



UNIVERSIDAD JOSÉ ANTONIO PÁEZ

**SOFTWARE DE GESTIÓN ÁGIL DE  
PROYECTOS UTILIZANDO ISO TOOLS  
FUNDAMENTALES**

**Autor:**

Andres Bravo

**Tutor:** MSc. Jetro López

Urb. Yuma II, calle N° 3. Municipio San Diego  
Teléfono: (0241) 8714240 (master) – Fax: (0241) 8712394



REPÚBLICA BOLIVARIANA DE VENEZUELA  
UNIVERSIDAD JOSÉ ANTONIO PÁEZ  
FACULTAD DE INGENIERÍA  
ESCUELA DE COMPUTACIÓN  
CARRERA INGENIERÍA DE COMPUTACIÓN

**SOFTWARE DE GESTIÓN ÁGIL DE PROYECTOS UTILIZANDO ISO  
TOOLS FUNDAMENTALES**

Proyecto del Trabajo de Grado para optar al título de

**INGENIERO DE COMPUTACIÓN**

**Autor:**

Andres Bravo

C.I. 26.389.379

**Tutor:** MSc. Jetro López

San Diego, Julio de 2020

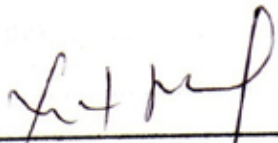


**REPÚBLICA BOLIVARIANA DE VENEZUELA  
UNIVERSIDAD JOSÉ ANTONIO PÁEZ  
FACULTAD DE INGENIERÍA  
ESCUELA DE COMPUTACIÓN**

**CONSTANCIA DE APROBACIÓN PARA LA PRESENTACIÓN PÚBLICA  
DEL TRABAJO DE GRADO**

Quien suscribe, MSc. Jetro López, portador de la cédula de identidad N.º 8.779.723, en mi carácter de tutor del trabajo de grado presentado por el ciudadano Andrés A. Bravo R., portador de la cédula de identidad N.º 26.389.379, titulado **SOFTWARE DE GESTIÓN ÁGIL DE PROYECTOS UTILIZANDO ISO TOOLS FUNDAMENTALES**, presentado como requisito parcial para optar al título de Ingeniero en Computación, considero que dicho trabajo reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del jurado examinador que se designe.

En San Diego, a los nueve días del mes de julio del año dos mil veinte

  
\_\_\_\_\_  
**MSc. Jetro López**  
**C.I.: 8.779.723**



**FI-C -017-2020-1CR (TG)**

Valencia, 19 de junio de 2020

Ciudadano:

**Bravo R., Andrés A.**

**C.I. 26.389.379**

Presente-

Cumplo con informarle que la Comisión de Trabajo de Grado y Pasantías de la Facultad de Ingeniería en su reunión N° 04-2020 de fecha 13-02-2020 aprobó el proyecto de trabajo de grado titulado **SOFTWARE DE GESTIÓN ÁGIL DE PROYECTOS UTILIZANDO ISO TOOLS FUNDAMENTALES** presentado por usted como requisito para optar al título de Ingeniero en Computación.

Se ratifica la designación del Ing. Jetro López, C.I. 8.779.723 como Tutor Académico que lo asesorará en el desarrollo de este proyecto.

Atentamente,



Prof. Luís Lira

**Decano de la Facultad de Ingeniería**

c.c. Coordinación de Pasantías y Trabajo de Grado (1).

Ll/a.a.

## **AGRADECIMIENTOS**

A mis padres Víctor y Elizabeth, quienes me guiaron y me dieron aliento para seguir adelante, además de brindarme todo su apoyo de forma incondicional.

A mi hermano Víctor, por brindarme la posibilidad de haber estudiado. De igual modo a mis hermanas Rebeca y Endrina quienes estuvieron al pendiente durante mi estadía en la universidad.

A mi abuela Miriam, mis tías Miriam y Virginia y mi primo Lenin, quienes con sus palabras de aliento me dieron motivación para continuar haciendo lo que me gusta.

A mi novia Valeria por ser mi compañera de vida y mi mayor impulso para lograr todas mis metas.

A todos los compañeros que a lo largo de la carrera me ayudaron de algún modo u otro, sobre todo a Estefania Gainza quien me brindó su apoyo incondicional y me dio muchas enseñanzas.

A los profesores Belkys Araujo, Oneida Jiménez y a mi tutor Jetro López por impartir sus conocimientos y brindarme asistencia durante toda la carrera.

A todos mis amigos, familiares y demás profesores, muchas gracias.

## DEDICATORIA

*A mis padres Víctor Bravo y Elizabeth Raga, por todo su amor, comprensión y enseñanzas.*

*A mis hermanos, quienes son un ejemplo a seguir y muestras de unas personas admirables.*

*A mis familiares por apoyarme y confiar en mí en todo momento.*

*A Valeria Reyes por ser una persona increíble que me ha ayudado y apoyado en todo.*

*A mi amiga Estefania Gainza y demás amigos y familiares por sus palabras de aliento.*

# ÍNDICE GENERAL

## CONTENIDO

<b>RESUMEN</b> .....	¡Error! Marcador no definido.
<b>INTRODUCCIÓN</b> .....	¡Error! Marcador no definido.
<b>CAPÍTULO I</b> .....	¡Error! Marcador no definido.
<b>I EL PROBLEMA</b> .....	¡Error! Marcador no definido.
1.1 Planteamiento del problema.....	¡Error! Marcador no definido.
1.2 Formulación del problema .....	¡Error! Marcador no definido.
1.3 Objetivos de la investigación .....	¡Error! Marcador no definido.
1.3.1 Objetivo General.....	¡Error! Marcador no definido.
1.3.2 Objetivos Específicos .....	¡Error! Marcador no definido.
1.4. Justificación de la investigación.....	¡Error! Marcador no definido.
1.5. Alcance.....	¡Error! Marcador no definido.
<b>CAPÍTULO</b>	
<b>II</b> .....	¡Error!
Marcador no definido.	
<b>II MARCO TEÓRICO</b> .....	¡Error! Marcador no definido.
2.1. Antecedentes .....	¡Error! Marcador no definido.
2.2. Bases teóricas.....	¡Error! Marcador no definido.
2.2.1 ¿Qué es un proyecto?.....	¡Error! Marcador no definido.
2.2.2 Gestión de Proyectos .....	¡Error! Marcador no definido.
2.2.3 Métodos de Gestión de Proyectos....	¡Error! Marcador no definido.
2.2.3.1 PMBOK .....	¡Error! Marcador no definido.
2.2.3.2 Modelo Canvas	
.....	¡Error! Marcador no definido.
2.2.4 Administración Ágil de Proyectos...	¡Error! Marcador no definido.
2.2.5 Metodologías Ágiles .....	¡Error! Marcador no definido.
2.2.6 Ciclos de vida adaptativos .....	¡Error! Marcador no definido.
2.2.6.1 Scrum.....	¡Error! Marcador no definido.

2.2.6.2 Kanban .....	<b>¡Error! Marcador no definido.</b>
2.2.6.3 XP Programming .....	<b>¡Error! Marcador no definido.</b>
2.2.7 Organización Internacional de la Estandarización (ISO) .....	<b>¡Error! Marcador no definido.</b>
2.2.8 ISOTools .....	<b>¡Error! Marcador no definido.</b>
2.2.9 Ventajas de ISOTools .....	<b>¡Error! Marcador no definido.</b>
2.2.10 Diseño Web Responsive .....	<b>¡Error! Marcador no definido.</b>
2.2.11 Ventajas del Diseño Web Responsive .....	<b>¡Error! Marcador no definido.</b>
2.2.12 Base de Datos (MySQL).....	<b>¡Error! Marcador no definido.</b>
2.3 Bases Legales .....	<b>¡Error! Marcador no definido.</b>
2.4 Definición de términos básicos .....	<b>¡Error! Marcador no definido.</b>

## **CAPÍTULO**

**III.....¡Error!**

Marcador no definido.

<b>III MARCO METODOLÓGICO.....</b>	<b>¡Error! Marcador no definido.</b>
3.1. Tipo y nivel de la investigación .....	<b>¡Error! Marcador no definido.</b>
3.2. Diseño de la investigación .....	<b>¡Error! Marcador no definido.</b>
3.3. Población y muestra .....	<b>¡Error! Marcador no definido.</b>
3.3.1. Población .....	<b>¡Error! Marcador no definido.</b>
3.3.2. Muestra .....	<b>¡Error! Marcador no definido.</b>
3.4. Técnicas e instrumentos de recolección de datos.....	<b>¡Error! Marcador no definido.</b>
3.5. Fases metodológicas .....	<b>¡Error! Marcador no definido.</b>

## **CAPÍTULO**

**IV.....¡Error!**

Marcador no definido.

<b>IV ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS .....</b>	<b>¡Error!</b>
Marcador no definido.	

4.1 Fase I: Planificación.....	¡Error! Marcador no definido.
4.1.1 Entrevista .....	¡Error! Marcador no definido.
4.1.2 Análisis de los resultados.....	¡Error! Marcador no definido.
4.1.3 Requerimientos funcionales y no funcionales ..	¡Error! Marcador no definido.
<b>definido.</b>	
4.2 Fase II: Diseño del sistema .....	¡Error! Marcador no definido.
4.2.1 Casos de Uso.....	40
4.2.2 Modelado de la base de datos .....	42
4.2.3 Diccionario de datos .....	43
4.3 Fase III: Desarrollo .....	59
4.3.1 Desarrollo del sistema.....	59
4.3.2 Diseño de interfaces.....	59
4.4 Fase IV: Pruebas .....	83
<b>CAPÍTULO V.....</b>	<b>86</b>
<b>V CONCLUSIONES Y RECOMENDACIONES.....</b>	<b>86</b>
5.1 Conclusiones .....	86
5.2 Recomendaciones .....	87
<b>REFERENCIAS .....</b>	<b>¡Error! Marcador no definido.</b>

## ÍNDICE DE FIGURAS

Figura 1: Caso de uso del rol administrador .....	41
Figura 2: Caso de uso del rol gerente .....	41
Figura 3: Caso de uso del rol programador .....	42
Figura 4: Modelado de datos .....	43
Figura 5: Pantalla Principal Login.....	60
Figura 6: Pantalla Reestablecer Contraseña. ....	60
Figura 7: Pantalla Principal Login Incorrecto .....	61
Figura 8: Pantalla Principal Rol Desarrollador .....	61
Figura 9: Pantalla Principal Rol Administrador .....	62
Figura 10: Pantalla Editar Perfil.....	62
Figura 11: Menú de acceso a opciones: Perfil, Proyectos, Clientes/Productos, Usuarios, Ajustes, Contáctanos, Salir .....	63
Figura 12: Pantalla Proyectos.....	63
Figura 13: Pantalla Crear Nuevo Proyecto.....	64
Figura 14: Pantalla Editar Detalles de Proyecto.....	64
Figura 15: Pantalla Editar Archivos de Proyecto .....	65
Figura 16: Pantalla Discusión de Proyecto.....	65
Figura 17: Pantalla Clientes/Productos .....	66
Figura 18: Pantalla Crear Nuevo Cliente/Producto .....	66
Figura 19: Pantalla Editar Cliente/Producto.....	67
Figura 20: Pantalla Usuarios .....	67
Figura 21: Pantalla Añadir Usuario.....	68
Figura 22: Pantalla Editar Usuario .....	68
Figura 23: Pantalla Eliminar Usuario.....	69
Figura 24: Pantalla Ajustes de la Empresa.....	69
Figura 25: Pantalla Dashboard. Actividades, Gráficas, Avances, Backlog, Metas, Reporte	70
Figura 26: Pantalla Actividades Pendientes. Modo Tablero Kanban.....	70
Figura 27: Pantalla Actividades Pendientes. Modo Lista.....	71

Figura 28: Pantalla Metas. Vista por Mes .....	71
Figura 29: Pantalla Añadir Meta .....	72
Figura 30: Pantalla Control de Horas del Proyecto .....	72
Figura 31: Pantalla Plan. Vista Simple Canvas .....	73
Figura 32: Pantalla Añadir Plan .....	73
Figura 33: Pantalla Editar Plan .....	74
Figura 34: Pantalla Plan. Vista Lean Canvas .....	74
Figura 35: Pantalla Eliminar Tablero de Plan .....	75
Figura 36: Pantalla Ideas. Vista Muro de Ideas.....	75
Figura 37: Pantalla Ideas. Vista Tablero Kanban.....	76
Figura 38: Pantalla Crear Tablero de Ideas .....	76
Figura 39: Pantalla Editar Tablero de Ideas .....	77
Figura 40: Pantalla Eliminar Tablero de Ideas .....	77
Figura 41: Pantalla Retrospectiva.....	78
Figura 42: Pantalla Crear Retrospectiva.....	78
Figura 43: Pantalla Editar Retrospectiva.....	79
Figura 44: Pantalla Eliminar Tablero de Retrospectiva .....	79
Figura 45: Pantalla Auditorias.....	80
Figura 46: Pantalla Activos .....	80
Figura 47: Pantalla Amenazas .....	81
Figura 48: Pantalla Empresas .....	81
Figura 49: Pantalla Integraciones: Slack, Mattermost, Zulip.....	82
Figura 50: Pantalla Control de Horas Personales .....	82
Figura 51: Pantalla Calendario .....	83
Figura 52: Pantalla Nuevo Evento.....	83

## ÍNDICE DE TABLAS

Tabla 1: Pregunta de Entrevista 1 .....	34
Tabla 2: Pregunta de Entrevista 2.....	35
Tabla 3: Pregunta de Entrevista 3.....	35
Tabla 4: Pregunta de Entrevista 4.....	36
Tabla 5: Pregunta de Entrevista 5.....	36
Tabla 6: Pregunta de Entrevista 6.....	37
Tabla 7: Pregunta de Entrevista 7.....	37
Tabla 8: Pregunta de Entrevista 8.....	37
Tabla 9: Pregunta de Entrevista 9.....	38
Tabla 10: account .....	44
Tabla 11: action_tabs.....	44
Tabla 12: action_tabs.....	44
Tabla 13: assets .....	45
Tabla 14: audits .....	45
Tabla 15: availability.....	45
Tabla 16: calendar .....	46
Tabla 17: canvas .....	46
Tabla 18: canvas_items .....	46
Tabla 19: clients .....	47
Tabla 20: comment.....	48
Tabla 21: confidentiales .....	48
Tabla 22: dashboard_widgets.....	48
Tabla 23: file .....	49
Tabla 24: gcallinks .....	49
Tabla 25: impacts .....	49
Tabla 26: integrities.....	49
Tabla 27: lead .....	50
Tabla 28: messages.....	50
Tabla 29: modulerights.....	50
Tabla 30: note.....	51

Tabla 31: probabilities .....	51
Tabla 32: projects .....	51
Tabla 33: punch_clock .....	51
Tabla 34: read .....	52
Tabla 35: relationuserproject .....	52
Tabla 36: risk_level .....	52
Tabla 37: roles .....	52
Tabla 38: settings .....	53
Tabla 39: sprints .....	53
Tabla 40: stats .....	54
Tabla 41: statuses .....	54
Tabla 42: submodulerights .....	55
Tabla 43: threats .....	55
Tabla 44: threat_type .....	55
Tabla 45: tickethistory .....	55
Tabla 46: tickets .....	57
Tabla 47: timesheets .....	57
Tabla 48: user .....	58
Tabla 49: wiki .....	58
Tabla 50: wiki_articles .....	58
Tabla 51: wiki_categories .....	59
Tabla 52: wiki_comments .....	59

REPÚBLICA BOLIVARIANA DE VENEZUELA  
UNIVERSIDAD JOSÉ ANTONIO PÁEZ  
FACULTAD DE INGENIERÍA  
ESCUELA DE COMPUTACIÓN  
CARRERA INGENIERÍA DE COMPUTACIÓN

## **SOFTWARE DE GESTIÓN ÁGIL DE PROYECTOS UTILIZANDO ISO TOOLS FUNDAMENTALES**

**Autor:** Andres Bravo

**Tutor:** MSc. Jetro López

**Fecha:** Febrero de 2020

### **RESUMEN**

El presente trabajo de grado plantea el desarrollo de un software de gestión ágil de proyectos mediante el uso de ISOTools fundamentales las cuales proporcionan herramientas tecnológicas para gestionar proyectos de una manera que sea compatible con el uso de metodologías ágiles, beneficiándose así de las ventajas que éstas poseen. De igual modo, se tiene como finalidad darles solución a problemáticas presentes en las organizaciones. Estas problemáticas se ven reflejadas principalmente debido a la falta de flexibilidad en los cambios, poca cohesión en el grupo y la cantidad de variables externas e internas presentes en el desarrollo. Para abordar estas problemáticas se hace uso de mapas de procesos, gestión de auditorías, hallazgos y un dashboard intuitivo. La investigación se basa en un diseño de investigación de campo y nivel descriptivo. Así mismo, se implementarán instrumentos de recolección de datos mediante la utilización de observación directa y entrevista no estructurada, lo que representa un aporte importante para la investigación. Para el desarrollo del software, se propuso el uso de la metodología de desarrollo de software XP (Programación Extrema), esta comprende cuatro fases, el análisis o planificación en la cual se define el propósito del software, sus objetivos específicos y el alcance del mismo, los principales riesgos. La segunda fase corresponde a el diseño, donde se identifica y valida la arquitectura de software. El desarrollo o codificación como tercera fase, incluye todo el proceso de modelado y desarrollo del software. Para finalizar, se explican la fase de las pruebas las cuales se realizan al software para determinar que el funcionamiento sea óptimo, adecuado y libre de errores.

Descriptores: gestión ágil de proyectos, software de gestión, ISOTools.

## INTRODUCCIÓN

El presente trabajo de investigación busca desarrollar un software de gestión de proyectos que se pueda implementar fácilmente en empresas, ya sean pequeñas, medianas o grandes empresas. De igual modo, se busca implementar en él, la capacidad de abordar proyectos de manera intuitiva con metodologías ágiles. Esto aporta beneficios cuantificables tanto para los involucrados como para la empresa.

Muchas veces los equipos de desarrollo de proyectos no cuentan con los recursos necesarios para la implementación de sistemas de gestión sofisticados, por lo que tienen que utilizar métodos sencillos y mayormente basados en la intuición, aún cuando no sean eficientes o sean defectuosos. Hoy en día, existen muchas alternativas en cuanto a sistemas de gestión, pero ninguna se adapta completamente al proyecto, por lo tanto, pocas veces terminan satisfaciendo las necesidades y objetivos de la empresa.

Por este motivo, las empresas buscan constantemente mecanismos que optimicen los recursos que están asociados a los proyectos (equipos, materiales, horas, etc.). Es por esto que se debe dar una mirada a empresas innovadoras a nivel global que han desarrollado sistemas que solucionan necesidades reales y por esta razón se puede extrapolar para que esa innovación sea aplicada en el diseño del software de gestión ágil.

Es muy importante desde la perspectiva interna de las organizaciones tener un software de gestión que sea robusto, flexible a cambios, y que controle riesgos. Y que, además, gestione el conocimiento, cree una cultura de colaboración y cohesión entre integrantes del proyecto y promueva la innovación y cambio organizacional.

De forma similar para las organizaciones, la automatización e integración de procesos ha marcado un antes y un después, los sistemas de información están orientados a dar respuesta a las necesidades empresariales y se han convertido en la forma de optimizar y perfeccionar todas las actividades que requieran de un alto

poder procesamiento de tareas, pues esto permite que la información esté siempre disponible, y con ello que las decisiones tomadas sean más acertadas. La tecnología se ha vuelto necesaria e indispensable, por los beneficios que ofrecen, tales como: la disminución de errores, la velocidad de procesamiento, control de variables, agilidad en cambios y un sinnúmero de ventajas que pueden ser utilizadas para enfrentar la competencia.

El propósito general de esta investigación es diseñar un software de gestión ágil de proyectos que sea capaz de administrar a los mismos mediante metodologías ágiles, con el fin de que los tiempos de respuestas sean efectivos, dar solución a las problemáticas existentes en las organizaciones, entre otras cosas que afectan directamente el desarrollo de los proyectos.

De este modo, la estructura del proyecto de investigación se divide en cuatro (4) capítulos, los cuales describen lo siguiente:

**Capítulo I**, donde se describe la problemática planteada a solucionar con el proyecto, la justificación de su aplicación, los objetivos deseables a cumplir (tanto específicos como generales) y el alcance que tendrá.

**Capítulo II**, donde se mencionan los antecedentes que guardan relación con este trabajo de investigación, de igual modo se explican los conceptos y teorías que se utilizan en la realización del proyecto y que sustentan al mismo, así como también la definición de términos básicos. Por último, se hace referencia a las bases legales.

**Capítulo III**, se describe el marco metodológico de la investigación, el cual consiste en especificar el tipo, nivel y diseño de la misma, así como también la definición de las fases y los instrumentos de recolección de datos para el cumplimiento de los objetivos.

**Capítulo IV**, en este capítulo se especifican los recursos y el diagrama de actividades que explica la distribución del tiempo a emplear para el cumplimiento de cada uno de los objetivos.

# CAPÍTULO I

## EL PROBLEMA

### 1.1. Planteamiento del problema

En un mundo cada vez más interconectado, nuestra capacidad para compartir opiniones e información parece ilimitada. Sin embargo, nuestra capacidad para comunicar nuestras ideas de manera concisa y que estas sean correctamente entendidas, parece más limitada que nunca. Los gerentes de proyectos tienen el importante papel de asimilar una cantidad de datos técnicos (gráficas, mapas de proceso, auditorías, modelos computarizados y muestras), liderar el proceso de convertir ideas y conceptos en un producto final concreto, que el mismo sea funcional y que, además, cumpla con los requisitos acordados con el cliente.

Estos gerentes también deben enfrentarse a un grupo variado de personas que intervienen en el proceso de desarrollo del proyecto, lo cual engloba una serie de variables que pueden influir tanto de forma positiva como de manera negativa en la evolución del mismo.

Estas múltiples variables que influyen en el proceso de desarrollo de proyectos pueden llegar a convertirse en una cantidad considerable, por lo que es pertinente tener un plan a seguir mientras está en marcha este proceso. Por ejemplo, un problema bastante común en los proyectos suele ser la cantidad de personas involucradas en el mismo; muchas veces no se tiene la certeza que estas personas puedan comunicarse efectivamente entre sí, y compartir ideas, datos o información relevante al proyecto. Además, muchas de las veces no se fomenta una cohesión del grupo, por lo que la cooperación entre ellos no fluye de la manera esperada.

De igual forma, se puede encontrar dificultad en los plazos de entrega de los proyectos. Esta imposibilidad de cumplimiento de los tiempos acordados es una de las causas principales de los problemas que surgen en los proyectos. Cabe destacar que también se pueden encontrar plazos de entrega muy cortos. Similarmente, otra situación de riesgo de plazos es que, aun teniendo plazo suficiente, surgen retrasos en

una fase determinada. Esto implica una reestructuración del plan de gestión y, quizá, alargar los plazos de entrega. Todas estas situaciones complican la entrega efectiva del proyecto, así como disminuyen la satisfacción del cliente y por supuesto, sugieren un incremento en la carga de trabajo, lo cual genera principalmente, frustración y disminución de los niveles de motivación.

Cada equipo de trabajo también se enfrenta a un entorno en rápida evolución, ya sea por los requisitos cambiantes de los sistemas o las necesidades de los clientes. Además de esto, se enfrentan al reto de aumentar continuamente su productividad al tiempo que mantiene y/o mejora la calidad. Sin embargo, existen diferencias entre los desafíos a los que se enfrentan los equipos en las organizaciones más pequeñas, en comparación con los de las organizaciones más grandes. Cuanto más grande es una organización, mayor es la complejidad, lo que implica que se debe cumplir en cada proyecto en lo que respecta a las funciones de control externo y a los procesos estandarizados que están predefinidos.

Además, si los requisitos dados al equipo de desarrollo son abstractos, la necesidad de entender al cliente aumenta con la realidad de no conocer muy bien a las organizaciones de los clientes, igualmente complejas, o incluso de tener la capacidad de comunicarse frecuentemente con el cliente. En paralelo, un pre-estudio detallado requerirá de mucho tiempo, sobre todo si se tiene en cuenta la gran complejidad que implica, y es posible que en el momento en que se finalice ya el mismo esté desfasado.

Es por ello que surge la necesidad de organizar, motivar y controlar los recursos de los proyectos, tratando de reducir así los riesgos que este implica, de modo que se puedan alcanzar los objetivos. Por esta razón, se han generado múltiples propuestas mediante diferentes estrategias y herramientas para encontrar un mejor procedimiento en el proceso de trabajo. Una de estas propuestas es la adaptación de metodologías de desarrollo ágil, éstas generalmente se basan en la transparencia del trabajo, lo que significa que todos los aspectos relevantes del proceso de desarrollo deben ser visibles para todos los responsables de su resultado. Los miembros del

equipo en entornos ágiles intercambian mucha información sobre el progreso de su trabajo individual a diario y utilizan herramientas como los tableros ágiles para fomentar la transparencia. Comunicar y documentar el progreso del trabajo de los involucrados, es también un aspecto de metodologías ágiles.

De este modo, la gestión de los proyectos proporciona beneficios tales como: la eficiencia en los plazos de entrega de los proyectos, aumento en la satisfacción de los clientes, mejora en la unión y desarrollo del equipo, mayor ventaja competitiva, mejora en la flexibilidad a la hora de abordar proyectos y mayor control de los riesgos. Es por esto que surgen otras herramientas útiles para la gestión de los proyectos. ISOTools es una de ellas, pues aporta resultados en el corto plazo para las problemáticas anteriormente planteadas. Desde optimizar el tiempo de resolución de una acción correctiva, menor tiempo en la preparación de las reuniones de gestión, hasta la reducción del tiempo dedicado a recopilar y tratar indicadores técnicos.

En tal sentido, las empresas también deben ser capaces de implementar estas metodologías ágiles en su sistema de gestión, que el mismo se mantenga actualizado y que el sistema mejore continuamente. Por lo que deberán lograr una mayor agilidad y optimización en la gestión de los procesos, al igual que permitir un fácil acceso, así como una rápida búsqueda de la información. Mediante el uso de herramientas ágiles e ISOTools todo esto se puede lograr, debido a que estos permiten a las organizaciones tener una gestión automatizada, dando respuestas a sus necesidades reales, así como, ayuda a éstas a la consecución de su compromiso con la mejora continua y excelencia en el desarrollo de los proyectos, facilitando así, la fluida comunicación tanto a nivel interno como externo, así como el trabajo de forma colaborativa.

Por esta razón, se plantea el desarrollo de un software de gestión ágil de proyectos que utilizando herramientas ISOTools permita a los proyectos tener un crecimiento progresivo de las funcionalidades sin ver afectado ningún otro parámetro. De igual manera, este software de gestión permitirá realizar el seguimiento adecuado para los diferentes proyectos en ejecución, proporcionando mapas de procesos,

riesgos y oportunidades, gestión de auditorías, hallazgos y un dashboard que permitirá controlar todas estas funcionalidades, además de la visualización de los diferentes proyectos que se estén llevando a cabo. Todo esto operando bajo un desarrollo ágil o inclusive, algún híbrido entre metodologías tradicionales y ágiles.

## **1.2. Formulación del problema**

Tomando en cuenta la problemática anteriormente puntualizada, se plantea el desarrollo de un software de gestión ágil de proyectos para la adquisición de la información en tiempo real, lo que permite visualizar los avances de los proyectos, flexibilizar los tiempos de entrega y reducir los riesgos, aumentando así las oportunidades de éxito que se conjugan entre los gerentes de proyectos y equipos de trabajo. De acuerdo a lo anteriormente expuesto, se plantea la siguiente interrogante: ¿Cómo mejorar los procesos y la productividad de los proyectos haciendo uso de la gestión ágil?

## **1.3. Objetivos de la investigación**

### **1.3.1. Objetivo General**

Desarrollar un software de gestión ágil de proyectos haciendo uso de ISOTools Fundamentales: Mapa de proceso, riesgos y oportunidades, gestión de auditorías, hallazgos, dashboard y gestión documental, aumentando así el éxito en el desarrollo de los proyectos.

### **1.3.2. Objetivos Específicos**

- Determinar los requerimientos funcionales y no funcionales del software, por medio de la aplicación de técnicas de recolección de datos.
- Diseñar el software de gestión ágil de proyectos con ISOTools Fundamentales, mediante la utilización de las herramientas de modelado UML.
- Desarrollar un software de gestión ágil de proyectos para la administración de los procesos empresariales haciendo uso de las herramientas ISOTools, permitiendo así la mejora continua y mantenimiento progresivo del software.

- Construir un plan de pruebas al software que asegure el funcionamiento óptimo del sistema y determine errores.

#### **1.4. Justificación de la investigación**

El mercado actual de software es altamente competitivo y la tecnología es muy cambiante. Y en el desarrollo del software se pide básicamente rapidez, calidad y reducción de costes. Pero para asumir estos retos es necesario tener agilidad y flexibilidad. Los ciclos de desarrollo, por un lado, acostumbran a ser largos, y lo que se exige por otra parte, es que esos ciclos sean lo más cortos posibles.

Por ello, el presente estudio se respalda con los fundamentos teóricos de la gestión de proyectos y se justifica la aplicación tanto de herramientas ágiles como de ISOTools debido a que estas permiten la agilidad y flexibilidad que se dan en un ciclo constante de reuniones diarias que involucra al recurso humano involucrado en el proceso de desarrollo de proyectos. Además, se considera que tanto organizaciones grandes como pequeñas podrían incrementar su productividad en el proceso de desarrollo de proyectos.

El desarrollo del software de gestión ágil de proyectos, permite administrar ágilmente los proyectos que puedan ser complejos. Esto permite que las herramientas para un equipo de trabajo sean muy útiles, pues facilitan encontrar una solución a la problemática. Asimismo, la función de la gestión de proyectos, asegura que todas las personas involucradas estén en sintonía con respecto a los requisitos, la priorización de los objetivos, los presupuestos y el cronograma, así como también de la retroalimentación recibida por parte del cliente.

Por último, la implementación de ISOTools Fundamentales permite la facilidad de uso ya que esta herramienta es ante todo intuitiva y usable, es decir, permite un fácil manejo de la misma. Incorpora funcionalidades de roles potentes, al igual que de personalización. Por otro lado, se tiene flexibilidad con total independencia del tamaño que presente la organización en cuestión, o sea cual sea el sector en el que la misma opere, se adaptará a las características de cada una de ellas,

configurándose de acuerdo a sus particularidades a fin de poder satisfacer sus necesidades. Además, es una herramienta que dispone de una amplia variedad de funcionalidades y características para dar respuesta a diferentes necesidades de las organizaciones, facilitando así la toma de decisiones ya que da una visión globalizada del conjunto de la organización.

### **1.5. Alcance**

El presente trabajo de grado tiene como alcance el desarrollo de un software ágil de proyectos, el cual consiste en una herramienta para estructurar, planificar y controlar el proceso de desarrollo y gestión de proyectos. Cabe destacar que esta herramienta estará diseñada bajo las mejores prácticas adoptadas del desarrollo de la Ingeniería de Software y bajo el uso de las metodologías ágiles adaptadas a los requerimientos de las empresas en términos de dirección de proyectos.

Se empleará los lenguajes de programación, PHP, y Javascript por la parte del servidor. Del mismo modo, se utilizará del lado del cliente el framework de estilos de cascada Bootstrap, así como otras librerías que incluye Chart.js, entre otras. Asimismo, se utilizará el gestor de base de datos que sirve como puente de comunicación entre las bases de datos y la interfaz, conocido como MySQL. Todo esto permitirá aplicar las mejoras técnicas de desarrollo durante el ciclo de vida de la aplicación de una manera dinámica, obteniendo así mejores resultados.

## **CAPÍTULO II MARCO TEÓRICO**

### **2.1. Antecedentes**

Para este trabajo de investigación ha sido importante y necesario explorar investigaciones previas que compartan similitudes con el presente estudio, de este modo, se cuenta con un respaldo teórico y se justifica la factibilidad de aplicación del mismo. A continuación, se presentan las investigaciones anteriormente mencionadas.

Acosta, D. (2017) realizó una investigación para la Universidad del Norte, Barranquilla, República de Colombia, titulada **Diseño de una metodología de gestión de proyectos basada en PMO, con el fin de elevar la productividad de esta área**, donde se establece como tema principal que la gestión de proyectos juega un papel importante en las organizaciones de hoy en día, ya que permite lograr eficientemente los cambios necesarios para mejorar procesos, actualizar sistemas, diseñar productos y, en algunas ocasiones, permanecer en el mercado.

La relación que guarda este estudio con la presente investigación se explica mediante las ventajas que aporta la aplicación de una metodología en un sistema de gestión de proyectos y como se puede proceder para que se logre de la manera más eficiente y organizada posible los cambios necesarios para mejorar procesos, actualizar sistemas, diseñar productos y, en algunas ocasiones, permanecer en el mercado.

Asimismo, García, G. (2017) presentó a la Universidad Técnica Federico Santa María, Valparaíso, Chile, su trabajo de grado titulado, **Diseño de sistema de gestión de proyectos para Pymes**, donde se busca diseñar un sistema de gestión de proyectos para pequeñas y medianas empresas que permite optimizar los recursos utilizados en cada proyecto que se realice, además se busca que la implementación del sistema sea fácil de ejecutar en las Pymes.

Además, la citada investigación contribuye en el estudio tanto de las actuales plataformas mundiales de gestión de proyectos como de aquellas con aplicaciones computacionales, así como las últimas que permiten la utilización de teléfonos

inteligentes y redes sociales. De este modo, también se busca que el sistema cuente con una metodología intuitiva que genere beneficios cuantificables en el corto y mediano plazo, así como la optimización de recursos asociados a los proyectos.

Por último, Hung, M. (2015), realizó un trabajo especial de grado para obtener el título de Especialista en Sistemas de información en la Universidad Católica Andrés Bello, Caracas, República Bolivariana de Venezuela, titulado **Diseño de una metodología de desarrollo de software basada en metodología ágil Scrum y las mejores prácticas de la gerencia de proyecto**, donde se establecieron las mejores prácticas que se pueden implementar en el desarrollo de software, el autor realizó una prueba piloto para concluir que cada proyecto es único y que aplicar una metodología al pie de la letra, no se adapta a la realidad, pues los imprevistos y cambios en los requerimientos de los proyectos obligan a que los procesos sean flexibles.

En este trabajo el autor integró los elementos de la metodología ágil y del PMI y los adaptó a los requerimientos de las empresas, por tal motivo se utilizará como referencia al momento de establecer el procedimiento para identificar las prácticas aplicables para automatizar la gestión ágil de proyectos, dando respuesta a uno de los objetivos planteados en la presente investigación.

## **2.2. Bases teóricas**

Las bases teóricas exponen cómo se sustenta teóricamente la presente investigación, es por ello que se deben aclarar conceptos e ideas que son importantes para el entendimiento de la misma. A continuación, se explicarán los conceptos referidos anteriormente. Cabe destacar que los conceptos que se muestran a continuación están basados en las definiciones de Naybour y Association of Project Management North West Branch. (Naybour, y otros, 2015).

### **2.2.1 ¿Qué es un proyecto?**

Es un modelo de planificación y desarrollo que consiste en un conjunto de actividades que se encuentran relacionadas y coordinadas, para alcanzar un objetivo que puede incluir presupuesto, calidad y tiempo. De allí se desprende el significado

de planificación la cual es un proceso metódico para también alcanzar un objetivo. Por otro lado, el proceso es la actividad o evento que se realiza de manera alternativa o simultánea con un fin determinado.

### **2.2.2 Gestión de Proyectos**

La gestión de proyectos es la aplicación de conocimiento, metodologías, técnicas, habilidades y herramientas para la definición, planificación y realización de actividades, con el objetivo de transformar objetivos o ideas en realidades. De forma general, se puede considerar a la gestión de proyectos como una aproximación sistemática y estructurada a cómo las organizaciones gestionan sus actividades y permiten cumplir con los requisitos del proyecto de principio a fin.

La persona encargada de garantizar la sinergia entre todas las partes que se interrelacionan en el proyecto es el director de proyecto, quien, mediante herramientas de gestión como la planificación, ejecución, seguimiento, control y cierre, es responsable de alcanzar los objetivos que se han establecido definido para el proyecto. Dicho esto, con la gestión de proyectos se busca logra realizar el mejor desempeño en:

- Costos, calidad, plazos y alcance.
- Necesidades identificadas.
- Expectativas de los involucrados.

### **2.2.3 Métodos de Gestión de Proyectos**

#### **2.2.3.1 PMBOK**

El PMBOK es una guía en el que se presentan estándares, pautas y normas para la gestión de proyectos. Para el PMBOK una buena forma de tratar los procesos es agrupándolos en áreas de conocimiento. Un área de conocimiento representa un conjunto completo de los conceptos, términos, y las actividades que componen un campo profesional, campo de la gestión de proyectos, o área de especialización. Se identifican diez áreas de conocimiento:

- Integración

- Alcance
- Tiempo
- Costes
- Calidad
- Recursos Humanos
- Comunicaciones
- Riesgo
- Adquisiciones
- Interesados

Según García, G. (2016). Las diez áreas de conocimiento que se utilizan en la mayoría de los proyectos. PMBOK no es una metodología, es un conjunto de buenas prácticas, en todo momento el equipo escoge la metodología, así como el conjunto de procesos apropiados para el proyecto.

### **2.2.3.2 Modelo Canvas**

Un modelo de negocio describe las bases sobre las que una empresa crea, proporciona y capta valor. Según Osterwalder A. y Pigneur Y. (2010), los nueve módulos son:

1. Segmentos de mercado: Una empresa atiende a uno o varios segmentos de mercado.
2. Propuestas de valor: Su objetivo es solucionar los problemas de los clientes y satisfacer sus necesidades mediante propuestas de valor.
3. Canales: Las propuestas de valor llegan a los clientes a través de canales de comunicación, distribución y venta.
4. Relaciones con clientes: Las relaciones con los clientes se establecen y mantienen de forma independiente en los diferentes segmentos de mercado.
5. Fuentes de ingresos: Las fuentes de ingresos se generan cuando los clientes adquieren las propuestas de valor ofrecidas.
6. Recursos clave Los recursos clave son los activos necesarios para ofrecer y

proporcionar los elementos antes descritos.

7. Actividades clave: mediante una serie de actividades clave.

8. Asociaciones clave: Algunas actividades se externalizan y determinados recursos se adquieren fuera de la empresa.

9. Estructura de costes: Los diferentes elementos del modelo de negocio conforman la estructura de costes.

En contraste, el Business Model Canvas lleva el siguiente orden:

1. Segmentos de clientes

2. Propuesta de valor

3. Canal

4. Relación con clientes

5. Flujos de ingresos

6. Recursos clave

7. Actividades clave

8. Alianzas

9. Estructura de costos

#### **2.2.4 Administración Ágil de Proyectos**

El desarrollo ágil o la administración ágil de proyectos es un método iterativo e incremental para administrar las actividades de diseño y construcción en cuanto a proyectos de ingeniería, tecnología de la información y desarrollo de nuevos productos o servicios se refiere, por ejemplo, el desarrollo de software ágil.

El desarrollo ágil requiere de un equipo con las habilidades que el negocio posea. Además, se necesita trabajar de una manera altamente colaborativa con el cliente, en pequeñas etapas, para completar porciones pequeñas de los entregables en cada ciclo (iteración) y, en cuanto sea posible, implementar los entregables (incremento) para luego obtener retroalimentación que sirva en la próxima iteración.

El resultado final es un producto que se adapta mejor a las necesidades actuales del cliente y se entrega con costos, desperdicios y tiempos mínimos, ya que

las iteraciones fomentan la retroalimentación y la revisión, por lo que se obtienen mayores beneficios que mediante los enfoques tradicionales.

### **2.2.5 Metodologías Ágiles**

Son métodos para la gestión de proyectos, enfocados en orientar equipos de trabajo en un desarrollo conjunto de manera efectiva para alcanzar los objetivos del negocio. Buscan resolver los problemas que empezaron a detectarse en la década de los 90's, cuando las expectativas y necesidades de los usuarios finales de los proyectos de desarrollo de software, se incumplían y las empresas se vieron inmersas en una serie de proyectos que no lograban sus objetivos; eran demasiado costosos y a la hora de salir al mercado ya no eran competitivos.

Las metodologías ágiles ofrecen, entonces, una alternativa a los métodos tradicionales que están caracterizados por el énfasis en la documentación, la definición de un alcance fijo, el establecimiento de un cronograma de ejecución del proyecto y un costo específico. Esta metodología está orientada en el logro de objetivos del proyecto, sin dar relevancia a las tareas que no son imprescindibles y reconoce como factor fundamental de éxito la construcción de un buen equipo de trabajo, por encima de los procesos y herramientas que sean requeridos en el proyecto.

Las empresas que llegan a implementar metodologías ágiles logran mejorar la gestión de sus proyectos, lo cual les significa una disminución de costos y un aumento de productividad al concebir lo incierto e impredecible como oportunidades de cambio que son bienvenidas en el proyecto, con el fin de cumplir con las expectativas del cliente. Sin embargo, este escenario no es el común de las empresas a nivel mundial, aún existe una gran brecha en el número de proyectos de software que terminan de manera exitosa y que logran el objetivo inicial planteado de manera eficaz e incrementa su propuesta de valor para el cliente (Alliance, 2016).

The Standish Group (2015), analizó 50.000 proyectos de la industria de desarrollo de software alrededor del mundo, y evidenció que existe una diferencia

porcentual considerable entre la probabilidad de que un proyecto con metodologías ágiles finalice cumpliendo con las expectativas esperadas, respecto a un proyecto que se desarrolla de manera tradicional o en método cascada.

### **2.2.6 Ciclos de vida adaptativos**

Los ciclos de vida adaptativos o metodologías ágiles son conocidos como métodos orientados al cambio, que responden a la participación continua de los interesados. Existen diferentes modelos básicos para este tipo de ciclos de vida, aquellos que se encuentran centrados en el flujo y otros centrados en ciclos iterativos e incrementales (ISOTools, 2018).

#### **2.2.6.1 Scrum**

Scrum es un marco de trabajo diseñado de tal forma que logra la colaboración eficaz del equipo de trabajo, emplea un conjunto de reglas y se definen roles para generar una estructura de correcto funcionamiento. Scrum utiliza un elemento representativo llamado Sprint que corresponde a una etapa de trabajo donde se crea una versión utilizable del producto. Cada sprint es considerado como un proyecto individual. Un Sprint está compuesto por los siguientes elementos: reunión de planeación del Sprint, Daily Scrum o reunión diaria, trabajo de desarrollo, revisión y retrospectiva del Sprint.

Según OBS Business School, (2017), Scrum es una metodología que surgió para administrar de manera dinámica proyectos de desarrollo de software, pero se puede adaptar para el trabajo en las organizaciones. Es un método para trabajar en equipo a partir de iteraciones o sprints. El objetivo de esta metodología es planificar y controlar proyectos donde hay gran incertidumbre por los cambios que suceden a última hora.

Lo principal de la metodología Scrum son los Sprints, el cual es un intervalo prefijado durante el cual se crea un producto funcional que formará parte del producto final. A lo largo del desarrollo hay sprints consecutivos de duración constante. Se podría decir que cada sprint es un mini proyecto de no más de un mes de duración. Cada sprint cuenta con un diseño, un plan flexible de desarrollo, y unos requisitos que

darán lugar al producto deseado. Cada Sprint puede tener una serie de eventos o etapas. Los eventos más comunes son:

1. Reunión para la planificación del Sprint: El trabajo a realizar para cada sprint se prevé en esta reunión. Es un evento o etapa en el que colabora todo el equipo Scrum y es de duración variable, por ejemplo, para un sprint de un mes tiene unas ocho horas de duración.
2. Scrum diario: Es un evento de 15 minutos cuyo fin es que el equipo de Desarrollo sincronice actividades, y establezca un plan para las próximas 24 horas. Se basa en poner en común y sincronizar actividades para elaborar el plan del día. Esto se realiza mediante la inspección del trabajo desde el último Scrum Diario, lo que aporta datos de la progresión del sprint y de la tendencia del progreso. Este evento aporta comunicación diaria entre todos los interesados, y elimina la necesidad de realizar otras reuniones.
3. Revisión del Sprint: Se lleva a cabo al final del Sprint, consiste en una reunión con el cliente o dueño del proyecto y todo el equipo Scrum, en la que se estudia y revisa el Product Backlog del Sprint. Se definen los aspectos a cambiar, en caso necesario, de mayor valor o probables para planificarlo en el siguiente Sprint.
4. Retrospectiva del proyecto: Oportunidad del equipo de desarrollo para mejorar su proceso de trabajo y aplicar los cambios en los siguientes Sprints. El propósito de la retrospectiva de Sprint es:
  - Revisar cómo fue el último Sprint en lo que respecta a las personas, relaciones, procesos y herramientas
  - Identificar y ordenar los temas principales que salieron bien y las potenciales mejoras
  - Crear un plan para la implementación de mejoras con respecto a cómo el Equipo Scrum hace su trabajo.

Por último, la metodología Scrum tiene unos roles y responsabilidades principales, asignados a sus procesos de desarrollo. Estos son:

- **Product Owner.** Se asegura de que el proyecto se esté desarrollando acorde con la estrategia del negocio. Escribe historias de usuario, las prioriza, y las coloca en el Product Backlog. Debe conocer la velocidad del equipo, para realizar estimaciones de cuándo estarán implementadas las necesidades en el producto. Es el responsable de cancelar el sprint si ocurre un imprevisto extremo.
- **Scrum Master o Facilitador.** Debe participar en las reuniones y asegurarse de que cumplan el tiempo y el objetivo establecido. Elimina los obstáculos que impiden que el equipo cumpla con su objetivo. Se encarga también de que todo el equipo siga la metodología Scrum y que sea entendido por todos.
- **Development Team Member.** Los encargados de crear el producto para que pueda estar listo con los requerimientos necesarios. Se recomienda que sea un equipo multidisciplinar, de no más de 10 personas.

#### **2.2.6.2 Kanban**

Kanban permite visualizar el flujo de trabajo en una barra de tareas a través de tarjetas. Propone distribuir las mismas en una serie de columnas. Kanban trabaja con tableros que pueden ser tanto físicos como digitales y permite una visualización clara de todas las tareas a realizar, en qué etapa está cada una y quién es el encargado de las mismas.

La principal diferencia con otras aproximaciones ágiles, es que en Kanban no existen las iteraciones. En su lugar, Kanban se centra en controlar el Work In Progress (WIP). Cuando hay poco WIP, se añade el ítem más prioritario del Product Backlog, y se controla que nunca se supere una cierta cantidad de WIP. Dadas sus características, no se adapta a un desarrollo basado en entregas, y actualmente se utiliza especialmente en entornos de mantenimiento.

### **2.2.6.3 XP Programming**

La programación extrema o eXtreme Programming (XP) es un enfoque de la ingeniería de software formulado por Kent Beck. Se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad. La programación extrema mejora un proyecto de software en cinco formas esenciales; Comunicación, simplicidad, retroalimentación, respeto y coraje. Los programadores extremos se comunican constantemente con sus clientes y compañeros programadores. Mantienen su diseño simple y limpio, reciben comentarios al probar su software desde el primer día y entregan el sistema a los clientes lo antes posible e implementan los cambios como se sugiere.

La programación extrema es exitosa porque enfatiza la satisfacción del cliente. En lugar de entregar todo lo que pueda desear en una fecha lejana en el futuro, este proceso entrega el software que necesita a medida que lo necesita. La programación extrema permite a los desarrolladores responder con confianza a los cambios en los requisitos de los clientes, incluso al final del ciclo de vida. (Extreme programming Organization)

### **2.2.7 Organización Internacional de la Estandarización (ISO)**

Las normas ISO son normativas internacionales para la optimización de los procesos empresariales, fueron elaboradas por la Organización Internacional de la Estandarización (ISO, International Standard Organization). Las normas ISO son documentos que especifican requerimientos que pueden ser empleados en organizaciones para garantizar que los productos y/o servicios ofrecidos por dichas organizaciones cumplen con su objetivo. Hasta el momento ISO, ha publicado alrededor de 19.500 normas internacionales que se pueden obtener desde la página oficial.

El objetivo es asegurar que los productos y/o servicios alcancen la calidad deseada. Para las organizaciones son instrumentos que permiten minimizar los costos, ya que hacen posible la reducción de errores y sobre todo favorecen el incremento de la productividad. Los estándares internacionales ISO son clave para acceder a

mercados nacionales e internacionales y de este modo, estandarizar el comercio en todos los países favoreciendo a los propios organismos públicos.

### **2.2.8 ISOTools**

ISOTools es una herramienta que permite a las organizaciones la implantación, mantenimiento y mejora continua de sus Sistemas de Gestión. Esta herramienta puede ser utilizada indistintamente por organismos de la administración pública, universidades (públicas y privadas), centros de enseñanza, fundaciones, grandes corporaciones y empresas de todos los sectores, es decir, es aplicable a todos los sistemas.

### **2.2.9 Ventajas de ISOTools**

La implementación de ISOTools aporta resultados en el corto plazo ya que proyecto a proyecto, ISOTools tiene el objetivo de conseguir retornos cuantificables sobre la inversión realizada. Desde menores costos y mayor velocidad de comercialización, hasta mejoras en cumplimiento y servicio al cliente. Esto aporta grandes ventajas como:

- Crecer con la organización: Permite aumentar progresivamente las funcionalidades gracias a su sistema modular. ISOTools puede escalarse al ritmo la empresa.
- Es adaptable: Gracias al alto nivel de parametrización, ISOTools puede adaptarse al 100% a las necesidades de su sistema de gestión.
- Ampliamente Funcional: Mediante aplicaciones destinadas a diferentes objetivos, ISOTools tiene la capacidad de automatizar un gran número de estándares y modelos de gestión.

Además, cabe destacar que ISOTools optimiza el tiempo de resolución de una acción correctiva, reduce el tiempo de preparación de las reuniones de gestión y de igual modo, reduce el tiempo dedicado a recopilar y tratar indicadores.

### **2.2.10 Diseño Web Responsive**

El diseño web adaptable o responsive, conocido por las siglas RWD (del

inglés, responsive web design) no consiste en hacer una versión móvil de un sitio. El diseño web responsive trata de redimensionar y colocar los elementos del sitio web para que se adapten a cada dispositivo. Esto permite que todo se visualice correctamente.

Dicho en otras palabras, el diseño responsive consiste en preparar la estructura de un sitio web para que sea capaz de adaptar los contenidos tanto al tamaño y resolución de la pantalla como al tipo de dispositivo desde el que se están visualizando, ya sea ordenador, tablet y smartphone. El diseño web responsive emplea hojas de estilo en CSS3, que usan el atributo media query para convertir una web normal en una web multiplataforma que pueda adaptarse a todos los tamaños y dispositivos.

Las media queries son una serie de órdenes que indican al documento HTML cómo debe comportarse en diferentes resoluciones de pantalla. Esto permite que se pueda convertir una web en un sitio multiplataforma, además esto trae muchas ventajas las cuales serán explicadas a continuación.

### **2.2.11 Ventajas del Diseño Web Responsive**

Es de suma importancia que se conozcan cuáles son las ventajas del diseño web responsive debido a que este sistema aplicará este tipo de diseño por lo que disponer de un diseño adaptativo es un factor clave para el diseño web de una empresa u organización. Hay gran variedad de motivos que impulsan hacia ese camino. Entre ellos tenemos:

- **Mejora la experiencia del usuario:** Que un usuario pueda acceder a los contenidos desde cualquier dispositivo y que dichos contenidos se visualicen correctamente mejora su experiencia. Una buena experiencia de usuario mejora la imagen de marca y aumenta la tasa de rebote en un sitio web. Además, potencia la interacción, la suscripción a boletines y, en el caso de los sitios de comercio electrónico, también las ventas.
- **Carga el sitio más rápidamente:** El diseño web responsive mejora la

velocidad de carga del sitio en los distintos dispositivos. Una mejor velocidad de carga no solo mejora la experiencia de usuario, sino que también influye en la mejora del posicionamiento web.

- **Mejora la experiencia de navegación:** El diseño web responsive no solo mejora la navegación del sitio en la distribución de los menús, sino que también facilita la experiencia de navegación usando botones adaptados al uso táctil.
- **Mejora la analítica web:** Herramientas de analítica web como Google Analytics están optimizadas para gestionar múltiples dispositivos e informes de sitios webs con diseño web responsive. Esto te permitirá saber desde qué dispositivos acceden los usuarios y cómo se comportan.
- **Mejora la visibilidad en los motores de búsqueda:** Algunas de las ventajas anteriores influyen directamente en el SEO de una web, como la velocidad de carga o prescindir de duplicar contenidos en las versiones móviles. En cualquier caso, Google recomienda a los webmasters que sigan la mejor práctica de la industria usando diseño web adaptable (responsive web design), empleando el mismo HTML para todos los dispositivos y usando solamente CSS media queries para que el contenido se muestre correctamente en cada caso.
- **Evita la duplicación de contenidos:** Las versiones para móviles de los sitios web exigían la duplicación de contenidos, que perjudica seriamente al SEO. El diseño web responsive no requiere que se cree el contenido otra vez, puesto que lo que hace es reorganizar el contenido existente para que se visualice mejor.
- **Reducción de costes de desarrollo y mantenimiento web:** Para crear un sitio con diseño web responsive solo hace falta una plantilla para todas las plataformas, lo que reduce costes respecto a la opción de crear una versión para smartphones e incluso otra para tablets. También reduce los costes de

mantenimiento, puesto que basta con modificar la plantilla principal para que surta efecto en la visualización desde todos los dispositivos.

### **2.2.12 Base de Datos (MySQL)**

Según Cobos, A. (2016) explica que: “MySQL es un sistema de administración de base de datos relacionales rápido, sólido y flexible. Es ideal para crear bases de datos con acceso desde páginas web dinámicas, para la creación de sistemas de transacciones on-line o para cualquier otra solución profesional que implique almacenar datos, teniendo la posibilidad de realizar múltiples y rápidas consultas”. Por otro lado, Centeno, E. y Cordonez, S. (2016) consideran que “es un gestor de base de datos Open Source relacional muy popular, confiable y rápido que nos permite almacenar todos los datos de un sistema.” (p. 22).

Es mayormente utilizando en conjunción con servidores web donde se lo encuentra relacionado a aplicaciones web o CMS para sitios online. Está muy ligado al lenguaje de programación PHP en lo que se refiere a este tipo de desarrollos. Es una base de datos que presenta rapidez en lectura, sobre todo cuando se utilizan ciertos motores como MyISAM o InnoDB.

A pesar de ello, se puede presentar problemas al utilizarla en entornos o desarrollos que presenten una alta concurrencia de modificación. Como en los desarrollos de aplicaciones web no se suele presentar un número elevado de modificaciones de datos, sino más bien de lectura frecuente de datos, MySQL se presenta como la base de datos ideal para este tipo de casos.

Estas bondades que se mencionan, además de ser código abierto, son las principales razones por las que se tomó en cuenta MySQL para poder almacenar la información correspondiente de la organización. Igualmente, sea cual sea el entorno y objetivo por el cual se pretenda utilizar MySQL, es necesario monitorizar el rendimiento para poder corregir errores, tanto de programación como de queries SQL.

## **2.3 Bases Legales**

Son las regulaciones que imponen metas y acciones correspondientes al tipo

de organización que elabora la planificación, fijando sanciones cuando estas no son cumplidas. Las bases legales dan sustento a la investigación por medio de leyes, reglamentos y decretos. A continuación, se citarán los fundamentos legales de la investigación:

- Constitución de la República Bolivariana de Venezuela (1999)
  - Artículo 98. La creación cultural es libre. Esta libertad comprende el derecho a la inversión, producción y divulgación de la obra creativa, científica, tecnológica y humanística, incluyendo la protección legal de los derechos del autor o de la autora sobre sus obras. El Estado reconocerá y protegerá la propiedad intelectual sobre las obras científicas, literarias y artísticas, invenciones, innovaciones, denominaciones, patentes, marcas y lemas de acuerdo con las condiciones y excepciones que establezcan la ley y los tratados internacionales suscritos y ratificados por la República en esta materia.
- Ley Orgánica de Ciencia, Tecnología e Innovación (2010)
  - Artículo 2. Las actividades científicas, tecnológicas y de innovación son de interés público y de interés general.
  - Artículo 9. El Ministerio de Ciencia y Tecnología apoyará a los organismos competentes por la materia, en la definición de políticas tendentes a proteger y garantizar la propiedad intelectual colectiva de los conocimientos, tecnologías e innovaciones de los pueblos indígenas y los conocimientos tradicionales.
- Gaceta oficial N.º 38.095 del decreto N.º 3.390 Artículo N.º 1 (2004)
  - Plan Migración al Software Libre. Para cumplir con lo establecido en el Decreto 3.390 sobre el uso prioritario del Software Libre en la Administración Pública Nacional, se propone el presente “Plan Nacional de Migración”, que servirá como pilar fundamental para que los Órganos y Entes de la Administración Pública Nacional, diseñen y ejecuten sus

respectivos planes de implantación progresiva del software libre desarrollado con estándares abiertos o “Planes Institucionales de Migración”, alcanzando de esta manera una Administración Pública Nacional con plataformas tecnológicas seguras, ínter operables, escalables, fácilmente replicables, metodológicamente fundamentadas y técnicamente independientes, todo ello basado principalmente en la libertad de uso del conocimiento y la transferencia tecnológica.

§ Artículo 1. La Administración Pública Nacional empleará prioritariamente Software Libre desarrollado con Estándares Abiertos, en sus sistemas, proyectos y servicios informáticos. A tales fines, todos los órganos y entes de la Administración Pública Nacional iniciarán los procesos de migración gradual y progresiva de éstos hacia el Software Libre desarrollado con Estándares Abiertos.

§ Artículo 5. El Ejecutivo Nacional fomentará la investigación y desarrollo de software bajo modelo Software Libre desarrollado con Estándares Abiertos, procurando incentivos especiales para desarrolladores.

· Ley Especial contra los Delitos Informáticos

○ Artículo 6. Acceso indebido. Toda persona que sin la debida autorización o excediendo la que hubiere obtenido, acceda, intercepte, interfiera o use un sistema que utilice tecnologías de información, será penado con prisión de uno a cinco años y multa de diez a cincuenta unidades tributarias. (p. 6)

○ Artículo 7. Sabotaje o daño a sistemas. Todo aquel que con intención destruya, dañe, modifique o realice cualquier acto que altere el funcionamiento o inutilice un sistema que utilice tecnologías de información o cualquiera de los componentes que lo conforman, será

penado con prisión de cuatro a ocho años y multa de cuatrocientas a ochocientas unidades tributarias. Incurrirá en la misma pena quien destruya, dañe, modifique o inutilice la data o la información contenida en cualquier sistema que utilice tecnologías de información o en cualquiera de sus componentes. La pena será de cinco a diez años de prisión y multa de quinientas a mil unidades tributarias, si los efectos indicados en el presente artículo se realizaren mediante la creación, introducción o transmisión, por cualquier medio, de un virus o programa análogo. (p. 7)

- Artículo 11. Espionaje informático. Toda persona que indebidamente obtenga revele o difunda la data o información contenida en un sistema que utilice tecnologías de información o en cualquiera de sus componentes, será penada con prisión de tres a seis años y multa de trescientas a seiscientas unidades tributarias. La pena se aumentará de un tercio a la mitad, si el delito previsto en el presente artículo se cometiere con el fin de obtener algún tipo de beneficio para sí o para otro. El aumento será de la mitad a dos tercios, si se pusiere en peligro la seguridad del Estado, la confiabilidad de la operación de las instituciones afectadas o resultare algún daño para las personas naturales o jurídicas, como consecuencia de la revelación de las informaciones de carácter reservado. (p. 8)
- Artículo 13. Hurto. Quien, a través del uso de tecnologías de información, acceda, intercepte, interfiera, manipule o use de cualquier forma un sistema o medio de comunicación para apoderarse de bienes o valores tangibles o intangibles de carácter patrimonial sustrayéndolas a su tenedor, con el fin de procurarse un provecho económico para sí o para otro, será sancionado con prisión de dos a seis años y multa de doscientas a seiscientas unidades tributarias. (p. 9)

- Artículo 14. Fraude. Todo aquel que, a través del uso indebido de tecnologías de información, valiéndose de cualquier manipulación en sistemas o cualquiera de sus componentes, o en la data o información en ellos contenida, consiga insertar instrucciones falsas o fraudulentas, que produzcan un resultado que permita obtener un provecho injusto en perjuicio ajeno, será penado con prisión de tres a siete años y multa de trescientas a setecientas unidades tributarias. (p. 9)

#### **2.4 Definición de términos básicos**

A continuación, se definen los conceptos básicos y fundamentales que se originan y hacen referencia a la presente investigación.

**Aplicación Web:** en la ingeniería de software se denomina aplicación web a aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de internet o de una intranet mediante un navegador

**Framework:** Es un esquema (un esqueleto, un patrón) para el desarrollo y/o la implementación de una aplicación.

**Base de Datos:** Según (Kendall & Kendall, 2005) “las bases de datos no son meramente una colección de archivos. Más bien, una base de datos es una fuente central de datos significativos, los cuales son compartidos por numerosos usuarios para diversas aplicaciones”.

**Software:** Está compuesto por un conjunto de programas que son diseñados para cumplir una determinada función dentro de un sistema.

**ISO:** Organización Internacional de Estandarización (International Organization Standard).

**Auditoria:** es un proceso sistemático de obtener y evaluar los registros patrimoniales de un individuo o empresa a fin de verificar su estado.

**Dashboard:** es una representación gráfica de los principales indicadores que intervienen en la consecución de los objetivos de negocio, y que está orientada a la toma de decisiones para optimizar la estrategia de la empresa. Un dashboard debe

transformar los datos en información y esta en conocimiento para el negocio.

**Mapas de procesos:** se define gráficamente, en lo que se conoce como diagramas de valor, combinando la perspectiva global de la compañía con las perspectivas locales del departamento respectivo en el que se inscribe cada proceso. Su desarrollo, por lo tanto, debe tratar de consensuar la posición local y el desempeño concreto de dichos procesos con los propósitos estratégicos corporativos, por lo que resulta imprescindible identificarlos y jerarquizarlos en función de su definición específica.

**Gestión documental:** está diseñado para almacenar, administrar y controlar el flujo de documentos dentro de una organización. Se trata de una forma de organizar los documentos e imágenes digitales en una localización centralizada a la que los empleados puedan acceder de forma fácil y sencilla.

**Hallazgos:** son los resultados de la evaluación de la evidencia de la auditoría recopilada frente a los criterios de auditoría.

## **CAPÍTULO III**

### **MARCO METODOLÓGICO**

En la presente investigación se describen los procedimientos que se utilizaron para abordar el problema planteado, de tal forma esta investigación se trabajó bajo un paradigma cuantitativo, definido por Arias (2012), como “el conjunto de pasos, técnicas y procedimientos que se emplean para formular y resolver problemas de investigación mediante la prueba o verificación de hipótesis” (p.19), dando a expresar que constituye el proceso que permite obtener nuevos conocimientos en el campo de la realidad social, o bien estudiar una situación para diagnosticar necesidades y problemas a efectos de aplicar los conocimientos con fines prácticos, es decir, una estrategia de acción para desarrollar este estudio de acuerdo a las etapas y momentos que se requiera.

#### **3.1. Tipo y nivel de la investigación**

El tipo de investigación según Finol (2008) se basa en "el problema planteado y los objetivos a alcanzar" (p.61). Esta afirmación coincide con Balestrini (2002), para quien "según sean los objetivos delimitados, se podrá definir el tipo de estudio de que se trata" (p.129). El objetivo principal de esta investigación es plantear la realización de una herramienta que permita la agilización de los procesos de gestión de las empresas haciendo uso de un software de gestión ágil de proyectos en conjunto de herramientas ISOTools.

Continuando con lo anteriormente mencionado, se considera que el nivel de la investigación es del tipo descriptiva, ya que se analizan procedimientos y elementos inherentes a la misma. Tal como señala Arias (2006), "los estudios descriptivos permiten reseñar las características primordiales de un fenómeno con el objeto de conocer cómo y cuáles son las maneras en que éste tiende a manifestarse" (p.45).

Además, se puede agregar la siguiente definición según Méndez (2012) donde destaca “el estudio descriptivo identifica características del universo de investigación o variable objeto de estudio. Este tipo de investigación requiere un considerable

conocimiento del área que se investiga para formular las preguntas específicas que busca responder”. Por este motivo, el nivel de investigación es útil para definir todos los parámetros establecidos mediante procedimientos que permitan conocer qué tipos de datos deben recolectarse, cómo se recolectarán y cómo se procesarán.

### **3.2. Diseño de la investigación**

Para Bastidas (2011) el diseño de investigación "es la modalidad específica de investigación relacionada usualmente con las fuentes de información y la forma de recoger los datos" (p.52). El diseño es de campo porque se buscará esta información directamente de las fuentes vivas que presentan la problemática, al respecto, Sabino (2002) destaca que en el diseño de campo "los datos de interés se recogen en forma directa de la realidad. Son datos de primera mano, originales, obtenidos sin ningún tipo de intermediación" (p.64). Por otra parte, el diseño es no experimental porque no se manipula ni se ejerce control alguno sobre las variables, el estudio consiste en recoger los datos tal cual como se presentan en la realidad para describirlos.

Para Camirra y Cartaya (2009) el diseño es no experimental porque: "es un diseño en el que no se ejerce manipulación ni control alguno sobre las variables bajo estudio" (p.25) De igual manera, el presente estudio es una investigación documental, ya que la información adicional que respalda la presente investigación será extraída de autores pasados. De este modo, Arias (2012) define la investigación documental como "un proceso basado, en la búsqueda, recolección, análisis, crítica e investigación de los datos secundarios, es decir los datos obtenidos y registrados por otros investigadores en fuentes documentales" (p.27).

### **3.3. Población y muestra**

#### **3.3.1. Población**

Es necesario establecer una población donde el estudio vaya dirigida a esta misma, es decir, un grupo de individuos que posean características comunes. Arias (2006) define la población como "el conjunto finito o infinito con características comunes para los cuales serán extensivas las conclusiones de la investigación. Esta

queda limitada por el problema y por los objetivos del estudio” (p.81).

Tomando en cuenta lo mencionado anteriormente, se puede definir la población del presente estudio como empresas de cualquier rubro pertenecientes al Edo. Carabobo. De este modo, se seleccionó una población al azar de un total de ocho (8) empresas de desarrollo de software ubicadas en el municipio Valencia (perteneciente al Edo. Carabobo).

### **3.3.2. Muestra**

De acuerdo con Balestrini (2006), muestra “es una parte representativa de una población, cuyas características deben reproducirse en ella lo más exactamente posible” (p.142). Para esta investigación la muestra se encontró representada por algunos gerentes de proyectos de empresas de desarrollo de software, tomando como muestra, un total de cinco (5) gerentes encargados de la gestión de proyectos de la empresa.

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

Con respecto a las técnicas e instrumentos de recolección de datos, Arias (2012) señala que: “se entenderá por técnica, el procedimiento o forma particular de obtener datos o información, y los instrumentos como medios materiales que se emplean para recoger y almacenar información” (p.67). Para el desarrollo de la investigación se emplearán las siguientes técnicas e instrumentos de recolección de datos: entrevista no estructurada y observación directa.

Según Hurtado (2007) define entrevista no estructurada de la siguiente manera: “la formulación de preguntas libres, cada una basada en la respuesta que va dando el interrogado, por lo cual, las preguntas pueden variar de un interrogatorio a otro” (p. 44). Por este motivo se plantea el uso de entrevista no estructurada, pues es bastante flexible y permite realizar preguntas cambiantes para cumplir con los objetivos planteados.

De este modo, Arias (2006) explica “la observación es una técnica que consiste en visualizar o captar mediante la vista, en forma sistemática, cualquier

hecho, fenómeno o situación que se produzca” (p. 69). Se emplea esta técnica para recibir información no sesgada acerca de los procesos que requieren mejoría.

### **3.5. Fases metodológicas**

El presente proyecto emplea la metodología XP (Programación Extrema) para el desarrollo del software en cuestión. Esta metodología es destacada por ser una de las más apropiadas para integrar en los procesos ágiles de desarrollo de software, debido a que esta hace énfasis en encontrar la solución más simple del problema. Luego por su flexibilidad, se pueden añadir funcionalidades extras hasta obtener un producto lo más adaptado posible a los requerimientos. Por otro lado, se hace uso de ISOTools, una herramienta que permite la implementación, mejora continua, mantenimiento, controles de estándares y auditorías del software.

Ambos elementos del software trabajan en conjunto para entregar el proyecto que necesitan dentro del parámetro de tiempo más óptimo, además, impulsan a los desarrolladores a responder a los requerimientos cambiantes de los clientes. De este mismo modo, las visualizaciones en tiempo real en un dashboard, el análisis de riesgos y oportunidades, los mapas de procesos y la gestión documental son las mejores herramientas para concretar los objetivos.

En este mismo orden de ideas, la metodología XP define cuatro variables fundamentales: calidad, tiempo, costo y alcance, las cuales pueden ser fijadas arbitrariamente (a excepción del alcance) por actores externos al equipo de desarrollo. Por este motivo, se establece un ciclo de vida en el que se definen 4 fases: analizar, planificar y definir las necesidades del cliente, proyectar los requerimientos funcionales del software, desarrollar el software mediante un proceso iterativo en el que se realizan varias entregas al cliente y, finalmente, realizar una entrega final para la puesta en producción del software; todo esto sometiendo el software a pruebas constantes. En la investigación se definen las siguientes fases:

#### **Fase I: Análisis y Planificación**

En esta primera fase se determina el propósito del software, se recopila la

mayor cantidad de información posible de los problemas que se requieren resolver, se definen sus objetivos específicos (requisitos funcionales y no funcionales) y el alcance del mismo que tendrá el mismo. De igual modo se asignan tiempos de desarrollo, se definen los principales riesgos y se plantea una aproximación inicial a las necesidades de información.

### **Fase II: Diseño.**

Pressman (2010), dice sobre el diseño: “El diseño XP sigue rigurosamente el principio MS (mantenlo sencillo). Un diseño sencillo siempre se prefiere sobre una representación más compleja” (P.62). En esta fase se realizará el diseño de la aplicación por medio de un prototipo no funcional. Se diseñará la arquitectura del sistema que permite implementar los requerimientos definidos. Tomando el caso de esta investigación, se hará uso del lenguaje unificado de modelado (UML). Para realizar el software es necesario un diseño intuitivo y sencillo que los usuarios vayan moldeando a través de cambios.

### **Fase III: Desarrollo o Codificación.**

En esta fase de desarrollo se codifica toda el área funcional del sistema adhiriéndose a los mejores estándares de codificación. La programación de software bajo estándares mantiene el código consistente y facilita su comprensión y escalabilidad. De igual modo, es importante que el código sea reutilizable, de este modo se podrá implementar en futuros sistemas que necesiten extender las funcionalidades partiendo de lo desarrollado anteriormente.

### **Fase IV: Pruebas.**

Por último, en la fase de la metodología XP en esta se realizan las distintas pruebas al software para determinar el funcionamiento óptimo y planificado del mismo, en caso de haber fallas o errores, realizar las respectivas correcciones. De igual modo, se procede a refactorización y optimización de código lo que lo convierte en más eficaz y óptimo.

## **CAPÍTULO IV**

### **ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS**

En el presente capítulo se exponen los análisis y deducciones de los resultados que vienen derivados de las fases metodológicas propuestas y que se acoplan, a su vez, con los objetivos planteados para el correcto desarrollo y funcionamiento del software. Es por este motivo que surgen distintas metodologías, las cuales pueden ser adaptables a un proyecto en concreto.

Esta etapa tiene como objetivo levantar la información de los procesos relacionados a la gestión de proyectos de desarrollo de software principalmente. Por esta razón, fueron evaluadas las diferentes metodologías existentes con el fin de obtener la más factible. Del mismo modo, se busca que la misma se adapte a las características del software, llegando así a la conclusión que la metodología más destacable es la metodología XP.

La metodología XP es una metodología de desarrollo ágil basada en una serie de prácticas de buenas maneras que persigue el objetivo de aumentar la productividad a la hora de desarrollar programas, logrando de esta manera dividir el desarrollo de esta aplicación en cuatro fases, tal y como lo dicta dicha metodología, siendo estas los cuatro objetivos fundamentales de la metodología, estos están divididos como objetivos específicos con los nombres de diagnóstico, diseño, desarrollo y pruebas, buscando lograr a través de estos el desarrollo óptimo de la aplicación.

#### **4.1 Fase I: Planificación**

##### **4.1.1 Entrevista**

En esta fase se realizó una investigación preliminar donde se aplicaron dos instrumentos de recolección de datos, siendo el primero la entrevista no estructurada y la segunda, el cuestionario. Esto con la finalidad de definir los parámetros necesarios para la realización del software. Esta fue aplicada de la siguiente forma:

---

**Pregunta 1: ¿Conoce alguna de las metodologías ágiles?, si su respuesta es afirmativa, ¿Aplica alguna metodología ágil para la ejecución de sus proyectos? Explique el porqué.**

**Respuesta Aleinis Arguello: Si, el desarrollo de proyectos en mi departamento aplicamos utilizamos una metodología propia, pero está en su mayoría basada en la metodología Scrum.**

**Respuesta Miguel Sarmiento: Si, conozco la metodología Scrum, pero no utilizo ni esta ni ninguna otra de las metodologías conocidas como ágiles.**

---

Tabla 1: Pregunta de Entrevista 1. Bravo A. (2020)

---

**Pregunta 2: ¿Cómo es el control de proyectos en su empresa, y quién es el responsable?**

**Respuesta Aleinis Arguello: Utilizamos una metodología propia adaptada de la metodología scrum. En principio, sigue sus mismas prácticas como reuniones cortas, sprints y objetivos a corto plazo y a largo plazo.**

---

---

**Respuesta Miguel Sarmiento:** Básicamente se crea un plan de trabajo en donde se establecen los objetivos y los plazos basándonos en las necesidades a satisfacer con el proyecto. La persona responsable puede variar según el tipo de proyecto y es asignada en base a la experiencia en dicha área.

---

Tabla 2: Pregunta de Entrevista 2. Bravo A. (2020)

---

**Pregunta 3:** ¿La planificación y cronogramas de trabajo se cumplen a cabalidad?

**Respuesta Aleinis Arguello:** Usualmente se cumplen los tiempos con satisfacción. Sin embargo, varias veces nos hemos encontrado con imprevistos que han atrasado el tiempo de entrega.

**Respuesta Miguel Sarmiento:** No siempre, pues como en todo proyecto hay imprevistos que pueden retrasar el desarrollo del mismo. Durante el proceso de planificación se trata siempre de dar cierto margen adicional de tiempo previendo la existencia de estos posibles imprevistos, pero puede haber algunos de importancia que son los que retrasan la ejecución en general.

---

Tabla 3: Pregunta de Entrevista 3. Fuente: Bravo A. (2020)

---

**Pregunta 4:** ¿Qué estrategias se implementan a la hora de fallar un plan de trabajo?

---

---

**Respuesta Aleinis Arguello: Siempre se intenta detectar, alertar, tratar y minimizar los riesgos que este conlleva. Lo más común es encontrar el origen de la falla y corregirlo, siempre utilizando el principio de la comunicación.**

**Respuesta Miguel Sarmiento: Buscar las causas de dicho fallo y mirar si puede haber alguna solución fácil de implementar o inmediata. De no ser así, se realiza una nueva planificación a partir de ese punto y si el proyecto pertenece a un cliente toca renegociar tiempos de entrega y tal vez costos.**

---

Tabla 4: Pregunta de Entrevista 4. Fuente: Bravo A. (2020)

---

**Pregunta 5: ¿Según sus estadísticas cómo es el nivel de conformidad de los clientes?**

**Respuesta Aleinis Arguello: En términos generales es bastante satisfactorio.**

**Respuesta Miguel Sarmiento: En promedio, el nivel de conformidad es bueno.**

---

Tabla 5: Pregunta de Entrevista 5. Fuente: Bravo A. (2020)

---

**Pregunta 6: ¿Qué elementos o funciones considera que debe poseer un software de gestión ágil de proyectos?**

**Respuesta Aleinis Arguello: Control de tiempos en el avance de los proyectos, asignación de tareas, porcentaje de eficiencia por empleado en base a los dos puntos anteriores y porcentaje de avance en los proyectos.**

---

---

**Respuesta Miguel Sarmiento: Que sea intuitivo, que una vez uno conozca la teoría de la metodología, esta se vea reflejada de manera sencilla en dicho software. La complejidad en el manejo le resta puntos a la usabilidad de un software y afecta proporcionalmente la productividad.**

---

Tabla 6: Pregunta de Entrevista 6. Fuente: Bravo A. (2020)

---

**Pregunta 7: ¿Conoce usted de las herramientas ISOTools?, Si es así, ¿ha aplicado alguna de ellas en su entorno de trabajo?**

**Respuesta Aleinis Arguello: Si, de hecho, en mi departamento utilizamos algunas para gestionar ciertos proyectos.**

**Respuesta Miguel Sarmiento: Si las conozco y he aplicado muchas de ellas, mayormente auditorías.**

---

Tabla 7: Pregunta de Entrevista 7. Fuente: Bravo A. (2020)

---

**Pregunta 8: ¿Cree usted que la integración de las herramientas ISOTools serían útiles en su día a día en el desarrollo de sistemas?**

**Respuesta Aleinis Arguello: Si, creo que son herramientas muy beneficiosas en una empresa, pues cuentan con unas herramientas muy útiles a la hora de gestionar los proyectos.**

**Respuesta Miguel Sarmiento: Creo que si serían bastante útiles.**

---

Tabla 8: Pregunta de Entrevista 8. Fuente: Bravo A. (2020)

---

**Pregunta 9: ¿Cuáles elementos de ISOTools cree usted que pueden ser adaptados a su metodología de gestión de proyectos?**

**Respuesta Aleinis Arguello: Adaptados a la metodología que usamos creo que las mejores herramientas serían la mejora continua, auditoría y la gestión por procesos.**

**Respuesta Miguel Sarmiento: Me gustaría ver integrados a una metodología de gestión, herramientas como la gestión de riesgos corporativos ya que**

---

---

**identifica mejor las oportunidades y amenazas a las que la empresa está expuesta.**

---

Tabla 9: Pregunta de Entrevista 9. Fuente: Bravo A. (2020)

#### **4.1.2 Análisis de los resultados**

Con base en las respuestas adquiridas en la entrevista, los entrevistados explicaron de manera detallada el proceso de gestión de sus proyectos, las ventajas que la gestión aporta a su organización y de igual modo, resaltaron los beneficios de realizar una integración con las herramientas ISOTools. En las respuestas también se pudo evidenciar debilidades en ciertas situaciones de fallos. De igual modo, se puede resaltar las ventajas de poseer un software de gestión ágil de proyectos para monitorear el avance de los proyectos. Los gerentes de proyectos manifestaron la importancia de poseer uno y las grandes virtudes que se pueden alcanzar si es integrado con ISOTools.

Dicho esto, una vez que se ha analizado con exactitud cuáles son los puntos fuertes y débiles, se procede con la definición de los requerimientos funcionales y no funcionales del software:

#### **4.1.3 Requerimientos funcionales y no funcionales**

Dada por finalizada la fase de recolección de datos, se procede a definir los requerimientos funcionales y no funcionales para garantizar que el sistema se ajuste a las necesidades expuestas por lo anteriormente planteado, además, se logra un mejor proceso de desarrollo. A continuación, se muestran los requerimientos funcionales y no funcionales del sistema:

##### **Requerimientos funcionales**

- Crear, editar, mostrar y/o eliminar, usuarios, metas y proyectos, donde el rol del administrador es el único que podrá ejecutar las cuatro funcionalidades mencionadas anteriormente.
- Generar gráficas relevantes a los proyectos realizados como, por ejemplo, el

gráfico segmentado circular y la lista del backlog o actividades pendientes. En ese caso será un diagrama de quemado que representa de forma gráfica el trabajo atrasado.

- Diseñar una vista donde se pueda visualizar la cantidad de actividades pendientes por proyectos y total de horas cargadas al proyecto.
- Generar reportes de auditorías y hallazgos que permitan analizar y dar seguimiento y control a los distintos procesos del proyecto.
- Exhibir un dashboard interactivo el cual permita acumular toda la información necesaria y relevante del proyecto en cuestión, presentándolo de manera ordenada e intuitiva.

#### **Requerimientos no funcionales**

- Interfaz y vistas del usuario que sean sencillas y agradables, pero con la información pertinente. Se muestra una interfaz bastante simple e intuitiva con una paleta de colores agradable a la vista, además de poseer gráficas donde se representa la mayor cantidad de información posible.
- Portabilidad que permita ser visualizado y utilizado en diferentes navegadores y sistemas operativos, es decir, al ser un servicio web no requiere de ningún tipo de instalación adicional.
- Rendimiento de alta fiabilidad, donde la velocidad de ejecución del sistema y la cantidad de memoria necesaria deben ser óptimas, además, la tasa de fallos del sistema debe ser lo más mínimo posible.
- Todas las comunicaciones externas entre los servidores de datos, la aplicación y el cliente del sistema deben estar cifradas mediante el protocolo criptográfico TLS.
- El sistema debe ser fácilmente extensible y ampliable. La adición de funcionalidades al sistema debe ser factible y, a su vez, se debe poder mejorar las funcionalidades ya existentes. Validación de todos los campos de entrada de información

- El sistema en su totalidad será desplegado mediante contenedores de software o de aplicación para mantener facilidad en el despliegue.

## **4.2 Fase II: Diseño del sistema**

En la siguiente fase de la metodología utilizada se aplicará una serie de herramientas para la realización del sistema. Por lo que dentro de la misma se realizan distintos prototipos simples, donde entre ellos se selecciona el más eficiente en cuanto a consumir el menor tiempo y esfuerzo posible a la hora de ser maquetados y anexados al sistema, asegurándose, además, de su fácil entendimiento para el usuario destino.

### **4.2.1 Casos de uso**

Comprendiendo de qué modo sería utilizado el sistema por los diferentes tipos de usuarios y establecer las funciones y roles de los mismos dentro del sistema, se utiliza esta herramienta para facilitar la visualización de las actividades que podrá realizar cada rol de usuario según su categoría dentro del sistema y como se relacionan entre ellas. Los actores serán clasificados del siguiente modo:

- Administrador: Puede realizar todas las operaciones de crear, consultar, eliminar y modificar.
- Gerente: Puede crear, modificar y consultar, pero no tiene permisos para eliminar. De este modo se mantiene la integridad del sistema.
- Programador: Solo posee permisos para crear y editar ciertas operaciones designadas por el gerente de proyectos.

De este modo, los modelos de caso de son los siguientes:

**Figura 1: Diagrama Casos de Uso del rol Administrador**

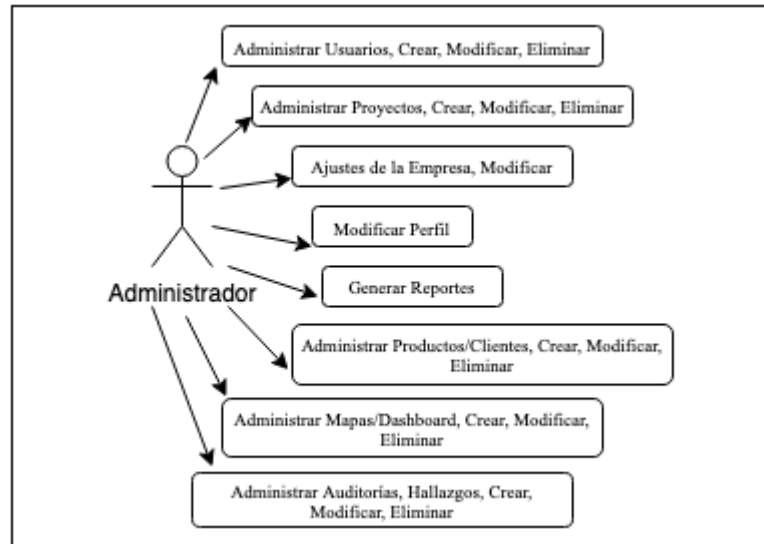


Figura 1: Casos de uso del rol administrador. Fuente: Bravo A. (2020).

**Figura 2: Diagrama Casos de Uso del rol Gerente**

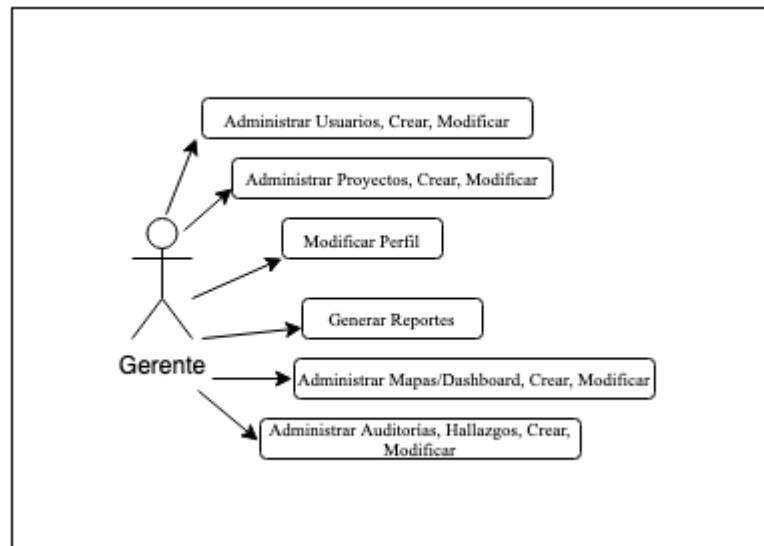


Figura 2: Casos de uso del rol gerente. Fuente: Bravo A. (2020)

**Figura 3: Diagrama Casos de Uso del rol Programador**

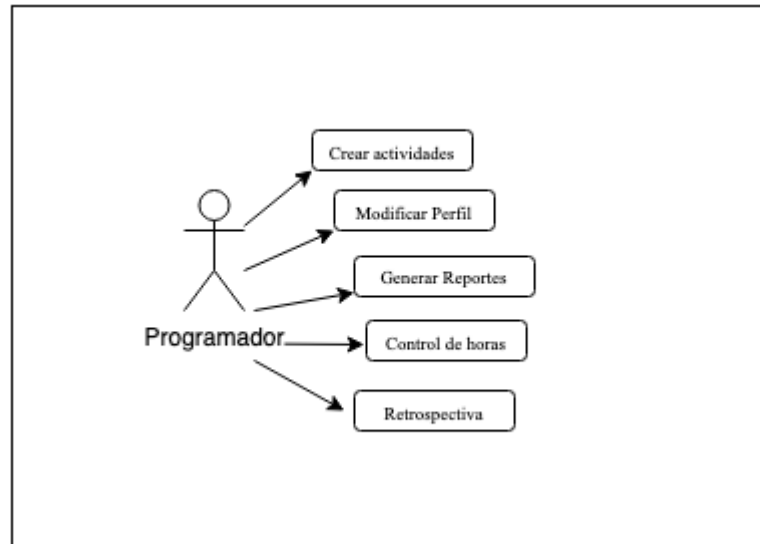


Figura 3: Casos de uso del rol programador. Fuente: Bravo A. (2020)

#### **4.2.2 Modelado de la base de datos**

En esta herramienta de modelado se representan las entidades, propiedades y relaciones. Mediante éstos, el esquema conceptual abstracto puede ser mostrado gráficamente y mantener una independencia conceptual con respecto a la implementación propiamente dicha. En realidad, podemos hacer que los diagramas sean un reflejo fiel de las relaciones, interrelaciones y atributos del modelo relacional de datos o podemos englobar diversas relaciones en una sola entidad o conjunto de propiedades.

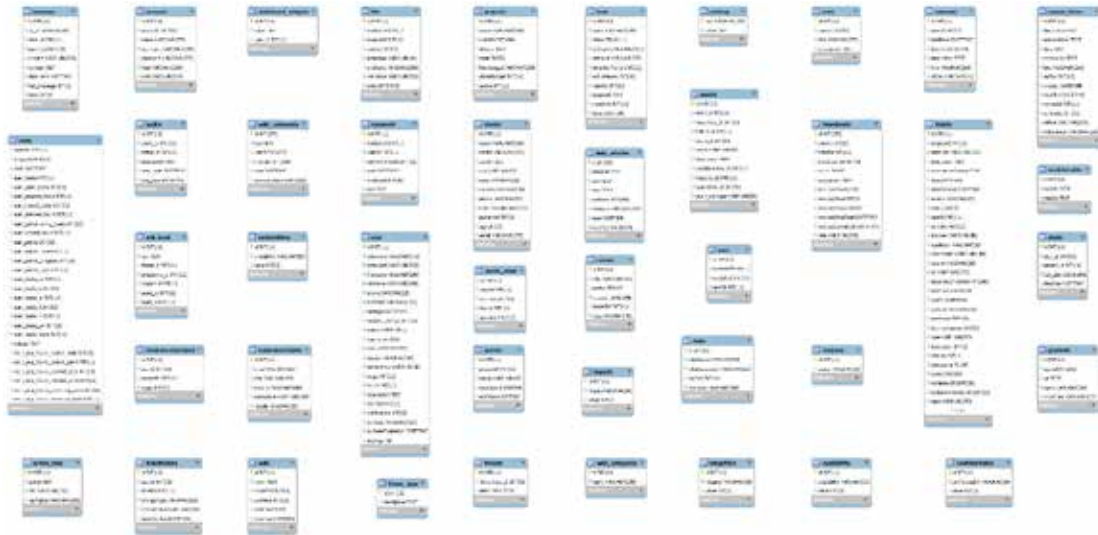


Figura 4: Modelado de datos. Fuente: Bravo A. (2020).

Para mostrar de forma más detallada las características de dichos datos se realizó una tabla de diccionario de datos, en la que se muestra cada una de las entidades insertadas o modificadas dentro de la base de datos del software, junto con su respectivo atributo, tipo de dato, longitud, restricciones, si puede ser nulo o no y por último una breve descripción del uso del dato.

#### 4.2.3 Diccionario de datos

En esta etapa se describen en detalle los procesos, flujo de datos y las estructuras de datos empleadas en el sistema propuesto. Solo se describirá el diccionario de datos de las tablas con mayor relevancia de la base de datos.

Columna	Tipo	Nulo	Predeterminado	Comentarios
<b>id</b>	int(11)	No		Id cuenta
<b>projectId</b>	int(11)	Sí	NULL	Id Proyecto
<b>name</b>	varchar(255)	Sí	NULL	Nombre cuenta
<b>username</b>	varchar(255)	Sí	NULL	Nombre usuario

<b>password</b>	varchar(255)	Sí	NULL	Clave
<b>host</b>	varchar(255)	Sí	NULL	Reside
<b>kind</b>	varchar(255)	Sí	NULL	Tipo de cuenta

Tabla 10: Diccionario de datos. Tabla “account”. Fuente: Bravo A. (2020)

Columna	Tipo	Nul	Predeterminado	Comentarios
<b>id</b>	int(11)	No		Id Acción
<b>action</b>	Text	Sí	NULL	Acción
<b>tab</b>	varchar(255)	Sí	NULL	Pestaña
<b>tabRights</b>	varchar(255)	Sí	NULL	Permisos

Tabla 11: Diccionario de datos. Tabla “action\_tabs”. Fuente: Bravo A. (2020)

Columna	Tipo	Nul	Predeterminado	Comentarios
<b>id</b>	int(11)	No		Id Alerta
<b>user_id</b>	int(11)	Sí	NULL	Id Usuario
<b>project_id</b>	int(11)	Sí	NULL	Id Proyecto
<b>last_alert</b>	datetime	Sí	NULL	Últ. Alerta
<b>deadline</b>	datetime	Sí	NULL	Plazo

Tabla 12: Diccionario de datos. Tabla “alerts”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Activo
<b>audit_id</b>	int(11)	Sí	NULL	Id Auditoría
<b>assetType_id</b>	int(11)	Sí	NULL	Id Tipo Activo
<b>status_id</b>	int(11)	Sí	NULL	Id Estatus

<b>client_id</b>	int(11)	Sí	NULL	Id Cliente
<b>name</b>	varchar(50)	Sí	NULL	Nombre
<b>description</b>	text	Sí	NULL	Descripción
<b>confidentiality_id</b>	int(11)	Sí	NULL	Id Confidencialidad
<b>integrity_id</b>	int(11)	Sí	NULL	Id Integridad
<b>availability_id</b>	int(11)	Sí	NULL	Id Disponibilidad
<b>audit_manager</b>	varchar(50)	Sí	NULL	Auditor

Tabla 13: Diccionario de datos. Tabla “assets”. Fuente: Bravo A. (2020)

<b>Columna</b>	<b>Tipo</b>	<b>Nulo</b>	<b>Predeterminado</b>	<b>Comentario</b>
<b>id</b>	int(11)	No		Id auditorías
<b>client_id</b>	int(11)	Sí	NULL	Id cliente
<b>status_id</b>	int(11)	Sí	NULL	Id estatus
<b>description</b>	text	Sí	NULL	Descripción
<b>start_date</b>	datetime	Sí	NULL	Fecha Inicio
<b>end_date</b>	datetime	Sí	NULL	Fecha Fin

Tabla 14: Diccionario de datos. Tabla “audits”. Fuente: Bravo A. (2020)

<b>Columna</b>	<b>Tipo</b>	<b>Nul</b>	<b>Predeterminado</b>	<b>Comentarios</b>
<b>id</b>	int(11)	No		Id Disponibilidad
<b>availability</b>	varchar(50)	Sí	NULL	Disponibilidad
<b>value</b>	Int(5)	Sí	NULL	Valor

Tabla 15: Diccionario de datos. Tabla “availability”. Fuente: Bravo A. (2020)

<b>Columna</b>	<b>Tipo</b>	<b>Nul</b>	<b>Predeterminado</b>	<b>Comentario</b>
<b>id</b>	int(11)	No		Id Calendario
<b>userId</b>	int(11)	Sí	NULL	Id Usuario
<b>dateFrom</b>	datetime	Sí	NULL	Fecha Desde

<b>dateTo</b>	datetime	Sí	NULL	Fecha Hasta
<b>description</b>	text	Sí	NULL	Descripción
<b>kind</b>	varchar(255)	Sí	NULL	Tipo
<b>allDay</b>	varchar(10)	Sí	NULL	Todo

Tabla 16: Diccionario de datos. Tabla “calendar”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Canvas
<b>title</b>	varchar(255)	Sí	NULL	Título
<b>author</b>	int(10)	Sí	NULL	Nombre autor
<b>created</b>	datetime	Sí	NULL	Creado
<b>projectId</b>	int(11)	Sí	NULL	Id proyecto
<b>type</b>	varchar(45)	Sí	NULL	Tipo

Tabla 17: Diccionario de datos. Tabla “canvas”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Canvas Items
<b>description</b>	Text	Sí	NULL	Descripción
<b>assumptions</b>	Text	Sí	NULL	Supuestos
<b>data</b>	Text	Sí	NULL	Datos
<b>conclusion</b>	Text	Sí	NULL	Conclusión
<b>box</b>	varchar(255)	Sí	NULL	Caja
<b>author</b>	int(11)	Sí	NULL	Autor
<b>created</b>	datetime	Sí	NULL	Creado
<b>modified</b>	datetime	Sí	NULL	Modificado
<b>canvasId</b>	int(11)	Sí	NULL	Id Canvas
<b>sortindex</b>	int(11)	Sí	NULL	Índice Ordenado
<b>status</b>	varchar(255)	Sí	NULL	Estatus
<b>milestoneId</b>	varchar(255)	Sí	NULL	Id Metas

Tabla 18: Diccionario de datos. Tabla “canvas\_items”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Cliente
<b>name</b>	varchar(200)	Sí	NULL	Nombre
<b>street</b>	varchar(200)	Sí	NULL	Calle
<b>zip</b>	int(10)	Sí	NULL	Código Postal
<b>city</b>	varchar(50)	Sí	NULL	Ciudad
<b>state</b>	varchar(50)	Sí	NULL	Estado
<b>country</b>	varchar(50)	Sí	NULL	Pais
<b>phone</b>	varchar(50)	Sí	NULL	Telefono
<b>internet</b>	varchar(200)	Sí	NULL	Internet
<b>published</b>	int(1)	Sí	NULL	Publicado
<b>age</b>	int(3)	Sí	NULL	Año
<b>email</b>	varchar(255)	Sí	NULL	Correo

Tabla 19: Diccionario de datos. Tabla “clients”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Comentario
<b>module</b>	enum('project', 'ticket', 'client', 'user', 'lead', 'leancanvasitem', 'idea', 'retrospective')	Sí	NULL	Modulo
<b>userId</b>	int(11)	Sí	NULL	Id Usuario
<b>commentParent</b>	int(11)	Sí	NULL	Comentarios Respuesta

<b>date</b>	datetime	Sí	NULL	Fecha
<b>moduleId</b>	int(11)	Sí	NULL	Id Módulos
<b>text</b>	Text	Sí	NULL	Comentario

Tabla 20: Diccionario de datos. Tabla “comment”. Fuente: Bravo A. (2020)

Columna	Tipo	Nul o	Predeterminado	Comentarios
<b>id</b>	int(11)	No		Id Confidencialidad
<b>confidentiality</b>	varchar(50)	Sí	NULL	Confidencialidad
<b>value</b>	Int(5)	Sí	NULL	Valor

Tabla 21: Diccionario de datos. Tabla “confidentialies”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Instrumento
<b>value</b>	Text	Sí	NULL	Valor
<b>user_id</b>	int(11)	Sí	NULL	Id Usuario

Tabla 22: Diccionario de datos. Tabla “dashboard\_widgets”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Archivo
<b>module</b>	enum('project', 'ticket', 'client', 'user', 'lead', 'export', 'private')	Sí	NULL	Modulo
<b>moduleId</b>	int(11)	Sí	NULL	Id Modulo
<b>userId</b>	int(11)	Sí	NULL	Id Usuario
<b>extension</b>	varchar(10)	Sí	NULL	Extensión
<b>encName</b>	varchar(25)	Sí	NULL	Nombre Alterno

	5)			
<b>realName</b>	varchar(255)	Sí	NULL	Nombre Real
	5)			
<b>date</b>	datetime	Sí	NULL	Fecha

Tabla 23: Diccionario de datos. Tabla “file”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Enlace
<b>userId</b>	int(255)	Sí	NULL	Id Usuario
<b>url</b>	Text	Sí	NULL	Url
<b>name</b>	varchar(255)	Sí	NULL	Nombre del Enlace
<b>colorClass</b>	varchar(100)	Sí	NULL	Clase de Color

Tabla 24: Diccionario de datos. Tabla “gcallinks”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Impacto
<b>impact</b>	varchar(50)	Sí	NULL	Impacto
<b>value</b>	int(5)	Sí	NULL	Valor

Tabla 25: Diccionario de datos. Tabla “impacts”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Integridad
<b>integrity</b>	varchar(50)	Sí	NULL	Integridad
<b>value</b>	int(5)	Sí	NULL	Valor

Tabla 26: Diccionario de datos. Tabla “integrities”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Digente
<b>name</b>	varchar(255)	Sí	NULL	Nombre
<b>status</b>	enum('lead',	Sí	NULL	Estatus

	'opportunity', 'client')			
<b>refSource</b>	varchar(100)	Sí	NULL	Fuente de Referencia
<b>refValue</b>	varchar(255)	Sí	NULL	Valor de Referencia
<b>potentialMone</b> y	int(11)	Sí	NULL	Dinero Potencial
<b>actualMoney</b>	int(11)	Sí	NULL	Dinero Real
<b>clientId</b>	int(11)	Sí	NULL	Id Cliente
<b>proposal</b>	text	Sí	NULL	Propuesta
<b>creatorId</b>	int(11)	Sí	NULL	Id Creador
<b>date</b>	datetime	Sí	NULL	Fecha

Tabla 27: Diccionario de datos. Tabla “lead”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Mensaje
<b>to_id</b>	varchar(50)	Sí	NULL	Id hasta
<b>from_id</b>	int(11)	Sí	NULL	Id Desde
<b>parent_id</b>	int(11)	Sí	NULL	Id parentesco
<b>subject</b>	varchar(255)	Sí	NULL	Sujeto
<b>content</b>	Text	Sí	NULL	Contenido
<b>date_sent</b>	datetime	Sí	NULL	Fecha de Enviado
<b>last_message</b>	int(1)	Sí	0	Ultimo Mensaje
<b>read</b>	int(1)	Sí	0	Leer

Tabla 28: Diccionario de datos. Tabla “message”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Permisos de Modulo
<b>module</b>	Text	Sí	NULL	Modulo
<b>roleIds</b>	Text	Sí	NULL	Id Roles

Tabla 29: Diccionario de datos. Tabla “modulerights”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Nota
<b>userId</b>	int(11)	Sí	NULL	Id Usuario
<b>title</b>	varchar(255)	Sí	NULL	Título
<b>description</b>	Text	Sí	NULL	Descripción

Tabla 30: Diccionario de datos. Tabla “note”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Probabilidad
<b>probability</b>	varchar(50)	Sí	NULL	Probabilidad
<b>value</b>	int(5)	Sí	NULL	Valor

Tabla 31: Diccionario de datos. Tabla “probabilities”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Proyecto
<b>name</b>	varchar(100)	Sí	NULL	Nombre
<b>clientId</b>	int(100)	Sí	NULL	Id Cliente
<b>details</b>	Text	Sí	NULL	Detalle
<b>state</b>	int(2)	Sí	NULL	Estado
<b>hourBudget</b>	varchar(255)	No		Horas Propuestas
<b>dollarBudget</b>	int(11)	Sí	NULL	Propuesta de Dinero
<b>active</b>	int(11)	Sí	NULL	Activo

Tabla 32: Diccionario de datos. Tabla “projects”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id tiempo
<b>userId</b>	int(11)	No		Id Usuario
<b>minutes</b>	int(11)	Sí	NULL	Minutos
<b>hours</b>	int(11)	Sí	NULL	Horas
<b>punchIn</b>	int(11)	Sí	NULL	

Tabla 33: Diccionario de datos. Tabla “punch\_clock”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Lectura
<b>module</b>	enum('ticket', 'message')	Sí	NULL	Modulo
<b>moduleId</b>	int(11)	Sí	NULL	Id Modulo
<b>userId</b>	int(11)	Sí	NULL	Id Usuario

Tabla 34: Diccionario de datos. Tabla “read”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Relación Usuario Proyecto
<b>userId</b>	int(11)	Sí	NULL	Id Usuario
<b>projectId</b>	int(11)	Sí	NULL	Id proyecto
<b>wage</b>	int(11)	Sí	NULL	Salario

Tabla 35: Diccionario de datos. Tabla “relationuserproject”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Riesgo
<b>risk</b>	text	Sí	NULL	Riesgo
<b>threat_id</b>	int(11)	Sí	NULL	Id Amenaza
<b>probability_id</b>	int(11)	Sí	NULL	Id Probabilidad
<b>impact_id</b>	int(11)	Sí	NULL	Id Impacto
<b>asset_id</b>	int(11)	Sí	NULL	Id Activo
<b>audit_id</b>	int(11)	Sí	NULL	Id Auditoría

Tabla 36: Diccionario de datos. Tabla “risk\_level”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>Id</b>	int(11)	No		Id Roles
<b>roleName</b>	varchar(255)	No		Nombre Rol
<b>roleDescription</b>	varchar(255)	Sí	NULL	descripción del Rol
<b>sysOrg</b>	int(11)	Sí	NULL	Organización
<b>Template</b>	varchar(100)	Sí	NULL	Modelo

Tabla 37: Diccionario de datos. Tabla “roles”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
key	varchar(175)	No		Llave
value	Text	Sí	NULL	Valor

Tabla 38: Diccionario de datos. Tabla “settings”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
id	int(11)	No		Id Sprints
projectId	int(11)	Sí	NULL	Id Proyecto
name	varchar(45)	Sí	NULL	Nombre
startDate	datetime	Sí	NULL	Fecha Inicio
endDate	datetime	Sí	NULL	Fecha Fin

Tabla 39: Diccionario de datos. Tabla “sprints”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
sprintId	int(11)	Sí	NULL	Id Sprint
projectId	int(11)	Sí	NULL	Id Proyecto
date	datetime	Sí	NULL	Fecha
sum_todos	int(11)	Sí	NULL	Suma Todo
sum_open_todos	int(11)	Sí	NULL	Suma Abierto Todo
sum_progres_todos	int(11)	Sí	NULL	Suma Progreso Todo
sum_closed_todos	int(11)	Sí	NULL	Suma Cierre Todo
sum_planned_hours	int(11)	Sí	NULL	Suma Plan Horas
sum_estrengthening_hours	int(11)	Sí	NULL	Suma Estremecimiento de horas
sum_logged_hours	int(11)	Sí	NULL	Suma Registro de Horas
sum_points	int(11)	Sí	NULL	Suma Puntos
sum_points_done	int(11)	Sí	NULL	Suma Puntos hechos
sum_points_progress	int(11)	Sí	NULL	Suma Puntos Progreso
sum_points_op	int(11)	Sí	NULL	Suma Puntos Abiertos

<b>en</b>				
<b>sum_todos_xs</b>	int(11)	Sí	NULL	Suma Todos XS
<b>sum_todos_s</b>	int(11)	Sí	NULL	Suma Todos S
<b>sum_todos_m</b>	int(11)	Sí	NULL	Suma Todos M
<b>sum_todos_l</b>	int(11)	Sí	NULL	Suma Todos L
<b>sum_todos_xl</b>	int(11)	Sí	NULL	Suma Todos XL
<b>sum_todos_xxl</b>	int(11)	Sí	NULL	Suma Todos XXL
<b>sum_todos_none</b>	int(11)	Sí	NULL	Suma Todos Ninguno
<b>ne</b>				
<b>tickets</b>	Text	Sí	NULL	Ticket
<b>daily_avg_hours_booked_today</b>	int(11)	Sí	NULL	Horario Medio diario Reservado
<b>daily_avg_hours_booked_point</b>	int(11)	Sí	NULL	Horas Diarias de puntos
<b>daily_avg_hours_planned_today</b>	int(11)	Sí	NULL	Horas Promedio diario Planificadas
<b>daily_avg_hours_planned_point</b>	int(11)	Sí	NULL	Horas Promedio diario de puntos Planificadas
<b>daily_avg_hours_remaining_point</b>	int(11)	Sí	NULL	Horas Promedios diario restantes de punto
<b>daily_avg_hours_remaining_today</b>	int(11)	Sí	NULL	Horas Promedios diarias restantes

Tabla 40: Diccionario de datos. Tabla “stats”. Fuente: Bravo A. (2020)

<b>Columna</b>	<b>Tipo</b>	<b>Nulo</b>	<b>Predeterminado</b>	<b>Comentario</b>
<b>id</b>	int(11)	No		Id Estatus
<b>status</b>	varchar(50)	Sí	NULL	Estatus

Tabla 41: Diccionario de datos. Tabla “statuses”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id SubMódulos
<b>alias</b>	varchar(155)	Sí	NULL	Alias
<b>title</b>	varchar(155)	Sí	NULL	Titulo
<b>module</b>	varchar(100)	Sí	NULL	Modulo
<b>submodule</b>	varchar(150)	Sí	NULL	SubModulo
<b>roleIds</b>	varchar(50)	Sí	NULL	Id Rol

Tabla 42: Diccionario de datos. Tabla “submodulerights”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Amenaza
<b>threatType_id</b>	int(11)	Sí	NULL	Id Tipo de Amenaza
<b>description</b>	text	Sí	NULL	Descripción

Tabla 43: Diccionario de datos. Tabla “threats”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Tipo de Amenzada
<b>description</b>	text	Sí	NULL	Descripción

Tabla 44: Diccionario de datos. Tabla “threats”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Ticket Historia
<b>userId</b>	int(11)	Sí	NULL	Id Usuario
<b>ticketId</b>	int(11)	Sí	NULL	Id Ticket
<b>changeType</b>	varchar(255)	Sí	NULL	Cambio Tipo
<b>changeValue</b>	varchar(150)	Sí	NULL	Cambio Valor
<b>dateModified</b>	datetime	Sí	NULL	Fecha Modificado

Tabla 45: Diccionario de datos. Tabla “tickethistory”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		
<b>projectId</b>	int(11)	Sí	NULL	Id Proyecto
<b>headline</b>	varchar(25)	Sí	NULL	Titulo

	5)			
<b>description</b>	Text	Sí	NULL	Descripción
<b>acceptanceCriteria</b>	Text	Sí	NULL	Criterios de Aceptación
<b>date</b>	datetime	Sí	NULL	Fecha
<b>dateToFinish</b>	datetime	Sí	NULL	Fecha Fin
<b>priority</b>	varchar(60)	Sí	NULL	Prioridad
	)			
<b>status</b>	int(2)	Sí	NULL	Estatus
<b>userId</b>	int(11)	Sí	NULL	Id Usuario
<b>os</b>	varchar(30)	Sí	NULL	Os
	)			
<b>browser</b>	varchar(30)	Sí	NULL	Navegador
	)			
<b>resolution</b>	varchar(30)	Sí	NULL	Resolución
	)			
<b>component</b>	varchar(100)	Sí	NULL	Componente
	)			
<b>version</b>	varchar(20)	Sí	NULL	Versión
	)			
<b>url</b>	varchar(100)	Sí	NULL	Ulr
	)			
<b>dependingTicketId</b>	int(100)	Sí	NULL	Depende ticket id
<b>editFrom</b>	datetime	Sí	NULL	Editar Desde
<b>editTo</b>	datetime	Sí	NULL	Editar Hasta
<b>editorId</b>	varchar(75)	Sí	NULL	Id Editor
	)			
<b>planHours</b>	int(10)	Sí	0	Plan de Horas
<b>hourRemaining</b>	int(10)	Sí	NULL	Horas Restante
<b>type</b>	varchar(255)	Sí	NULL	Tipo
	)			
<b>production</b>	int(1)	Sí	0	Producción
<b>staging</b>	int(1)	Sí	0	Puesto
<b>storypoints</b>	Float	Sí	NULL	Punto de Historia
<b>sprint</b>	int(100)	Sí	NULL	Sprint
<b>sortindex</b>	bigint(20)	Sí	NULL	Índice Clasificado
<b>kanbanSortIndex</b>	bigint(20)	Sí	NULL	Orden de Índice Kanban
<b>tags</b>	varchar(255)	Sí	NULL	Etiqueta
	)			
<b>milestoneid</b>	int(11)	Sí	NULL	Id Hito

<b>leancanvasitemid</b>	int(11)	Sí	NULL	Id de LeanCanvas
<b>retrospectiveid</b>	int(11)	Sí	NULL	Id Retrospectiva
<b>ideaid</b>	int(11)	Sí	NULL	Id Idea
<b>ticketscol</b>	varchar(45)	Sí	NULL	Ticket Col

Tabla 46: Diccionario de datos. Tabla “tickets”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(255)	No		Id Tiempo Hoja
<b>userId</b>	int(11)	Sí	NULL	Id Usuario
<b>ticketId</b>	int(11)	Sí	NULL	Id Ticket
<b>workDate</b>	datetime	Sí	NULL	Fecha Trabajo
<b>hours</b>	Float	Sí	NULL	Horas
<b>description</b>	Text	Sí	NULL	Descripción
<b>kind</b>	varchar(175)	Sí	NULL	tipo
<b>invoicedEmpl</b>	int(2)	Sí	NULL	Facturas Empleado
<b>invoicedCom</b>	int(2)	Sí	NULL	Facturas compra
<b>invoicedEmplDate</b>	datetime	Sí	NULL	Fecha Factura Empleado
<b>invoicedComDate</b>	datetime	Sí	NULL	Fecha Factura Compra
<b>rate</b>	varchar(255)	Sí	NULL	Velocidad

Tabla 47: Diccionario de datos. Tabla “timesheets”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Usuario
<b>username</b>	varchar(175)	No		Nombre Usuario
<b>password</b>	varchar(255)	No		Contraseña
<b>firstname</b>	varchar(100)	No		Segundo Nombre
<b>lastname</b>	varchar(100)	No		Apellido
<b>phone</b>	varchar(25)	No		Teléfono
<b>profileId</b>	varchar(100)	No		Id Perfil
<b>lastlogin</b>	datetime	Sí	NULL	Ultimo Acceso
<b>lastpwd_change</b>	int(11)	Sí	NULL	Cambio Contraseña

<b>status</b>	varchar(1)	No	A	Estatus
<b>expires</b>	int(11)	Sí	NULL	Expira
<b>role</b>	varchar(200)	No		Rol
<b>session</b>	varchar(100)	Sí	NULL	Sesión
<b>sessiontime</b>	varchar(50)	Sí	NULL	Tiempo de Sesión
<b>wage</b>	int(11)	Sí	NULL	
<b>hours</b>	int(11)	Sí	NULL	Horas
<b>description</b>	Text	Sí	NULL	Descripción
<b>clientId</b>	int(11)	Sí	NULL	Id Cliente
<b>notifications</b>	int(2)	Sí	NULL	Notificación
<b>pwReset</b>	varchar(100)	Sí	NULL	Reiniciar Contraseña
<b>pwResetExpiration</b>	datetime	Sí	NULL	Reiniciar Contraseña Expirada
<b>settings</b>	Text	Sí	NULL	Ajustes

Tabla 48: Diccionario de datos. Tabla “user”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Wiki
<b>name</b>	Text	Sí	NULL	Nombre
<b>projectId</b>	int(11)	Sí	NULL	Id Proyecto
<b>authorId</b>	int(11)	Sí	NULL	Id Autor
<b>date</b>	datetime	Sí	NULL	Fecha
<b>modified</b>	datetime	Sí	NULL	Fecha Modificada

Tabla 49: Diccionario de datos. Tabla “wiki”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Artículo Wiki
<b>headline</b>	Text	Sí	NULL	Ayuda
<b>text</b>	Text	Sí	NULL	Texto
<b>tags</b>	Text	Sí	NULL	Etiqueta
<b>authorId</b>	int(255)	Sí	NULL	Id Autor
<b>category</b>	varchar(255)	Sí	NULL	Categoría
<b>date</b>	datetime	Sí	NULL	Fecha
<b>modified</b>	datetime	Sí	NULL	Fecha Modificar

Tabla 50: Diccionario de datos. Tabla “wiki\_articles”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(11)	No		Id Caterigoria Wiki
<b>name</b>	varchar(255)	Sí	NULL	Nombre

Tabla 51: Diccionario de datos. Tabla “wiki\_categories”. Fuente: Bravo A. (2020)

Columna	Tipo	Nulo	Predeterminado	Comentario
<b>id</b>	int(255)	No		Id Comentarios
<b>text</b>	Text	Sí	NULL	texto
<b>userId</b>	int(255)	Sí	NULL	Id Usuario
<b>articleId</b>	int(255)	Sí	NULL	Id Articulo
<b>date</b>	datetime	Sí	NULL	Fecha
<b>commentParent</b>	int(100)	Sí	NULL	Comentario Parentesco

Tabla 52: Diccionario de datos. Tabla “wiki\_comments”. Fuente: Bravo A. (2020)

### 4.3 Fase III: Desarrollo

En esta fase, la mayor parte del trabajo está representado por el desarrollo en programación del software y da como resultado un producto listo para correr en producción.

#### 4.3.1 Desarrollo del sistema

A continuación, se presentan representaciones gráficas en la que se muestran las relaciones entre las diferentes partes del sistema, la cuales corresponden a la arquitectura del software observado desde los distintos roles de usuario, por medio de mapas de navegación, para desarrollar la interacción de componentes de forma gráfica

#### 4.3.2 Diseño de interfaces

Las siguientes interfaces fueron diseñadas y desarrolladas tomando en cuenta los criterios expuestos en la entrevista. Todo esto permitió que las mismas sean agradables para la vista, rápidas y con un alto contenido de información sin estar sobrecargadas.



Figura 5: “Pantalla Principal Login”. Autor: Bravo A. (2020)



Figura 6: “Pantalla Reestablecer Contraseña”. Autor: Bravo A. (2020)



Figura 7: “Pantalla Principal Login Incorrecto”. Autor: Bravo A. (2020)

Si no se cuenta con ning3n proyecto asignado, el usuario desarrollador tendr3 esta vista:

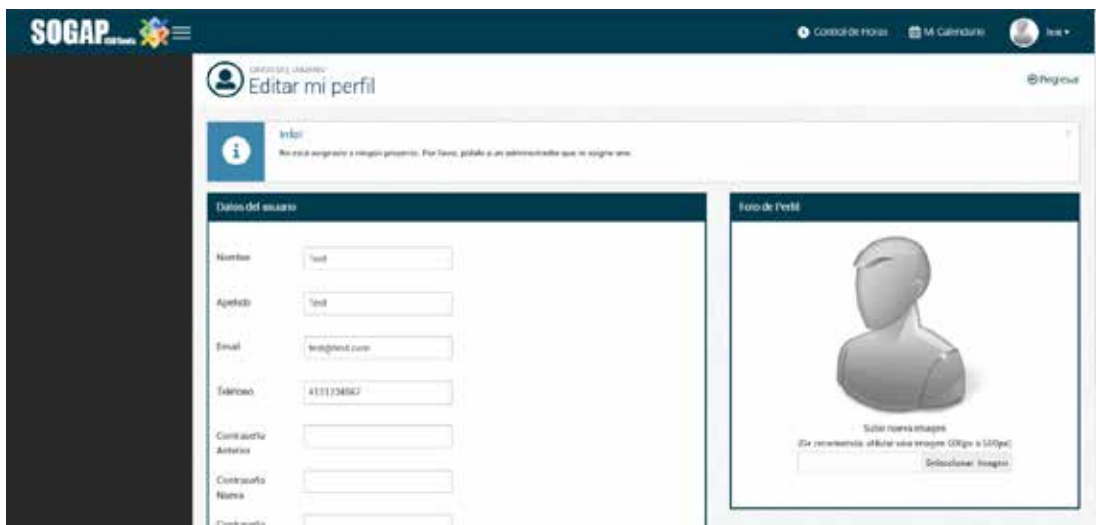


Figura 8: “Pantalla Principal Rol Desarrollador”. Autor: Bravo A. (2020)



Figura 9: “Pantalla Principal Rol Administrador”. Autor: Bravo A. (2020)

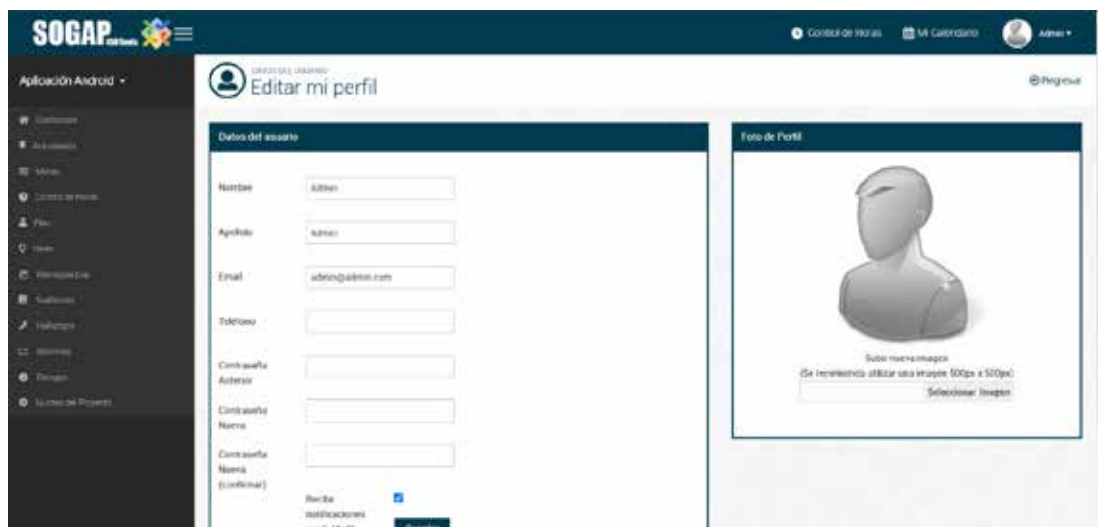


Figura 10: “Pantalla Editar Perfil”. Autor: Bravo A. (2020)



Figura 11: “Menú de acceso a opciones: Perfil, Proyectos, Clientes/Productos, Usuarios, Ajustes, Contáctanos, Salir”. Autor: Bravo A. (2020)

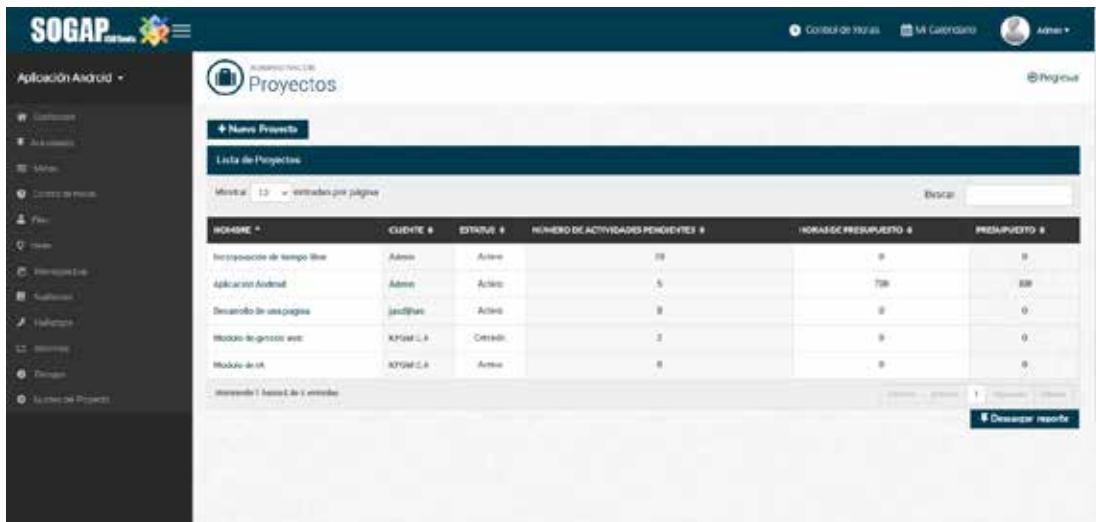


Figura 12: “Pantalla Proyectos”. Autor: Bravo A. (2020)

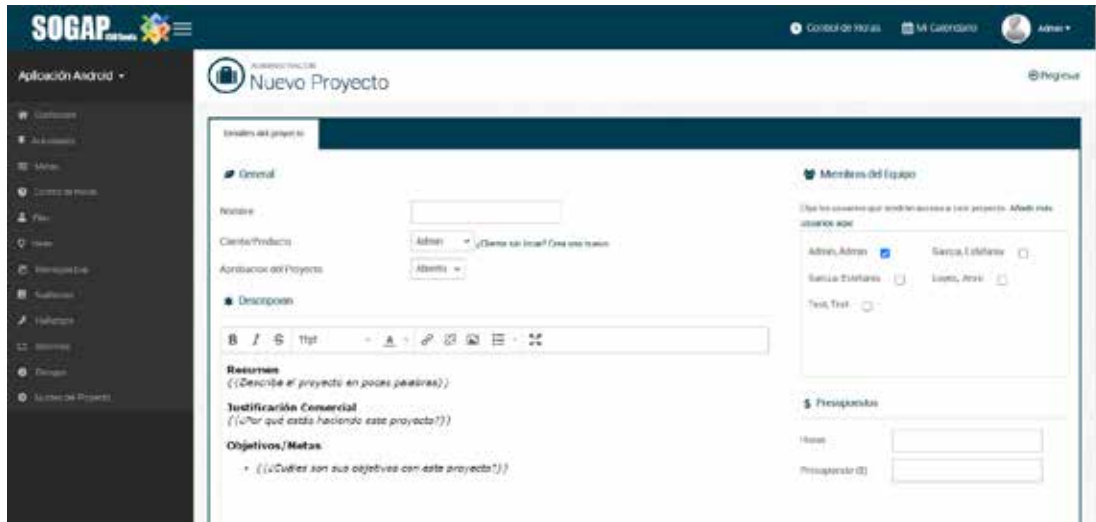


Figura 13: “Pantalla Crear Nuevo Proyecto”. Autor: Bravo A. (2020)

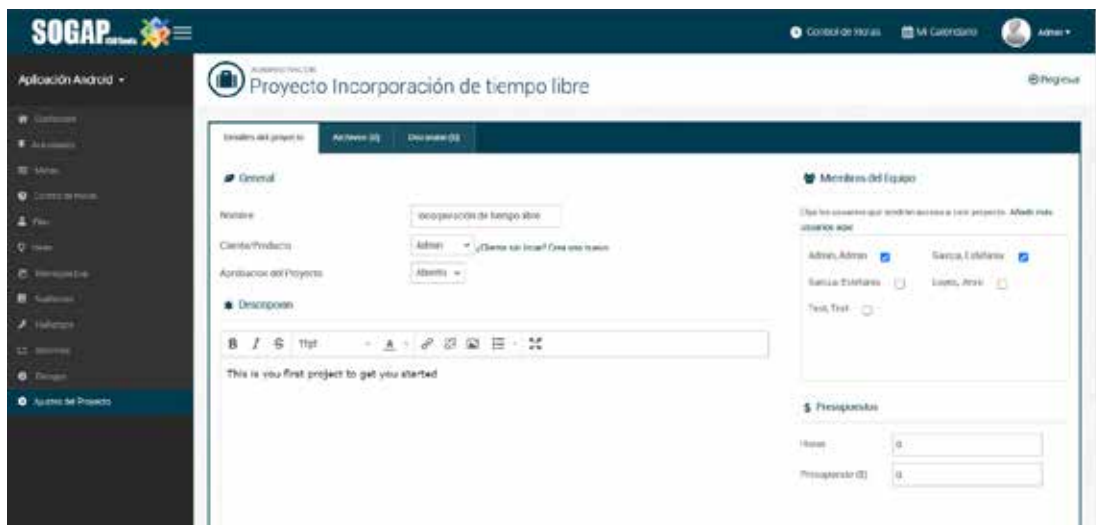


Figura 14: “Pantalla Editar Detalles de Proyecto”. Autor: Bravo A. (2020)



Figura 15: “Pantalla Editar Archivos de Proyecto”. Autor: Bravo A. (2020)

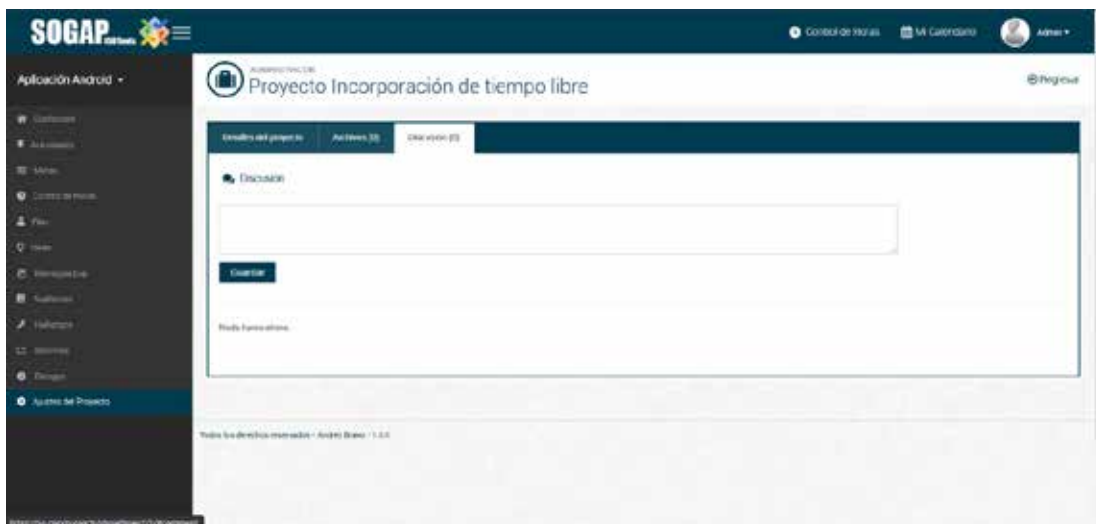


Figura 16: “Pantalla Discusión de Proyecto”. Autor: Bravo A. (2020)

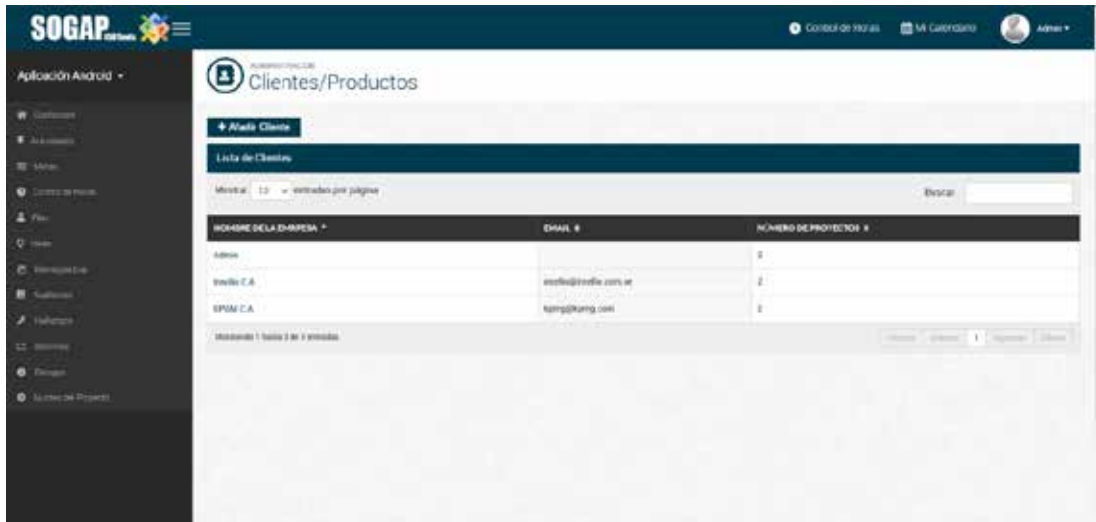


Figura 17: “Pantalla Clientes/Productos”. Autor: Bravo A. (2020)

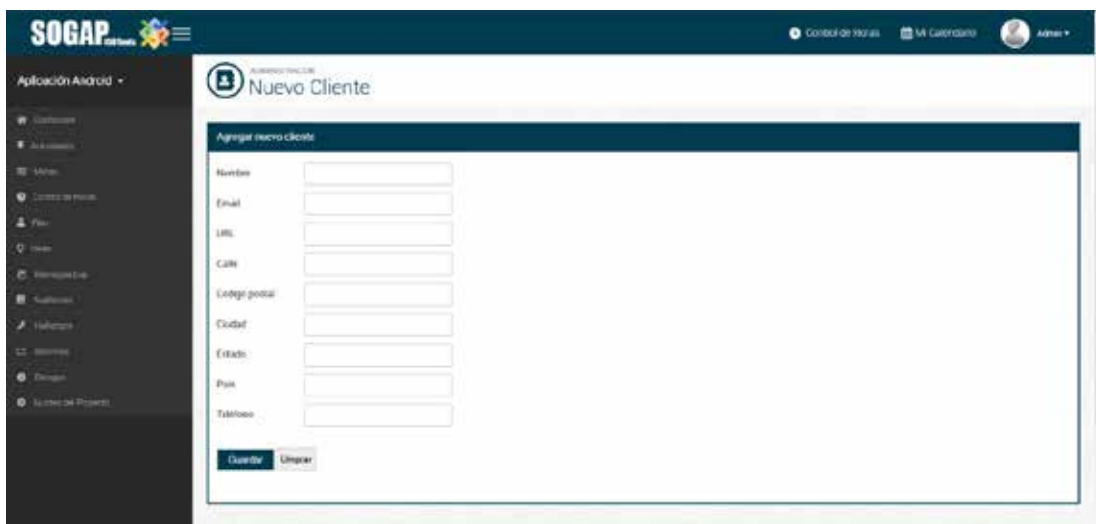


Figura 18: “Pantalla Crear Nuevo Cliente/Producto”. Autor: Bravo A. (2020)



Figura 19: “Pantalla Editar Cliente/Productos”. Autor: Bravo A. (2020)

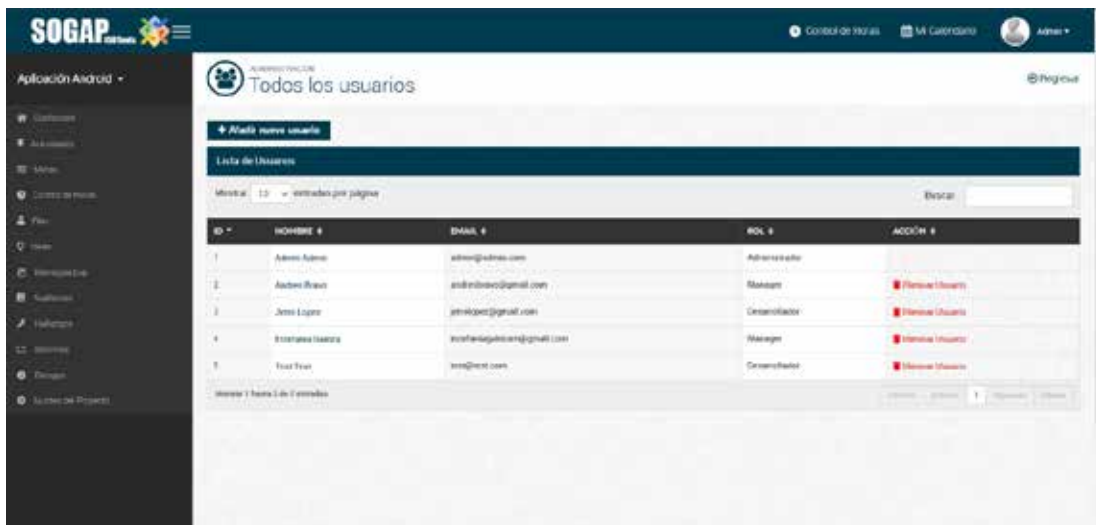


Figura 20: “Pantalla Usuarios”. Autor: Bravo A. (2020)

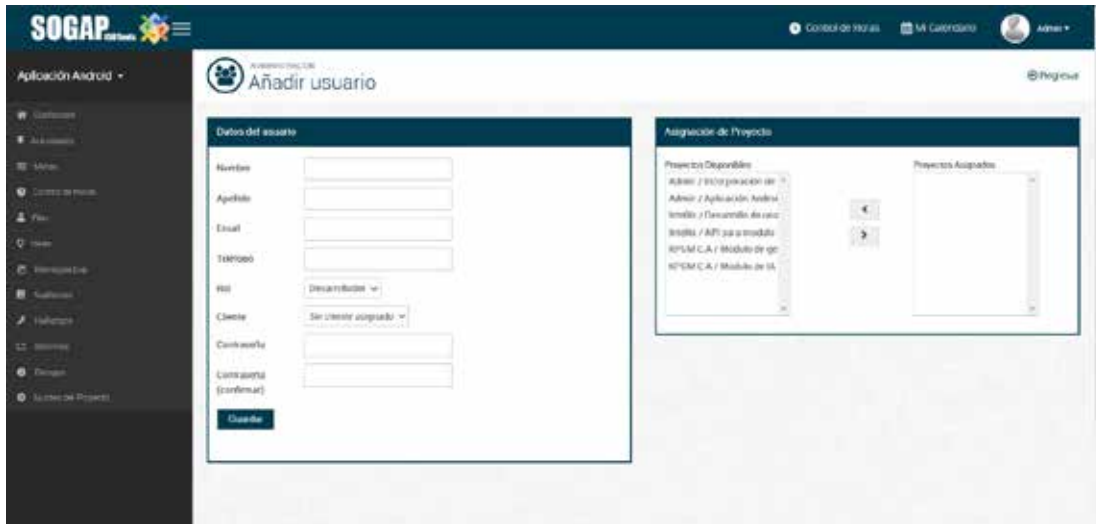


Figura 21: “Pantalla Añadir Usuario”. Autor: Bravo A. (2020)

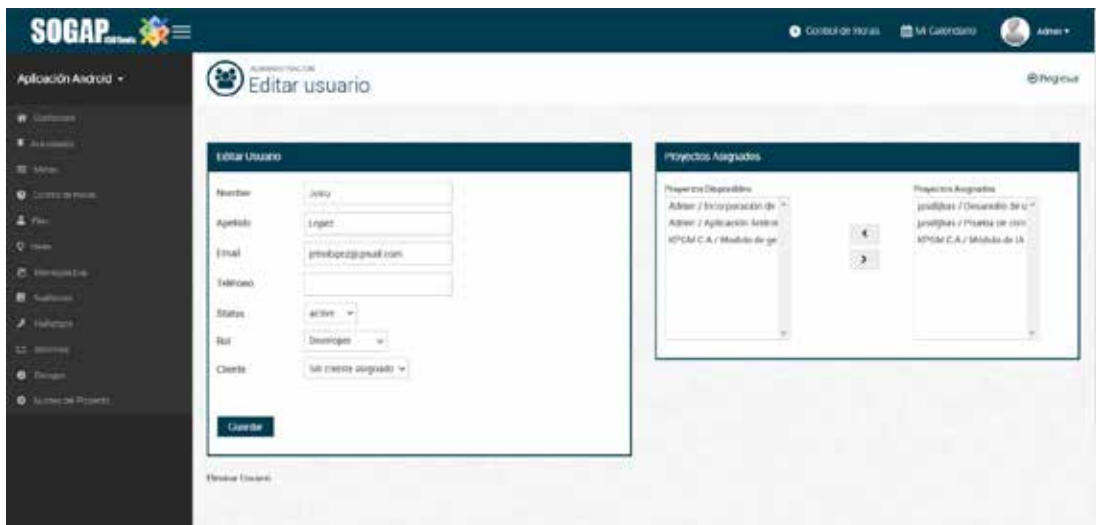


Figura 22: “Pantalla Editar Usuario”. Autor: Bravo A. (2020)

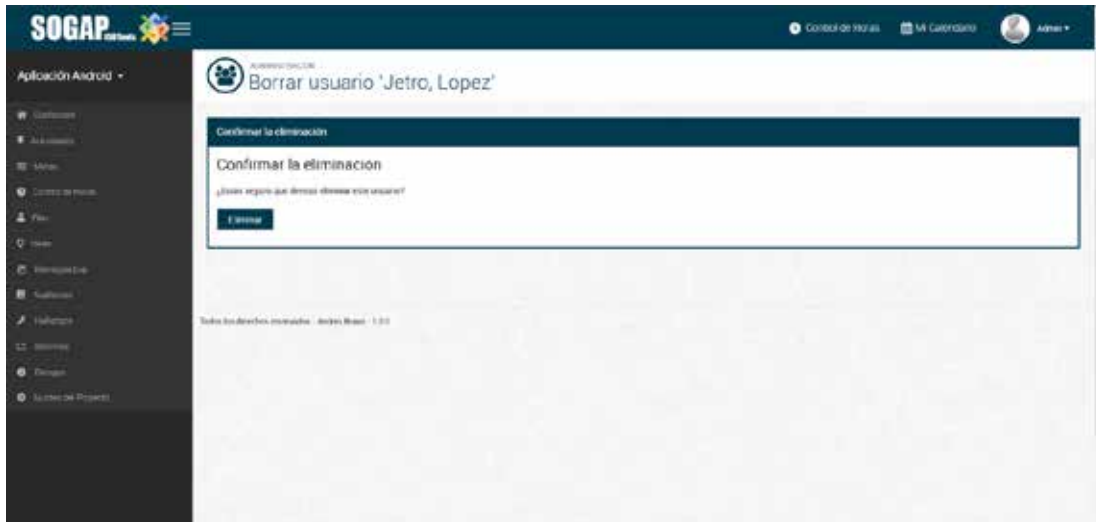


Figura 23: “Pantalla Eliminar Usuario”. Autor: Bravo A. (2020)



Figura 24: “Pantalla Ajustes de la Empresa”. Autor: Bravo A. (2020)



Figura 25: “Pantalla Dashboard. Actividades, Gráficas, Avances, Backlog, Metas, Reporte”.

Autor: Bravo A. (2020)

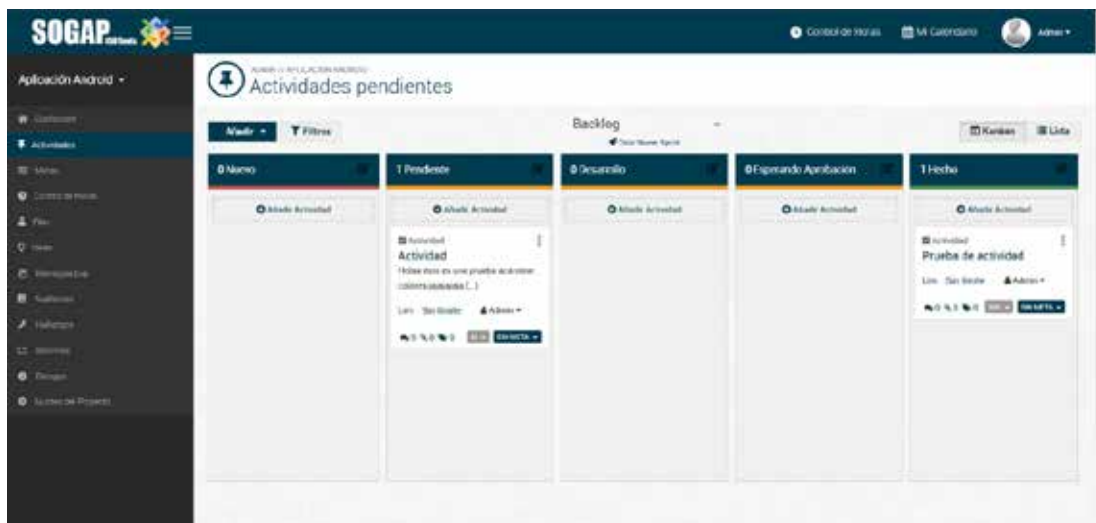


Figura 26: “Pantalla Actividades Pendientes. Modo Tablero Kanban”.



Figura 27: “Pantalla Actividades Pendientes. Modo Lista”. Autor: Bravo A. (2020)

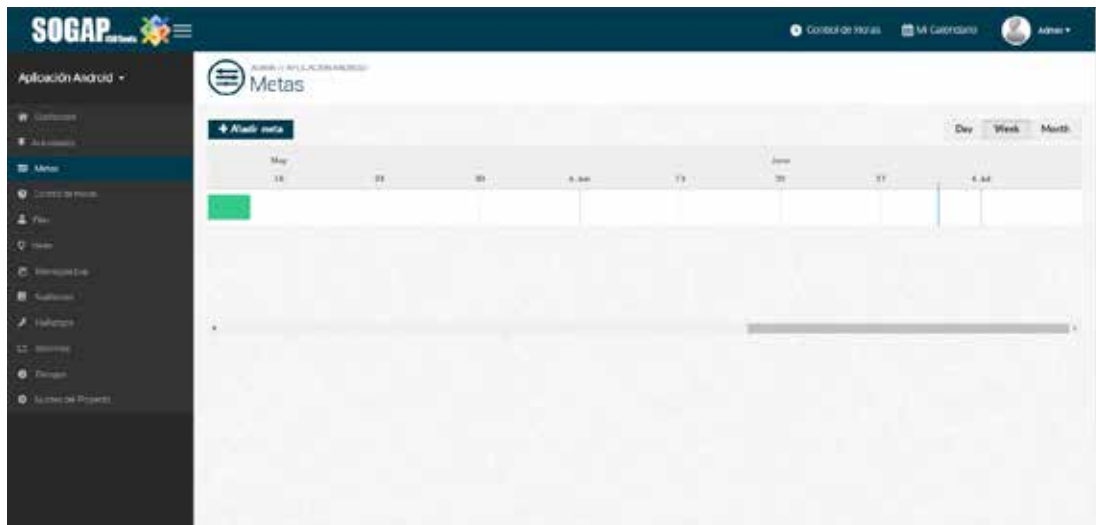


Figura 28: “Pantalla Metas. Vista por Mes”. Autor: Bravo A. (2020)

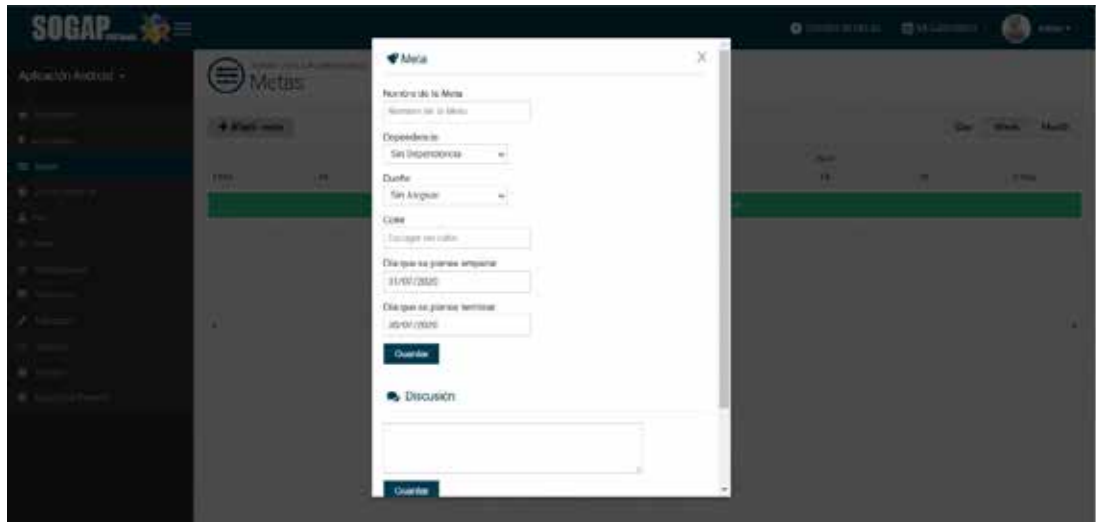


Figura 29: “Pantalla Añadir Meta”. Autor: Bravo A. (2020)

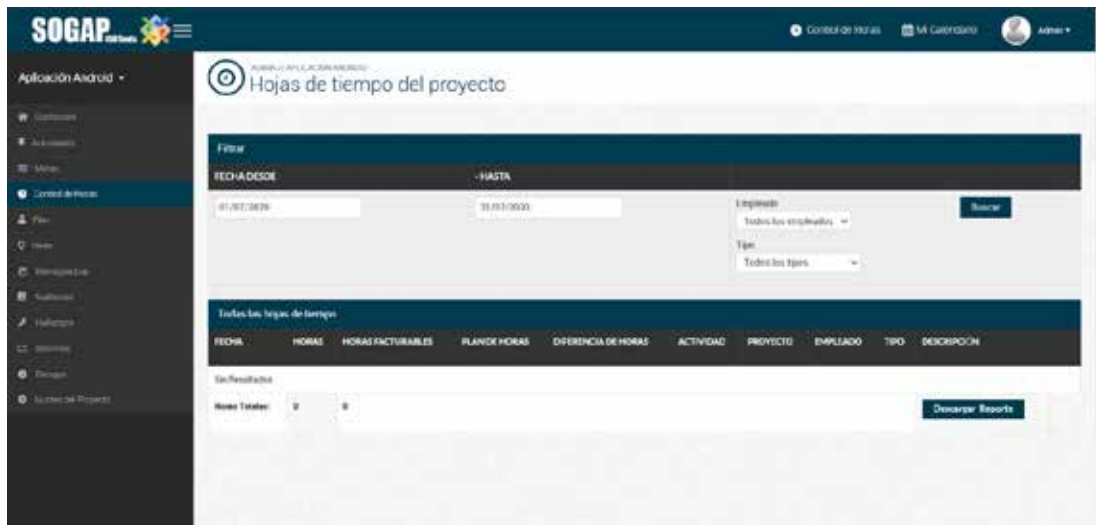


Figura 30: “Pantalla Control de Horas del Proyecto”. Autor: Bravo A. (2020)

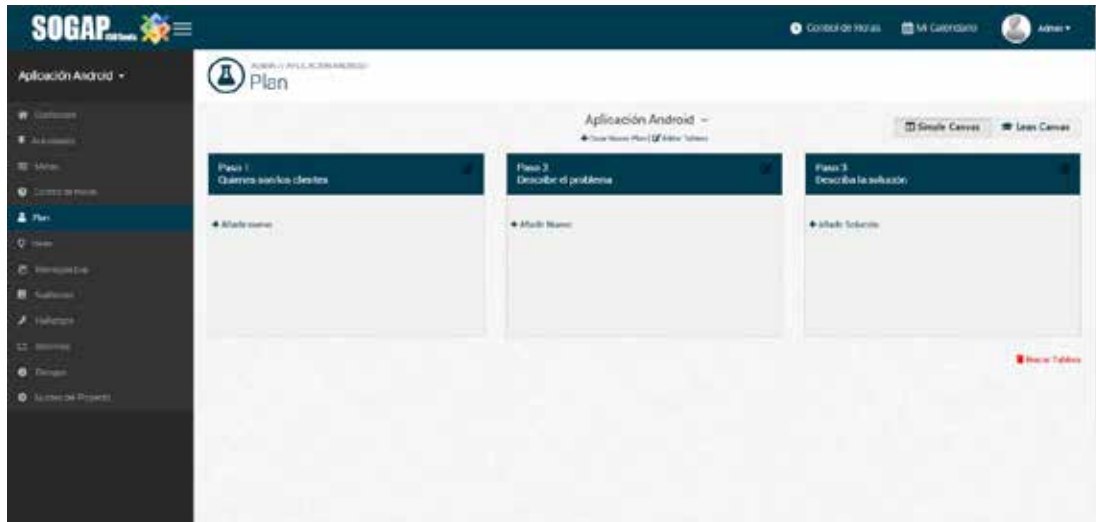


Figura 31: “Pantalla Plan. Vista Simple Canvas”. Autor: Bravo A. (2020)

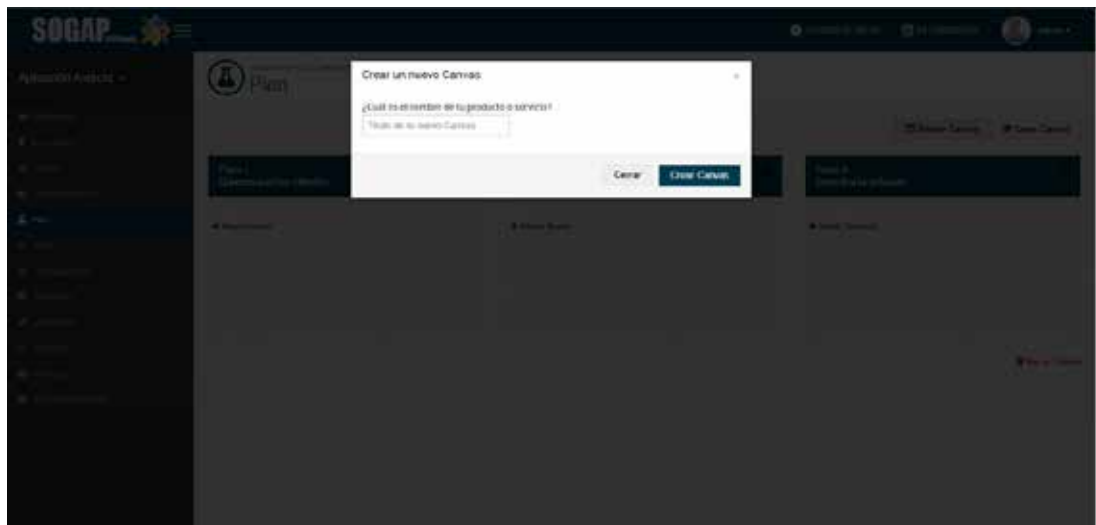


Figura 32: “Pantalla Añadir Plan”. Autor: Bravo A. (2020)



Figura 33: “Pantalla Editar Plan”. Autor: Bravo A. (2020)

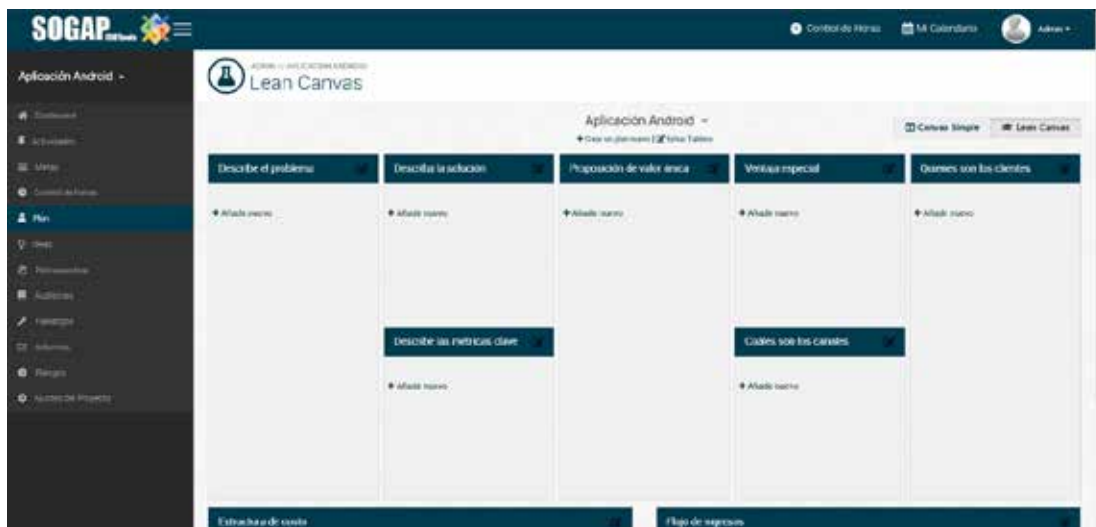


Figura 34: “Pantalla Plan. Vista Lean Canvas”. Autor: Bravo A. (2020)

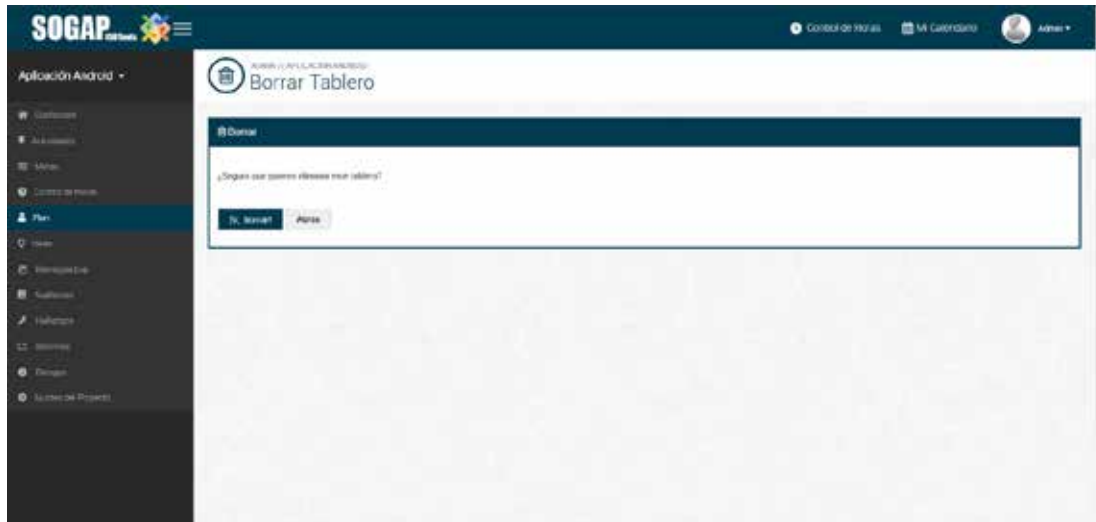


Figura 35: “Pantalla Eliminar Tablero de Plan”. Autor: Bravo A. (2020)

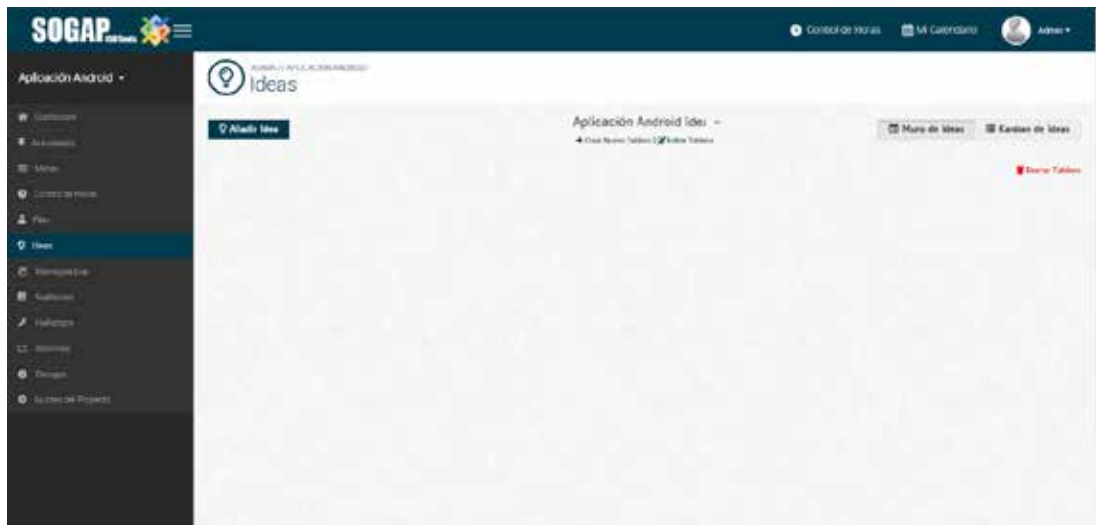


Figura 36: “Pantalla Ideas. Vista Muro de Ideas”. Autor: Bravo A. (2020)

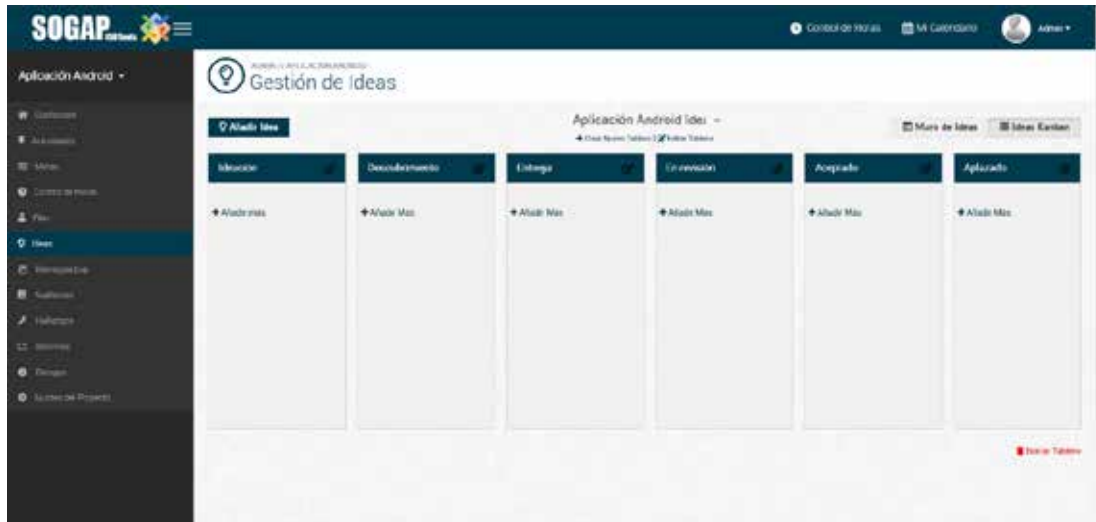


Figura 37: “Pantalla Ideas. Vista Tablero Kanban”. Autor: Bravo A. (2020)



Figura 38: “Pantalla Crear Tablero de Ideas”. Autor: Bravo A. (2020)



Figura 39: “Pantalla Editar Tablero de Ideas”. Autor: Bravo A. (2020)

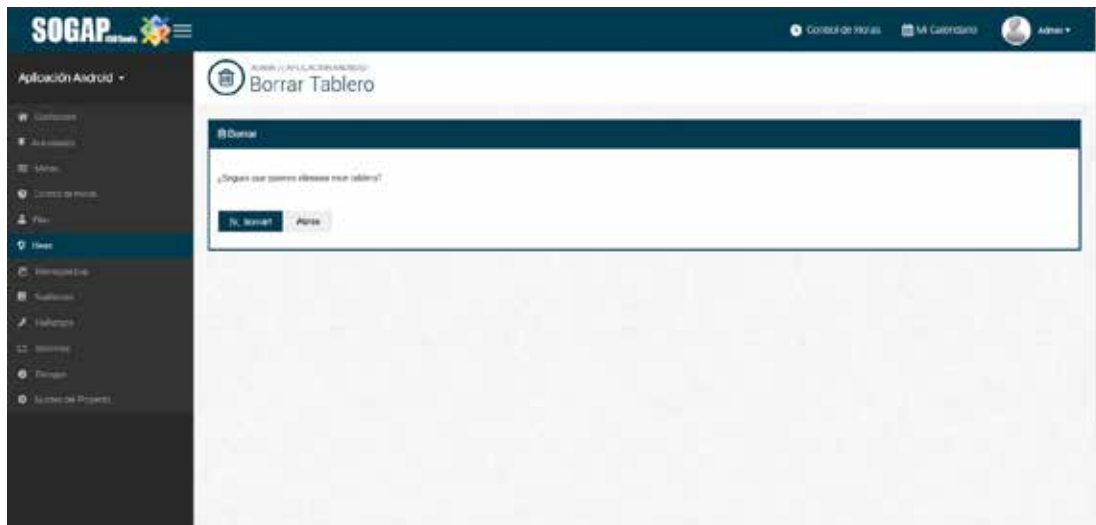


Figura 40: “Pantalla Eliminar Tablero de Ideas”. Autor: Bravo A. (2020)

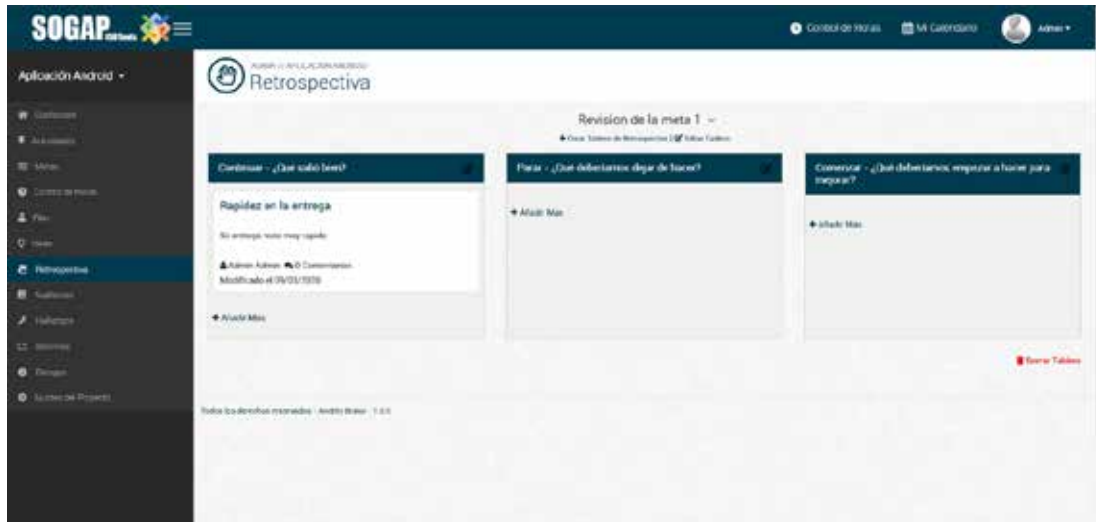


Figura 41: “Pantalla Retrospectiva”. Autor: Bravo A. (2020)



Figura 42: “Pantalla Crear Retrospectiva”. Autor: Bravo A. (2020)

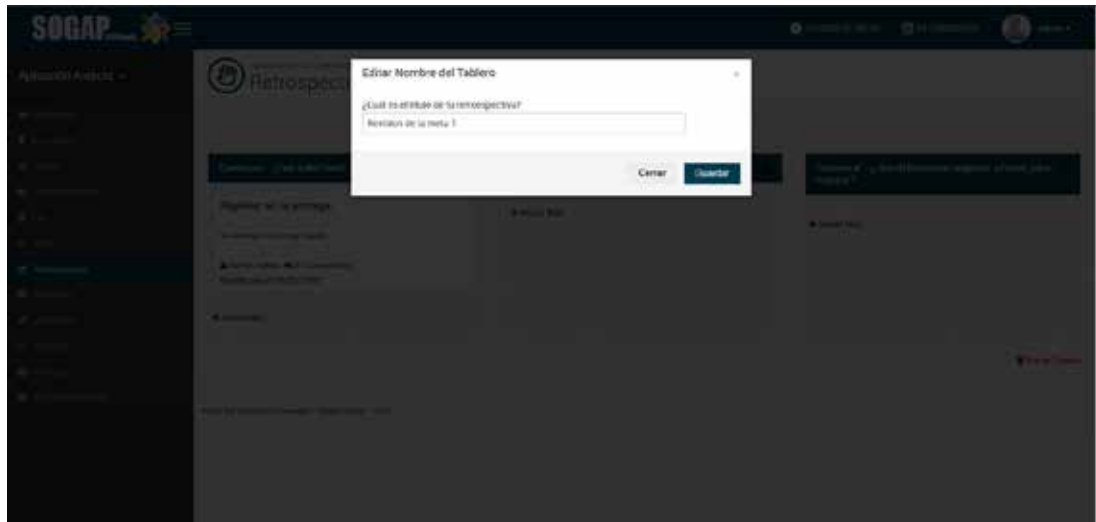


Figura 43: “Pantalla Editar Retrospectiva”. Autor: Bravo A. (2020)

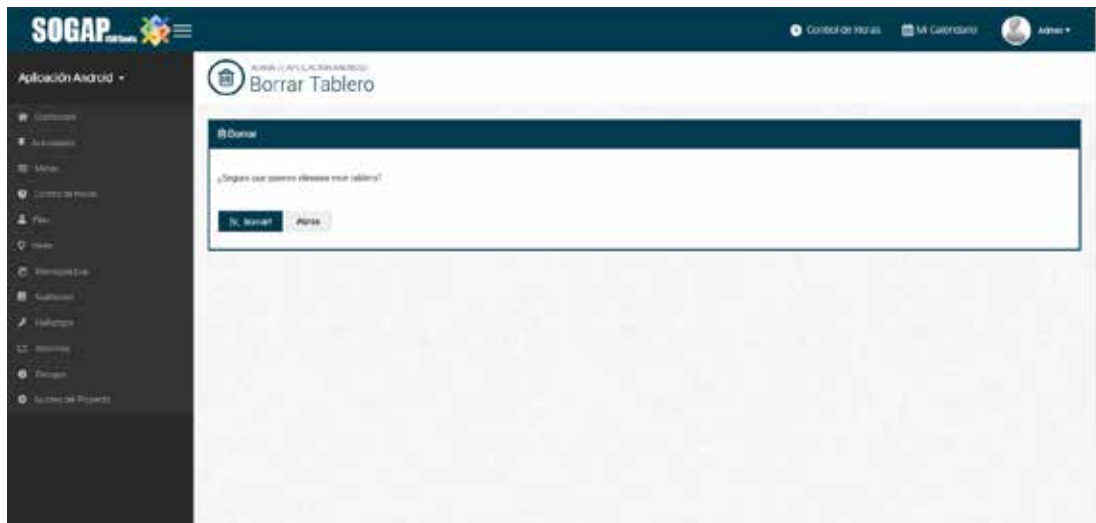


Figura 44: “Pantalla Eliminar Tablero de Retrospectiva”. Autor: Bravo A. (2020)

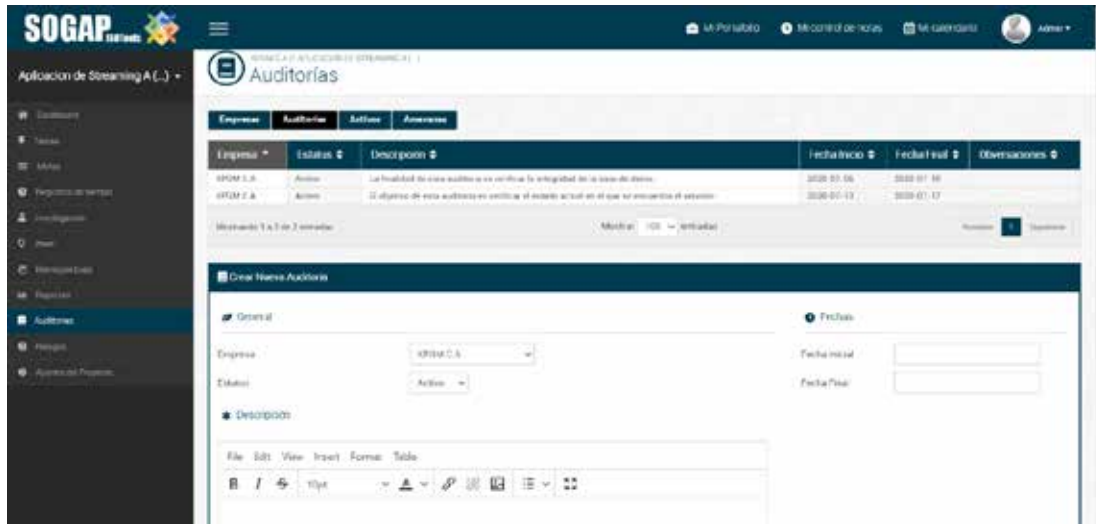


Figura 45: “Pantalla Auditorías”. Autor: Bravo A. (2020)

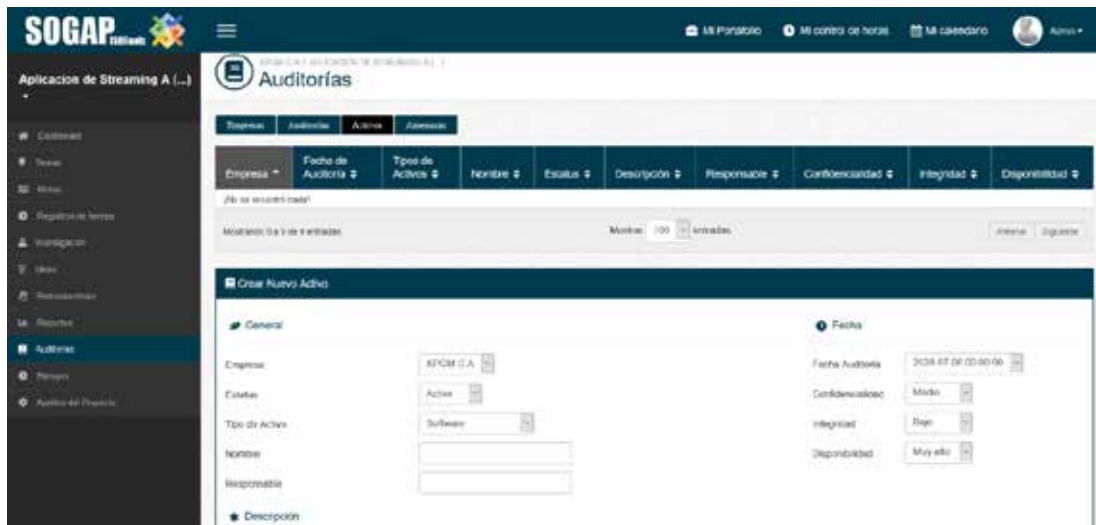


Figura 46: “Pantalla Activos”. Autor: Bravo A. (2020)

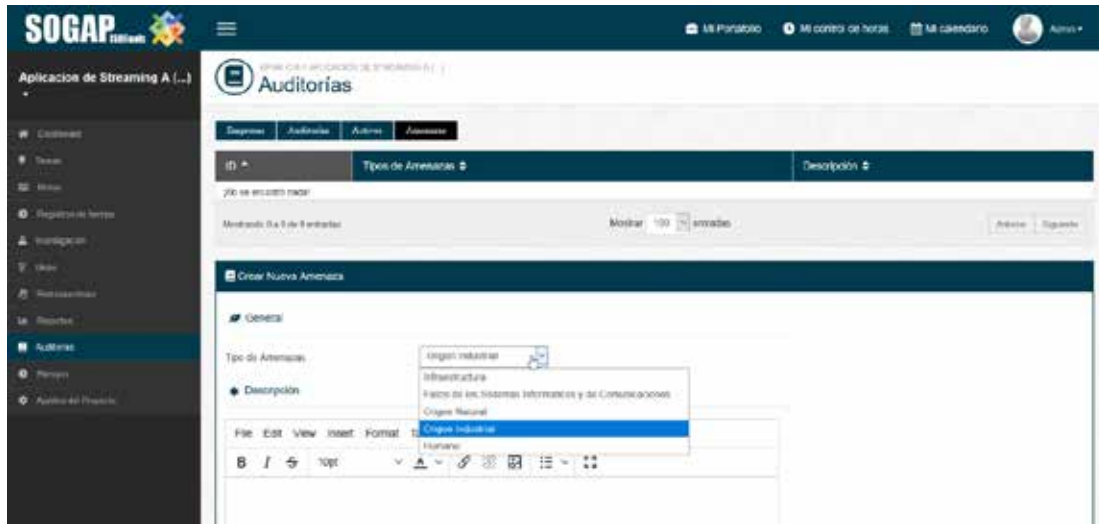


Figura 47: “Pantalla Amenazas”. Autor: Bravo A. (2020)

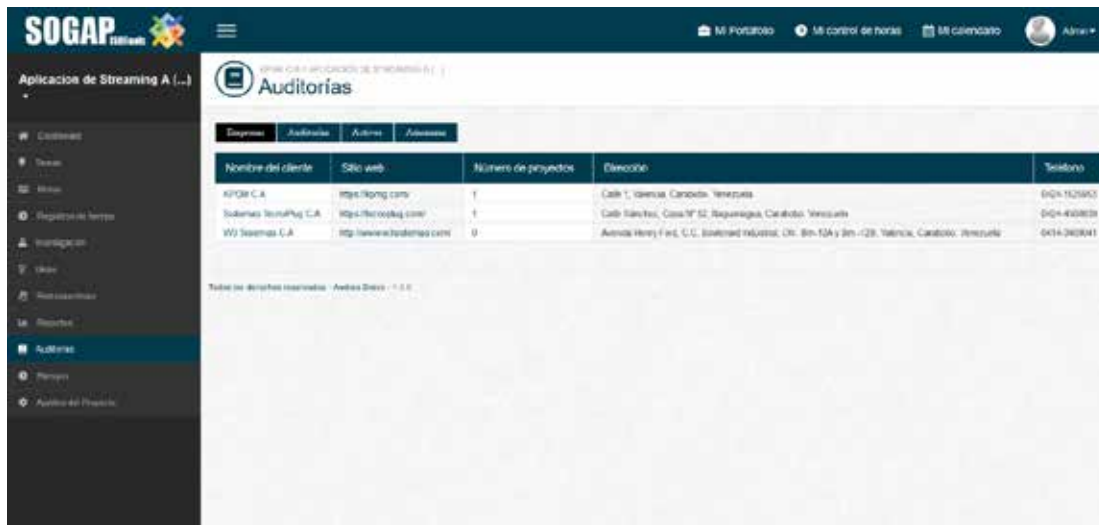


Figura 48: “Pantalla Empresas”. Autor: Bravo A. (2020)

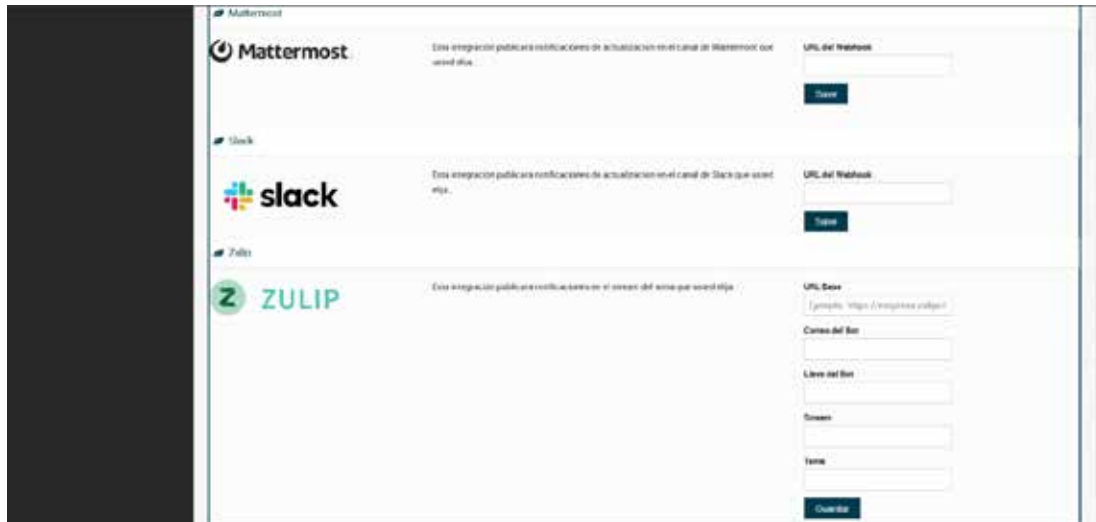


Figura 49: “Pantalla Integraciones: Slack, Mattermost, Zulip”. Autor: Bravo A. (2020)



Figura 50: “Pantalla Control de Horas Personales”. Autor: Bravo A. (2020)

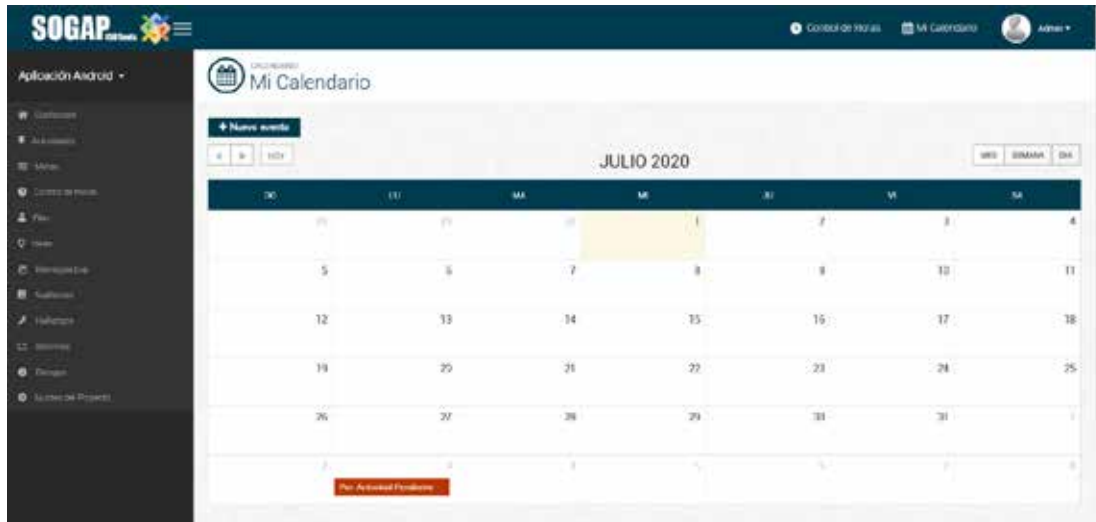


Figura 51: “Pantalla Calendario”. Autor: Bravo A. (2020)

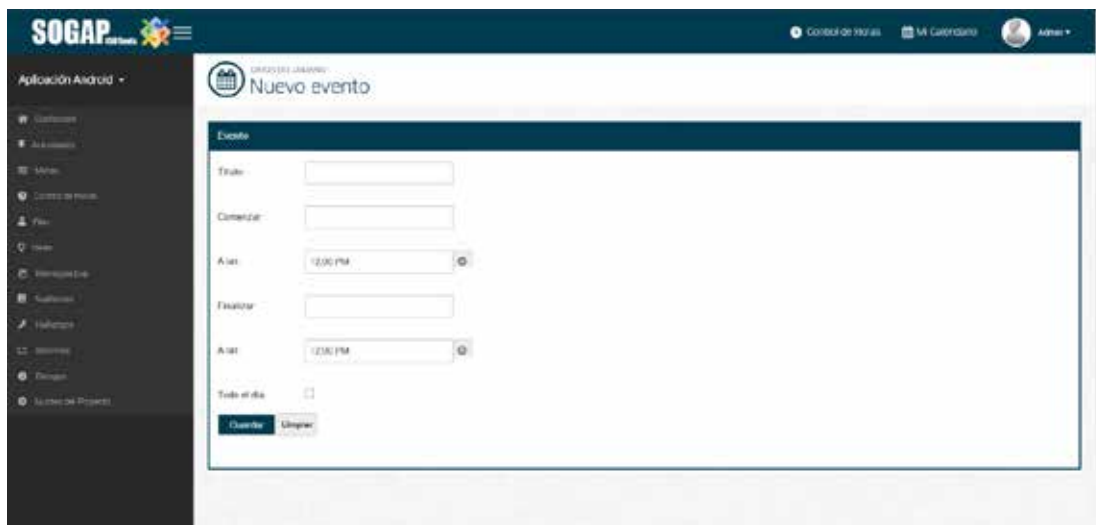


Figura 52: “Pantalla Nuevo Evento”. Autor: Bravo A. (2020)

#### 4.4 Fase IV: Pruebas

Esta es la última fase contemplada en la metodología XP en la misma se llevaron a cabo las pruebas pertinentes para garantizar la efectividad del sistema, estas pruebas se aplicaron a los diferentes módulos y acciones del sistema.

<b>Programa:</b> Inicio de sesión en el sistema con usuario inexistente	
<b>Estrategia de Prueba:</b>	
Caja Negra <input checked="" type="checkbox"/>	Caja Blanca <input type="checkbox"/>
<b>Tipo de Prueba:</b>	
Unidad <input type="checkbox"/>	Integración <input type="checkbox"/>
Sistema <input checked="" type="checkbox"/>	Validación <input checked="" type="checkbox"/>
<b>Técnica de Prueba:</b>	
Prueba para el inicio de sesión en el sistema web con usuario inexistente.	
<b>Resultados:</b> El inicio de sesión no fue exitoso debido a que el usuario no se encuentra registrado en la base de datos. Sin embargo, no redirige al usuario a la pagina de inicio de sesión.	
<b>Decisión:</b> Enviar nuevamente al usuario a la página de inicio de sesión.	

Tabla 53: Caso de Prueba N°1. Fuente: Bravo A. (2020)

<b>Programa:</b> Crear nuevo cliente	
<b>Estrategia de Prueba:</b>	
Caja Negra <input type="checkbox"/>	Caja Blanca <input checked="" type="checkbox"/>
<b>Tipo de Prueba:</b>	
Unidad <input type="checkbox"/>	Integración <input type="checkbox"/>
Sistema <input type="checkbox"/>	Validación <input checked="" type="checkbox"/>
<b>Técnica de Prueba:</b>	
Prueba para la creación de un nuevo cliente.	
<b>Resultados:</b> En el momento del registro es necesario especificar el email del cliente debido a que de este modo será contactado.	
<b>Decisión:</b> Refactorizar el código del formulario para verificar que se realice la comprobación del email y este sea completamente necesario para la creación.	

Tabla 54: Caso de Prueba N°2. Fuente: Bravo A. (2020)

<b>Programa:</b> Descargar informe
<b>Estrategia de Prueba:</b> Caja Negra <input checked="" type="checkbox"/> Caja Blanca <input type="checkbox"/>
<b>Tipo de Prueba:</b> Unidad <input type="checkbox"/> Integración <input type="checkbox"/> Sistema <input type="checkbox"/> Validación <input checked="" type="checkbox"/>
<b>Técnica de Prueba:</b> Prueba para la descarga de un informe.
<b>Resultados:</b> En el momento de la descarga no se obtenía el recurso PDF.
<b>Decisión:</b> Establecer la ruta correcta donde se deben obtener los recursos.

Tabla 55: Caso de Prueba N°3. Fuente: Bravo A. (2020)

<b>Programa:</b> Subir archivos
<b>Estrategia de Prueba:</b> Caja Negra <input type="checkbox"/> Caja Blanca <input checked="" type="checkbox"/>
<b>Tipo de Prueba:</b> Unidad <input type="checkbox"/> Integración <input type="checkbox"/> Sistema <input type="checkbox"/> Validación <input checked="" type="checkbox"/>
<b>Técnica de Prueba:</b> Prueba para la subida de archivos en un proyecto específico.
<b>Resultados:</b> Al momento de la subida el archivo es codificado en otro formato, dando como resultado corrupción del archivo.
<b>Decisión:</b> Aceptar nuevas extensiones de archivos y no modificar las mismas.

Tabla 56: Caso de Prueba N°4. Fuente: Bravo A. (2020)

## **CAPÍTULO V**

### **CONCLUSIONES Y RECOMENDACIONES**

#### **5.1 Conclusión**

Una vez concluido el desarrollo del software de gestión ágil de proyectos utilizando ISOTools para la gestión y administración de proyectos por medio de metodologías ágiles, y tomando como referencia los resultados obtenidos en la presente investigación, se presentan las siguientes conclusiones:

- En primer lugar, durante la fase de iniciación se realizó el proceso de análisis y se obtuvo la determinación de los objetivos de la investigación. Luego, a través de las técnicas de recolección de datos como lo fueron la entrevista no estructurada y la observación directa se logró el correcto uso de las mismas para obtener los requisitos esenciales para el correcto desarrollo del software propuesto.
- De igual modo, la correcta definición de los requerimientos funcionales y no funcionales fueron de vital importancia a la hora de realizar el software pues partiendo de estos se logró realizar una estructura completa que cumpliera con los requerimientos reales de los usuarios.
- Ahora bien, con el desarrollo del software se logró representar gráficamente avances y trayectorias de los distintos proyectos realizados, llevando así un control de gestión sobre el recurso humano, monetario y de tiempo, así como también procesamiento del tiempo que toma realizar actividades, objetivos, metas e ideas que en conjunto con herramientas ISOTools permite la correcta gestión de cualquier proyecto. Todo esto se logró aportando el desarrollo de una interfaz agradable, multiplataforma, responsive, segura y rápida que contribuye a que la experiencia sea lo más agradable posible para cualquier usuario que podría interactuar con la plataforma.
- Por último, durante la fase final se hicieron pruebas de tipo estructural y funcional, por lo que se logró obtener un grado de respuesta adecuado

verificando que los resultados obtenidos a través de ellos sean los correctos para la realización de los distintos procesos que el software posee.

## **5.2 Recomendaciones**

De modo que el sistema pueda seguir creciendo y mejorando, se recomiendan las siguientes indicaciones para un desarrollo a futuro:

- Ampliar módulos de ISOTools para abarcar mayores funcionalidades en la gestión de proyectos.
- Containerizar completamente la aplicación de modo que en el despliegue el sistema se encargue de gestionar todo el clúster de servidores a través de tecnologías de containers como Kubernetes.
- Se recomienda que durante el desarrollo del software se documente lo mejor posible el código, ya que esto es muy importante para efectos de comprensión y mantenimiento.
- Crear automatización de pruebas con Selenium para generar un código base más sólido y estable.

## REFERENCIAS

### Referencias impresas

- Acosta, Dayana (2017). **Diseño de una metodología de gestión de proyectos basada en PMO, con el fin de elevar la productividad de esta área en ACOPI seccional atlántico.** Universidad del Norte, Barranquilla, Colombia.
- Arboleda, Carolina (2018). **Los enfoques ágiles y la teoría del conocimiento aplicados a los proyectos de desarrollo de software.** Universidad EAFIT, Medellín, Colombia.
- Arcentales, Arévalo (2018), **Diseño e implementación de una oficina de gestión de proyecto ágil en una empresa de desarrollo de software.** Universidad de Chile, Santiago de Chile, Chile.
- Arias, Fidias (2012). **El proyecto de investigación: introducción a la metodología científica.** Caracas: Episteme. 6ta edición.
- Balestrini, M. (2006). **Cómo se Elabora el Proyecto de Investigación.** Caracas: BL Consultores Asociados, Servicio Editorial.
- Balladares, Jaime (2017). **Sistema de autogestión de la salud para pacientes con diabetes y asma, desarrollado e implementado en una plataforma Android; con monitoreo de una aplicación web en PHP dirigida a los médicos tratantes, enfocado en la administración y gestión de la base de datos en la implementación de auditorías en base de datos para llevar control de la información de todas las áreas existentes en el proyecto.** Universidad de Guayaquil, Guayaquil, Ecuador.
- Bastidas, F. (2011). **Ejecución de un proyecto de investigación como modalidad de Trabajo Especial de Grado.** Universidad de Carabobo, Carabobo, Venezuela.
- Camirra, H. y Cartaya, S. (2009). **Guía para la investigación académica. Una orientación metodológica diseñada para el postgrado-Iupma.** Caracas: Universidad Pedagógica Experimental Libertador.

- Cifuentes, María (2019). **Análisis comparativo de la gestión de recursos de un proyecto en un ciclo de vida clásico versus las metodologías ágiles.** Pontificia Universidad Javeriana, Bogotá, Colombia.
- Conejo, Marco (2017). **Implementación de una herramienta de software libre para la gestión de proyectos ágiles en la empresa ImagineXYZ.** Universidad Hispanoamericana, San José, Costa Rica.
- Finol, M y Camacho, H. (2008). **El proceso de investigación científica.** Maracaibo: Ediciones de la Universidad del Zulia (Ediluz). (Segunda edición).
- Hung, Marco (2011). **Diseño de una metodología de desarrollo de software basada en la metodología ágil scrum y las mejores prácticas de la gerencia de proyectos.** Universidad Católica Andrés Bello, Caracas, Venezuela.
- Kendall, Kenneth E. y Kendall, Julie E. (2011). **Análisis y diseño de sistemas.** 8va Edición. Naucalpan de Juárez, Edo. De México: Pearson Educación de México.
- Rodríguez, Andrés (2014). **Control de la medición, análisis y mejora de los procesos automatizados por la herramienta mantizonline, por medio de un módulo de mejora continua bajo la norma ISO 9001:2008.** Pontificia Universidad Javeriana, Bogotá, Colombia.
- UJAP (2007). **Normas para la Elaboración y Presentación de los proyectos, Informes, Trabajo de Grado y de Pasantía.** UJAP, Valencia, Venezuela.

#### Referencias electrónicas

- Dashboard (2014). **Qué es un Dashboard y para qué sirve.** Dirección:  
<https://tristanlosegui.com/2014/10/27/que-es-y-para-que-sirve-un-dashboard/>
- Gestión ágil de proyectos (2015). **Gestion Ágil de proyectos.** Dirección:  
<https://jmbeas.es/guias/gestion-agil-de-proyectos/>
- ISOTools. **ISO Software.** Dirección:  
<https://www.isotools.org/>
- Mapas de procesos (2017). **Mapas de procesos: Tipos, definición y desarrollo.** Dirección:

<https://retos-operaciones-logistica.eae.es/tipos-definicion-y-desarrollo-de-un-mapa-de-procesos/>

Metodologías ágiles (2019). **Qué son las metodologías ágiles y cuáles son sus ventajas.** Dirección:

<https://www.iebschool.com/blog/que-son-metodologias-agiles-agile-scrum/>

MySQL (2019). **¿Qué es? Características, Ventajas y Desventajas.** Dirección:

<https://hostingpedia.net/mysql.html>

Programación extrema (1996). **Extreme programming.** Dirección:

<https://extremeprogramming.org/>.

Python (2019). **Qué es Python.** Dirección:

<https://desarrolloweb.com/articulos/1325.php>.

Respuesta ante incidencias para el desarrollo ágil (2020). **Metodología ágil para cuando las cosas van mal.** Dirección:

<https://www.atlassian.com/es/agile/software-development/incident-response>

Rómmel, Filein. **SQLite: La Base de Datos Embebida.** SG Revista #17. Dirección:

<https://sg.com.mx/revista/17/sqlite-la-base-datos-embebida>