

**UNIVERSIDAD NACIONAL DE INGENIERIA
DIRECCIÓN DE ÁREA DE CONOCIMIENTO DE
TECNOLOGIA DE LA INFORMACIÓN Y COMUNICACIÓN**

**TRABAJO MONOGRÁFICO PARA OPTAR AL TÍTULO DE INGENIERO EN
COMPUTACIÓN**

Tema:

**DESARROLLO DEL SISTEMA WEB GERENCIAL Y APLICACIÓN MÓVIL PARA
GESTIÓN DE CONTROL DE CRÉDITO Y RECUPERACIÓN DE CARTERA PARA LA
PYME PRESTA FLASH**

Autores:

Br. Liesel Carolina Cruz Contreras.

Br. Marvin Antonio Gómez Torrez.

Br. Tareq Zamir Moreno Delgado.

Tutor:

Msc. Marvin David Castañeda.

Febrero del 2024
Managua, Nicaragua

Agradecimientos

Agradecemos la vida, y la salud necesaria que nos permitió nuestros años de educación superior y este momento, la culminación de nuestros esfuerzos.

A nuestras familias que nos acompañaron y alentaron, creyendo en nosotros y que con amor fueron nuestro aliento cuando nuestras fuerzas parecían decaer.

A nuestro tutor Marvin Castañeda por ser nuestra guía y por su arduo esfuerzo y desmedida dedicación, sin la cual este proyecto no hubiese sido de éxito.

A nuestra Alma Mater, y sus docentes que nos formaron con integridad y sabiduría. Somos fruto de su abnegada enseñanza.

Dedicatoria

El presente Trabajo Monográfico lo dedicamos especialmente a:

Mi amado esposo Josué, quien hace parte fundamental de los logros en mi vida. A mis padres, Neftali y Celnia quienes construyeron con su esfuerzo un futuro para mí. Finalmente, a mis amigos y compañeros de universidad que me acompañaron en el transcurso de mi formación profesional.

Br. Liesel Cruz Contreras

Quiero agradece a mis hermanos, a mi padre y a mi difunta madre, a mis amigos que me brindaron su incondicional apoyo para completar esta etapa de mi vida.

Br. Tareq Moreno Delgado

¡Pues el SEÑOR concede sabiduría!

De su boca provienen el saber y el entendimiento. Proverbios 2:6

Agradezco a Dios porque él es quien da la sabiduría.

Agradezco a mis padres y mis hermanos, Gracias a mis amigos y sobre todo a los que siempre están. Agradezco a mi esposa Ana, mi hija Zoé y mi hijo Zaid; ellos son el impulso para seguir andando.

Br. Marvin Gómez Torrez

Resumen

El presente trabajo monográfico constituye el análisis y desarrollo del sistema web gerencial y aplicación móvil para gestión de control de crédito y recuperación de cartera que será implementada en la pyme presta flash.

Este documento presenta a detalle las herramientas utilizadas para la recopilación de requerimientos del sistema, vinculados a los procesos esenciales de la pyme PrestaFlash. Los cuales nos permitieron lograr nuestros objetivos planteados en el desarrollo de este proyecto.

Para el desarrollo de la aplicación móvil y el sistema web se utilizó la metodología RAD, la cual nos permitió un desarrollo rápido mediante entrega de prototipos, los cuales fueron presentando se a la parte interesada, para finalmente completar un producto de software finalizado.

Se desarrolló una aplicación móvil android haciendo uso de una base de datos no relacional FireBase y sistema web gerencial en html y javascript.

Se diseñó y ejecuto un plan de pruebas para la detección de errores, con el propósito de evitar fallas de sistema y lograr el óptimo funcionamiento.

Fueron elaborados manuales de usuario para el sistema web y la aplicación móvil, con el propósito de capacitar adecuadamente a los usuarios finales.

INDICE

I. Introducción	11
II. Antecedentes	12
III. Objetivo General	13
General.....	13
Específicos	13
IV. Justificación	14
V. Marco Teórico	15
5.1 Aplicación Móvil.....	15
5.2 Android	15
5.3 Plataforma de Desarrollo Java.....	15
5.4 Lenguaje JAVA.....	16
5.5 Android SDK.....	16
5.6 Sistema Web	16
5.7 Framework Web	17
5.8 FireBase	18
5.8.1 Base de Datos en Tiempo Real:	18
5.8.2 Trabajando sin Conexión en Aplicaciones Móviles con Firebase	20
5.9 Base de Datos no Relacional.....	22
5.10 Rapid Application Development (RAD).....	22
5.11 Pruebas Unitarias.....	23
5.12 DevTools (development tools)	24
5.13 Junit	25
5.14 Herramientas CASE	25
5.15 UML (Unified Modeling Language)	26
VI. Diseño Metodológico	27
6.1 Metodología de Investigación	27
6.2 Desarrollo de objetivos.....	27
6.3 Metodología de desarrollo.....	28
VII. Factibilidad del desarrollo	30
7.1 Factibilidad técnica.....	30
7.2 Factibilidad económica.....	32
7.2.1 Costos de hardware.	32
7.2.2 Costos de software.....	32
7.2.3 Costos de desarrollo.....	33
7.2.4 Costos totales.....	33

7.3	Factibilidad operativa	34
CAPITULO I.....		35
VIII. Fase de Iniciación.....		36
8.1	Requerimientos Funcionales	38
8.2	Analizar los requerimientos y alcances	40
8.3	Gestión de requerimientos	46
8.4	Requerimientos No Funcionales.....	48
8.5	Requerimientos de seguridad	49
8.6	Modelado de Requerimientos	49
8.7	Modelado del negocio	49
8.8	Casos de Uso del Sistema.....	50
8.8.1	Caso de Uso gestionar Usuario	51
8.8.2	Caso de Uso Gestión de Cliente.....	54
8.8.3	Caso de Uso Gestión de Préstamo.....	56
8.8.4	Caso de uso gestionar Cartera	58
8.8.5	Caso de uso gestionar Plan de Negocio.....	61
8.8.6	Caso de uso Gestionar Dispositivo	63
8.8.7	Caso de Uso Generar Reporte.....	65
8.9	Diagrama de Secuencia.....	67
8.9.1	Diagrama de secuencia crear usuario	67
8.9.2	Diagrama de secuencia Agregar Préstamo	68
8.9.3	Diagrama de secuencia Agregar Cliente	68
8.10	Diagrama de Estados.....	69
8.10.1	Diagrama de estado cliente.....	69
8.10.2	Diagrama de estado usuario	69
8.10.3	Diagrama de estado préstamo.....	69
IX. Fase de Construcción		70
CAPITULO II.....		75
X. Fase de Implementación		76
10.1	Modelo de Desarrollo RAD	76
10.2	Prototipos Aplicación Móvil	76
10.2.1	Prototipo 1.....	76
10.2.2	Prototipo 2.....	76
10.2.3	Prototipo 3.....	77
10.2.4	Prototipo 4 y Final.....	77
10.3	Prototipos sistema web	77

10.3.1	Prototipo 1.....	77
10.3.2	Prototipo 2.....	78
10.3.3	Prototipo 3.....	78
10.3.4	Prototipo 4 y Final.....	78
10.4	Plan de Pruebas.....	79
10.4.1	Pruebas unitarias sistema Web.....	79
10.4.2	Pruebas Unitarias Aplicación Móvil.....	82
10.4.3	Conclusiones plan de pruebas.....	85
XI.	Conclusiones.....	86
XII.	Recomendaciones.....	87
XIII.	Bibliografía.....	88
XIV.	Anexos.....	90

INDICE DE TABLAS

Tabla 1	Hardware de la pyme PrestaFlash.....	30
Tabla 2	Requisitos mínimo de hardware.....	31
Tabla 3	Requisitos Óptimos de hardware.....	31
Tabla 4	Costos de hardware.....	32
Tabla 5	Costos de software.....	33
Tabla 6	Costos de desarrollo.....	33
Tabla 7	Costos Totales del Proyecto.....	33
Tabla 8	Porcentaje de Intereses corriente según plazo.....	38
Tabla 9	Entrevista al Cliente.....	42
Tabla 10	Análisis FODA.....	43
Tabla 11	Análisis de Involucrados.....	43
Tabla 12	Tabla de Actores.....	46
Tabla 13	Modelo de Calidad.....	47
Tabla 14	Caso de Uso Gestionar Usuario.....	53
Tabla 15	Caso de Uso Gestionar Cliente.....	56
Tabla 16	Caso de Uso Gestionar Préstamo.....	58
Tabla 17	Caso de uso gestionar Cartera.....	60
Tabla 18	Caso de Uso Gestionar Plan de Financiamiento.....	63
Tabla 19	Caso de uso Gestionar Dispositivo.....	65
Tabla 20	Caso de Uso Generar Reportes.....	66

INDICE DE ILUSTRACIONES

Ilustración 1 Modelo de Desarrollo Rápido de Aplicaciones	23
Ilustración 2 Estructura organizacional de la pyme Presta Flash.	37
Ilustración 3 Formula de Amortización de préstamos. Cuota Nivelada	38
Ilustración 4 Árbol de problemas.	44
Ilustración 5 Árbol de Objetivos	45
Ilustración 6 Diagrama de Caso de Uso del Negocio.....	50
Ilustración 7 Diagrama de Caso de Uso del Sistema.....	50
Ilustración 8 Diagrama de Caso de Uso Gestión de Usuario.....	51
Ilustración 9 Diagrama de Caso de Uso Gestión de Cliente.....	54
Ilustración 10 Diagrama de Caso de Uso Gestión de Préstamos.....	56
Ilustración 11 Caso de uso gestionar Cartera	58
Ilustración 12 Caso de uso Gestionar Plan de Financiamiento	61
Ilustración 13 Caso de Uso Gestionar Dispositivo	63
Ilustración 14 Diagrama de Caso de Uso Generar Reportes.....	65
Ilustración 15 Diagrama de secuencia crear usuario	67
Ilustración 16 Diagrama de secuencia crear Préstamo	68
Ilustración 17 Diagrama de secuencia crear Cliente.....	68
Ilustración 18 Diagrama de estado de cliente	69
Ilustración 19 Diagrama de estado de usuario	69
Ilustración 20 Diagrama de estado de préstamo.....	69
Ilustración 21 Panel de administración de FIREBASE.....	71
Ilustración 22 Codificación de aplicación móvil Presta Flash.....	71
Ilustración 23 Codificación de sistema web gerencial PrestaFlash	72
Ilustración 24 Estructura Código Fuente Aplicación Móvil Presta Flash	72
Ilustración 25 Estructura del Sistema Web.....	73
Ilustración 26 Capa de Datos aplicación Móvil.....	73
Ilustración 27 Capa de presentación de aplicación móvil	74
Ilustración 28 Resultado prueba unitaria Reporte Recuperaciones.....	80
Ilustración 29 Resultado prueba unitaria Reporte Colocación	80
Ilustración 30 Resultado prueba unitaria Dispositivos Móviles	81
Ilustración 31 Resultado prueba Unitarias Usuarios	81
Ilustración 32 Dependencia de Junit agregada al gradle	82
Ilustración 33 Creación de Prueba Unitaria Generar Amortización	82
Ilustración 34 Prueba Unitaria Generar Amortización.....	83
Ilustración 35 Prueba Unitaria Generar fechas de pago.	83

Ilustración 36 Resultado de Prueba Unitaria Generara Fechas de Pago con días de Gracia	83
Ilustración 37 Prueba Unitaria Generara Fechas de Pago con 0 días de Gracia...	84
Ilustración 38 Resultado de Prueba Unitaria Generara Fechas de Pago con 0 días de Gracia	84

I. Introducción

Presta flash es una micro Pyme que facilita créditos a pequeños y medianos empresarios apoyando el crecimiento de sus negocios, fomentan el emprendedurismo y potencian su liquidez.

El otorgamiento de dichos créditos es gestionado (tramitado) por un gestor de colocación y cobranza. Quien registra a cada solicitante manualmente en tarjetas otorgadas de forma individual a cada mediano empresario por préstamo. Los préstamos otorgados pertenecen a una de las diferentes carteras de cobro.

La cartera de préstamos está distribuida según la ubicación geográfica del negocio del solicitante. En la actualidad son 4 carteras gestionadas cada una por un gestor de colocación y cobranza, quien visita a los clientes diariamente y registra sus pagos en las tarjetas asignadas.

El objetivo de este proyecto es brindar una solución informática para la gestión de los préstamos. La cual requiere ser asistido mediante una aplicación móvil en la cual se registrará los préstamos y sus pagos realizados por agentes de colocación y cobranza que deberán contar con un dispositivo móvil Android.

La herramienta software consistirá en un sistema integrado por una aplicación móvil para la gestión y cobranza de préstamos y un sistema web gerencial para realizar consultas relacionadas a control de cartera, registrar usuarios, registrar dispositivos y realizar copias de seguridad.

II. Antecedentes

La PYME presta flash desde sus inicios ha llevado registros manuales de sus préstamos y pagos realizados. Para lograr gestionar el estado de su cartera debe replicar estos registros manuales en un documento de Excel (ver anexo 3) el cual se actualiza periódicamente, esto permite conocer el saldo de colocaciones y el saldo recuperado.

El proceso de cobro dentro de todos los procesos que realiza la pyme es el de mayor interés para el proyecto. Los ejecutores de él son los gestores de cobro (o cobradores) los cuales visitan diariamente a los clientes con créditos activos o vencidos con el propósito de recolectar la cuota.

El registro de la cuota se realiza mediante tarjetas manuales (ver anexo 2) asignadas de forma individual a cada cliente por préstamo que lleva el gestor de cobro al momento de visitar al cliente. Este a su vez cuenta con una tarjeta propia la cual es equivalente a la del gestor.

Las tarjetas contienen el nombre del cliente, la fecha de entrega del crédito y se representa individualmente las cuotas para el préstamo en una cuadrícula numerada, la cual se va tachando de acuerdo al cumplimiento de la cuota por el cliente. El gestor marca únicamente la casilla correspondiente al número de cuota abonada si fue pagada por completo. Ya que se aceptan pagos parciales. Los gestores de cobro tienen asignados un grupo de clientes que usualmente habitan en una zona delimitada a la cual visita el gestor. Son responsables del cobro de los créditos, también tienen la autoridad de facilitar un crédito nuevo a un cliente que lo solicita al momento de cancelar su crédito.

El cliente recibe una nueva tarjeta con los datos del nuevo crédito solicitado, y se asigna como fecha de pago de la primera cuota el día siguiente del que recibe el monto del crédito.

Al terminar cada semana se arquea lo recolectado por cada gestor de cobro y se actualiza el documento de Excel con los saldos colocados y los saldos recuperados.

El gestor de cobro recibe un 3% como comisión de todo lo recuperado en el mes. Esto se agrega al salario básico de los gestores correspondiente al mínimo de la actividad económica a la que pertenecen: C\$9,346.59 córdobas nicaragüenses.

El pago a los gestores de cobro y al resto del personal de la pyme se realiza manualmente en un archivo de Excel donde se registra los salarios brutos por cada empleado, sus retenciones y comisiones mensuales.

III. Objetivo General

General

Desarrollar un sistema web gerencial y aplicación móvil para gestión de control de crédito y recuperación de cartera para la pyme presta flash.

Específicos

1. Analizar los requerimientos y alcances necesarios para la realización del sistema web y la aplicación móvil para la pyme Presta Flash.
2. Implementar el modelo de desarrollo RAD a partir de los elementos encontrados en los requerimientos, usando metodología ágil y auxiliándonos de herramientas del lenguaje UML.
3. Codificar los diferentes elementos del modelo seleccionado haciendo uso del lenguaje de programación JAVA Android SDK en la aplicación móvil y JavaScript, HTML y Bootstrap en el sistema web.
4. Elaborar y ejecutar un plan de pruebas unitarias para el sistema web y la aplicación móvil.
5. Implantar el software del modelo realizado, en los equipos de la pyme Presta Flash.

IV. Justificación

La aplicación propuesta a desarrollar en este proyecto es más eficiente ya que los pagos diarios de los créditos otorgados serán registrados digitalmente evitando las omisiones de pagos y facilitando conocer los saldos sin tener que esperar el registro manual al final del día. El proceso actual de gestión de cobro de la cartera global de préstamos no permite tener saldos actualizados diariamente de los préstamos debido al proceso tardío y/o incompleto de registrar en el archivo Excel de control de cada sub cartera los pagos recibidos, lo cual debe ser registrado al final del día por cada gestor de la pyme Presta Flash.

El módulo web será capaz de realizar consultas de colocaciones y recuperaciones de la cartera, facilitando la información requerida para evitar problemas de rentabilidad derivados de la cartera morosa ya que actualmente no existe un registro unificada de las 4 sub carteras con las que cuenta la PYME, lo cual dificulta conocer con rapidez y eficacia los saldos de la cartera global para tomar decisiones sobre morosidad y las acciones requeridas para disminuir los índices y prevenir los vencimientos de créditos activos. El módulo web también registrara usuarios, dispositivos y realizara copias de seguridad.

Otro aspecto de relevancia en la actual gestión de préstamos de la PYME es la reiterada incongruencia de saldos que existe entre las tarjetas de pagos del gestor y la de los clientes. Evitar dicha problemática reafirma la credibilidad de la PYME ante sus clientes, fomentando la renovación de créditos.

V. Marco Teórico

5.1 Aplicación Móvil

Una aplicación móvil es una aplicación informática diseñada para ser ejecutada en teléfonos inteligentes, tabletas y otros dispositivos móviles y que permite al usuario efectuar una tarea concreta de cualquier tipo facilitando las gestiones o actividades a desarrollar.

En esencia, una aplicación no deja de ser un software, podemos decir que las aplicaciones móviles son para los móviles lo que los programas son para ordenadores de escritorio. (J. Cuello, J. Venotte)

Por lo general, se encuentran disponibles a través de plataformas de distribución, operadas por las compañías propietarias de los sistemas operativos Móviles como Android, iOS, BlackBerry OS, Windows Phone, entre otros. Existen aplicaciones móviles gratuitas u otras de pago, donde en promedio el 20 a 30 % del coste de la aplicación se destina al distribuidor y el resto es para el desarrollador.

Dada la necesidad de la Presta Flash de poder usar dispositivos de fácil acceso y económicos disponibles en el mercado, se pretende desarrollar la Aplicación móvil para dispositivos Android, bajo el lenguaje JAVA utilizando el SDK de Android.

5.2 Android

Es un sistema operativo basado en el núcleo Linux. Fue diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos. Inicialmente fue desarrollado por Android Inc., empresa que Google respaldó económicamente y más tarde.

Como se hace referencia anteriormente se utilizará dispositivos que utilicen este sistema operativo debido a las necesidades de la empresa y a la disponibilidad de las herramientas de desarrollo gratuitas.

5.3 Plataforma de Desarrollo Java

Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo.

Lo que se conoce como plataforma Java, antes Plataforma Java 2, abarca una serie de tecnologías basada en varios conceptos. (Dominguez,J. 2019)

- Un lenguaje de programación para codificar aplicaciones.
- Un conjunto de librerías que facilitan el trabajo a los programadores.

Las principales API de java se dividen en 3 grandes bloques:

- a. Java SE: Edición Estándar (Standard Edition), antes J2SE. Dirigida a aplicaciones de escritorio, applets, ...
 - b. Java EE: Edición Empresa (Enterprise Edition), antes J2EE. Enfocada en dispositivos de consumo, con recursos limitados.
 - c. Java ME: Edición Micro (Micro Edtition), anes J2ME. Util para aplicaciones empresariales (Web, Distribuidas)
 - d. Java ME: Edición Micro (Micro Edtition)
- Unas herramientas (JDK) para generar las aplicaciones a partir del código fuente.

5.4 Lenguaje JAVA

Es un lenguaje de programación de propósito general, concurrente, orientado a objetos que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo, lo que quiere decir que el código que es ejecutado en una plataforma no tiene que ser re compilado para correr en otra.

5.5 Android SDK

SDK responde a las siglas Software Development Kit, lo que viene a ser un kit de desarrollo de software. Con él podremos desarrollar aplicaciones y ejecutar un emulador del sistema Android de la versión que sea. Todas las aplicaciones Android se desarrollan en lenguaje Java con este kit.

5.6 Sistema Web

Se denomina aplicación web o sistema web a aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de internet o de una intranet mediante un navegador.

En otras palabras, es un programa que se codifica en un lenguaje interpretable por los navegadores web en la que se confía la ejecución al navegador.

Las aplicaciones web son populares debido a lo práctico del navegador web como cliente ligero, a la independencia del sistema operativo, así como a la facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales.

Para el caso de presta flash se optó por un sistema web debido a su capacidad para correr bajo casi cualquier hardware y la amplia disponibilidad de frameworks de desarrollo web que soportan bases de datos no relacionales como son Firebase.

5.6.1 Independencia del Sistema Móvil y Registro de Dispositivos

La arquitectura del sistema desarrollado se ha concebido con el objetivo fundamental de garantizar la independencia del sistema operativo Android en dispositivos móviles. Esta característica estratégica permite que la aplicación web asociada al sistema pueda desempeñar un papel crucial en situaciones donde los dispositivos móviles pueden experimentar fallas o daños.

En caso de que un dispositivo móvil registrado con la aplicación sufra algún inconveniente, la aplicación web proporciona una alternativa eficaz para el registro y seguimiento de dispositivos. Esto significa que los usuarios pueden acceder y utilizar la funcionalidad completa del sistema al registrar nuevos dispositivos.

Esta flexibilidad en el acceso al sistema ofrece una capa adicional de continuidad y respaldo, asegurando que las operaciones críticas de gestión de crédito y recuperación de cartera no se vean obstaculizadas por eventos imprevistos en los dispositivos móviles. Además, facilita el proceso de incorporación de nuevos dispositivos a la red, simplificando la experiencia de los usuarios y asegurando la continuidad operativa del sistema.

La capacidad de operar sin depender totalmente del sistema operativo Android refuerza la robustez y versatilidad del sistema, marcando un avance significativo en la adaptabilidad a escenarios del mundo real donde la infraestructura móvil puede ser sujeta a variaciones inesperadas.

5.7 Framework Web

El concepto framework se emplea en muchos ámbitos del desarrollo de sistemas software, no solo en el ámbito de aplicaciones Web. Podemos encontrar frameworks para el desarrollo de aplicaciones médicas, de visión por

computador, para el desarrollo de juegos, y para cualquier ámbito que pueda ocurrir se nos. En general, con el término framework, nos estamos refiriendo a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. (J. Gutierrez, pág. 1)

Bootstrap es un framework o conjunto de herramientas de Código abierto para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como, extensiones de JavaScript opcionales adicionales.

5.8 FireBase

Firebase es una plataforma de desarrollo de aplicaciones móviles y web, adquirida por Google en 2014, que ofrece una variedad de servicios y herramientas para facilitar la creación, el despliegue y el crecimiento de aplicaciones. Algunas de las características principales de Firebase incluyen:

5.8.1 Base de Datos en Tiempo Real:

Firebase ofrece una base de datos NoSQL en tiempo real que sincroniza datos en tiempo real entre los clientes conectados. Es particularmente útil para aplicaciones que requieren actualizaciones instantáneas y colaboración en tiempo real.

A. Autenticación:

Proporciona servicios de autenticación segura que permiten a los desarrolladores autenticar a los usuarios mediante direcciones de correo electrónico, contraseñas, proveedores de identidad social como Google, Facebook y más.

B. Almacenamiento en la Nube:

Firebase Storage permite almacenar y recuperar archivos, como imágenes o videos, de manera segura y eficiente en la nube.

C. Hosting Web:

Ofrece hosting web que permite a los desarrolladores implementar y alojar fácilmente aplicaciones web estáticas y dinámicas directamente desde la infraestructura de Firebase.

D. Cloud Functions:

Permite ejecutar código en respuesta a eventos en Firebase y las aplicaciones de Google Cloud. Es útil para manejar eventos como la creación de usuarios, actualizaciones en la base de datos, entre otros.

E. Cloud Firestore:

Es una base de datos NoSQL más avanzada que ofrece escalabilidad y consultas más complejas. Puede utilizarse como alternativa a la base de datos en tiempo real.

F. Cloud Messaging:

Facilita el envío de mensajes a dispositivos móviles y navegadores para mantener a los usuarios comprometidos con la aplicación.

G. Analytics:

Proporciona análisis detallados sobre cómo los usuarios interactúan con la aplicación, lo que ayuda a los desarrolladores a comprender mejor el rendimiento y la experiencia del usuario.

H. Test Lab para Android:

Ofrece pruebas automáticas en dispositivos reales y virtuales en la infraestructura de Google, ayudando a garantizar la calidad de la aplicación.

I. Dynamic Links:

Permite crear enlaces que llevan a los usuarios a cualquier ubicación de la aplicación, independientemente de si la aplicación está instalada.

J. Machine Learning:

Integra servicios de aprendizaje automático de Google Cloud para facilitar la implementación de capacidades de aprendizaje automático en las aplicaciones.

Firebase proporciona una solución integral para el desarrollo y la administración de aplicaciones, reduciendo la carga operativa y permitiendo a los desarrolladores concentrarse más en la creación de características únicas para sus aplicaciones. La plataforma es especialmente popular entre los desarrolladores de aplicaciones móviles y startups debido a su facilidad de uso y su conjunto completo de servicios.

Firebase nos proporciona un servicio de base de datos con la particularidad de ser en tiempo real. Esto significa que cualquier cambio realizado en los datos por cualquier cliente (usuario, aplicación, dispositivo...) se sincronizará automáticamente y de forma inmediata (siempre que la conexión lo permita) en el resto de clientes, sin necesidad de que éstos vuelvan a consultar los datos.

5.8.2 Trabajando sin Conexión en Aplicaciones Móviles con Firebase

En el mundo actual, donde la conectividad a Internet no siempre es constante, la capacidad de trabajar sin conexión se ha convertido en un aspecto crucial para muchas aplicaciones móviles. Firebase, la plataforma de desarrollo de aplicaciones móviles y web de Google, ofrece una solución robusta para abordar este desafío. En este artículo, exploraremos la característica de trabajar sin conexión en aplicaciones móviles mediante Firebase.

¿Qué significa trabajar sin conexión?

Trabajar sin conexión se refiere a la capacidad de una aplicación para funcionar y realizar operaciones incluso cuando el dispositivo no está conectado a Internet. En el contexto de Firebase, esta funcionalidad permite a las aplicaciones móviles almacenar y sincronizar datos localmente, proporcionando una experiencia de usuario fluida y continua.

Ventajas de Trabajar sin Conexión:

Experiencia del Usuario Continua: Los usuarios pueden continuar utilizando la aplicación incluso en entornos con conectividad intermitente o nula.

A. Mayor Confianza

La capacidad de realizar operaciones sin conexión aumenta la confianza del usuario en la aplicación, ya que no depende totalmente de la disponibilidad de Internet.

B. Eficiencia en el Uso de Datos

Al almacenar datos localmente, se reduce la necesidad de descargar la misma información repetidamente, lo que ahorra ancho de banda y mejora la eficiencia en el uso de datos.

C. Cómo Firebase Facilita el Trabajo sin Conexión

Firebase ofrece varias herramientas y servicios que permiten a los desarrolladores implementar la funcionalidad de trabajar sin conexión de manera eficiente. Algunos de estos componentes clave incluyen:

D. Firestore Database

La base de datos en tiempo real de Firebase permite a las aplicaciones almacenar y sincronizar datos localmente, manteniendo la coherencia con la nube cuando se restablece la conexión.

E. Cloud Firestore

Esta base de datos NoSQL en la nube de Firebase facilita el almacenamiento y recuperación eficientes de datos, incluso en modo sin conexión.

F. Firebase Authentication

La capacidad de autenticación sin conexión permite a los usuarios acceder a partes específicas de la aplicación incluso cuando no están conectados.

G. Firebase Hosting

Permite la entrega de recursos estáticos incluso sin conexión, mejorando la experiencia del usuario al garantizar la disponibilidad de activos clave.

La capacidad de trabajar sin conexión en aplicaciones móviles es esencial para brindar experiencias de usuario fluidas y confiables. Con Firebase, los desarrolladores pueden aprovechar herramientas poderosas para implementar esta funcionalidad de manera efectiva, asegurando que sus aplicaciones sigan siendo útiles y eficientes, independientemente de las condiciones de conectividad. Al integrar estas capacidades, Firebase contribuye significativamente a la construcción de aplicaciones móviles modernas y resilientes.

5.9 Base de Datos no Relacional

Las bases de datos no relacionales son un sistema de almacenamiento de datos en la cual no encontramos el típico esquema entidad-relación al que estamos acostumbrados. Al contrario de la estructura de tablas que usan las bases de datos relacionales utilizan formatos como clave-valor (Json), mapeo de columnas o grafos.

El tipo de formato de almacenamiento Json utiliza una clave única para cada registro por lo cual se pueden realizar búsquedas clave-valor. Resultan ser el formato de Base de datos no relaciones más versátiles.

5.10 Rapid Application Development (RAD)

El desarrollo rápido de aplicaciones (RAD) es un proceso de desarrollo de software, desarrollado inicialmente por James Martin en 1980. El término fue usado originalmente para describir dicha metodología. Más recientemente, el término y su acrónimo se están usando en un sentido genérico, más amplio, que abarca una variedad de técnicas dirigidas al desarrollo de aplicaciones rápidas.

La metodología de desarrollo rápido de aplicaciones se creó para responder a la necesidad de entregar sistemas muy rápido. El enfoque de RAD no es apropiado para todos los proyectos. El alcance, el tamaño y las circunstancias, todo ello determina el éxito de un enfoque RAD. El método RAD tiene una lista de tareas y una estructura de desglose de trabajo diseñada para la rapidez. El método comprende el desarrollo iterativo, la construcción de prototipos y el uso de herramientas CASE (Ingeniería de Software Asistida por Computadora).

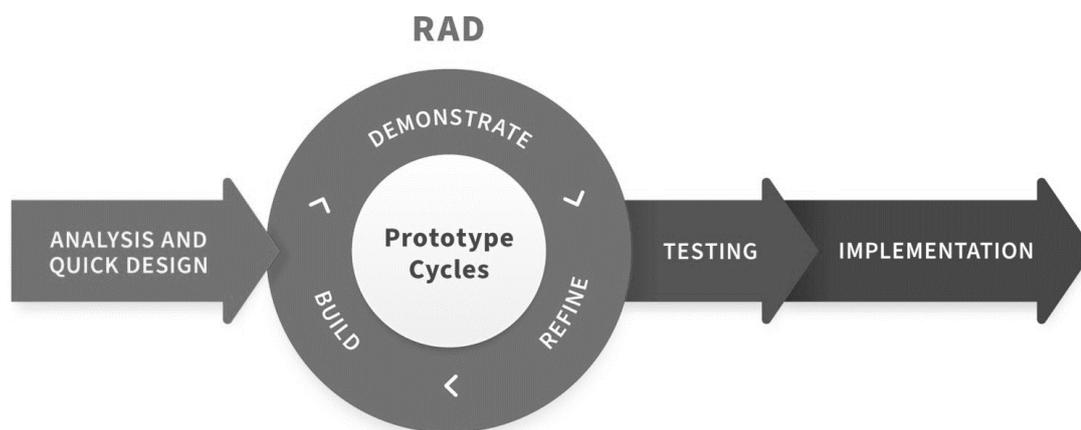


Ilustración 1 Modelo de Desarrollo Rápido de Aplicaciones
Fuente: Rapid Application Development (RAD) - 2021 Guía Completa

Ventajas del desarrollo rápido de aplicaciones:

- Velocidad de desarrollo.
- Ciclos de desarrollo más cortos.
- Visibilidad temprana debido al uso de técnicas de prototipos.
- La calidad del sistema: Se mide en términos de hasta qué punto, al momento de la implementación del sistema este cumple con los+ requerimientos.

La metodología de desarrollo rápido propone proceso de desarrollo que permite que se creen sistema en utilizables en periodos cortos de entre 60 y 90 días.

5.11 Pruebas Unitarias

Las pruebas unitarias consisten en aislar una parte del código y comprobar que funciona a la perfección. Son pequeños test que validan el comportamiento de un objeto y la lógica. Las pruebas unitarias demuestran que la lógica del código está en buen estado y que funcionará en todos los casos.

Aumentan la legibilidad del código y ayudan a los desarrolladores a entender el código base, lo que facilita hacer cambios más rápidamente.

Los test unitarios bien realizados sirven como documentación del proyecto.

Se realizan en pocos milisegundos, por lo que podrás realizar cientos de ellas en muy poco tiempo.

5.12 DevTools (development tools)

DevTools se refiere a las herramientas de desarrollo integradas en los navegadores web, diseñadas para proporcionar un entorno interactivo y eficiente para el desarrollo y la depuración de aplicaciones web. Estas herramientas ofrecen diversas funciones que abarcan desde la inspección y manipulación del DOM hasta la depuración de código JavaScript y el análisis de rendimiento.

Funciones Clave de DevTools

1. Inspector de Elementos:

Permite examinar y modificar la estructura del DOM de una página web en tiempo real.

Facilita la identificación de elementos HTML, estilos CSS y atributos asociados.

2. Consola:

Proporciona una interfaz interactiva para ejecutar comandos JavaScript directamente.

Muestra mensajes de error, advertencias y mensajes de registro.

3. Depurador:

Permite establecer puntos de interrupción en el código JavaScript.

Facilita la inspección de variables, la ejecución paso a paso y la identificación de errores.

4. Red:

Ofrece visibilidad sobre las solicitudes HTTP realizadas por la página.

Facilita el análisis de rendimiento y la identificación de problemas de red.

5. Rendimiento:

Proporciona herramientas para analizar y mejorar el rendimiento de la aplicación.

Permite la grabación y el análisis detallado del rendimiento durante la carga y ejecución.

6. Application:

Ofrece información sobre el almacenamiento local, cookies y otros aspectos de la aplicación.

Facilita la gestión de datos almacenados en el navegador.

7. Auditoría:

Permite realizar auditorías automáticas para evaluar el rendimiento, la accesibilidad y las mejores prácticas.

Proporciona informes detallados con sugerencias de mejora.

Uso Durante el Desarrollo

8. Depuración de Código:

DevTools se utiliza para identificar y corregir errores en el código JavaScript y otros recursos.

9. Inspección de Elementos:

Facilita la inspección y ajuste de la estructura del DOM, ayudando en problemas visuales y de diseño.

10. Análisis de Rendimiento:

Permite la identificación de cuellos de botella y la optimización del rendimiento de la aplicación.

11. Uso Durante las Pruebas

- Pruebas Interactivas:

DevTools permite la ejecución de comandos JavaScript en tiempo real, facilitando las pruebas interactivas de funciones y el estado de la aplicación.

- Depuración en Tiempo Real:

Ayuda en la identificación y resolución de problemas durante las pruebas, mejorando la eficiencia del proceso de desarrollo.

5.13 Junit

Es un framework Java que permite la escritura y ejecución de pruebas repetitivas, con el propósito de confirmar la correcta y óptima funcionalidad de los métodos. Es una instancia de la arquitectura xUnit para frameworks de pruebas unitarias.

5.14 Herramientas CASE

Las herramientas de productividad, conocidas como herramientas de Ingeniería de Software Asistida por Computadora (CASE), se crearon de manera explícita para mejorar el trabajo rutinario a través del uso del soporte

automatizado. Los analistas emplean herramientas CASE para aumentar la productividad, comunicarse con los usuarios de una manera más efectiva e integrar el trabajo que realizan en el sistema, desde el inicio hasta el fin del ciclo de vida.

5.15 UML (Unified Modeling Language)

El lenguaje Unificado de Modelo (UML) es un lenguaje de modelo visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software (j. Rumbaugh, 2000). Se utiliza para entender, diseñar, hojear, configurar, mantener y controlar la información sobre el sistema que se desea construir.

El UML está compuesto por diversos elementos gráficos que se combinan para conformar diagramas, de entre ellos los siguientes:

- Diagrama de flujos.
- Diagrama de contexto.
- Diagrama de flujo de datos.
- Diagrama de casos de uso.
- Diagrama de secuencia.
- Diagrama de actividades.
- Diagrama de despliegue.

VI. Diseño Metodológico

6.1 Metodología de Investigación

Esta propuesta de trabajo monográfico, se realizará bajo el enfoque de una investigación mixta (cualitativa y cuantitativa), que se basa en el levantamiento de los requerimientos mediante las herramientas de recolección de datos, posteriormente se realizará un análisis de los datos para generar la solución. Se debe comprobar hasta qué punto se han cumplido los objetivos y en qué medida se ha resuelto el problema.

6.2 Desarrollo de objetivos

1. Analizar los requerimientos y alcances necesarios para la realización de la aplicación móvil y el módulo web para la pyme Presta Flash.

La identificación de los requerimientos se realizará mediante herramientas de recolección de datos (entrevistas, cuestionarios, etc.). Se representarán los procesos mediante diagramas de flujo.

El análisis de los requerimientos nos dará como resultados las características necesarias para que el sistema cumpla con las necesidades de los usuarios.

2. Implementar el modelo de desarrollo RAD a partir de los elementos encontrados en los requerimientos, usando metodología ágil y auxiliándonos de herramientas del lenguaje UML.

En esta etapa se utilizan los requerimientos para realizar el diseño lógico de la aplicación móvil y del módulo web. Se creará una representación o modelo de software detallando aspectos como la arquitectura del sistema, la estructura de los datos e interfaces del sistema

3. Codificar los diferentes elementos del modelo seleccionado haciendo uso del lenguaje de programación JAVA Android SDK en la aplicación móvil y JavaScript, HTML y Bootstrap en el módulo web.

Se centra en la generación del código del sistema basándonos en el diseño previamente establecido. Se debe llevar una revisión iterativa del código de acuerdo al modelo de desarrollo de software RAD. Esta fase incluye la elaboración de manuales técnicos y de usuario.

4. Elaborar y ejecutar un plan de pruebas unitarias para el sistema web y la aplicación móvil.

Las pruebas unitarias son una forma de comprobar que un fragmento de código funciona correctamente. Estas pruebas se realizan en todo el ciclo de vida del desarrollo del sistema web y la aplicación móvil.

5. Implementar el sistema desarrollado en los equipos de la PYME Presta Flash para su implementación.

Esta fase no ocurre una sola vez sino iterativamente por cada prototipo presentado. La implementación es también incrementable.

Por cada iteración se pone a la disposición del usuario el sistema para retroalimentación, es decir la detección de errores.

6.3 Metodología de desarrollo

El desarrollo rápido de aplicaciones (RAD) es una metodología orientada a objetos para el desarrollo de aplicaciones, se centra en desarrollar aplicaciones rápidamente por medio de iteraciones frecuentes y aprobaciones con comentarios continuos de los usuarios. RAD prioriza los lanzamientos de prototipos ágiles y rápidos.

A continuación, se presentan las fases RAD:

- 1. Planificación de los requisitos:** Se definen los requisitos con la ayuda de usuario que posean vastos conocimientos de los procesos de la empresa. En esta etapa usaremos algunas herramientas de recolección de datos (cuestionarios, entrevistas, etc.) para conocer los procesos que realiza la pyme para cada área de trabajo. Luego identificaremos las necesidades para definir las funciones que la aplicación móvil y el sistema web, serán capaces de realizar, con esto cumpliremos el primer objetivo del proyecto.
- 2. Diseño:** Se analiza detalladamente de las actividades de la empresa en relación al sistema propuesto. En esta fase realizaremos un diseño de una arquitectura que cumpla con los requerimientos identificados

utilizando herramientas UML y diseñando una base de datos que almacene la información requerida. Se diseñan prototipos cada vez más adecuados a la funcionalidad que se requiere. El usuario responde a los prototipos funcionales reales y se refinan los módulos diseñados en base a los requerimientos.

- 3. Construcción:** Se desarrollan los prototipos diseñados en la fase 2. Esta es la fase de construcción rápida de prototipos, la participación es interactiva. El usuario responde a los prototipos funcionales reales y se refinan los módulos desarrollados en base a los requerimientos.
- 4. Implementación:** Una vez que se concluyen los ciclos de prueba, se verifica que la Aplicación Móvil y el sistema web cumplen las especificaciones de análisis y diseño, entonces, se realiza la aceptación por parte del usuario y es puesta en producción.

VII. Factibilidad del desarrollo

En consideración de los recursos de cómputos (hardware y software) con los que cuenta la PYME Prestaflash se muestran los siguientes aspectos, teniendo en cuenta que es un sistema web estos recursos son suficiente para su funcionamiento debido a que se está ejecutando desde un navegador mediante una url en la cual se encuentra alojada la aplicación web permitiendo que sea multiplataforma.

En cuanto a la aplicación móvil serán necesarios dispositivos móviles en la plataforma móvil Android como versión mínima de ejecución Ice Cream Sandwich (Android 4.0.3 – 4.0.4).

7.1 Factibilidad técnica

Hardware y software de la phyme

Cantidad	Dispositivo	Descripción
4	Laptop	Procesador Intel Core I5, 256GB SSD, 8GB RAM.
1	Computadora de escritorio	Procesador Intel Core i5, 8GB RAM, 500GB.
1	Conexión a Internet	8 megas.

Tabla 1 Hardware de la pyme PrestaFlash
Fuente: Elaboración propia

Cabe mencionar que estos son los recursos con los que cuenta la pyme, siendo óptimos, en cuanto a capacidad, para la puesta en producción del proyecto sobrepasando los recursos mínimos para el software. Se requieren adquirir más dispositivos celulares con las mismas características.

Hardware y software mínimos para el proyecto

Cantidad	Dispositivo	Descripción
1	Computadora de escritorio	Procesador Intel Core i5, 8GB RAM, 500GB.
6	Telefono Celular Samsung	Procesador Exynos 7880, 3 GB RAM, Android 7.0
6	Impresora Térmica	Impresora térmica de 58mm
6	Datos Móviles	7 GB
1	Conexión a Internet	8 megas
3	Laptop	Procesador Intel Core I5, 256GB SSD, 8GB RAM.

Tabla 2 Requisitos mínimo de hardware
Fuente: Elaboración propia

En esta tabla se muestran los requisitos mínimos para el uso del software, actualmente la pyme cuenta con recursos superiores y óptimos para el uso del proyecto. Se requieren adquirir 6 dispositivos celulares con las características mínimas especificadas.

Hardware y software óptimos

Cantidad	Dispositivo	Descripción
1	Computadora de escritorio	Procesador Intel Core i5, 16GB RAM, 500GB.
6	Telefono Celular Samsung	Procesador Exynos 7880, 3 GB RAM, Android 7.0
6	Impresora Térmica	Impresora térmica de 58mm
6	Datos Móviles	7 GB
1	Conexión a Internet	8 megas
3	Laptop	Procesador Intel Core I5, 256GB SSD, 32GB RAM.

Tabla 3 Requisitos Óptimos de hardware
Fuente: Elaboración propia

En la tabla se muestran los requisitos óptimos para el uso del sistema.

7.2 Factibilidad económica.

En este punto, se realizó un desglose los gastos incurridos en el proyecto como tal, estos gastos serán clasificados como:

- a. Costos de hardware.
- b. Costos de software.
- c. Costos de desarrollo.
- d. Costos totales.

También se ha realizado las investigaciones correspondientes para determinar cuáles serán los costos más factibles y viables para hacer posible el desarrollo del Sistema web y aplicación móvil para la pyme PrestaFlash.

Se provee una lista detallada de estos gastos. (Ver **Tabla 4 - Costos de hardware.**, **Tabla 5 – Costos de software.**, **Tabla 6 - Costos de desarrollo.** y **Tabla 7 - Costos totales.**)

7.2.1 Costos de hardware.

Estimación de costos.

Cantidad	Descripción	Precio Unitario	Total
6	Telefono Celular Samsung A5. Procesador Exynos 7880, 3 GB RAM, Android 7.0	\$150	\$900
6	Impresora térmica de 58mm	\$60	\$360
6	Datos Móviles ,7 GB	\$10 (mensuales)	\$60
3	Laptops. Procesador Intel Core I5, 256GB SSD, 32GB RAM.	\$310	\$930
Total			\$2,250

Tabla 4 Costos de hardware
Fuente: Elaboración propia

Las computadoras para el desarrollo no representaron ningún costo ya que la pyme cuenta con los recursos. El monto de datos móviles es mensual.

7.2.2 Costos de software.

Cantidad	Descripción	Precio Unitario	Total
1	Android SDK	\$0	\$0
1	FireBase (Realtime DataBase, Storage, Hosting)	\$0	\$0
1	Junit	\$0	\$0

1	HTML	\$0	\$0
1	CSS	\$0	\$0
1	Boostrap	\$0	\$0
1	Hosting	\$0	\$0
Totales			\$0

Tabla 5 Costos de software

Fuente: Elaboración propia

7.2.3 Costos de desarrollo.

Cantidad	Descripción	Salario Mensual	Total	Cantidad
2	Desarrollador Móvil Senior	\$1,000	5 1/2	\$5,500
1	Desarrollador Web Senior	\$500	5 1/2	\$2,750
Total				\$8,250

Tabla 6 Costos de desarrollo

Fuente: Elaboración propia

7.2.4 Costos totales.

Descripción	Total (Ya se cuenta con los recursos)	Total (adquiriendo los recursos)
Costos de hardware	\$1,320	\$1,320
Costos de software	\$0	\$0
Costos de desarrollo	\$8,250	\$8,250
Total	\$9,570	\$9,570

Tabla 7 Costos Totales del Proyecto

Fuente: Elaboración propia

La inversión total en el proyecto es de \$9,570 este costo el cual involucra el total de los costos de software y desarrollo se lo está ahorrando la pyme ya que los softwares de desarrollo son de uso gratuito, así también el alojamiento del mismo. Siendo el único costo que debe asumir la pyme, la compra de los dispositivos móviles, impresoras térmicas y los datos móviles que usaran para la implementación de la aplicación móvil. Cabe mencionar también que los equipos y recursos con los que cuenta la pyme (laptops) son óptimos para la puesta en marcha del proyecto y el correcto funcionamiento del mismo.

7.3 Factibilidad operativa.

Para el desarrollo de este apartado hay que tomar en cuenta que el uso de sistemas que automatizan procesos suelen ser un reto intimidante para personas que no estén familiarizadas con las computadoras y el software, en consecuencia, se tomara en cuenta que la interfaz de usuario sea lo más sencilla y amigable posible, de fácil uso, que permita una ágil navegación y no sea complicado en ningún aspecto, siempre enfocados en la comodidad del usuario, la capacidad de generar reportes según la necesidad del usuario y la seguridad que este percibe al conceder permisos según los distintos roles que sean asignados, todo esto se lograra usando diagramas UML como base en el desarrollo.

La implementación de este sistema web y la aplicación móvil le dará la facilidad al usuario de pretaflash de guardar de manera más organizada y sencilla grandes cantidades de información, así como el acceso a esta información será hará de forma más segura y rápida, se tendrá un control aun mayor y efectivo de la información de la pyme y la información de los créditos, con una estructura interna de los usuarios mejor organizada y controlando el acceso a la información para cada uno de ellos.

Según los descrito anteriormente, denotamos que el sistema es viable a nivel operativo, debido a que, con la implementación de este, se solucionan varios problemas, como el manejo de los procesos que abarca el sistema, generando mayor agilidad en los distintos procesos anteriormente descritos a los cuales el usuario tendrá un control mejor optimizado y organizado, evitando tiempos de espera innecesarios.

CAPITULO I

Fase de Desarrollo

VIII. Fase de Iniciación

Es el punto de partida del proyecto, donde se define sus objetivos, alcance, viabilidad y recursos necesarios. Es fundamental para establecer una base sólida y garantizar que los objetivos del proyecto estén alineados con los objetivos de la micro PYME.

1. Análisis del Entorno

Presta Flash es una pyme centrada en la colocación de créditos y la gestión de cobranza de los mismos, en sus diversas carteras. La buena gestión de recuperación de los créditos otorgados es un proceso fundamental para el éxito de la pyme.

2. Recopilación de la Información

- Fuentes Primarias

Para conocer el entorno actual de la pyme acudiremos con el personal de PRESTA FLASH vinculado a los procesos operativos de funcionamiento del negocio. Para el desarrollo del sistema se realizaron consultas a libros y trabajos monográficos relacionados al tema.

- Fuentes Secundaria

Se consultaron informes, artículos y libros encontrados en internet sobre los procesos de amortización de créditos y gestión de cobranza de cartera para las consultas técnicas al personal de Presta Flash.

3. Estructura Organizacional

Para efectos de estudio se muestra la estructura organizacional de la pyme Presta Flash.

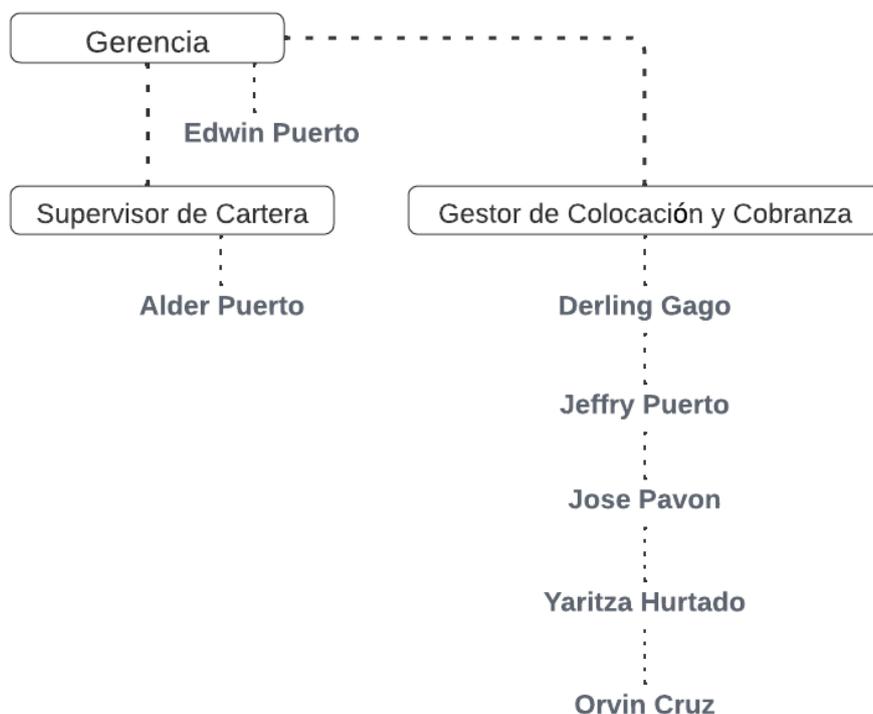


Ilustración 2 Estructura organizacional de la pyme Presta Flash.

Fuente: Elaboración Propia

4. Misión y Visión

- Misión

Contribuir al desarrollo económico de familias nicaragüenses proporcionando créditos de fácil acceso que apoyen a pequeños emprendedores.

- Visión

Ser una pyme de financiamiento comprometida con sus clientes y reconocida por su transparencia y excelente servicio, eficiente y en constante crecimiento e innovación.

8.1 Requerimientos Funcionales

Sistema Móvil

- *Gestores de colocación y cobranza*

1. Registro de cliente: se requerirá nombre completo, cédula de Identidad, dirección (domiciliaria obligatoria, del negocio y de cobro) y número de matrícula de negocio.
2. Registro plan de Préstamo: plazo en días para vencimiento del crédito, porcentaje de interés según plazo, días de gracia (solo para clientes nuevos), monto o cuota de pago.
3. Iniciar Préstamo: Seleccionar un cliente, establecer fecha de inicio, monto solicitado, monto aprobado, Asignar plan de préstamo o crear uno personalizado.
4. Definiciones de préstamos:

- a. Vencimiento de Cuota: Diario.

Los días Domingos no son hábiles en la calendarización de la tarjeta de pago, por tanto, no debe ser tomado en cuenta para la calendarización de pagos, ni el cálculo de días vencidos.

- b. Tipo de Plazo en Cuotas:

- 24
- 30
- 40

- c. Tipo de Interés: Corriente, definido según plazos.

Tipo de Plazo (Cuotas en días)	Interés
24	20%
30	20%
40	32%

Tabla 8 Porcentaje de Intereses corriente según plazo.

Fuente: Elaboración propia

- d. Tipo de Amortización: Cuota Nivelada

$$Cuota = \frac{(Monto Aprobado * Porcentaje de Interes)}{Plazo en días para vencimiento del crédito} + Monto Aprobado$$

Ilustración 3 Formula de Amortización de préstamos. Cuota Nivelada

Fuente: Matemáticas Financieras. Guillermo Pastor.

- e. Refinanciamiento de deuda para préstamos vencidos: El saldo vencido se amortizará de nuevo con el mismo porcentaje de interés a un plazo mayor o igual al convenido en el crédito original.

5. Registro de pagos: Se permiten pagos parciales o totales.

- **Datos necesarios:** cliente, monto pagado, fecha de pago, número de cuota, gestor de cobro, forma de pago.
- **Definición de formas de pago:** Efectivo, Cancelación por garantía
 - **Efectivo:** Pago en moneda local o extranjera al cambio diario definido por la administración al comienzo del día.
 - **Cancelación por garantía:** Hacer efectiva la garantía como método de pago para cancelar la deuda.
- Gerente y/o Administrador:
 1. Registrar un Dispositivo mediante Código de Dispositivo.
 2. Agregar un nuevo usuario: Nombre de usuario, contraseña, Número de cédula, nombre completo, Rol de usuario, cartera.

Definición de roles: Gestor, Supervisor de Cartera y Administrador

- **Gestor:** Tendrá acceso a todas las funcionalidades relacionadas con la gestión de clientes y préstamos.
 - **Supervisor:** Tendrá acceso a las funcionalidades
 - **Administrador:** Tendrá acceso a todas las funcionalidades de un gestor y la administración de los gestores y generación de reportes.
3. Consultar saldos vigentes y vencidos de los clientes.
 4. Asignación de cartera de clientes a un gestor de cobro determinado.

5. Resumen de estados de cuentas en tres niveles: por cliente, por gestor y en general.
6. Generación de reportes.

Definición de reportes

- Cartera Detalle
- Cartera consolidada
- Clientes por fecha de Vencimiento
- Resumen por rango de fechas.

Sistema Web

1. Registrar un Dispositivo mediante el código del dispositivo.
2. Agregar un nuevo Usuario: Nombre de usuario, contraseña, Número de cédula, nombre completo, Rol de usuario, cartera.

Definición de roles: Gestor de Colocación y Cobro (Gestor), y Gerente (Admin)

- Gestor: Tendrá acceso a todas las funcionalidades relacionadas con la gestión de clientes y préstamos.
- Admin: Tendrá acceso a todas las funcionalidades de un gestor y la administración de los gestores y generación de reportes.

3. Consultar Colocaciones de la cartera.
4. Consultar Recuperaciones de la cartera.
5. Realizar copias de seguridad locales.
6. Realizar copias de seguridad en la nube.

8.2 Analizar los requerimientos y alcances

En este apartado se analizarán los requerimientos y alcances necesarios para la realización de la aplicación móvil y el sistema web para la pyme Presta Flash.

- 1. Para iniciar el levantamiento de los requerimientos se realizó un Análisis FODA.**

El análisis FODA se usa para identificar las fortalezas, oportunidades, debilidades y amenazas de un negocio o proyecto determinado. De esa forma,

pueden establecerse estrategias y acciones efectivas para el desarrollo del proyecto.

También realizamos entrevistas.

Parte I: Identificando perfil de usuario/cliente	
Nombre del entrevistado	
Rol en organización	
Responsabilidades del entrevistado	
Parte II: Evaluación del problema	
¿Qué problemas debe solucionar el Sistema?	
(Para cada problema) Cómo se resuelven ahora	
(Para cada problema) Cómo los resolvería	
Parte III: Entendiendo el Entorno	
¿Quiénes serán los usuarios del Sistema?	
¿Cuál será su nivel de control de PC, dispositivos?	
Tienen los usuarios experiencia en este tipo de aplicaciones	
Hay aplicaciones/sistemas relevantes a tener en cuenta en nuestro Sistema	
Cuáles son las expectativas de usabilidad del Producto	
¿Cuáles son las expectativas de formación a usuarios para el uso del Sistema?	
¿Qué tipo de ayuda requerirá el usuario (ayuda online...)?	
Parte IV: Evaluando la oportunidad	
¿Quién en la organización necesita la aplicación?	
¿Cuántos tipos de usuarios usarán la aplicación?	
¿Cómo valoraría que la solución ha sido un éxito?	
Parte V: Evaluando la solución	
Resumen de las capacidades clave de la solución propuesta:	
¿Añadiría alguna?	
¿En qué posición de un ranking pondría cada una de estas capacidades?	
Parte VI: Evaluando las necesidades de fiabilidad, rendimiento y soporte	
¿Cuáles son tus expectativas en cuanto a fiabilidad del sistema?	
¿Cuáles son tus expectativas en cuanto a rendimiento del sistema?	

¿Dará soporte el ayuntamiento al producto? ¿O lo hará otro equipo?	
¿Existen necesidades especiales para dar el soporte?	
¿Qué pasa con el mantenimiento y acceso a los servicios?	
¿Cuáles son los requisitos de seguridad?	
¿Cuáles son los requisitos de instalación y configuración?	
¿Existen requisitos especiales sobre licenciamiento?	
¿Cómo se distribuirá el software?	
¿Existen requisitos de empaquetado?	
Parte VII: Otros requisitos	
¿Existen requisitos legales que deban soportarse?	
¿Existen requisitos de entorno?	
¿Existen otro tipo de requisitos?	
Parte VIII: Recapitulación	
Después de la entrevista y mientras los datos están frescos haz un resumen de las principales necesidades y problemas identificados por el usuario:	
(lista de temas tratados)	
¿Esto representa lo que debe hacer el Sistema?	

Tabla 9 Entrevista al Cliente.
Fuente: Elaboración Propia

2. Identificación FODA

Con el análisis FODA se identificaron los siguientes factores de influencia para la PYME.

Fortalezas	Debilidades
<ul style="list-style-type: none"> - Factor humano - Personal experimentado en el campo de cobranza. 	<ul style="list-style-type: none"> - Procesos lentos de actualización de saldos de carteras - Información descentralizada - Uso básico de las herramientas informáticas. - No posee herramientas para la gestión de cartera.
Oportunidades	Amenazas

<ul style="list-style-type: none"> - Desarrollo de nuevas herramientas informáticas. - Centralizar la información - Reportes actualizados 	<ul style="list-style-type: none"> - Situación socioeconómica del país que afecta el mercado. - Competencia con mayor capital y con mejores procesos establecidos.
--	--

Tabla 10 Análisis FODA
Fuente: Elaboración propia

3. Análisis de involucrados

El análisis de involucrados es una herramienta que permite identificar a aquellos actores (personas, grupos o instituciones), interesados en el éxito o fracaso de un proyecto. Son también aquellos que contribuyen o que son afectados o que tiene influencia sobre problemas a enfrentar.

Involucrado	Intereses	Problemas Percibidos
Gerente	<ul style="list-style-type: none"> - Crear una herramienta informática para la gestión de cartera. - Centralizar la información - Reportes actualizados. 	<ul style="list-style-type: none"> - El costo para desarrollar una herramienta informática para la gestión de cartera es muy alto.
Supervisor de cartera	<ul style="list-style-type: none"> - Reportes de cartera actualizado - intercambio de carteras - reporte de saldos actualizados 	<ul style="list-style-type: none"> - Información descentralizada - Reportes desactualizados
Gestor de colocación y cobro	<ul style="list-style-type: none"> - Reporte de saldos actualizados 	<ul style="list-style-type: none"> - Información descentralizada - Reportes desactualizados
Desarrolladores	<ul style="list-style-type: none"> - Culminación de estudios. - Desarrollo académico y profesional 	<ul style="list-style-type: none"> - Tiempo limitado para desarrollo. - Pocos recursos económicos.

Tabla 11 Análisis de Involucrados
Fuente: Elaboración propia

4. Determinación del problema

Durante las entrevistas realizadas al personal de la pyme, se logró identificar los siguientes problemas:

- No posee una herramienta informática para la gestión de cartera.
- Uso básico de herramientas informáticas.
- Información descentralizada.
- Manejo de información con métodos manuales.
- El manejo de la información con métodos manuales genera excesivos tiempos de respuesta.
- Retrasos en la elaboración de reportes.

- Falta de control de actividades y tiempos de los gestores de cobro.

5. Elaboración de objetivos

La solución del problema principal encontrado en la pyme debe cumplir los siguientes objetivos:

Crear una herramienta informática para gestionar las carteras.

- Centralizar y organizar la información.
- Agilizar el procesamiento de datos.
- Disponibilidad de la información centralizada.

6. Árbol de problemas



Ilustración 4 Árbol de problemas.
Fuente: Elaboración propia

7. Árbol de objetivos

En el árbol de problemas se realizó un diagrama de árbol de problemas, encontrado como problema principal: No posee una herramienta informática para la gestión de cartera.

A partir de este diagrama se construye el árbol de objetivos que reúne los medios y alternativas para solucionar el problema principal. Este diagrama nos brinda una visión positiva de las situaciones negativas que aparecen en la imagen anterior.

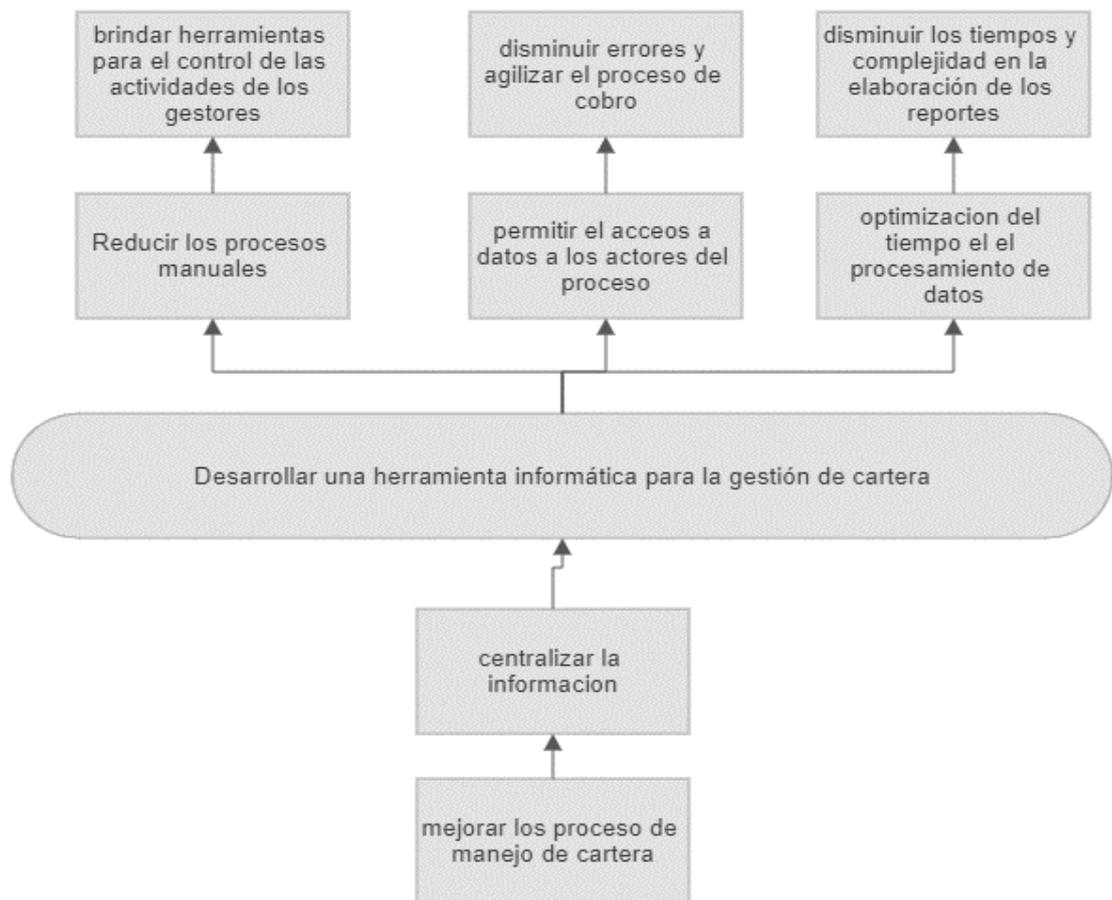


Ilustración 5 Árbol de Objetivos
Fuente: Elaboración propia

8. Selección de alternativa óptima

La selección de la alternativa óptima se obtuvo de la identificación de las acciones a realizar para contrarrestar los principales problemas expresados por el equipo de la pyme PrestaFlash. La alternativa a llevar a cabo es: Desarrollo del sistema web gerencial y aplicación móvil para gestión de control de crédito y recuperación de cartera para la pyme presta flash.

8.3 Gestión de requerimientos

1. Identificación de requerimientos

El propósito de la gestión de requerimientos es asegurar que el proyecto cumple con las expectativas de los interesados. Se entiende por requerimiento todas aquellas características observables que el usuario desea que estén contenidas en el sistema.

Para la identificación de requerimientos partiremos de la determinación de los problemas expuestos anteriormente en el árbol de problemas, esto nos dará una visión clara de las necesidades de la pyme, ya que expresan los problemas que desea resolver la empresa. Luego de este análisis y con ayuda del árbol de objetivos se deberán detallar las características de la solución.

2. Actores

Perfil usuario	Rol	Es Desarrollador	Es Usuario	Es Cliente
Gerente	Gerente		✓	✓
Supervisor	Supervisor de Cartera		✓	
Gestor	Gestor de Cobro	✓		

Tabla 12 Tabla de Actores
Fuente: Elaboración propia

3. Control de calidad

Los Fundamentos del Control de Calidad de Software presentan modelos y estándares definidos para el aseguramiento de la calidad de un sistema informático, ya sea en sus procesos y/o junto a la gestión y dimensión del mismo sistema.

La calidad de un software se define como el grado de relación que tiene un producto para satisfacer las necesidades del usuario. Un software que cumple con todos los requisitos de sus usuarios y que sus procesos se ejecutan correctamente, garantiza una buena calidad.

El control de calidad se refiere a una serie de revisiones y pruebas utilizadas durante el desarrollo de un sistema para asegurar que sigan los procedimientos y los estándares de garantía de calidad. Este incluye la comprobación de que las entregas cumplan los estándares definidos.

Para obtener un sistema competitivo de calidad y que cumpla con los objetivos propuestos pretendemos cumplir con las características indicadas en la metodología de normas de la familia ISO 25000:

Características	Detalle
Fiabilidad	Capacidad de recuperación
	Disponibilidad
	Madurez
	Tolerancia a fallos
Usabilidad	Accesibilidad
	Estética
	Operabilidad
	Protección frente a errores de usuario
Eficiencia de desempeño	Capacidad
	comportamiento temporal
	Utilización de recursos
Seguridad	Autenticidad
	Confidencialidad
	Integridad
	Responsabilidad
Mantenibilidad	Analizabilidad
	Capacidad de ser modificado
	Capacidad de ser probado
	Modularidad
Portabilidad	Adaptabilidad
	Facilidad de instalación

Tabla 13 Modelo de Calidad
Fuente: Elaboración propia

El control de calidad de este modelo cubre las fases del ciclo de vida de desarrollo de software indicado por la metodología **RAD (rapid application development)**:

- Planificación de los requisitos:
- Diseño.
- Construcción.
- Implementación.

8.4 Requerimientos No Funcionales

1. Sistema Móvil

- Disponibilidad sin conexión a internet y posterior sincronización de datos.
- Interfaz de usuario agradable y con facilidad de uso.
- Tiempos de respuesta rápida y en tiempo real.
- Debe asegurar un acompañamiento remoto para solución de incidentes en la aplicación.
- La aplicación será específicamente desarrollada en la plataforma móvil Android como versión mínima de ejecución (Android 4.0.3 – 4.0.4).

2. Sistema Web

Los requerimientos no funcionales de un sistema web son atributos de calidad o restricciones que definen cómo debe comportarse el sistema en términos de rendimiento, seguridad, usabilidad y otros aspectos.

- **Seguridad:**

- Seguridad en la Comunicación:

Se deben utilizar conexiones seguras (HTTPS) para proteger la transferencia de datos entre el cliente y el servidor.

- Autenticación y Autorización:

Se debe implementar en el sistema la autenticación de firebase OAUTH y definirse roles para los usuarios.

- **Disponibilidad:**

- Respaldo:

Se deben crear respaldos locales y respaldos en la nube. El usuario administrador es quien debe tener la responsabilidad de realizar los respaldos.

Se deben definir políticas de respaldo y planificación de su ejecución.

- **Escalabilidad:**

Firebase nos da una escalabilidad vertical y horizontal del sistema web, dependiendo del crecimiento de usuarios y recursos podemos adquirir planes que aumenten la capacidad del servidor.

- Manejo de Carga:
La arquitectura es escalable para manejar un aumento en el número de usuarios concurrentes.
- Escalabilidad de la Base de Datos:
Firebase puede escalar para manejar el crecimiento de datos.
- **Usabilidad:**
 - Diseño Responsivo:
La interfaz de usuario debe ser amigable y se adaptarse a diferentes dispositivos y tamaños de pantalla.
 - Experiencia de Usuario:
La interfaz debe ser intuitiva y con una navegación sencilla.

8.5 Requerimientos de seguridad

1. Sistema Móvil

- Se limitará el acceso de los dispositivos mediante el identificador código del dispositivo.
- Validación del acceso mediante cuentas de usuario y los privilegios que poseen, para garantizar la seguridad del sistema.

2. Sistema Web

- Validación del acceso mediante cuentas de usuario y los privilegios que poseen, para garantizar la seguridad del sistema.
- Para acceder a la Base de Datos se utiliza el oauth de FireBase.

8.6 Modelado de Requerimientos

Consiste principalmente en el modelado de casos de requerimientos (Casos de Uso), modelado de presentación (interfaces) y modelado de la información (Diagramas de clases). Utilizando el modelado de los casos de uso como base para el desarrollo de los demás (Chacón, 2006)

8.7 Modelado del negocio

El objetivo del modelado del negocio es entender la estructura y dinámica de la organización en la cual el sistema será desarrollado y asegurar que los clientes, usuarios finales y desarrolladores tengan un entendimiento común del problema actual de dicha organización. (Hernández, 20108)

En el siguiente diagrama se describen los actores y procesos principales del sistema, mediante un el modelado de caso de uso del negocio.

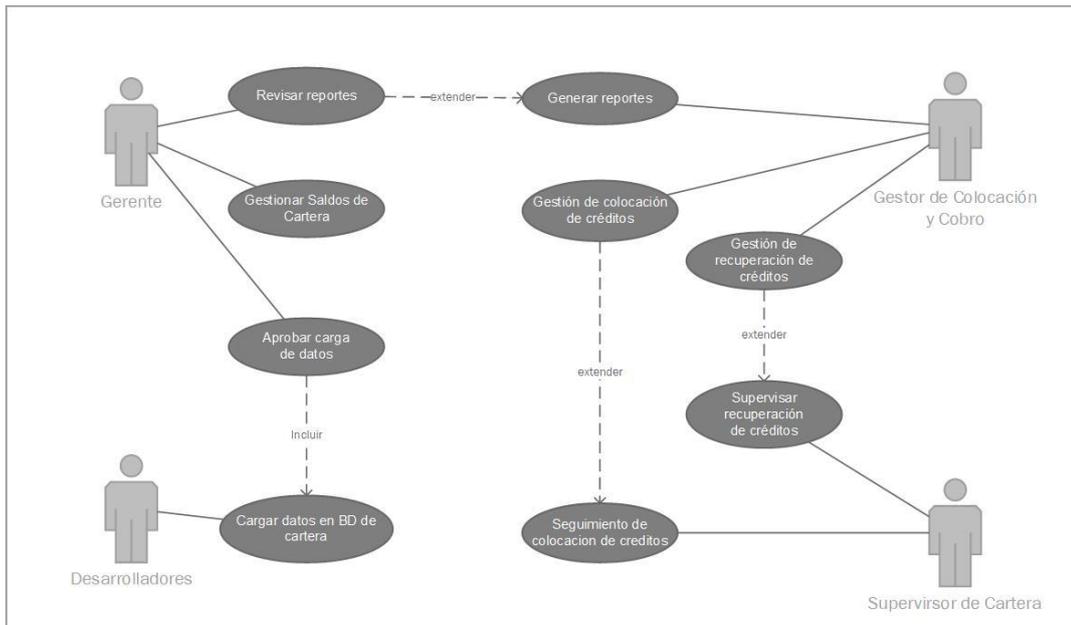


Ilustración 6 Diagrama de Caso de Uso del Negocio
Fuente: Elaboración propia

8.8 Casos de Uso del Sistema

El modelado del caso de uso del sistema es un modelado que ayuda a analista a comprender la forma en que un sistema deberá comportarse antes sus usuarios finales. (Schmullere, 1999)



Ilustración 7 Diagrama de Caso de Uso del Sistema
Fuente: Elaboración propia

8.8.1 Caso de Uso gestionar Usuario

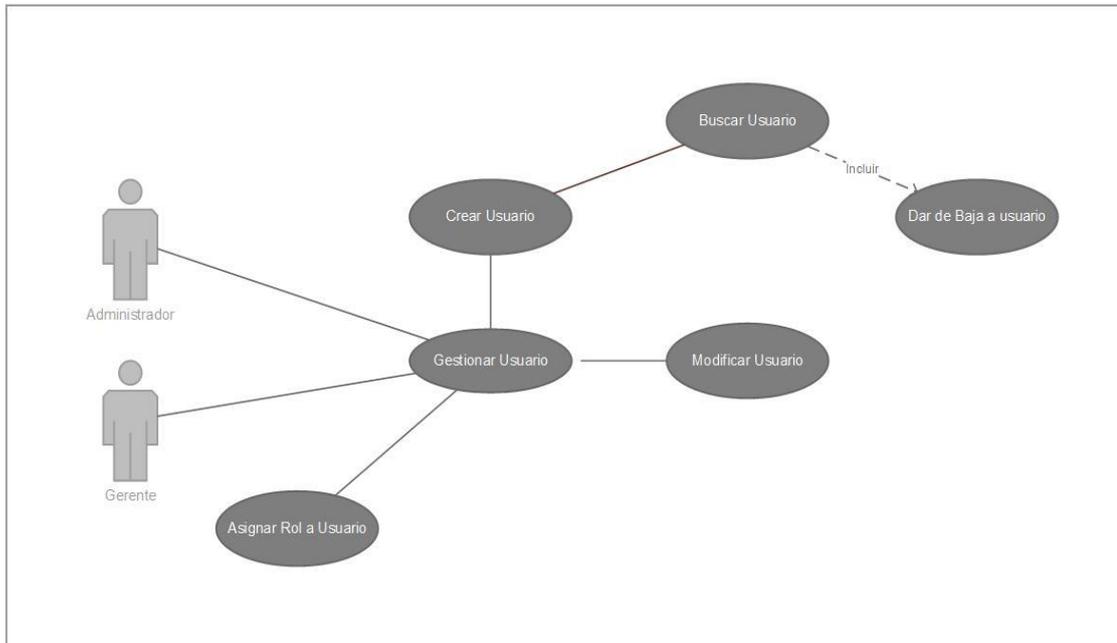


Ilustración 8 Diagrama de Caso de Uso Gestión de Usuario
Fuente: Elaboración propia

Caso de Uso	Gestionar Usuario
Descripción	Permite al usuario crear, modificar, buscar, dar de abaja a usuarios.
Tipo	Esencial.
Actores	
Nombre	Definición
Administrador	Es el encargo de crear, dar de baja a usuarios, modificar, buscar usuarios.
Gerente	Puede crear, buscar, modificar y dar de baja a usuarios.
Sistema	
Presta Flash app	El sistema ejecuta las operaciones básicas para los usuarios.
Sistema Web	
Escenarios	
Nombre	Crear usuario
Precondición	<ul style="list-style-type: none"> - Deben existir roles registrados. - El usuario debe tener los permisos requeridos para realizar la acción.

Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de usuario. - Ejecutar el botón agregar usuario. - Ingresa los datos requeridos para el registro del usuario. - Ejecutar el botón guardar. - El sistema ingresa el registro a la Base de Datos. - El sistema muestra en la lista de usuarios incluyendo el nuevo registro.
Poscondición	El sistema realiza el registro del nuevo usuario.
Excepciones	<ul style="list-style-type: none"> - Información incompleta. - Registro Existente.
Nombre	Modificar Usuario
Precondición	<ul style="list-style-type: none"> - El usuario debe tener los permisos requeridos para realizar la acción. - El usuario debe estar registrado en la Base de Datos.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de usuario. - Buscar el registro a modificar. - Ejecutar el botón editar. - Registrar nuevos datos. - Ejecutar el botón guardar. - El sistema modifica el registro seleccionado en la Base de Datos. - El sistema muestra en la lista de usuarios, el registro modificado con los datos actualizados.
Poscondición	El sistema actualiza los datos del usuario.
Excepciones	<ul style="list-style-type: none"> - Información incompleta.
Nombre	Dar de Baja Usuario
Precondición	<ul style="list-style-type: none"> - El usuario debe existir y estar activo en la Base de Datos. - El usuario debe tener los permisos requeridos para realizar la acción.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de usuario. - Buscar el registro a modificar. - Para la aplicación se debe dar clic sostenido sobre el registro a dar de baja. - El sistema solicitara confirmación para dar de baja al usuario. - Ejecutar el botón aceptar. - El sistema dará de baja el registro seleccionado en la Base de Datos. - El sistema muestra la lista de usuarios actualizada sin el registro dado de baja.
Poscondición	<ul style="list-style-type: none"> - El sistema debe dar de baja al usuario. - El registro del usuario se conserva en la base de datos.
Excepciones	N/A

Nombre	Buscar Usuario
Precondición	<ul style="list-style-type: none"> - El usuario debe tener los permisos requeridos para realizar la acción.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de usuario. - Seleccione la opción búsqueda desde el menú desplegable desde la aplicación móvil. - Digite el criterio de búsqueda. - El sistema realizara la búsqueda según el criterio. - El sistema muestra la lista de usuarios que cumplen el criterio de búsqueda.
Poscondición	El sistema muestra el usuario buscado.
Excepciones	<ul style="list-style-type: none"> - Criterio de búsqueda vacío. - Criterio de búsqueda incorrecto.
Nombre	Asignar Cartera a Usuario
Precondición	<ul style="list-style-type: none"> - El usuario debe tener los permisos requeridos para realizar la acción. - El usuario a asignar cartera debe tener roles de Gerente o Gestor de Colocación y Cobro.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de usuario. - Seleccionar el usuario a asignar cartera. - Ejecutar el botón editar. - Asignar la nueva cartera de la lista de carteras. - Ejecutar el botón guardar. - El sistema modifica el registro seleccionado en la Base de Datos. - El sistema muestra en la lista de usuarios, el registro actualizado.
Poscondición	El sistema asigna correctamente la cartera al Usuario.
Excepciones	<ul style="list-style-type: none"> - El usuario no se encuentra activo. -

Tabla 14 Caso de Uso Gestionar Usuario
Fuente: Elaboración propia

8.8.2 Caso de Uso Gestión de Cliente

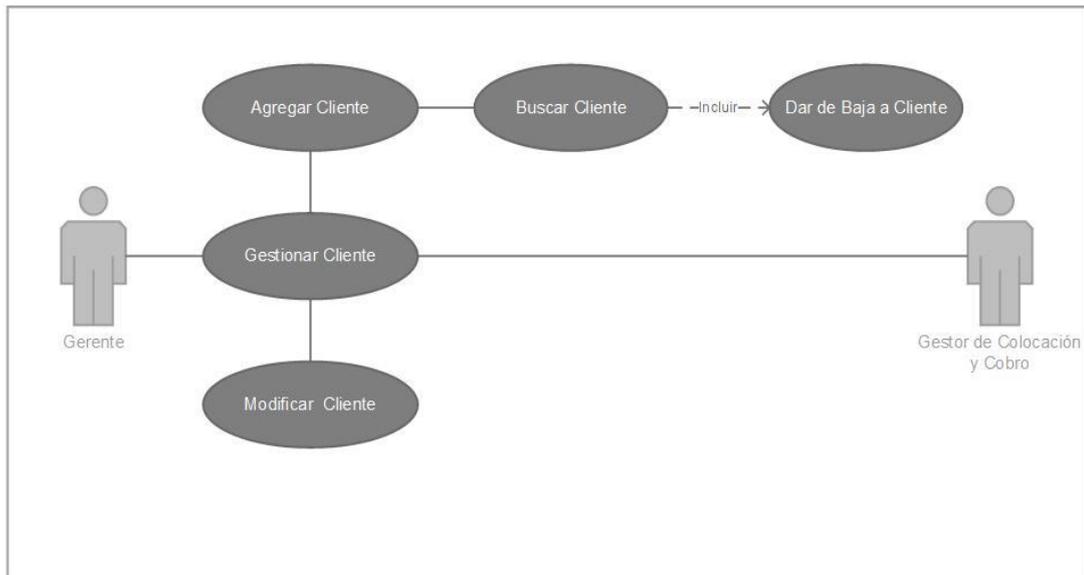


Ilustración 9 Diagrama de Caso de Uso Gestión de Cliente
Fuente: Elaboración propia

Caso de Uso	Gestionar Cliente
Descripción	Permite al usuario crear, modificar, buscar, dar de baja a clientes.
Tipo	Esencial
Actores	
Nombre	Definición
Gestor de Cartera y Cobro	Es el encargo de crear, dar de baja a clientes, modificar, buscar clientes.
Gerente	Puede crear, buscar, modificar y dar de baja a clientes.
Sistema	
App móvil PrestaFlash	
Escenarios	
Nombre	Crear Cliente
Precondición	<ul style="list-style-type: none"> - Deben existir carteras registrados. - El usuario debe tener los permisos requeridos para realizar la acción.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de clientes. - Ejecutar el botón agregar clientes. - Ingresa los datos requeridos para el registro del cliente. - Ejecutar el botón guardar. - El sistema ingresa el registro a la Base de Datos.
Postcondición	El sistema muestra la lista de cliente incluyendo el nuevo registro, según la cartera asignada al cliente.

Excepciones	<ul style="list-style-type: none"> - Información incompleta. - Registro Existente.
Nombre	Modificar Cliente
Precondición	<ul style="list-style-type: none"> - El usuario debe tener los permisos requeridos para realizar la acción. - El cliente debe estar registrado en la Base de Datos.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de clientes. - Seleccionar la cartera a la que pertenece el cliente. - Buscar el registro a modificar. - Ejecutar el botón editar. - Registrar nuevos datos. - Ejecutar el botón guardar. - El sistema modifica el registro seleccionado en la Base de Datos.
Postcondición	El sistema muestra en la lista de clientes, el registro modificado con los datos actualizados.
Excepciones	Información incompleta.
Nombre	Dar de baja a Cliente
Precondición	<ul style="list-style-type: none"> - El cliente debe existir y estar activo en la Base de Datos. - El usuario debe tener los permisos requeridos para realizar la acción.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de clientes. - Buscar el registro a modificar. - Ejecutar el botón editar - El sistema solicitara confirmación para dar de baja al usuario. - Ejecutar el botón aceptar. - El sistema dará de baja el registro seleccionado en la Base de Datos. <p>El sistema muestra la lista de clientes actualizada sin el registro dado de baja.</p>
Postcondición	El sistema muestra la lista de clientes incluyendo el nuevo registro, según la cartera asignada al cliente.
Excepciones	N/A
Nombre	Buscar Cliente
Precondición	<ul style="list-style-type: none"> - El usuario debe tener los permisos requeridos para realizar la acción.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de clientes. - Digite en la caja de búsqueda el criterio de búsqueda. - El sistema realizara la búsqueda según el criterio. - El sistema muestra la lista de clientes que cumplen el criterio de búsqueda.
Poscondición	El sistema muestra el cliente buscado.

Excepciones	<ul style="list-style-type: none"> - Criterio de búsqueda vacío. - Criterio de búsqueda incorrecto.
--------------------	---

Tabla 15 Caso de Uso Gestionar Cliente
Fuente: Elaboración propia

8.8.3 Caso de Uso Gestión de Préstamo

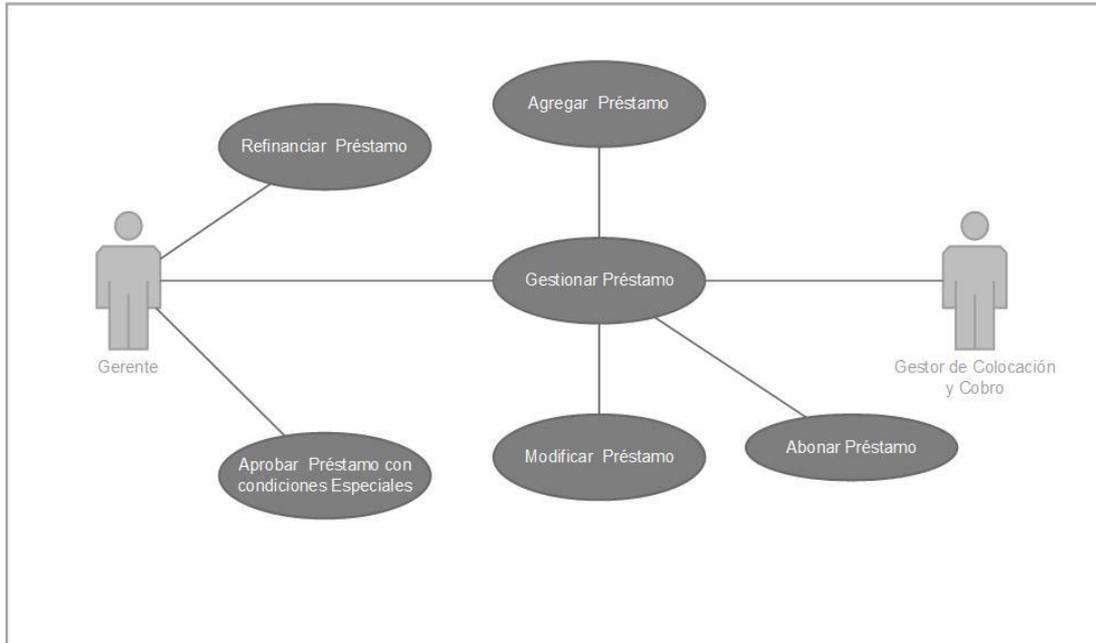


Ilustración 10 Diagrama de Caso de Uso Gestión de Préstamos
Fuente: Elaboración propia

Caso de Uso	Gestionar Préstamo
Descripción	Permite al usuario crear, modificar, buscar, abonar y refinanciar los créditos.
Tipo	Esencial
Actores	
Nombre	Definición
Gestor de Cartera y Cobro	Es el encargo de crear, buscar y abonar créditos.
Gerente	Puede crear, buscar, modificar, abonar y refinanciar créditos.
Sistema	
App móvil PrestaFlash	
Escenarios	
Nombre	Agregar Préstamo
Precondición	<ul style="list-style-type: none"> - El usuario debe tener los permisos requeridos para realizar la acción. - El usuario debe estar asignado a una cartera.

	<ul style="list-style-type: none"> - Deben existir clientes activos en la cartera a la que está asignado el usuario. - Deben existir al menos un plan de pagos activo.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de Préstamos. - Ejecutar el botón agregar préstamos. - Seleccione el cliente a quien otorgará el crédito. - Ingresa los datos requeridos para el registro del cliente. - Ejecutar el botón guardar.
Postcondición	<ul style="list-style-type: none"> - El sistema genera la tabla de amortización, de cuota nivelada, según la tasa de interés y plazos seleccionado en el plan de pago al momento de agregar el préstamo. - El sistema muestra la lista de préstamos incluyendo el nuevo registro, según la cartera asignada cliente.
Excepciones	Información incompleta.
Nombre	Buscar Préstamo
Precondición	<ul style="list-style-type: none"> - El usuario debe tener los permisos requeridos para realizar la acción. - El usuario debe estar asignado a la cartera a la que pertenece el cliente.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de préstamos. - Seleccionar la cartera a la que pertenece el préstamo, desde el menú desplegable. - Digite en la caja de búsqueda el criterio de búsqueda. - El sistema realizara la búsqueda según el criterio. - El sistema muestra la lista de préstamos que cumplen el criterio de búsqueda.
Postcondición	El sistema muestra el préstamo buscado.
Excepciones	<ul style="list-style-type: none"> - Criterio de búsqueda vacío. - Criterio de búsqueda incorrecto.
Nombre	Refinanciar Préstamo
Precondición	<ul style="list-style-type: none"> - El usuario debe tener los permisos requeridos para realizar la acción. - El usuario debe estar asignado a una cartera. - El préstamo a refinanciar debe estar sin cancelar. - Deben existir al menos un plan de pagos activo.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de Préstamos. - Seleccionar la cartera a la que pertenece el préstamo, desde el menú desplegable. - Seleccionar el préstamo deseado. - Ejecutar el botón editar préstamos. - Ingresa los datos requeridos para el refinanciamiento del préstamo. - Ejecutar el botón guardar.

Postcondición	<ul style="list-style-type: none"> - El sistema genera la nueva tabla de amortización, de cuota nivelada, según la tasa de interés y plazos seleccionado en el plan de pago al momento de editar el préstamo. - El sistema muestra la lista de préstamos incluyendo el registro modificado, según la cartera asignada cliente.
Excepciones	N/A
Nombre	Abonar Préstamo
Precondición	<ul style="list-style-type: none"> - El usuario debe tener los permisos requeridos para realizar la acción. - El usuario debe estar asignado a la cartera a la que pertenece el cliente.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de Préstamos. - Seleccionar la cartera a la que pertenece el préstamo, desde el menú desplegable. - Ejecutar el botón Agregar Recibo, del menú lateral. - Ingresa los datos requeridos. - Ejecutar el botón guardar.
Postcondición	<ul style="list-style-type: none"> - El sistema agregar el registro del recibo y actualiza el saldo de la deuda. - El sistema muestra la lista de préstamos incluyendo registro abonado, según la cartera asignada cliente.
Excepciones	N/A

Tabla 16 Caso de Uso Gestionar Préstamo
Fuente: Elaboración propia

8.8.4 Caso de uso gestionar Cartera

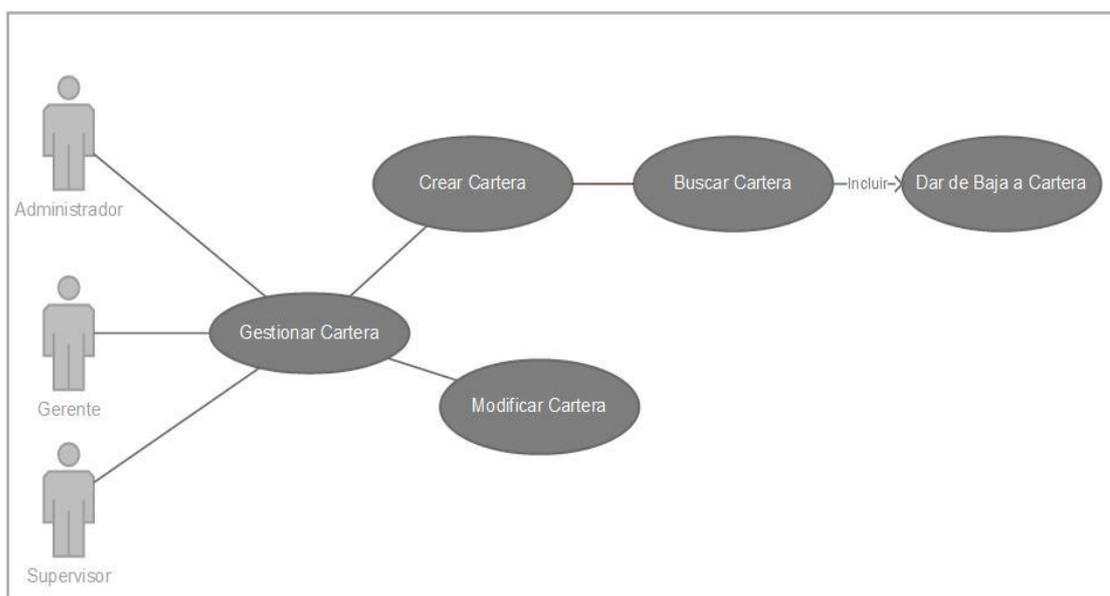


Ilustración 11 Caso de uso gestionar Cartera
Fuente: Elaboración propia

Caso de Uso	Gestionar Cartera
Descripción	Permite al usuario crear, modificar, buscar, dar de baja a carteras.
Tipo	Primario.
Actores	
Nombre	Definición
Administrador	Es el encargo de crear, dar de baja a carteras, modificar, buscar carteras.
Gerente	Puede crear, buscar, modificar y dar de baja a carteras.
Superviso de Cartera	Puede buscar y modificar carteras.
Sistema	
Presta Flash app	El sistema ejecuta las operaciones básicas para las carteras.
Escenarios	
Nombre	Crear Cartera
Precondición	- El usuario debe tener los permisos requeridos para realizar la acción.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de Configuración. - Seleccionar la opción Agregar cartera del menú lateral izquierdo. - Ingresa los datos requeridos para el registro de la cartera. - Ejecutar el botón guardar. - El sistema ingresa el registro a la Base de Datos. - El sistema muestra la lista de carteras, incluyendo el nuevo registro.
Poscondición	El sistema realiza el registro de la nueva cartera.
Excepciones	- Información incompleta.
Nombre	Modificar Cartera
Precondición	<ul style="list-style-type: none"> - El usuario debe tener los permisos requeridos para realizar la acción. - La cartera debe estar registrado en la Base de Datos.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de Configuración. - Seleccionar la opción Agregar cartera del menú lateral izquierdo. - Seleccionar el registro a editar de la lista de carteras visibles en el formulario. - Registrar nuevos datos. - Ejecutar el botón guardar.

	<ul style="list-style-type: none"> - El sistema modifica el registro seleccionado en la Base de Datos. - El sistema muestra en la lista de carteras, el registro modificado con los datos actualizados.
Poscondición	El sistema actualiza los datos de la cartera.
Excepciones	- Información incompleta.
Nombre	Dar de baja a Cartera
Precondición	<ul style="list-style-type: none"> - La cartera debe existir y estar activo en la Base de Datos. - El usuario debe tener los permisos requeridos para realizar la acción.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de Configuración. - Seleccionar la opción Agregar cartera del menú lateral izquierdo. - Seleccionar el registro a editar de la lista de carteras visibles en el formulario. - Ejecutar el botón aceptar. - El sistema dará de baja el registro seleccionado en la Base de Datos. - El sistema muestra la lista de carteras actualizada con el registro dado de baja.
Poscondición	<ul style="list-style-type: none"> - El sistema debe dar de baja a la cartera. - El registro de la cartera se conserva en la base de datos.
Excepciones	La cartera tiene créditos activos.
Nombre	Buscar Cartera
Precondición	- El usuario debe tener los permisos requeridos para realizar la acción.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de Configuración. - Seleccionar la opción Agregar cartera del menú lateral izquierdo. - Digite en la caja de búsqueda el criterio de búsqueda. - El sistema realizara la búsqueda según el criterio. - El sistema muestra la lista de carteras que cumplen el criterio de búsqueda.
Poscondición	El sistema muestra la cartera buscada.
Excepciones	<ul style="list-style-type: none"> - Criterio de búsqueda vacío. - Criterio de búsqueda incorrecto.

Tabla 17 Caso de uso gestionar Cartera

Fuente: Elaboración propia

8.8.5 Caso de uso gestionar Plan de Negocio

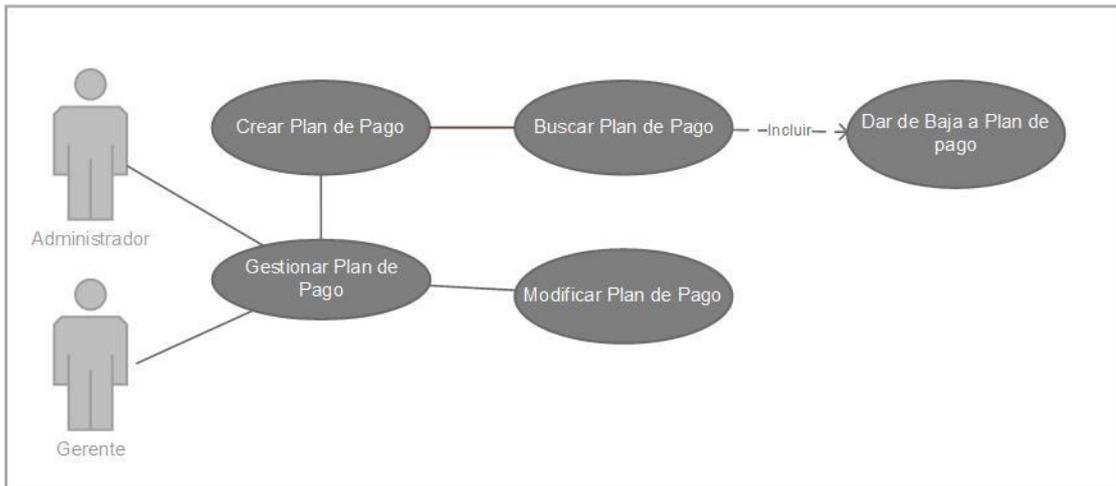


Ilustración 12 Caso de uso Gestionar Plan de Financiamiento
Fuente: Elaboración propia

Caso de Uso	Gestionar Plan de Financiamiento
Descripción	Permite al usuario crear, modificar, buscar, dar de abaja a Plan de financiamiento.
Tipo	Primario.
Actores	
Nombre	Definición
Administrador	Es el encargo de crear, dar de baja a plan de financiamiento, modificar, buscar plan de financiamiento.
Gerente	Puede crear, buscar, modificar y dar de baja a planes de financiamiento.
Sistema	
Presta Flash app	El sistema ejecuta las operaciones básicas para los planes de financiamiento.
Escenarios	
Nombre	Crear Plan de Financiamiento.
Precondición	- El usuario debe tener los permisos requeridos para realizar la acción.
Operaciones	- Seleccionar formulario de Configuración. - Seleccionar la opción Agregar Plan de Financiamiento del menú lateral izquierdo. - Ingresar los datos requeridos para el registro del plan de financiamiento. - Ejecutar el botón guardar. - El sistema ingresa el registro a la Base de Datos. - El sistema muestra la lista de planes de financiamiento, incluyendo el nuevo registro.

Poscondición	El sistema realiza el registro del nuevo plan de financiamiento
Excepciones	- Información incompleta.
Nombre	Modificar Plan de Financiamiento
Precondición	<ul style="list-style-type: none"> - El usuario debe tener los permisos requeridos para realizar la acción. - El plan de financiamiento debe estar registrado en la Base de Datos.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de Configuración. - Seleccionar la opción Agregar plan de financiamiento del menú lateral izquierdo. - Seleccionar el registro a editar de la lista de planes de financiamiento visibles en el formulario. - Registrar nuevos datos. - Ejecutar el botón guardar. - El sistema modifica el registro seleccionado en la Base de Datos. - El sistema muestra en la lista de planes de financiamiento, el registro modificado con los datos actualizados.
Poscondición	El sistema actualiza los datos del plan de financiamiento.
Excepciones	- Información incompleta.
Nombre	Dar de baja a Plan de Financiamiento
Precondición	<ul style="list-style-type: none"> - El plan de financiamiento debe existir y estar activo en la Base de Datos. - El usuario debe tener los permisos requeridos para realizar la acción.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de Configuración. - Seleccionar la opción Agregar plan de financiamiento del menú lateral izquierdo. - Seleccionar el registro a editar de la lista de planes de financiamiento visibles en el formulario. - Ejecutar el botón aceptar. - El sistema dará de baja el registro seleccionado en la Base de Datos. - El sistema muestra la lista de planes de financiamiento actualizada con el registro dado de baja.
Poscondición	<ul style="list-style-type: none"> - El sistema debe dar de baja al plan de financiamiento. - El registro del plan de financiamiento se conserva en la base de datos.
Excepciones	El plan de negocio es usado por créditos activos.
Nombre	Buscar Plan de Financiamiento
Precondición	- El usuario debe tener los permisos requeridos para realizar la acción.
Operaciones	- Seleccionar formulario de Configuración.

	<ul style="list-style-type: none"> - Seleccionar la opción Agregar plan de financiamiento del menú lateral izquierdo. - Digite en la caja de búsqueda el criterio de búsqueda. - El sistema realizara la búsqueda según el criterio. - El sistema muestra la lista de planes de financiamiento que cumplen el criterio de búsqueda.
Poscondición	El sistema muestra el plan de financiamiento buscado.
Excepciones	<ul style="list-style-type: none"> - Criterio de búsqueda vacío. - Criterio de búsqueda incorrecto.

Tabla 18 Caso de Uso Gestionar Plan de Financiamiento
Fuente: Elaboración propia

8.8.6 Caso de uso Gestionar Dispositivo

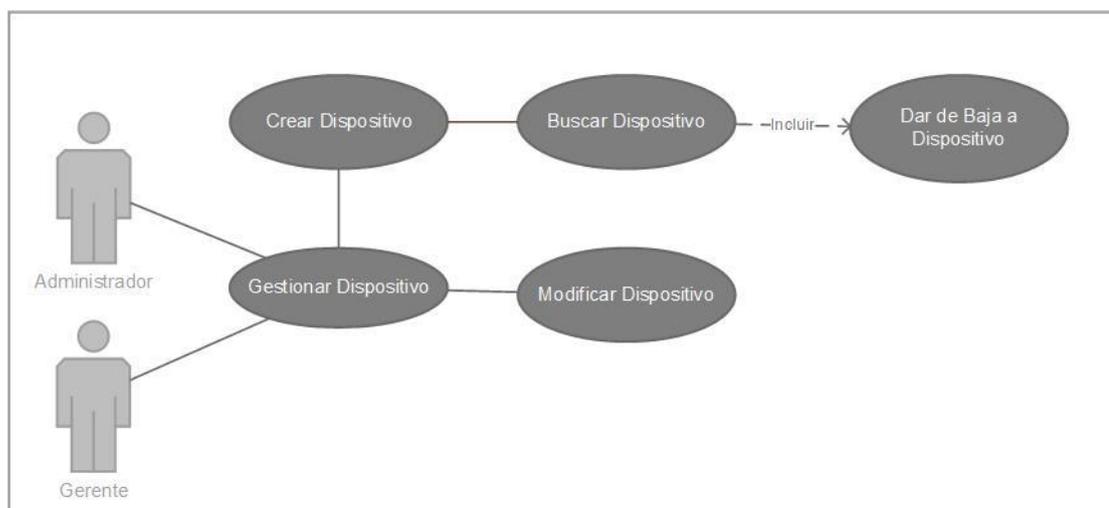


Ilustración 13 Caso de Uso Gestionar Dispositivo
Fuente: Elaboración propia

Caso de Uso	Gestionar Dispositivo
Descripción	Permite al usuario crear, modificar, buscar, dar de abaja a un dispositivo.
Tipo	Primario.
Actores	
Nombre	Definición
Administrador	Es el encargo de crear, dar de baja a dispositivos, modificar, buscar dispositivo.
Gerente	Puede crear, buscar, modificar y dar de baja a dispositivo.
Sistema	
Presta Flash app	El sistema ejecuta las operaciones básicas para los planes de financiamiento.

Sistema Web	El sistema ejecuta las operaciones básicas para los planes de financiamiento.
Escenarios	
Nombre	Crear Dispositivo.
Precondición	- El usuario debe tener los permisos requeridos para realizar la acción.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de Configuración. - Seleccionar la opción Dispositivo del menú lateral izquierdo. - Ingresar los datos requeridos para el registro del dispositivo. - Ejecutar el botón guardar. - El sistema ingresa el registro a la Base de Datos. - El sistema muestra la lista de dispositivos, incluyendo el nuevo registro.
Poscondición	El sistema realiza el registro del nuevo dispositivo.
Excepciones	- Información incompleta.
Nombre	Modificar Dispositivo
Precondición	<ul style="list-style-type: none"> - El usuario debe tener los permisos requeridos para realizar la acción. - El dispositivo debe estar registrado en la Base de Datos.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de Configuración. - Seleccionar la opción Agregar dispositivo del menú lateral izquierdo. - Seleccionar el registro a editar de la lista de planes de financiamiento visibles en el formulario. - Registrar nuevos datos. - Ejecutar el botón guardar. - El sistema modifica el registro seleccionado en la Base de Datos. - El sistema muestra en la lista de dispositivos, el registro modificado con los datos actualizados.
Poscondición	El sistema actualiza los datos del dispositivo.
Excepciones	- Información incompleta.
Nombre	Dar de baja a Dispositivo
Precondición	<ul style="list-style-type: none"> - El dispositivo debe existir y estar activo en la Base de Datos. - El usuario debe tener los permisos requeridos para realizar la acción.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de Configuración. - Seleccionar la opción Agregar dispositivo del menú lateral izquierdo. - Seleccionar el registro a editar de la lista de dispositivos visibles en el formulario. - Ejecutar el botón aceptar.

	<ul style="list-style-type: none"> - El sistema dará de baja el registro seleccionado en la Base de Datos. - El sistema muestra la lista de dispositivos actualizada con el registro dado de baja.
Poscondición	<ul style="list-style-type: none"> - El sistema debe dar de baja al plan de financiamiento. - El registro del plan de financiamiento se conserva en la base de datos.
Excepciones	N/A
Nombre	Buscar Dispositivo
Precondición	<ul style="list-style-type: none"> - El usuario debe tener los permisos requeridos para realizar la acción.
Operaciones	<ul style="list-style-type: none"> - Seleccionar formulario de Configuración. - Seleccionar la opción Agregar dispositivo del menú lateral izquierdo. - Digite en la caja de búsqueda el criterio de búsqueda. - El sistema realizara la búsqueda según el criterio. - El sistema muestra la lista de planes de financiamiento que cumplen el criterio de búsqueda.
Poscondición	El sistema muestra el dispositivo buscado.
Excepciones	<ul style="list-style-type: none"> - Criterio de búsqueda vacío. - Criterio de búsqueda incorrecto.

Tabla 19 Caso de uso Gestionar Dispositivo
Fuente: Elaboración propia

8.8.7 Caso de Uso Generar Reporte

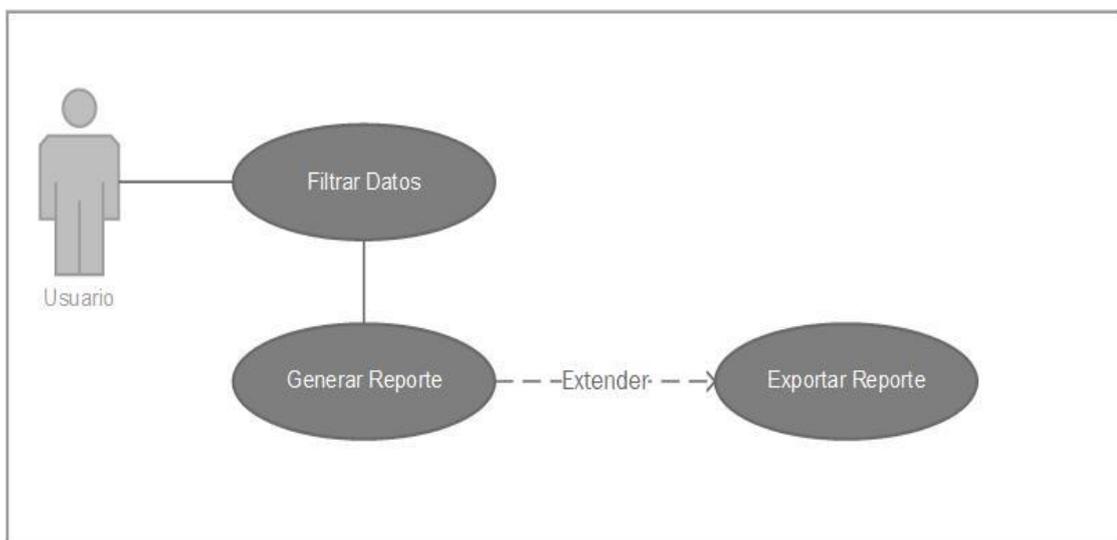


Ilustración 14 Diagrama de Caso de Uso Generar Reportes
Fuente: Elaboración propia

Caso de Uso	Generar Reportes
Descripción	Permite al Usuario visualizar y descargar reportes según los criterios deseados.
Tipo	Esencial
Actores	
Nombre	Definición
Gerente	Puede generar reportes para visibilidad de indicadores.
Supervisor de Cartera	Responsable de seguimiento de cartera.
Sistema	
App móvil PrestaFlash	
Sistema Web	
Escenarios	
Nombre	Generar reportes de Cartera
Precondición	- El usuario debe tener los permisos requeridos para realizar la acción.
Operaciones	- Seleccione el formulario de reportes. - Seleccione el reporte a generar. - Indique los parámetros deseados. - Ejecute el botón Generar. - El sistema genera la información del reporte. - El sistema genera un archivo PDF que puede ser exportado.
Postcondición	- El sistema reportes solicitados por el usuario. - El sistema sustituye el reporte existente (según el tipo de reporte) en el directorio indicado al momento de generar lo.
Excepciones	No existen registros que cumplen el criterio de filtro.

Tabla 20 Caso de Uso Generar Reportes
Fuente: Elaboración propia

8.9 Diagrama de Secuencia

Los diagramas de secuencia en el UML representan los eventos en orden cronológico, describe básicamente como los actores y objetos interactúan entre sí en una secuencia dinámica. (J. Rumbaugh, 2000)

8.9.1 Diagrama de secuencia crear usuario

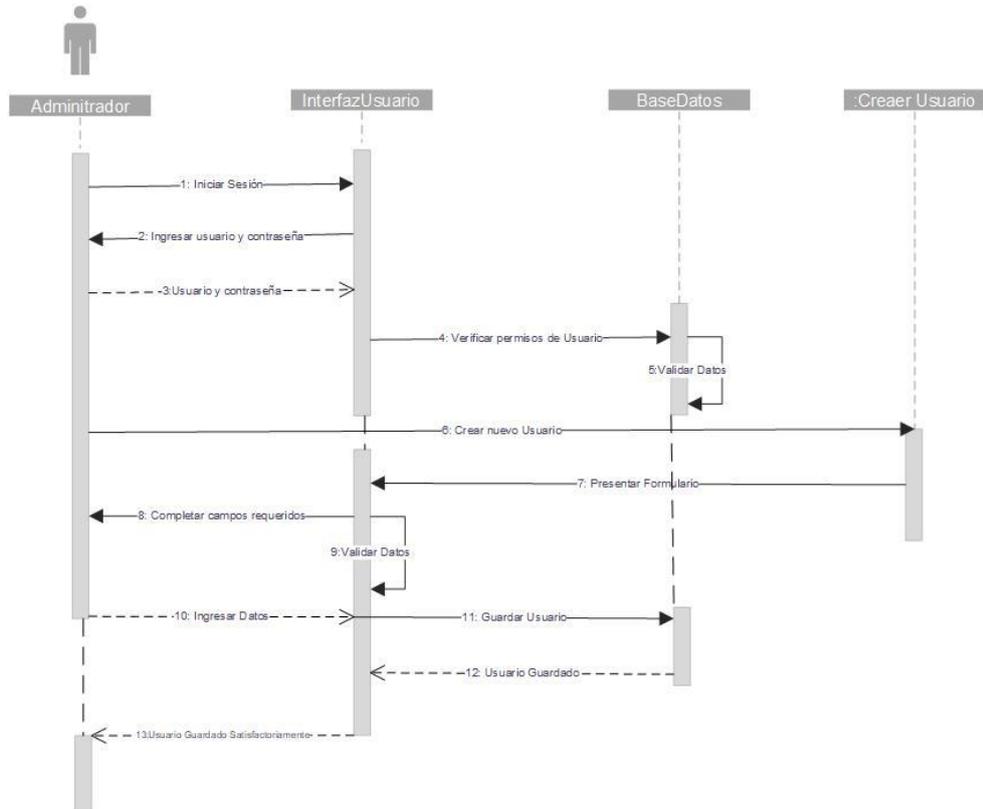


Ilustración 15 Diagrama de secuencia crear usuario
Fuente: Elaboración propia

8.9.2 Diagrama de secuencia Agregar Préstamo

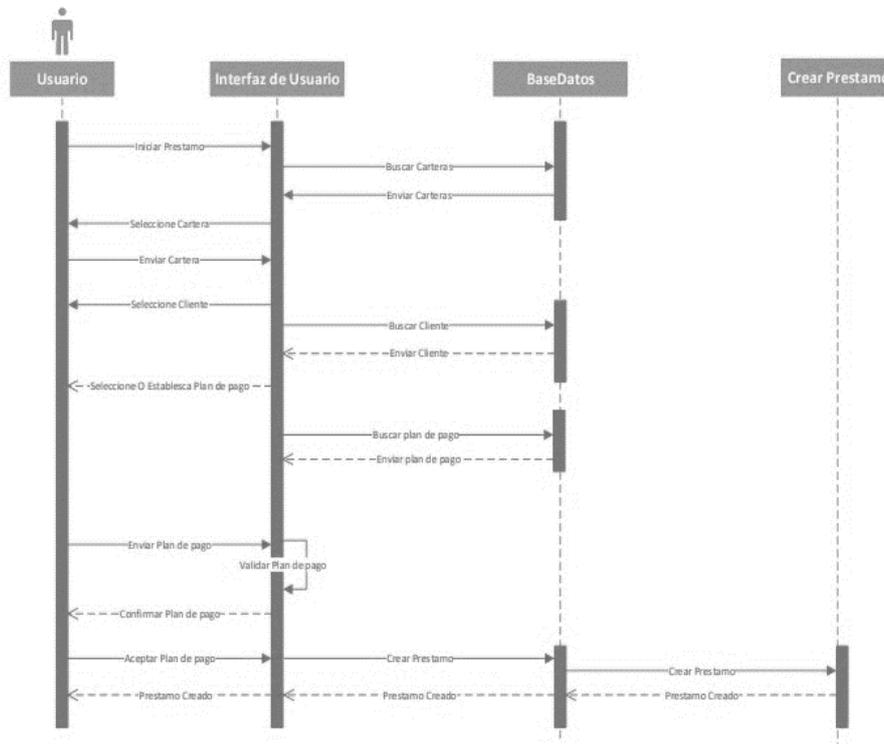


Ilustración 16 Diagrama de secuencia crear Préstamo
Fuente: Elaboración propia

8.9.3 Diagrama de secuencia Agregar Cliente

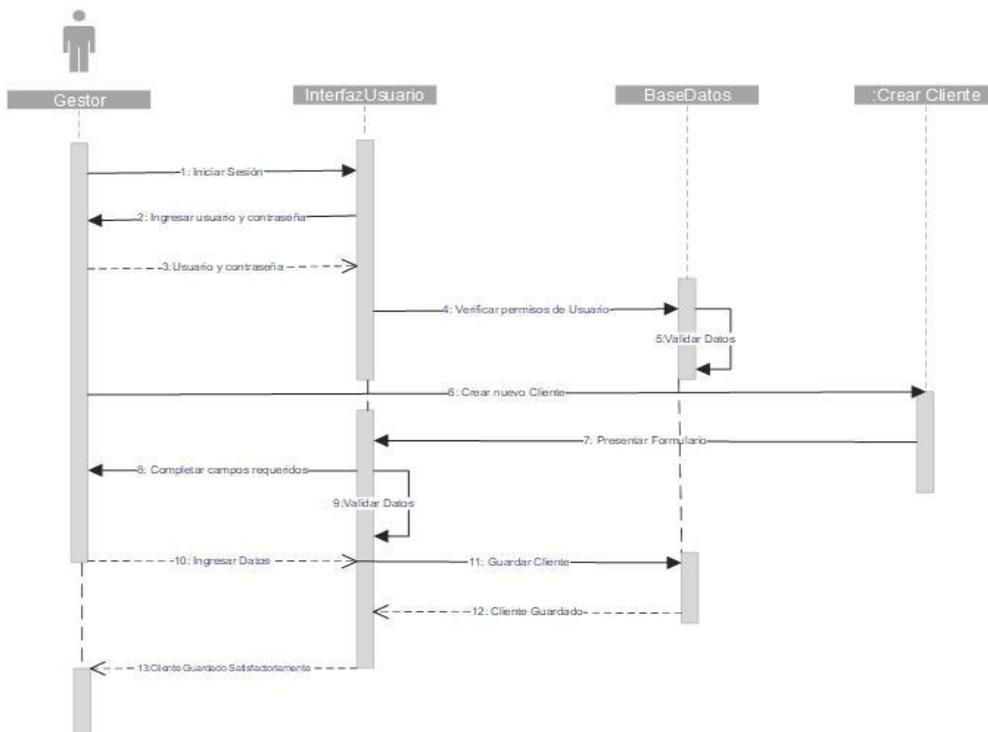


Ilustración 17 Diagrama de secuencia crear Cliente
Fuente: Elaboración propia

8.10 Diagrama de Estados

Los diagramas de estados muestran el conjunto de estados por los cuales pasa o pueden pasar un objeto durante su vida en el sistema.

8.10.1 Diagrama de estado cliente

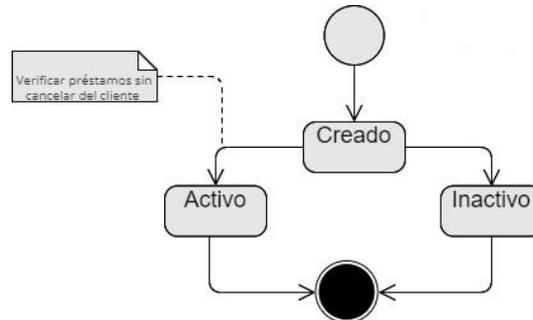


Ilustración 18 Diagrama de estado de cliente
Fuente: Elaboración propia

8.10.2 Diagrama de estado usuario

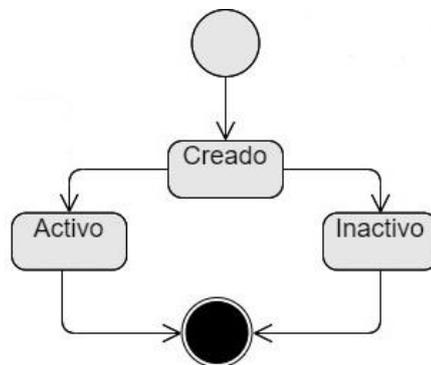


Ilustración 19 Diagrama de estado de usuario
Fuente: Elaboración propia

8.10.3 Diagrama de estado préstamo

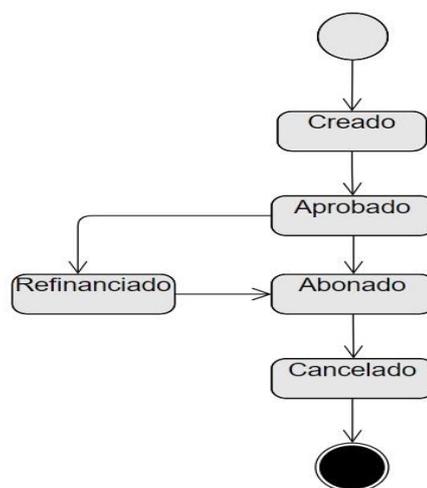


Ilustración 20 Diagrama de estado de préstamo
Fuente: Elaboración propia

IX. Fase de Construcción

Preparación del entorno para codificar los diferentes elementos del modelo seleccionado.

El objetivo de esta actividad es asegurar la disponibilidad de todos los medios y facilidades con la cual se pudo llevar a cabo la construcción de la herramienta de software. Entre estos medios están la preparación del gestor de base de datos, bibliotecas de programas, herramientas de generación de código, entre otros.

Ambiente de desarrollo:

- Instalación y configuración de Android Studio
- Instalación y configuración de Visual Studio Code
- Instalación y configuración del Gestor de bases de datos FIREBASE
- Lenguajes de programación:
 1. Java
 2. JavaScript (Bootstrap, html y CSS)

Codificar los elementos de modelo seleccionado

La codificación: se refiere a la codificación del problema a un lenguaje de programación. El objetivo de esta actividad es interpretar todas las especificaciones del usuario y técnicas a un lenguaje entendible por la computadora. La programación es una actividad totalmente técnica que se inicia tomando como base la documentación obtenida de las especificaciones técnicas.

Esta actividad da como resultado programas de aplicación terminados que han sido compilados del lenguaje de programación al lenguaje objeto, y que han sido probados. Esta actividad debe generar programas documentados y sustentables.

Con esta fase cumplimos con el objetivo específico número 3 del proyecto.

En la Siguiete imagen se muestra el panel de administración de FIREBASE

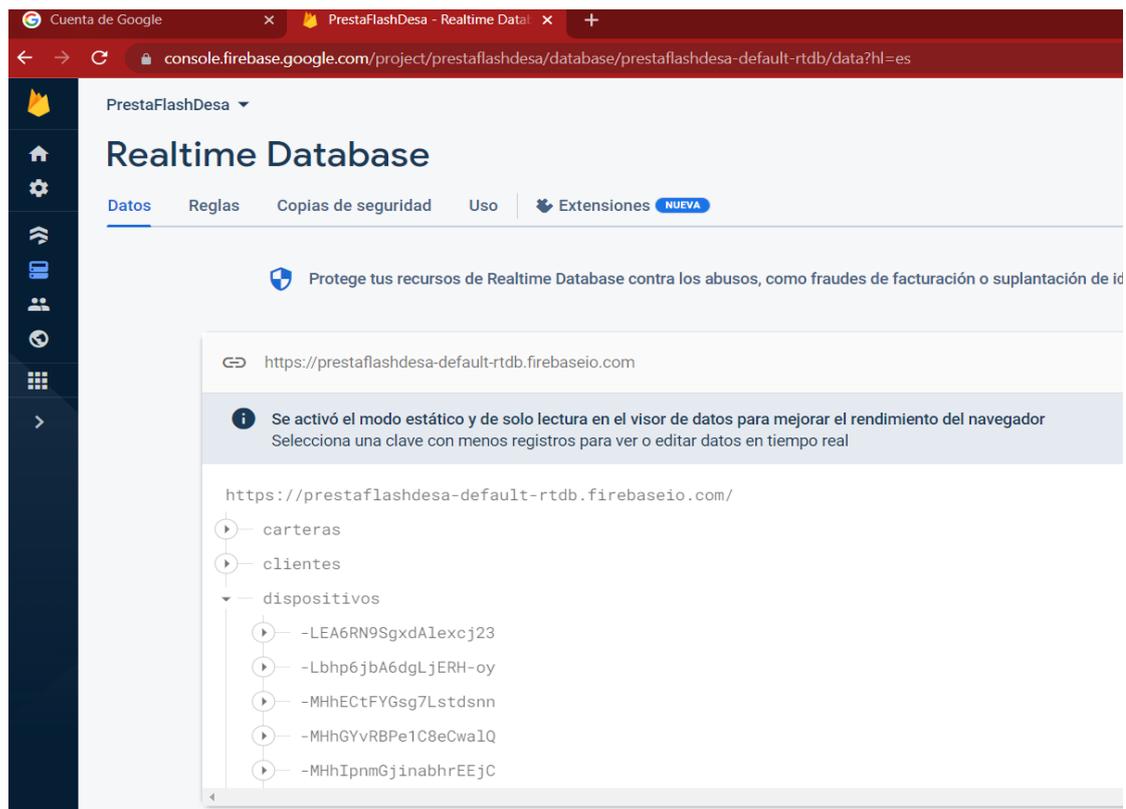


Ilustración 21 Panel de administración de FIREBASE
Fuente: Elaboración propia

En la siguiente imagen se muestra la codificación del sistema móvil en Android Studio

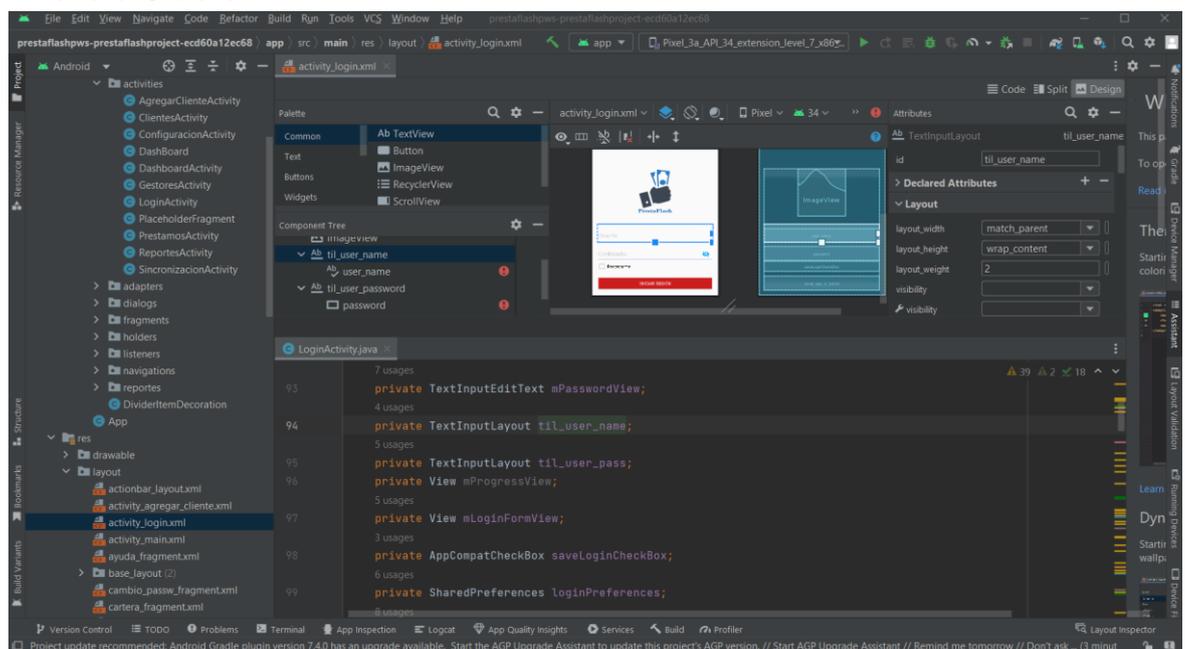


Ilustración 22 Codificación de aplicación móvil Presta Flash
Fuente: Elaboración propia

En la siguiente imagen se muestra la codificación del sistema web en el editor Visual Studio Code.

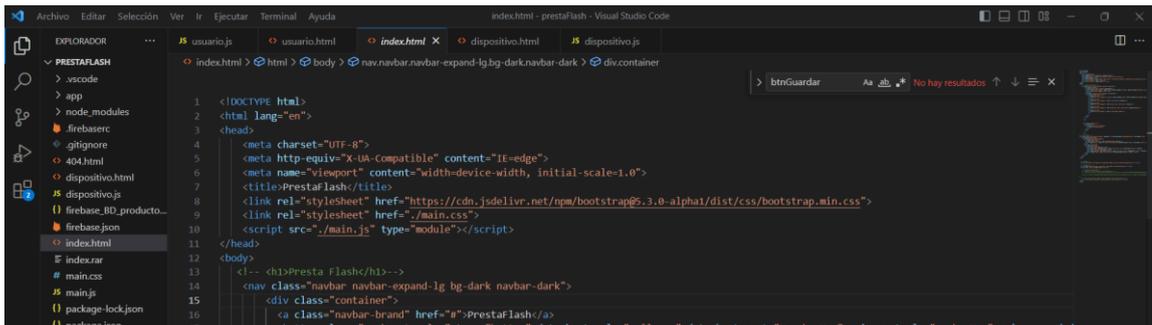


Ilustración 23 Codificación de sistema web gerencial PrestaFlash
Fuente: Elaboración propia

Estructura del código en la aplicación móvil.

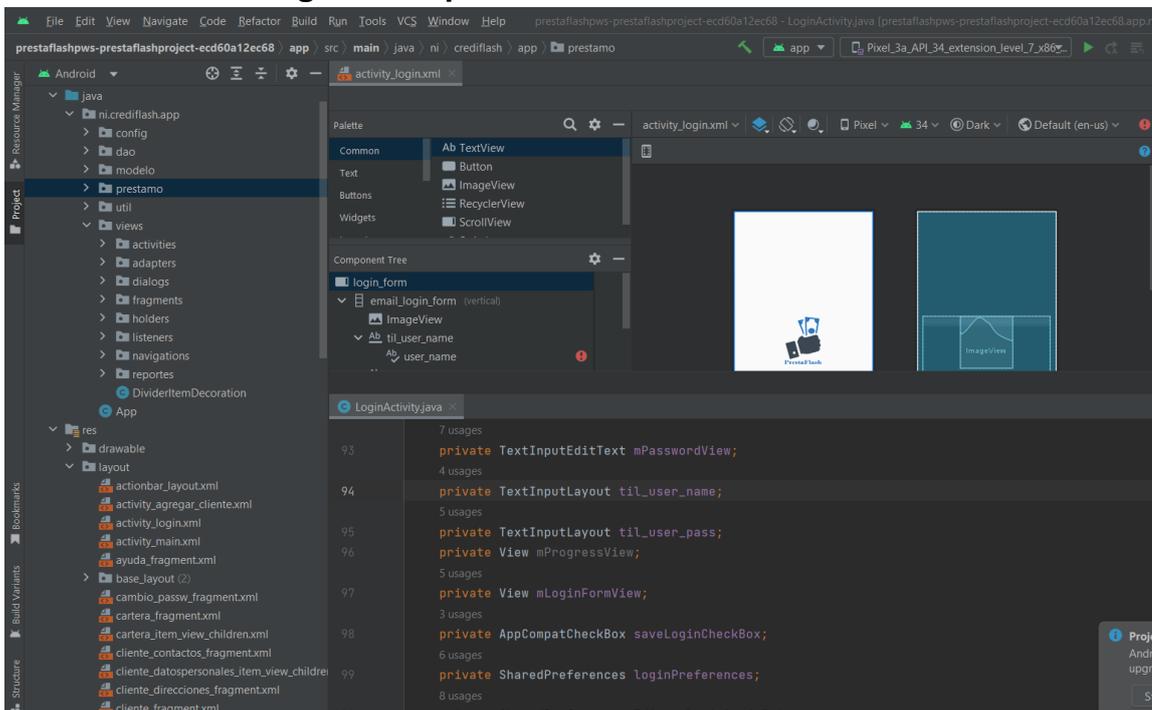


Ilustración 24 Estructura Código Fuente Aplicación Móvil Presta Flash
Fuente: Elaboración propia

Estructura del Código en el sistema Web

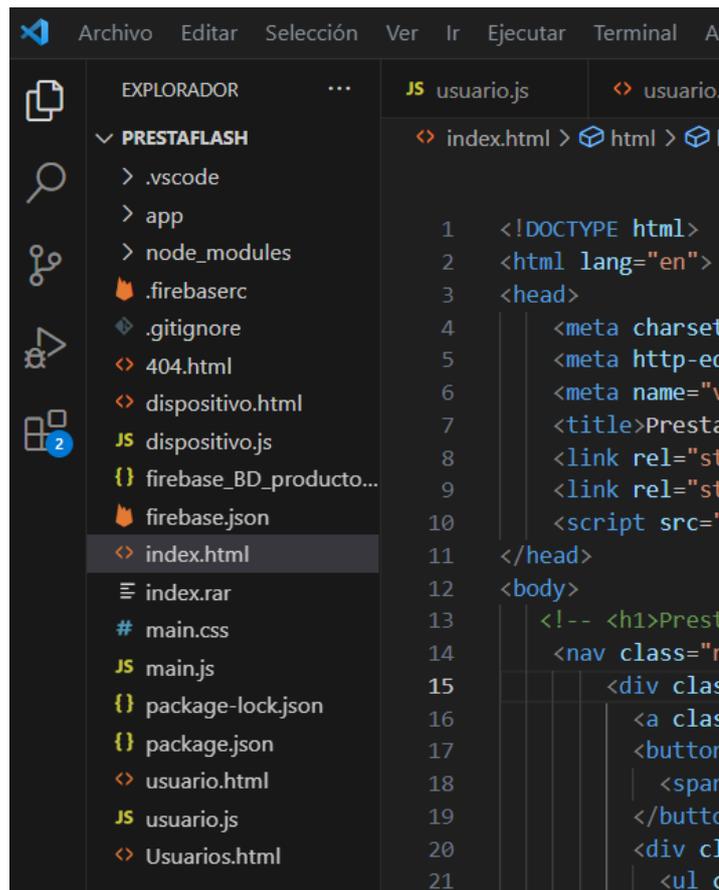


Ilustración 25 Estructura del Sistema Web
Fuente: Elaboración propia

Capa de Datos en la aplicación móvil

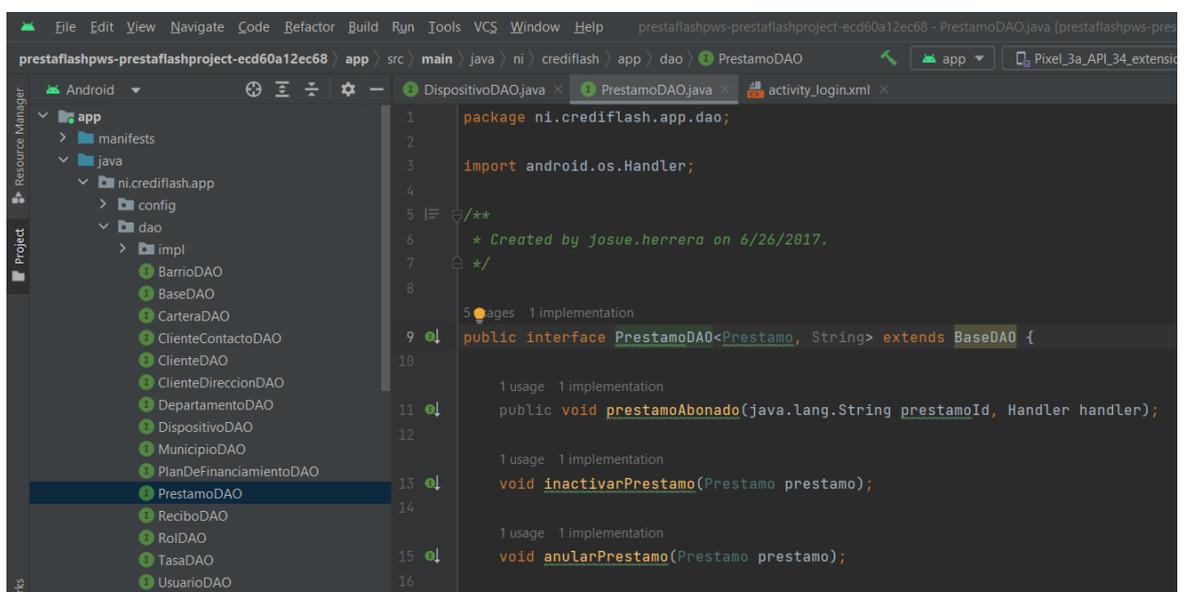


Ilustración 26 Capa de Datos aplicación Móvil
Fuente: Elaboración propia

Capa de presentación de la aplicación móvil

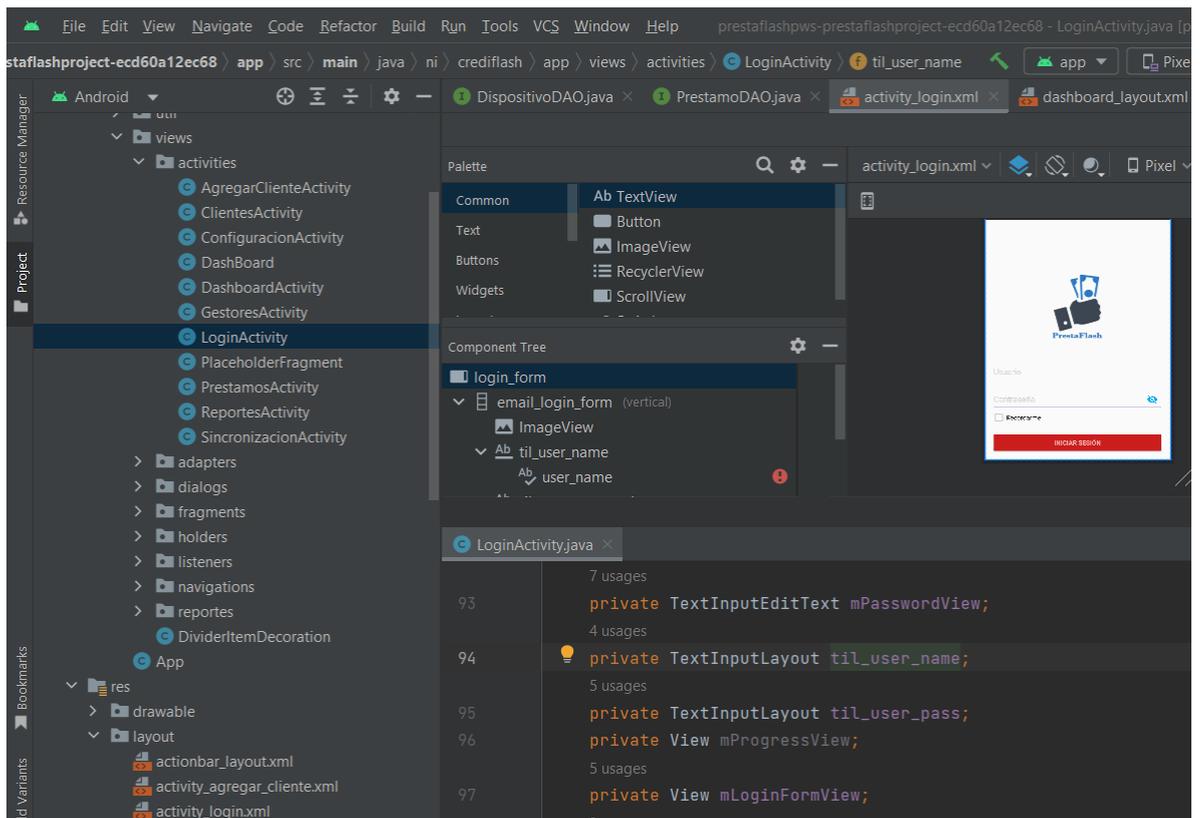


Ilustración 27 Capa de presentación de aplicación móvil
Fuente: Elaboración propia

CAPITULO II

Fase de Implementación

X. Fase de Implementación

10.1 Modelo de Desarrollo RAD

La metodología de desarrollo RAD, por sus siglas en español: Desarrollo rápido de Aplicaciones usada en el desarrollo de la aplicación móvil y el sistema web, nos permitió la creación de prototipos entregables, los cuales permitieron la retroalimentación de parte del usuario final.

10.2 Prototipos Aplicación Móvil

10.2.1 Prototipo 1

En este primer prototipo se crearon las siguientes implementaciones:

- Interfaz de Login para la aplicación móvil, con validaciones de usuario y contraseñas.
- Interfaz de Inicio de la aplicación móvil, a mostrarse luego de iniciar sesión.
- Interfaz principal de opción Configuración. Menú lateral izquierdo con opciones de Agregar Dispositivos, Agregar Cartera.
- Registro y modificación de Dispositivos que pueden acceder a la aplicación.
- Registro, modificación y baja de Cartera de clientes.
- Interfaz principal de opción Gestores. Menú lateral izquierdo con opciones de Agregar, editar y cambiar contraseña.

10.2.2 Prototipo 2

En el prototipo 2 se crearon las siguientes mejoras en base a los comentarios y pruebas por el usuario realizadas al prototipo 1 al igual que las nuevas implementaciones:

- Se agregó la mejora de crear filtros de búsquedas a carteras y dispositivos registrados.
- Se agregó la mejora de asignación de carteras a gestores de cobro y colocación.
- Se agregó la opción en el menú lateral izquierdo de la interfaz principal Configuración: Agregar Plan de Financiamiento. Así mismo el registro, modificación de Plan de Financiamiento.
- Se agregó la opción en el menú lateral izquierdo de la interfaz principal Configuración: Agregar Tasa de Cambio. Así mismo el registro de Tasa de Cambio.
- Se agregó la opción Clientes de la interfaz principal. Se agregó interfaz principal de gestión de clientes.

- Se agregó la opción Agregar, modificar y dar de baja a clientes.

10.2.3 Prototipo 3

En el prototipo 3 se crearon las siguientes mejoras en base a los comentarios y pruebas por el usuario realizadas al prototipo 2 al igual que las nuevas implementaciones:

- Interfaz principal de opción Préstamo. Menú lateral izquierdo con opciones de Editar, Agregar recibo, Inactivar Préstamo, Estado de cuenta.
- Opción Agregar prestamos en la interfaz principal de préstamos.
- Se realizó la mejora de dar de baja a un plan de financiamiento.
- Se realizó mejora en la interfaz de cliente, filtro de clientes por carteras. Todas las carteras activas para los usuarios Admin o Gerente. Opción de filtrar clientes por cartera asignada al gestor de cobro.
- Opción Reportes en la interfaz principal.
- Se agregó reportes de Cartera Detalle, Cartera Consolidación, Cliente por vencimiento y resumen por rango de fechas.

10.2.4 Prototipo 4 y Final

En el prototipo 4 y final se realizaron las correcciones de errores y mejoras en el sistema en base a los comentarios del usuario final las cuales fueron:

- Se agregó el logotipo diseñado para la aplicación.
- Se realizó la configuración de dispositivos de impresora para la impresión de recibos.
- Se agregó la mejora para consulta del último recibo realizado a un préstamo por cliente.

10.3 Prototipos sistema web

10.3.1 Prototipo 1

En este primer prototipo se crearon las siguientes implementaciones:

- Página de Loguin para el sistema web, con validaciones de usuario y contraseñas.
- Interfaz de menú inicio, a mostrarse luego de iniciar sesión.

10.3.2 Prototipo 2

En el prototipo 2 se crearon las siguientes mejoras en base a los comentarios y pruebas por el usuario realizadas al prototipo 1 al igual que las nuevas implementaciones:

- Se agregó la página de equipos móviles.
- Se agregó la carga de los datos de los equipos móviles.
- Se agregó la opción de agregar equipos móviles.
- Se agregó la opción de dar de baja a equipos móviles.

10.3.3 Prototipo 3

En el prototipo 3 se crearon las siguientes mejoras en base a los comentarios y pruebas por el usuario realizadas al prototipo 2 al igual que las nuevas implementaciones:

- Se agregó la página de Usuarios.
- Se agregó la carga de los usuarios.
- Se agregó la opción de agregar usuarios.
- Se agregó la opción de dar de baja a usuarios.

10.3.4 Prototipo 4 y Final

En el prototipo 3 se crearon las siguientes mejoras en base a los comentarios y pruebas por el usuario realizadas al prototipo 3 al igual que las nuevas implementaciones:

- Se agregó la página de Reportes.
- Se agregaron las opciones de consultas de recuperaciones y colocaciones de cartera por rango de fecha.
- Se agregó la opción de exportar las consultas realizadas.
- Se agregó la página de Respaldos.
- Se agregaron las opciones para crear respaldos locales y respaldos en la nube.

10.4 Plan de Pruebas

Esta es la etapa en la cual mediante pruebas unitarias nos aseguramos del buen funcionamiento del sistema.

10.4.1 Pruebas unitarias sistema Web

La sección de pruebas utilizando DevTools tiene como objetivo evaluar el rendimiento, la integridad y la eficiencia de la aplicación web desarrollada con JavaScript y Firebase Realtime Database. A través del uso de las herramientas disponibles en DevTools, se busca obtener información detallada sobre el comportamiento de la aplicación durante su ejecución en el navegador.

Las pruebas se centran en aspectos clave de la aplicación, incluyendo la interacción con la base de datos en tiempo real proporcionada por Firebase, la eficacia de las consultas, la manipulación del DOM, y la identificación de posibles problemas de rendimiento o errores en la consola del navegador.

Al realizar estas pruebas con DevTools, se persigue validar la correcta funcionalidad de la aplicación, asegurarse de que responda eficientemente a las solicitudes del usuario, identificar posibles cuellos de botella o errores de implementación, y garantizar una experiencia de usuario óptima.

La utilización de DevTools como herramienta de prueba proporciona una visión profunda de la ejecución interna de la aplicación, permitiendo una evaluación exhaustiva de su desempeño y contribuyendo a la mejora continua del desarrollo.

Resultados de las pruebas realizadas (Capturas de Pantalla de las pruebas realizadas en la página Reportes)

Recuperaciones.

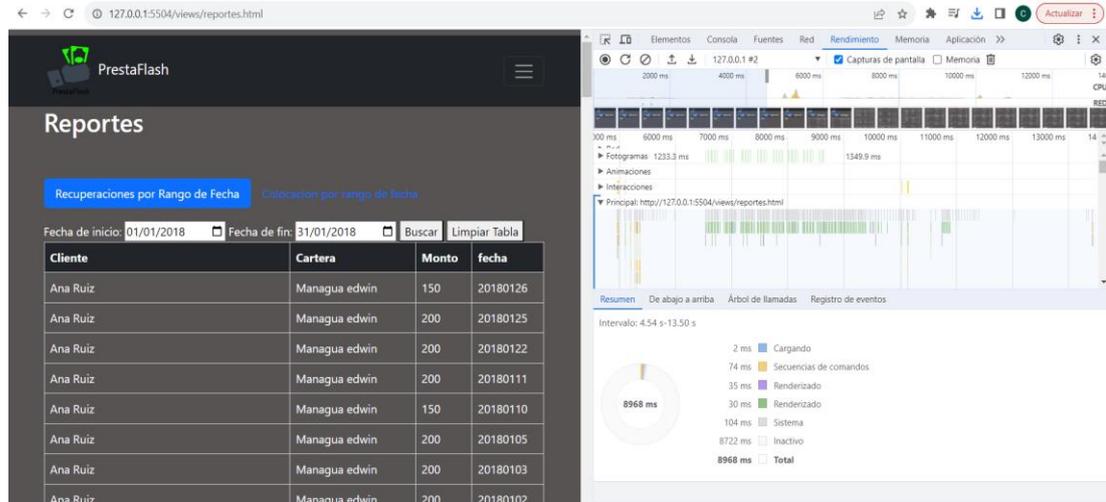


Ilustración 28 Resultado prueba unitaria Reporte Recuperaciones
Fuente: Elaboración propia

Colocaciones

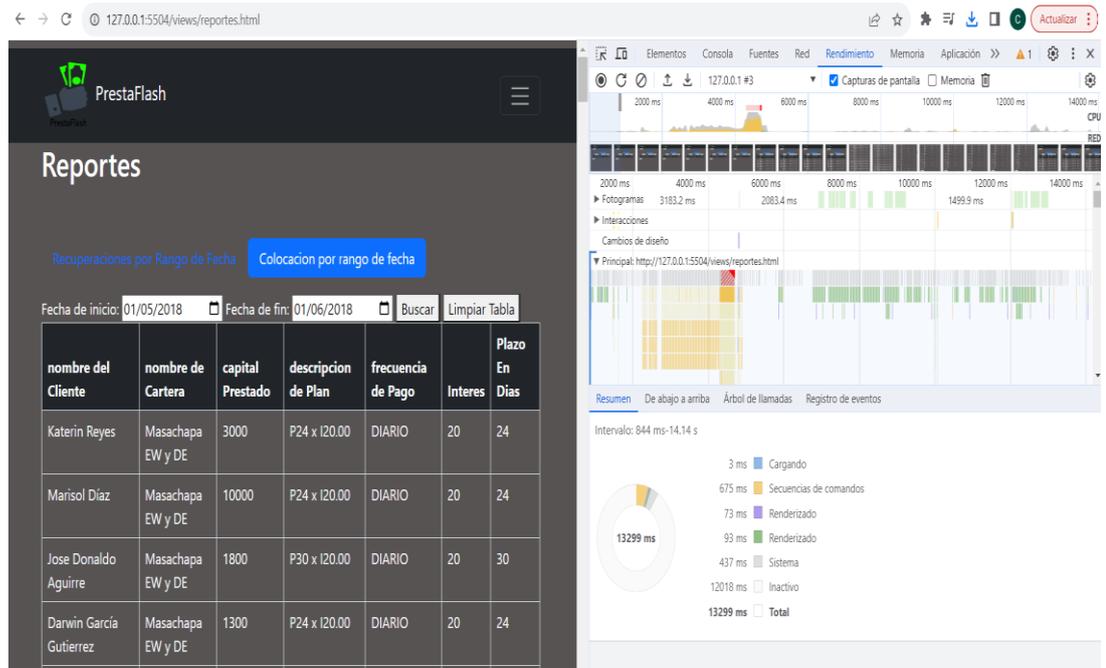


Ilustración 29 Resultado prueba unitaria Reporte Colocación
Fuente: Elaboración propia

Captura de pantalla de las pruebas realizadas en la página Equipos.

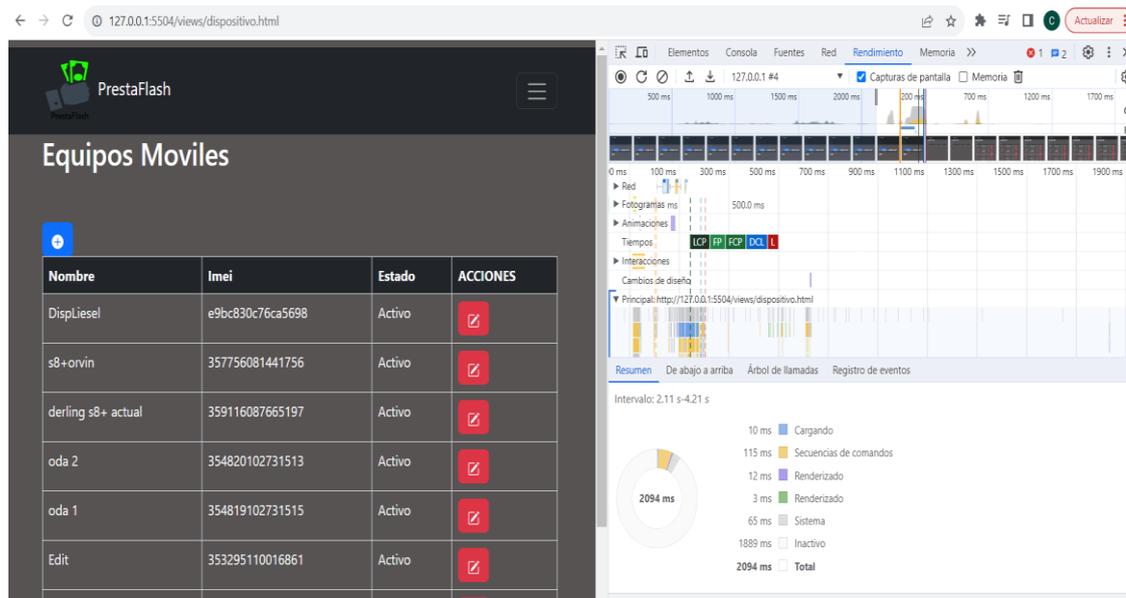


Ilustración 30 Resultado prueba unitaria Dispositivos Móviles
Fuente: Elaboración propia

Captura de pantalla de las pruebas realizadas en la página usuarios

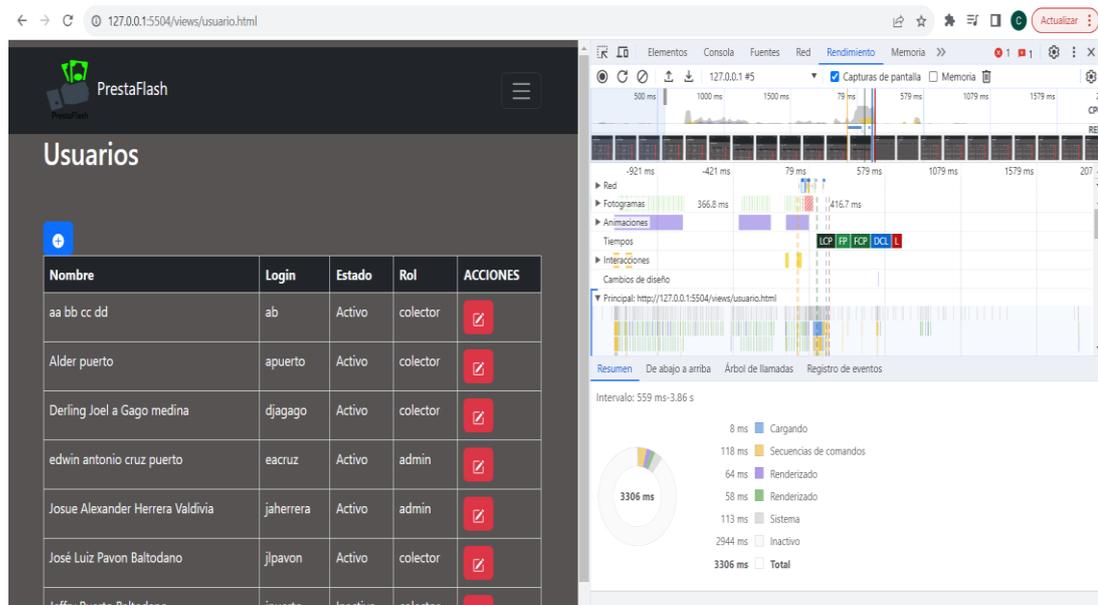


Ilustración 31 Resultado prueba Unitarias Usuarios
Fuente: Elaboración propia

10.4.2 Pruebas Unitarias Aplicación Móvil

Para la realización de pruebas unitarias en la aplicación móvil se utilizó Junit el cual nos permitió testear procesos fundamentales de la aplicación y así garantizar el funcionamiento correcto de estos.

Se importó la dependencia del framework de testing al gradle de la app.

```
// Dependencies for local unit tests
testImplementation 'junit:junit:' + rootProject.ext.junitVersion
testImplementation 'org.mockito:mockito-all:' + rootProject.ext.mockitoVersion
testImplementation 'org.hamcrest:hamcrest-all:' + rootProject.ext.hamcrestVersion

// Android Testing Support Library's runner and rules
androidTestImplementation 'com.android.support.test:runner:' + rootProject.ext.runnerVersion
androidTestImplementation 'com.android.support.test:rules:' + rootProject.ext.rulesVersion
```

Ilustración 32 Dependencia de Junit agregada al gradle
Fuente: Elaboración propia

Generar tabla de Amortización de Pagos

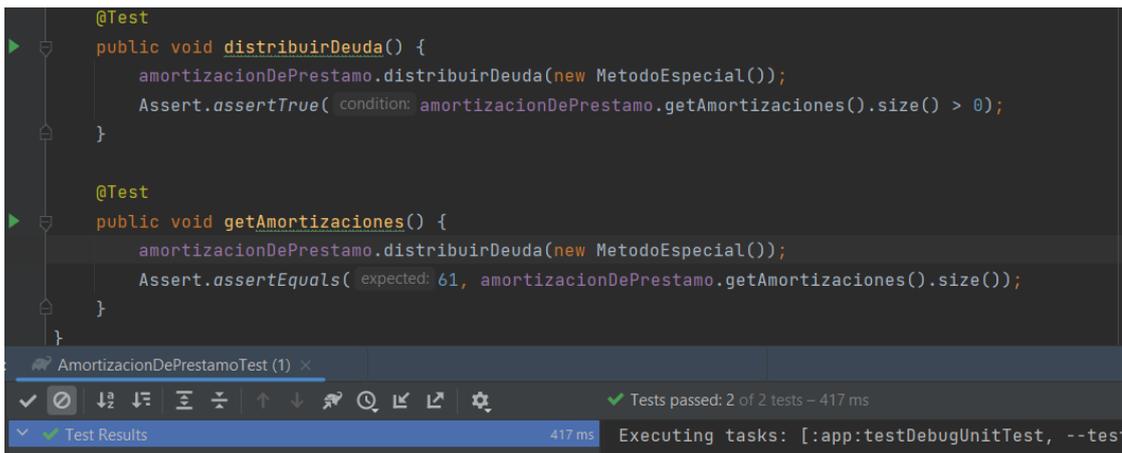
```
public class AmortizacionDePrestamoTest {

    5 usages
    AmortizacionDePrestamo amortizacionDePrestamo;

    @Before
    public void setUp() throws Exception {
        PlanDeFinanciamiento planDeFinanciamiento = new PlanDeFinanciamiento();
        planDeFinanciamiento.setPlazoEnDias(60);
        planDeFinanciamiento.setInteresEfectivo(1.4);
        Prestamo prestamo = new Prestamo();
        prestamo.setCapitalPrestado(100000);
        prestamo.setFechaDesembolso(new Date().getTime());
        prestamo.setFrecuenciaDePago(FrecuenciaDePago.DIARIO);
        prestamo.setPlanDeFinanciamiento(planDeFinanciamiento);
        prestamo.setDiasDeGracia(0);
        amortizacionDePrestamo = new AmortizacionDePrestamo(prestamo);
    }
}
```

Ilustración 33 Creación de Prueba Unitaria Generar Amortización
Fuente: Elaboración propia

Resultado de prueba unitaria a métodos Distribuir Deuda la cual



```
@Test
public void distribuirDeuda() {
    amortizacionDePrestamo.distribuirDeuda(new MetodoEspecial());
    Assert.assertTrue( condition: amortizacionDePrestamo.getAmortizaciones().size() > 0);
}

@Test
public void getAmortizaciones() {
    amortizacionDePrestamo.distribuirDeuda(new MetodoEspecial());
    Assert.assertEquals( expected: 61, amortizacionDePrestamo.getAmortizaciones().size());
}
```

AmortizacionDePrestamoTest (1) x

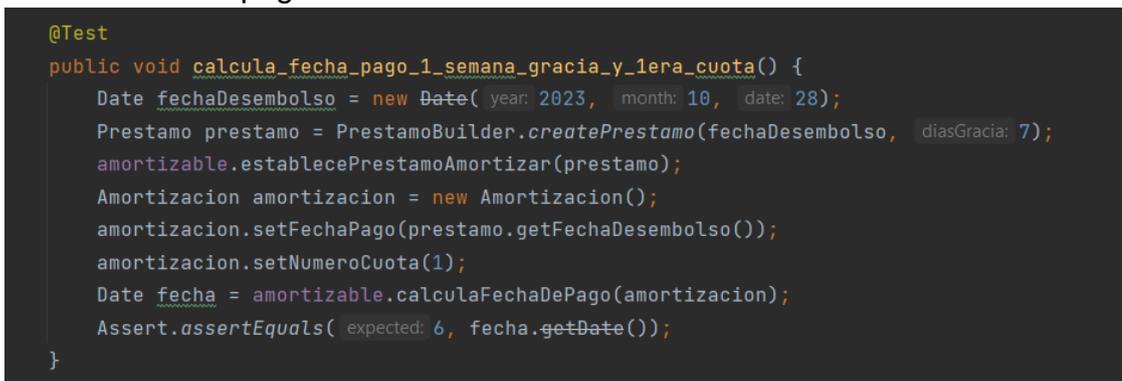
✓ Tests passed: 2 of 2 tests - 417 ms

Test Results 417 ms Executing tasks: [:app:testDebugUnitTest, --tes

Ilustración 34 Prueba Unitaria Generar Amortización
Fuente: Elaboración propia

Generar tabla de Amortización de Pagos, testeando fechas de pago con días de gracia.

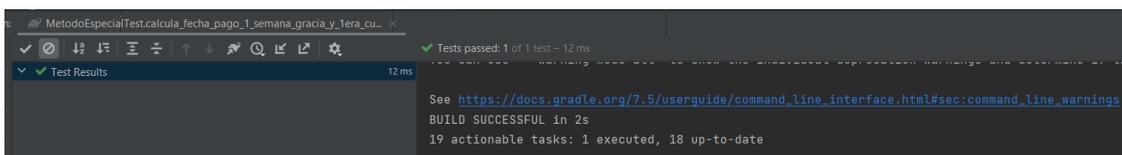
Con esta prueba confirmamos que las fechas calendarizadas en la tabla de amortización de pagos sean las correctas.



```
@Test
public void calcula_fecha_pago_1_semana_gracia_y_1era_cuota() {
    Date fechaDesembolso = new Date( year: 2023, month: 10, date: 28);
    Prestamo prestamo = PrestamoBuilder.createPrestamo(fechaDesembolso, diasGracia: 7);
    amortizable.establishPrestamoAmortizar(prestamo);
    Amortizacion amortizacion = new Amortizacion();
    amortizacion.setFechaPago(prestamo.getFechaDesembolso());
    amortizacion.setNumeroCuota(1);
    Date fecha = amortizable.calculaFechaDePago(amortizacion);
    Assert.assertEquals( expected: 6, fecha.getDate());
}
```

Ilustración 35 Prueba Unitaria Generar fechas de pago.
Fuente: Elaboración propia

Resultado de prueba unitaria a método Generar tabla de Amortización de Pagos, testeando fechas de pago con días de gracia.



MetodoEspecialTest.calcula_fecha_pago_1_semana_gracia_y_1era_cuota... x

✓ Tests passed: 1 of 1 test - 12 ms

Test Results 12 ms

See https://docs.gradle.org/7.5/userguide/command_line_interface.html#sec:command_line_warnings

BUILD SUCCESSFUL in 2s

19 actionable tasks: 1 executed, 18 up-to-date

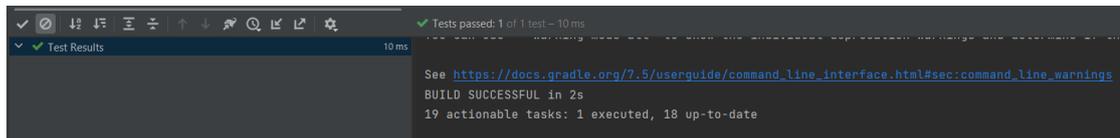
Ilustración 36 Resultado de Prueba Unitaria Generara Fechas de Pago con días de Gracia
Fuente: Elaboración propia

Generar tabla de Amortización de Pagos, testeando fechas de pago con 0 días de gracia.

```
@Test
public void calcula_fecha_pago_sin_dias_gracia_y_1era_cuota() {
    Date fechaDesembolso = new Date( year: 2023, month: 10, date: 28);
    Prestamo prestamo = PrestamoBuilder.createPrestamo(fechaDesembolso, diasGracia: 0);
    amortizable.establecePrestamoAmortizar(prestamo);
    Amortizacion amortizacion = new Amortizacion();
    amortizacion.setFechaPago(prestamo.getFechaDesembolso());
    amortizacion.setNumeroCuota(1);
    Date fecha = amortizable.calculaFechaDePago(amortizacion);
    Assert.assertEquals( expected: 29, fecha.getDate());
}
```

Ilustración 37 Prueba Unitaria Generara Fechas de Pago con 0 días de Gracia
Fuente: Elaboración propia

Resultado de prueba unitaria a método Generar tabla de Amortización de Pagos, testeando fechas de pago con 0 días de gracia.



The screenshot shows a test results window with a green checkmark and the text "Tests passed: 1 of 1 test - 10 ms". Below this, there is a link to the Gradle documentation: "See https://docs.gradle.org/7.5/userguide/command_line_interface.html#sec:command_line_warnings". The main text in the window reads: "BUILD SUCCESSFUL in 2s" and "19 actionable tasks: 1 executed, 18 up-to-date".

Ilustración 38 Resultado de Prueba Unitaria Generara Fechas de Pago con 0 días de Gracia
Fuente: Elaboración propia

10.4.3 Conclusiones plan de pruebas

Las pruebas realizadas utilizando DevTools en nuestra aplicación web desarrollada con JavaScript y Firebase Realtime Database arrojaron resultados positivos. Durante la evaluación, se verificó la correcta funcionalidad de la aplicación, se observó un rendimiento eficiente en la interacción con la base de datos en tiempo real proporcionada por Firebase, y se identificaron y corrigieron posibles problemas de rendimiento y errores en la consola del navegador.

Estos resultados indican que la aplicación se comporta según lo previsto, respondiendo de manera eficiente a las solicitudes del usuario y proporcionando una experiencia de usuario óptima. La exitosa ejecución de las pruebas valida la integridad y calidad de nuestro desarrollo, destacando el buen funcionamiento de la aplicación en condiciones diversas.

Este proceso de pruebas con DevTools ha sido fundamental para garantizar la fiabilidad y el rendimiento de la aplicación, brindando confianza en su implementación y preparándola para su despliegue en un entorno de producción. En resumen, la utilización de DevTools como herramienta de prueba proporciona una visión profunda de la ejecución interna de la aplicación, permitiendo una evaluación exhaustiva de su desempeño y contribuyendo a la mejora continua del desarrollo.

XI. Conclusiones

A través del desarrollo y finalización de este proyecto podemos llegar a las siguientes conclusiones:

- 1- Se han cumplido los objetivos planteados.
 - a. Se recopiló la información necesaria para elaborar este proyecto de acuerdo con los requerimientos y necesidades del modelo de negocio.
 - b. el diseño y modelado del sistema, la documentación y la construcción del mismo se elaboraron mediante RAD (Rapid application development).

- 2- El desarrollo del sistema web gerencial y aplicación móvil para gestión de control de crédito y recuperación de cartera para la pyme presta flash es una herramienta fundamental en los procesos que realizan, va mejorar la eficiencia operativa, reducirá el riesgo crediticio y va facilitar la gestión de la cartera.

Después de realizar todas las actividades mencionadas, se concluye que el desarrollo de la aplicación móvil y el sistema web gerencial se realizó exitosamente y que implementando el sistema propuesto en la pyme PrestaFlash:

- Facilitará las actividades de recuperación de préstamos colocados, evitando la omisión de pagos y haciendo posible conocer diariamente los saldos de cada uno de ellos.

- Evitará problemas de rentabilidad para la pyme que deriven de la mora en la cartera de créditos, pues permite tomar decisiones para accionar efectivamente y disminuir los índices de mora y prevenir los vencimientos de créditos sin cancelar

XII. Recomendaciones

Para maximizar los beneficios del sistema web y aplicación móvil se recomienda lo siguiente:

- Adquirir un plan de pago en firebase.
Actualmente se cuenta con un plan gratis en firebase, este plan es suficiente para mantener la carga de datos y accesos actualmente. Adquirir un plan de pago aportará algunas funciones que beneficiaran la pyme; una de ellas es los respaldos automáticos.

- Desarrollar el módulo de cálculo de comisiones para cada gestor.
El gestor de cobro recibe un 3% como comisión de todo lo recuperado en el mes.

- Adquirir un dominio y hosting personalizado
Una de las principales razones para adquirir un dominio personalizado, radica en la memorización del nombre del dominio. Es decir, un nombre de dominio.com, es más fácil de recordar que un subdominio.dominio.com, además de que obviamente es más presentables y da una mejor imagen al negocio.

XIII. Bibliografía

J. Cuello, J. Venotte(2014). Diseñando Apps para móviles. CreateSpace Independent Publishing Platform.

Domiguez, G. (2019). Curso de Programación Android con Java <https://docplayer.es/66403425-Presentacion-programacion-android-con-java-que-es-una-plataforma-de-software.html>

2018, 8 de octubre, XATAKA ANDROID. Fecha de consulta: octubre 12, 2018, Sitio web: <https://www.xatakandroid.com/programacion-android/como-instalar-el-android-sdk-y-para-que-nos-sirve>.

2021, 24 de mayo, stackpath, Sitio web: <https://www.stackpath.com/edge-academy/what-is-a-web-application/>

2017, 24 de enero. Sgolive. Fecha de consulta: octubre 12, 2018, Sitio web: <http://www.sgoliver.net/blog/firebase-android-base-datos-tiempo-real-1/>

Zoiner Tejada (2022, 25 de julio). Datos no relacionales y NoSql. <https://learn.microsoft.com/es-es/azure/architecture/data-guide/big-data/non-relational-data>

Chrome for Developers. (s. f.). Chrome for Developers. <https://developer.chrome.com/docs/devtools/>

Carrillo, P. Oliva,K. Herramientas para el desarrollo rápido de aplicaciones web. <http://metodologiarad.weebly.com/>

www.ionos.es, 2019, marzo 14, sitio web <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/el-papel-del-unit-test-en-el-desarrollo-de-software/>

J. Rumbaugh, J. G. (2000). El lenguaje Unificado de Modelo. Manual de referencia. Madrid, España: Pearson Educación, S.A.

Schmuller, J. (1999). Aprendiendo UML en24 Horas. México: Pearson Educación Latinoamérica.

J. Rumbaugh, J. G. (2000). El lenguaje Unificado de Modelo. Manual de Referencia. Madrid, España: Pearson Educación, S.S.

Chacón, J.C. (01 de 03 de 2006) Universidad de San Carlos de Guatemala. Obtenido de Biblioteca USAC.

https://biblioteca.usac.edu.gt/tesis/08/08_0308_CS.pdf

Hernández, M.L. (2018) Apuntes de Análisis y Diseño Orientado a Objeto. Mexico: Universidad Autónoma del Estado de Hidalgo.

XIV. Anexos

Anexo 1: Manuales de Usuario App PrestaFlash y Sistema Web. Anexado en el CD entregado.

Anexo 2: Tarjeta de Pagos por créditos de cliente

13

DIA		MES		AÑO	
4	11	17			

C\$ 6000

Juana Escobar

4	1	2	3	4
5	6	7	8	
	10	11	12	
13	14	15	16	
17	18	19	20	
21	22	23	24	
25	26	27	28	
29	30			

DIARIO: _____ 200

FIRMA: _____

Anexo 3: Archivos Excel de control de carteras

RUTA DERLING										
	SÁBADO			DOMINGO			LUNES			
CLIENTE	SALDO	ABONO	PRESTAMO	SALDO	ABONO	PRESTAMO	SALDO	ABONO	PRESTAMO	
42. LOURDES ESPINOZA G	C\$ 1,500.00			C\$ 1,500.00			C\$ 1,000.00			
43. LUIS C. ESPINOZA	C\$ 2,600.00	C\$ 2,600.00		C\$ -			C\$ -			
44. OSCAR M. CHABARRIA	C\$ 4,750.00	C\$ 500.00		C\$ 4,250.00		C\$ 250.00	C\$ 4,000.00			
45. JAIME CAJINA	C\$ 14,250.00	C\$ 1,500.00		C\$ 12,750.00		C\$ 750.00	C\$ 12,000.00			
46. MARIA RAMIRES	C\$ 3,000.00	C\$ 500.00		C\$ 2,500.00		C\$ 500.00	C\$ 2,000.00	C\$ 500.00		
47. MILEYDI RODRIGUES	C\$ 100.00	C\$ -		C\$ 100.00		C\$ 100.00	C\$ -			
48. WILDER A. DIAS	C\$ 2,200.00	C\$ 800.00		C\$ 1,400.00		C\$ 200.00	C\$ 1,200.00			
49. YESSICA SOLIS	C\$ 2,650.00	C\$ 400.00		C\$ 2,250.00		C\$ 200.00	C\$ 2,050.00			
50. MIRIAM HERNANDEZ	C\$ 2,100.00	C\$ 300.00		C\$ 1,800.00			C\$ 1,800.00	C\$ 300.00		
51. MARGARITA SEVILLA	C\$ -			C\$ -			C\$ -			
52. VIRGINIA SANTANA	C\$ -			C\$ -			C\$ -			
53. LEONEL RODRIGUES	C\$ -			C\$ -			C\$ -			
54. SANTOS VERONICA R.	C\$ -			C\$ -			C\$ -			
55. REYNA SOLIS SANCHES	C\$ -			C\$ -			C\$ -			
56. MARTHA ZAMORA	C\$ -			C\$ -			C\$ -			
57. MARIZOL DIAS BERMUDEZ	C\$ -			C\$ -			C\$ -			
58. GLADYS PALMA	C\$ -			C\$ -			C\$ -			
59. MARIO MOYA	C\$ -			C\$ -			C\$ -			
60	C\$ -			C\$ -			C\$ -			
61	C\$ -			C\$ -			C\$ -			
62	C\$ -			C\$ -			C\$ -			
63	C\$ -			C\$ -			C\$ -			
64	C\$ -			C\$ -			C\$ -			
65	C\$ -			C\$ -			C\$ -			
66	C\$ -			C\$ -			C\$ -			
67	C\$ -			C\$ -			C\$ -			
68	C\$ -			C\$ -			C\$ -			
69	C\$ -			C\$ -			C\$ -			

