



# UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

DESARROLLO DE UNA APLICACIÓN MÓVIL PARA EL PROCESO DE  
COBRANZA UTILIZANDO LA METODOLOGÍA MMS

ALONZO ZHINGRE ANIBAL ARTURO  
INGENIERO DE SISTEMAS

MACHALA  
2022



# UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

DESARROLLO DE UNA APLICACIÓN MÓVIL PARA EL  
PROCESO DE COBRANZA UTILIZANDO LA METODOLOGÍA  
MMS

ALONZO ZHINGRE ANIBAL ARTURO  
INGENIERO DE SISTEMAS

MACHALA  
2022



# UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

TRABAJO TITULACIÓN  
PROPUESTAS TECNOLÓGICAS

DESARROLLO DE UNA APLICACIÓN MÓVIL PARA EL PROCESO DE COBRANZA  
UTILIZANDO LA METODOLOGÍA MMS

ALONZO ZHINGRE ANIBAL ARTURO  
INGENIERO DE SISTEMAS

VALAREZO PARDO MILTON RAFAEL

MACHALA, 24 DE FEBRERO DE 2022

MACHALA  
2022

# DESARROLLO DE UNA APLICACIÓN MÓVIL PARA EL PROCESO DE COBRANZA UTILIZANDO LA METODOLOGÍA MMS

---

## INFORME DE ORIGINALIDAD

---

8%

INDICE DE SIMILITUD

7%

FUENTES DE INTERNET

0%

PUBLICACIONES

3%

TRABAJOS DEL ESTUDIANTE

---

## FUENTE QUE CONTIENE COINCIDENCIAS

---

1

[www.3ciencias.com](http://www.3ciencias.com)

Fuente de Internet

2%

---

2%

★ [www.3ciencias.com](http://www.3ciencias.com)

Fuente de Internet

---

Excluir citas

Activo

Excluir coincidencias  Apagado

Excluir bibliografía

Activo

## CLÁUSULA DE CESIÓN DE DERECHO DE PUBLICACIÓN EN EL REPOSITORIO DIGITAL INSTITUCIONAL

El que suscribe, ALONZO ZHINGRE ANIBAL ARTURO, en calidad de autor del siguiente trabajo escrito titulado DESARROLLO DE UNA APLICACIÓN MÓVIL PARA EL PROCESO DE COBRANZA UTILIZANDO LA METODOLOGÍA MMS, otorga a la Universidad Técnica de Machala, de forma gratuita y no exclusiva, los derechos de reproducción, distribución y comunicación pública de la obra, que constituye un trabajo de autoría propia, sobre la cual tiene potestad para otorgar los derechos contenidos en esta licencia.

El autor declara que el contenido que se publicará es de carácter académico y se enmarca en las disposiciones definidas por la Universidad Técnica de Machala.

Se autoriza a transformar la obra, únicamente cuando sea necesario, y a realizar las adaptaciones pertinentes para permitir su preservación, distribución y publicación en el Repositorio Digital Institucional de la Universidad Técnica de Machala.

El autor como garante de la autoría de la obra y en relación a la misma, declara que la universidad se encuentra libre de todo tipo de responsabilidad sobre el contenido de la obra y que asume la responsabilidad frente a cualquier reclamo o demanda por parte de terceros de manera exclusiva.

Aceptando esta licencia, se cede a la Universidad Técnica de Machala el derecho exclusivo de archivar, reproducir, convertir, comunicar y/o distribuir la obra mundialmente en formato electrónico y digital a través de su Repositorio Digital Institucional, siempre y cuando no se lo haga para obtener beneficio económico.

Machala, 24 de febrero de 2022



ALONZO ZHINGRE ANIBAL ARTURO  
0706611936

## **DEDICATORIA**

Primero doy gracias a Dios por cada ayuda que me ha brindado en el transcurso de mi vida universitaria, ya que sin él hubiera logrado nada.

Como segundo lugar a mis padres Byron Alonzo y María Zhingre, quienes siempre han estado presentes en la buenas y malas, apoyándome, guiándome, dándome el mejor de los consejos para no caer en ningún tipo de circunstancias sea de la universidad o en la vida cotidiana.

A mi hermano Junior de igual manera por apoyarme en diversas labores de la casa para poder así terminar mis tareas, proyectos y por brindarme una de las mejores sazones en su comida.

A mi novia Ginger Vásquez por ser mi gran apoyo en el estudio, en la vida misma, ella me ha enseñado lo bonito que es darse un respiro en todo y seguir en las diversas dificultades.

Y, por último, pero no menos importante, a mis pilares académicos, a los forjadores de mi vida universitaria, a mis docentes, que ayudaron a formar mi carácter académico mediante sus guías, consejos y oportunidades y a mis compañeros con quienes compartí grandes momentos.

Sr. Anibal Arturo Alonzo Zhingre

## **AGRADECIMIENTO**

Agradezco en primer lugar a Dios por permitirme llegar hasta este punto y brindarme las mejores personas a lo largo de mi vida universitaria. De manera muy especial a mis padres y hermano, que me brindaron su apoyo moral e incondicional día tras día.

Por ayudarme a salir adelante agradezco a mi novia Ginger Vásquez, quien siempre estuvo ahí para mí, dándome todo lo mejor de ella, logrando superar todo lo que nos hemos propuesto y espero que Dios nos siga bendiciendo.

A mi tutor, el Ing. Milton Valarezo, gracias por entender la situación que estaba atravesando y brindarme su apoyo en el tema y guía en mi tesis.

A mi buen amigo Darwin Pilalao, por ser una gran persona y excelente compañero ya que sin él no hubiera superado algunos problemas en el transcurso de la universidad.

Finalmente, agradezco a todos los docentes que me apoyaron y motivaron en mi vida universitaria, especialmente al Ingeniero Jimmy Molina quien siempre nos ha inculcado que la carrera de Sistemas debe generar los mejores profesionales, gracias por formarme académicamente y moralmente, son su ética profesional en todo momento.

Sr. Anibal Arturo Alonzo Zhingre

## RESUMEN

En la actualidad, existe un amplio mercado en el área de comercialización y prestación de servicios, diversas empresas enfocan sus ventas en cobros ya sea al contado o a crédito, con el fin de manejar un mejor proceso para su crecimiento; para que estas empresas puedan avanzar y llegar al éxito es necesario que cuenten con una gestión de cobranza eficiente y transparente, lo que garantiza la sustentabilidad de la mismas. La ciudad de Machala cuenta con un gran número de entidades que realizan dichas actividades comerciales, las cuales no generan los ingresos esperados debido a la ineficiencia en la gestión de sus cobros, de esta manera se maximizan errores al momento de realizar cobranzas manuales y se desaprovecha todos los beneficios posibles en nuevas formas de cobranza automatizada. Esta problemática genera una desventaja competitiva respecto a empresas extranjeras, por lo cual es de suma importancia establecer una solución.

El incremento desmesurado de la tecnología en los últimos tiempos es considerable, lo que ha permitido el uso de aplicaciones móviles para solucionar las diferentes situaciones que se generan a diario en los usuarios, ya que siempre y en todo ámbito se busca la facilidad y eficiencia a la hora de realizar diversas tareas.

Mediante lo expresado en párrafos anteriores, se establece una solución a la problemática, la cual se basa en el desarrollo de una aplicación móvil, con el fin de gestionar el proceso de cobranza en las empresas que realicen esta actividad.

Como se conoce, muchas son las herramientas en las cuales las personas estamos inmersas, pero solo unas cuantas nos brindan todas las características y múltiples ventajas al momento de programar, razón por la cual la aplicación móvil del presente proyecto fue desarrollada, bajo las tecnologías de: Java como lenguaje principal de desarrollo, Android Studio como IDE, Firebase correspondiente al marco back-end, y sus complementos Firebase Realtime Database como base de datos NoSQL, y Firebase Authentication para gestionar el proceso de autenticación de usuarios, a su vez se hizo uso de la metodología MMS (Modelo Mobile Sprint) la cual rescata las mejores prácticas ágiles y las

combina con los procesos tradicionales, haciendo de ésta metodología la más óptima al momento de desarrollar.

Existe en la aplicación móvil dos tipos de usuarios: un administrador, el cual tiene acceso a todo lo referente al sistema, gestiona cada uno de los procesos, un usuario de cobranza, el cual maneja un rol muy importante debido a que gestiona a los clientes deudores y procese a su cobranza de una manera segura y cómoda tanto para el cliente como para él mismo. Dichos usuarios se autentican mediante un usuario y contraseña previamente establecidos, creando su validación mediante el sistema de envío de correo electrónico. De esta forma la empresa que se acoja a este sistema contará con un incremento exponencial en sus finanzas, mediante los procesos automatizados. El prototipo desarrollado se basó en los estándares de calidad mediante la norma ISO/IEC 25010 la cual se centra en la confiabilidad sobre la calidad de productos.

**Palabras claves:**

Android Studio, MMS, Aplicación móvil, Gestión de cobranza.

## **ABSTRACT**

Currently, there is a wide market in the area of marketing and provision of services, various companies focus their sales on collections, either in cash or on credit, in order to manage a better process for their growth; In order for these companies to move forward and be successful, they must have efficient and transparent collection management, which guarantees their sustainability. The city of Machala has a large number of entities that carry out these commercial activities, which do not generate the expected income due to inefficiency in the management of their collections, in this way errors are maximized when making manual collections and is wasted. all possible benefits in new forms of automated collection. This problem generates a competitive disadvantage with respect to foreign companies, for which it is of the utmost importance to establish a solution.

The disproportionate increase in technology in recent times is considerable, which has allowed the use of mobile applications to solve the different situations that are generated daily in users, since ease and efficiency are always sought in all areas. when performing various tasks.

Through what is expressed in previous paragraphs, a solution to the problem is established, which is based on the development of a mobile application, in order to manage the collection process in the companies that carry out this activity.

As is known, there are many tools in which people are immersed, but only a few provide us with all the features and multiple advantages at the time of programming, which is why the mobile application of this project was developed, under the technologies of : Java as the main development language, Android Studio as the IDE, Firebase corresponding to the back-end framework, and its plugins Firebase Realtime Database as the NoSQL database, and Firebase Authentication to manage the user authentication process, in turn was made use of the MMS methodology (Mobile Sprint Model) which rescues the best agile practices and combines them with traditional processes, making this methodology the most optimal at the time of development.

There are two types of users in the mobile application: an administrator, who has access to everything related to the system, manages each of the processes, a collection user, who manages a very important role because he manages

customers debtors and process their collection in a safe and comfortable way for both the client and himself. These users are authenticated by means of a previously established username and password, creating their validation through the email delivery system. In this way, the company that uses this system will have an exponential increase in its finances, through automated processes. The prototype developed was based on quality standards through the ISO/IEC 25010 standard, which focuses on reliability over product quality.

**Keywords:**

Android Studio, MMS, Mobile Application, Collection Management.

## INTRODUCCIÓN

Actualmente, el uso de las aplicaciones móviles tiene un gran crecimiento en el mercado de las tecnologías, ya que estas proporcionan un servicio fiable a la hora de ofertar productos o servicios, lo cual es muy útil para las empresas destinadas a todo tipo de comercio.

Los datos estadísticos según el Global System Mobile Association (GSMA), revelan que existen más dispositivos móviles que seres humanos, contando 7422 millones de conexiones móviles, mientras que un censo poblacional, demuestra que existen alrededor de 7228 millones de personas, de esta manera, las aplicaciones móviles han creado una forma diferente en la que se vive, se trabaja y se comunica, haciendo prescindible el uso de las mismas en el futuro. [1]. Según [2] el año 2020 fue un éxito para las aplicaciones móviles ya que obtuvo un crecimiento del 30.2% anual, en aplicaciones de entretenimiento y diversos sectores, además el congreso de Mobile World Congress, realizado por Mobile Economy, confirmó en un estudio lo mencionado anteriormente, donde se demuestra que si existen más dispositivos móviles que personas en el mundo, teniendo en consideración un crecimiento aun mayor para los años por venir de casi mil millones de usuarios para el año 2025 [3].

Con los datos obtenidos de las diversas fuentes se visualiza un futuro mucho más tecnológico, razón por la cual es importante que las empresas aprovechen estos recursos para su crecimiento. Estas tecnologías pueden ser usadas en áreas de toda índole, pero muy en especial en empresas dedicadas al comercio y cobro de productos y servicios ofertados.

En base a la problemática de cobros ineficientes por parte de las empresas a sus clientes, se desarrolla una aplicación móvil para la gestión de cobranzas, con el fin de llevar un control de los deudores y sus aportes, logrando un mayor crecimiento en la empresa, no solo en lo económico sino también en los procesos internos que realice.

Al realizar dicha aplicación se hace uso del lenguaje de programación Java, como entorno principal, junto a su plataforma Android Studio, Firebase como marco back-end, Firebase Realtime Database como gestor de base de datos y

Firestore Authentication para la gestión de autenticaciones, todo esto permite la creación de aplicaciones eficientes, integrando diversas características lo cual sirve como base al momento de crear cualquier tipo de aplicación.

La metodología usada en el desarrollo de la aplicación es la MMS, la cual se basa en un enfoque híbrido, y se caracteriza por tener los mejores aspectos ágiles y tradicionales en cada una de sus etapas, lo que permite obtener aplicaciones móviles de calidad.

El presente documento se encuentra estructurado por capítulos, como se visualiza a continuación:

**Capítulo 1:** se detalla la problemática de la presente investigación, el diagnóstico de las necesidades, su justificación y los requerimientos del aplicativo a desarrollar.

**Capítulo 2:** se presenta la descripción del prototipo, fundamentación teórica, objetivos a alcanzar, metodología y diseño.

**Capítulo 3:** se evalúa el prototipo, mediante un plan previo de evaluación, verificando que su funcionalidad vaya acorde a los requerimientos planteados y luego se presentan los resultados de dicha evaluación.

## CONTENIDO

DEDICATORIA.....	1
AGRADECIMIENTO.....	2
RESUMEN .....	3
ABSTRACT .....	5
INTRODUCCIÓN .....	7
1. CAPITULO I. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS	17
1.1 ÁMBITO DE APLICACIÓN: DESCRIPCIÓN DEL CONTEXTO Y HECHOS DE INTERÉS .....	17
1.2 ESTABLECIMIENTO DE REQUERIMIENTOS.....	18
1.3 JUSTIFICACIÓN DE REQUERIMIENTOS A SATISFACER.....	19
2. CAPITULO II. DESARROLLO DEL PROTOTIPO .....	20
2.1 DEFINICIÓN DEL PROTOTIPO TECNOLÓGICO .....	20
2.1.1 Arquitectura de software .....	20
2.1.1.1 Modelo de capas .....	21
2.1.1.1.1 Capa de presentación .....	21
2.1.1.1.2 Capa de aplicación o lógica.....	21
2.1.1.1.3 Capa de persistencia de datos .....	21
2.2 FUNDAMENTACIÓN TEÓRICA DEL PROTOTIPO.....	22
2.2.1 Java.....	22
2.2.2 Android Studio.....	22
2.2.3 Firebase.....	22
2.2.3.1 Firebase Realtime Database.....	22
2.2.3.2 Firebase Authentication.....	23
2.2.4 Aplicación móvil.....	23
2.2.5 Prototipado.....	23
2.2.5.1 Balsamiq Mockups .....	23
2.2.5.2 Lucidchart .....	23
2.2.6 Base de datos no relacional.....	23
2.2.7 Metodología de desarrollo de software .....	24
2.2.7.1 Metodologías tradicionales .....	24
2.2.7.2 Metodologías ágiles .....	24
2.2.7.3 Metodología híbrida .....	24

2.2.7.3.1 Metodología MMS .....	25
2.3 OBJETIVOS DEL PROTOTIPO .....	25
2.3.1 Objetivo General .....	25
2.3.2 Objetivos específicos .....	25
2.4 DISEÑO DEL PROTOTIPO .....	26
2.4.1 Planificación.....	26
2.4.1.1 Entorno de la empresa .....	26
2.4.1.1.1 Análisis FODA.....	27
2.4.1.1.2 Objetivos del proyecto .....	28
2.4.1.2 Modelo de Negocio .....	28
2.4.1.2.1 Identificación de procesos .....	28
2.4.1.2.2 Segmentación de mercado / usuarios .....	47
2.4.1.2.3 Reglas del negocio.....	47
2.4.1.3 Análisis de Factibilidad.....	47
2.4.1.4 Resultados esperados.....	48
2.4.1.4.1 Beneficios tangibles .....	49
2.4.1.4.2 Beneficios intangibles .....	49
2.4.1.5 Gestión de Stakeholders.....	49
2.4.1.5.1 Establecimiento de roles .....	49
2.4.1.5.2 Formación de equipos .....	50
2.4.1.5.3 Asignación de actividades .....	54
2.4.1.6 Gestión del Alcance .....	58
2.4.1.6.1 Requerimientos funcionales y no funcionales .....	58
2.4.1.6.1.1 Requerimientos funcionales.....	58
2.4.1.6.1.2 Requerimientos no funcionales .....	64
2.4.1.6.2 Elicitación de historias de usuario .....	66
2.4.1.6.3 Creación del Sprint Backlog .....	70
2.4.1.6.4 Elaboración del EDT .....	73
2.4.1.7 Gestión del Tiempo.....	74
2.4.1.7.1 Priorización del Sprint.....	74
2.4.1.7.2 Holgura del proyecto.....	74
2.4.1.7.3 Elaboración del cronograma.....	75
2.4.1.8 Gestión de Costos.....	76

2.4.1.8.1	Estimación de Costos .....	76
2.4.1.8.2	Determinación del presupuesto .....	76
2.4.1.9	Gestión de Cambios .....	76
2.4.1.9.1	Solicitudes de Cambio.....	77
2.4.1.9.2	Matriz de seguimiento de cambios .....	78
2.4.1.10	Gestión de Calidad.....	78
2.4.1.10.1	Métricas de calidad para aplicaciones móviles.....	78
2.4.1.10.2	Estándares de calidad .....	79
2.4.2	Diseño.....	79
2.4.2.1	Análisis de requerimientos.....	79
2.4.2.1.1	Reconocimiento y evaluación del problema .....	79
2.4.2.1.2	Observaciones múltiples .....	80
2.4.2.2	Planteamiento de solución ágil .....	81
2.4.2.3	Modelado de la base de datos .....	81
2.4.2.3.1	Esquema NoSQL .....	82
2.4.2.4	Definición de interfaz .....	82
2.4.2.4.1	Diseño modular.....	82
2.4.3	Ejecución.....	83
2.4.3.1	Codificación.....	83
2.4.3.1.1	Convenciones para el código .....	83
2.4.3.1.2	Documentación.....	84
2.4.3.2	Refactorización .....	84
2.4.3.2.1	Omitir el uso de controles y sentencias anidadas.....	84
2.4.4	Pruebas .....	86
2.4.4.1	Pruebas del sistema .....	86
2.4.4.1.1	Pruebas de calidad .....	86
2.4.4.1.2	Pruebas de tendencia.....	87
2.4.4.1.3	Pruebas de usabilidad.....	88
2.4.4.1.4	Pruebas de mantenibilidad .....	88
2.4.4.1.5	Pruebas de interfaz de usuario .....	89
2.4.5	Lanzamiento .....	90
2.4.5.1	Preparación.....	90
2.5.1	Módulo de Inicio de sesión .....	90

2.5.2	Módulo de agregar cliente .....	93
2.5.3	Módulo de agregar cobradores .....	95
2.5.4	Módulo de agregar ubicación.....	97
3.	CAPITULO III. EVALUACIÓN DEL PROTOTIPO.....	100
3.1	PLAN DE EVALUACIÓN .....	100
3.2	EVALUACIÓN CON HERRAMIENTAS.....	100
3.2.1	Axe .....	100
3.2.2	MobSF .....	107
3.3	CONCLUSIONES .....	108
3.4	RECOMENDACIONES .....	109
	BIBLIOGRAFÍA .....	110
	ANEXOS.....	114
	ANEXO I: CASO DE USO DE LA APLICACIÓN MÓVIL.....	114
	REGISTRO E INICIO DE SESIÓN .....	114
	MENÚ ADMINISTRADOR .....	114
	MENÚ COBRADOR.....	115

## INDICE DE TABLAS

Tabla 1. Análisis FODA.....	27
Tabla 2. Registro.....	31
Tabla 3. Inicio de sesión .....	31
Tabla 4. Selección opción del menú general .....	31
Tabla 5. Selección opción del menú clientes .....	32
Tabla 6. Agregar cliente.....	32
Tabla 7. Guardar cliente .....	32
Tabla 8. Buscar cliente.....	33
Tabla 9. Ingreso de datos a buscar .....	33
Tabla 10. Seleccionar cliente .....	33
Tabla 11. Subir foto .....	34
Tabla 12. Seleccionar foto.....	34
Tabla 13. Editar datos .....	34
Tabla 14. Ingresar datos .....	35
Tabla 15. Seleccionar WhatsApp.....	35
Tabla 16. Seleccionar Ubicaciones.....	35
Tabla 17. Seleccionar Ubicaciones.....	36
Tabla 18. Guardar Ubicación .....	36
Tabla 19. Guardar Ubicación .....	36
Tabla 20. Eliminar Ubicación .....	37
Tabla 21. Seleccionar Crear Deuda .....	37
Tabla 22. Crear Deuda.....	37
Tabla 23. Seleccionar Deuda.....	38
Tabla 24. Realizar abonos.....	38
Tabla 25. Guardar abono.....	38
Tabla 26. Ver seguimiento del cobrador .....	39
Tabla 27. Seleccionar cobradores .....	39
Tabla 28. Agregar cobrador .....	39
Tabla 29. Guardar cobrador.....	40
Tabla 30. Buscar cobrador .....	40
Tabla 31. Seleccionar cobrador.....	40
Tabla 32. Subir foto .....	41
Tabla 33. Seleccionar próximos cobros .....	41
Tabla 34. Seleccionar cliente .....	41
Tabla 35. Asignar cobrador .....	42
Tabla 36. Realizar abono.....	42
Tabla 37. Guardar abono.....	42
Tabla 38. Seleccionar Reportes .....	43
Tabla 39. Seleccionar Configuración.....	43
Tabla 40. Actualizar cambios.....	43
Tabla 41. Visualizar clientes asignados .....	44
Tabla 42. Visualizar información del cliente y deudas creadas.....	44

Tabla 43. Seleccionar “Ubicaciones” .....	44
Tabla 44. Tomar foto .....	45
Tabla 45. Guardar .....	45
Tabla 46. Seleccionar ubicación.....	45
Tabla 47. Eliminar ubicación .....	46
Tabla 48. Seleccionar deuda .....	46
Tabla 49. Seleccionar deuda .....	46
Tabla 50. Seleccionar deuda .....	47
Tabla 51. Especificación de los recursos del proyecto.....	48
Tabla 52. Matriz general de los stakeholders.....	50
Tabla 53. Matriz específica de los stakeholders (Planificación).....	51
Tabla 54. Matriz específica de los stakeholders (Diseño).....	52
Tabla 55. Matriz específica de los stakeholders (Ejecución) .....	52
Tabla 56. Matriz específica de los stakeholders (Pruebas) .....	53
Tabla 57. Matriz específica de los stakeholders (Lanzamiento) .....	53
Tabla 58. Matriz de actividades de gerente.....	54
Tabla 59. Matriz de actividades de analista de requerimientos .....	55
Tabla 60. Matriz de actividades de desarrollador .....	55
Tabla 61. Matriz de actividades de tester .....	56
Tabla 61. Acta de constitución del proyecto.....	57
Tabla 63. Requerimientos funcionales .....	59
Tabla 64. Requerimiento funcional #1 .....	60
Tabla 65. Requerimiento funcional #2.....	60
Tabla 66. Requerimiento funcional #3.....	60
Tabla 67. Requerimiento funcional #4.....	61
Tabla 68. Requerimiento funcional #5.....	61
Tabla 69. Requerimiento funcional #6.....	61
Tabla 70. Requerimiento funcional #7 .....	62
Tabla 71. Requerimiento funcional #8.....	62
Tabla 72. Requerimiento funcional #9.....	62
Tabla 73. Requerimiento funcional #10.....	63
Tabla 74. Requerimiento funcional #11.....	63
Tabla 75. Requerimiento funcional #12.....	63
Tabla 76. Requerimiento funcional #13.....	64
Tabla 77. Requerimientos no funcionales.....	64
Tabla 78. Requerimiento no funcional #1 .....	65
Tabla 79. Requerimiento no funcional #2 .....	65
Tabla 80. Requerimiento no funcional #3 .....	66
Tabla 81. Historia de Usuario #1 .....	67
Tabla 82. Historia de Usuario #2 .....	68
Tabla 83. Historia de Usuario #3.....	69
Tabla 84. Historia de Usuario #4 .....	70
Tabla 85. Sprint Basklog A .....	71
Tabla 86. Sprint Basklog B.....	72

Tabla 87.Sprint Basklog C.....	73
Tabla 88.Cronograma del proyecto .....	75
Tabla 89.Determinación del presupuesto .....	76
Tabla 90.Solicitudes de Cambios.....	77
Tabla 91.Matriz de seguimiento de cambios.....	78
Tabla 92.Observaciones múltiples .....	80
Tabla 93.Observaciones múltiples .....	80
Tabla 94.Diseño Modular.....	82
Tabla 95.Pruebas.....	86
Tabla 96.Pruebas de sistema .....	86
Tabla 97.Pruebas de calidad .....	87
Tabla 98.Pruebas de tendencia.....	87
Tabla 99.Pruebas de usabilidad .....	88
Tabla 100.Pruebas de mantenibilidad .....	89
Tabla 101.Pruebas de interfaz de usuario .....	89

## INDICE DE FIGURAS

Figura 1. Arquitectura del modelo cliente-servidor.....	21
Figura 2. Ciclo de vida Metodología MMS.....	25
Figura 3. Diagrama de Flujo Administrador.....	29
Figura 4. Diagrama de Flujo Cobrador.....	30
Figura 5. Bosquejo Registro e inicio de sesión.....	67
Figura 6. Bosquejo Gestión Administrador.....	68
Figura 7. Bosquejo Gestión Cobrador.....	69
Figura 8. Bosquejo Gestión Seguimiento Cobrador.....	70
Figura 9. EDT del proyecto.....	74
Figura 10. Convenciones para el código.....	83
Figura 11. Documentación.....	84
Figura 12. Optimización “Antes”.....	85
Figura 13. Optimización “Después”.....	85
Figura 14. Ingreso al sistema.....	91
Figura 15. Ingreso de credenciales.....	92
Figura 16. Registro de clientes.....	93
Figura 17. Detalle del cliente.....	94
Figura 18. Registro de Cobradores.....	95
Figura 19. Lista de cobradores.....	96
Figura 20. Agregar Ubicación al cliente.....	97
Figura 21. Visualizar Ubicación.....	98
Figura 22. Visualizar ruta del cliente.....	99
Figura 23. Resultado de la evaluación Axe.....	100
Figura 24. Resultado de la evaluación Axe ColorConstrast.....	101
Figura 25. Resultado de la evaluación Axe ActiveViewName.....	102
Figura 26. Resultado de la evaluación Axe ActiveViewName.....	103
Figura 27. Resultado de la evaluación Axe ActiveViewName.....	104
Figura 28. Resultado de la evaluación Axe ColorConstrast.....	105
Figura 29. ¡Resultado de la evaluación Axe No violations found!.....	106
Figura 30. Evaluación con herramienta MobSF.....	107
Figura 31. Dominios evaluados con la herramienta MobSF.....	107
Figura 32. Anexo I: Caso de uso de registro e inicio de sesión.....	114
Figura 33. Anexo I: Caso de uso menú del administrador.....	114
Figura 34. Anexo I: Caso de uso menú del cobrador.....	115

## **1. CAPITULO I. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS**

### **1.1 ÁMBITO DE APLICACIÓN: DESCRIPCIÓN DEL CONTEXTO Y HECHOS DE INTERÉS**

Toda empresa cuenta con diversas áreas que juntas ayudan a la misma a salir a flote y mantenerse en el mercado, es por eso que un sistema de gestión en el área de cobranzas es fundamental para poder estar a la par y cubrir las necesidades del mismo, dándole índole en el beneficio y facilitando el trabajo del administrador a cargo de esta área y la persona encargada de efectuar el cobro directamente con el cliente (cobrador o recaudador).

La mayoría de empresas dedicadas al sistema de cobranzas, realizan sus procesos de manera tradicional, donde el cliente es quien se acerca a efectuar sus abonos en la mayoría de los casos, según el tiempo y el monto establecido, por otra parte al no cumplirse con lo mencionado, los cobradores se ven en la necesidad de acercarse al domicilio del deudor y efectuar los cobros correspondiente, lo cual resulta tedioso en la forma de gestionar dichos cobros tanto para el deudor como para los cobradores ya que las empresas se manejan con diverso personal respecto a un mismo departamento, en este caso el de cobranzas.

La forma en que se realizan los cobros, genera retrasos al cobrador debido a que desconoce la ubicación exacta del cliente, por ende, se generan un sin números de casos negativos para la empresa, haciendo que el servicio de esta área sea deficiente.

Una gran ayuda en esta área es conocer el estado o ruta del cobrador en su labor, ya que esto permite a la empresa tener un control de sus cobradores, haciendo este proceso aún más ágil y dinámico.

Es por eso que se plantea el desarrollo de una aplicación móvil en la plataforma de java con IDE Android Studio y Firebase, de esta manera se lograra automatizar el proceso de gestión de cobranza en las empresas, permitiéndolo hacer de manera ágil y eficiente

## 1.2 ESTABLECIMIENTO DE REQUERIMIENTOS

Desde su inicio hasta su crecimiento total cada empresa debe tomar las buenas prácticas, para su buen funcionamiento, teniendo en cuenta el servicio que oferta, se debe tener en consideración diferentes aspectos, como el servicio que se brinda al momento de las cobranzas, los procesos internos que gestione la empresa, pago de los deudores, cobro hacia los mismos, entre otros, por lo que se han establecido los siguientes requisitos:

### Administrador

- Inicio de sesión mediante correo y contraseña
- Visualizar clientes, cobradores, próximas deudas, reportes y configuración
- CRUD de los clientes
- Comunicación por red social WhatsApp
- Visualizar ubicaciones de los clientes
- Agregar referencias
- Crear deudas, visualizarlas, ver detalle de las mismas.
- Realizar abonos
- Visualizar deudas ordenadas por fecha de creación, de más reciente a la más antigua.
- Visualizar la ruta del cobrador hacia el cliente

### Cobrador

- Inicio de sesión mediante correo y contraseña
- Visualizar clientes asignados por el administrador, ya sea por día, semanal o mensual
- Visualizar la información de cada cliente, la deuda contraída, fecha, abonos y ubicación.
- Tomar fotografía de la casa del cliente con el fin que su ubicación exacta se guarde en tiempo real.
- Realizar cobros en efectivo, transferencia o depósito, luego subir el comprobante correspondiente.

### **1.3 JUSTIFICACIÓN DE REQUERIMIENTOS A SATISFACER**

Hoy en día, la gestión de cobranza en empresas administrativas es un tema de mucho interés, debido a que las empresas administrativas demandan un considerable número de cuentas basadas en servicios de cobros, lo cual constituye un elemento importante en los objetivos que manejan las mismas.

Según [4] La decisión de liquidez es una de las principales gestiones financieras donde las empresas necesitan mantener un equilibrio entre liquidez y rentabilidad, es por eso que las mismas deben ser correctamente estructuradas contando con cuentas de cobranzas, activos, pasivos, eficiencia en diversas actividades y oportunidades para las nuevas empresas.

En la actualidad tanto empresas nacionales como mundiales poseen un entorno económico el cual hace que enfrenten una falta de fondos para financiar sus actividades siendo su mayoría de cuentas de cobranzas en sus principales activos, motivo por el cual según [5] es necesario que dichas empresas posean una mejor gestión en cuanto a sus procesos de cobranza tanto internos como externos con el fin de generar mayor liquidez y satisfacción para todos los involucrados, es por eso que según [6] se plantea implementar un sistema de cobranzas, el mismo que permitirá cambios positivos en la gestión de cobranzas, controlando y optimizando las diferentes causas y efectos de la mismas, lo cual permite obtener resultados agradables enfocados en minimizar los diferentes problemas causados por la falta de ética de un grupo de personas, dicho proceso una vez optimizado, permite que la empresa recupere los bienes invertidos en las diferentes actividades efectuadas en los préstamos.

La gestión de cobranzas de toda empresa, debe cumplir con las buenas prácticas establecidas en su inicio, por ende se plantea el desarrollo de una aplicación móvil que cubra la necesidad en el área de cobranzas, reduzca el tiempo de ubicación al momento de cobrar y mejora la satisfacción tanto del cliente como la persona que realiza la cobranza, de esta manera la empresa competirá en el mercado de una mejor manera con respecto al área de cobro, logrando disminuir el grupo de personas sin ética, recuperando los bienes de la empresa.

Como guía se implementará la metodología híbrida MMS para el desarrollo de la aplicación, asegurando el cumplimiento de las buenas prácticas y de los requerimientos para garantizar la funcionalidad de la aplicación.

## **2. CAPITULO II. DESARROLLO DEL PROTOTIPO**

### **2.1 DEFINICIÓN DEL PROTOTIPO TECNOLÓGICO**

El prototipo usado en la aplicación móvil tiene su base en lo que respecta a la arquitectura que conlleva. El presente apartado muestra la estructura general de la aplicación, la arquitectura utilizada y su modelo de capas.

#### **2.1.1 Arquitectura de software**

Una arquitectura de software se basa en un conjunto de modelos, los cuales brindan las directrices necesarias al momento de crear un software, por lo cual es importante la colaboración de todo el personal involucrado para obtener aplicaciones de calidad [7].

Además, ésta se considera como el mayor nivel respecto al diseño de softwares debido a que facilita una buena estructura, correlación e interactividad en lo que respecta a los procesos que posee un software, razón por la cual una arquitectura de software bien diseñada es clave para las organizaciones debido a que les permite avanzar y centrarse en sus objetivos [8].

La arquitectura en la que se centra el presente proyecto es la de tipo cliente-servidor, en la cual los procesos se realizan de manera distribuida entre ambas partes del sistema. El cliente realiza las peticiones al servidor, y éste devuelve la respuesta acorde a lo que se ejecutó. Según [9] este modelo es uno de los mejores debido a que se pueden realizar múltiples tareas en un mismo computador obteniendo resultados en tiempo real de manera rápida y eficaz. En la presente figura se visualiza el bosquejo utilizado respecto a la arquitectura mencionada.

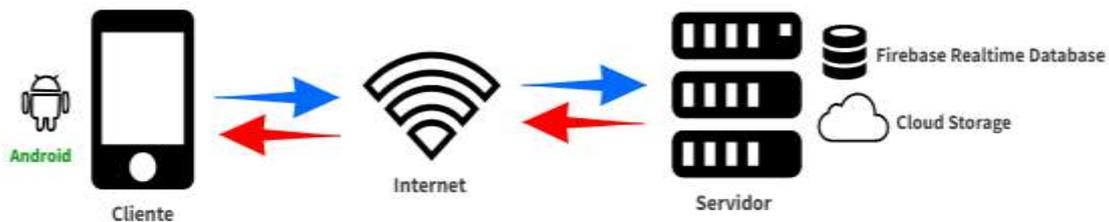


Figura 1. Arquitectura del modelo cliente-servidor

Fuente: Elaboración propia

### 2.1.1.1 Modelo de capas

Según [10] el modelo de capas posee tres capas, Lógica, Física y Aplicación. Cada capa consta de diferentes aspectos es por eso que la capa Física cuenta con sensores y actuadores, la parte Lógica permite que los componentes software puedan comunicarse con los objetos inteligentes del internet y finalmente la capa de Aplicación, quien es la encargada de la interfaz gráfica permitiendo que el sistema IOT interactúe con los usuarios.

#### 2.1.1.1.1 Capa de presentación

La capa de presentación es una recopilación de aplicaciones que ayudan a la interfaz de usuario a interactuar con los clientes para poder manejar las aplicaciones de una manera intuitiva [11].

#### 2.1.1.1.2 Capa de aplicación o lógica

La capa lógica va de la mano con la capa anterior y permite que el usuario interactúe con la funcionalidad de las aplicaciones [12].

#### 2.1.1.1.3 Capa de persistencia de datos

La capa de persistencia de datos es la encargada de almacenar, y tener a la mano para realizar la consulta de los datos anteriormente recopilados anteriormente por la capa de aplicación [13], dicha capa ocupa datos relacionales y no relacionales, los cuales utilizan los archivos ya creados, se actualizan y se guardan posteriormente pueden ser eliminados.

## **2.2 FUNDAMENTACIÓN TEÓRICA DEL PROTOTIPO**

### **2.2.1 Java**

Java se usa para crear aplicaciones que cuentan con diferentes características, siendo esta una tecnología de desarrollo que mejora los sitios web a diferencia de JavaScript que crea y ejecuta páginas webs sencillas. Permite cargar fotografías hasta chatear en línea y servicios bancarios entre otras cualidades muy útiles que se usan hoy en día. Además, cuenta con un sistema que notifica si existe alguna actualización debido a que suele estar en constante mejora [14].

### **2.2.2 Android Studio**

Android Studio es un entorno de desarrollo integrado enfocado al sistema operativo Android de Google [15], posee varias funciones incorporadas, entre ellas un asistente usado en la creación de ilimitados números de aplicaciones [16]. Android Studio permite también acelerar el desarrollo y realizar compilaciones de aplicaciones de la más alta calidad, con un emulador eficaz y lleno de funciones útiles al momento de crear código fiable [17].

### **2.2.3 Firebase**

Firebase brinda servicios en la nube gratuitamente hasta un determinado número, es una combinación de dichos servicios como la mensajería instantánea, permite un almacenamiento NoSql en bases de datos en tiempo real, así como autenticaciones, alojamiento de archivos, entre otros [18] [19]. Además, cuenta con herramientas como Firebase Realtime Database, Firebase Authentication y a su vez brinda una integración a los sistemas IOS, Android y a la web [20].

#### **2.2.3.1 Firebase Realtime Database**

Firebase Realtime Database es una base de datos NoSQL alojada en la nube, permite compilar aplicaciones sin necesidad de servidores, guardando y sincronizando datos Json en tiempo real entre diversos usuarios sin necesidad de conexión y de manera segura [21].

### **2.2.3.2 Firebase Authentication**

Firebase Authentication facilita el desarrollo de sistemas de autenticación seguros y fiables, permite encontrar una solución en base a líneas de código reducido para las aplicaciones con el fin de facilitar la autenticación, acceso e integración de los usuarios [22].

### **2.2.4 Aplicación móvil**

Una aplicación móvil es un software que puede ser instalado en un dispositivo, más comúnmente en celulares, el cual permite ejecutar diversas funciones acordes a los fines para los cuales haya sido diseñado [23] [24].

### **2.2.5 Prototipado**

Realizar un prototipado, es crear la simulación mediante interfaces o en un programa que permita apreciar la veracidad del proyecto en la vida real, ya que se dan a conocer las diferentes vulnerabilidades del mismo y tener la solución de manera que se minimicen los gastos. Dando a conocer la cara final del producto al cliente, de esta manera existe la relación, analista, cliente y los usuarios [25]. Un buen prototipado permite medir el rendimiento y el alcance de cada proyecto, fortaleciendo el proceso de cada proyecto hasta su etapa de ejecución [26].

#### **2.2.5.1 Balsamiq Mockups**

Balsamiq mockups es una valiosa herramienta que permite diseñar excelentes interfaces en el front-end en los diferentes proyectos que tengas en mente, sean estas aplicaciones móviles o sistemas webs [27].

#### **2.2.5.2 Lucidchart**

La herramienta Lucidchart sirve para describir el funcionamiento de cada uno de los sistemas informáticos que se trabaje, mediante la representación en diagramas de flujo, diagramas de clases, paquetes de casos de uso, diagrama de actividades. Que permiten el desarrollo de los procesos educativos, lo cual se aprovecha hoy en día el uso en la nube [28].

### **2.2.6 Base de datos no relacional**

La base de datos NoSQL, sirve para crear e innovar las antiguas practicas sobre el uso de datos, ya que incluye los nuevos datos siendo esta una nueva base de

datos flexible. Ya que permite adaptarse al dato debido a que no presenta un esquema de tablas fijas, permite leer sus datos de manera horizontal de esta manera evita las principales operaciones en la unión de datos. Esto genera una menor latencia y alto rendimiento [29].

### **2.2.7 Metodología de desarrollo de software**

Las metodologías de desarrollo de software, ha tenido un cambio fenomenal en el transcurso de la historia, debido a que sufrió cambios como el implementar modelos y metodologías clásicas, logrando con ellas pruebas y gestión de las mismas y aún más, mejorando con la llegada del internet, el cual a permitido crear la interacción equipos – usuarios y esto permitió dar vida a un desarrollo de software de calidad y ágil para su próximo mantenimiento [30].

#### **2.2.7.1 Metodologías tradicionales**

Las metodologías tradicionales, nacen en el transcurso de la vida en el desarrollo de software, permiten tener una visión clara en los diferentes prototipos en desarrollo de software, logrando tener un control de la mismas en cada una de sus etapas. Las cuales son funcionales en prototipo pequeños ya que tranquilamente se logra el objetivo del mismo, sin embargo, se ven afectadas en prototipo de mayor alcance ya que no permiten tener el control del flujo en el mismo [31].

#### **2.2.7.2 Metodologías ágiles**

Una metodología ágil es aquella que se basa en trabajos de desarrollo en lapsos cortos, permite eliminar incertidumbres, fomenta la eficiencia respecto a la producción y calidad de softwares finales y permite realizar cambios con rápidos tiempos de respuesta, brindando una mayor satisfacción al cliente en cada una de sus etapas [32].

#### **2.2.7.3 Metodología híbrida**

Una metodología híbrida permite el desarrollo de software enfocado a la filosofía del manifiesto ágil y la documentación de las bases tradicionales, lo que permite tener la mejor combinación de estos procesos a la hora de elaborar productos de calidad y eficientes [33].

### 2.2.7.3.1 Metodología MMS

La metodología MMS o también denominada Modelo Móvil Sprint, es una metodología híbrida desarrollada especialmente para la creación de aplicaciones móviles, consta de varias fases las cuales poseen las características esenciales para obtener un buen desarrollo [34].

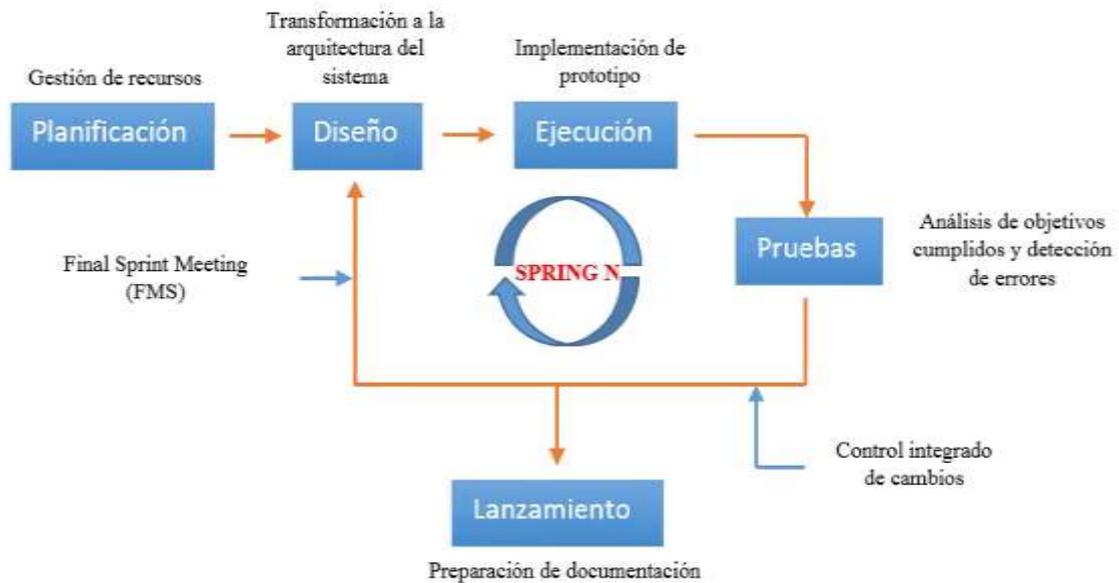


Figura 2. Ciclo de vida Metodología MMS

Fuente: Elaboración a partir de [34]

## 2.3 OBJETIVOS DEL PROTOTIPO

### 2.3.1 Objetivo General

Desarrollar una aplicación móvil de cobranza que utilice la metodología MMS, plataforma de Android Studio, Firebase en sus distribuciones de Authentication para autenticaciones del sistema y Realtime Database como base de datos NoSql con el fin de mejorar el proceso de gestión de cobranzas hacia las empresas destinadas en esta área.

### 2.3.2 Objetivos específicos

- Indagar en fuentes confiables información acerca del desarrollo e implementación de aplicaciones móviles.

- Seleccionar las plataformas tecnológicas que permitan llegar al objetivo general planteado.
- Hacer uso de la metodología MMS, que brinda una guía completa acerca del desarrollo de aplicaciones móviles, combinando las mejores prácticas ágiles y tradicionales.
- Establecer los requerimientos funcionales y no funcionales con los que contará la aplicación.
- Prototipar los diseños creados en bosquejos anteriores para su posterior implementación.
- Desarrollar la aplicación móvil de gestión de cobros mediante el uso de las plataformas Android Studio, Firebase, Firebase Authentication y Firebase Realtime Database.
- Realizar una evaluación de la aplicación móvil desarrollada con el fin de confirmar la funcionalidad de la misma.

## **2.4 DISEÑO DEL PROTOTIPO**

La metodología utilizada en el desarrollo del presente proyecto es la MMS, la cual es una metodología híbrida, y se basa en las siguientes fases:

### **2.4.1 Planificación**

En todo proyecto de software se inicia con una planificación, siendo esta la primera etapa donde se establecen parámetros de tiempo, capital y recursos humanos, motivo por el cual su objetivo se basa en realizar un seguimiento que permita obtener aplicaciones móviles eficientes. Para realizar lo antes mencionado es necesario definir el entorno de la empresa, modelo de negocio, análisis de factibilidad, resultados esperados, gestión de stakeholders, gestión del alcance, gestión del cronograma, gestión de costos, gestión de cambios y gestión de calidad [34].

#### **2.4.1.1 Entorno de la empresa**

El presente proyecto se enfoca en las empresas dedicadas al sector comercial, las cuales poseen una planificación relacionada con sus ventas ya sea al contado o crédito, los créditos los dan a clientes con el fin de aumentar las ganancias en ventas, lo cual resulta bueno debido a que existe un movimiento constante en

las cuentas que se deben cobrar y por ende existe una mejor organización en cuanto a la gestión de cobranzas. Estas empresas a su vez, manejan unas cobranzas deficientes, la manera en que realizan los cobros no es la más adecuada, debido a que en el momento que el cobrador procede a cobrar, no existe constancia de dicho cobro, así como se evade algunos aspectos como, el acceso a la ubicación exacta del cliente mediante una captura fotográfica, lo cual es muy importante en futuros cobros ya que los cobradores varían, la ruta que lleva el cobrador, lo que permite al administrador conocer por donde avanza el mismo, los tipos de pagos realizados ya sea al contado, por depósito o por transferencia, entre otros. Estos aspectos son un factor clave al gestionar cobranzas motivo por el cual se desarrolló una aplicación móvil que resuelva dicha problemática y brinde un mejor manejo de los cobros en las empresas destinadas a este fin.

#### 2.4.1.1.1 Análisis FODA

Toda empresa debe tener fortalezas y oportunidades que le permitan crecer de una mejor manera, pero también es necesario determinar las debilidades que la misma posee y las amenazas a las que se encuentra expuesta, por esta razón se plantea el análisis FODA de las empresas dedicadas a la gestión de cobros.

Tabla 1. Análisis FODA

FORTALEZAS	OPORTUNIDADES
Tener una cartera de clientes fijos	Automatización de la gestión de cobranzas
Priorización del trato al cliente	Incentivos a clientes
Interés bajo	
DEBILIDADES	AMENAZAS
Proceso de gestión de cobros tradicional	Inestabilidad por pagos atrasados o no cancelados.
Desconocimiento de la ubicación exacta del cliente al momento al momento de efectuar los cobros	Pérdida de clientes respecto a la competencia que brinda servicios de igual proceder automatizados.

Fuente: Elaborado a partir de [34]

#### **2.4.1.1.2 Objetivos del proyecto**

##### Objetivo General

Automatizar el proceso de gestión de cobros mediante una aplicación móvil que satisfaga las necesidades de cobranza con el fin de brindar eficiencia en las empresas que brinden servicios destinados a este fin.

##### Objetivos Específicos

- Establecer los requerimientos funcionales y no funcionales con los que contará la aplicación móvil
- Realizar los diagramas necesarios para poder obtener una aplicación de calidad.
- Realizar evaluaciones finales una vez terminada la aplicación, con el fin de verificar su funcionalidad.

#### **2.4.1.2 Modelo de Negocio**

Un modelo de negocio en las empresas, se relaciona con la manera en la que un negocio genera dinero, prioriza a los clientes, mejora la lógica en los procesos y se centra en las oportunidades [35], esto es fundamental, motivo por el cual el modelo de negocio del presente proyecto se basa en la identificación de procesos, segmentación de mercado, y reglas del negocio.

##### **2.4.1.2.1 Identificación de procesos**

Los procesos en los que se basa la aplicación desarrollada se ven reflejados en diagramas de flujo, los cuales permiten visualizar lo que realiza cada usuario (administrador, cobrador), como se observa a continuación:

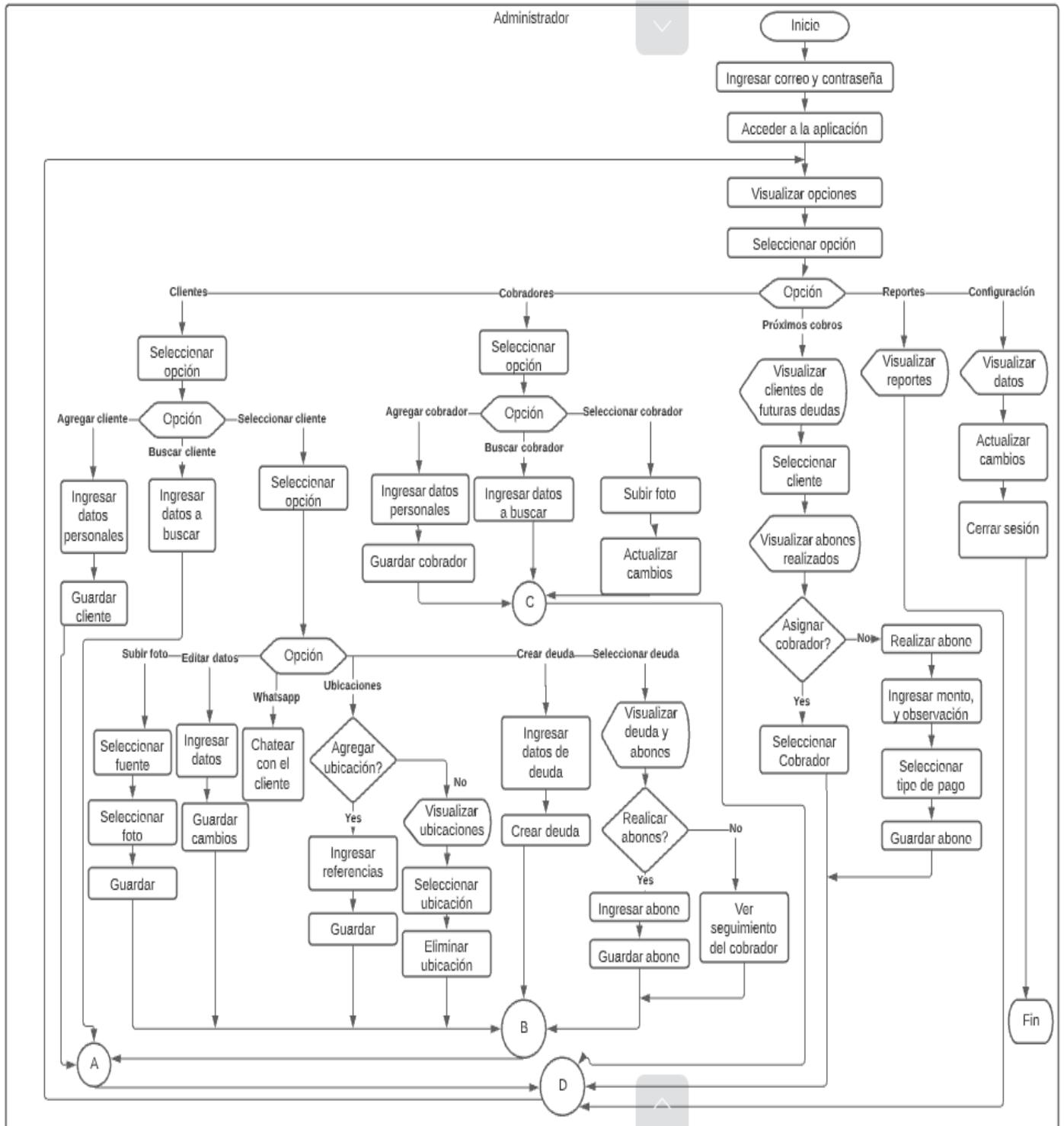


Figura 3. Diagrama de Flujo Administrador

Fuente: Elaboración propia

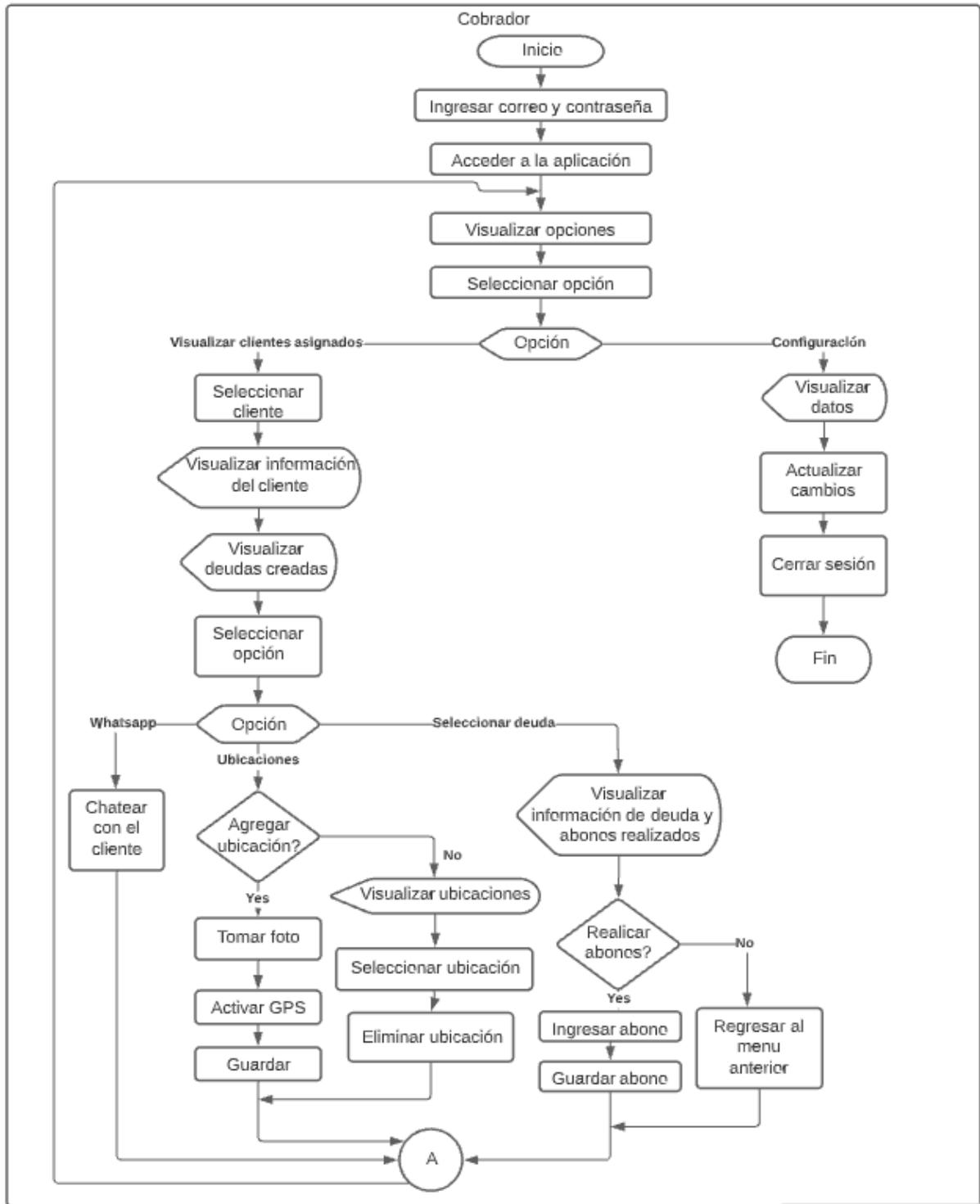


Figura 4. Diagrama de Flujo Cobrador

Fuente: Elaboración propia

Una vez realizados los diagramas de flujo, es necesario tener definido de manera más clara las actividades de la aplicación, por lo cual se hace uso de diversas tablas que detallan a fondo dichas actividades.

La Tabla 2, hace mención al registro de los usuarios al iniciar la aplicación.

Tabla 2. Registro

CAMPO	DESCRIPCION
Actividad	Registrarse en la aplicación
Origen	Acceder a la aplicación
Actor	Administrador, Cobrador
Pre-condición	Tener un correo electrónico
Post-condición	Iniciar sesión en la aplicación

Fuente: Elaborado a partir de [34]

La Tabla 3, hace referencia al inicio de sesión de los usuarios previamente registrados.

Tabla 3. Inicio de sesión

CAMPO	DESCRIPCION
Actividad	Iniciar sesión
Origen	Registrarse en la aplicación
Actor	Administrador, cobrador
Pre-condición	Estar registrado
Post-condición	Visualizar menú (opciones)

Fuente: Elaborado a partir de [34]

La Tabla 4, muestra la selección de las opciones que se muestran una vez los usuarios hayan iniciado sesión

Tabla 4. Selección opción del menú general

CAMPO	DESCRIPCION
Actividad	Seleccionar opción del menú general
Origen	Iniciar sesión
Actor	Administrador, cobrador
Pre-condición	Haber visualizado las opciones
Post-condición	Ejecución de opción seleccionada

Fuente: Elaborado a partir de [34]

La Tabla 5, refiere la selección de las opciones del menú clientes, dentro del usuario administrador.

Tabla 5. Selección opción del menú clientes

CAMPO	DESCRIPCION
Actividad	Seleccionar opción del menú clientes
Origen	Seleccionar opción del menú general
Actor	Administrador
Pre-condición	Haber seleccionado opción del menú general
Post-condición	Agregar cliente, buscar cliente, seleccionar cliente

Fuente: Elaborado a partir de [34]

La Tabla 6, permite agregar un cliente, mediante el ingreso de sus datos personales.

Tabla 6. Agregar cliente

CAMPO	DESCRIPCION
Actividad	Agregar cliente
Origen	Seleccionar opción del menú clientes
Actor	Administrador
Pre-condición	Haber seleccionado opción agregar clientes
Post-condición	Ingresar datos personales

Fuente: Elaborado a partir de [34]

La Tabla 7, muestra el guardado de un cliente, mediante un previo registro de la información requerida.

Tabla 7. Guardar cliente

CAMPO	DESCRIPCION
Actividad	Guardar cliente
Origen	Ingresar datos personales
Actor	Administrador
Pre-condición	Haber Ingresado los datos personales
Post-condición	Volver a la opción "Visualizar opciones"

Fuente: Elaborado a partir de [34]

La Tabla 8, permite visualizar la búsqueda de un cliente mediante su nombre o apellido.

Tabla 8. Buscar cliente

CAMPO	DESCRIPCION
Actividad	Buscar cliente
Origen	Seleccionar opción del menú clientes
Actor	Administrador
Pre-condición	Haber seleccionado opción buscar cliente
Post-condición	Ingresar datos a buscar

Fuente: Elaborado a partir de [34]

La Tabla 9, hace referencia al ingreso de los datos que se desean buscar, una vez realizada la búsqueda, se procede a volver al inicio, donde se visualizan las opciones iniciales.

Tabla 9. Ingreso de datos a buscar

CAMPO	DESCRIPCION
Actividad	Ingresar datos a buscar
Origen	Buscar cliente
Actor	Administrador
Pre-condición	Haber seleccionado “ingresar datos a buscar”
Post-condición	Volver a la opción “Visualizar opciones”

Fuente: Elaborado a partir de [34]

La Tabla 10, permite seleccionar un cliente para luego poder subir foto, editar datos, acceder a WhatsApp, ver la ubicación, crear deuda, o seleccionar deuda.

Tabla 10. Seleccionar cliente

CAMPO	DESCRIPCION
Actividad	Seleccionar cliente
Origen	Seleccionar opción del menú clientes
Actor	Administrador
Pre-condición	Haber seleccionado opción del menú general
Post-condición	Seleccionar opción

Fuente: Elaborado a partir de [34]

La Tabla 11, permite subir una foto a elección propia del cliente ingresado.

Tabla 11. Subir foto

CAMPO	DESCRIPCION
Actividad	Subir foto
Origen	Seleccionar opción
Actor	Administrador
Pre-condición	Haber seleccionado opción
Post-condición	Seleccionar foto

Fuente: Elaborado a partir de [34]

La Tabla 12, hace referencia a la selección de la foto antes de subirla, para luego proceder a guardarla.

Tabla 12. Seleccionar foto

CAMPO	DESCRIPCION
Actividad	Seleccionar foto
Origen	Subir foto
Actor	Administrador
Pre-condición	Haber seleccionado la fuente
Post-condición	Guardar foto

Fuente: Elaborado a partir de [34]

La Tabla 13, hace mención a la edición de los datos del cliente previamente ingresados.

Tabla 13. Editar datos

CAMPO	DESCRIPCION
Actividad	Editar datos
Origen	Seleccionar opción
Actor	Administrador
Pre-condición	Haber seleccionado opción
Post-condición	Ingresar datos

Fuente: Elaborado a partir de [34]

La Tabla 14, muestra el ingreso de la información al momento de haber seleccionado “editar datos”.

Tabla 14. Ingresar datos

CAMPO	DESCRIPCION
Actividad	Ingresar datos
Origen	Editar datos
Actor	Administrador
Pre-condición	Haber seleccionado “Ingresar datos”
Post-condición	Guardar cambios

Fuente: Elaborado a partir de [34]

La Tabla 15, muestra la selección de la red social WhatsApp, con la cual el administrador puede comunicarse con el cliente.

Tabla 15. Seleccionar WhatsApp

CAMPO	DESCRIPCION
Actividad	Seleccionar Whatsapp
Origen	Seleccionar opción
Actor	Administrador, cobrador
Pre-condición	Haber seleccionado opción
Post-condición	Chatear con el cliente

Fuente: Elaborado a partir de [34]

La Tabla 16, muestra las ubicaciones de cada cliente.

Tabla 16. Seleccionar Ubicaciones

CAMPO	DESCRIPCION
Actividad	Seleccionar ubicaciones
Origen	Seleccionar opción
Actor	Administrador
Pre-condición	Haber seleccionado opción
Post-condición	Agregar ubicación

Fuente: Elaborado a partir de [34]

La Tabla 17, muestra el ingreso de una nueva ubicación, para su posterior almacenamiento.

Tabla 17. Seleccionar Ubicaciones

CAMPO	DESCRIPCION
Actividad	Agregar ubicación
Origen	Seleccionar ubicaciones
Actor	Administrador
Pre-condición	Haber seleccionado ubicaciones
Post-condición	Guardar ubicación

Fuente: Elaborado a partir de [34]

La Tabla 18, muestra el almacenamiento de la ubicación previamente ingresada, con sus referencias y toda la información necesaria.

Tabla 18. Guardar Ubicación

CAMPO	DESCRIPCION
Actividad	Guardar ubicación
Origen	Ingresar referencias
Actor	Administrador
Pre-condición	Haber ingresado las referencias
Post-condición	Volver a la opción "Visualizar opciones"

Fuente: Elaborado a partir de [34]

La Tabla 19, permite visualizar las ubicaciones de los clientes.

Tabla 19. Guardar Ubicación

CAMPO	DESCRIPCION
Actividad	Visualizar ubicación
Origen	Seleccionar ubicación
Actor	Administrador
Pre-condición	Haber seleccionado "ubicaciones"
Post-condición	Seleccionar ubicación

Fuente: Elaborado a partir de [34]

La Tabla 20, hace referencia a la opción de “eliminar ubicación” en caso de haber ingresado referencias incorrectas o haber tomado la foto de la casa del cliente equivocado.

Tabla 20. Eliminar Ubicación

CAMPO	DESCRIPCION
Actividad	Eliminar ubicación
Origen	Seleccionar ubicación
Actor	Administrador
Pre-condición	Haber seleccionado ubicación
Post-condición	Volver a la opción “Visualizar opciones”

Fuente: Elaborado a partir de [34]

La Tabla 21, muestra la selección de la opción “Crear deuda”.

Tabla 21. Seleccionar Crear Deuda

CAMPO	DESCRIPCION
Actividad	Seleccionar Crear deuda
Origen	Seleccionar opción
Actor	Administrador
Pre-condición	Haber seleccionado opción
Post-condición	Ingresar datos de deuda

Fuente: Elaborado a partir de [34]

La Tabla 22, muestra la creación de las deudas por parte del administrador, donde el mismo debe haber ingresado previamente todos los datos para dicha creación.

Tabla 22. Crear Deuda

CAMPO	DESCRIPCION
Actividad	Crear deuda
Origen	Ingresar datos de deuda
Actor	Administrador
Pre-condición	Haber ingresado los datos de la deuda
Post-condición	Volver a la opción “Visualizar opciones”

Fuente: Elaborado a partir de [34]

La Tabla 23, muestra la selección de la opción “Seleccionar deuda”

Tabla 23. Seleccionar Deuda

CAMPO	DESCRIPCION
Actividad	Seleccionar deuda
Origen	Seleccionar opción
Actor	Administrador
Pre-condición	Haber seleccionado opción
Post-condición	Realizar abonos

Fuente: Elaborado a partir de [34]

La Tabla 24, hace referencia al hecho de realizar abonos, una vez visualizados los mismos y las deudas en sí.

Tabla 24. Realizar abonos

CAMPO	DESCRIPCION
Actividad	Realizar abonos
Origen	Visualizar deuda y abonos
Actor	Administrador
Pre-condición	Haber visualizado deudas y abonos
Post-condición	Guardar abono

Fuente: Elaborado a partir de [34]

La Tabla 25, muestra el proceso de “guardar abonos” los cuales fueron ingresados con anterioridad.

Tabla 25. Guardar abono

CAMPO	DESCRIPCION
Actividad	Guardar abono
Origen	Ingresar abono
Actor	Administrador
Pre-condición	Haber ingresado abono
Post-condición	Volver a la opción “Visualizar opciones”

Fuente: Elaborado a partir de [34]

La Tabla 26, hace referencia al seguimiento que se le hace al cobrador al momento de trasladarse hacia los clientes.

Tabla 26. Ver seguimiento del cobrador

CAMPO	DESCRIPCION
Actividad	Ver seguimiento del cobrador
Origen	Visualizar deudas y abonos
Actor	Administrador
Pre-condición	Haber visualizado deudas y abonos
Post-condición	Volver a la opción "Visualizar opciones"

Fuente: Elaborado a partir de [34]

La Tabla 27, muestra la opción de seleccionar a los cobradores, de acuerdo al menú general de opciones.

Tabla 27. Seleccionar cobradores

CAMPO	DESCRIPCION
Actividad	Seleccionar cobradores
Origen	Seleccionar opción del menú general
Actor	Administrador
Pre-condición	Haber seleccionado opción del menú general
Post-condición	Agregar cobrador

Fuente: Elaborado a partir de [34]

La Tabla 28, permite agregar un cobrador, para luego almacenarlo.

Tabla 28. Agregar cobrador

CAMPO	DESCRIPCION
Actividad	Agregar cobrador
Origen	Seleccionar opción
Actor	Administrador
Pre-condición	Haber seleccionado opción
Post-condición	Guardar cobrador

Fuente: Elaborado a partir de [34]

La Tabla 29 menciona la opción “Guardar cobrador” la cual se realiza con el ingreso previo de la información personal de los cobradores.

Tabla 29. Guardar cobrador

CAMPO	DESCRIPCION
Actividad	Guardar cobrador
Origen	Ingresar datos personales
Actor	Administrador
Pre-condición	Haber ingresado datos personales
Post-condición	Volver a la opción “Visualizar opciones”

Fuente: Elaborado a partir de [34]

La Tabla 30, muestra la opción de buscar cobradores, mediante el ingreso de su nombre o apellido.

Tabla 30. Buscar cobrador

CAMPO	DESCRIPCION
Actividad	Buscar cobrador
Origen	Seleccionar opción
Actor	Administrador
Pre-condición	Haber seleccionado opción
Post-condición	Ingresar datos a buscar

Fuente: Elaborado a partir de [34]

La Tabla 31, muestra la opción de “Seleccionar a los cobradores”, para poder realizar otras actividades posteriores.

Tabla 31. Seleccionar cobrador

CAMPO	DESCRIPCION
Actividad	Seleccionar cobrador
Origen	Seleccionar opción
Actor	Administrador
Pre-condición	Haber seleccionado opción
Post-condición	Subir foto

Fuente: Elaborado a partir de [34]

La Tabla 32, muestra la actividad de “Subir foto”, la cual luego de realizada la actividad se procede a actualizar los cambios realizados.

Tabla 32. Subir foto

CAMPO	DESCRIPCION
Actividad	Subir foto
Origen	Seleccionar cobrador
Actor	Administrador
Pre-condición	Haber seleccionado cobrador
Post-condición	Actualizar cambios

Fuente: Elaborado a partir de [34]

La Tabla 33, muestra la opción de los próximos cobros a realizarse, la cual luego de haberla seleccionado permite escoger el cliente a quien se le efectúa el cobro posterior.

Tabla 33. Seleccionar próximos cobros

CAMPO	DESCRIPCION
Actividad	Seleccionar próximos cobros
Origen	Seleccionar opción del menú general
Actor	Administrador
Pre-condición	Haber seleccionado opción del menú general
Post-condición	Seleccionar cliente

Fuente: Elaborado a partir de [34]

La Tabla 34, muestra los clientes que poseen futuras deudas.

Tabla 34. Seleccionar cliente

CAMPO	DESCRIPCION
Actividad	Seleccionar cliente
Origen	Visualizar clientes de futuras deudas
Actor	Administrador
Pre-condición	Haber visualizado clientes de futuras deudas
Post-condición	Asignar cobrador

Fuente: Elaborado a partir de [34]

La Tabla 35, hace referencia al cobrador que es asignado, mediante una previa selección.

Tabla 35. Asignar cobrador

CAMPO	DESCRIPCION
Actividad	Asignar cobrador
Origen	Visualizar abonos realizados
Actor	Administrador
Pre-condición	Haber visualizado abonos realizados
Post-condición	Seleccionar cobrador

Fuente: Elaborado a partir de [34]

La Tabla 36, muestra la opción de realizar abonos, la cual se ejecuta luego de haber visualizado los abonos realizados.

Tabla 36. Realizar abono

CAMPO	DESCRIPCION
Actividad	Realizar abono
Origen	Visualizar abonos realizados
Actor	Administrador
Pre-condición	Haber visualizado abonos realizados
Post-condición	Guardar abono

Fuente: Elaborado a partir de [34]

La Tabla 37, hace referencia a “guardar abono”, actividad que requiere un ingreso previo del monto, observación y tipo de pago.

Tabla 37. Guardar abono

CAMPO	DESCRIPCION
Actividad	Guardar abono
Origen	Ingresar monto, observación y tipo de pago
Actor	Administrador
Pre-condición	Haber ingresado monto, observación y tipo de pago
Post-condición	Volver a la opción “Visualizar opciones”

Fuente: Elaborado a partir de [34]

La Tabla 38, muestra la visualización de los reportes generados en base a las deudas y abonos.

Tabla 38. Seleccionar Reportes

CAMPO	DESCRIPCION
Actividad	Seleccionar Reportes
Origen	Seleccionar opción del menú general
Actor	Administrador
Pre-condición	Haber seleccionado opción del menú general
Post-condición	Visualizar reportes

Fuente: Elaborado a partir de [34]

La Tabla 39, muestra la opción de configuración de la aplicación, tanto para el administrador como para el cobrador.

Tabla 39. Seleccionar Configuración

CAMPO	DESCRIPCION
Actividad	Seleccionar configuración
Origen	Seleccionar opción del menú general
Actor	Administrador, cobrador
Pre-condición	Haber seleccionado opción del menú general
Post-condición	Visualizar datos

Fuente: Elaborado a partir de [34]

La Tabla 40, muestra la actualización de los cambios que se realicen en el perfil del usuario, luego se procede o no a cerrar sesión.

Tabla 40. Actualizar cambios

CAMPO	DESCRIPCION
Actividad	Actualizar cambios
Origen	Visualizar datos
Actor	Administrador, cobrador
Pre-condición	Haber visualizado los datos
Post-condición	Cerrar sesión

Fuente: Elaborado a partir de [34]

La Tabla 41, hace referencia a la opción de visualización de clientes que se les ha asignado ya un cobrador.

Tabla 41. Visualizar clientes asignados

CAMPO	DESCRIPCION
Actividad	Visualizar clientes asignados
Origen	Seleccionar opción del menú general
Actor	Cobrador
Pre-condición	Haber seleccionado opción del menú general
Post-condición	Seleccionar cliente

Fuente: Elaborado a partir de [34]

La Tabla 42, permite la visualización de la información de los clientes y las deudas contraídas.

Tabla 42. Visualizar información del cliente y deudas creadas

CAMPO	DESCRIPCION
Actividad	Visualizar información del cliente y deudas creadas
Origen	Seleccionar cliente
Actor	Cobrador
Pre-condición	Haber seleccionado un cliente
Post-condición	Seleccionar opción

Fuente: Elaborado a partir de [34]

La Tabla 43, hace referencia a la opción “Ubicaciones”, seleccionada previamente.

Tabla 43. Seleccionar “Ubicaciones”

CAMPO	DESCRIPCION
Actividad	Seleccionar “ubicaciones”
Origen	Seleccionar opción
Actor	Cobrador
Pre-condición	Haber seleccionado opción
Post-condición	Agregar ubicación

Fuente: Elaborado a partir de [34]

La Tabla 44, hace mención a la opción de agregar una ubicación.

Tabla 44. Tomar foto

CAMPO	DESCRIPCION
Actividad	Tomar foto
Origen	Agregar ubicación
Actor	Cobrador
Pre-condición	Haber seleccionado agregar ubicación
Post-condición	Activar GPS

Fuente: Elaborado a partir de [34]

La Tabla 45, permite guardar una ubicación, mediante la activación previa del GPS, con el fin de generar automáticamente dicha ubicación tomada de google maps.

Tabla 45. Guardar

CAMPO	DESCRIPCION
Actividad	Guardar
Origen	Activar GPS
Actor	Cobrador
Pre-condición	Haber activado el GPS
Post-condición	Volver a la opción "Visualizar opciones"

Fuente: Elaborado a partir de [34]

La Tabla 45, hace referencia a las ubicaciones seleccionadas.

Tabla 46. Seleccionar ubicación

CAMPO	DESCRIPCION
Actividad	Seleccionar ubicación
Origen	Visualizar ubicaciones
Actor	Cobrador
Pre-condición	Haber visualizado las ubicaciones
Post-condición	Eliminar ubicación

Fuente: Elaborado a partir de [34]

La Tabla 46, muestra la opción de eliminar una ubicación previamente seleccionada.

Tabla 47. Eliminar ubicación

CAMPO	DESCRIPCION
Actividad	Eliminar ubicación
Origen	Seleccionar ubicación
Actor	Cobrador
Pre-condición	Haber seleccionado ubicación
Post-condición	Volver a la opción "Visualizar opciones"

Fuente: Elaborado a partir de [34]

La Tabla 47, hace mención a la opción "Seleccionar deuda".

Tabla 48. Seleccionar deuda

CAMPO	DESCRIPCION
Actividad	Seleccionar deuda
Origen	Seleccionar opción
Actor	Cobrador
Pre-condición	Haber seleccionado opción
Post-condición	Realizar abonos

Fuente: Elaborado a partir de [34]

La Tabla 48, muestra los abonos realizados, luego de haber visualizado la información de las deudas y abonos anteriores.

Tabla 49. Seleccionar deuda

CAMPO	DESCRIPCION
Actividad	Realizar abonos
Origen	Visualizar información de deuda y abonos realizados
Actor	Cobrador
Pre-condición	Haber visualizado información de deudas y abonos realizados
Post-condición	Ingresar abono

Fuente: Elaborado a partir de [34]

La Tabla 49, hace referencia a los abonos guardados, los cuales fueron ingresados con anterioridad.

Tabla 50. Seleccionar deuda

CAMPO	DESCRIPCION
Actividad	Guardar abono
Origen	Ingresar abono
Actor	Cobrador
Pre-condición	Haber ingresado abono
Post-condición	Volver a la opción "Visualizar opciones"

Fuente: Elaborado a partir de [34]

#### **2.4.1.2.2 Segmentación de mercado / usuarios**

El presente proyecto está enfocado en toda empresa dentro de la ciudad de Machala que tenga un área de gestión de cobranza, dicha empresa puede ser de menor, de medio o superiores ingresos, que cuenten con Smartphone inteligentes y su sistema local.

#### **2.4.1.2.3 Reglas del negocio**

Para que una empresa o negocio pueda cumplir con todos los procesos planteados se deben tener en cuenta reglas que permitan llevar un equilibrio entre la funcionalidad de la empresa y los procesos éstas realicen, por lo cual estas reglas deben estar especificadas de manera clara y precisa [36].

#### **2.4.1.3 Análisis de Factibilidad**

El análisis factibilidad es esencial en todos los proyectos que estén por desarrollar ya que permite tener en claro los recursos que se van a utilizar en el mismo, dando como resultado el uso de esta buena práctica para un proyecto de éxito, logrando un punto esencial que es el avance con los objetivos de investigación. Se debe tener en claro los tres objetivos de factibilidad, siendo el primero la factibilidad operativa, que es la encargada de gestionar los recursos en la empresa, como segundo objetivo se encuentra, la factibilidad técnica la cual es responsable de la estructura de la empresa y como ultimo la factibilidad

económica, que como su nombre lo indica es acerca del dinero en beneficio de la empresa [37], esta a su vez se divide en tres recursos esenciales que son: primero el recurso humano, los recursos materiales y los recursos tecnológicos; todos ellos generan beneficios en la empresa dando como resultado el crecimiento de la misma [38].

A continuación, se muestran los recursos requeridos para el presente proyecto:

Tabla 51. Especificación de los recursos del proyecto

RECURSOS HUMANOS			
Rol/Profesión	Cantidad	Costo Individual	Costo Total
Gerente de proyecto, analista de requerimientos, desarrolladora	1	\$660	\$ 660
RECURSOS MATERIALES			
Material	Cantidad	Costo individual	Costo total
Resma de hojas	1	\$3,25	\$3,25
Lápiz	1	\$0,30	\$0,30
RECURSOS TECNOLOGICOS			
HARDWARE			
Material	Cantidad	Costo individual	Costo total
computador	1	\$1200	\$1200
SOFTWARE			
Material	Cantidad	Costo individual	Costo total
Firebase	1	\$0	\$0
Android Studio	1	\$0	\$0
Cuenta en Play Store	1	\$25	\$25
TOTAL		\$1888,55	\$1888,55

Fuente: Elaborado a partir de [34]

#### 2.4.1.4 Resultados esperados

Es necesario que en todo proyecto que tengamos en mente o que se esté realizando, es tener en claro los resultados que se van a dar mediante dicho proyecto se han estos que los podamos palpar o no, ya que los resultados se basan en los diferentes objetivos que hemos trazado a su inicio, de esta manera se puede aprovechar sus beneficios de manera esperada. Por ende, es recomendable realizar pruebas de simulación antes de la ejecución del proyecto sea cual sea el área de desarrollo [39].

#### **2.4.1.4.1 Beneficios tangibles**

- Incremento de seguridad en ubicación al momento del cobro.
- Mejor gestión en la implementación de cobranza al domicilio.
- Eficiencia en la gestión de cobranza por tiempo establecido.

#### **2.4.1.4.2 Beneficios intangibles**

- Mayor rendimiento en tiempo de ubicación por parte del cobrador.
- Mejor rendimiento por parte del administrador.

#### **2.4.1.5 Gestión de Stakeholders**

La gestión de los stakeholders es el total acuerdo de las partes interesadas con cada uno de los cambios que se realicen dentro del proyecto, de esta manera se obtiene el cubrimiento total en las necesidades de los clientes, por ende, cada persona es responsable de cumplir con los roles que se le establece en determinada actividad [40].

##### **2.4.1.5.1 Establecimiento de roles**

En todo proyecto se establecen roles y por ende responsabilidades, las que permiten tener un mejor control en el flujo del proyecto, por lo cual es más fácil la administración de las actividades que se realicen, para ello se ha implementado el uso de una matriz en la cual se especifica los aspectos que dan fluidez al proyecto, como se aprecia a continuación, sabiendo que la letra “I”, representa a los stakeholders internos y la letra “E”, representa a los stakeholders externos.

Tabla 52.Matriz general de los stakeholders

MATRIZ GENERAL DE STAKEHOLDERS				
Id	Interesado	Cargo	Rol	responsabilidades
I-001	Anibal Alonzo	Estudiante	Gerente de proyecto	- Guiar y tomar decisiones en la planificación y ejecución del proyecto
			Analista de requerimientos	- Transforma los requerimientos en actividades que se deben realizar. - Creación de historias de usuarios. - Crear y establecer los entregables.
			Equipo de desarrollo (desarrolladora. tester)	- Establecer el código según las especificaciones. - Depurar el código sin alterar su funcionamiento
E-001	Ing. Milton Valarezo	Ingeniero de sistemas	Tutor Encargado	- Dar guía y resolver cualquier duda a lo largo del proyecto.
E-002	Usuario final	N/D	Cliente	- Empresas u usuarios q usen la aplicación.

Fuente: Elaborado a partir de [34]

#### 2.4.1.5.2 Formación de equipos

La asignación de equipos es esencial para el desarrollo del proyecto por ende se asignan roles y actividades en base a los interesados, en este caso solo existe una persona que es encargada de las distintas actividades, en donde se especifica el nivel el nivel de implicación en donde 1 representa a la mayor prioridad y 5 a la menos prioridad.

Tabla 53. Matriz específica de los stakeholders (Planificación)

MATRIZ ESPECIFICA DE STAKEHOLDERS				
Fase	Planificación			
Fecha inicio	03/11/2021			
Fecha Finalización	17/11/2021			
Id	Rol	Actividades	Nivel Implicación	Acceso Información
I-001	Gerente del proyecto	Establecer el entorno de la empresa (Crear el análisis FODA y definir los objetivos)	2	Acta de constitución del proyecto.
		Definir el modelo de negocio del proyecto (Tener en claro los procesos, segmentar el mercado y establecer las reglas del negocio).	2	Acta de constitución del proyecto
		Crear el análisis de factibilidad del proyecto (Definir los recursos humanos, materiales y tecnológicos).	2	Matriz de especificación de recursos del proyecto.
		Plantear los resultados esperados (Determinar los beneficios tangibles e intangibles de la gestión de cobranza.	2	Acta de constitución del proyecto
		Definir lo interesados del proyecto (Establecer una matriz general, específica y de actividades para agregar los roles de los stakeholders.	3	Matriz general y específica de stakeholders
		Crear el EDT del proyecto en base a la metodología seleccionada (MMS).	3	Plantilla de desglose de trabajo
		Determinar la holgura del proyecto mediante un cronograma, además que permita priorizar los sprint.	3	Cronograma del proyecto
		Definir el presupuesto mediante la estimación en costos del proyecto.	3	Matriz de determinación del presupuesto.
		Determinar las métricas y estándares para la aplicación móvil.	4	--
	Análisis de requerimientos	Elaborar la gestión del alcance del proyecto (Determinar los requerimientos funcionales y no funcionales que llevara la aplicación mediante la utilización de plantillas generales y específicas).	4	Matriz de requerimientos funcionales y no funcionales.
		Elaboración de los Sprintg Backlog y creación de las historias de usuarios.	3	-Matriz de historias de usuario -Matriz de Sprint Backlog
		Tratar los cambios del proyecto haciendo uso de las plantillas para informar al gerente de proyecto.	3	Matriz de seguimiento de cambios

Fuente: Elaborado a partir de [34]

Tabla 54. Matriz específica de los stakeholders (Diseño)

MATRIZ ESPECIFICA DE STAKEHOLDERS				
Fase	Diseño			
Fecha inicio	18/11/2021			
Fecha Finalización	08/12/2021			
Id	Rol	Actividades	Nivel Implicación	Acceso Información
I-001	Analista de requerimientos	Realizar el análisis de la perspectiva de los requerimientos mediante un reconocimiento y evaluación de problemas, haciendo uso de plantillas.	4	Matriz para la perspectiva de análisis
	Equipo de proyecto (Desarrollador)	Crear modelos con enfoque a la simplicidad y calidad para la aplicación.	5	----
		Crear una base de datos que tenga que tenga el modelo basado en entidad – relación y a su vez diccionario de datos.	3	-Modelo entidad relación. -Modelo de diccionario de datos.
		Diseñar los módulos que llevara la aplicación.	5	----

Fuente: Elaborado a partir de [34]

Tabla 55. Matriz específica de los stakeholders (Ejecución)

MATRIZ ESPECIFICA DE STAKEHOLDERS				
Fase	Ejecución			
Fecha inicio	09/12/2021			
Fecha Finalización	29/12/2021			
Id	Rol	Actividades	Nivel Implicación	Acceso Información
I-001	Desarrollador	Se implementó un control de versiones y editor de código.	4	Matriz para la documentación
		Implementar buenas prácticas al momento de codificar.	5	----

Fuente: Elaborado a partir de [34]

Tabla 56.Matriz específica de los stakeholders (Pruebas)

MATRIZ ESPECIFICA DE STAKEHOLDERS				
Fase	Pruebas			
Fecha inicio	30/12/2021			
Fecha Finalización	19/01/2022			
Id	Rol	Actividades	Nivel Implicación	Acceso Información
I-001	Equipo de proyecto (Tester)	Detectar bugs y errores	5	Matriz de pruebas de la aplicación.
		Identificar fallas en el funcionamiento de la aplicación	5	Matriz de pruebas de la aplicación
		Corregir errores identificados.	5	Matriz de control de cambios.
		Identificar posibles mejoras en los módulos funcionales.	5	Matriz de control de cambios en base a errores detectados.

Fuente: Elaborado a partir de [34]

Tabla 57.Matriz específica de los stakeholders (Lanzamiento)

MATRIZ ESPECIFICA DE STAKEHOLDERS				
Fase	Lanzamiento			
Fecha inicio	20/01/2022			
Fecha Finalización	10/02/2022			
Id	Rol	Actividades	Nivel Implicación	Acceso Información
I-001	Gerente del proyecto	Establecer todos los elementos previos al lanzamiento.	5	----
		Chequear el correcto funcionamiento en las funciones posteriores al despliegue de la app.	4	----
		Crear matrices que permitan una correcta documentación.	3	Matriz historial de versiones

Fuente: Elaborado a partir de [34]

### 2.4.1.5.3 Asignación de actividades

Toda asignación de actividades se la representa mediante una matriz individual, en donde se desglosa una por una, se la diferencia mediante un identificador que es asignado o modificado acorde al gerente, al analista de requerimientos o tester, como observamos a continuación:

Tabla 58. Matriz de actividades de gerente

MATRIZ DE ACTIVIDADES		
Identificador	Actividad	Descripción
AG-01	Definir el entorno de la empresa (elaborar el análisis FODA, definir los objetivos del proyecto).	Actividad a realizar en la fase de la planificación por el gerente de proyecto.
AG-02	Establecer el modelo de negocio del proyecto (identificar los procesos, segmentar el mercado. Establecer reglas del negocio).	Actividad a realizar en la fase de planificación por el gerente de proyecto.
AG-03	Elaborar el análisis de factibilidad del proyecto (determinar los recursos humanos, materiales y tecnológicos).	Actividad a realizar en la fase de planificación por el gerente de proyecto.
AG-04	Determinar los resultados esperados (establecimiento de beneficios tangibles e intangibles para el resultado)	Actividad a realizar en la fase de planificación por el gerente de proyecto.
AG-05	Gestionar los interesados del proyecto (establecer roles a los stakeholders mediante una matriz general, específica y de actividades).	Actividad a realizar en la fase de planificación por el gerente de proyecto.
AG-06	Elaborar la EDT del proyecto basado en la metodología seleccionada.	Actividad a realizar en la fase de planificación por el gerente de proyecto.
AG-07	Gestionar el tiempo del proyecto mediante la creación de un cronograma que priorice los sprints y determine la holgura del proyecto.	Actividad a realizar en la fase de planificación por el gerente de proyecto.
AG-08	Gestionar los costos del proyecto mediante una estimación de los mismos y determinación del presupuesto.	Actividad a realizar en la fase de planificación por el gerente de proyecto.
AG-09	Establecer métricas y estándares de calidad para la aplicación móvil.	Actividad a realizar en la fase de planificación por el gerente de proyecto.
AG-10	Identificar todos los elementos previos al lanzamiento.	Actividad a realizar en la fase de planificación por el gerente de proyecto.
AG-11	Verificar el funcionamiento de las funciones posterior al despliegue de la app.	Actividad a realizar en la fase de planificación por el gerente de proyecto.
AG-12	Elaborar matrices que permitan una correcta documentación detallada.	Actividad a realizar en la fase de planificación por el gerente de proyecto.

Fuente: Elaborado a partir de [34]

Tabla 59. Matriz de actividades de analista de requerimientos

MATRIZ DE ACTIVIDAD		
Identificador	Actividad	Descripción
AAR-01	Elaborar la gestión del alcance del proyecto (Determinar los requerimientos funcionales y no funcional que llevara la aplicación mediante la utilización de plantillas generales y específicas).	Actividad a realizar en la fase de planificación por el analista de requerimientos.
AAR-02	Elaboración de las historias de usuario y creación del Sprint Bcklog.	Actividad a realizar en la fase de planificación por el analista de requerimientos.
AAR-03	Gestionar los cambios del proyecto mediante el uso de una plantilla para este fin, dirigida al gerente del proyecto.	Actividad a realizar en la fase de planificación por el analista de requerimientos.
AAR-04	Analizar los requerimientos mediante el reconocimiento y evaluación de problemas existentes haciendo uso de una plantilla para la perspectiva del análisis.	Actividad a realizar en la fase de planificación por el analista de requerimientos.

Fuente: Elaborado a partir de [34]

Tabla 60. Matriz de actividades de desarrollador

MATRIZ DE ACTIVIDADES		
Identificador	Actividad	Descripción
AD-01	Diseñar la aplicación en base a modelos con enfoque a la simplicidad y calidad.	Actividad a realizar en la fase de diseño por el desarrollador.
AD-02	Diseñar la base de datos no SQL.	Actividad a realizar en la fase de diseño por el desarrollador.
AD-03	Desarrollar los módulos que llevara la aplicación.	Actividad a realizar en la fase de diseño por el desarrollador.
AD-04	Implementar un control de versiones y editor de código.	Actividad a realizar en la fase de ejecución por el desarrollador.
AD-05	Implementar buenas prácticas el momento de codificar.	Actividad a realizar en la fase de ejecución por el desarrollador.

Fuente: Elaborado a partir de [34]

Tabla 61.Matriz de actividades de tester

MATRIZ DE ACTIVIDADES		
Identificador	Actividad	Descripción
AT-01	Detectar bugs y errores.	Actividad a realizar en la fase de pruebas por el tester.
AT-02	Identificar las fallas en el funcionamiento de la aplicación.	Actividad a realizar en la fase de pruebas por el tester.
AT-03	Corregir errores identificados.	Actividad a realizar en la fase de pruebas por el tester.
AT-04	Identificar posibles mejoras en los módulos funcionales.	Actividad a realizar en la fase de pruebas por el tester.

Fuente: Elaborado a partir de [34]

Habiendo definido las actividades por cada interesado del proyecto, se recopila la información más importante respecto a la fase en la que se encuentra dicho proyecto y se obtiene como documento final, el acta de constitución del proyecto, la cual la visualizamos a continuación:

Tabla 62. Acta de constitución del proyecto

ACTA DE CONSTITUCION DEL PROYECTO		
Empresa/Organización	Toda aquella cuenta con un área de cobranza.	
Proyecto	DESARROLLO DE UNA APLICACIÓN MÓVIL PARA EL PROCESO DE COBRANZA UTILIZANDO LA METODOLOGÍA MMS	
Fecha de preparación	03/11/2021	
Cliente	Usuarios en general	
Gerente de proyecto	Anibal Alonzo	
Justificación del proyecto		
<p>Toda empresa que presta dinero tiene un área de gestión de cobranza la misma que se ve en la obligación de contar no con uno si no con algunas personas encargadas de gestionar dicho trámite es por eso que es esencial que cada una de ellas llegue al destino sin tener ningún inconveniente ni retrasar el cobro, por eso es de gran ayuda implementar el uso de la ubicación mediante una foto, la misma que permitirá dar a conocer al próximo cobrador en la ubicación exacta, por ende este proceso de realce en ayudar a la empresa para que de este modo se agilite dicho proceso y se puede abarcar más mercado en el sentido de llegar a efectuar todos los cobros que constan en la empresa. Dando una imagen de prestigio al estar de acorde a la tecnología y ser eficiente en su área. Teniendo como resultado la creación de una aplicación de cobranza que me permita mitigar dichos percances brindando seguridad y valor del mismo.</p>		
Lista de Stakeholders		
Nombre	Cargo	Departamento/División
Anibal Alonzo	Gerente de proyecto, analista de requerimientos, equipo de desarrollo (desarrolladora, tester)	Departamentos de Sistemas
Ing. Milton Valarezo	Tutor encargado	--
Usuario final	Cliente	Departamento de la gestión de cobranza.
Premisas y Restricciones		
<p>Solo el administrador podrá generar los cambios a los usuarios, el cobrador Debra tener encendido su ubicación para así poder dar con el cliente o usuario y el cobrador podrá ver a las personas que debe de cobrar solo si está registrado, además de que al administrador le proporciones la contraseña para acceder.</p>		
Síntesis de análisis de factibilidad		
<p>En el análisis de factibilidad se deben tener en cuenta, tres tipos de factibilidades como lo son la factibilidad operativa, económica y técnica, lo cual es la guía para que un proyecto continúe en su desarrollo o no, con este proceso se establecen los resultados esperados ya sean estos tangibles o intangibles. De esa manera se llega a cumplir con los objetivos planteados dando como resultado beneficios a la empresa.</p>		
Presupuesto Inicial		
<p>El presupuesto del proyecto se encuentra basado en el análisis de factibilidad, el cual es de 1,888.55</p>		
Requisitos de aprobación del proyecto		
<p>En esta primera fase el proyecto debe tener la aprobación del entorno de la empresa como el modelo de negocio, análisis de factibilidad, resultados esperados y gestión de los stakeholders, para que la aplicación móvil tenga luz verde.</p>		
Aprobaciones		
Interesados/Patrocinadores	Fecha	Firma
Anibal Alonzo	03/11/2021	
Ing. Milton Valarezo	03/11/2021	
Usuario final	03/11/2021	

Fuente: Elaborado a partir de [34]

#### **2.4.1.6 Gestión del Alcance**

En este punto se determinan algunos aspectos que enmarcan el perfecto desarrollo del proyecto, los cuales están apuntando a diferentes direcciones que a su vez se enfocan en un mismo objetivo, el cual es tener un resultado de calidad como proyecto, en el transcurso de estas actividades se pueden diferenciar los requerimientos funcionales y no funcionales los que permiten tener una mejor visión del proyecto que se está desarrollando, en lo que se observa todas las cualidades que tiene el software. De esta manera se plantean las historias de usuarios, las cuales permiten tener en claro las distintas actividades de los stakeholders para así obtener todos los procesos de manera ágil y eficientes, teniendo un proyecto de calidad.

##### **2.4.1.6.1 Requerimientos funcionales y no funcionales**

###### **2.4.1.6.1.1 Requerimientos funcionales**

Los requerimientos funcionales que se enmarcan en el desarrollo del proyecto son las siguientes:

Tabla 63.Requerimientos funcionales

REQUERIMIENTOS FUNCIONALES		
Identificador	Requerimiento	Observación
RF-01	La aplicación contará con autenticación	Constará de dos usuarios, un administrador y un cobrador
RF-02	Crear, visualizar, actualizar y eliminar clientes	El administrador podrá realizar un CRUD de los clientes deudores.
RF-03	Visualizar clientes, cobradores, próximas deudas, reportes y configuración.	El administrador podrá visualizar las diferentes opciones, entre ellas los clientes, cobradores, próximas deudas, reportes y configuración.
RF-04	Comunicación por red social WhatsApp	Tanto el administrador como el cobrador podrán comunicarse con el cliente mediante WhatsApp
RF-05	Visualizar ubicación del cliente	El administrador podrá visualizar las ubicaciones de los clientes
RF-06	Agregar referencias	El administrador podrá agregar referencias sobre el lugar de residencia del cliente.
RF-07	Crear, visualizar, y ver detalladamente las deudas de los clientes	El administrador podrá crear deudas a los clientes, así como gestionar las mismas, mediante la creación, visualización y orden por fecha de las mismas, dicho orden va desde la fecha más reciente a la más antigua.
RF-08	Realizar abonos	Tanto el administrador como el cobrador podrán realizar abonos de los clientes.
RF-09	Visualizar la ruta del cobrador	El administrador podrá ver la ruta del cobrador, es decir podrá ver en tiempo real por donde se moviliza el cobrador y si ya llegó a su destino (cliente).
RF-10	Visualizar clientes asignados	El cobrador podrá ver a que clientes el administrador le ha asignado para realizar el cobro. Este cobro puede ser diario, semanal o mensual.
RF-11	Visualizar información de los clientes	Tanto el administrador como el cobrador podrá visualizar la información de cada cliente, así como las deudas contraídas por el mismo, fecha de la deuda, abonos y ubicación.
RF-12	Guardar ubicación en tiempo real.	El cobrador podrá tomar una foto de la casa del cliente, y ésta automáticamente se guardará junto con las coordenadas exactas en tiempo real, obtenidas de google maps.
RF-13	Realizar cobros	Tanto el administrador como el cobrador podrán realizar los cobros a los clientes, ya sea mediante pagos en efectivo, transferencias o depósito. Luego de haber efectuado el pago, se deberá subir el comprobante correspondiente.

Fuente: Elaborado a partir de [34]

Tabla 64.Requerimiento funcional #1

ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES					
Identificador	RF-01	Fecha de Recepción	03/11/2021	Versión	A
Responsable	Aníbal Alonzo				
Actor (es)	Administrador, cobrador				
Pre- Condición	Tener correo electrónico				
Post- Condición	Registrarse en la aplicación, iniciar sesión				
Descripción					
La aplicación contará con autenticación					
Observación					
La aplicación Constará de dos usuarios, un administrador y un cobrador					

Fuente: Elaborado a partir de [34]

Tabla 65.Requerimiento funcional #2

ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES					
Identificador	RF-02	Fecha de Recepción	03/11/2021	Versión	A
Responsable	Aníbal Alonzo				
Actor (es)	Administrador				
Pre- Condición	Haber seleccionado opción Clientes del menú principal				
Post- Condición	Ejecución de la opción seleccionada				
Descripción					
Crear, visualizar, actualizar y eliminar clientes					
Observación					
El administrador podrá realizar un CRUD de los clientes deudores.					

Fuente: Elaborado a partir de [34]

Tabla 66.Requerimiento funcional #3

ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES					
Identificador	RF-03	Fecha de Recepción	03/11/2021	Versión	A
Responsable	Aníbal Alonzo				
Actor (es)	Administrador				
Pre- Condición	Iniciar sesión				
Post- Condición	Ejecución de la opción seleccionada				
Descripción					
Visualizar clientes, cobradores, próximas deudas, reportes y configuración					
Observación					
El administrador podrá visualizar las diferentes opciones, entre ellas los clientes, cobradores, próximas deudas, reportes y configuración					

Fuente: Elaborado a partir de [34]

Tabla 67.Requerimiento funcional #4

ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES					
Identificador	RF-04	Fecha de Recepción	03/11/2021	Versión	A
Responsable	Aníbal Alonzo				
Actor (es)	Administrador, cliente				
Pre- Condición	Haber seleccionado un cliente				
Post- Condición	Chatear con el cliente seleccionado				
Descripción					
Comunicación por red social WhatsApp					
Observación					
Tanto el administrador como el cobrador podrán comunicarse con el cliente mediante WhatsApp					

Fuente: Elaborado a partir de [34]

Tabla 68.Requerimiento funcional #5

ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES					
Identificador	RF-05	Fecha de Recepción	03/11/2021	Versión	A
Responsable	Aníbal Alonzo				
Actor (es)	Administrador				
Pre- Condición	Haber seleccionado opción "Ubicaciones del menú Clientes"				
Post- Condición	Ejecución de la opción seleccionada				
Descripción					
Visualizar ubicación del cliente					
Observación					
El administrador podrá visualizar las ubicaciones de los clientes					

Fuente: Elaborado a partir de [34]

Tabla 69.Requerimiento funcional #6

ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES					
Identificador	RF-06	Fecha de Recepción	03/11/2021	Versión	A
Responsable	Aníbal Alonzo				
Actor (es)	Administrador				
Pre- Condición	Haber visualizado las ubicaciones				
Post- Condición	Guardar referencias				
Descripción					
Agregar referencias					
Observación					
El administrador podrá agregar referencias sobre el lugar de residencia del cliente.					

Fuente: Elaborado a partir de [34]

Tabla 70.Requerimiento funcional #7

ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES					
Identificador	RF-07	Fecha de Recepción	03/11/2021	Versión	A
Responsable	Aníbal Alonzo				
Actor (es)	Administrador				
Pre- Condición	Haber seleccionado opción "Crear deuda" del menú clientes				
Post- Condición	Ejecución de la opción seleccionada				
Descripción					
Crear, visualizar, y ver detalladamente las deudas de los clientes					
Observación					
El administrador podrá crear deudas a los clientes, así como gestionar las mismas, mediante la creación, visualización y orden por fecha de las mismas, dicho orden va desde la fecha más reciente a la más antigua.					

Fuente: Elaborado a partir de [34]

Tabla 71.Requerimiento funcional #8

ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES					
Identificador	RF-08	Fecha de Recepción	03/11/2021	Versión	A
Responsable	Aníbal Alonzo				
Actor (es)	Administrador				
Pre- Condición	Haber visualizado deudas y abonos				
Post- Condición	Guardar abono				
Descripción					
Realizar abonos					
Observación					
Tanto el administrador como el cobrador podrán realizar abonos de los clientes.					

Fuente: Elaborado a partir de [34]

Tabla 72.Requerimiento funcional #9

ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES					
Identificador	RF-09	Fecha de Recepción	03/11/2021	Versión	A, C
Responsable	Aníbal Alonzo				
Actor (es)	Administrador				
Pre- Condición	Haber seleccionado las ubicaciones				
Post- Condición	Ejecución de la opción seleccionada				
Descripción					
Visualizar la ruta del cobrador					
Observación					
El administrador podrá ver la ruta del cobrador, es decir podrá ver en tiempo real por donde se moviliza el cobrador y si ya llegó a su destino (cliente).					

Fuente: Elaborado a partir de [34]

Tabla 73.Requerimiento funcional #10

ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES					
Identificador	RF-10	Fecha de Recepción	03/11/2021	Versión	B
Responsable	Aníbal Alonzo				
Actor (es)	Cobrador				
Pre- Condición	Haber seleccionado opción del menú general				
Post- Condición	Ejecución de la opción seleccionada				
Descripción					
Visualizar clientes asignados					
Observación					
El cobrador podrá ver a que clientes el administrador le ha asignado para realizar el cobro. Este cobro puede ser diario, semanal o mensual					

Fuente: Elaborado a partir de [34]

Tabla 74.Requerimiento funcional #11

ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES					
Identificador	RF-11	Fecha de Recepción	03/11/2021	Versión	B
Responsable	Aníbal Alonzo				
Actor (es)	Administrador, Cobrador				
Pre- Condición	Haber seleccionado un determinado cliente				
Post- Condición	Ejecución de la opción seleccionada				
Descripción					
Visualizar información de los clientes					
Observación					
Tanto el administrador como el cobrador podrá visualizar la información de cada cliente, así como las deudas contraídas por el mismo, fecha de la deuda, abonos y ubicación					

Fuente: Elaborado a partir de [34]

Tabla 75.Requerimiento funcional #12

ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES					
Identificador	RF-12	Fecha de Recepción	03/11/2021	Versión	B, C
Responsable	Aníbal Alonzo				
Actor (es)	Cobrador				
Pre- Condición	Haber seleccionado "Agregar ubicación"				
Post- Condición	Guardar ubicación				
Descripción					
Guardar ubicación en tiempo real.					
Observación					
El cobrador podrá tomar una foto de la casa del cliente, y ésta automáticamente se guardará junto con las coordenadas exactas en tiempo real, obtenidas de google maps.					

Fuente: Elaborado a partir de [34]

Tabla 76.Requerimiento funcional #13

ESPECIFICACION DE REQUERIMIENTOS FUNCIONALES					
Identificador	RF-13	Fecha de Recepción	03/11/2021	Versión	B
Responsable	Aníbal Alonzo				
Actor (es)	Administrador, cobrador				
Pre- Condición	Haber visualizado información de las deudas y abonos realizados				
Post- Condición	Subir comprobante de transacción.				
Descripción					
Realizar cobros o abonos					
Observación					
Tanto el administrador como el cobrador podrán realizar los cobros a los clientes, ya sea mediante pagos en efectivo, transferencias o depósito. Luego de haber efectuado el pago, se deberá subir el comprobante correspondiente.					

Fuente: Elaborado a partir de [34]

#### 2.4.1.6.1.2 Requerimientos no funcionales

Los requerimientos no funcionales que se enmarcan en el desarrollo del proyecto son las siguientes:

Tabla 77.Requerimientos no funcionales

REQUERIMIENTOS NO FUNCIONALES		
Identificador	Requerimiento	Observación
RNF-01	El usuario es capaz de realizar todas las funciones del sistema.	Ninguna
RNF-02	Cuando exista más de 100 usuario conectados, el tiempo de respuesta de la aplicación será no más de 2 segundos.	Ninguna
RNF-03	Si existiera un fallo en el sistema el tiempo de respuesta en recuperación no será más de 5 minutos.	Ninguna

Fuente: Elaborado a partir de [34]

Tabla 78.Requerimiento no funcional #1

ESPECIFICACION DE REQUERIMIENTOS NO FUNCIONALES					
Identificador	RNF-01	Fecha de recepción	03/11/2021	Versión	A
Responsable	Anibal Alonzo				
Actor (es)	Cliente y Sistema				
Pre- Condición	Acceder al sistema				
Post- Condición	Se ejecuta a la acción con un tiempo de respuesta menor o igual a 2 segundos.				
Descripción					
Un usuario debe ser capaz de poder utilizar todas las funciones del sistema.					
Observación					
Ninguna					

Fuente: Elaborado a partir de [34]

Tabla 79.Requerimiento no funcional #2

ESPECIFICACION DE REQUERIMIENTOS NO FUNCIONALES					
Identificador	RNF-02	Fecha de recepción	03/11/2021	Versión	A
Responsable	Anibal Alonzo				
Actor (es)	Cliente y sistema.				
Pre- Condición	Acceden al sistema más de 100 personas.				
Post- Condición	No permitir el acceso al sistema.				
Descripción					
No existirá retrasos por parte del sistema ya que, aunque tengan más de 100 no se presentaran problemas mayores a 2 segundos.					
Observación					
Ninguna					

Fuente: Elaborado a partir de [34]

Tabla 80.Requerimiento no funcional #3

ESPECIFICACION DE REQUERIMIENTOS NO FUNCIONALES					
Identificador	RNF-03	Fecha de recepción	03/11/2021	Versión	A
Responsable	Anibal Alonzo				
Actor (es)	Cliente y sistema				
Pre- Condición	Se produzca un fallo				
Post- Condición	Reiniciar el sistema				
Descripción					
El sistema no tardara más de 5 minutos para recuperarse y volver a cargar los datos e inmediatamente ponerse en marcha.					
Observación					
Ninguna					

Fuente: Elaborado a partir de [34]

#### 2.4.1.6.2 Elicitación de historias de usuario

La elicitación de las historias de usuario, permite tener la solución al momento de combinar cada historia de usuario hacia un bosquejo de la aplicación, ya que en esta fase se realiza el diseño de cada interfaz, para entender el desarrollo de las demás fases en cuanto a la vida del proyecto. Teniendo procesos como; preparación, generación, consolidación y su documentación [41].

Tabla 81.Historia de Usuario #1

HISTORIA DE USUARIO	
Identificador	HU-01
Actor	Administrador, cobrador y sistema
Prioridad	Alta
Sprint	A
Responsable	Anibal Alonzo
Requerimiento Funcional	
La aplicación contará con autenticación	
Observaciones	
Registro e inicio de sesión de usuarios de la aplicación	
Bosquejos - Prototipos	
<p>Figura 5. Bosquejo Registro e inicio de sesión</p> <p>Fuente: Elaboración propia</p>	

Fuente: Elaborado a partir de [34]

Tabla 82.Historia de Usuario #2

HISTORIA DE USUARIO	
Identificador	HU-02
Actor	Administrador, cobrador y sistema
Prioridad	Alta
Sprint	A
Responsable	Anibal Alonzo
Requerimiento Funcional	
Visualizar cliente y registrar cliente, registrar deuda y ver detalle del cliente más su deuda, además de colocar notas en la deuda como recordatorios.	
Observaciones	
El administrador agregara los clientes y sus deudas además de visualizar los mismo.	
Bosquejos - Prototipos	
<p>Figura 6. Bosquejo Gestión Administrador</p> <p>Fuente: Elaboración propia</p>	

Fuente: Elaborado a partir de [34]

Tabla 83.Historia de Usuario #3

HISTORIA DE USUARIO	
Identificador	HU-03
Actor	Cobrador y sistema
Prioridad	Alta
Sprint	B
Responsable	Anibal Alonzo
Requerimiento Funcional	
Visualizar cobradores y agregar los mismo, además de ver el avance del mismo hacia el cliente.	
Observaciones	
Asignas cobradores, ver cobradores y dar seguimiento a los mismos.	
Bosquejos - Prototipos	
	
<p>Figura 7. Bosquejo Gestión Cobrador</p> <p>Fuente: Elaboración propia</p>	

Fuente: Elaborado a partir de [34]

Tabla 84.Historia de Usuario #4

HISTORIA DE USUARIO	
Identificador	HU-04
Actor	Cobrador y sistema
Prioridad	Alta
Sprint	C
Responsable	Anibal Alonzo
Requerimiento Funcional	
Visualizar clientes con deuda, visualizar monto y ruta	
Observaciones	
Podrá agregar una foto del cliente y se guardará su ubicación	
Bosquejos - Prototipos	
	

Figura 8. Bosquejo Gestión Seguimiento Cobrador

Fuente: Elaboración propia

Fuente: Elaborado a partir de [34]

### 2.4.1.6.3 Creación del Sprint Backlog

En este punto se manifiestan los entregables que ha tenido el proyecto durante la fase de ejecución de la aplicación, los cuales son 2 que se han realizado en base a las diferentes modificaciones que tiene la aplicación.

Tomando en consideración las actividades que se realizaron, el primer sprint que se manifiesta es el administrador.

Tabla 85.Sprint Basklog A

SPRINT BASKLOG															
Fecha inicio	Fecha de Culminación	Sprint	Requerimiento	Tarea	Responsable	Semana	1								
						Días	1	2	3	4	5	6	7		
						Horas									
10/11/2021	17/11/2021	Sprint A	La aplicación contara con autenticación	Crear funciones necesarias que permitan registrarse a un cobrador.	Anibal Alonzo		x	x							
				Desarrollar las funciones necesarias para permitir el acceso a la aplicación mediante inicio de sesión				x	x						
			Registrarse, ingresar, agregar clientes y deudas, agregar direcciones	Ingresar clientes con sus datos y sus deudas.	Anibal Alonzo							x	x		
				Crear detalle de deuda, fecha y plazos a pagar.								x	x		
				Crear, visualizar y asignar a un cobrador por cliente.								x	x		
			Administrar a cobradores, crear detalle de cliente.	Validar a los cobradores y gestionar el cobro de clientes.	Anibal Alonzo										x

Fuente: Elaborado a partir de [34]

El segundo sprint que se ha manifestado según las actividades que se realizaron es el cobrador al momento de ingresar a la aplicación.

Tabla 86.Sprint Basklog B

SPRINT BASKLOG														
Fecha inicio	Fecha de Culminación	Sprint	Requerimiento	Tarea	Responsable	Semana	2							
						Días	1	2	3	4	5	6	7	
						Horas								
18/11/2021	24/11/2021	Sprint B	Agregar cobradores, visualizarlos, asignar un cobrador al cliente, ver la ruta del cliente a cobrar, hacer el abono y observar la trayectoria del administrador hacia el cobrador.	Agregar cobradores por parte del administrador	Anibal Alonzo		x	x						
				Visualización de ruta el cliente por parte del administrador y cobrador				x	x					
				Abono del cobrador por parte del cliente					x	x				
				Anexar el comprobante de depósito por el valor del pedido.	Anibal Alonzo						x	x		
				Seguimientos a los cobradores para saber si están próximos a llegar o aun demoran.	Anibal Alonzo								x	x

Fuente: Elaborado a partir de [34]

El tercer sprint que se ha manifestado según las actividades que se realizaron es el cobrador al momento de capturar la ubicación del cliente al momento de realizar el cobro.

Tabla 87.Sprint Basklog C

SPRINT BASKLOG														
Fecha inicio	Fecha de Culminación	Sprint	Requerimiento	Tarea	Responsable	Semana	3							
						Días	1	2	3	4	5	6	7	
						Horas								
25/11/2021	12/12/2021	Sprint C	Aceptar un cliente que el administrador le asigne.	Desarrollar las funciones necesarias	Anibal Alonzo		x	x						
			Visualizar la ubicación del cliente	Desarrollar las funciones necesarias	Anibal Alonzo			x						
			Ver ubicación o referencia del cliente	Desarrollar las funciones necesarias	Anibal Alonzo				x					
			Cuando haya más de 100 usuarios el sistema contara con un retraso de 2 segundos.	Desarrollar las funciones necesarias	Anibal Alonzo					x	x			
			Agregar ubicación mediante una foto por parte del cobrador	Desarrollar las funciones necesarias	Anibal Alonzo							x	x	
			Agregar forma de pago.	Desarrollar las funciones necesarias	Anibal Alonzo									x

Fuente: Elaborado a partir de [34]

#### 2.4.1.6.4 Elaboración del EDT

Un EDT, es esencial al momento de detallar la estructura de desglose de todo trabajo, ya que cada elemento será distribuido según el tiempo establecido en el cronograma, teniendo en consideración las actividades o líneas de costos [42].

De esta manera se asegura el logro del proyecto mediante el intercambio de experiencias, ya que se potencian las actividades que cada steckholder puede desarrollar y esto permite mitigar todos los detalles impulsando una mejor gestión de riesgos. [43]



Figura 9. EDT del proyecto

Fuente: Elaboración propia

### 2.4.1.7 Gestión del Tiempo

La gestión del tiempo es importante que permite tener el control por etapas según el cronograma que hemos establecido, siendo fundamental en la ejecución del proyecto, debido a la buena administración que se le da al recurso humano y las actividades que desempeña cada uno de ellos en las distintas áreas, de esta forma se eliminan los riesgos de retrasos, aumentando los buenos tiempos de entrega, por ende, debe existir una buena holgura del proyecto, acompañada de un buen cronograma [39].

#### 2.4.1.7.1 Priorización del Sprint

Los sprint son importantes para dar tiempo a determinadas tareas, ya que la gestión del alcance, define como prioridad a los Sprint A y C [44].

#### 2.4.1.7.2 Holgura del proyecto

Todo proyecto posee un determinado número de actividades, en donde se debe tener en cuenta su tiempo de culminación y adicionarle entre uno o dos días más, así se prevé cualquier circunstancia que se presente o bien en si se da un cierto tiempo al finalizar el proyecto.

### 2.4.1.7.3 Elaboración del cronograma

Tabla 88.Cronograma del proyecto

TAREA	ASIGNADO	PROGRESO	INICIO	FIN
Aplicación móvil para la gestión de cobranza	Anibal Alonso	100%	03/11/2021	18/02/2022
<b>Planificación</b>			03/11/2021	17/11/2021
Definir el entorno de la empresa (elaborar el análisis FODA, definir los objetivos del proyecto)		100%	03/11/2021	06/11/2021
Establecer el modelo de negocio del proyecto (identificar los procesos segmentar el mercado, establecer reglas del negocio)	Anibal Alonso	100%	03/11/2021	06/11/2021
Elaborar el análisis de factibilidad del proyecto (determinar los recursos humanos, materiales y tecnológicos)	Anibal Alonso	100%	03/11/2021	06/11/2021
Determinar los resultados esperados (establecimiento de beneficios tangibles e intangibles para la empresa)	Anibal Alonso	100%	03/11/2021	06/11/2021
Gestionar los interesados del proyecto (establecer roles a los stakeholders mediante una matriz general, específica y de actividades)	Anibal Alonso	100%	03/11/2021	06/11/2021
Elaborar el EDT del proyecto basado en la metodología seleccionada.	Anibal Alonso	100%	07/11/2021	08/11/2021
Gestionar el tiempo del proyecto mediante la creación de un cronograma que priorice los sprints y determine la holgura del proyecto.	Anibal Alonso	100%	09/11/2021	10/11/2021
Gestionar los costos del proyecto mediante una estimación de los mismos y determinación del presupuesto	Anibal Alonso	100%	11/11/2021	12/11/2021
Establecer métricas y estándares de calidad para la aplicación móvil	Anibal Alonso	100%	13/11/2021	13/11/2021
Elaborar la gestión de alcance del proyecto (Determinar los requerimientos funcionales y no funcionales que llevará la aplicación mediante la utilización de plantillas generales y específicas.	Anibal Alonso	100%	14/11/2021	15/11/2021
Elaboración de las historias de usuario y reacción de Sprint Backlog	Anibal Alonso	100%	16/11/2021	16/11/2021
Gestionar los cambios del proyecto mediante el uso de una plantilla para este fin, dirigida al gerente del proyecto.	Anibal Alonso	100%	17/11/2021	17/11/2021
<b>DISEÑO</b>			18/11/2021	08/12/2021
Comprobar los requerimientos en base al reconocimiento y evaluación de problemas existentes haciendo uso de una plantilla para la perspectiva del análisis.	Anibal Alonso	100%	19/11/2021	23/11/2021
Diseñar la aplicación mediante el enfoque a la simplicidad y calidad.	Anibal Alonso	100%	24/11/2021	29/11/2021
Diseñar la base de datos no SQL y su direccionarlo de datos	Anibal Alonso	100%	30/11/2021	04/12/2021
Diseñar los módulos que llevara la aplicación	Anibal Alonso	100%	05/12/2021	08/12/2021
<b>Ejecución</b>			09/12/2021	29/12/2021
Introducir en el proceso un control de versiones y editor de código	Anibal Alonso	100%	10/12/2021	14/12/2021
Introducir buenas prácticas y codificar	Anibal Alonso	100%	15/12/2021	16/12/2021
SPRINT A	Anibal Alonso	100%	17/12/2021	19/12/2021
SPRINT B	Anibal Alonso	100%	20/12/2021	20/12/2021
SPRINT C	Anibal Alonso	100%	21/12/2021	23/12/2021
<b>Pruebas</b>			30/12/2021	19/01/2022
Identificar bugs y errores	Anibal Alonso	100%	31/12/2021	05/01/2022
Detectar fallas en el funcionamiento de la aplicación	Anibal Alonso	100%	06/01/2022	09/01/2022
Corregir errores identificados	Anibal Alonso	100%	10/01/2022	14/01/2022
Detectar posibles mejoras en los módulos funcionales	Anibal Alonso	100%	15/01/2022	19/01/2022
<b>Lanzamiento</b>			20/01/2022	10/02/2022
Tener en claro todos los elementos previos al lanzamiento	Anibal Alonso	100%	21/01/2022	24/01/2022
Verificar el funcionamiento de las funciones posterior al despliegue de la aplicación	Anibal Alonso	100%	25/01/2022	29/01/2022
Elaborar matrices que permitan una correcta documentación detallada.	Anibal Alonso	100%	30/01/2022	02/02/2022
Redacción el informe final	Anibal Alonso	100%	03/02/2022	05/02/2022
Aprobación del documento final de titulación	Anibal Alonso	100%	06/02/2022	10/02/2022

Fuente: Elaboración propia

### 2.4.1.8 Gestión de Costos

La gestión de costos, es tratar el presupuesto del proyecto de una manera que pueda llegar hasta la culminación del presente, sin tener la necesidad de replantear el costo, se debe tener en cuenta cualquier imprevisto; por eso se realizan pruebas o simulaciones, además se determina los gastos tangibles e intangibles por ende se priorizan los aspectos como la estimación de los mismos más la determinación de un presupuesto.

#### 2.4.1.8.1 Estimación de Costos

Para realizar la estimación de costos en todo proyecto se debe tener en cuenta muchos factores por eso este proceso, es un proceso tedioso, para esto una solución más eficaz es el planteamiento del modelo Sprint, el mismo que es un modelo de costos libre.

#### 2.4.1.8.2 Determinación del presupuesto

Tabla 89. Determinación del presupuesto

DETERMINACION DE PRESUPUESTO	
Recurso Humano	\$600
Recurso Material	\$3.55
Recurso Tecnológico	\$1.200
Recursos Generales	\$0
Imprevistos	\$0
<b>PRESUPUESTO</b>	<b>\$1888.55</b>

Fuente: Elaborado a partir de [34]

### 2.4.1.9 Gestión de Cambios

Todo proyecto tiende a sufrir cambios durante su desarrollo, es por eso que cada cambio se debe registrar para ver que tanto se ha desviado del objetivo o si a superado al mismo, de esta manera se determina su viabilidad o éxito, por ende, cada cambio tiene un control denominado matriz de seguimientos o solicitudes de cambio.

### 2.4.1.9.1 Solicitudes de Cambio

Todo cambio debe ser positivo y antes de ser ejecutado se debe enseñar al gerente del proyecto el cual determinara si dicha solicitud sea aprobada en bien del proyecto o denegada en caso de que afecte al mismo, es por eso que en el presente se ha determinado una solicitud de cambio que a continuación se detalla:

Tabla 90.Solicitudes de Cambios

SOLICITUD DE CAMBIO					
N. Solicitud	SC1				
Solicitante	Cobrador				
Área/Departamento					
Gerente	Anibal Alonzo				
Categoría de Cambio					
Alcance		Procedimientos		Recursos	
Cronograma		Documentación		Otros	x
Costos		calidad			
Causa de Cambio					
Alcance		Reparación			
Cronograma		Otros	x		
Costos					
Descripción de la propuesta de cambio					
Modificación en la interfaz de la aplicación para que sea fácil de comprender e intuitiva al ser usada.					
Justificación de la propuesta de cambio					
No se comprende el funcionamiento al realizar la toma de ubicación por parte del administrador.					
Impacto de cambio					
Alcance	Ninguno				
Cronograma	Ninguno				
Costo	Ninguno				
Calidad	Ninguno				
Implicaciones de recursos					
Recursos humanos y tecnológicos					

Fuente: Elaborado a partir de [34]

### 2.4.1.9.2 Matriz de seguimiento de cambios

El cambio que se realizó se lo llevo de una manera organizada y con un correcto control, de esta manera se dio paso a lo solicitado por el cliente.

Tabla 91.Matriz de seguimiento de cambios

GESTION DE CAMBIOS					
Fecha	Identificador	Fase/Sprint	Descripción general	Solicitante	Estado
15/12/2021	SC1	Ejecución / Sprint A	Modificación en la interfaz para que esta sea intuitiva y fácil de comprender.	Cobrador	Aceptado

Fuente: Elaborado a partir de [34]

### 2.4.1.10 Gestión de Calidad

En el presente proyecto se usará una norma de calidad, ISO/IEC 25010, siendo una de tantas que cumplen con las buenas prácticas, de esta manera se da a conocer que en esta etapa de proyecto se describen los estándares y métricas de calidad, la mismas que se detallara a continuación:

#### 2.4.1.10.1 Métricas de calidad para aplicaciones móviles

Para que los resultados del presente proyecto sea un software de calidad, mantenible y 100% productivo se establecen métricas de calidad, las cuales son las siguientes:

Se estableció que el software posea una interfaz amigable e intuitiva para el usuario.

- La aplicación no genera exceso de recursos por lo que cumple con minimizar los recursos del hardware como son la memoria, almacenamiento y red, maximizando la app.
- Los datos están perfectamente seguros, garantizando la privacidad y su disponibilidad.
- Cabe resaltar que la aplicación es para uso único del sistema operativo Android.

#### **2.4.1.10.2 Estándares de calidad**

El estándar utilizado es la ISO/IEC 25010, la misma que garantiza que el proyecto cumpla con las siguientes características:

- Portabilidad
- Adecuación funcional
- Seguridad
- Fiabilidad
- Eficiencia de desempeño
- Usabilidad

#### **2.4.2 Diseño**

En la fase de diseño se transforman las ideas en bocetos mediante cada circunstancia que genere el cliente, generando una recopilación de información, se crean soluciones que permitan dar con el inicio del proyecto siendo estas de manera ágil, que se basan en un modelo de base de datos

##### **2.4.2.1 Análisis de requerimientos**

Para que se genere un buen análisis de requerimientos se establecen un sin número de reconocimientos del problema de esta manera se evalúa dicho problema y se determina su solución, es por eso que se detallan sus cualidades:

###### **2.4.2.1.1 Reconocimiento y evaluación del problema**

Un administrador de cobranza, para tener éxito debe estar al día en el cobro de los clientes, para que no existe ningún percance al momento de cerrar caja, de esta manera se puede apreciar la relación cliente-cobrador, por ende, se genera una seguridad y se eliminan pérdida de tiempo, al ser exacto con el cobro.

La gestión de cobranza en las empresas, no cumplen con el factor de ubicación exacta ya que seguían al principio mediante sugerencias o direcciones cercanas al domicilio del deudor, por ende no es eficiente ya que la persona que va directamente es la que conoce el lugar de residencia más no una tercera y es por eso que se limita la función de cobranza, aumentando la pérdida de tiempo y malestar del cliente al estar pendiente del cobrador, por tal motivo se propone crear una aplicación móvil que permita contar con la ubicación al momento de

tomar la foto en el lugar de cobro y es ahí donde se puede aprovechar a evitar los errores de ubicación para una tercera persona, por ende también el administrador cuenta con el seguimiento al cobrador, logrando aprovechar el tiempo del mismo, hacia una nueva cobranza, haciendo que este proceso se a más eficiente.

#### 2.4.2.1.2 Observaciones múltiples

Una vez identificado el problema, se procede a designar dos soluciones de acuerdo a los puntos de vistas. La primera solución se enfoca en dar comodidad al cobrador al momento de realizar su trabajo, la segunda se establece en la comodidad de llevar sus usuarios y tramites como administrador.

Tabla 92.Observaciones múltiples

Perspectiva de análisis			
Código:	PA-01	Usuario:	Cobrador
Descripción:	La aplicación cuenta con dos tipos de usuarios, el primero que es el administrador, el cual es encargado de gestionar toda las funciones y un segundo usuario que permite gestionar la ubicación del cliente mediante una foto la cual cuenta con ubicación exacta de la su casa, cabe realzar que ambos usuarios ingresan al sistema mediante una autenticación.		
Observaciones:	Está relacionada al módulo de gestión de cobrador		
Numero de iteración:	1	Nivel prioridad:	1

Fuente: Elaborado a partir de [34]

Tabla 93.Observaciones múltiples

Perspectiva de análisis			
Código:	PA-02	Usuario:	Cobrador
Descripción:	Al momento de registrar el monto de un cliente el cobrador debe elegir otro cliente asignado por el administrador		
Observaciones:	Este proceso se relaciona el módulo de ubicación del cliente		
Numero de iteración:	2	Nivel prioridad:	3

Fuente: Elaborado a partir de [34]

#### **2.4.2.2 Planteamiento de solución ágil**

La aplicación móvil cuenta con dos tipos de usuario; el primero, un administrador que se encarga de realizar todas las operaciones del sistema y como segundo usuario, un cobrador; que es la persona encargada de gestionar la parte de cobranzas, en la cual se observan a los clientes que el administrador le asigna.

Además, existe la opción de capturar la ubicación del cliente, mediante una imagen; tomada en el momento de ir a realizar la cobranza, de esta manera su trabajo es eficiente ya que permite a los futuros cobradores llegar al sitio de una forma precisa. Por otro lado, el administrador, puede dar seguimiento a todos sus cobradores, debido a que cuenta con rastreo al momento de efectuar los cobros. La aplicación también cuenta con un listado de personas que deben, las cuales son enlistadas según el orden que se debe ir a realizar el cobro, facilitando la productividad en la gestión de cobranza, ya que este proceso se lo realiza de una forma ordenada y mitigando los percances en el día a día.

Existen 3 entregables en el desarrollo de la aplicación, los mismo que se han denominado SPRINT A, es el encargado de la parte administrativa como el ingreso de usuarios, autenticaciones, permisos y demás actividades; el SPRINT B, se implementan los requerimientos funcionales en la gestión de cobranzas, de esta manera se aplica los conceptos de desarrollo ágil y el SPRINT C, en el cual el cobrador puede capturar la ubicación y obtener las coordenadas en tiempo real del cliente deudor.

#### **2.4.2.3 Modelado de la base de datos**

Toda base de datos debe tener un buen modelo ya que esto permite ejecutar los procesos en base a los datos que se obtengan, por ende es de buen saber que el desarrollo de la aplicación debe contar con diversos tipos de modelos para que así se obtengan casos de buen proceder, es por eso que el sistema cuenta con el modelo entidad- relación, modelo relacional, pero cabe resaltar que implemento una base de datos NoSQL, la misma que permite una mayor agilidad de datos representados en un esquema de acorde a lo que se utiliza en la aplicación [45].

### 2.4.2.3.1 Esquema NoSQL

Este tipo de esquemas utilizan datos que son uniformes pero que poseen una gran flexibilidad y esto permite que se los trabaje con mayor facilidad debido a que si se desea implementar un módulo más no se tendría que borrar tablas o reescribir el modelo, sin embargo, este no es el caso ya que, al ser no relacional, se pueden dar cambios en diversas entidades, siendo este el caso del presente proyecto las cuales se describirán a continuación:

### 2.4.2.4 Definición de interfaz

Toda aplicación debe tener una buena interfaz ya que esta permite al usuario navegar por la mismas, siendo de mayor preferencia en tuitiva, ya que ofrece cumplir con el objetivo de brindar una mayor funcionalidad, por ende, la aplicación esta creada por módulos, los cuales permiten al usuario tener una mejor experiencia

#### 2.4.2.4.1 Diseño modular

Según la metodología seleccionada la aplicación se divide en los siguientes módulos:

Tabla 94.Diseño Modular

Módulos	Funcionalidades
Módulo de autenticación	--- Administra las sesiones de los usuarios, mediante correo electrónico y contraseña
Módulo de gestión de datos	--- Administra los procesos internos en base a la accesibilidad de la aplicación.
Módulo de coordinador	--- Se enfoca en desarrollar un diseño que sea simple pero interactivo, para que así mejore la experiencia del usuario.
Módulo de usuarios	--- Administra los usuarios que se registran en la aplicación (administrador y cobrador).
Módulo de contención	--- Administra las funciones del sistema, mediante una correcta distribución.
Módulo de servicios	--- Administra los servicios que oferta la aplicación.

Fuente: Elaborado a partir de [34]

### 2.4.3 Ejecución

En esta etapa se ponen a prueba distintas características como lo es la edición de código la cual fue establecida mediante la herramienta Visual Studi Code y Json Editor Online, además para el control de versiones se trabajó con la herramienta GitHub, se mantuvo en consideración los aspectos de versionamiento y edición de código, además se establecieron etapas o fases de codificación, pruebas y refactorización.

#### 2.4.3.1 Codificación

La codificación es la forma práctica en un proyecto de pasar los requerimientos a código para su ejecución, en cualquier lenguaje de programación en nuestro caso se realizará con el lenguaje Android Studio en su plataforma Java, mediante sus estándares y documentación haciendo uso de las buenas prácticas.

##### 2.4.3.1.1 Convenciones para el código

Para que exista un código de calidad o consistente es necesario realizar las buenas prácticas, como los nombres entendibles de las variables y de las funciones que ocuparemos, además de usar correctamente la sangría; de esta manera se mitigan las demoras por parte de un depuramiento en el futuro.

En la presente imagen se puede apreciar el correcto uso de las buenas prácticas en su totalidad haciendo del código un código entendible y presto a futuros cambios o mantenimientos.

```
public void registrar_cliente(final Cliente cli, final Interfaces.estado estado) {  
  
    try {  
  
        databaseReference.child("clientes").push().setValue(cli).addOnCompleteListener(task -> {  
            estado.verestado(task.isSuccessful());  
        });  
  
    } catch (Exception e) {  
        Log.e( tag: "ERROR", e.getMessage());  
    }  
  
}
```

Figura 10. Convenciones para el código.

Fuente: Elaboración propia

### 2.4.3.1.2 Documentación

Al momento de codificar es necesario el uso de la documentación ya que este proceso forma parte del correcto funcionamiento en las buenas prácticas, detectando los diferentes atributos del software.

```
/**
 * @param cli Objeto de Tipo Cliente
 * @param estado Obtención del estado del Cliente
 * @implNote Registro de Cliente
 */
public void registrar_cliente(final Cliente cli, final Interfaces.estado estado) {

    try {

        databaseReference.child("clientes").push().setValue(cli).addOnCompleteListener(task -> {
            estado.verestado(task.isSuccessful());
        });

    } catch (Exception e) {
        Log.e( tag: "ERROR", e.getMessage());
    }

}
```

Figura 11. Documentación

Fuente: Elaboración propia

### 2.4.3.2 Refactorización

La refactorización del código es esencial ya que permite depurar las funciones anidadas sin tener que cambiar la esencia o funcionamiento del proyecto, un ejemplo claro es la omisión de controles y sentencias anidadas, las mismas que se describen a continuación.

#### 2.4.3.2.1 Omitir el uso de controles y sentencias anidadas

En el presente código se hace uso de las buenas costumbres, en donde se evitó programar con funciones anidadas, en su caso se las reemplazo por funciones menos complejas, de esta forma se busca reducir código y agilizar el tiempo de respuesta del sistema.

```

/**
 * @implNote Obtener los Datos para el dashboard
 */
Fragment_cobranza.cobranza_dao.TotalDeudas(deuda ->
{
    txt_deuda.setText("Total Deuda: $" +deuda);

    Fragment_cobranza.cobranza_dao.TotalRecaudado(recaudado ->
    {
        txt_recaudado.setText("Total Recaudado: $" +recaudado);

        Double por_recaudado = (recaudado * 100)/deuda;

        txt_porcentaje.setText(Math.round(por_recaudado)+"%");
        progress_dashboard.setProgress(((Integer)Integer.parseInt(String.valueOf(Math.round(por_recaudado)))));

    });
});
});

```

Figura 12. Optimización "Antes"

Fuente: Elaboración propia

```

/**
 * @implNote Obtener los Datos para el dashboard
 */
Fragment_cobranza.cobranza_dao.TotalDeudas_recaudado((deuda, recaudado) -> {

    txt_deuda.setText("Total Deuda: $" +deuda);
    txt_recaudado.setText("Total Recaudado: $" +recaudado);

    Double por_recaudado = (recaudado * 100)/deuda;

    txt_porcentaje.setText(Math.round(por_recaudado)+"%");

    progress_dashboard.setProgress(((Integer)Integer.parseInt(String.valueOf(Math.round(por_recaudado)))));

});
});

```

Figura 13. Optimización "Después"

Fuente: Elaboración propia

## 2.4.4 Pruebas

Todo trabajo o desarrollo sea aplicación móvil o web, se le debe realizar todo tipo de pruebas para así eliminar los problemas que se presenten y estar seguro de la funcionalidad para la cual se crea el sistema con el fin de que el mismo sea estable y eficiente. Por lo que se creó una bitácora para llevar el control y fecha; de las pruebas que se le han realizado.

Tabla 95.Pruebas

BITACORA DE PRUEBAS			
N°	Fecha de ejecución	Responsable	Tipo de prueba
1	20/12/2021	Anibal Alonzo	Prueba de calidad
2	25/12/2021		Prueba de tendencia
3	30/12/2021		Prueba de usabilidad
4	06/01/2022		Prueba de accesibilidad
5	10/01/2022		Prueba de interfaz de usuario

Fuente: Elaborado a partir de [34]

### 2.4.4.1 Pruebas del sistema

La siguiente matriz permite que las pruebas de sistemas sean pertinentes al momento de ser realizadas.

Tabla 96.Pruebas de sistema

INDICADORES DE EVALUACION	
Valorización	Interpretación del resultado
1	Excelente
2	Bueno
3	Regular
4	Malo
5	Pésimo

Fuente: Elaborado a partir de [34]

#### 2.4.4.1.1 Pruebas de calidad

En las pruebas de calidad se evalúan aspectos de eficiencia, compatibilidad, seguridad y portabilidad, dando como resultado un software de calidad basado en el cumplimiento de los requisitos de usuarios.

Tabla 97. Pruebas de calidad

EVALUACION DE CALIDAD						
Características	Sub Características	1	2	3	4	5
Funcionalidad	- Pertinencia funcional - Corrección funcional - Completitud funcional	x				
Eficiencia	- Utilización de recursos - Comportamiento temporal - Capacidad		x			
Compatibilidad	- Coexistencia - Interoperabilidad		x			
Fiabilidad	- Disponibilidad - Madures - Tolerancia a fallos - Capacidad de recuperación		x			
Seguridad	- Integridad - Confidencialidad - No repudio					
Portabilidad	- Capacidad de ser instalado - Capacidad de ser reemplazado - Completitud funcional	x				

Fuente: Elaborado a partir de [34]

#### 2.4.4.1.2 Pruebas de tendencia

Toda aplicación está basada en Responsive Design, dando como valoración “1”, generando como resultado las pruebas de tendencia.

Tabla 98. Pruebas de tendencia

EVALUACION DE TENDENCIAS					
Tendencia	1	2	3	4	5
Responsive Desing	x				

Fuente: Elaborado a partir de [34]

### 2.4.4.1.3 Pruebas de usabilidad

Este tipo de prueba genera que el usuario permita moverse libremente por la aplicación, de esta manera se puede apreciar si la misma posee aspectos como aprendizaje, Operabilidad, estética y accesibilidad.

Tabla 99. Pruebas de usabilidad

EVALUACION DE USABILIDAD						
Características	Sub Características	1	2	3	4	5
Inteligibilidad	- Ejecución de procesos - Facilidad de entendimiento - Calidad de resultados	x				
Aprendizaje	- Simplicidad de opciones - Auto descripción			x		
Operabilidad	- Verificación de entradas validas - Prevención de errores	x				
Protección	- Medir errores	x				
Estética	- Interfaz agradable - Interacción con el usuario	x				
Accesibilidad	- Acciones mínimas - Acceso multi dispositivo		x			

Fuente: Elaborado a partir de [34]

### 2.4.4.1.4 Pruebas de mantenibilidad

En este tipo de prueba el software tiene la ventaja de poder cambiar, brindando eficiencia, teniendo en consideración aspectos como el modularidad, reusabilidad, es capaz de analizar y posee el hecho de ser modificado y aprobado.

Tabla 100. Pruebas de mantenibilidad

EVALUACION DE MANTENIBILIDAD						
Características	Sub Características	1	2	3	4	5
Modularidad	- Cambios con mínimo impacto.	x				
Reusabilidad	- Construcción con varios sistemas.		x			
Analizabilidad	- Análisis de los impactos producidos por los cambios efectuados dentro de la aplicación.			x		
Capacidad de ser modificado	- Modificación sin alteraciones.		x			
Capacidad de ser probado	- Facilidad de crear criterios en pruebas.		x			

Fuente: Elaborado a partir de [34]

#### 2.4.4.1.5 Pruebas de interfaz de usuario

Este tipo de prueba permiten brindar a los usuarios una mejor experiencia ya que incluye adaptabilidad, son concisas, coherente, flexible y visualmente atractivo.

Tabla 101. Pruebas de interfaz de usuario

EVALUACION DE INTERFAZ DE USUARIO					
Características	1	2	3	4	5
Adaptabilidad	x				
Legibilidad	x				
Concisión	x				
Coherencia	x				
Flexibilidad	x				
Atractivo visual	x				

Fuente: Elaborado a partir de [34]

## **2.4.5 Lanzamiento**

Las pruebas de lanzamiento se realizan al a ver concluido las de más fases, por lo que se ha tiene en consideración las posibles fallas, de esta manera se procede a tener una aplicación móvil apta para su uso, todo en base a una pre - preparación final.

### **2.4.5.1 Preparación**

La etapa de preparación se la realiza antes del lanzamiento de la aplicación móvil y es considerada una etapa de pre – lanzamiento, en donde se debe tener en cuenta el nombre y host que se utilizara, por ende, es recomendable que el nombre de la aplicación sea corto y de acuerdo a los servicios que realice el sistema además de ser único debe permitir llegar a los usuarios, para que sea intuitivo y rápido aprendizaje.

Nombre de la aplicación: “Sistema de cobranza VAGZ”.

## **2.5 EJECUCIÓN Y/O EMSABLAJE DEL PROTOTIPO**

Es importante tener bien definido los módulos en el sistema, es por eso que el presente trabajo de titulación se enfoca en el cobro de las distintas deudas en donde se focaliza la ubicación del cliente mediante una foto con ubicación GPS, para una mayor exactitud al momento de realizar el cobro.

### **2.5.1 Módulo de Inicio de sesión**

Una vez descargada la aplicación se procede a registrarse por medio del correo y una contraseña, después se accede con las mismas credenciales y si en caso de que se le olvide su contraseña tiene la facilidad de recuperación de la misma mediante el correo electrónico.

Para el inicio de sesión existen 2 tipos de usuarios un administrador y un cobrador, según lo que se procede a ejecutar:



Figura 14. Ingreso al sistema

Fuente: Elaboración propia

Ingreso de correo electrónico y contraseña, esencial para poder ingresar al sistema.



The image shows a mobile application interface for logging into a system. At the top, there is a purple status bar with the time 14:39 and various icons. Below it is a blue header with the text 'VAGZ'. The main content area has a white background with the title 'Ingresar al Sistema' in blue and 'Credenciales' in black. There are two input fields: 'Email:' with a placeholder 'Correo electrónico' and 'Contraseña:' with a placeholder 'Clave de acceso' and an eye icon. A link '¿Olvidaste la Contraseña?' is positioned below the password field. Two buttons are at the bottom: a teal 'INICIAR SESIÓN' button and a blue 'VOLVER' button with a left arrow. The bottom of the screen shows a black navigation bar with three icons.

Figura 15. Ingreso de credenciales

Fuente: Elaboración propia

## 2.5.2 Módulo de agregar cliente

Una vez ingresado como administrador se podrá registrar los n clientes, agregando la deuda; el motivo y el valor además de poner el tiempo en terminar de pagarlo.



14:47

VAGZ

### Registro de Clientes

#### Creación de Nuevo Cliente

\* Cédula:

\* Nombres:

\* Apellidos:

\* Email:

\* Celular:

\* Dirección:

Figura 16. Registro de clientes

Fuente: Elaboración propia

Una vez registrado el cliente se le podrá asignar una foto, editar sus datos y poder comunicarse por medio de WhatsApp, además de crear la deuda y poder visualizar la deuda creada así mismo como el día en que se le creó la mismas, finalmente se visualizará los datos del cliente.



Figura 17. Detalle del cliente

Fuente: Elaboración propia

### 2.5.3 Módulo de agregar cobradores

El administrador ingresara a los cobradores los mismo que tendrán un que loguearse mediante su correo y una contraseña hecha por el administrador.



The screenshot displays a mobile application interface for creating a new collector. The top status bar shows the time 14:51 and various system icons. Below the status bar is a blue header with a back arrow and the text "VAGZ". The main content area has a title "Registro de Cobradores" and a subtitle "Creación de Nuevo Cobrador". The form consists of several input fields, each with a label and a placeholder text:

- \* Cédula: Cédula de Identidad
- \* Nombres: Nombres
- \* Apellidos: Apellidos
- \* Email: Correo electrónico
- \* Celular: Celular
- \* Clave: (empty field)

The bottom of the screen shows the standard Android navigation bar with three icons: a home button, a square button, and a back arrow.

Figura 18. Registro de Cobradores

Fuente: Elaboración propia

Una vez ingresado todos los cobradores, contarán con un estado de ocupado (en caso que el administrador le asigne un cliente), cuenta con un buscador, además de enseñar con que persona está ocupado mediante un clic y si está libre le saldrá para editar los clientes.



Figura 19. Lista de cobradores

Fuente: Elaboración propia

## 2.5.4 Módulo de agregar ubicación

Desde el usuario cobrador se puede agregar la ubicación mediante una imagen que se guarda la dirección del cliente cuando es cobrado por el mismo.



Figura 20. Agregar Ubicación al cliente

Fuente: Elaboración propia

Una vez agregados los clientes se le puede añadir una ubicación exacta para así en nuevas visitas se les facilite su trabajo. Y para agregar una ubicación simplemente sigue los pasos.

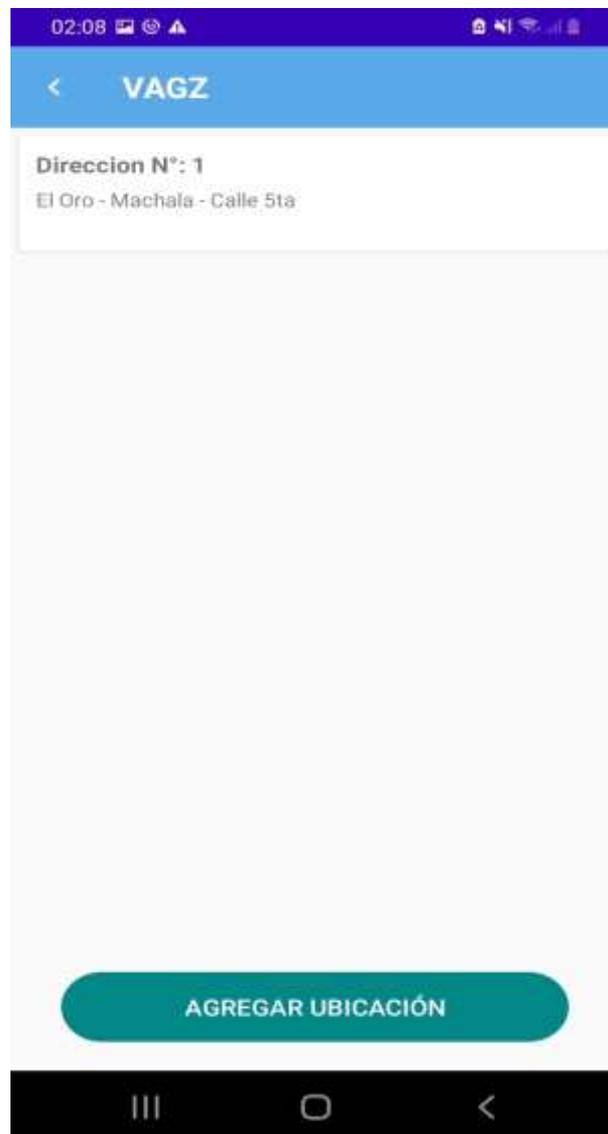


Figura 21. Visualizar Ubicación

Fuente: Elaboración propia

Cuando el cobrador decide guiarse por una ubicación que se le da, la misma que es grabada mediante una foto. Además, de poder ver hacia donde tiene que ir y hacia donde está usted.

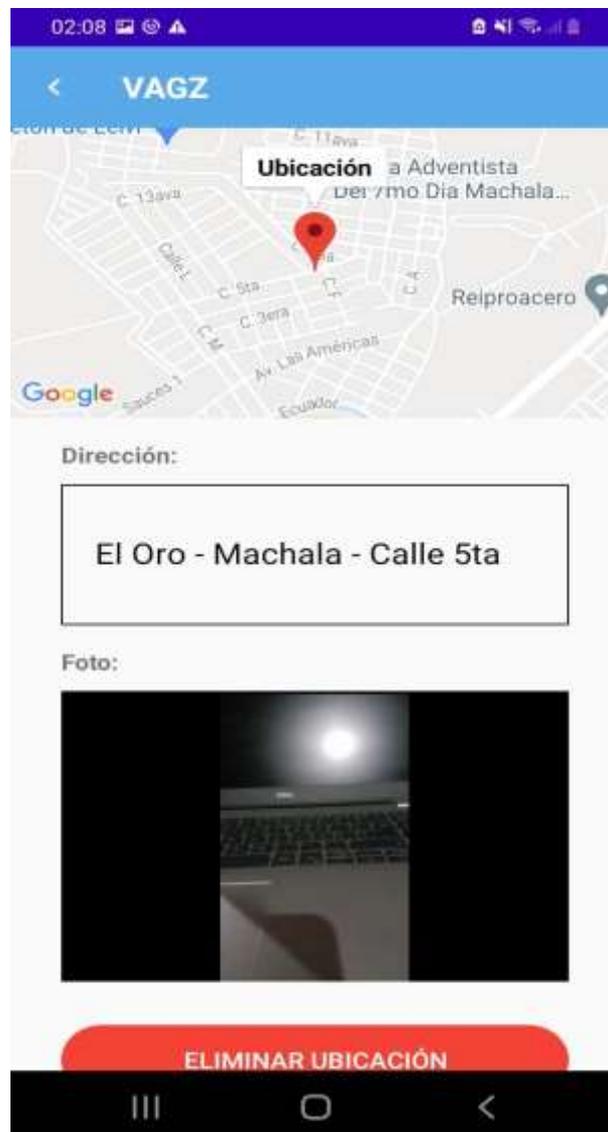


Figura 22. Visualizar ruta del cliente

Fuente: Elaboración propia

### 3. CAPITULO III. EVALUACIÓN DEL PROTOTIPO

#### 3.1 PLAN DE EVALUACIÓN

En un sistema es esencial la aplicación de métricas y estándares de calidad, para existe un correcto desenvolvimiento en la misma. La misma que se detalla con mayor precisión en el capítulo II, en la sub – fase de gestión de calidad y en la sub – fase de pruebas los resultados obtenidos.

#### 3.2 EVALUACIÓN CON HERRAMIENTAS

##### 3.2.1 Axe

“Axe” es la herramienta encargada de evaluar la accesibilidad del sistema, en base a los estándares WCAG, lo cual implica que se ejecute la aplicación mediante una excelente accesibilidad. Su funcionamiento es ejecutar la herramienta y se procese abrir la aplicación la misma y es ahí donde la herramienta empieza a dar sugerencias acerca de un cambio que se debe de dar en la aplicación.

En la primera ventana no se debe de cambiar nada, todo está correcto, eso nos da una luz verde para que el usuario pueda tener una mejor experiencia al usar la aplicación.

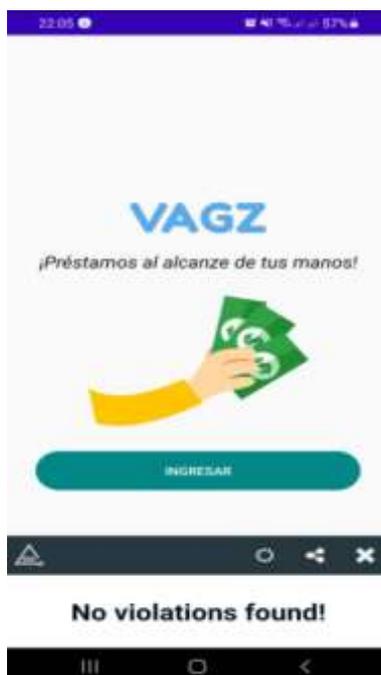


Figura 23. Resultado de la evaluación Axe

Fuente: Elaboración propia

La segunda observación nos indica que el tipo de color de letra encaja perfecto con el contraste del fondo, eso indica que la aplicación si es intuitiva.



Figura 24. Resultado de la evaluación Axe ColorConstrast

Fuente: Elaboración propia.

Aquí la observación trata de indicar que se debe de poner un label que indique para qué sirve el botón, solo que se lo hizo así debido a que es más intuitivo para el usuario.

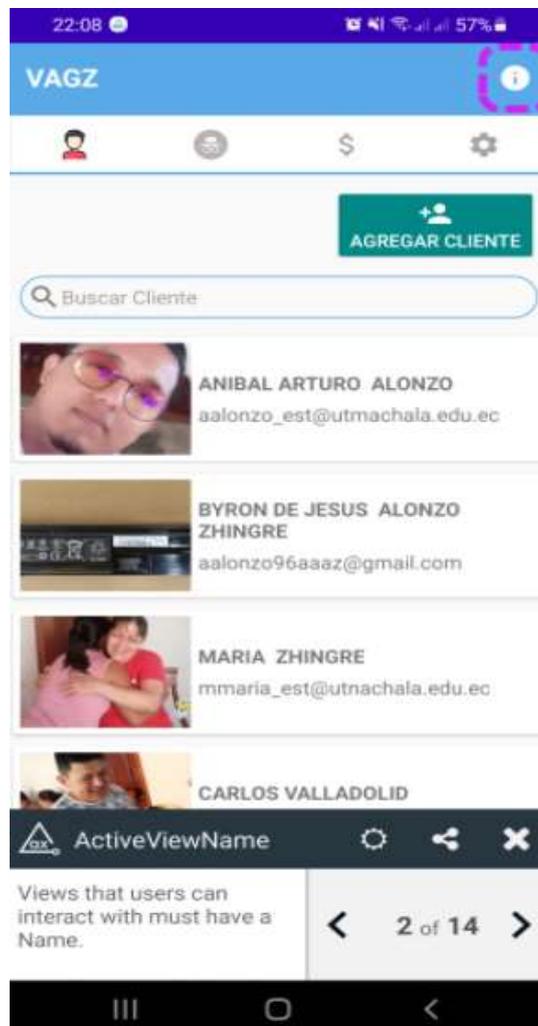


Figura 25. Resultado de la evaluación Axe ActiveViewName

Fuente: Elaboración propia

En este caso los botones están muy bien posicionados y adecuados a su funcionamiento.



Figura 26. Resultado de la evaluación Axe ActiveViewName

Fuente: Elaboración propia

En la siguiente imagen se aprecia el tamaño y diseño adecuados basado en la WCAG.



Figura 27. Resultado de la evaluación Axe ActiveViewName

Fuente: Elaboración propia

En esta ventana nos indica que los label y los botones están bien posesionados.



Figura 28. Resultado de la evaluación Axe ColorContrast

Fuente: Elaboración propia

No existe errores, todo está muy bien detallado y es muy intuitivo.



Figura 29. ¡Resultado de la evaluación Axe No violations found!

Fuente: Elaboración propia

### 3.2.2 MobSF

Para la segunda evaluación se la llevo a cabo la implementación de la herramienta MobSF, quien es la encargada de medir la seguridad que arroja la aplicación, a través de un análisis estático y a la vez se emplea un análisis dinámico. La cual se aplicó y se obtuvo un promedio del 80%, respecto al puntaje de seguridad, el cual es un muy buen porcentaje para la aplicación.

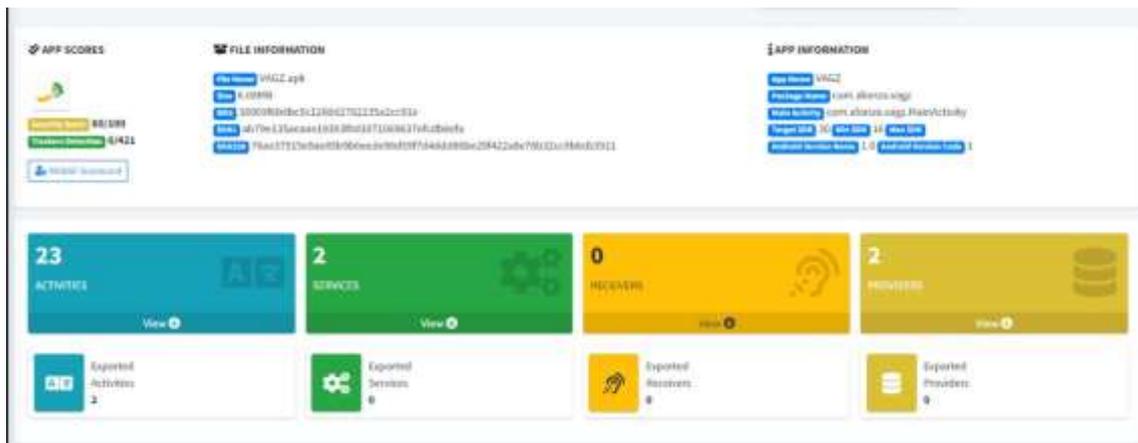


Figura 30. Evaluación con herramienta MobSF

Fuente: Elaboración propia

Además, se puede evidencia mediante los dominios a los cuales se conectan quienes son los siguientes.

DOMAIN	STATUS	GEOLOCATION
api.whatsapp.com	🟢	IP: 31.31.81.32 Country: United States of America Region: Florida City: Miami Latitude: 25.774209 Longitude: -80.193857 <a href="#">View Google Map</a>
vagg-project-default-rtzb.firebaseio.com	🟢	IP: 35.203.57.83 Country: United States of America Region: Missouri City: Kansas City Latitude: 39.099731 Longitude: -94.575558 <a href="#">View Google Map</a>
vagg-travis-349999.com	🟢	IP: 172.217.2.133 Country: United States of America Region: California City: Mountain View Latitude: 37.400991 Longitude: -122.078254 <a href="#">View Google Map</a>

Figura 31. Dominios evaluados con la herramienta MobSF

Fuente: Elaboración propia

### 3.3 CONCLUSIONES

- La metodología en este caso es MMS, la misma que fue de gran importancia para llevar a cabo la ejecución de las métricas que permite tener un correcto desarrollo de la aplicación, dando como resultado un software de calidad; debido a que cuenta con las mejores prácticas tanto de las metodologías antiguas como las nuevas, dando como resultado un seguimiento minucioso en cada etapa de esta manera se genera una situación de seguridad entre el cliente y el desarrollador.
- Las revistas indexadas, fueron de gran uso al momento de recolectar la información; generando un ambiente de confianza y seguridad en presente investigación.
- El uso de herramientas es esencial para una mejor captación de cada problema que se venga, logrando abarcar más procesos en la fase de desarrollo, como es el uso de diagramas para casos de uso o modelando acorde a los requerimientos establecidos.
- En un buen diseño es recomendable saber si la herramienta cuenta con todos los recursos necesarios para poner en marcha final los proyectos; en el presente realizo el diseño, mediante la herramienta Balsamiq Makups.
- Es necesario tener un control correcto, el cual permita el avance del proyecto, dando como elección principal las herramientas actuales y multi usos por ende se trabajó con la herramienta Android Studio y Firebase Realtime Database, las cuales permite generar un sistema con perfectos acabados y eficiente.

### 3.4 RECOMENDACIONES

- Para evitar contratiempos como errores o fracasos, es necesario la implementación de una metodología que en cada etapa o proceso pueda desarrollarse de una manera amplia y de buen control para así lograr una metodología que ejecuta inmensos proyectos.
- Realizar encuesta que evite el desperdicio de tiempo y recursos. Mediante una entrevista: lo cual impulsara la probabilidad de no cometer errores.
- Para contar con un sistema que este en constante cambios y genere buenos resultados; es necesario un correcto uso de las evaluaciones periódicas o mediante las herramientas de evaluación de sistemas.

## BIBLIOGRAFÍA

- [1] J. Alonso Arévalo y J. A. Mirón Canelo, «Aplicaciones móviles en salud: potencial, normativa de seguridad y regulación,» *Scielo*, pp. 2307-2113, 2017.
- [2] S. Chan, «Sensor Tower,» Enero 2021. [En línea]. Available: <https://sensortower.com/blog/app-revenue-and-downloads-2020>. [Último acceso: Enero 2022].
- [3] G. Association, «The Mobile Economy 2021,» *The Mobile Economy*, London, 2021.
- [4] M. Vinueza Morales y M. Cedillo Fajardo, «Gestión de cobranzas en la administración pública: una revisión sistemática de literatura,» *Repositorio Dspace*, 2019.
- [5] E. Rimarachin Torres, «Gestión de cobranza y su incidencia en la recaudación del impuesto predial en la municipalidad distrital de José Leonardo Ortiz, 2017,» *Repositorio USS*, 2018.
- [6] S. V. Guzman Jara, «Implementación de un sistema de cobranza para disminuir la morosidad de los préstamos otorgados en una Empresa Pública,» *Universidad Norbert Wiener*, Lima, 2017.
- [7] Á. Madridano, «Arquitectura de software para navegación autónoma y coordinada de enjambres de drones en labores de lucha contra incendios forestales y urbanos,» *Dianlet*, 2021.
- [8] Ó. Agudelo Varela, F. Riveros Sanabria y S. Valbuena Rodríguez, «Evaluación de una Arquitectura de Software,» *Dialnet*, vol. XIX, nº 2, 2021.
- [9] J. Moyano, K. Cenci y J. Ardenghi, «Arquitectura Cliente-Servidor de Alto Rendimiento para servicio RTK,» de *XXVI Congreso Argentino de Ciencias de la Computación*, Buenos Aires, Red de Universidades con Carreras en Informática, 2020, pp. 470-479.
- [10] D. O. Alulema Flores, «Una Metodología basada en Modelos y Servicios para la Integración de Sistemas IoT,» *Dialnet*, p. 266, 2021.
- [11] Z. Mamani Rodríguez, L. Del Pino Rodríguez y J. Gonzales Suarez, «Arquitectura basada en Microservicios y DevOps para una Ingeniería de software continua,» *Redalyc*, 2020.
- [12] E. A. Espina Suárez y G. E. Gómez Hormaza, «Mitigación de riesgos a través del uso de una arquitectura de ciberseguridad mediante modelamiento de amenazas en la implementación de sistemas de

información basados en internet de las cosas,» Universidad Peruana de Ciencias Aplicadas, Lima, 2021.

- [13] C. Rojas, G. Chanchi y K. Villalba, «Propuesta de una Arquitectura IoT para el control domótico e inmótico de edificaciones,» *Revista Ibérica de Sistemas e Tecnologías de Informacao*, pp. 28-40, 2019.
- [14] [https://www.java.com/es/about/whatis\\_java.jsp](https://www.java.com/es/about/whatis_java.jsp), «Java,» [En línea]. Available: [https://www.java.com/es/about/whatis\\_java.jsp](https://www.java.com/es/about/whatis_java.jsp). [Último acceso: Enero 2021].
- [15] X. Meng, K. Qian, D. Io, H. Shahriar, M. A. Islam Talukder y P. Bhattacharya, «Desarrollo de software IPC móvil seguro con detectores de vulnerabilidades en Android Studio,» *IEEE*, 2018.
- [16] N. Verma, S. Kansal y M. Huned, «Desarrollo de Aplicación Móvil Nativa Usando Android Studio para Cabs y un vistazo a las aplicaciones multiplataforma,» *Revista Internacional de Investigación en Ingeniería Aplicada*, vol. XIII, nº 16, pp. 12527-12530, 2018.
- [17] «Developers,» 12 Noviembre 2021. [En línea]. Available: <https://developer.android.com/studio/features?hl=es-419>. [Último acceso: Enero 2022].
- [18] L. F. Litano Ramos, «Desarrollo de una aplicación móvil utilizando Flutter y Firebase para realizar el seguimiento de los tratamientos farmacológicos de un paciente,» Universidad Nacional de Piura, Piura, 2021.
- [19] R. M. Toasa Guachi, «Evaluación de rendimiento entre el estándar de mensajería MQTT y la plataforma Firebase a través de un prototipo, modelo de comunicación IoT,» Universidad Tecnológica Israel, Quito, 2020.
- [20] Firebase, «Firebase,» [En línea]. Available: [https://firebase.google.com/?hl=es-419&gclid=Cj0KCCQiAuP-OBhDqARIsAD4XHpc7aYEdKx0l1NddRiKQoH1kkPhER2mvWsD2aKq9mND\\_GsigccVu6vwaAineEALw\\_wcB&gclsrc=aw.ds](https://firebase.google.com/?hl=es-419&gclid=Cj0KCCQiAuP-OBhDqARIsAD4XHpc7aYEdKx0l1NddRiKQoH1kkPhER2mvWsD2aKq9mND_GsigccVu6vwaAineEALw_wcB&gclsrc=aw.ds).
- [21] Firebase, «Firebase,» [En línea]. Available: <https://firebase.google.com/products/realtime-database?hl=es-419>.
- [22] Firebase, «Firebase,» [En línea]. Available: <https://firebase.google.com/products/auth?hl=es-419>.
- [23] A. Castilla Martínez, L. Carbonel Riquett, Y. Ramos Villegas, L. Quintana Pájaro y L. Moscote Salazar, «Aplicaciones móviles en las neurociencias: un nuevo aliado,» *Revista Chilena de Neurocirugía*, vol. LXV, pp. 136-139, 2019.

- [24] J. Llerena Izquierdo, M. Robalino Alfonso, M. Andina Zambrano y J. Grijalva Segovia, «Aplicación móvil para fortalecer el aprendizaje de ajedrez en estudiantes de escuela utilizando realidad aumentada y m-learning,» *Revista Ibérica de Sistemas y Tecnologías de la Información*, nº 22, pp. 120-133, 2019.
- [25] J. E. Herrera Rubio, K. Y. Sánchez Mojica y E. A. López Jaramillo, «Estudio del modelo de capas de IoT para enlaces descendentes en plataforma de interconexión de la red Sifgox,» *Revista Logos Ciencia & Tecnología-Scielo*, vol. XIII, nº 3, 2021.
- [26] J. Medina Cruz, E. Pineda Ballesteros y F. R. Téllez Acuña, «Requerimientos de software: prototipado, software heredado y análisis de documentos,» *Revista de Ingeniería - Redalyc*, 2019.
- [27] «Balsamiq,» [En línea]. Available: <https://balsamiq.com/>.
- [28] R. A. Salas Rueda y J. d. J. Vázquez Estupiñan, «Aplicación en la nube Lucidchart: ¿Herramienta necesaria para la innovación del proceso educativo en el siglo XXI,» *Revista de Comunicación de la SEECI*, 2018.
- [29] J. Durán Cazar, E. Tandazo Gaona, M. Morales Morales y S. Morales Cardoso, «Rendimiento de bases de datos columnares,» *Ingenius. Revista de Ciencia y Tecnología - Scielo*, nº 22, 2019.
- [30] J. P. Zumbra Gamboa y C. A. León Arreaga, «Evolución de las Metodologías y Modelos utilizados en el Desarrollo de Software,» *INNOVA - Dialnet*, vol. III, nº 10, pp. 20-33, 2018.
- [31] «La metodología genexus de gonda y la tradicional en el aprendizaje del desarrollo de software sobre base de datos en los estudiantes del IV ciclo de la especialidad de Informática de la Universidad Nacional de Educación Enrique Guzmán y Valle,» *Horizonte de la Ciencia - Redalyc*, 2020.
- [32] S. Martín Gómez, «Aplicación de las Metodologías Ágiles al proceso de enseñanza aprendizaje universitario,» *RIDU - Dialnet*, vol. XII, nº 12, pp. 62-73, 2020.
- [33] M. A. Santos Romero y N. E. Escudero López, «Propuesta de Metodología Híbrida y Base de Documentación para el Desarrollo de Software Actual,» *Conciencia Tecnológica*, nº 60, 2020.
- [34] J. R. Molina Ríos, M. P. Zea Ordóñez, F. F. Redrován Castillo, M. R. Valarezo Pardo, J. A. Honores Tapia, R. F. Morocho Román, J. L. Armijos Carrión y O. E. Cárdenas Villavicencio, "MMS" Metodología para el diseño y desarrollo de aplicaciones móviles, Machala: Área de Innovación y Desarrollo, S.L., 2021.

- [35] V. Espinola Verdín y L. A. Torres González, «Análisis cualitativo de modelos de negocio para el emprendimiento social,» *Entreciencias: Diálogos en la Sociedad del Conocimiento, Scielo*, vol. VIII, nº 22, 2020.
- [36] R. Ceballos, D. Borrego, M. Gómez y R. Gasca, «Análisis de decisión multicriterio para el diagnóstico de no conformidad: una estrategia basada en prioridades que combina datos y reglas comerciales,» *Elsevier*, vol. 183, 2021.
- [37] F. B. Armijos Mayón, A. I. Bermúdez Burgos y N. V. Mora Sánchez, «Gestión de administración de los Recursos Humanos,» *Universidad y Sociedad*, vol. XI, nº 04, 2019.
- [38] R. I. Ramírez Molina, G. A. Royero Orozco y O. N. EL Kadi Janbeih, «Gestión tecnológica como factor clave de éxito en universidades privadas,» *TeloS*, 2018.
- [39] M. Á. Aires Borrás, G. J. Martins Lourencao y R. Coser Mergulhao, «Interacción Universidad-Empresa para la enseñanza de Ingeniería basada en casos reales,» *Scielo*, 2021.
- [40] V. Bolshakova, A. Guerrero y G. Halin, «Identificar los roles de las partes interesadas y los documentos de proyecto relevantes para la toma de decisiones colaborativa basada en 4D,» *Fronteras de la gestión de la ingeniería- SpringerLink*, vol. VII, pp. 104-118, 2020.
- [41] G. Parra Quero, O. Palma Urdaneta, M. E. Torres Samuel y F. Durán Garrido, «Caracterización de buenas prácticas en la elicitación de requisitos de software referidas en el estándar ISO/IEC/IEEE 29148,» *Publicaciones en Ciencias y Tecnologías-Dialnet*, vol. XIV, nº 2, pp. 91-99, 2020.
- [42] J. M. Cruz Montero, H. E. Guevara Gómez y J. P. Flores Arocutipa, «Áreas de conocimiento y fases clave en la gestión de proyectos: consideraciones técnicas,» *Revista Venezolana de Gerencia*.
- [43] R. Carvajal, «Aproximación a la comprensión del gerenciamiento de proyectos públicos,» *EnContexto- Dialnet*, 2020.
- [44] S. Bustamante García, M. Á. Valles Coral y I. E. Cuéllar Rodríguez, «Políticas basadas en la ISO 27001:2013 y su influencia en la gestión de seguridad de la información en municipalidades de Perú,» *Revista Enfoque- Redalyc*, 2021.
- [45] J. J. Espinoza Zúñiga, «Aplicación de metodología CRISP-DM para segmentación geográfica de una base de datos pública,» *Ingeniería, Investigación y Tecnología -Redalyc*, vol. XXI, nº 1, 2019.

## ANEXOS

### ANEXO I: CASO DE USO DE LA APLICACIÓN MÓVIL REGISTRO E INICIO DE SESIÓN

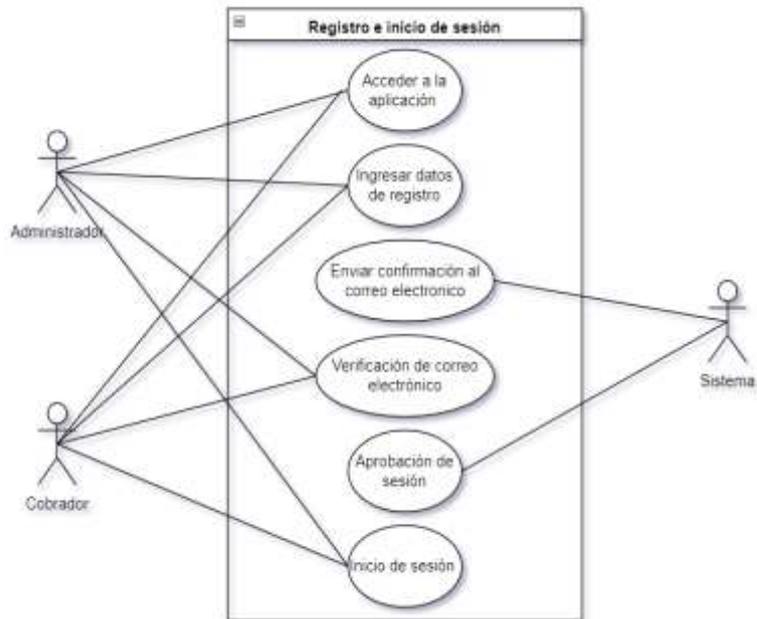


Figura 32. Anexo I: Caso de uso de registro e inicio de sesión

Fuente: Elaboración propia

### MENÚ ADMINISTRADOR

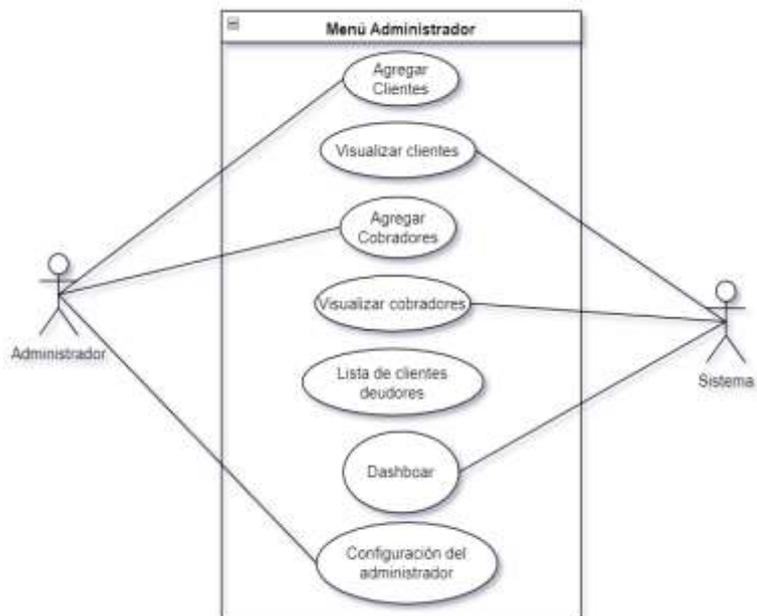


Figura 33. Anexo I: Caso de uso menú del administrador

Fuente: Elaboración propia

## MENÚ COBRADOR

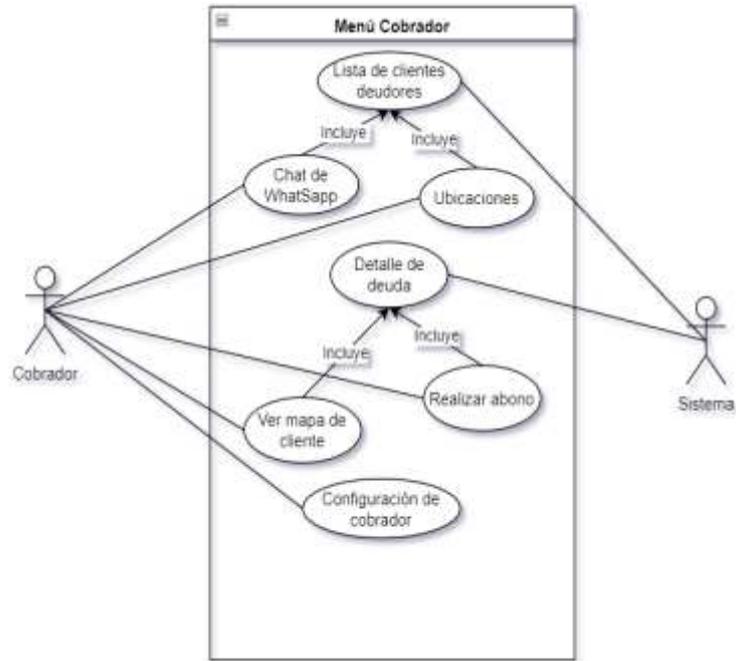


Figura 34. Anexo I: Caso de uso menú del cobrador

Fuente: Elaboración propia