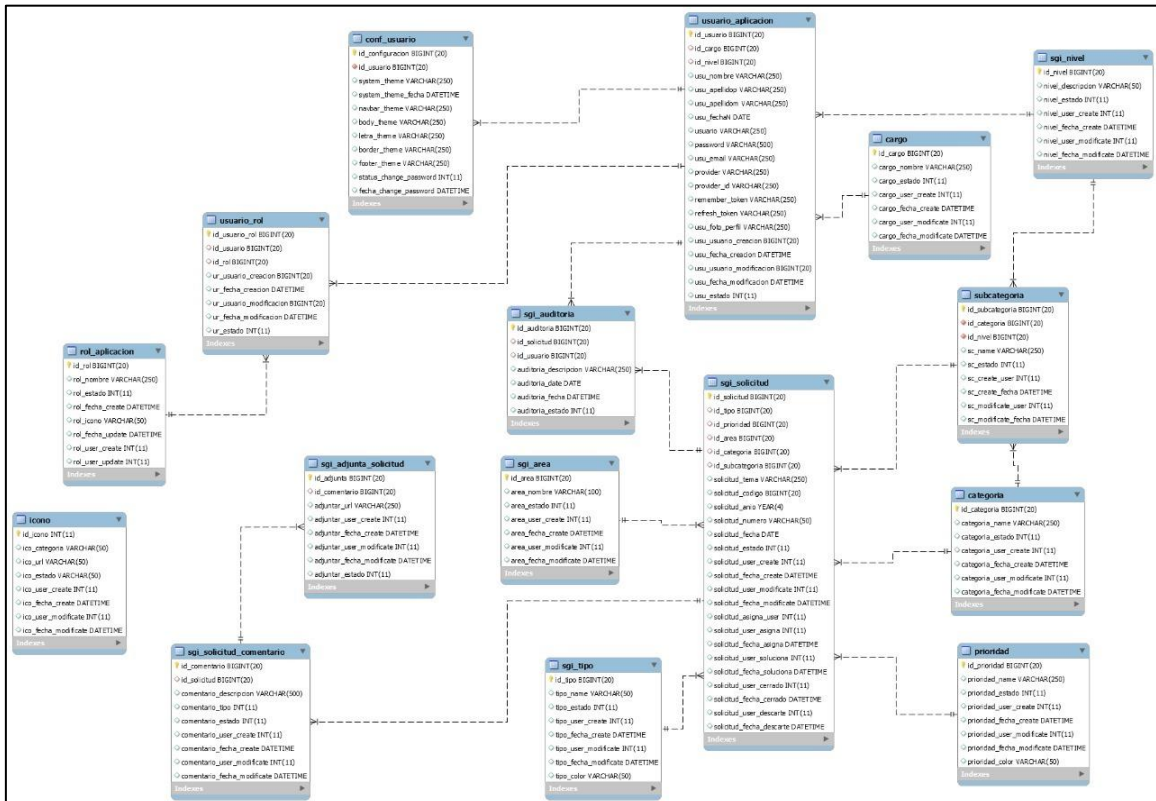


Ilustración 143: Modelo Físico de la BD  
Fuente: Elaboración Propia

## 3.4. PROTOTIPOS DE DISEÑO DEL SPRINT 3

### a. Reporte de Indicador de RRI

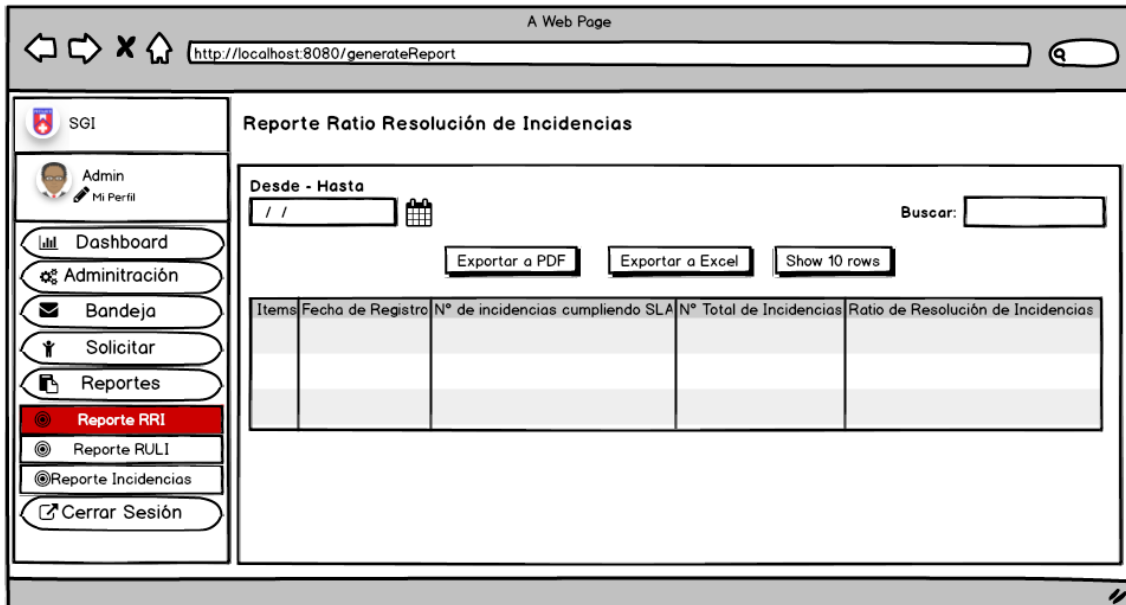


Ilustración 144: Reporte de Indicador Ratio de Resolución de Incidencias  
Fuente: Elaboración Propia

### b. Reporte de Indicador de RULI

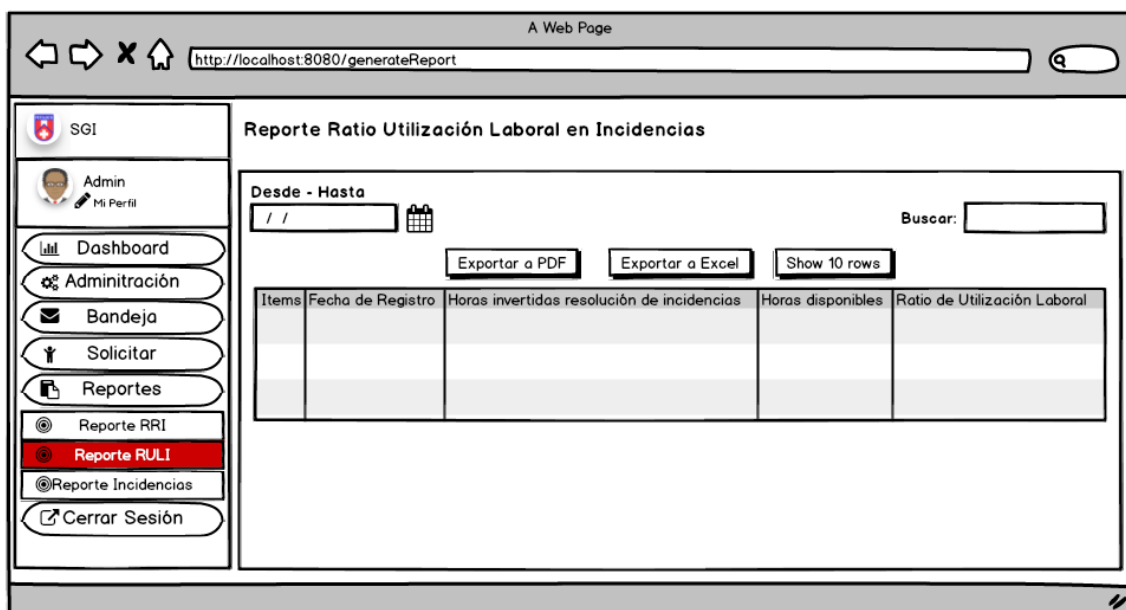


Ilustración 145: Reporte de Indicador Ratio de Utilización Laboral de Incidencias  
Fuente: Elaboración Propia

### c. Reporte de Incidencias

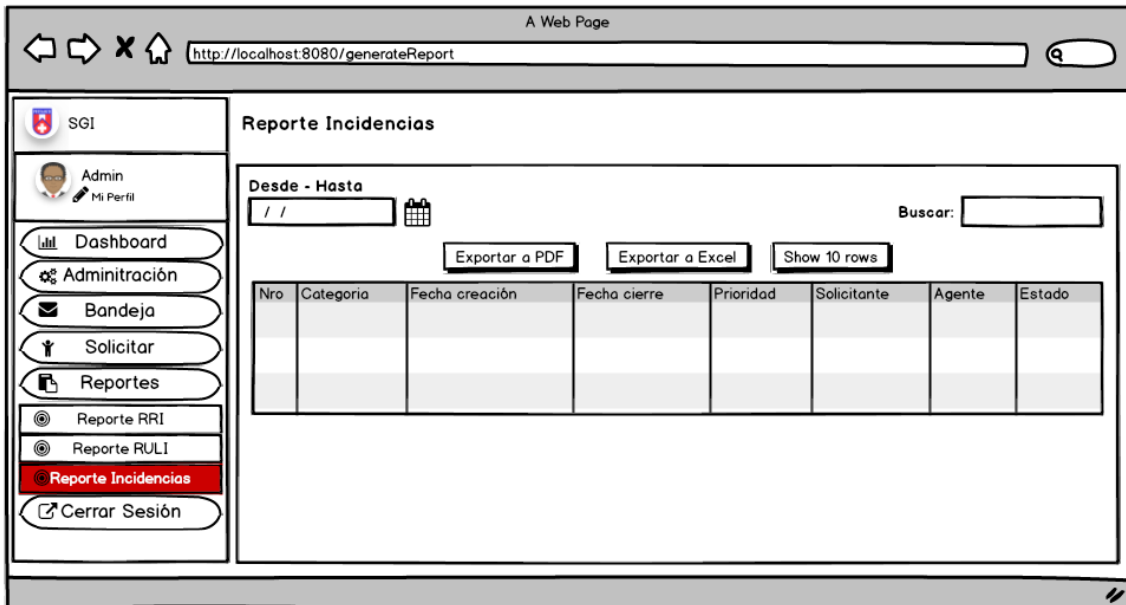


Ilustración 146: Reporte de Incidencias  
Fuente: Elaboración Propia

#### d. Visualizar gráficas de Incidencias

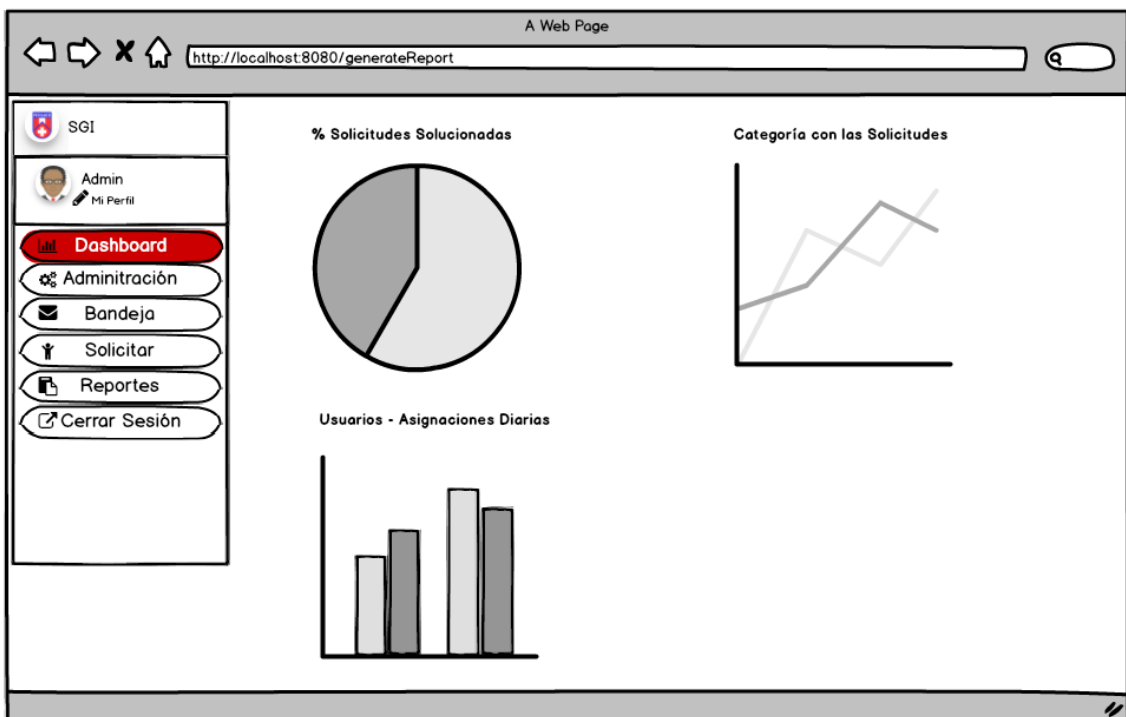


Ilustración 147: Gráficas de incidencias  
Fuente: Elaboración Propia

### 3.5. CODIFICACIÓN DEL SPRINT 3

#### a. Reporte de Indicador de RRI

```

1  |:?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use App\Models\Reporte;
7
8  class ReporteController extends Controller{
9
10     public function __construct(){
11         $this->middleware('auth');
12     }
13
14     public function generateReport($type){
15         if($type == 1){
16             $texto = 'Reporte Ratio Resolución de Incidencias';
17             $subtexto = 'RRI';
18         }else if($type == 2){
19             $texto = 'Reporte Ratio Utilización Laboral en Incidencias';
20             $subtexto = 'RULI';
21         }else if($type == 3){
22             $texto = 'Reporte Incidencias';
23             $subtexto = 'INCIDENCIAS';
24         }
25         return view('reports.reportrri',compact('texto','type','subtexto'));
26     }
27
28     public function dashboard(){
29         $reporte = Reporte::getQueryDashboard();
30         return view('pages.home',compact('reporte'));
31     }
32
33 }
34

```

*Ilustración 148: Controlador de Vista de Reportes  
Fuente: Elaboración Propia*

```

12     public static function getRRI(){
13         $fecha = explode(' - ',request()->fecha);
14         $fecha1 = str_replace('/', '-', $fecha[0]);
15         $fecha2 = str_replace('/', '-', $fecha[1]);
16
17         $query = DB::SELECT("SELECT SEC_TO_TIME(SUM(TIME_TO_SEC(HORAS))) AS HORAS,
18             COUNT(*) AS TOTAL_RESUELTAS,
19             (SELECT
20                 COUNT(*)
21                 FROM sgi_solicitud
22                 WHERE solicitud_fecha = A.solicitud_fecha
23             ) AS TOTAL_SOLICITADAS,
24             (COUNT(*)/(SELECT
25                 COUNT(*)
26                 FROM sgi_solicitud
27                 WHERE solicitud_fecha = A.solicitud_fecha
28             ) ) AS RRI,
29             solicitud_fecha
30         FROM
31             (
32                 SELECT
33                     SEC_TO_TIME(TIMESTAMPDIFF(SECOND, solicitud_fecha_create, solicitud_fecha_soluciona)) HORAS,
34                     solicitud_fecha_create,solicitud_fecha
35                 FROM sgi_solicitud s
36                 WHERE solicitud_estado = 3 OR solicitud_estado = 4
37             ) A
38             WHERE A.solicitud_fecha BETWEEN '$fecha1' AND '$fecha2'
39             GROUP BY A.solicitud_fecha");
40
41         foreach($query as $key => $val){
42             $val->RRI = round(($val->RRI),2, PHP_ROUND_HALF_DOWN);
43             // $val->RRI = 0.01 * (int)($val->RRI*100);
44         }
45
46         return $query;
47     }
48 }

```

*Ilustración 149: Modelo de la Función de Reportes RRI  
Fuente: Elaboración Propia*

```

1 @if(!empty($reporte))
2 <table class="table table-bordered table-striped" id="table1">
3   <thead style="background-color:#3B9FF5">
4     <tr>
5       <th>ITEMS</th>
6       <th>FECHA DE REGISTRO</th>
7       <th>Nº INCIDENCIAS RESUELTAS CUMPLIENDO LOS SLA</th>
8       <th>Nº TOTAL INCIDENCIAS</th>
9       <th>RATIO DE RESOLUCIÓN DE INCIDENCIAS</th>
10    </tr>
11  </thead>
12  <tbody>
13    @foreach($reporte as $key => $value)
14      <tr>
15        <td>{{ $key+1 }}</td>
16        <td>{{ $value->solicitud_fecha }}</td>
17        <td>{{ $value->TOTAL_RESUELTAS }}</td>
18        <td>{{ $value->TOTAL_SOLICITADAS }}</td>
19        <td>{{ $value->RRI }}</td>
20      </tr>
21    @endforeach
22  </tbody>
23 </table>
24 @else
25 <div class="alert alert-danger" role="alert">
26   <h4 class="alert-heading">¡Alerta!</h4>
27   <p>No se encontraron registros para las fechas Seleccionadas.</p>
28 </div>
29 @endif
30
31 <script type="text/javascript">
32   $(document).ready( function () {
33     $('#table1').DataTable({
34       responsive: true,
35       columnDefs: [
36         { responsivePriority: 1, targets: 1 },
37         { responsivePriority: 2, targets: 2 }
38       ],
39       order: [[ 0, 'asc' ]]

```

*Ilustración 150: Vista del Reporte RRI  
Fuente: Elaboración Propia*

## b. Reporte de Indicador de RULI

```

50 public static function getRULI(){
51   $fecha = explode(' - ',request()->fecha);
52   $fecha1 = str_replace('/', '-', $fecha[0]);
53   $fecha2 = str_replace('/', '-', $fecha[1]);
54
55   $query = DB::SELECT("SELECT SEC_TO_TIME(SUM(TIME_TO_SEC(HORAS))) AS HORAS,
56     ((SELECT COUNT(*)
57      FROM usuario_aplicacion u
58      INNER JOIN usuario_rol ur ON ur.id_usuario = u.id_usuario
59      WHERE id_rol = 3
60      AND usu_estado = 1)*7) HORAS_DISPONIBLES,
61      solicitud_fecha
62     FROM
63     (
64      SELECT
65      SEC_TO_TIME(TIMESTAMPDIFF(SECOND, solicitud_fecha_create, solicitud_fecha_solucionada)) HORAS,
66      solicitud_fecha_create,solicitud_fecha
67      FROM sgi_solicitud s
68      WHERE solicitud_estado = 3 OR solicitud_estado = 4
69     ) A
70     WHERE A.solicitud_fecha BETWEEN '$fecha1' AND '$fecha2'
71     GROUP BY A.solicitud_fecha");
72
73   foreach($query as $key => $val){
74     $HORAS = explode(':', $val->HORAS);
75     $val->HORAS = $HORAS[0].'.'.$HORAS[1];
76     // $val->RULI = 0.01 * (int)($val->RULI*100);
77     $val->RULI = round(($val->HORAS/$val->HORAS_DISPONIBLES),2);
78   }
79
80   return $query;
81 }

```

*Ilustración 151: Modelo de Función de Reporte RULI  
Fuente: Elaboración Propia*

```

1  @if(!empty($reporte))
2  <table class="table table-bordered table-striped" id="table1">
3  <thead style="background-color:#389FF5">
4  <tr>
5  <th>ITEMS</th>
6  <th>FECHA DE REGISTRO</th>
7  <th>HORAS INVERTIDAS RESOLUCIÓN DE INCIDENCIAS(H)</th>
8  <th>HORAS DISPONIBLES PARA ATENDER INCIDENCIAS</th>
9  <th>RATIO DE UTILIZACIÓN LABORAL EN INCIDENCIAS</th>
10 </tr>
11 </thead>
12 <tbody>
13 @foreach($reporte as $key => $value)
14 <tr>
15 <td>{{ $key+1 }}</td>
16 <td>{{ $value->solicitud_fecha }}</td>
17 <td>{{ $value->HORAS }}</td>
18 <td>{{ $value->HORAS_DISPONIBLES }}</td>
19 <td>{{ $value->RULI }}</td>
20 </tr>
21 @endforeach
22 </tbody>
23 </table>
24 @else
25 <div class="alert alert-danger" role="alert">
26 <h4 class="alert-heading">¡Alerta!</h4>
27 <p>No se encontraron registros para las fechas Seleccionadas.</p>
28 </div>
29 @endif
30
31 <script type="text/javascript">
32 $(document).ready( function () {
33   $('#table1').dataTable({
34     responsive: true,
35     columnDefs: [
36       { responsivePriority: 1, targets: 1 },
37       { responsivePriority: 2, targets: 2 }
38     ],
39     order: [[ 0, 'asc' ]]

```

*Ilustración 152: Vista del Reporte RULI  
Fuente: Elaboración Propia*

### c. Reporte de Incidencias

```

83 public static function getIncidencias(){
84   $fecha = explode(' - ',request()->fecha);
85   $fecha1 = str_replace('/', '-', $fecha[0]);
86   $fecha2 = str_replace('/', '-', $fecha[1]);
87
88   $ticket = Ticket::select('solicitud_numero','categoria_name','solicitud_fecha_create','solicitud_fecha_solucionada','prioridad_name',
89     'u.usu_nombre as nombre_creador','u.usu_apellido as apellido_creador','u.usu_apellido as apellido_creador',
90     'u2.usu_nombre as nombre_agente','u2.usu_apellido as apellido_agente','u2.usu_apellido as apellido_agente','solicitud_estado')
91     ->join('categoria as c','c.id_categoria','sgi_solicitud.id_categoria')
92     ->join('prioridad as p','p.id_prioridad','sgi_solicitud.id_prioridad')
93     ->join('usuario_aplicacion as u','u.id_usuario','sgi_solicitud.solicitud_user_create')
94     ->join('usuario_aplicacion as u2','u2.id_usuario','sgi_solicitud.solicitud_asigna_user')
95     ->whereBetween('solicitud_fecha', [$fecha1, $fecha2])
96     ->get();
97
98   foreach($ticket as $key => $value){
99     $value->solicitante = $value->nombre_creador.' '.$value->apellido_creador.' '.$value->apellido_creador;
100    $value->agente = $value->nombre_agente.' '.$value->apellido_agente.' '.$value->apellido_agente;
101    if($value->solicitud_estado == 1){
102      $value->estado = 'ABIERTO';
103    }else if($value->solicitud_estado == 2){
104      $value->estado = 'EN PROGRESO';
105    }else if($value->solicitud_estado == 3){
106      $value->estado = 'RESUELTA';
107    }else if($value->solicitud_estado == 4){
108      $value->estado = 'CERRADO';
109    }
110  }
111
112   return $ticket;
113 }

```

*Ilustración 153: Modelo de Función de Reportes de Incidencias  
Fuente: Elaboración Propia*



```

1  @if(!empty($reporte))
2  <table class="table table-bordered table-striped" id="table1">
3    <thead style="background-color:#3B9FF5">
4      <tr>
5        <th>NRO</th>
6        <th>CATEGORIA</th>
7        <th>FECHA CREACIÓN</th>
8        <th>FECHA CIERRE</th>
9        <th>PRIORIDAD</th>
10       <th>SOLICITANTE</th>
11       <th>AGENTE</th>
12       <th>ESTADO</th>
13     </tr>
14   </thead>
15   <tbody>
16     @foreach($reporte as $key => $value)
17     <tr>
18       <td>{{ $value->solicitud_numero }}</td>
19       <td>{{ $value->categoria_name }}</td>
20       <td>{{ $value->solicitud_fecha_create }}</td>
21       <td>{{ $value->solicitud_fecha_soluciona }}</td>
22       <td>{{ $value->prioridad_name }}</td>
23       <td>{{ $value->solicitante }}</td>
24       <td>{{ $value->agente }}</td>
25       <td>{{ $value->estado }}</td>
26     </tr>
27   @endforeach
28 </tbody>
29 </table>
30 @else
31 <div class="alert alert-danger" role="alert">
32   <h4 class="alert-heading">¡Alerta!</h4>
33   <p>No se encontraron registros para las fechas Seleccionadas.</p>
34 </div>
35 @endif

```

*Ilustración 154: Vista del Reporte de Incidencias  
Fuente: Elaboración Propia*

```

28 <div class="col-md-12">
29 <div class="card {{ Auth::user()->getConfig()->body_theme.' '.Auth::user()->getConfig()->border_theme }}">
30 <div class="card-header {{ Auth::user()->getConfig()->border_theme }}">
31 <h3 class="card-title">{{ $subtexto }}</h3>
32 <div class="card-tools">
33 <button type="button" class="btn btn-tool" data-card-widget="collapse" data-toggle="tooltip" title="collapse">
34 <i class="fas fa-minus"></i></button>
35 </div>
36 </div>
37 <div class="card-body">
38 <div class="row">
39 <div class="col-md-3">
40 <div class="form-group">
41 <label>DESDE - HASTA:</label>
42 <div class="input-group">
43 <div class="input-group-prepend">
44 <span class="input-group-text">
45 <i class="far fa-calendar-alt"></i>
46 </span>
47 </div>
48 <input type="text" class="form-control float-right" id="reservation">
49 <input type="hidden" name="type" id="type" value="{{ $type }}">
50 </div>
51 </div>
52 </div>
53 </div>
54 <div class="row" id="divContentRRI" hidden">
55 <div class="col-md-12 text-center">
56 <div class="preloader"><br><br><br><br><strong style="color: #adadad!important;font-size:13px;"> OBTENIENDO
+ DATOS</strong></div>
57 </div>
58 </div>
59 </div>
60 </div>

```

*Ilustración 155: Vista de la generación y búsqueda de reportes  
Fuente: Elaboración Propia*

#### **d. Visualizar gráficas de Incidencias**

```

22 <div class="container-fluid">
23   <div class="main">
24     <div class="row sparkboxes mt-4">
25       <div class="col-md-3">
26         <div class="box box1">
27           <div class="details">
28             <h3></h3>
29             <h4 id="title_spark0"></h4>
30           </div>
31           <div id="spark0"></div>
32         </div>
33       </div>
34       <div class="col-md-3">
35         <div class="box box2">
36           <div class="details">
37             <h3></h3>
38             <h4 id="title_spark1"></h4>
39           </div>
40           <div id="spark1"></div>
41         </div>
42       </div>
43       <div class="col-md-3">
44         <div class="box box3">
45           <div class="details">
46             <h3></h3>
47             <h4 id="title_spark2"></h4>
48           </div>
49           <div id="spark2"></div>
50         </div>
51       </div>
52       <div class="col-md-3">
53         <div class="box box4">
54           <div class="details">
55             <h3></h3>
56             <h4 id="title_spark3"></h4>
57           </div>
58           <div id="spark3"></div>
59         </div>
60       </div>
61     </div>
62   </div>
63 </div>

```

*Ilustración 156: Vista de Dashboards  
Fuente: Elaboración Propia*

```

28 public function dashboard(){
29     $reporte = Reporte::getQueryDashboard();
30     return view('pages.home',compact('reporte'));
31 }

```

*Ilustración 157: Controlador de Dashboards  
Fuente: Elaboración Propia*

```

115 public static function getQueryDashboard(){
116     $reporte = new Ticket();
117
118     $array_meses = array();
119     $array_porcentaje = array();
120     $array_fecha = array();
121     $mes_actual = date("m");
122
123
124     $start = Carbon::now()->startOfMonth()->format('d-m-Y');
125     $dia = date("d-m-Y",strtotime($start."- 3 month"));
126
127     $statement = DB::statement("SET lc_time_names = 'es_ES'");
128     $ultimos_meses = Ticket::select(DB::raw('UPPER(MONTHname(solicitud_fecha)) as mes'))
129         ->where('solicitud_fecha_create','>=', $dia)
130         ->groupby(DB::raw('YEAR(solicitud_fecha_create)'))
131         ->groupby(DB::raw('MONTH(solicitud_fecha_create)'))
132         ->orderBy('mes','desc')
133         ->get();
134
135     foreach($ultimos_meses as $key => $value){
136         if(!in_array($value->mes, $array_meses)){
137             array_push($array_meses,$value->mes);
138         }
139     }
140     $reporte->ultimos_meses = $array_meses;
141
142     $ultimos_porcentaje = DB::select('SELECT *,total,
143         (select COUNT(*)
144         from sgi_solicitud
145         where year(solicitud_fecha_create) = YEAR(A.MAX)
146         AND month(solicitud_fecha_create) = month(A.MAX)
147         AND (solicitud_estado = 3 OR solicitud_estado = 4)
148         group by MonthName(solicitud_fecha_create)) AS contador
149         FROM
150         (select COUNT(*) AS total,MAX(solicitud_fecha_create) AS MAX,
151         LAST_DAY(MAX(solicitud_fecha_create)) + interval 1 DAY - INTERVAL 1 MONTH AS fecha
152         from sgi_solicitud
153         where year(solicitud_fecha_create) = year(curdate())

```

*Ilustración 158: Modelo I dashboard*

Fuente: Elaboración Propia

```
157         order by max asc');
158
159     foreach($ultimos_porc as $key => $value){
160         $value->porc = (100 * $value->contador)/$value->total;
161         array_push($array_porc, (int)$value->porc);
162     }
163     $reporte->ultimos_porc = $array_porc;
164
165     $fechas = Ticket::select('solicitud_fecha')
166         ->groupBy('solicitud_fecha')
167         ->orderBy('solicitud_fecha','desc')
168         ->distinct('solicitud_fecha')
169         ->limit(6)
170         ->get();
171
172     $fechas = $fechas->sortBy('solicitud_fecha');
173     $fecha2 = $fechas[0]->solicitud_fecha;
174     $fecha1 = $fechas[count($fechas)-1]->solicitud_fecha;
175     $array_fechas = array();
176
177     foreach($fechas as $key => $value){
178         array_push($array_fecha,date("d/m/Y", strtotime($value->solicitud_fecha)));
179         array_push($array_fechas,$value->solicitud_fecha);
180     }
181     $reporte->fechas = $array_fecha;
182     $reporte->fecha_last = $array_fechas;
183     $reporte->fecha1 = $fecha1;
184     $reporte->fecha2 = $fecha2;
185
186
187     $categorias = Ticket::select('categoria_name')
188         ->join('categoria as c','c.id_categoria','sgi_solicitud.id_categoria')
189         ->where('solicitud_fecha','>=',$fecha1)
190         ->where('solicitud_fecha','<=',$fecha2)
191         ->orderBy('c.id_categoria','asc')
192         ->distinct('categoria_name')
193         ->get();
194
```

Ilustración 159: Modelo II dashboard  
Fuente: Elaboración Propia

```
196     $query = '';
197     foreach($reporte->fecha_last as $key => $value){
198         $query .= " SUM(CASE WHEN b.solicitud_fecha = '$value'
199             THEN (SELECT COUNT(*) AS conta FROM sgi_solicitud
200                 WHERE id_categoria = b.id_categoria
201                 AND solicitud_fecha = '$value' ) ELSE 0
202             END) AS '$value',";
203     }
204     $query = substr($query, 0, -1);
205
206
207     $frecuentes = (DB::select("SELECT
208         $query
209         from
210             (SELECT *,
211              (
212                  SELECT COUNT(*)AS conta FROM sgi_solicitud
213                  WHERE id_categoria = A.id_categoria
214                  AND solicitud_fecha = A.solicitud_fecha
215                  ) AS contador
216              from
217                  (SELECT DISTINCT categoria_name, solicitud_fecha,s.id_categoria
218                   FROM sgi_solicitud s
219                   INNER JOIN sgi_area a ON a.id_area = s.id_area
220                   INNER JOIN categoria c ON c.id_categoria = s.id_categoria
221                   WHERE solicitud_fecha >= '$fecha1'
222                   AND solicitud_fecha <= '$fecha2'
223                   GROUP by solicitud_fecha_create,id_solicitud
224                  )AS A
225                  ORDER BY solicitud_fecha,id_categoria) AS b
226              GROUP BY id_categoria
227              ORDER BY id_categoria asc"));
228
229
230     $array_serie = array();
231
232     foreach($frecuentes as $value){
233         $array_serie2 = array();
234         foreach($reporte->fecha_last as $key => $val) {
```

**Ilustración 160: Modelo III Dashboard**  
**Fuente: Elaboración Propia**

```

241 foreach($array_serie3 as $key => $value){
242     array_push($array_seriefinal,array('name'=>$categorias[$key]->categoria_name,'data'=>$value));
243 }
244 $reporte->categorias = $array_seriefinal;
245
246 $soporte = Ticket::select(DB::raw('CONCAT(usu_apellidoop,\ ' \',usu_apellidoom) as usuario'))
247     ->join('usuario_aplicacion as u','u.id_usuario','sgi_solicitud.solicitud_asigna_user')
248     ->where('solicitud_fecha','>=',$fecha1)
249     ->where('solicitud_fecha','<=',$fecha2)
250     ->orderBy('usuario','asc')
251     ->distinct('usuario')
252     ->get();
253
254 $query_sop = '';
255 foreach($reporte->fecha_last as $key => $value){
256     $query_sop .= " SUM(CASE WHEN b.solicitud_fecha = '$value'
257         THEN (SELECT COUNT(*) AS conta FROM sgi_solicitud
258             WHERE solicitud_asigna_user = b.solicitud_asigna_user
259             AND solicitud_fecha = '$value' ) ELSE 0
260         END) AS '$value',";
261 }
262 $query_sop = substr($query_sop, 0, -1);
263
264 $frecuentes_sop = (DB::select("SELECT
265     $query_sop
266     from
267     (SELECT *,
268     (
269         SELECT COUNT(*)AS conta FROM sgi_solicitud
270         WHERE solicitud_asigna_user = A.solicitud_asigna_user
271         AND solicitud_fecha = A.solicitud_fecha
272     ) AS contador
273     from
274     (SELECT DISTINCT solicitud_asigna_user, solicitud_fecha
275     FROM sgi_solicitud s
276     WHERE solicitud_fecha >= '$fecha1'
277     AND solicitud_fecha <= '$fecha2'
278     GROUP by solicitud_fecha_create,id_solicitud
279     )AS A

```

**Ilustración 161: Modelo IV Dashboard**  
**Fuente: Elaboración Propia**

```

286 foreach($reporte->fecha_last as $key => $value){
287     $query_sop_sol .= " SUM(CASE WHEN b.solicitud_fecha = '$value'
288         THEN (SELECT COUNT(*) AS conta FROM sgi_solicitud
289             WHERE solicitud_user_soluciona = b.solicitud_user_soluciona
290             AND solicitud_fecha = '$value' ) ELSE 0
291         END) AS '$value',";
292 }
293 $query_sop_sol = substr($query_sop_sol, 0, -1);
294
295 $frecuentes_sop_sol = (DB::select("SELECT
296     $query_sop_sol
297     from
298     (SELECT *,
299     (
300         SELECT COUNT(*)AS conta FROM sgi_solicitud
301         WHERE solicitud_user_soluciona = A.solicitud_user_soluciona
302         AND solicitud_fecha = A.solicitud_fecha
303     ) AS contador
304     from
305     (SELECT DISTINCT solicitud_user_soluciona, solicitud_fecha
306     FROM sgi_solicitud s
307     WHERE solicitud_fecha >= '$fecha1'
308     AND solicitud_fecha <= '$fecha2'
309     AND (solicitud_estado = 3 or solicitud_estado = 4)
310     GROUP by solicitud_fecha_create,id_solicitud
311     )AS A
312     ORDER BY solicitud_fecha,solicitud_user_soluciona) AS b
313     GROUP BY solicitud_user_soluciona
314     ORDER BY solicitud_user_soluciona asc"));
315
316 foreach($frecuentes_sop as $value){
317     $array_sop = array();
318     foreach($reporte->fecha_last as $key => $val) {
319         array_push($array_sop,$value->$val);
320     }
321     $array_usu_sop[] = $array_sop;
322 }
323
324 foreach($frecuentes_sop_sol as $value){

```

*Ilustración 162: Modelo V Dashboard  
Fuente: Elaboración Propia*

```
316     foreach($frecuentes_sop as $value){
317         $array_sop = array();
318         foreach($reporte->fecha_last as $key => $val) {
319             array_push($array_sop,$value->$val);
320         }
321         $array_usu_sop[] = $array_sop;
322     }
323
324     foreach($frecuentes_sop_sol as $value){
325         $array_sop_sol = array();
326         foreach($reporte->fecha_last as $key => $val) {
327             array_push($array_sop_sol,$value->$val);
328         }
329         $array_usu_sop_sol[] = $array_sop_sol;
330     }
331
332     $array_soportefinal = [];
333     foreach($array_usu_sop as $key => $value){
334         array_push($array_soportefinal,array('name'=>$soporte[$key]->usuario,'data'=>$value));
335     }
336     $reporte->soporte = $array_soportefinal;
337
338     $array_soportefinal_sol = [];
339     foreach($array_usu_sop_sol as $key => $value){
340         array_push($array_soportefinal_sol,array('name'=>$soporte[$key]->usuario,'data'=>$value));
341     }
342     $reporte->soporte_solucion = $array_soportefinal_sol;
343
344     return $reporte;
345 }
```

*Ilustración 163: Modelo VI dashboard  
Fuente: Elaboración Propia*

```
1 function dashboardfunction(json){
2   var json_meses = json.ultimos_meses;
3   var json_porc = json.ultimos_porc;
4   var fechas = json.fechas;
5   var categorias = json.categorias
6   var soporte = json.soporte
7   var solucion = json.soporte_solucion
8
9   window.Apex = {
10    chart: {
11      foreColor: '#ccc',
12      toolbar: {
13        show: false
14      },
15    },
16    stroke: {
17      width: 3
18    },
19    dataLabels: {
20      enabled: false
21    },
22    tooltip: {
23      theme: 'dark'
24    },
25    grid: {
26      borderColor: "#535A6C",
27      xaxis: {
28        lines: {
29          show: true
30        }
31      }
32    }
33  };
34
35  $.each( categorias, function( key, value ) {
36    $("#title_spark"+key).html(value.name)
37    var count = {
38      chart: {
39        id: 'spark1',
```

*Ilustración 164: Script I construcción Dashboards  
Fuente: Elaboración Propia*

```

43     height: 80,
44     sparkline: {
45       enabled: true
46     },
47     dropShadow: {
48       enabled: true,
49       top: 1,
50       left: 1,
51       blur: 2,
52       opacity: 0.2,
53     }
54   },
55   series: [{
56     data: value.data
57   }],
58   stroke: {
59     curve: 'smooth'
60   },
61   markers: {
62     size: 0
63   },
64   grid: {
65     padding: {
66       top: 20,
67       bottom: 10,
68       left: 110
69     }
70   },
71   colors: ['#fff'],
72   tooltip: {
73     x: {
74       show: false
75     },
76     y: {
77       title: {
78         formatter: function formatter(val) {
79           return '';
80         }
81     }
82   }

```

*Ilustración 165: Script II construcción Dashboards  
Fuente: Elaboración Propia*

```

90 var optionsLine = {
91   chart: {
92     height: 328,
93     type: 'line',
94     zoom: {
95       enabled: false
96     },
97     dropShadow: {
98       enabled: true,
99       top: 3,
100      left: 2,
101      blur: 4,
102      opacity: 1,
103    }
104  },
105  stroke: {
106    curve: 'smooth',
107    width: 2
108  },
109  //colors: ['#3F51B5', '#2196F3'],
110  series: categorias,
111  title: {
112    text: 'CATEGORIAS CON MÁS SOLICITUDES',
113    align: 'left',
114    offsetY: 25,
115    offsetX: 20
116  },
117  subtitle: {
118    text: 'ESTADISTICAS',
119    offsetY: 55,
120    offsetX: 20
121  },
122  markers: {
123    size: 0,
124    strokeWidth: 0,
125    hover: {
126      size: 9
127    }

```

*Ilustración 166: Script III construcción Dashboards  
Fuente: Elaboración Propia*

```

147
148 var chartLine = new ApexCharts(document.querySelector('#line-adwords'), optionsLine);
149 chartLine.render();
150
151 var optionsCircle4 = {
152   chart: {
153     type: 'radialBar',
154     height: 310,
155     width: 340,
156   },
157   plotOptions: {
158     radialBar: {
159       size: undefined,
160       inverseOrder: true,
161       hollow: {
162         margin: 5,
163         size: '40%',
164         background: 'transparent',
165       },
166     },
167     track: {
168       show: false,
169     },
170     startAngle: -180,
171     endAngle: 180
172   },
173 },
174 stroke: {
175   lineCap: 'round'
176 },
177 title: {
178   text: '% SOLICITUDES SOLUCIONADAS',
179   align: 'left',
180   offsetY: 25,
181   offsetX: 20
182 },
183 series: json_porc,
184 labels: json_meses,

```

*Ilustración 167: Script IV construcción Dashboards  
Fuente: Elaboración Propia*

```

190 var chartCircle4 = new ApexCharts(document.querySelector('#radialBarBottom'), optionsCircle4);
191 chartCircle4.render();
192
193 var optionsBar = {
194   chart: {
195     height: 380,
196     type: 'bar',
197     stacked: true,
198   },
199   plotOptions: {
200     bar: {
201       columnwidth: '30%',
202       horizontal: false,
203     },
204   },
205   title: {
206     text: 'USUARIOS - Nº ASIGNACIONES DIARIAS',
207     align: 'left',
208     offsetY: 25,
209     offsetX: 20
210   },
211   series: soporte,
212   xaxis: {
213     categories: fechas,
214   },
215   fill: {
216     opacity: 1
217   },
218 }
219
220 var chartBar = new ApexCharts(
221   document.querySelector("#barchart"),
222   optionsBar
223 );
224
225 chartBar.render();

```

*Ilustración 168: Script V construcción Dashboards  
Fuente: Elaboración Propia*



```

237 var optionsArea = {
238   chart: {
239     height: 380,
240     type: 'area',
241     stacked: false,
242   },
243   stroke: {
244     curve: 'straight'
245   },
246   title: {
247     text: 'USUARIOS - Nº SOLICITUDES SOLUCIONADAS DIARIAS',
248     align: 'left',
249     offsetY: 25,
250     offsetX: 20
251   },
252   series: solucion,
253   xaxis: {
254     categories: fechas,
255   },
256   tooltip: {
257     followCursor: true
258   },
259   fill: {
260     opacity: 1,
261   },
262 }
263 }
264
265 var chartArea = new ApexCharts(
266   document.querySelector("#areachart"),
267   optionsArea
268 );
269
270 chartArea.render();
271
272 /* fusionexport integrations START */
273 (() => {
274   const btn = document.getElementById('fusionexport-btn')

```

*Ilustración 169: Script VI construcción Dashboards  
Fuente: Elaboración Propia*

### 3.6. IMPLEMENTACIÓN DEL SPRINT 3

Una vez concluidas las etapas de desarrollo del sistema web (análisis, diseño y código), se pasa a la presentación final de las vistas y funcionalidades del sprint 3.

ITEMS	FECHA DE REGISTRO	Nº INCIDENCIAS RESUELTAS CUMPLIENDO LOS SLA	Nº TOTAL INCIDENCIAS	RATIO DE RESOLUCIÓN DE INCIDENCIAS
1	2020-10-14	30	32	0.94
2	2020-10-15	27	29	0.93
3	2020-10-16	32	33	0.97
4	2020-10-19	24	24	1
5	2020-10-20	20	21	0.95
6	2020-10-21	36	36	1
7	2020-10-22	37	38	0.97
8	2020-10-23	21	22	0.95
9	2020-10-26	18	19	0.95

*Ilustración 170: Interfaz de Ratio de Resolución de Incidencias  
Fuente: Elaboración Propia*

Reporte Ratio Utilización Laboral en Incidencias

DESDE - HASTA: 2020/10/14 - 2020/11/11

Exportar a PDF Exportar a EXCEL Show 10 rows

Buscar:

ITEMS	FECHA DE REGISTRO	HORAS INVERTIDAS RESOLUCIÓN DE INCIDENCIAS(H)	HORAS DISPONIBLES PARA ATENDER INCIDENCIAS	RATIO DE UTILIZACIÓN LABORAL EN INCIDENCIAS
1	2020-10-14	18.40	21	0.88
2	2020-10-15	17.34	21	0.83
3	2020-10-16	18.36	21	0.87
4	2020-10-19	18.24	21	0.87
5	2020-10-20	15.36	21	0.73
6	2020-10-21	19.06	21	0.91
7	2020-10-22	18.38	21	0.88
8	2020-10-23	17.06	21	0.81
9	2020-10-26	16.03	21	0.76
10	2020-10-27	18.15	21	0.86

Copyright © 2020 Sistema de Gestión de Incidencias. Todos los derechos reservados. Versión 1.0.0

Ilustración 171: Interfaz de Ratio de Utilización Laboral de Incidencias  
Fuente: Elaboración Propia

Reporte Incidencias

DESDE - HASTA: 2020/10/14 - 2020/11/11

Exportar a PDF Exportar a EXCEL Show 10 rows

Buscar:

ID	CATEGORIA	FECHA CREACIÓN	FECHA CIERRE	PRIORIDAD	SOLICITANTE	AGENTE	ESTADO
2020111141	HARDWARE	2020-11-11 16:54:33	2020-11-11 17:10:33	BAJA	Angela Magalón Pinzon	Robert Tinoco Colque	EN PROGRESO
2020111140	HARDWARE	2020-11-11 16:33:09	2020-11-11 17:04:09	NORMAL	Nancy Creppi Garcia	Carlos Alberto Alvarado Flores	CERRADO
2020111139	HARDWARE	2020-11-11 16:21:45	2020-11-11 16:56:45	BAJA	Magdalena González Dávila	Jenny Mendoza Contreras	CERRADO
2020111138	HARDWARE	2020-11-11 16:09:21	2020-11-11 16:33:21	BAJA	Daniel Herrera Garateca	Robert Tinoco Colque	CERRADO
2020111137	HARDWARE	2020-11-11 15:48:57	2020-11-11 16:18:57	NORMAL	Alvaro Huber Ponce	Carlos Alberto Alvarado Flores	CERRADO
2020111136	SOFTWARE	2020-11-11 15:37:31	2020-11-11 16:09:33	NORMAL	Arienne Muller Jacob	Jenny Mendoza Contreras	CERRADO
2020111135	SOFTWARE	2020-11-11 15:26:09	2020-11-11 15:57:09	BAJA	Urs Güter Steiner	Robert Tinoco Colque	CERRADO
2020111134	SOFTWARE	2020-11-11 15:14:45	2020-11-11 15:44:45	BAJA	Marciano Rivera Silvestre	Carlos Alberto Alvarado Flores	CERRADO
2020111133	INFRAESTRUCTURA	2020-11-11 15:03:21	2020-11-11 15:27:21	NORMAL	Milagros Medina Figueroa	Jenny Mendoza Contreras	CERRADO
2020111132	SOFTWARE	2020-11-11 14:52:07	2020-11-11 15:09:07	BAJA	Arienne Muller Jacob	Robert Tinoco Colque	CERRADO

Copyright © 2020 Sistema de Gestión de Incidencias. Todos los derechos reservados. Versión 1.0.0

Ilustración 172: Interfaz de Reporte de Incidencias  
Fuente: Elaboración Propia



Ilustración 173: Interfaz de Visualización de gráficas de incidencias  
Fuente: Elaboración Propia

### 3.7. REVISION DE SPRINT 3

El desarrollo de las historias de usuario es supervisado de acuerdo a la Tabla N°29 donde se detalle el seguimiento de las horas trabajadas de acuerdo a los días programados para finalizar y cumplir con el objetivo del Sprint 3.

Tareas	Horas Iniciales	Dia 1-3	Dia 4-6	Dia 7-9	Total de Horas
Reporte de Indicador Ratio de Resolución de Incidencias	22	21	0	0	21
Reporte de Indicador Ratio de Utilización Laboral de Incidencias	22	0	21	0	21
Reporte de Incidencias	14	0	4	9	13
Visualización de Gráficos	14	0	0	15	15
Horas Restantes - Estimado	72	22	22	14	14
Horas Restantes - Actual	70	21	21	13	15

Tabla 35: Resumen del Sprint 3

*Fuente: Elaboración Propia*

De acuerdo con la tabla N° 29, se puede observar que el desarrollo del tercer sprint se ha cumplido o incluso más, por lo que, según el tiempo de desarrollo inicial estimado, el tiempo de finalización de cada tarea se puede completar en menos tiempo.

### **3.8. RETROSPECTIVA DE SPRINT 3**

¿Qué salió bien en la presentación del Sprint 3?	¿Qué no salió bien en la presentación del Sprint 3?	¿Qué vamos a implementar en la próxima presentación?
--	---	--

El Sistema permite generar los gráficos mediante charts señalando las fechas y su cantidad de incidencias atendidas y el tiempo llevado por cada atención. Por otro lado, cada tipo de usuario; ya sea admin, soporte puede ver la cantidad de incidencias registradas	Los reportes de visualización no contenían los rangos escogidos en los gráficos; ya que el administrador dependía saber cómo iba evolucionando su empresa en los últimos días.	Después de finalizar el desarrollo de software con el método SCRUM, Aparece en forma de GO LIVE para la institución pone en práctica todo lo desarrollado y ve la evolución de su proceso.
--	--	--

*Tabla 36: Retrospectiva del Sprint 3  
Fuente: Elaboración Propia*

## Acta de Reunión de Sprint 3

### Acta de Reunión

**Fecha:** 06 de octubre ,2020

En la presente reunión se presentaron los prototipados sobre el sprint 3, acerca de los diseños, de vistas, los roles o perfiles de los usuarios. Esto para que puedan tener una idea del funcionamiento de idea para el Sistema Web.

En dicha reunión se describieron ciertos requerimientos adicionales para los prototipos sobre el Diseño N° 2 en adelante y posteriormente para que impacte en el desarrollo. Luego de esto se procede a la codificación teniendo como guía los prototipos presentados, los requerimientos adicionales solicitados, y las herramientas para poder trabajar e iniciar.

Se acordó también durante la reunión, que el administrador / soporte tendrían que visualizar la mayoría de pestañas dentro del panel, para tener un cotejo de como va realizándose el proceso de las incidencias.

Luego de todo lo presentado al administrador de la empresa, accedió a su conformidad e inicio inmediato del desarrollo del Sistema Web.

Finalmente, cabe decir que estarán a la espera del desarrollo del Sprint 3, e inicio de pruebas unitarias con data ingresada por personal del Colegio.

  
Rica Estrada Mejía  
Jefe de Tecnologías de Información



*Ilustración 174: Acta de Reunión de Sprint 3*