



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E
INDUSTRIAL**

**CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES E
INFORMÁTICOS**

**TEMA: “Implementación de una aplicación web Help Desk para la Cooperativa de
ahorro y crédito Kullki Wasi.”**

Trabajo de Graduación Modalidad: TEMI. Trabajo Estructurado de Manera Independiente, presentada como requisito previo a la Obtención del Título de Ingeniero en Sistemas Computacionales e Informáticos.

Sublínea de Investigación: Aplicaciones Web.

Autor: Gonzalo Alexander Ruiz

Tutor: Ing. Clay Aldas, Mg.

Ambato – Ecuador

APROBACIÓN DEL TUTOR

En mi calidad de tutor del trabajo de investigación sobre el tema: “**Implementación de una aplicación web Help Desk para la Cooperativa de Ahorro y Crédito Kullki Wasi**”, del señor Gonzalo Alexander Ruiz S, estudiante de la carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los trámites y consiguientes y aprobación de conformidad con el Art. 16 del Capítulo II, del Reglamento de Graduación para obtener el Título Terminal de Tercer Nivel de la Universidad Técnica de Ambato.

Ambato, noviembre de 2017

EL TUTOR

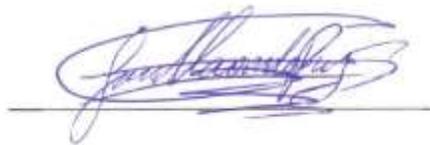


Ing. Clay Aldas, Mg.

AUTORÍA

El presente trabajo de investigación titulado: **“Implementación de una aplicación web Help Desk para la Cooperativa de Ahorro y Crédito Kullki Wasi.”** es absolutamente original, auténtico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, noviembre de 2017



Gonzalo Alexander Ruiz Sánchez

CI: 0503451510

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi trabajo de Titulación, con fines de difusión pública, además autorizo su reproducción dentro de las regulaciones de la Universidad.

Ambato, noviembre de 2017

A handwritten signature in blue ink, appearing to read 'Gonzalo Alexander Ruiz Sánchez', is written over a horizontal line.

Gonzalo Alexander Ruiz Sánchez

CI: 0503451510

APROBACIÓN DE LA COMISIÓN CALIFICADORA

La Comisión Calificadora del presente trabajo conformada por los señores docentes PhD. Félix Fernández, Ing. Carlos Núñez, revisó y aprobó el Informe Final del Proyecto de Investigación titulado “**Implementación de una aplicación web Help Desk para la Cooperativa de Ahorro y Crédito Kullki Wasi**”, presentado por el señor Gonzalo Alexander Ruiz Sánchez de acuerdo al numeral 9.1 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.



Ing. Pilar Urrutia

PRESIDENTA DEL TRIBUNAL



PhD. Félix Fernández

DOCENTE CALIFICADOR



Ing. Carlos Núñez

DOCENTE CALIFICADOR

DEDICATORIA

Dedico este proyecto a mis padres, familia, amigos y profesores quienes con su cariño, amistad y sincero apoyo han sido la fortaleza que me permitió seguir día a día hasta culminar este sueño. Y por último pero no menos importante a mi novia por brindarme su amor y comprensión cuando más lo necesite.

Gonzalo Alexander Ruiz Sánchez

AGRADECIMIENTO

Quiero agradecer a mis padres por brindarme la oportunidad de estudiar y vivir la vida universitaria.

A toda mi familia por apoyarme en todo momento, a mis amigos por ofrecer su amistad sincera.

A Karina y su padre quienes me brindaron su cariño y apoyo para no rendirme y continuar.

Y por último pero no menos importante, a mi tutor Ing. Clay Aldas, profesores y trabajadores universitarios por toda sus enseñanzas a lo largo de mi vida universitaria.

Sin ustedes nada hubiera sido posible.

Gracias.

Gonzalo Alexander Ruiz Sánchez

ÍNDICE DE CONTENIDOS

ÍNDICE DE CONTENIDOS	viii
ÍNDICE DE FIGURAS	xii
ÍNDICE DE TABLAS	xiv
ÍNDICE DE FÓRMULAS	xvi
INTRODUCCIÓN	xvii
RESUMEN EJECUTIVO.....	xviii
CAPÍTULO 1 EL PROBLEMA	1
1.1 Tema.....	1
1.2 Planteamiento del problema.....	1
1.3 Delimitación.....	3
1.4 Justificación.....	3
1.5 Objetivos	5
1.5.1 Objetivo general.....	5
1.5.2 Objetivo específicos.....	5
2 CAPÍTULO 2 MARCO TEÓRICO	6
2.1 Antecedentes investigativos	6
2.2 Fundamentación teórica	7
2.2.1 Introducción a ITIL.....	7
2.2.2 Ventajas de ITIL para el cliente/ usuario.....	8

2.2.3	Ventajas de ITIL para la organización	8
2.2.4	Problemas potenciales de ITIL	9
2.2.5	Aplicación web	9
2.2.6	Intranet	10
2.2.7	Help Desk	10
2.2.8	Microsoft SQL Server.....	11
2.2.9	Microsoft Visual Studio.....	12
2.2.10	Diseño web responsivo	12
2.2.10.1	¿En qué consiste el diseño responsivo?	12
2.2.11	Framework Bootstrap	13
2.2.12	Jquery.....	14
2.2.13	.NET Framework	17
2.2.14	Aplicaciones ASP.NET	18
2.2.15	Estructura de un sitio web ASP.NET.....	18
2.2.16	Cliente web	19
2.2.17	IIS (Internet Information Server-Servicios de información de Internet) ...	19
2.2.18	Metodología ágil	20
2.2.19	Manifiesto ágil	20
2.2.20	Metodología de desarrollo XP (eXtreme Programming).....	21
2.3	Propuesta de solución.....	22

3	CAPÍTULO 3 METODOLOGÍA	24
3.1	Modalidad de la investigación	24
3.2	Población y muestra	24
3.3	Recolección de información.....	24
3.4	.Procesamiento y análisis de datos	25
3.4.1	Términos generales	25
3.4.2	Analizando el manual mesa de ayuda.....	25
3.4.3	Análisis para roles de usuario	27
3.4.4	Análisis de criticidad y clasificación de incidentes	30
3.4.5	Comparativa de sistemas help desk	34
3.5	Desarrollo del proyecto	37
4	CAPÍTULO 4 DESARROLLO DE LA PROPUESTA.....	38
4.1	Fase I: Exploración	38
4.1.1	Entrevista realizada al Jefe de TIC de la Cooperativa Kullki Wasi.....	38
4.2	Fase II: Planificación del proyecto.....	39
4.2.1	Historias de usuario	39
4.2.2	Estimación de Historias de Usuario.....	44
4.3	Fase III: Iteraciones.....	45
4.3.1	Plan de Entrega	45
4.4	Fase IV: Producción.....	46

4.4.1	Diseño del modelo de Atención Help Desk.....	47
4.4.2	Análisis para el almacenamiento de la información	49
4.4.3	Diseño Base de datos	50
4.4.4	Desarrollo del módulo de ingreso al sistema Help Desk	52
4.4.5	Desarrollo del módulo registro de casos.....	57
4.4.6	Desarrollo del módulo base de conocimiento.....	62
CAPÍTULO 5 CONCLUSIONES Y RECOMENDACIONES.....		67
5.1	Conclusiones	67
5.2	Recomendaciones.....	68
BIBLIOGRAFÍA		69

ÍNDICE DE FIGURAS

Figura 2–1 Diseño Responsive	13
Figura 2–2 Sitio Web Oficial de JQuery.....	14
Figura 2–3 Arquitectura ASP.NET.....	18
Figura 3–1 Estructura organizacional Kullki Wasi.....	28
Figura 3–2 Estructura organizacional Departamento financiero	29
Figura 3–3 Aplicación Centro de Asistencia CRM DESK.....	34
Figura 3–4 Base de conocimiento Aplicación Centro de Asistencia CRM DESK	35
Figura 3–5 Sistema service desk plus	36
<i>Figura 4–1 Proceso Help Desk</i>	<i>47</i>
Figura 4–2 Proceso Help Desk	48
<i>Figura 4–3 Modelo Entidad Relación de la base de datos Help Desk.....</i>	<i>51</i>
Figura 4–4 Modelo de Interfaz-Ingreso al sistema	52
Figura 4–5 Interfaz-Ingreso al sistema	53
Figura 4–6 Código fuente para ingreso al sistema.....	55
Figura 4–7 Modelo de Interfaz-Ingreso al registro y revisión de casos al sistema.....	57
Figura 4–8 Interfaz de para el registro de casos.	58
Figura 4–9 Código fuente registro de casos.....	60
Figura 4–10 Modelo de Interfaz-Base de Conocimiento.....	63
Figura 4–11 Interfaz para el ingreso al sistema	64

Figura 4–12 Código fuente para la base de conocimiento..... 65

ÍNDICE DE TABLAS

Tabla 3-1 Roles de usuario	30
Tabla 3-2 Valores estimados para problemas de hardware	32
Tabla 3-3 Valores estimados para problemas de software	32
Tabla 3-4 Valores estimados para problemas de comunicación.....	33
Tabla 3-5 Valores estimados para problemas varios	33
Tabla 3-6 Cuadro comparativo de sistemas help desk.....	37
Tabla 4-1 Base para Historias de Usuario	40
Tabla 4-2 Historia de Usuario Bienvenida al sistema.....	41
Tabla 4-3 Historia de Usuario Base de conocimiento	41
Tabla 4-4 Historia de Usuario Directorio telefónico	41
Tabla 4-5 Historia de Usuario Ingreso al sistema.....	42
Tabla 4-6 Historia de Usuario Registro de casos.....	42
Tabla 4-7 Historia de Usuario Dashboard o menú	42
Tabla 4-8 Historia de Usuario Atender Caso.....	43
Tabla 4-9 Historia de Usuario Subir documentos.....	43
Tabla 4-10 Historia de Usuario Reportes	43
Tabla 4-11 Historia de Usuario Formularios de mantenimiento	44
Tabla 4-12 Estimación historias para Registro de Usuarios	44
Tabla 4-13 Estimación historia para Registro y solución de casos.....	44

Tabla 4-14 Estimación historias para Base de Conocimiento	45
Tabla 4-15 Iteraciones de los módulos	45
Tabla 4-16 Plan de entrega	46
Tabla 4-17 Niveles de criticidad	49
Tabla 4-18 Prueba de funcionamiento	57
Tabla 4-19 Prueba de Funcionamiento	62
Tabla 4-20 Prueba de Funcionamiento	66

ÍNDICE DE FÓRMULAS

Fórmula 3-1 Valor de criticidad.....	33
--------------------------------------	----

INTRODUCCIÓN

El desarrollo del informe final del presente trabajo de investigación se encuentra dividido en capítulos, los mismos que facilitan la comprensión del contenido del mismo:

El Capítulo I.- Denominado “EL PROBLEMA”, se encuentra el problema a investigar, la justificación respectiva de la investigación y el planteamiento de los objetivos a obtener tras la culminación del trabajo de investigación.

El Capítulo II.- Contiene el “MARCO TEÓRICO”, en donde se establece el conjunto de conocimientos en los cuales se sustenta la investigación, además de presentar investigaciones previas que sirven de soporte a la investigación y se establece la propuesta de solución del problema.

El Capítulo III.- Comprende la “METODOLOGÍA”, en la cual se especifica la metodología de investigación a utilizar, el proceso de recolección de la información, el procesamiento y análisis de la información recabada y define las etapas para el desarrollo del proyecto.

El Capítulo IV.- Consta del “DESARROLLO DE LA PROPUESTA”, que detalla la metodología seleccionada para el desarrollo de la Aplicación Web Help Desk Kullki Wasi.

El Capítulo V.- Comprende las “CONCLUSIONES Y RECOMENDACIONES”, donde se refleja las enseñanzas que deja el proyecto y las sugerencias presentadas al término de la investigación.

RESUMEN EJECUTIVO

Las Ciencias Informáticas trabajan día a día para poder tecnificar la actividad humana y traer progreso a la sociedad. La Universidad Técnica de Ambato con su Carrera de Ingeniería en Sistemas Computacionales e Informáticos orienta a sus estudiantes y futuros profesionales a ser partícipes en el desarrollo de las empresas e instituciones mediante aplicaciones que solucionen y faciliten la optimización de tiempo en los procesos que se realizan a diario.

La Cooperativa de Ahorro y Crédito Kullki Wasi institución financiera Tungurahuese, planea agrupar todos sus procesos cotidianos de TI en específico las actividades que surgen en soporte técnico, como los préstamos de equipos, reparación de dispositivos, asistencia técnica, etc. En un proceso más estructurado y unificado como son los servicios de TI para lo cual se ve en la necesidad de tecnificar la manera de solicitar y registrar todas estas peticiones que se realizan al agente de soporte técnico. Para lo cual se realizó una investigación en la que se encontró una solución rápida y eficaz.

La solución ha liberado la sobrecarga de trabajo que viene acarreado el agente de soporte técnico día tras día al tener que registrar y atender todas las solicitudes de los empleados de la Cooperativa Kullki Wasi, la aplicación desarrollada realiza el registro de casos o incidencias en una base de datos y registra las actividades realizadas por el agente de soporte técnico.

Por último se ha implementado una base de datos llamada “base de conocimiento” en la aplicación web Help Desk Kullki Wasi que ayuda a los empleados a ser más productivos y proactivos al solucionar sus problemas mediante la consulta o revisión de los problemas solucionados en el departamento de TIC por parte del agente de soporte técnico.

CAPÍTULO 1

EL PROBLEMA

1.1 Tema

Implementación de una aplicación web Help Desk para la Cooperativa de Ahorro y Crédito Kullki Wasi.

1.2 Planteamiento del problema

La Cooperativa de Ahorro y Crédito Kullki Wasi es una entidad financiera Indígena y privada que impulsa el desarrollo socio económico de la población rural y urbana marginal, especializada en micro-finanzas. Creada mediante acuerdo ministerial número N° 6582 de enero 23 del 2003 en la Provincia de Tungurahua y hoy cuenta con más de 60.000 socios y está ubicada en el segmento dos a nivel financiero.

Este incremento en el número de socios se debe a que la Cooperativa Kullki Wasi ha logrado impulsar su crecimiento y expansión a lo largo de la Sierra Centro este crecimiento le ha traído puntos buenos y malos, como puntos buenos podemos mencionar el reconocimiento como entidad financiera segura y estable, el crecimiento inmobiliario, incremento en el personal y un gran aumento en su capital interno razón por la cual momentáneamente se encuentra en el segmento dos a nivel financiero como podemos observar en la lista de segmentos de la COSEDE (ver anexo N°1) pero próximamente gracias a su gran gestión empresarial puede ascender a segmento uno lo cual traerá nuevos retos a la entidad.

No obstante, no se libra de los problemas que conlleva este desarrollo, al ser una empresa que ha incrementado sus activos fijos con un total de nueve sucursales en la actualidad, se

vio con el problema de comunicación y tecnología, para lo cual creó el Departamento de Sistemas quienes son los llamados a encargarse de estas áreas, dicho departamento está conformado por tres áreas: Gerencia, Desarrollo y Soporte Técnico juntas las tres áreas conforman un personal minúsculo de cuatro empleados.

El crecimiento en el personal ha hecho que varios empleados usen recursos informáticos para su diaria labor, al usar recursos informáticos estos tienden a deteriorarse o a su vez presentan algún inconveniente debido a varias causas como por ejemplo fallos en el software, fallos de hardware y uno de los más usuales el factor humano. Al ser el departamento de sistemas la vigilia de la tecnología su área de soporte técnico es la más utilizada de todas, ya que los problemas se presentan a diario.

Soporte técnico tiene que resolver cualquier fallo que se genere a lo largo de la jornada laboral, eso indica que, el agente de soporte técnico debe movilizarse constantemente entre las nueve agencias, para lo cual el agente debe viajar y muchas veces dejar de alto otras actividades.

Esta excesiva movilidad que se genera cuesta tiempo y mucho esfuerzo al agente de soporte, el agente de soporte al tener varias tareas por cumplir a diario se ve en la necesidad de registrar y administrar sus tiempos y tareas cumplidas a lo largo de la jornada, algo que al momento se lo está llevando de manera ordenada pero ineficiente, ya que se lo hace por medio de hojas de cálculo. Esta administración es muy básica y no está generando frutos a medida que pasa el tiempo, los problemas se resuelven, pero son reincidentes ya que capacitar al personal para que sea auto suficiente es una tarea muy ardua y extenuante.

Como respuesta a los múltiples problemas con los empleados el departamento de sistemas ha optado por unificar sus servicios de soporte técnico, como anticipo a un cercano ascenso al segmento uno donde ya es necesario que los procesos informáticos cumplan con ITIL y a su vez con normas COBIT es necesario iniciar con el uso de algunos fundamentos de ITIL como son soporte al servicio y entrega del servicio, como explica los fundamentos de mejores prácticas informáticas, para esto se ha pensado en empezar a

implementar una central de ayuda y varias políticas de atención al cliente, capacitación a los usuarios entre otras.

Como inicio de esta central esta contar con una aplicación base por lo cual la necesidad de desarrollarla es imperativo la aplicación web Help Desk en un principio ayudará en tres aspectos manejados por el agente de soporte técnico como son: registro de peticiones, registro de actividades y la denominada base de conocimiento para el uso del personal.

La herramienta web Help Desk será un inicio para a un futuro cumplir con los requerimientos que ascenso a segmento uno pueda solicitar.

1.3 Delimitación

- Área Académica: Software.
- Línea de Investigación: Desarrollo de software.
- Sublínea de Investigación: Aplicaciones Web.
- Delimitación Espacial: El presente proyecto se desarrollará en la Cooperativa de Ahorro y Crédito Kullki Wasi Agencia Matriz, Departamento de Sistemas.
- Delimitación Temporal: El desarrollo de este trabajo durará 6 meses a partir de la fecha de aprobación por parte del Consejo Directivo.

1.4 Justificación

El desarrollo del presente proyecto nace de la problemática que enfrenta hoy por hoy la Cooperativa de ahorro y crédito Kullki Wasi, en específico el Departamento de Sistemas, el cual al ser un departamento con recursos técnicos y humanos limitados se ve en la necesidad optimizar el tiempo que gasta el agente de soporte técnico en registrar tanto los problemas como las soluciones en el proceso de soporte técnico de la cooperativa y en el departamento.

El desarrollar una aplicación web Help Desk a medida de las actuales necesidades ha sido la petición del jefe del departamento cuyo titular es Ingeniero Hernán Sánchez el sistemas a desarrollarse debe ser capaz de cumplir su cometido para luego ser mejorado paulatinamente, los procesos actuales que han sido planteados y aprobados en el manual

de mesa de ayuda como podemos observar en el anexo N°2, refleja el diario acontecer en la resolución de los problemas originados por el personal. Si bien el manual fue elaborado siguiendo fundamentos ITIL, la institución aún no cuenta con una certificación que lo avale; lo cual descarta la pronta adquisición de un software licenciado o a su vez implementar un sistema help desk gratuito por la razón del no cumplimiento con la manera de actuar en relación a los procesos internos de la cooperativa, el adquirir o implementar una aplicación web help desk podría causar una pérdida económica innecesaria o a su vez obligan al departamento y a los empleados a adaptarse a una nueva manera de trabajo que puede causar más problemas que soluciones, razón por la cual cabe recalcar el Jefe del departamento Ingeniero Hernán Sánchez solicitó el desarrollo de una aplicación en base a los procesos actuales que facilite los registros al agente de soporte técnico y empiece a recopilar información que puede ser relevante posteriormente.

Implementar una herramienta sencilla de entender y fácil de manipular proporcionará a los empleados una pronta asimilación del software. Además que permitirá al departamento de sistemas poner en práctica los fundamentos de ITIL como son el “soporte al servicio” y “entrega del servicio” que fueron inculcados en el curso de FUNDAMENTOS DE ITIL 2011 para las buenas prácticas informáticas, realizado en la Universidad Técnica de Ambato dictado por CEDIA en el mes de Abril del 2015, como se puede validar mediante la certificación obtenida al término del curso como se puede constatar en el anexo N° 3.

Al poder contar con una aplicación web desarrollada con los procesos actuales del departamento se hará fácil poder a un futuro implementar más funcionalidades al aplicativo, pero para iniciar es necesarios contar con un software base que cumpla con lo establecido.

La aplicación será diseñada y estructurada según las políticas de desarrollo de software del departamento de sistemas por motivos de licenciamiento, de aprobarse este proyecto, me permitirá apoyar el desarrollo de esta creciente entidad financiera Tungurahuese.

1.5 Objetivos

1.5.1 Objetivo general

- Desarrollar una aplicación web Help Desk a la medida de las necesidades de la Cooperativa de Ahorro y Crédito Kullki Wasi agencia matriz como herramienta para la gestión de solicitudes al Departamento de Sistemas.

1.5.2 Objetivo específicos

- Analizar los procesos y actividades realizadas por el Departamento de Sistemas para el desarrollo de la aplicación Help Desk.
- Categorizar las solicitudes de los usuarios en la aplicación Help Desk, mediante un orden de prioridad para su respectiva resolución por parte de los agentes de soporte técnico.
- Alimentar un banco de datos con las soluciones de las solicitudes que se presenten a diario.
- Diseñar e implementar una aplicación web para la Cooperativa de Ahorro y Crédito Kullki Wasi.

CAPÍTULO 2

MARCO TEÓRICO

2.1 Antecedentes investigativos

La Cooperativa de Ahorro y Crédito Kullki Wasi al verse en la necesidad de crear su departamento de sistemas también prevé un ingreso al mundo de las buenas prácticas informáticas, para lo cual crea y establece políticas en el área de soporte técnico basándose en fundamentos ITIL que fueron inculcados en el curso de FUNDAMENTOS DE ITIL 2011 para las buenas prácticas informáticas, realizado en la Universidad Técnica de Ambato dictado por CEDIA en el mes de Abril del 2015, no obstante, al tener muchos proyectos en marcha la oportunidad de implementar una aplicación Help Desk queda abierta. En mi calidad de tesista investigador se me ha dado la oportunidad de realizar el mencionado proyecto bajo especificaciones y parámetros establecidos por el Jefe de TIC Ingeniero Hernán Sánchez.

Al iniciar la recolección de información relevante para el estudio me encuentro con una primera negativa, la cual es que nadie ha realizado un proyecto de Help Desk en la Facultad de Ingeniería en Sistemas Computacionales e Informáticos de la Universidad Técnica de Ambato.

La plataforma será desarrollada mediante el uso de nuevos aspectos tecnológicos como Bootstrap que nos otorga la cualidad de tener un diseño responsivo en la aplicación, esta tecnología es perteneciente a la compañía Twitter pero su código es abierto y libre, cumpliendo así con los parámetros de licenciamiento de la Cooperativa Kullki Wasi [1].

2.2 Fundamentación teórica

Como toda investigación es necesaria una base teórica para poder realizar la investigación, a continuación, se presenta información base que se debe conocer para dar inicio a la investigación.

2.2.1 Introducción a ITIL

Hace unas décadas que los desarrollos en tecnologías de la información vienen teniendo un gran impacto en los procesos del negocio. La introducción de la tecnología, cliente / servidor e Internet han permitido que las organizaciones lleven sus productos al mercado más rápido. Estos desarrollos han marcado la transición de la era industrial a la de la información.

Las organizaciones jerárquicas tradicionales tienen dificultades para adaptarse a mercados en constante cambio, ello ha marcado una tendencia hacia organizaciones menos jerárquicas y más flexibles. De igual manera, dentro de las organizaciones se ha puesto énfasis en cambiar de funciones verticales, o departamentos a procesos horizontales que se extienden a toda la organización, y se le otorga al personal de menor nivel la autoridad para tomar decisiones.

Teniendo en cuenta estos aspectos básicos se desarrollaron los procesos operativos de la gerencia de servicio en tecnologías de la información.

Computación, hoy Ministerio de Comercio, OGC) a que desarrollara una propuesta para que los ministerios y demás oficinas del sector público de Gran Bretaña utilizaran de manera eficaz y con eficiencia de costos los recursos en tecnologías de la información. El objetivo era desarrollar una propuesta sin compromisos con proveedor alguno. Esto dio como resultado la Information Technology Infrastructure Library (ITIL). ITIL nació de una colección de las mejores prácticas observadas en la industria del servicio de las tecnologías de la información.

ITIL brinda una descripción detallada de un número de prácticas importantes en tecnologías de la información, a través de una amplia lista de verificación, tareas, procedimientos y responsabilidades que pueden adaptarse a cualquier organización en

tecnologías de la información. En algunos casos hasta se han definido las prácticas como procesos que cubren las actividades más importantes de las organizaciones de servicio en tecnologías de la información. La vasta cantidad de temas cubiertos por las publicaciones transforma a la ITIL en un elemento de referencia útil para fijar nuevos objetivos de mejora para la organización en tecnologías de la información. La organización puede crecer y madurar con ellos.

En base a ITIL se han desarrollado varios sistemas para la administración de servicio en tecnologías de la información, generalmente organizaciones del negocio. Los ejemplos incluyen Hewlett & Packard (HP ITSM modelo de referencia), IBM (Modelo de Proceso), Microsoft y muchos otros.

Ésta es una de las razones por las que ITIL se ha convertido en el estándar de facto para describir.[2]

2.2.2 Ventajas de ITIL para el cliente/ usuario

La entrega de servicios de las tecnologías de la información se orienta más al cliente y gracias a los acuerdos sobre la calidad del servicio se mejora la relación.

Gracias a la metodología ITIL se describen mejor los servicios, se consigue un lenguaje más cómodo para el cliente, y a la vez con mayores detalles.

Como consecuencia de la utilización de ITIL se manejan mejor parámetros de calidad y de costo del servicio.

Su uso también mejora la comunicación con la organización al acordar los puntos de contacto.[2]

2.2.3 Ventajas de ITIL para la organización

La organización IT desarrolla una estructura más clara, se vuelve más eficaz, y se centra más en los objetivos corporativos

La administración tiene más control y los cambios resultan más fáciles de manejar.

Una estructura de proceso eficaz brinda un marco para concretar de manera más eficaz el outsourcing de los elementos de los servicios en tecnologías de la información.

Seguir las mejores prácticas de ITIL alienta el cambio cultural hacia la provisión de servicio, y sustenta la introducción de un sistema de administración de calidad basado en las series ISO 9000.

ITIL establece un marco de referencia para la comunicación interna y la comunicación con los proveedores, como así también la estandarización y la identificación de los procedimientos.[2]

2.2.4 Problemas potenciales de ITIL

Su introducción puede llevar tiempo y bastante esfuerzo, y supone un cambio de cultura en la organización. Una introducción demasiado ambiciosa puede llevar a la frustración porque nunca se alcanzan los objetivos.

Si la estructura de procesos se convierte en un objetivo en sí misma, la calidad del servicio se puede ver afectada de forma adversa. En ese caso, los procedimientos se transforman en obstáculos burocráticos que tratan de evitarse.

No hay progreso por la falta de comprensión sobre lo que deben dar los procesos, cuáles son los indicadores de desempeño, y cómo se controlan los procesos.

No se ven las reducciones de costo y la mejora en la entrega de los servicios.[2]

2.2.5 Aplicación web

Las aplicaciones web reciben este nombre porque se ejecutan en el Internet. Es decir que los datos o los archivos en los que trabajas son procesados y almacenados dentro de la web. Estas aplicaciones, por lo general, no necesitan ser instaladas en tu computador.

El concepto de aplicaciones web está relacionado con el almacenamiento en la nube. Toda la información se guarda de forma permanente en grandes servidores de Internet y nos envían a nuestros dispositivos o equipos los datos que requerimos en ese momento, quedando una copia temporal dentro de nuestro equipo.

Estos grandes servidores de Internet que prestan el servicio de alojamiento están ubicados alrededor de todo el mundo, así hacen que el servicio prestado no sea tan costoso o gratuito en la mayoría de los casos y extremadamente seguro [3].

2.2.6 Intranet

La aplicación web Help Desk funcionará en un entorno de intranet, lo que hace necesario saber su concepto para su aplicación. Una intranet es una red de ordenadores privados que utiliza tecnología Internet para compartir dentro de una organización parte de sus sistemas de información y sistemas operacionales. El término intranet se utiliza en oposición a Internet, una red entre organizaciones, haciendo referencia por contra a una red comprendida en el ámbito de una organización [4].

2.2.7 Help Desk

El proceso a tecnificar es el de asistencia técnica denominado Help Desk, para entender lo que es y cómo funciona se dice que Help Desk es un conjunto de servicios destinados a la gestión y solución de todas las posibles incidencias relacionadas con las tecnologías de la información y comunicación. Con Help Desk se puede recibir reportes de fallos, consultas de información o resolución de dudas y seguimiento de problemas. La Help Desk ofrece servicios acerca de soporte técnico en la detección de bugs o fallas en el software y hardware.

Se basa en un conjunto de recursos tecnológicos y humanos que brindan soporte técnico a los usuarios del área informática de una empresa para incrementar la productividad y la satisfacción de los usuarios internos o externos. Algunos ejemplos del personal destinado a la Help Desk son:

- Soporte a usuarios de sistemas microinformáticos
- Soporte telefónico u online
- Técnicos especializados en atención individualizada
- Apoyo sobre un sistema informático en particular

La Help Desk generalmente administra las peticiones de los usuarios por vía software para dar seguimiento a todos los requerimientos del sistema con un único número de ticket. A

esto se le llama “Seguimiento local de fallos”. Cuando se utiliza este software resultan ser herramientas muy eficaces en la búsqueda, análisis y eliminación de problemas comunes en el sistema.

Cuando el usuario notifica un problema, la Help Desk emite un ticket donde están especificados los detalles del problema. La Help Desk contiene varios niveles de solución de problemas de acuerdo con la complejidad de estos, si en un primer nivel, generalmente conocido como preguntas frecuentes el problema queda resuelto, el ticket es cerrado y actualizado de forma automática con la documentación de la solución para que los siguientes usuarios tengan una referencia del problema.

Sin embargo, si el problema persiste será desplazado a un segundo nivel. Por otro lado, la Help Desk asiste en la notificación de tendencias y situaciones para mejorar el servicio del software y que le permitirán alcanzar los más altos niveles de servicios para la gran comunidad de usuarios que la utilizan. Los servicios que brinda la Help Desk son:

- Atención inmediata a los incidentes o problemas con la continuidad operativa y reducción de tiempos en que el sistema no opera.
- Seguimiento y progreso a cada solicitud de servicio.
- Análisis de información sobre cada incidente que se presente para tomar decisiones, detectar áreas de vulnerabilidad y ejecutar acciones para la solución del problema presentado y la prevención de problemas potenciales.

El agente destinado a la atención de la Help Desk debe contar con la habilidad, conocimiento y capacidad para brindar una atención eficiente al usuario. Además de brindar soluciones a los problemas del usuario debe identificar las posibles consecuencias de cada acción para obtener la satisfacción del usuario, que es el objetivo principal de la Help Desk [5].

2.2.8 Microsoft SQL Server

La Cooperativa de Ahorro y Crédito Kullki Wasi en su calidad de institución que respeta todas las leyes de propiedad intelectual y uso de herramientas licenciadas adquirió por libre voluntad Microsoft SQL Server como sistema operativo para el funcionamiento de

sus servidores, para iniciar la investigación es conveniente saber que Microsoft SQL Server es un sistema de manejo de bases de datos del modelo relacional, desarrollado por la empresa Microsoft [6].

2.2.9 Microsoft Visual Studio

La Cooperativa de Ahorro y Crédito Kullki Wasi en su calidad de institución que respeta todas las leyes de propiedad intelectual y uso de herramientas licenciadas adquirió por libre voluntad Microsoft Visual Studio como herramienta para el desarrollo de nuevas aplicaciones que ayuden a la institución en su diario labor, este es un entorno de desarrollo integrado para sistemas operativos Windows. Soporta múltiples lenguajes de programación, tales como C++, C#, Visual Basic .NET, F#, Java, Python, Ruby y PHP, al igual que entornos de desarrollo web, como ASP.NET (Active Server Pages), a lo cual hay que sumarle las nuevas capacidades online bajo Windows Azure en forma del editor Monaco.

Visual Studio permite a los desarrolladores crear sitios y aplicaciones web como será la aplicación Kullki Wasi Help Desk, así como servicios web en cualquier entorno que soporte la plataforma .NET. Así, se pueden crear aplicaciones que se comuniquen entre estaciones de trabajo, páginas web, dispositivos móviles, dispositivos embebidos y consolas, entre otros [7].

2.2.10 Diseño web responsivo

Para un mejor desarrollo y previendo el cómo accederán los empleados de la Cooperativa de ahorro y crédito Kullki Wasi al aplicativo web Help Desk se usará el diseño web responsivo o adaptativo el cual es una técnica de diseño web que busca la correcta visualización de una misma página en distintos dispositivos. Desde ordenadores de escritorio a tabletas y móviles. Hoy en día accedemos a sitios web desde todo tipo de dispositivos; ordenador, tablet, smartphone por lo que, cada vez más, nos surge la necesidad de que nuestra web se adapte a los diferentes tamaños de los mismos [8].

2.2.10.1 ¿En qué consiste el diseño responsivo?

Se trata de re dimensionar y colocar los elementos de la web de forma que se adapten al ancho de cada dispositivo permitiendo una correcta visualización y una mejor experiencia

de usuario. Se caracteriza porque los contenidos e imágenes son fluidos y se usa código de CSS3.

El diseño responsivo permite reducir el tiempo de desarrollo, evita los contenidos duplicados, y aumenta la viralidad de los contenidos ya que permite compartirlos de una forma mucho más rápida y natural. Se basa en proporcionar a todos los usuarios de una web los mismos contenidos y una experiencia de usuario lo más similar posible, frente a otras aproximaciones al desarrollo web móvil como la creación de apps, el cambio de dominio o webs servidas dinámicamente en función del dispositivo [8].



Figura 2–1 Diseño Responsivo

Elaborado por: El investigador

2.2.11 Framework Bootstrap

La aplicación web Kullki Wasi Help Desk al usar diseño responsivo necesita del framework Bootstrap el cual es una herramienta Open Source creada originalmente por Twitter para el desarrollo rápido de aplicaciones web, este framework trabaja principalmente con CSS y JavaScript, ha ido creciendo en popularidad hasta convertirse en uno de los proyectos más destacados en la plataforma de código abierto GitHub. Las principales razones por cual adquirió gran popularidad se debe a características como:

- El soporte: Al ser un framework open source y a su vez muy popular, existen una gran cantidad de foros, guías, tutoriales, libros, videos y otra cantidad de recursos, que ayudan de una manera autodidacta el correcto uso del framework.

- Comodidad y Velocidad: Gracias a la magnífica cantidad de propiedades que conforman el framework para cada elemento, la facilidad para personalizar se destaca haciendo el diseño mediante el uso de hojas de CSS, una cualidad muy útil para establecer parámetros de diseño en el proyecto.
- Componentes: Por ultimo esta la cantidad ilimitada de componentes a los cuales se puede acceder gracias a que son desarrollados por la comunidad open source [9].

2.2.12 JQuery

Para un correcto desempeño del framework Bootstrap en el proyecto es necesario hacer el uso de la tecnología JQuery que es una librería de JavaScript. Esta librería de código abierto, simplifica la tarea de programar en JavaScript y permite agregar interactividad a un sitio web sin tener conocimientos del lenguaje.

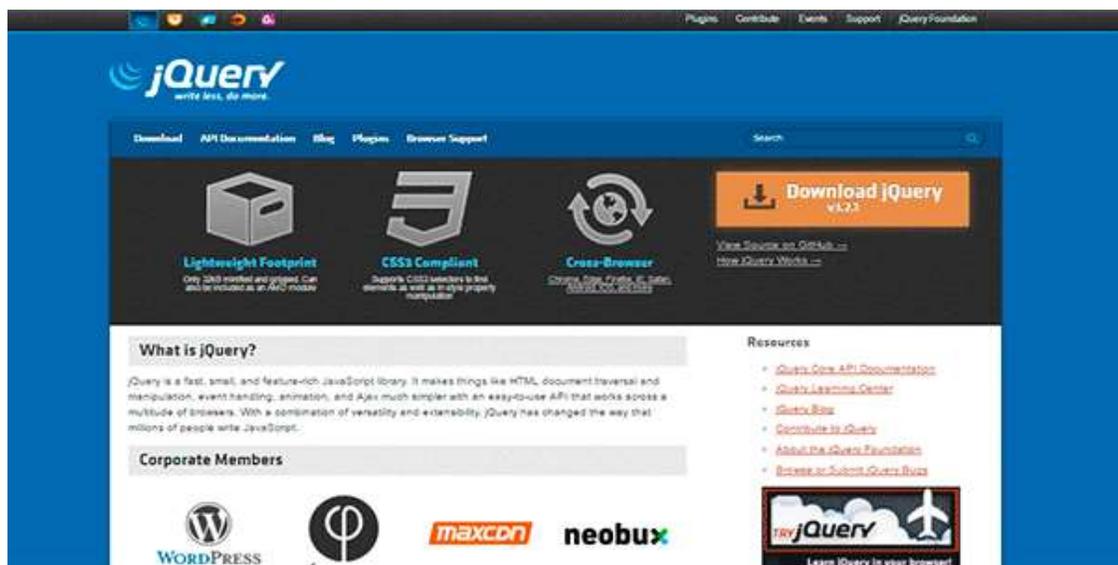


Figura 2–2 Sitio Web Oficial de JQuery

Fuente: laurachuburu / jQuery

Basados en esta librería, existe una infinita cantidad de plugins creados por desarrolladores de todo el mundo. Estos plugins resuelven situaciones concretas dentro del maquetado de un sitio, por ejemplo: un menú responsivo, una galería de fotos, un

carrousel de imágenes, un slide, un header que cambia de tamaño, el deslizamiento del scroll al hacer clic en un botón, la transición entre páginas y miles de efectos más.

Cada plugin tiene un sitio web desde donde se pueden descargar sus archivos, con demos, instrucciones para su implementación, opciones de configuración e información de las licencias. En la web hay cientos de blogs que recopilan y analizan los plugins según sus funcionalidades, reuniendo en un sólo post los links a varios plugins de función similar, lo que facilita mucho la búsqueda.

Jquery posee algunos elementos claves o con los cuales trabaja estos son:

- **Librería JQuery**

Todos los plugins de JQuery necesitan la librería de JQuery para funcionar. La librería es un archivo .js que se puede descargar desde el sitio oficial: <https://jquery.com/> colocar en una carpeta js y luego vincular con una etiqueta <script>, por ejemplo:

```
<script src="js/jquery-3.2.1.min.js"></script>
```

Ese vínculo conviene colocarlo dentro del body y hacia el final del HTML para no demorar la carga de los contenidos. Si descargamos el archivo y lo vinculamos de esa manera, debemos subir el archivo a nuestro hosting.

Otra forma de vincular la librería es utilizando el servidor de Google. De esta manera no necesitamos descargarla ni subirla a nuestro servidor. En ese caso, el código es el mismo pero con ruta absoluta:

```
<script  
src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js">  
</script>
```

La página donde Google ofrece los vínculos a diferentes recursos es la siguiente:

```
https://developers.google.com/speed/libraries/#jquery
```

Los archivos .min son archivos que están optimizados (con el código comprimido) y no son editables, se usan en producción cuando se considera que no es necesario hacerle modificaciones. La ventaja es que son más livianos y por lo tanto cargan más rápido.

- **Archivo .js del plugin**

Este es el archivo principal del plugin y se obtiene descargando del sitio. Habitualmente el nombre del archivo coincide con el nombre del plugin, por ejemplo:

```
<script src="js/responsiveslides.min.js"></script>
```

Hay que tener en cuenta que si copiamos este código de las instrucciones de instalación tenemos que chequear las rutas de los archivos, especialmente el nombre de las carpetas.

- **Estructura HTML (Lenguaje de Marcado para Hipertextos)**

En algunos casos las instrucciones del plugin sugieren una estructura determinada en HTML, por ejemplo, que una galería de fotos esté resuelta con listas o con divs o que un elemento sea identificado con class o con ID. Si en el momento de instalar el plugin ya tenemos escrito nuestro HTML, es conveniente adaptarlo a la estructura que sugiere el plugin para asegurarnos de que funcione y no tener que modificar otros elementos como reglas CSS.

- **Función**

La función es un bloque de código JavaScript que contiene instrucciones que necesita el plugin para ejecutarse. Se escribe dentro del body entre las etiquetas <script></script>. Puede ir inmediatamente debajo de la estructura HTML del plugin o antes de que cierre la etiqueta body. Si es posible es mejor colocarla hacia el final del documento para que la descarga de la página sea más rápida.

```
<script>

$(document).ready(function() { $("#owl").owlCarousel({ autoPlay: 3000, items
: 3, itemsDesktop : [1199,3], itemsDesktopSmall : [979,3] }); });

</script>
```

- **Reglas CSS**

La mayoría de los plugins tienen sus propias reglas CSS. Si son pocas líneas conviene incorporarlas a nuestra hoja de estilos, pero si son muchas, entonces mejor guardar el archivo .css descargado en nuestra carpeta css y colocar el vínculo desde el head. El vínculo a la hoja de estilos normalmente está indicado en las instrucciones del plugin, pero es necesario chequear la ruta para que coincida con la ubicación que le dimos en nuestro proyecto.

- **Imágenes**

Algunos plugins incorporan imágenes de íconos que son llamados desde el código CSS. En este caso, y si el código es extenso, es conveniente mantener la relación de la ubicación entre el CSS y las imágenes para no romper las rutas. Es más seguro que la carpeta imágenes del plugin quede junto con los CSS que moverla a nuestra carpeta imágenes.

En relación a las rutas y en algunos casos si el plugin tiene mucha cantidad de archivos (todos relacionados entre sí) conviene mantener la carpeta intacta y colocarla por ejemplo dentro de nuestra carpeta js aunque dentro de ella haya archivos css o imágenes. Además, en el caso de querer remover el plugin de nuestra página, sería mucho más simple encontrar los archivos. [10]

2.2.13 .NET Framework

Para permitir generar y ejecutar el aplicativo Help Desk se hará uso de .Net Framework en su versión 4.0 en la que algunos componentes facilitan la integración, ya que comparten datos y funcionalidad a través de una red mediante protocolos estándar independientes de la plataforma, como XML (eXtensible Markup Language-Lenguaje de Marcado

Extensible) y HTTP (HyperText Markup Language -Protocolo de transferencia de hipertexto) [11].

2.2.14 Aplicaciones ASP.NET

El lenguaje web que se usará en el desarrollo es ASP.NET este nos proporciona un modelo de desarrollo Web unificado que incluye los servicios necesarios para crear aplicaciones Web para la empresa. ASP.NET forma parte de .NET Framework y permite aprovechar al máximo las características de Common Language Runtime, como la seguridad de tipos, la herencia, la interoperabilidad de lenguajes y el control de versiones. Los temas de esta sección proporcionan información conceptual sobre el funcionamiento de ASP.NET y procedimientos que muestran cómo escribir código que permite aprovechar las posibilidades de ASP.NET para crear aplicaciones Web dinámicas [11].

2.2.15 Estructura de un sitio web ASP.NET

Para estructurar el sitio web a desarrollar se hará mediante una implementación de ASP, escrita de cero en C#. Utiliza lenguajes de programación compilados como C#, para escribir aplicaciones Web. Las aplicaciones son compiladas en el servidor, y las páginas son generadas en HTML específicamente para el browser que hizo la invocación.

Se integra en la autenticación del sistema operativo Windows y permite una configuración a nivel de aplicación.



Figura 2-3 Arquitectura ASP.NET

Fuente: Microsoft

Los conceptos centrales de ASP .NET son: Separar presentación de lógica del negocio, Usar servicios provistos por el .NET Framework, El código es compilado la primera vez que se accede a una página, Administración de estado, Utilización de cualquier lenguaje, Actualizar archivos mientras se está ejecutando la aplicación [11].

2.2.16 Cliente web

El cliente web trabajará en intranet mediante un requerimiento de servicio. El requerimiento inicial puede convertirse en múltiples requerimientos de trabajo a través de redes LAN. La ubicación de los datos o de las aplicaciones será totalmente transparente para el cliente [12].

2.2.17 IIS (Internet Information Server-Servicios de información de Internet)

Para la publicación del sitio web Help Desk Kullki Wasi se ha pensado realizarlo en IIS el cual es un conjunto de servicios para servidores usando Microsoft Windows. IIS viene integrado con Windows NT 4.0. Dado que el IIS está tan íntimamente integrado con el sistema operativo, es relativamente fácil de administrar.

El rol Servidor web (IIS) en Windows Server 2012 proporciona una plataforma segura, fácil de administrar, modular y extensible donde hospedar sitios web, servicios y aplicaciones de manera confiable. Con IIS 8, puede compartir información con usuarios en Internet, en una intranet. IIS 8 es una plataforma web unificada que integra IIS, ASP.NET, servicios de FTP (Protocolo de Transferencia de Archivos).

En la lista siguiente, se indican solo algunas de las ventajas que obtendrá al usar IIS 8 y él porque será usado en el proyecto de investigación:

- La seguridad web se refuerza gracias a una superficie reducida de servidor y al aislamiento automático de aplicaciones.
- Podrá implementar y ejecutar aplicaciones web de ASP.NET y ASP clásico en el mismo servidor de forma sencilla.
- Se logra el aislamiento de aplicaciones al proporcionar a los procesos de trabajo una identidad única y una configuración en espacio aislado de manera predeterminada, lo que reduce aún más los riesgos de seguridad.

- Podrá agregar y eliminar componentes IIS integrados e incluso reemplazarlos fácilmente por módulos personalizados que se adapten a las necesidades del cliente.
- Aumenta la velocidad del sitio web mediante el almacenamiento en caché dinámico integrado y la compresión mejorada [13].

2.2.18 Metodología ágil

“En una reunión celebrada en febrero de 2001 en Utah-EEUU, nace el término "ágil" aplicado al desarrollo de software. En esta reunión participan un grupo de 17 expertos de la industria del software, incluyendo algunos de los creadores o impulsores de metodologías de software. Su objetivo fue esbozar los valores y principios que deberían permitir a los equipos desarrollar software rápidamente y respondiendo a los cambios que puedan surgir a lo largo del proyecto. Se pretendía ofrecer una alternativa a los procesos de desarrollo de software tradicionales, caracterizados por ser rígidos y dirigidos por la documentación que se genera en cada una de las actividades desarrolladas. Varias de las denominadas metodologías ágiles ya estaban siendo utilizadas con éxito en proyectos reales, pero les faltaba una mayor difusión y reconocimiento” [14].

El inicio de esta metodología es el Manifiesto Ágil, un documento que plasma esta filosofía.

2.2.19 Manifiesto ágil

Comienza enumerando los principales valores del desarrollo ágil. Se valora:

- “Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas. La gente es el principal factor de éxito de un proyecto software. Si se sigue un buen proceso de desarrollo, pero el equipo falla, el éxito no está asegurado; sin embargo, si el equipo funciona, es más fácil conseguir el objetivo final, aunque no se tenga un proceso bien definido. No se necesitan desarrolladores brillantes, sino desarrolladores que se adapten bien al trabajo en equipo. Así mismo, las herramientas (compiladores, depuradores, control de versiones, etc.) son importantes para mejorar el rendimiento del equipo, pero el disponer más recursos que los estrictamente necesarios también puede afectar negativamente. En resumen, es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de

construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.

- Desarrollar software que funciona más que conseguir una buena documentación. Aunque se parte de la base de que el software sin documentación es un desastre, la regla a seguir es “no producir documentos a menos que sean necesarios de forma inmediata para tomar una decisión importante”. Estos documentos deben ser cortos y centrarse en lo fundamental. Si una vez iniciado el proyecto, un nuevo miembro se incorpora al equipo de desarrollo, se considera que los dos elementos que más le van a servir para ponerse al día son: el propio código y la interacción con el equipo.
- La colaboración con el cliente más que la negociación de un contrato. Las características particulares del desarrollo de software hacen que muchos proyectos hayan fracasado por intentar cumplir unos plazos y unos costes preestablecidos al inicio del mismo, según los requisitos que el cliente manifestaba en ese momento. Por ello, se propone que exista una interacción constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.
- Responder a los cambios más que seguir estrictamente un plan. La habilidad de responder a los cambios que puedan surgir a lo largo del proyecto (cambios en los requisitos, en la tecnología, en el equipo, etc.) determina también el éxito o fracaso del mismo. Por lo tanto, la planificación no debe ser estricta puesto que hay muchas variables en juego, debe ser flexible para poder adaptarse a los cambios que puedan surgir. Una buena estrategia es hacer planificaciones detalladas para unas pocas semanas y planificaciones mucho más abiertas para unos pocos meses” [15].

2.2.20 Metodología de desarrollo XP (eXtreme Programming)

“Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje

para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.”

“Un proyecto XP tiene éxito cuando el cliente selecciona el valor de negocio a implementar basado en la habilidad del equipo para medir la funcionalidad que puede entregar a través del tiempo. El ciclo de desarrollo consiste en los siguientes pasos:”

1. El cliente especifica el valor de negocio a implementar.
2. El programador estima el esfuerzo necesario para su implementación.
3. El cliente selecciona qué desarrollar, de acuerdo con sus prioridades y las restricciones de tiempo.
4. El programador desarrolla ese valor de negocio.
5. Se regresa al paso 1.

“En todas las iteraciones de este ciclo tanto el cliente como el programador aprenden. No se debe presionar al programador a realizar más trabajo que el estimado, ya que se perderá calidad en el software o no se cumplirán los plazos. De la misma forma el cliente tiene la obligación de manejar el ámbito de entrega del producto, para asegurarse que el sistema tenga el mayor valor de negocio posible con cada iteración.” [16].

“El ciclo de vida ideal de XP consiste de seis fases: Exploración, Planificación de la Entrega, Iteraciones, Producción, Mantenimiento y Muerte del Proyecto” [17].

2.3 Propuesta de solución

La investigación en curso se ha encaminado al desarrollo de una aplicación web que pueda ser usada por todos los empleados o usuarios que estén vinculados de una u otra manera al Departamento de Sistemas, está ayudaría en gran medida al cumplimiento de las políticas que posee el Departamento de Sistemas y proporcionará un impulso técnico al mismo.

La aplicación Help Desk podrá ser usar en tres sentidos: como administradora de tareas, como banco de datos para solucionar inconvenientes menores y como supervisora de los recursos técnicos y humanos. Esta aplicación debe poder ser usada de la manera más autodidacta posible y manipulada desde cualquier medio tecnológico, que pueda ingresar

a un navegador de internet, lo cual genera la investigación en el uso del diseño responsivo que efectivamente fue creado con este fin.

CAPÍTULO 3 METODOLOGÍA

3.1 Modalidad de la investigación

La presente investigación es aplicable porque pretende automatizar el proceso Registro y control de la actividad de soporte técnico de la Cooperativa de ahorro y crédito Kullki Wasi, manteniendo información actualizada en tiempo real, beneficiando a la toma de decisiones y llevando un óptimo control de tiempos y actividades diarias.

Para la realización de la presente investigación se ha considerado utilizar las siguientes modalidades:

Modalidad Bibliográfica o Documental: Esta modalidad se debe aplicar porque en la investigación se recolectada información teórica de libros, libros digitales, artículos científicos, publicaciones, etc.

Modalidad de Campo: Esta también es aplicada en la investigación porque el investigador deberá asistir directamente a la institución para recolectar información acerca de sus necesidades de información.

Se utilizará la metodología de programación XP debido que es un método ágil y permite al sistema adaptarse a cambios no previstos durante la etapa del diseño.

3.2 Población y muestra

El proyecto de investigación presenta una naturaleza en la cual no es indispensable realizar una muestra de población ya que el aplicativo será instalado y utilizado únicamente por los miembros de la institución lo cual no necesita aceptación de un cliente potencial.

3.3 Recolección de información

Para la recolección de información del proyecto se contará con una plena disposición del personal del área de sistemas de la cooperativa Kullki Wasi, el personal en especial el agente de soporte técnico Paul Ruiz, serán los encargados de facilitar los procesos internos y bitácoras de trabajo como se puede apreciar en el “Instructivo de la Help Desk”. Para los lineamientos del desarrollo de la aplicación Help Desk, el Ingeniero Hernán Sánchez

será el encargado de aprobar el uso de las herramientas de desarrollo permitidas en la institución para el uso en la investigación y posterior implantación.

3.4 .Procesamiento y análisis de datos

Como fase inicial en el proceso de análisis de los datos se revisará los procesos internos actuales del departamento de sistemas para conocer la forma de trabajo de sus miembros, como paso a seguir se analizará los datos facilitados en las bitácoras de soporte técnico, en las cuales se validará y rectificará la información que manejará la aplicación, dicha información será analizada con el fin de saber si es o no útil para ser almacenada en la base de datos ya que la aplicación sustituirá el modo actual de realizar el registro de actividades del soporte técnico.

3.4.1 Términos generales

Incidente (o incidencia): Cualquier evento que no forma parte de la operación estándar de un servicio de TI y que causa, o puede causar, una interrupción o una reducción de la calidad del mismo.

Requerimientos relacionados con la tecnología de información: Consultas, preguntas o solicitudes para atención de incidentes, que realizan los usuarios de los sistemas, equipos informáticos y comunicaciones.

3.4.2 Analizando el manual mesa de ayuda

Tras una primera impresión al leer el manual “mesa de ayuda” se logra apreciar que refleja enormemente el trabajo realizado por el agente de soporte técnico, mas no refleja completamente la realidad del proceso y el día a día, para esto recopilaremos más información concerniente a la operatividad del agente.

Como podemos apreciar en la página 1 del anexo N° 2 al establecer acciones tentativas en el departamento de sistemas se plantean objetivos a cumplir, los objetivos que más se alinean al proceso de mejores prácticas puede decirse que son:

- Orientar al personal de la Cooperativa de Ahorro y Crédito Kullki Wasi en el procedimiento a seguir para Soportar los servicios de TI acordados, administrando

el aplicativo de mesa de ayuda, de forma ágil, eficiente y oportuna, en las dependencias de la Cooperativa de Ahorro y Crédito Kullki Wasi edificio Matriz, en las Gerencias Departamentales y Agencias.

- Conocer las Políticas y Procedimientos de configuración y Funcionamiento de los elementos de Hardware y Software de la Entidad.
- Determinar los lineamientos para ejecutar las actividades de soporte técnico Informático y de comunicaciones, en cada dependencia y/o Gerencia Departamental.

Como se menciona en la magazine digital Magazitum, con respecto a las fases iniciales de implementación de ITIL y de manejo de mejores prácticas TI, para establecer un inicio y cambiar la forma de pensar de las personas, en este caso los usuarios, es necesario empezar a entender que un departamento de sistemas no funciona únicamente como facilitador de dispositivos o reparador de dispositivos, un departamento de TI necesita establecer que el área de TI debe entender que lo administra y da a sus usuarios son servicios de TI, no dispositivos. El reto es lograr la integración eficiente de usuarios, procesos y tecnología para una mejor administración de los servicios de TI, optimizando el uso de los recursos y mejorando constantemente los niveles de servicio.

Los tres objetivos que más énfasis realizan en este proceso de cambio y mejora son los suscritos anteriormente, ya que cumplen con los niveles iniciales de ITIL [18].

Otro punto a tomar en cuenta es el alcance que se desea tener con todo el proceso de la mesa de ayuda planteado por el departamento, actualmente no se cumple enteramente con todo lo establecido.

Para citar el alcance se puede decir que: “Este procedimiento incluye la recepción y registro de la solicitud de servicio de soporte técnico, el análisis del tipo de soporte requerido, la atención al incidente, el registro del nivel de satisfacción por parte del usuario solicitante, y finalmente el análisis estadístico del nivel de satisfacción del servicio para la correspondiente toma de acciones de mejoramiento (si amerita)” [anexo N°2, pág. 1].

Para el sistema a desarrollar se solicita inicialmente tomar en cuenta los siguientes aspectos del alcance:

- Recepción y registro de la solicitud de servicio de soporte técnico.
- Análisis del tipo de soporte requerido.
- Atención al incidente.

3.4.3 Análisis para roles de usuario

Como se puede apreciar en la entrevista realizada al Jefe de TI en el anexo N° 4, los actores principales para el sistema son los trabajadores o empleados de la cooperativa Kullki Wasi, si bien se puede tomar como población a todos los empleados, en este caso en particular no será así, ya que no todos los empleados están relacionados con el departamento de TI para sus asuntos laborales, a continuación se van a poder observar mediante un organigrama oficial de la cooperativa Kullki Wasi las áreas a las que se les brinda soporte técnico, para una mejor apreciación se anexará el organigrama como anexo N° 5.

ESTRUCTURA ORGANIZACIONAL DE COAC. KULLKI WASI Ltda... 2015 - 2018

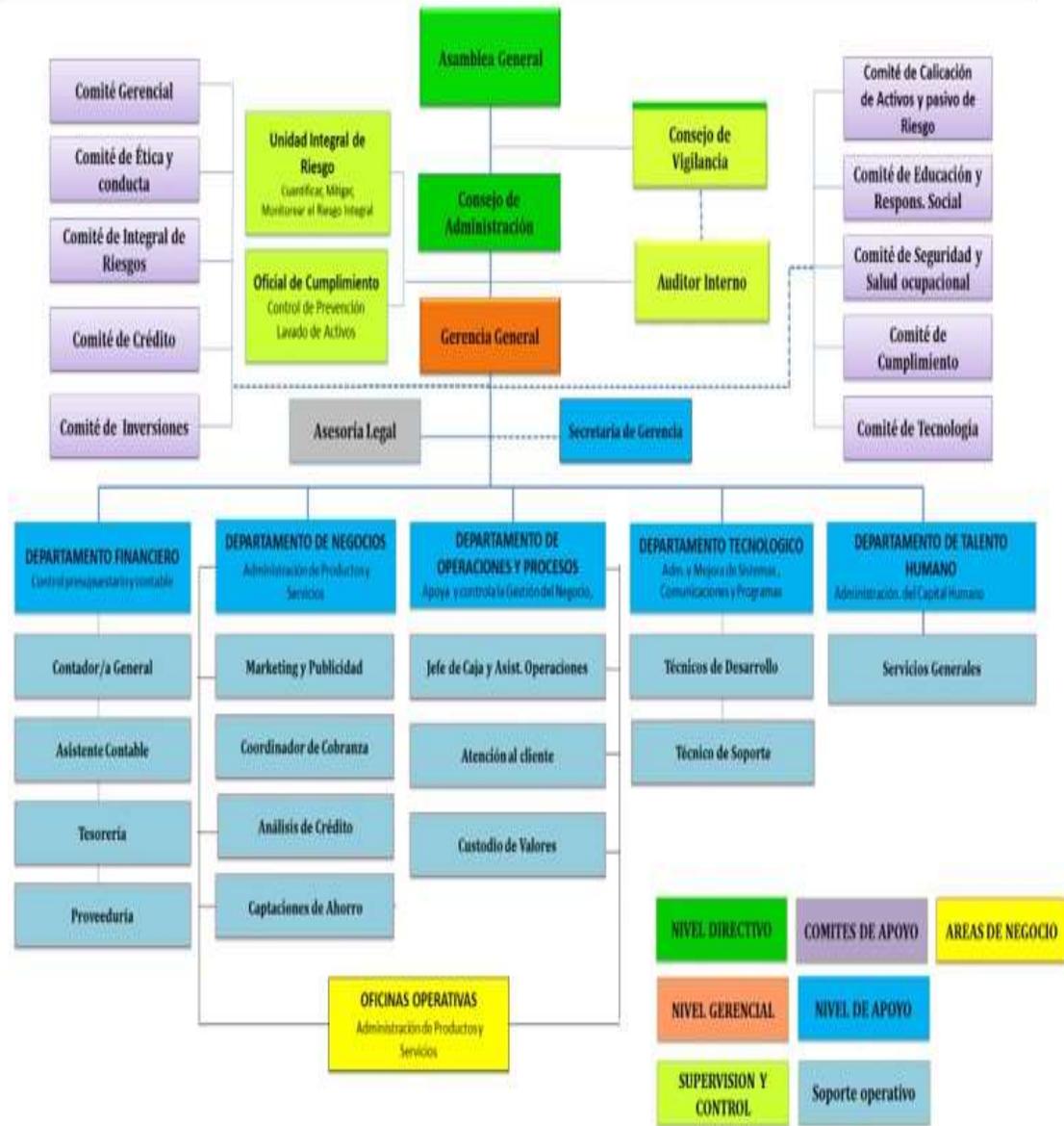


Figura 3–1 Estructura organizacional Kullki Wasi

Fuente: Kullki Wasi/ TI

Como podemos apreciar en el organigrama se estableció por parte del agente de soporte técnico que las áreas marcadas con color azul son las que mayormente tienen relación con el departamento de sistemas, pero a estas se suman las áreas de:

- Gerencia General

- Unidad Integral de riesgo.
- Oficial de Cumplimiento.
- Asesoría legal.
- Auditor interno.

Ahora si bien todos son usuarios a la vista del departamento de sistemas son importantes, unos usuarios por el rango o el cargo del que son parte tienen una importancia significativa.

Los roles de usuario describen a los empleados involucrados con el sistema y se detallan las características relevantes para el estudio de su función dentro de la cooperativa.

Para la clasificación tomaremos como base el rol que cumple el empleado en la cooperativa Kullki Wasi.



Figura 3-2 Estructura organizacional Departamento financiero

Fuente: Kullki Wasi/ TI

Como se menciona posteriormente se debe atender a los usuarios según su nivel de prioridad en la empresa por lo que se tomarán tres niveles:

- Nivel Gerencial
- Nivel de apoyo (Subjefes)
- Nivel operativo (Empleados)

Ahora si bien el esquema organizacional no está claro, si se puede evidenciar que en ciertos departamentos existe un subjefe o supervisor de departamento y a nivel gerencial existen plazas como es el caso de los gerentes para cada sucursal.

Para poder evidenciar de una mejor manera el nivel de rol y de prioridad que va a tomar cada empleado que sea considerado como usuario del sistema se representará su rol y su valor en la tabla genérica (3-1 Rol de usuario)

Nivel usuario	Valor asignado	Prioridad
Nivel Gerencial	3	Alta
Nivel Subjefe	2	Medio
Nivel Operativo	1	Medio

Tabla 3-1 Roles de usuario

Elaborado por: El investigador/ Jefe de TI

A partir de esta tabla de roles genérica, el resto de empleados podrán obtener su usuario y al momento de registrar se le colocará al nivel operativo que pertenecen.

3.4.4 Análisis de criticidad y clasificación de incidentes

Como ya se estableció anteriormente la mayoría de los usuarios son tratados de manera similar en el proceso de soporte técnico, para lo cual se ha realizado una valoración de cada rol para darle importancia.

No obstante, no es suficiente, ya que para cada usuario sus problemas no son igual que para otro. Tras un análisis a las bitácoras presentadas por el agente de soporte técnico que pueden ser apreciadas en el anexo N° 6, todos los problemas registrados son diferentes entre los usuarios.

Si bien los problemas están clasificados por grupos como son hardware, software, comunicacionales, y otros (problemas eléctricos y de proveedores externos), los usuarios no siempre van a saber en qué clase de grupo se encuentra su problema y ponerlos a describir es muy difícil e innecesario.

Para esto se va a describir una serie de problemas que son los más generales que se pueden observar en las bitácoras como son:

A nivel de hardware:

- Monitor
- Teclado
- Mouse
- Impresora
- Parlantes
- Scanner

A nivel de software:

- Sistema operativo
- Office
- Financiamiento
- Navegadores
- Instalar programa
- Antivirus

A nivel comunicacional:

- Antenas
- Teléfonos
- Celulares

Y por último a nivel de problemas varios:

- Cajero
- Problema eléctrico

Como se puede apreciar se tienen una gran cantidad de subcategorías donde pueden ingresar los problemas relacionados que manifieste cualquier usuario de la cooperativa,

Ahora la pregunta a resolver es; ¿Qué problema es más importante para cada usuario?,

Para responder a esta pregunta, las subcategorías serán ponderadas en torno a su importancia operacional en una escala, siendo uno el valor más bajo y 10 el valor más alto.

Para la ponderación para cada subcategoría se tomará en cuenta una entrevista al agente de soporte técnico y la revisión a las bitácoras del mismo donde se ha realizado una ponderación básica para cada subclase de problema.

Como se puede apreciar en el siguiente ordenamiento:

A nivel de hardware:

Subproblema	Valor estimado
Monitor	4
Teclado	4
Mouse	4
Impresora	4
Parlantes	3
Scanner	3

Tabla 3-2 Valores estimados para problemas de hardware

Elaborado por: El investigador/ Jefe de TI

A nivel de software:

Subproblema	Valor estimado
Sistema operativo	10
Office	8
Financiamiento	10
Navegadores	5
Instalar programa	4
Antivirus	4

Tabla 3-3 Valores estimados para problemas de software

Elaborado por: El investigador/ Jefe de TI

A nivel comunicacional:

Subproblema	Valor estimado
Antenas	8
Teléfonos	3
Celulares	3

Tabla 3-4 Valores estimados para problemas de comunicación

Elaborado por: El investigador/ Jefe de TI

Y por último a nivel de problemas varios:

Subproblema	Valor estimado
Cajero	5
Problema eléctrico	5

Tabla 3-5 Valores estimados para problemas varios

Elaborado por: El investigador/ Jefe de TI

El valor estimado para cada subcategoría puede cambiar en un futuro si así se lo desea.

Ahora que ya tenemos una equivalencia tanto para usuarios como para problemas, podemos decir que el valor de criticidad para la atención de los usuarios según el problema será:

$$\text{Valor de rol del usuario} + \text{Valor de la subcategoría del problema} = \text{Valor de criticidad}$$

Fórmula 3-1 Valor de criticidad

Elaborado por: El investigador

Este valor de criticidad nos permitirá atender las solicitudes de una manera ordenada y nos permitirá priorizar los casos que sean operativamente más urgentes en su resolución.

3.4.5 Comparativa de sistemas help desk

Como ya se ha mencionado antes es necesario entender que el factor monetario es muy limitado para realizar mejoras que no sean de primera necesidad y lastimosamente invertir en un sistema help desk a medida está fuera de opción, sin embargo también existen sistemas help desk de software libre que pueden ser implementados con facilidad pero están basados en procesos regidos y su código no es abierto por lo cual no se puede realizar mejoras o modificaciones, estas son las razones principales por las cuales desarrollar software a la medida es la mejor opción.

Sin embargo, es necesario estudiar a la competencia para poder analizar si vale la pena tomarse el tiempo para desarrollar, a continuación, se han tomado dos sistemas help desk uno pagado y uno de software libre para realizar la comparativa.

- Sistema Aplicación Centro de Asistencia CRM DESK

En la imagen siguiente (fig.3-42) se nos muestra la pantalla principal donde la información queda organizada en cinco pestañas: Inicio, Base de conocimientos, Preguntas, Clientes, Informes y Configuración.

En esta imagen se observa todos los ítems, cuales están abiertos y cuantos están cerrados, quienes son los que han generado los ítems, que ítems están asignados, etc.



Figura 3-3 Aplicación Centro de Asistencia CRM DESK

Fuente: El investigador

En la siguiente imagen (fig.3-3) se nos muestra la pantalla de búsqueda sobre la base de conocimiento (KB) –Frequent Asked Question FAQ , clasificada por Categorías (producto) ordenar por fecha de actualización.

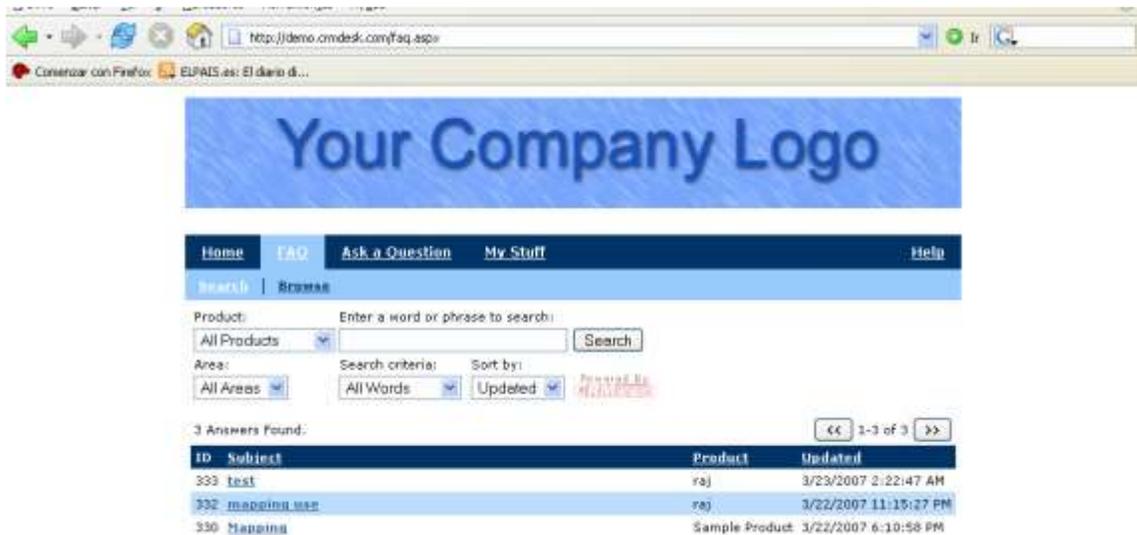


Figura 3-4 Base de conocimiento Aplicación Centro de Asistencia CRM DESK

Fuente: El investigador

Nótese que la base de conocimiento es solo una base de datos y el aplicativo solo puede ser implementado mas no presenta liberación de su código para realizar modificaciones al gusto del jefe de TI [19].

- Sistema Service desk plus

Esta solución está disponible on-premise o SaaS. Service desk plus está alineada con ITIL y cuenta con funcionalidades clave, entre las que se incluyen: Gestión de tickets, base de conocimiento, alineación con ITIL, Reportes y análisis, autoayuda, gestión de activos, soporte técnico y servicios de SLA's (medidores de rendimiento).

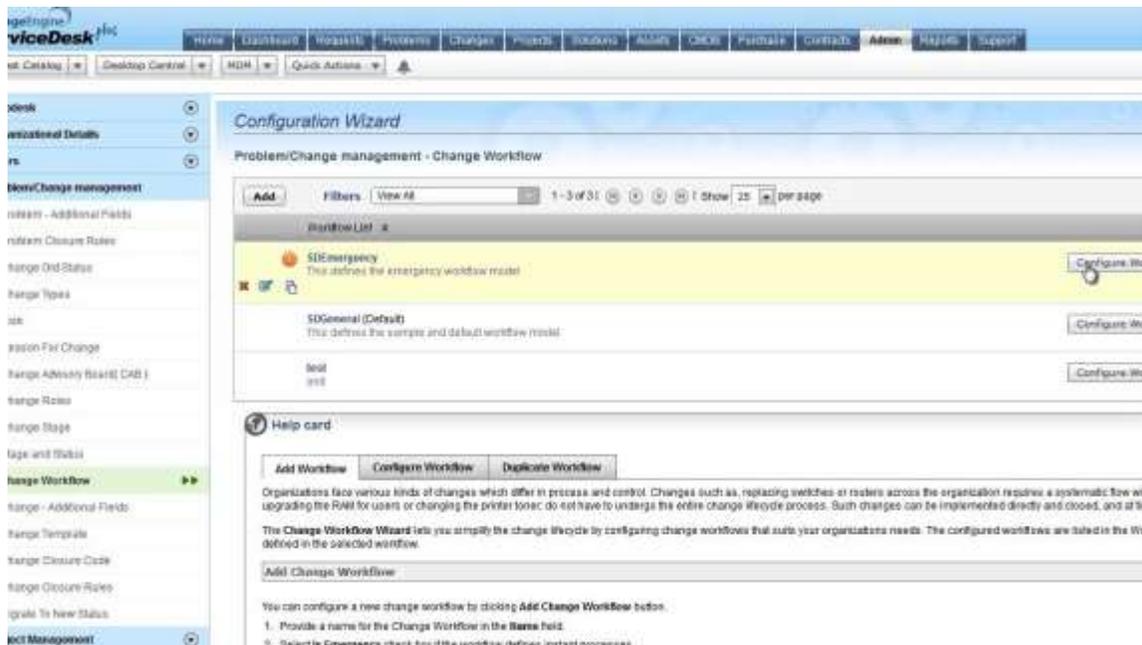


Figura 3–5 Sistema service desk plus

Fuente: El investigador

El sistema Service desk plus es una gran opción para tomar en cuenta, sin embargo su costo es de \$995 dólares americanos anuales, una cifra que no está a disposición para el proyecto help desk en la cooperativa Kullki Wasi además de tampoco contar con un código abierto para realizar modificaciones [20].

Para resumir algunos puntos podemos decir que:

Funcionalidades	Sistemas		
	Centro de Asistencia CRM DESK	Service desk plus	Help Desk Kullki Wasi
Alineado a ITIL	SI	SI	SI
Registro de incidentes	SI	SI	SI
Uso de categorías	SI	SI	SI
Criticidad de incidentes	NO	SI	SI
Reportes	NO	SI	SI
Base de conocimiento	SI	SI	SI
Base de datos para documentación	NO	NO	SI
Diseño de interfaz personalizable	NO	NO	SI
Código libre	NO	NO	SI
Costo de funcionamiento	NO	SI	NO
Lenguaje de desarrollo	ASP.NET	JAVA	ASP.NET

Tabla 3-6 Cuadro comparativo de sistemas help desk

Elaborado por: El investigador

3.5 Desarrollo del proyecto

- Analizar la información de la bitácora digital entregada por el agente de soporte técnico.
- Analizar los requerimientos necesarios para lograr obtener una interfaz de aplicación web amigable con el usuario.
- Diseñar las interfaces usando criterios de diseños web que sean acordes a la temática oficial de la Cooperativa Kullki Wasi.
- Implementar la aplicación web usando las herramientas de desarrollo permitidas oficialmente en la Cooperativa Kullki Wasi.
- Realizar las pruebas de integración y validación a la aplicación web para corregir o modificar algún aspecto de ser necesario.
- Realizar un informe final que valide el término de esta investigación.

CAPÍTULO 4

DESARROLLO DE LA PROPUESTA

Como parte esencial en la investigación está el uso de la metodología XP para el desarrollo de la propuesta, la metodología seleccionada nos permite realizar cambios sobre la marcha, lo cual resulta muy útil para un desarrollo veloz y receptivo a las modificaciones que puedan surgir.

4.1 Fase I: Exploración

En esta primera fase la investigación nos llevará a comprender la visión de la Cooperativa Kullki Wasi referentes a su propio proceso de soporte técnico, para lo cual se tomará información relevante acerca del “Instructivo de la Help Desk” elaborado por el Jefe de TIC de la Cooperativa Kullki Wasi y a su vez también se ha elaborado un par de preguntas sobre requerimientos adicionales en el aplicativo.

4.1.1 Entrevista realizada al Jefe de TIC de la Cooperativa Kullki Wasi

Con el fin de entender la visión del departamento de sistemas, se ha realizado preguntas concretas al Jefe de TIC, preguntas que se las pueden apreciar en la entrevista realizada al mismo, esta entrevista podemos revisarla en el anexo N°4.

Las preguntas realizadas en la entrevista nos dan una primera explicación de cómo solucionan las incidencias en el departamento de sistemas, las primeras conclusiones que podemos notar es que:

- Existen dos actores muy marcados empleados y agentes de soporte técnico.

- La segunda es que existe un proceso desarrollado y seguido por el agente de soporte técnico para poder realizar su trabajo de la manera más ordenada y oportuna
- Podemos mencionar que este proceso puede ser usado para obtener el esquema para desarrollar la base de datos que se usará en la aplicación Help Desk Kullki Wasi.

Una vez analizada la entrevista realizada al Jefe de TIC podemos manifestar que los objetivos claves a realizar y la información que se desea almacenar va a guardar mucha concordancia con los datos registrados antiguamente en la bitácora del agente de soporte técnico, para poder desarrollar un sistema totalmente intuitivo y personalizado, se tomará en cuenta en toda la información útil, así, la transición al nuevo proceso de registro de solicitudes no será un inconveniente.

4.2 Fase II: Planificación del proyecto

En esta parte vamos a comprender las reglas de negocio que vamos aplicar y las ideas de los usuarios referentes al sistema y al proceso de soporte técnico para posteriormente tomar de base este análisis y poder pasar a la siguiente fase del desarrollo.

4.2.1 Historias de usuario

Como pudimos notar en la fase uno del desarrollo tenemos dos actores principales los cuales son el empleado quien será el “Usuario” y el agente de soporte técnico quien será el “Administrador”, para poder establecer información útil se ha tomado en cuenta a varias personas y se a resumida sus respuestas en las siguientes historias de usuarios. Para la operatividad de los posibles módulos a desarrollar se está establecerá tiempos y criticidad para los módulos importantes o críticos.

A continuación, se observa la plantilla base que ha sido usada en las historias de usuario.

Historia de usuario	
Número:	Nombre:
Usuario:	
Puntos estimados:	Iteración Asignada:
Prioridad en negocios (alta-media-baja):	Riesgo en desarrollo (alto-medio-bajo):
Fecha inicio:	Fecha fin:
Descripción:	
Observaciones:	

Tabla 4-1 Base para Historias de Usuario

Elaborado por: El investigador

Esta plantilla consta de los siguientes elementos:

- **Número:** Es el identificador numérico de la historia del usuario.
- **Nombre:** Es el título que posee la historia del usuario, está relacionada con el caso descrito.
- **Usuario:** Es la persona que relata la historia.
- **Puntos estimados:** Es el tiempo estimado en días para concluir la historia, se trabaja solo con días enteros, si una historia tiene un día y medio inmediatamente se aproxima a dos días, con un máximo de 5 días para el desarrollo de cualquier módulo.
- **Iteración Asignada:** Es el número máximo de revisiones que va a tener el módulo antes de iniciar las pruebas, con un máximo de 5 revisiones, una por día si es posible.
- **Prioridad en Negocio (Alta/Media/Baja):** Se le asigna una calificación a la prioridad en base a la necesidad del desarrollo, esta prioridad se le da más importancia si tiene un tiempo de desarrollo alto.
- **Riesgo en Desarrollo (Alto/Medio/Bajo):** Se establece el valor del riesgo al momento de desarrollar el módulo, el riesgo crece entre más importante es la prioridad.
- **Fecha Inicio:** Fecha de inicio para iniciar el desarrollo.
- **Fecha Fin:** Fecha límite para entregar el desarrollo.
- **Descripción:** En esta sección va la historia del usuario en sus propias palabras. Estableciendo el que hacer y no el cómo hacer.
- **Observaciones:** Se detalla los procesos que tienen relación con dicha historia de usuario.

A continuación se muestra las historias de usuarios recolectadas con los empleados que interactuarán con el sistema Help Desk.

Historia de Usuario	
Número: 1	Usuario: Todos
Nombre de Historia: Bienvenida al sistema	
Prioridad de negocio: Baja	Riesgo en desarrollo: Baja
Puntos estimados: 1	Iteración asignada: 1
Fecha Inicio: 2017	Fecha Fin: 2017
Programador responsable: Alexander Ruiz	
Descripción:	
Esta es la pagina principal que verán todos los usuarios indistintamente de cual sea su rol en el sistema, en esta pagina se debe mostrar el acceso a la base de conocimiento y el acceso al sistema.	
Observaciones:	
No usar imágenes con personal en la pantalla principal, limitar el diseño de imágenes solo al logotipo de la institución.	

Tabla 4-2 Historia de Usuario Bienvenida al sistema

Elaborado por: El investigador

Historia de Usuario	
Número: 2	Usuario: Usuario
Nombre de Historia: Base de Conocimiento	
Prioridad de negocio: Media	Riesgo en desarrollo: Medio
Puntos estimados: 3	Iteración asignada: 3
Fecha Inicio: 2017	Fecha Fin: 2017
Programador responsable: Alexander Ruiz	
Descripción:	
La base de conocimiento debe contar con toda la información de los casos resueltos y documentación que ayude a los usuarios a solucionar los problemas semejantes o iguales	
Observaciones:	
Se debe clasificar los casos resueltos por categorias para poder guiar al usuario de manera más breve.	

Tabla 4-3 Historia de Usuario Base de conocimiento

Elaborado por: El investigador

Historia de Usuario	
Número: 3	Usuario: Usuario
Nombre de Historia: Directorio telefónico	
Prioridad de negocio: Baja	Riesgo en desarrollo: Bajo
Puntos estimados: 1	Iteración asignada: 1
Fecha Inicio: 2017	Fecha Fin: 2017
Programador responsable: Alexander Ruiz	
Descripción:	
Se espera contar con un directorio telefónico que facilite el número de extensión de los agentes de soporte técnico.	
Observaciones:	
Solo usar los números institucionales.	

Tabla 4-4 Historia de Usuario Directorio telefónico

Elaborado por: El investigador

Historia de Usuario	
Número: 4	Usuario: Todos
Nombre de Historia: Ingreso al sistema	
Prioridad de negocio: Media	Riesgo en desarrollo: Medio
Puntos estimados: 3	Iteración asignada: 3
Fecha Inicio: 2017	Fecha Fin: 2017
Programador responsable: Alexander Ruiz	
Descripción:	
El ingreso al sistema dependerá exclusivamente del rol que ocupe el usuario , en este caso puede ser Usuario o Administrador; los roles serán previamente definidos y se otorgará usuario y contraseña para diferenciar a las opciones a cumplir en el sistema.	
Observaciones:	

Tabla 4-5 Historia de Usuario Ingreso al sistema

Elaborado por: El investigador

Historia de Usuario	
Número: 5	Usuario: Usuario
Nombre de Historia: Registro de Caso	
Prioridad de negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 5	Iteración asignada: 5
Fecha Inicio: 2017	Fecha Fin: 2017
Programador responsable: Alexander Ruiz	
Descripción:	
Esta será la pantalla más crítica y deberá de contener las opciones pertinentes para poder describir el problema sucitado, se le permitirá al usuario solo ingresar ciertos parámetros como el título una descripción nivel de criticidad, tipo de caso y opcionalmente podrá adjuntar archivos relevantes	
Observaciones:	

Tabla 4-6 Historia de Usuario Registro de casos

Elaborado por: El investigador

Historia de Usuario	
Número: 6	Usuario: Administrador
Nombre de Historia: DashBoard o Menú	
Prioridad de negocio: Media	Riesgo en desarrollo: Media
Puntos estimados: 3	Iteración asignada: 3
Fecha Inicio: 2017	Fecha Fin: 2017
Programador responsable: Alexander Ruiz	
Descripción:	
El dashboard tendrá que tener las opciones necesarias para que el administrador pueda manejar todo el sistema.	
Observaciones:	

Tabla 4-7 Historia de Usuario Dashboard o menú

Elaborado por: El investigador

Historia de Usuario	
Número: 7	Usuario: Administrador
Nombre de Historia: Atender Caso	
Prioridad de negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 5	Iteración asignada: 5
Fecha Inicio: 2017	Fecha Fin: 2017
Programador responsable: Alexander Ruiz	
Descripción:	
Se debe recuperar los casos y toda su información aquí se deberá responder el caso y corregir la información brindada por el usuario con respecto al problema sucitado.	
Observaciones:	
Recuperar toda la información pero no modificar las fechas.	

Tabla 4-8 Historia de Usuario Atender Caso

Elaborado por: El investigador

Historia de Usuario	
Número: 8	Usuario: Administrador
Nombre de Historia: Subir Documentos	
Prioridad de negocio: Media	Riesgo en desarrollo: Media
Puntos estimados: 3	Iteración asignada: 3
Fecha Inicio: 2017	Fecha Fin: 2017
Programador responsable: Alexander Ruiz	
Descripción:	
El administradr deberá subir documentos tutoriales para la alimentar más la base de conocimiento.	
Observaciones:	
Estos documentos deberán ser siempre formato .pdf	

Tabla 4-9 Historia de Usuario Subir documentos

Elaborado por: El investigador

Historia de Usuario	
Número: 9	Usuario: Administrador
Nombre de Historia: Reportes	
Prioridad de negocio: Media	Riesgo en desarrollo: Media
Puntos estimados: 3	Iteración asignada: 3
Fecha Inicio: 2017	Fecha Fin: 2017
Programador responsable: Alexander Ruiz	
Descripción:	
Se debe elaborar una pantalla donde se pueda tener las opciones de reportes disponibles.	
Observaciones:	

Tabla 4-10 Historia de Usuario Reportes

Elaborado por: El investigador

Historia de Usuario	
Número: 10	Usuario: Administrador
Nombre de Historia: Formularios de mantenimiento	
Prioridad de negocio: Media	Riesgo en desarrollo: Media
Puntos estimados: 3	Iteración asignada: 3
Fecha Inicio:	Fecha Fin: 2017
Programador responsable: Alexander Ruiz	
Descripción:	
Es necesario poder ingresar nueva información a la base de datos sin necesidad de ingresar al gestor de base de datos, todas las tablas que guarden alguna relacion con el sistema deben poder tener su formulario de mantenimiento.	
Observaciones:	
Aquí se debe entender que tablas como usuarios, criticidad, estado , etc deben tener su propio formulario para poder ingresar nueva información .	

Tabla 4-11 Historia de Usuario Formularios de mantenimiento

Elaborado por: El investigador

4.2.2 Estimación de Historias de Usuario

Para la estimación de las historias de usuario cuya base son los días, se definen 4 horas de trabajo diario para la elaboración de la aplicación, se destacan los módulos que más esfuerzo tomarán realizar y así poder tener una referencia para módulos similares.

A continuación podremos apreciar las historias de usuarios planteadas para su análisis:

Módulo	N°	Historia de usuario	Tiempo estimado	
			Días	Horas
Registro de Usuarios	4	Ingreso al sistema	3	12
	10	Formularios de mantenimiento	3	12
Total de tiempo estimado:			6	24

Tabla 4-12 Estimación historias para Registro de Usuarios

Elaborado por: El investigador

Módulo	N°	Historia de usuario	Tiempo estimado	
			Días	Horas
Registro de casos	5	Registro de casos	5	20
	6	Dashboard o menú	3	12
	7	Atender casos	5	20
Total de tiempo estimado			13	52

Tabla 4-13 Estimación historia para Registro y solución de casos

Elaborado por: El investigador

Módulo	N°	Historia de usuario	Tiempo estimado	
			Días	Horas
Base de conocimiento	1	Bienvenido al sistema	1	4
	2	Base de conocimiento	3	12
	3	Directorio telefónico	1	4
	8	Subir documentos	3	12
	9	Reportes	3	12
Total de tiempo estimado			11	44

Tabla 4-14 Estimación historias para Base de Conocimiento

Elaborado por: El investigador

4.3 Fase III: Iteraciones

Las iteraciones nos van ayudar en tomar decisiones en cuanto al proceso de desarrollo y entrega de los módulos más importantes, para lo cual tendremos un plan de entrega en el que cada iteración será tomada como una revisión diaria antes del ingreso a las pruebas del módulo.

Módulo	N°	Historia de usuario	Iteraciones Revisiones	
			Iteraciones	Revisiones
Registro de Usuarios	4	Ingreso al sistema	1	1
	10	Formularios de ingreso	3	3
Registro de casos	5	Registro de casos	5	5
	6	Dashboard o menú	3	3
	7	Atender casos	5	5
Base de conocimiento	1	Bienvenida al sistema	1	1
	2	Base de conocimiento	3	3
	3	Directorio telefónico	1	1
	8	Subir documentos	3	3
	9	Reportes	3	3

Tabla 4-15 Iteraciones de los módulos

Elaborado por: El investigador

4.3.1 Plan de Entrega

Para la tercera fase, las historias de usuario nos darán un estimado del tiempo que necesitará cada una para su desarrollo. Los tiempos establecidos serán aceptados o modificados según acuerdos del cliente y el desarrollador. Se ha definido el tiempo por días laborales y las horas ya establecidas como tiempo de desarrollo para la aplicación.

Módulo	N°	Historia de usuario	Dias	Horas
Registro de Usuarios	4	Ingreso al sistema	1	4
	10	Formularios de ingreso	3	12
Registro de casos	5	Registro de casos	5	20
	6	Dashboard o menú	3	12
	7	Atender casos	5	20
Base de conocimiento	1	Bienvenida al sistema	1	4
	2	Base de conocimiento	3	12
	3	Directorio telefónico	1	4
	8	Subir documentos	3	12
	9	Reportes	3	12
Total			28	112

Tabla 4-16 Plan de entrega

Elaborado por: El investigador

En la presente fase, cada historia de usuario es representada y estimada por el tiempo en desarrollo de cada una. Estos tiempos se plantean de acuerdo a la comunicación entre el cliente final y desarrollador. Se definen en días de 4 horas de trabajo debido a disponibilidad de tiempo de la empresa.

4.4 Fase IV: Producción

Para iniciar la fase de producción fue necesario analizar todo el negocio y los actores involucrados en el mismo. Tal y como se puede comprobar en fases anteriores tenemos dos actores claramente marcados. Estos actores serán nuestros usuarios a los cuales se los puede agrupar en como: los empleados en este caso los llamaremos usuarios quienes registrarán los casos o incidentes y harán uso de la base de conocimiento y por otra parte los agentes de soporte técnico llamados a ser los administradores quienes serán los que resuelvan los casos o incidentes y mediante esta información alimentarán la base de conocimiento.

Para iniciar el desarrollo es necesario tener en cuenta el proceso actual de atención por parte del agente de soporte técnico.

4.4.1 Diseño del modelo de Atención Help Desk

Establecer un modelo de proceso de soporte técnico plantea varios factores a tomar en cuenta sin embargo la Cooperativa Kullki Wasi conjuntamente con su departamento de sistemas, han elaborado un proceso netamente pensado en todas las posibles implicaciones que puedan surgir en la labor del soporte técnico, tal como se establece en el Instructivo de la Help Desk.

Para poder realizar una aplicación Help Desk es necesario tener en cuenta las reglas de negocio, en este caso el proceso de soporte técnico, gracias a la pronta elaboración de este instructivo tan solo se tomará el proceso ya desarrollado y se trabajará con el mismo para plasmar la lógica de negocio en el sistema Help Desk Kullki Wasi.

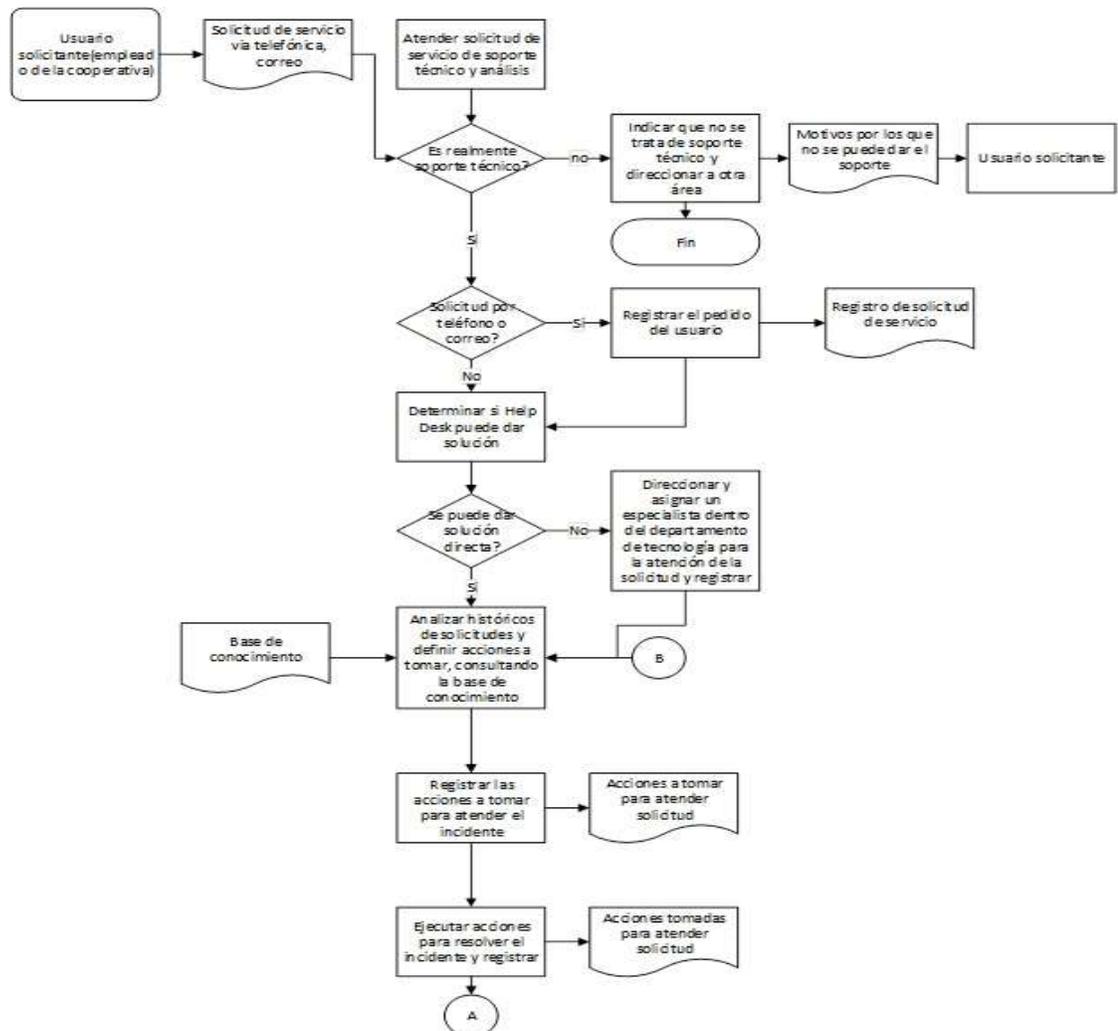


Figura 4-1 Proceso Help Desk
Fuente: Departamento TIC Kullki Wasi
Elaborado por: Gerencia General / Jefe de TIC

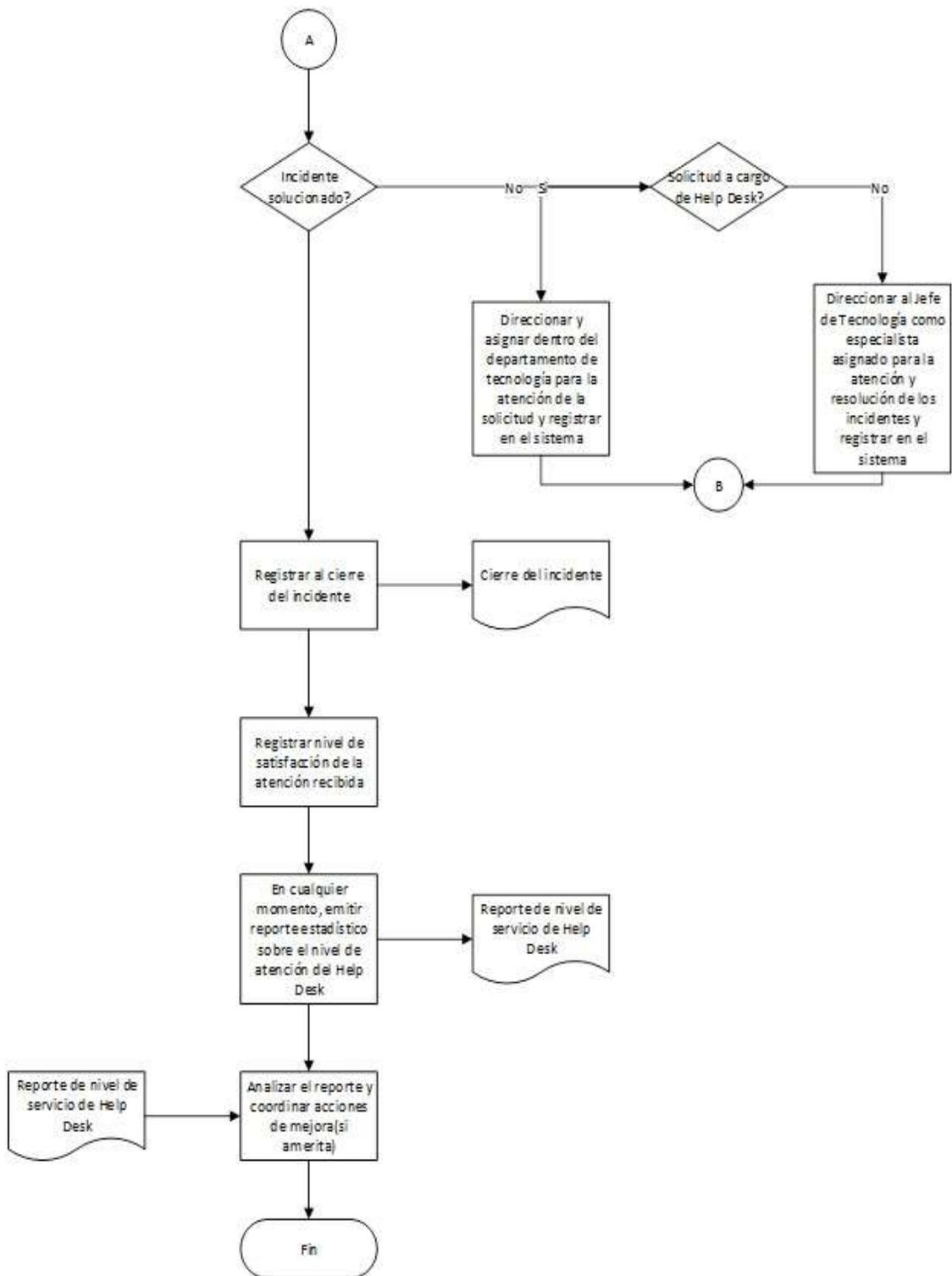


Figura 4-2 Proceso Help Desk

Fuente: Departamento TIC Kullki Wasi

Elaborado por: Gerencia General / Jefe de TIC

4.4.2 Análisis para el almacenamiento de la información

Una vez analizado el Instructivo de la Help Desk, verificamos el uso ordenado de la actividad soporte técnico, el cual posee lineamientos muy claros e instrucciones definidas para casi todos los tipos de situaciones a presentarse por parte de los usuarios, la definición de la criticidad en los tiempos de respuesta según la prioridad de algún problema o incidente, define con claridad el tiempo máximo de respuesta que posee cada agente para responder, como ya se analizó en el punto (3.4.4 .Análisis de criticidad y clasificación de problemas), ahora incluiremos los tiempos máximos de respuesta para cada incidente.

Niveles de prioridad	Tiempo de respuesta	Solución
Alto	30 minutos	2 horas
Medio	60 minutos	4 horas
Bajo	2 horas	6 horas

Tabla 4-17 Niveles de criticidad

Fuente: Departamento TIC Kullki Wasi

Elaborado por: Gerencia General / Jefe de TIC

Otro aspecto fundamental para no cambiar totalmente la forma de trabajar del agente de soporte técnico, es necesario analizar la información que normalmente registra en su bitácora de trabajo, varios de estos campos son acertados para poder llevarlos a uso en la aplicación y así simplemente mejorar la tediosa manera de llevar su bitácora y hacer de su tiempo más productivo y su trabajo menos agobiante.

Sujetándonos a la manera de trabajar y la forma de reportar los incidentes podemos también definir con claridad la escalabilidad de los casos, descarte de problemas y lo más importante la cultura que los empleados de la cooperativa han venido llevando en el tiempo que se han adoptado las normas.

Finalmente el análisis nos muestra una halagadora transición a la nueva forma de reportar los incidentes cotidianos ya que la aplicación luego de su implantación será una manera

mejorada de su actual proceso, proceso que a su vez será sujeto a pruebas de funcionamiento.

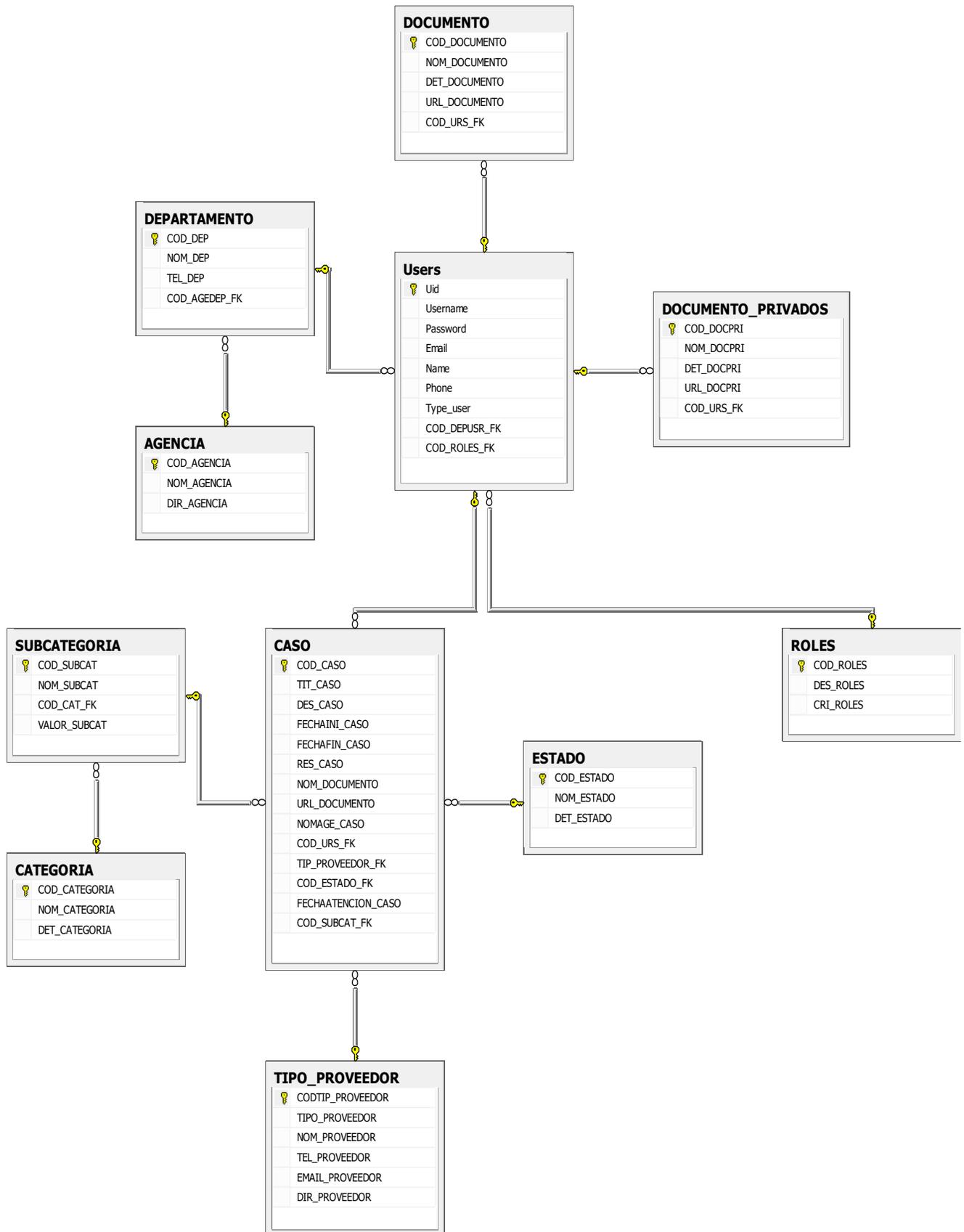
Al terminar de analizar los requisitos del sistema y la información a manejar en los formularios respectivos, se procede a crear las estructuras para las tablas con los datos que deberán ser almacenados, al mismo tiempo también se procederá a relacionar las tablas respectivas para posteriormente crear el esquema para la base de datos.

4.4.3 Diseño Base de datos

Para el diseño final de la base de datos a usar en la Aplicación Help Desk se llegó a un consenso entre el Jefe de TIC, el agente de soporte técnico y el investigador, en cuya reunión se tomaron todas las posibles opciones y variaciones para el sistema tanto como para escalamiento, almacenamiento de documentos, modificación de políticas actuales hasta la generación de reportes con la información almacenada.

Mediante toda la información recolectada el esquema para la base de datos final es el siguiente como podemos observar en la Figura 4-3 Modelo entidad relación.

La base de datos como se decidió en un principio será colocada en un servidor SQL Server 2008 R2 por motivos de aprovechamiento de recursos institucionales.



Fuente: El investigador

4.4.4 Desarrollo del módulo de ingreso al sistema Help Desk

Para empezar el desarrollo de uno de los módulos más críticos como es el de ingreso al sistema tomaremos la tabla de Usuarios en la cual crearemos el campo TIPE_USR el cual será el encargado de controlar la operatividad de acciones al usuario dentro del sistema.

Siendo en este caso un empleado este vendría a ser un “Usuario” el cual solo podrá tener acceso al módulo de registro de casos o a su vez un agente de soporte técnico el cual tomará el rol de “Administrador” en cual caso tendría a disposición el dashboard o menú de actividades en la cual podrá gestionar el sistema entero.

Diseño de interfaz

El siguiente modelo de interfaz es una idea conceptual de cómo puede lucir la pantalla de ingreso.

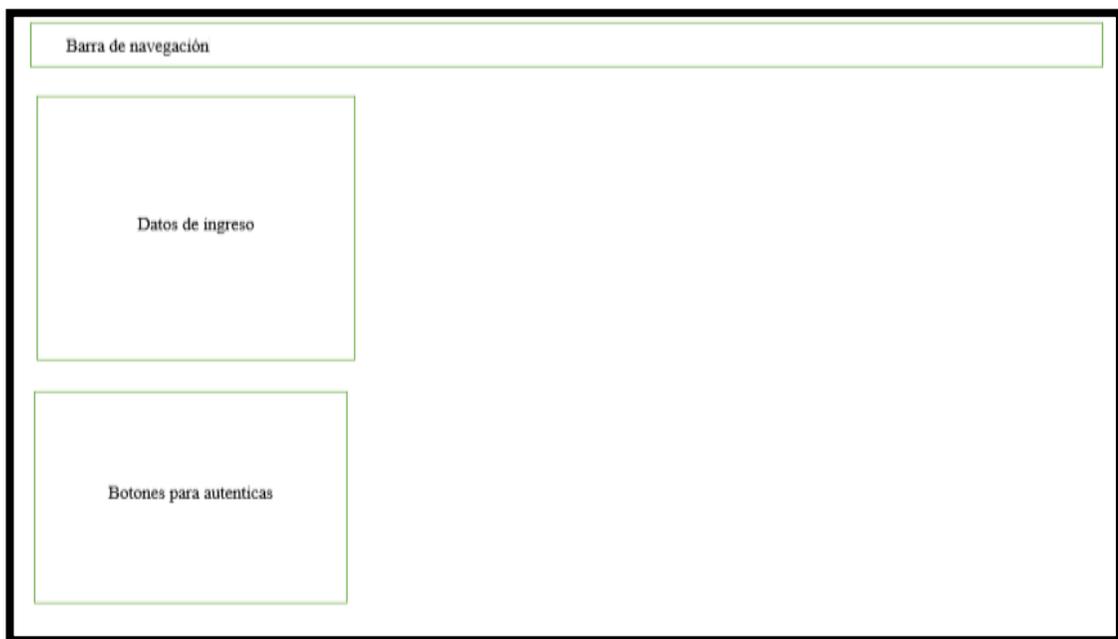
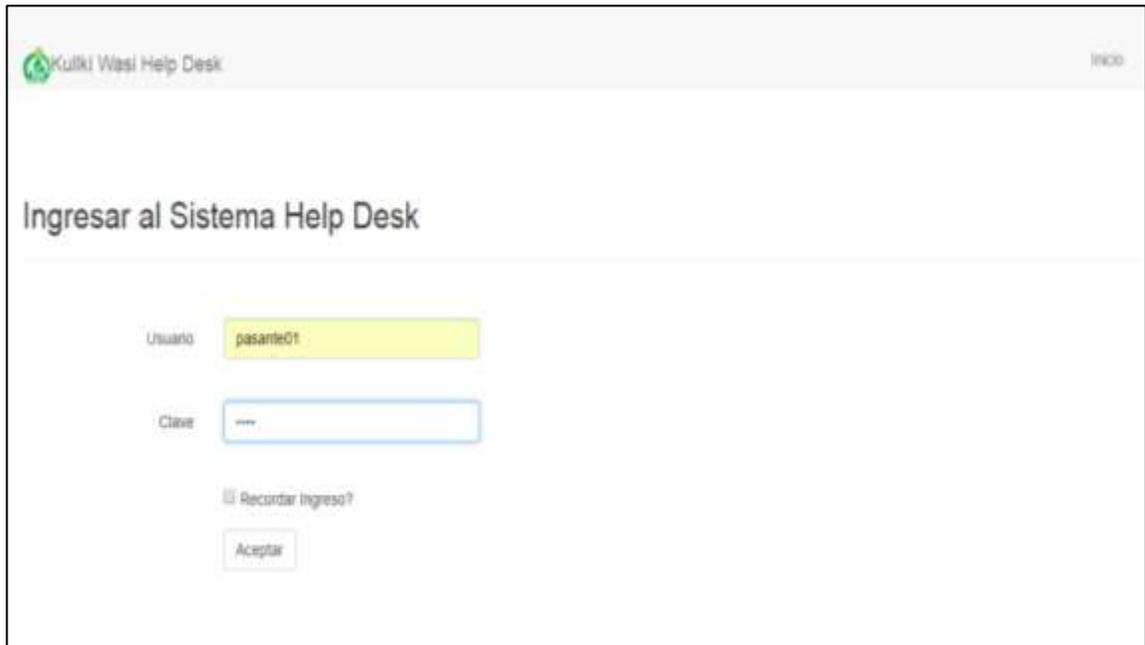


Figura 4-4 Modelo de Interfaz-Ingreso al sistema

Elaborado por: El investigador

Interfaz HTML para módulo ingreso al sistema

El diseño de la interfaz anterior se puede observar en la figura #, la cual cómo podemos observar es un ingreso muy sencillo e intuitivo.



The screenshot shows a web browser window with the title 'Kulki Wasi Help Desk' and a sub-header 'Ingresar al Sistema Help Desk'. Below the header, there is a login form with the following elements: a 'Usuario' label and a text input field containing 'pasante01'; a 'Clave' label and a password input field with three asterisks; a 'Recordar ingreso?' checkbox; and an 'Aceptar' button.

Figura 4-5 Interfaz-Ingreso al sistema

Elaborado por: El investigador

Código fuente para módulo ingreso al sistema

Para ingresar al sistema es necesario hacer una categorización previa y se lo hace de manera simple, mediante una consulta SQL que comparará los parámetros del tipo de usuario, algo muy importante debido a que las utilidades del sistema cambian totalmente según el tipo de usuario.

Como podemos observar al ser el usuario 'A' nos enviara al menú para usuario normal y al ser 'U' nos enviara al Dashboard del agente o administrador del sistema.

```
// Código para el ingreso al Sistema help desk  
  
protected void Button1_Click(object sender, EventArgs e)
```

```

        {
            String CS =
ConfigurationManager.ConnectionStrings["MyDatabaseConnectionString1"].Connect
ionString;

            using (SqlConnection con = new SqlConnection(CS))
            {
                SqlCommand cmd = new SqlCommand("select * from Users U,
AGENCIA A, DEPARTAMENTO D WHERE A.COD_AGENCIA=D.COD_AGEDEP_FK" +
                " AND
D.COD_DEP=U.COD_DEPUSR_FK AND U.Username='" + UserName.Text + "' and
U.Password='" + Password.Text + "'", con);

                con.Open();

                SqlDataAdapter sda = new SqlDataAdapter(cmd);

                DataTable dt = new DataTable();

                sda.Fill(dt);

                if (dt.Rows.Count != 0)
                {

                    if (CheckBox1.Checked)
                    {

                        Response.Cookies["UNAME"].Value = UserName.Text;

                        Response.Cookies["PWD"].Value = Password.Text;
                    }
                }
            }
        }

```

```

                Response.Cookies["UNAME"].Expires =
DateTime.Now.AddDays(15);

                Response.Cookies["PWD"].Expires =
DateTime.Now.AddDays(15);
            }
        else
        {
                Response.Cookies["UNAME"].Expires =
DateTime.Now.AddDays(-1);

                Response.Cookies["PWD"].Expires =
DateTime.Now.AddDays(-1);
        }

```

Figura 4-6 Código fuente para ingreso al sistema

Elaborador por: El investigador

El resultado final nos permite una validación muy precisa y de buena seguridad al hacer él envío de parámetros para realizar el ingreso contando a su vez con una validación desde interfaz para los datos.

Prueba de funcionamiento y aprobación de entrega

Para el completo cumplimiento de la metodología XP las pruebas de funcionamiento son un requisito indispensable, las pruebas nos permiten indagar sobre posibles errores o inconformidades que puedan surgir por alguna mal interpretación en las historias de usuarios o a su vez saber si lo implementado es lo que el cliente tenía en mente. Para este proceso de pruebas por petición del Jefe de TIC se pondrá a funcionamiento las iteraciones de las historias de usuario:

Número:	<ul style="list-style-type: none"> • 1
Historia de usuario:	<ul style="list-style-type: none"> • #4, Ingreso al sistema
Descripción:	<ul style="list-style-type: none"> • Se proporcionará la información de nombre de usuario y contraseña, se realiza la validación de existencia y concordancia de los datos proporcionados y se diferencia entre usuario y administrados para ingresar a la próxima pantalla.
Condiciones de ejecución:	<ul style="list-style-type: none"> • Conexión a la intranet. • Publicación de sitio web activa. • Datos de usuarios guardados en la base de datos.
Entradas:	<ul style="list-style-type: none"> • El usuario digita el nombre de usuario y contraseña. • EL usuario confirma su contraseña.
Resultados esperados:	<ul style="list-style-type: none"> • La validación de datos es correcta y el direccionamiento es acorde al tipo de usuario.

Evaluación de la prueba:	<ul style="list-style-type: none"> • Prueba satisfactoria.
--------------------------	---

Tabla 4-18 Prueba de funcionamiento

Elaborado por: El investigador

4.4.5 Desarrollo del módulo registro de casos

El módulo base para alimentar el sistema Help Desk es el módulo de registros de casos, este módulo es el resultado de un largo análisis que nos ha permitido obtener los valores más relevantes y útiles para el sistema.

Diseño de la interfaz

El modelo base para empezar este módulo es el siguiente.

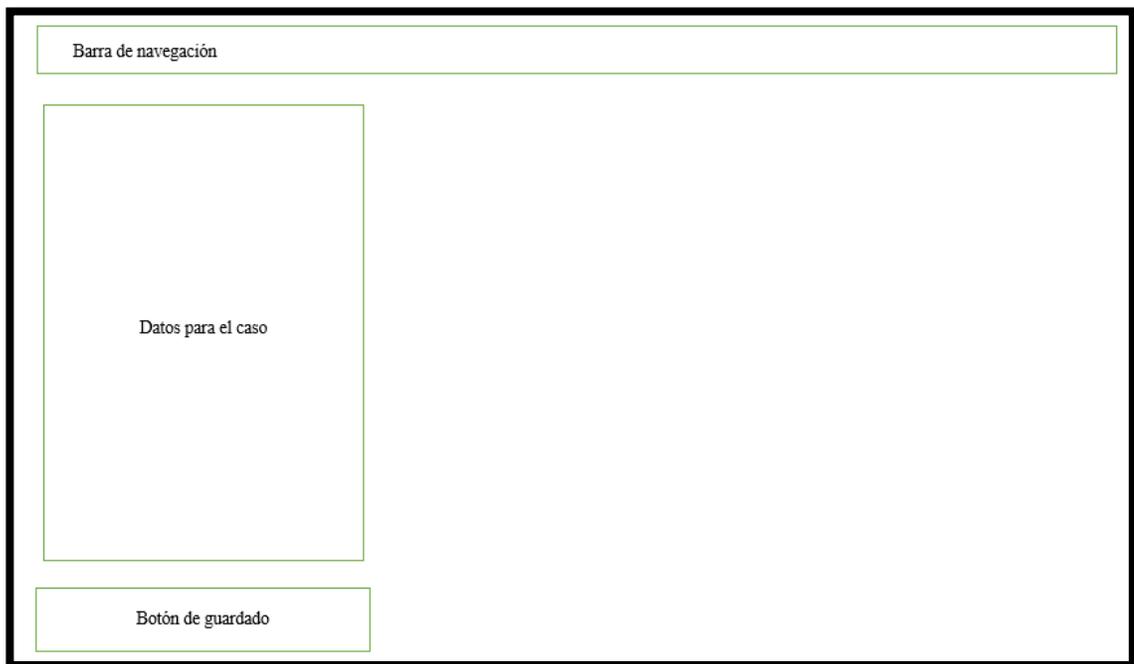


Figura 4-7 Modelo de Interfaz-Ingreso al registro y revisión de casos al sistema

Elaborado por: El investigador

Interfaz HTML para módulo registro de casos

El diseño de la interfaz anterior se puede observar en la figura #, la cual cómo podemos observar es un ingreso muy sencillo e intuitivo.

Titulo del Caso

Descripción del problema

Estado del Caso

NUEVO

Problema referente a:

Monitor

Fecha de Creacion del Caso

30/10/2017 11:42:06

Fecha de Vencimiento del Caso

31/10/2017 11:42:06

Desea Adjuntar algun archivo?

Adjuntar Archivo

Elegir archivos Ningún archivo seleccionado

Enviar

Figura 4–8 Interfaz de para el registro de casos.

Elaborado por: El investigador

Código fuente para el módulo registro de casos

Para el módulo de registro y actualización de casos es necesario ser muy cuidadosos en diferenciar cual información está expuesta a cambios y cual no, por esta razón se realiza esta actualización mediante procedimientos almacenados, como petición del Jefe de TIC, se ha establecido tomar la hora local del servidor para fecha de inicio del caso y aumentar un día a dicha fecha para establecer el periodo de vencimiento de cualquier caso.

```
//Envío de información mediante parámetros

        SqlConnection conn = new SqlConnection(CS);

        conn.Open();

        SqlCommand Query = new SqlCommand("GuardarCaso", conn);
```

```

        Query.CommandType = CommandType.StoredProcedure;

        Query.Parameters.Add("@TIT_CASO", SqlDbType.VarChar, 100);

        Query.Parameters["@TIT_CASO"].Value = TIT_CASO;

        Query.Parameters.Add("@DES_CASO", SqlDbType.VarChar, 500);

        Query.Parameters["@DES_CASO"].Value = DES_CASO;

        Query.Parameters.Add("@FECHAINI_CASO", SqlDbType.DateTime);

        Query.Parameters["@FECHAINI_CASO"].Value = System.DateTime.Now;

        Query.Parameters.Add("@FECHAFIN_CASO", SqlDbType.DateTime);

        Query.Parameters["@FECHAFIN_CASO"].Value=
System.DateTime.Now.AddDays(1);

        Query.Parameters.Add("@RES_CASO", SqlDbType.VarChar, 200);

        Query.Parameters["@RES_CASO"].Value = RES_CASO;

        Query.Parameters.Add("@NOM_DOCUMENTO",    SqlDbType.VarChar,
100);

        Query.Parameters["@NOM_DOCUMENTO"].Value = NOM_DOCUMENTO;

        Query.Parameters.Add("@URL_DOCUMENTO",    SqlDbType.VarBinary);

        Query.Parameters["@URL_DOCUMENTO"].Value = URL_DOCUMENTO;

        Query.Parameters.Add("@NOMAGE_CASO", SqlDbType.VarChar, 50);

        Query.Parameters["@NOMAGE_CASO"].Value = NOMAGE_CASO;

        Query.Parameters.Add("@COD_URS_FK", SqlDbType.Int);

        Query.Parameters["@COD_URS_FK"].Value = COD_URS_FK;

```

```

        Query.Parameters.Add("@TIP_PROVEEDOR_FK", SqlDbType.Int);
Query.Parameters["@TIP_PROVEEDOR_FK"].Value = TIP_PROVEEDOR_FK;

        Query.Parameters.Add("@COD_CATEGORIA_FK", SqlDbType.Int);
Query.Parameters["@COD_CATEGORIA_FK"].Value = COD_CATEGORIA_FK;

        Query.Parameters.Add("@COD_CRITICIDAD_FK", SqlDbType.Int);
Query.Parameters["@COD_CRITICIDAD_FK"].Value =
COD_CRITICIDAD_FK;

        Query.Parameters.Add("@COD_ESTADO_FK", SqlDbType.Int);
Query.Parameters["@COD_ESTADO_FK"].Value = COD_ESTADO_FK;

        Query.Parameters.Add("@Result", SqlDbType.Int, 4);
Query.Parameters["@Result"].Direction =
ParameterDirection.Output;

        Query.ExecuteNonQuery();

conn.Close();

```

Figura 4–9 Código fuente registro de casos

Elaborado por: El investigador

Prueba de funcionamiento y aprobación de entrega

Para el completo cumplimiento de la metodología XP las pruebas de funcionamiento son un requisito indispensable, las pruebas nos permiten indagar sobre posibles errores o inconformidades que puedan surgir por alguna mal interpretación en las historias de usuarios o a su vez saber si lo implementado es lo que el cliente tenía en mente. Para este proceso de pruebas por petición del Jefe de TIC se pondrá a funcionamiento las iteraciones de las historias de usuario:

Número:	<ul style="list-style-type: none"> • 2
Historia de usuario:	<ul style="list-style-type: none"> • #5 - Registro de casos
Descripción:	<ul style="list-style-type: none"> • Se proporciona solo los campos estimados por el usuario para hacer el registro más eficiente estos campos que se ingresaran a través de textbox son: título del caso, descripción del caso, y se seleccionara de una lista el nivel de criticidad y categoría del caso al final se puede proporcionar más detalle mediante el ingreso de archivo adjunto (opcional).
Condiciones de ejecución:	<ul style="list-style-type: none"> • Conexión a la intranet • Publicación de sitio web activa • Ingreso con datos de usuario
Entradas:	<ul style="list-style-type: none"> • El usuario digita el título del caso y la descripción del problema. • El usuario selecciona de las listas el nivel de criticidad y la categoría. • El usuario adjunta un archivo guía para el problema (opcional).
Resultados esperados:	<ul style="list-style-type: none"> • Los datos son enviados con éxito a la base de datos.

Evaluación de la prueba:	<ul style="list-style-type: none"> • Prueba satisfactoria.
--------------------------	---

Tabla 4-19 Prueba de Funcionamiento

Elaborado por: El investigador

4.4.6 Desarrollo del módulo base de conocimiento

Como aclaración válida, la base de conocimiento está pensada a un futuro ya que por el momento no es necesaria la inferencia en los datos ya que el sistema no cuenta con información suficiente por lo cual será solo un banco de datos que almacene información, cuando el departamento de sistemas crea necesario se implementará la base de conocimiento, pero por motivos de terminología y de documentación se estableció con el Jefe de TIC en nombrar al banco de datos como base de conocimiento.

Una vez aclarada la terminología este será un módulo muy importante en la función didáctica del sistema Help Desk, este módulo está pensado para poder desarrollar las capacidades investigativas de los usuarios, el módulo se alimentará día a día de los casos solucionados por los agentes de soporte técnico los cuales a su vez también podrán subir documentos oficiales para poder orientar mejor al personal de la Cooperativa Kullki Wasi.

Diseño de interfaz

El esquema inicial para la interfaz del módulo base de conocimiento cuenta con una tabla que contendrá todos los casos registrados además de controles de búsqueda para hacer más fácil la experiencia investigativa, al lado derecho podremos ver el acceso a los documentos oficiales.



Figura 4–10 Modelo de Interfaz-Base de Conocimiento

Elaborado por: El investigador

Diseño HTML para el módulo base de conocimiento

Uno de los principales beneficios del sistema Help Desk Kullki Wasi es la autosuficiencia que puede desarrollar en los usuarios al acceder a la base de conocimiento para lo cual se ha implementado una búsqueda de similitud para cualquier palabra que ingrese el usuario y el acceso claro a los documentos oficiales, cuyo módulo también contará con una búsqueda para hacer más fácil su uso.



Figura 4–11 Interfaz para el ingreso al sistema

Elaborado por: El investigador

Código fuente para el módulo base de conocimiento

Para poder desarrollar la base de conocimiento no hizo falta crear una tabla en la base de datos ni enviar parámetros para registrar ninguna información, como se explicó anteriormente la base de conocimiento es un acumulador de actividades resueltas por parte del personal del área de sistemas.

Tal y como podremos observar en el siguiente código.

```
//Recuperación de datos para la consulta

public DataTable CargarGrid()
{
    //Declarar e instanciar un "Connection".
    SqlConnection conexion = new SqlConnection();
    DataTable tablaCategorias = new DataTable();
    conexion.ConnectionString = CS;
```

```
String sql = "select COD_CASO, TIT_CASO, RES_CASO FROM CASO WHERE
COD_ESTADO_FK=3 AND COD_CATEGORIA_FK<3";
```

Figura 4–12 Código fuente para la base de conocimiento

Elaborado por: El investigador

Prueba de funcionamiento y aprobación de entrega

Para cumplir con la metodología XP es necesario realizar la prueba de funcionamiento del módulo y si el cliente a si decide establecer como entrega exitosa el módulo.

Número:	<ul style="list-style-type: none"> • 3
Historia de usuario:	<ul style="list-style-type: none"> • #2, Base de conocimiento
Descripción:	<ul style="list-style-type: none"> • Se debe recuperar los casos solucionados por los administradores del sistema para que pueda ser usado por los usuarios.
Condiciones de ejecución:	<ul style="list-style-type: none"> • Conexión a la intranet • Publicación de sitio web activa • Datos recuperados con éxito
Entradas:	<ul style="list-style-type: none"> • El administrador da como solucionado cualquier caso. • El usuario ingresa a la base de conocimiento. • El usuario realiza una búsqueda en la tabla • El usuario puede acceder a revisar la solución anterior del caso en cuestión

	<ul style="list-style-type: none"> • El usuario ingresa a ver documentos oficiales. • El usuario realiza una búsqueda de documentos.
Resultados esperados:	<ul style="list-style-type: none"> • Solo los casos solucionados con estado cerrado son mostrados en la base de conocimiento. • Los documentos oficiales se muestran correctamente. • Las búsquedas dinámicas muestran información coherente como resultado.
Evaluación de la prueba:	<ul style="list-style-type: none"> • Prueba satisfactoria.

Tabla 4-20 Prueba de Funcionamiento

Elaborado por: El investigador

Al término de la fase 4 de la metodología XP se procedió a realizar la implementación de todo el sistema, las 3 historias de usuario documentadas resultaron ser las claves para el funcionamiento de todo el sistema Help Desk Kullki Wasi.

CAPÍTULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

- El instructivo Help Desk elaborado por el departamento de sistemas ayudó para poder realizar la categorización de incidentes que puedan surgir, además al analizar las bitácoras de registro se pudo comprender el trabajo que realiza un agente de Soporte Técnico en una institución financiera y lo importante que es tecnificar este proceso de registro para optimizar este recurso humano tan valioso.
- Poder aplicar la metodología de desarrollo ágil XP permitió realizar cambios rápidos en sistema Help Desk Kullki Wasi, logrando modificar de manera sencilla aspectos de lógica y diseño en la aplicación, lo que facilitó la elaboración del software.
- El desarrollo web gracias al uso de Bootstrap fue ligero y nada agobiante, al usar el diseño responsivo en la elaboración de las pantallas, las diferentes resoluciones en los dispositivos electrónicos no son un problema del cual preocuparse.

5.2 Recomendaciones

- Para un óptimo desempeño del personal en el uso de la aplicación Help Desk Kullki Wasi se recomienda socializar el proceso establecido en el “Instructivo Help Desk”, así el personal podrá tener claro el registro de casos o incidentes que puedan acontecer.
- Otro aspecto importante es el otorgar credenciales a los empleados, por lo cual se recomienda implementar la herramienta de Windows Active Directory lo antes posible para que todos los empleados puedan acceder al aplicativo con toda normalidad, ya que por el momento se realizó la implementación con los empleados que contaban con credenciales en el sistema Financial Business System de la empresa SifizSoft.
- Se recomienda también contratar más personal para poder resolver a la brevedad los casos de las agencias lejanas, una vez sea implantado el sistema en todas las agencias.
- Se recomienda realizar respaldos de la base de datos mensualmente como mantenimiento del sistema para evitar cualquier pérdida de información.
- Para un mejor uso del sistema a nivel operativo, se recomienda designar un agente administrador global del sistema, para que sea el encargado de modificar y registrar la información de los casos resueltos y así ganar eficiencia en el personal de soporte técnico.

BIBLIOGRAFÍA

- [1] Cooperativa de ahorro y crédito Kullki Wasi, «kullkiwasi,» 2013. [En línea]. Available: <http://www.kullkiwasi.com.ec/index.php/nosotros>. [Último acceso: 07 2017].
- [2] Jan van Bon, Arjen de Jong, Axel Kolthof, Mike Pieper, Ruby Tjassing, Annelies van der Veen, Tienneke Verheijen, Fundamentos de ITIL V3, Van Haren Publishing, Tercera Edición, 2010.
- [3] S. Luján Mora, Programación de Aplicaciones Web: Historia, principios básicos y clientes web, España: Club Universitario, 2012.
- [4] M. A. Contreras Castañeda, Desarrollo de Aplicaciones Web Multiplataforma, España: Aula Mentor, 2013.
- [5] M. J. García Casermeiro, Técnicas de información y atención al cliente/consumidor, México: Vértice, 2014.
- [6] R. Hawthorne, Desarrollo de bases de datos en Microsoft SQL Server 2000 desde el principio, México: PEARSON EDUCACIÓN, 2002.
- [7] M. Snell y L. Powers, Microsoft Visual Studio 2012 Unleashed, Indiana: PEARSON EDUCACIÓN, 2012.
- [8] C. Aubry, HTML5 y CSS3 para sitios con Diseño Web Responsive, Barcelona: eni, 2014.
- [9] M. Arias, Webs Responsivas. Responsive Design con Bootstrap, México: PEARSON, 2014.

- [10] L. Chuburu, JQuery, Conceptos y Definiciones, <<Jquery>>. [En línea]. Available: <https://jquery.com/>. [Último acceso: 07 2017].
- [11] Á. Arias, Aprende a programar ASP.NET y C#, México: IT, 2015.
- [12] B. Guérin, ASP .NET en C# con Visual Studio 2013, Barcelona: ENI, 2014.
- [13] W. Stanek, The Personal Trainer for IIS 8.0 and IIS 8.5, Stanek & Associates, 2015.
- [14] L. Crispin y T. House, Testing Extreme Programming, Boston: Addison-Wesley, 2013.
- [15] C. P. Patricio Letelier, «Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP),» 2006.
- [16] Jeffries R., Anderson A., Hendrickson C, Extreme Programming Installed, Addison-Wesley, 2001.
- [17] K. Beck, Extreme Programming Explained. Embrace Change, Pearson Education, 2000.
- [18] Héctor Acevedo Juárez, ITIL: ¿qué es y para qué sirve?, Magazciturum 2013.
- [19] Crmdesk, Aplicación demo CRMDESK, ForeSoft Corporation, 2017.on-line. <http://www.crmdesk.com/demo.htm>
- [20] ManageEngine, Sistema Service Desk plus, Zoho Corp, 2017. online <https://www.manageengine.com>