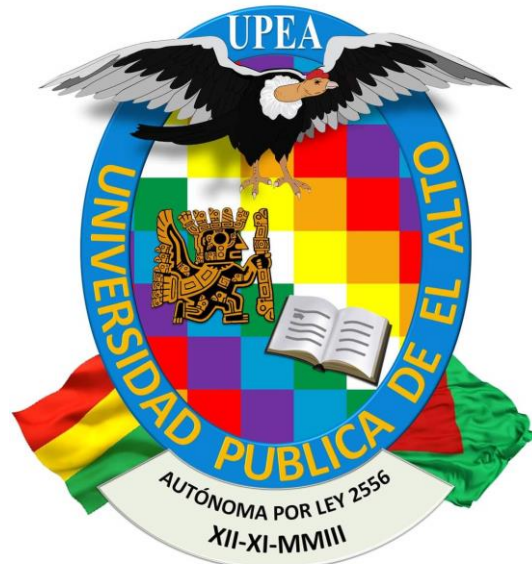


UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

“SISTEMA DE ADMINISTRACIÓN WEB PARA PRÉSTAMOS Caso: CAJA POPULAR BOLIVIANA SRL”

Para optar al título de Licenciatura en Ingeniería de Sistemas
Mención: INFORMÁTICA Y COMUNICACIONES

Postulante: Reynaldo Daniel Yuri Rada Jimenez
Tutor Metodológico: Ing. Marisol Arguedas Balladares
Tutor Revisor: Ing. Yolanda Escobar Mancilla
Tutor Especialista: Ing. Freddy Salgueiro Trujillo

EL ALTO – BOLIVIA
2020

DEDICATORIA

Dedico este trabajo a mis padres por el apoyo brindado a lo largo de toda mi vida ellos son un importante pilar de inspiración y superación, Por demostrarme siempre su apoyo y cariño incondicional

AGRADECIMIENTOS

Quiero agradecer primeramente a Dios por ser siempre la guía en mi vida, por darme sabiduría y fortaleza.

Agradecer a mi tutora revisora Ing. Yolanda Escobar Mancilla, por brindarme su conocimiento, experiencia, apoyo y confianza incondicional, el tiempo dedicado y por su paciencia, que me ayudaron a culminar el presente proyecto. Agradecer a mi tutor metodológico Ing. Marisol Arguedas Balladares, por su tiempo, comprensión, paciencia y orientación con sus conocimientos en la realización de este proyecto

Agradecer a mi tutor especialista Ing. Freddy Salgueiro Trujillo, por su apoyo desinteresado, y su conocimiento compartido que me ayudo a construir este proyecto.

Agradecer a la Ing. Maricel Yarari Mamani, por su ayuda tiempo, paciencia y orientación con sus conocimientos en la realización de este proyecto.

A mis padres, hermanos y familiares le doy gracias por el apoyo amor y comprensión que siempre me han brindado.

RESUMEN

El presente proyecto se desarrolla para dar conformidad a los requisitos exigidos por la Universidad Pública de El Alto para dar fin a la carrera de Ingeniería de Sistemas mediante el proceso de titulación. El proyecto lleva por nombre “Sistema de administración web para préstamos para el caso: CAJA POPULAR BOLIVIANA SRL”.

EL desarrollo del proyecto trata de brindar un apoyo al proceso de solicitudes de préstamos realizadas por el personal de la institución, mismo que componen las tareas como, recibir e ingresar al prestatario en el sistema, registrar datos personales, apoyo en la verificación de requisitos, registro del pago y emisión de comprobante desembolso.

Para el desarrollo del proyecto se utilizó el framework Laravel que se basa en el modelo MVC (modelo, vista, controlador) y herramientas de desarrollo de aplicaciones Web como ser PHP con la utilización de la metodología RUP (*Rational Unified Process*). En cuanto a los aspectos visuales, se utilizó Bootstrap.

Así mismo, para determinar la calidad de sistema Web desarrollado, se hace uso de los factores de calidad ISO 9126.

Finalmente se realiza el análisis de costos utilizando COCOMO II

INDICE DE CONTENIDO

CAPITULO I		Pág.
1	MARCO PRELIMINAR	1
1.1	INTRODUCCIÓN.....	1
1.2	ANTECEDENTES	2
1.2.1	Institucionales.....	2
1.2.2	Antecedentes académicos	2
1.3	PLANTEAMIENTO DEL PROBLEMA.....	4
1.3.1	Problema Principal	4
1.3.2	Problemas Secundarios	4
1.4	OBJETIVOS	5
1.4.1	Objetivo General.....	5
1.4.2	Objetivo Específicos	5
1.5	JUSTIFICACIONES.....	6
1.5.1	Técnica.....	6
1.5.2	Económica.....	6
1.5.3	Social.....	7
1.6	METODOLOGÍA.....	7
1.6.1	Proceso Racional Unificado (<i>Rational Unified Process</i>).....	7
1.6.2	Método de Ingeniería.....	8
1.7	HERRAMIENTAS	10
1.8	LÍMITES Y ALCANCES	11
1.8.1	Límites.....	11
1.8.2	Alcances.....	11
1.9	APORTES	12
CAPITULO II		
2	MARCO TEÓRICO	13
2.1	CAJA POPULAR BOLIVIANA SRL.....	13
2.2	TEORÍA SOBRE FINANZAS	14

2.2.1	Conceptos clave finanzas.....	15
2.2.2	Cartera	17
2.2.3	Tipos Amortización financiera	17
2.2.4	Tipos de préstamos	19
2.2.5	Requisitos para préstamo.....	20
2.2.6	Ente regulatorio	21
2.3	INTRANET	22
2.3.1	Elementos de la intranet.....	23
2.3.2	Ventajas	23
2.3.3	Desventajas.....	24
2.4	INGENIERÍA DE REQUERIMIENTOS.....	24
2.4.1	La ingeniería de requerimientos y sus principales actividades.....	26
2.4.2	Técnicas utilizadas en la ingeniería de requerimientos	28
2.5	<i>RATIONAL UNIFIED PROCESS (RUP)</i>	30
2.5.1	Principios de desarrollo	31
2.5.2	Ciclo de vida.....	32
2.5.3	Principales características.....	33
2.5.4	Fases.....	34
2.5.5	Dinámica de RUP	35
2.5.6	Artefactos	36
2.6	<i>UNIFIED MODELING LANGUAGE (UML)</i>	38
2.6.1	ANÁLISIS	39
2.6.2	DISEÑO.....	44
2.7	ESTANDAR DE CALIDAD ISO 9126	45
2.8	MODELO CONSTRUCTIVO DE COSTOS (COCOMO II)	49
2.8.1	COCOMO II.....	49
CAPITULO III		
3	MARCO APLICATIVO.....	58
3.1	PLAN DE DESARROLLO	58
3.2	ANALISIS DE LA SITUACIÓN ACTUAL DEL SISTEMA.....	59

3.2.1	Requisitos funcionales y no funcionales.....	70
3.3	ANÁLISIS DEL SISTEMA FUTURO	71
3.3.1	Descripción de funciones	71
3.3.2	Diagrama de casos de uso.....	72
3.3.3	Diagrama de secuencias.....	82
3.3.4	Diagramas de actividades	87
3.3.5	Diagramas de clases	93
3.3.6	Diagrama de base de datos	94
3.4	DISEÑO EL SISTEMA.....	95
3.4.1	Diagrama de navegación.....	95
3.4.2	Diagrama de interfaz	98
CAPITULO IV		
4	CALIDAD, PRUEBAS Y SEGURIDAD.....	109
4.1	MÉTRICAS DE CALIDAD.....	109
4.2	PRUEBA DEL SISTEMA	115
4.2.1	Pruebas de caja negra	115
4.3	SEGURIDAD	117
4.3.1	Seguridad a nivel de base de datos	117
4.3.2	Seguridad a nivel de aplicación.....	117
CAPITULO V		
5	ANÁLISIS DE COSTOS.....	119
5.1	CALCULO DE COSTOS.....	119
CAPITULO VI		
6	CONCLUSIONES Y RECOMENDACIONES.....	122
6.1	CONCLUSIONES.....	122
6.2	RECOMENDACIONES	122
BIBLIOGRAFÍA		124
ANEXOS		

ANEXO A. ARBOL DE PROBLEMAS
ANEXO B. ARBOL DE OBJETIVOS
ANEXO C. COMPROBANTE DE PAGO
ANEXO D. PLAN DE PAGOS
ANEXO E. MANUAL DE USUARIO
ANEXO F. MANUAL TÉCNICO
ANEXO G. AVALES

INDICE DE FIGURAS

	Pág.
Figura 2.1 Ciclo de vida de RUP	8
Figura 2.1 Organigrama de la empresa	13
Figura 2.2 Distribución del mercado de software actual y futuro.....	50
Figura 3.1 Organigrama de la empresa	60
Figura 3.2 Diagrama caso de uso cliente.....	60
Figura 3.3 Diagrama de caso de uso de plataforma	61
Figura 3.4 Diagrama de caso de uso de encargado operativo.....	61
Figura 3.5 Diagrama de caso de uso de encargado operativo.....	62
Figura 3.6 Diagrama de caso de uso de gerente general.	62
Figura 3.7 Diagrama de caso de uso de la comisión evaluadora.....	63
Figura 3.8 Diagrama de caso de uso de contabilidad.	63
Figura 3.9 Diagrama de caso de uso de cajero.....	64
Figura 3.10 Diagrama de secuencia Solicitud de préstamo	64
Figura 3.11 Diagrama de secuencia Revisión de requisitos	65
Figura 3.12 Diagrama de secuencia Plan de pagos y desembolso.....	66
Figura 3.13 Diagrama de secuencia Programar desembolso	67
Figura 3.14 Diagrama de secuencia Elaboración de contrato.....	67
Figura 3.15 Diagrama de secuencia Inicio de trámites legales	68
Figura 3.16 Diagrama de secuencia Votación.....	68
Figura 3.17 Diagrama de secuencia Informes de pagos.....	69
Figura 3.18 Diagrama de secuencia Genera informes.....	69
Figura 3.19 Diagrama de caso de uso solicitud de información y entrega de requisitos.....	72
Figura 3.20 Diagrama de caso de uso registro de solicitud	73
Figura 3.21 Diagrama de casos de reporte pagos y cartera vencida.....	74
Figura 3.22 Diagrama de caso de uso verificación de documentación	75
Figura 3.23 Diagrama de caso de uso de programación de desembolso	76
Figura 3.24 Diagrama de casos de uso aprobación de prestamos	76

Figura 3.25 Diagrama de caso de uso Gerente general	77
Figura 3.26 Diagrama de casos de uso gestionar prestamos	78
Figura 3.27 Diagrama de caso de uso Administrador	79
Figura 3.28 Diagrama de caso de uso Cajero.....	80
Figura 3.29 Diagrama de caso de uso Administrador del sistema	81
Figura 3.30 Diagrama de secuencia Solicitud de préstamo	82
Figura 3.31 Diagrama de secuencia Verificación de requisitos.....	83
Figura 3.32 Diagrama de secuencia Plan de pagos.....	84
Figura 3.33 Diagrama de secuencia Programar desembolso	84
Figura 3.34 Diagrama de secuencia Generar reportes	85
Figura 3.35 Diagrama de secuencia Informe de pagos.....	86
Figura 3.36 Diagrama de secuencia Gestionar prestamos	86
Figura 3.37 Diagrama de actividad Solicitud de préstamo	87
Figura 3.38 Diagrama de actividad Verificación de requisitos.....	88
Figura 3.39 Diagrama de actividad Plan de pagos.....	89
Figura 3.40 Diagrama de actividad Programar desembolso	90
Figura 3.41 Diagrama de actividad Informe de pagos.....	91
Figura 3.42 Diagrama de actividad Generación de reportes e informes	92
Figura 3.43 Diagrama de actividad Gestionar prestamos	92
Figura 3.44 Diagrama de clases del sistema	93
Figura 3.45 Diagrama de base de datos del sistema	94
Figura 3.46 Diagrama navegacional de plataforma.....	95
Figura 3.47 Diagrama navegacional de encargado operativo	95
Figura 3.48 Diagrama navegacional de sub gerente.....	96
Figura 3.49 Diagrama de navegacional de gerente general	96
Figura 3.50 Diagrama navegacional de cajas	97
Figura 3.51 Diagrama navegacional del administrador	97
Figura 3.52 Interfaz de inicio de sesión.....	98
Figura 3.53 Interfaz principal de plataforma.....	98
Figura 3.54 Interfaz de lista de solicitudes plataforma	99
Figura 3.55 Interfaz registro de datos persona	99

Figura 3.56 Interfaz registro de datos de trabajo.....	100
Figura 3.57 Interfaz registro de solicitud	100
Figura 3.58 Interfaz verificación de requisitos	100
Figura 3.59 Interfaz lista de las operaciones que pueden ser refinanciadas.....	101
Figura 3.60 Interfaz mostrar los datos de la solicitud	101
Figura 3.61 Interfaz principal de encargado operativo	101
Figura 3.62 Interfaz lista de solicitudes	102
Figura 3.63 Interfaz de pagos cliente	102
Figura 3.64 Interfaz principal de sub gerente	102
Figura 3.65 Interfaz de solicitud para la aprobación de monto.....	103
Figura 3.66 Interfaz para ver los datos de solicitudes	103
Figura 3.67 Interfaz principal de cajero	103
Figura 3.68 Interfaz lista de clientes y pagos	104
Figura 3.69 Interfaz principal gerente general.....	104
Figura 3.70 Interfaz de los usuarios y sus roles	105
Figura 3.71 Interfaz de los productos.....	105
Figura 3.72 Interfaz de creación de productos y asignación de requisitos.....	106
Figura 3.73 Interfaz vista de documentos para los requisitos	106
Figura 3.74 Interfaz valor de interés para los productos	107
Figura 3.75 Interfaz de tipo de amortización	107
Figura 3.76 Interfaz tipo de monedas.....	107
Figura 3.77 Interfaz principal de administrador	108
Figura 3.78 Interfaz lista de usuarios	108
Figura 3.79 Interfaz creación de usuarios y asignación de roles.....	108
Figura 3.80 Interfaz de lista de roles	108
Figura 4.1 Prueba de caja negra del inicio de sesión.....	115
Figura 5.1 Herramienta: USC-COCOMO II.2000.4	119
Figura 5.2 Resultados de USC-COCOMO II.2000.4	120

INDICE DE TABLAS

	Pág.
Tabla 2.1 Tabla de amortización a cuota fija.....	19
Tabla 2.2 Comparación de internet, intranet y extranet	22
Tabla 2.3 Características de la Normas ISO-9126.....	46
Tabla 2.4 Valores de escala.....	54
Tabla 2.5 Atributos de costo para el modelo Post-Arquitectura	55
Tabla 2.6 Valores de atributos de costos	56
Tabla 3.1 Plan de desarrollo	58
Tabla 3.2 Descripción de funciones	71
Tabla 3.3 Especificación de los casos de uso Cliente.....	73
Tabla 3.4 Especificación de caso de uso registro solicitud	74
Tabla 3.5 Especificación de caso de uso reporte de pagos y cartera vencida	74
Tabla 3.6 Especificación de caso de uso verificación de documentación	75
Tabla 3.7 Diagrama de caso de uso Subgerente	76
Tabla 3.8 Especificación de caso de uso aprobación de un préstamo.....	77
Tabla 3.9 Especificación de caso de uso Gestionar préstamo.....	78
Tabla 3.10 Especificación de caso de uso gestionar usuarios	79
Tabla 3.11 Especificación de los casos de uso Generar reportes.....	80
Tabla 3.12 Especificación de casos de uso Registro de pagos	80
Tabla 3.13 Especificación de casos de uso Generar reportes de pagos	81
Tabla 4.1 Factores de ponderación.....	109
Tabla 4.2 Valores de ajuste de complejidad	110
Tabla 4.3 Escala de punto función	111
Tabla 4.4 Ajuste de preguntas	112
Tabla 4.5 Descripción de prueba de caja negra inicio sesión	116
Tabla 4.6 Prueba de caja negra registro usuario	116
Tabla 4.7 Descripción de prueba de caja negra de registro usuario	116
Tabla 4.8 Recomendaciones ISO 27002	117
Tabla 5.1 Datos introducidos en USC-COCOMO II.2000.4	120
Tabla 5.2 Parámetros de medida	120

CAPÍTULO I

MARCO PRELIMINAR

1 MARCO PRELIMINAR

1.1 INTRODUCCIÓN

Desde el inicio de la computación tuvo un impacto en la sociedad en general y en las empresas. Ya que sus aplicaciones optimizan todos los ámbitos, por su capacidad de almacenamiento y su manera de procesar la información, Tal es el caso de las instituciones que otorgan préstamos, hoy en día empresas que utilizan la tecnología para poder administrar, gestionar y ofrecer sus productos o servicios estas empresas pasan a formar parte de la nueva realidad tecnológica.

La tecnología de hoy en día va creciendo exponencialmente, mejorando su capacidad de almacenamiento y procesamiento, haciendo que operaciones complejas vayan reduciendo su tiempo y por eso todas las empresas requieren mejorar sus herramientas de trabajo para sus actividades cotidianas, tal es el caso de la CAJA POPULAR BOLIVIANA SRL por su manera de manejar la información semi automatizada utilizando programas de ofimática como Word y Excel para realizar sus cálculos y operaciones suelen tener errores en el manejo de su información y demora en las búsquedas. Este tipo de operaciones requiere de un control exhaustivo ocasionando pérdida de tiempo quedando en desventaja con otras instituciones, por es necesario crear herramientas que ayuden a administrar y controlar los préstamos para competir con mayor eficiencia con otras empresas.

El presente proyecto está abocado en la elaborar un sistema web que ayude con la administración de préstamos y gestión de pagos otorgando una nueva forma de manejo de la información en la Caja Popular Boliviana. Se definió distintos tipos de usuario como son plataforma, encargado operativo, sub gerente y caja para tener un manejo fácil y sistematizado en las actividades de un préstamo, con eso se logrará que la institución de un paso más hacia la modernización tecnológica implementando su propio sistema para la modernización tecnológica. Se realizando

el análisis, diseño y desarrollo de un prototipo con el propósito de mejorar la administración de préstamos y la gestión de pagos otorgando una nueva forma de administrar la información.

1.2 ANTECEDENTES

1.2.1 Institucionales

La CAJA POPULAR BOLIVIANA S.R.L. es una empresa fundada el 6 agosto 2010 por la Lic. Sonia Jimenez Calle la actividad principal que realiza la empresa es la concesión de créditos. La institución tiene 2 tipos de créditos:

- Créditos de libre disponibilidad.
- Créditos hipotecarios.

La institución maneja un sistema de amortización de pago a cuotas fijas (Sistemas Francés) que se calcula con una hoja Excel para cada plan de pagos, los comprobantes están hechos en Excel cada prestamos cuenta con una carpeta creada por un asesor de préstamos que realiza todo el proceso de papeleo y se encarga de llevar control de los pagos hasta el cierre del mismo.

1.2.2 Antecedentes académicos

Internacional

- Un primer trabajo que cabe destacar correspondete a la división de informática de argentina (2016) que titula Gestión integral de entidades financieras y casa de crédito. Es un programa de escritorio que permite una administración eficaz y segura del crédito de consumo. Efectivo y/o tarjeta de crédito. Administra el seguimiento de morosidad. Simplifica y agiliza el manejo del negocio

disminuyendo la labor de control y administración, obteniendo más tiempo para diseñar estrategias que optimicen la rentabilidad. El cual contiene los siguientes módulos Operaciones, Liquidaciones, Caja, Contabilidad, Cobranza, Facturación, Estadística, Morosidad, Administración, Tarjeta de recomendación y Manejo de adicionales. (informática, 2018)

- Por otra parte, que corresponde a Sistemas Paez (2016) que titula Prestacob PRO-LAN. Es un programa que gestión de pagos de cuotas mensuales (o de otros periodos) que una institución (Asociación, centro deportivo, academia, club, etc) cobra a sus afiliados, socios o clientes. Con este programa es posible realizar un seguimiento de los socios y llevar un control preciso de las cuotas y recibos emitidos, Permite los sistemas de amortización: simple, francés, alemán y americano. Registro de deudores morosos es un registro de deudores que permite llevar cuenta de las deudas y los cobros realizados. (Paez, 2018)
- El proyecto titulado ultimateCredito que corresponde al grupo empresarial especializado El Software para préstamos personales más Completo del mercado por su facilidad de uso y prestaciones, en esta versión 2015 que contiene, además de las funcionalidades de la versión (clientes), los módulos de pagos y deudas, créditos, liquidación de comisiones, intimaciones u escritos, control de pagos, estadísticas, compras. Este software está pensado como si usted mismo lo hubiera hecho. Todo lo que usted necesita para manejar su negocio de préstamos en un solo software. No necesita ninguna otra aplicación, centralizando así todas las operaciones en un único lugar. De esa manera usted posee información precisa y consistente en el momento que la necesita. Este software en versión cuenta con gestión de correo electrónico, módulo de recuperación de deudas y mensajes de texto como también reportes de datos en todos sus tipos. (eportal, 2018)

1.3 PLANTEAMIENTO DEL PROBLEMA

1.3.1 Problema Principal

Luego de realizar un análisis de los problemas, se ha diseñado un árbol de problemas (ver Anexo A) que permitió identificar la falta de una estrategia de comunicación en la gestión de préstamos genera retardo¹ en las solicitudes de préstamos y generando cuellos de botella en los pagos que se realizaran en el transcurso del día en la ciudad de El Alto, La CAJA POPULAR BOLIVIANA S.R.L es una micro financiera que en la actualidad utiliza tecnología semi automatizada², aletargando sus operaciones³.

¿Cómo se podría optimizar la administración de préstamos y mejorar la calidad del servicio en la Caja Popular Boliviana srl?

1.3.2 Problemas Secundarios

- Las solicitudes se verifican manualmente, prestándose a errores involuntarios de los empleados.
- La verificación de solicitudes de préstamos es lenta y lleva más tiempo de los necesario.
- El proceso que tienen para las solicitudes de los préstamos es poco eficiente ya que no tiene un seguimiento preciso de las fases en la que se encuentran las solicitudes ocasionado cuellos de botella en la atención lo que genera descontento y pérdida de clientes.
- Los pagos realizados con ayuda de las hojas de Excel no son tan seguros ya que pueden llegar a tener errores de humanos así recurrir al expediente lo cual genera la pérdida de tiempo en la institución.

¹ retardo. - demora en los cálculos y errores generando un retrasando en desembolso ocasionalmente.

² Semi automatizada. – El manejo de programas de ofimática como Word y Excel.

³ Operaciones. – cálculo de plan de pagos, errores de taipeo y búsquedas demorasas.

- Demora en el proceso de cálculo del plan de pagos para cada cliente se realiza en hojas Excel realizando la configuración del monto y tiempo del mismo provoca que un desembolso tardío.
- Para la verificación de estado de los pagos, cada cliente debe realizar la espera en la casa de préstamos lo que genera cuellos de botella en la atención determinados días del mes pérdida de tiempo un lapso de 15 a 20 minutos para los clientes y la institución.
- La información se encuentra de forma dispersas llevando así un demoroso proceso para generar reportes e informes.

1.4 OBJETIVOS

El objetivo del presente proyecto nace como respuesta a las necesidades de la institución, luego de un análisis a los problemas y su relación causa-efecto se realizó el diseño del árbol de objetivos. (ver anexo A.2)

1.4.1 Objetivo General

Desarrollar un portal web para que coadyuve a mejorar la administración en los procesos de cobro y administración de los préstamos en la CAJA POPULAR BOLIVIANA S.R.L. en la gestión de préstamos.

1.4.2 Objetivo Específicos

- Realizar el diagnóstico de la situación actual para poder conocer los procesos de la organización y sistematizarlo.
- Realizar el análisis del manejo de la información del sistema solucionando los problemas específicos y generar información confiable y oportuna.

- Diseñar y desarrollar un prototipo en que mejoren la otorgación de préstamos, control y seguimiento de pagos ofreciendo información oportuna que coadyube aún mejor control de la información y resolviendo las necesidades de la empresa.
- Realizar un análisis de la calidad para definir si el sistema es confiable.
- Realizar un análisis de costos para tener una estimación del valor del portal web.

1.5 JUSTIFICACIONES

1.5.1 Técnica

Se justifica técnicamente este proyecto porque se utiliza métodos, técnicas y herramienta que ayuden en el proceso de análisis, diseño y desarrollo, como es RUP y UML como metodologías de desarrollo. RUP no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de la organización.

1.5.2 Económica

Se justifica desde un punto de vista económico porque se disminuirá los costos gracias a la automatización de los procesos se optimizarán los tiempos del personal para las actividades de control y seguimiento de préstamos, disminuyendo los tiempos y obteniendo una mejor administración de préstamos y control de pagos. A la hora de interactuar con el usuario será de un fácil manejo evitando pérdidas económicas y el proyecto se desarrollará con tecnologías libres⁴.

⁴ Tecnologías libres. – las tecnologías libres no tiene ningún costo para el uso de la misma es el trabajo colaborativo de un conjunto de personas que ayudan en el creación, mantenimiento, actualización y mejoras de la misma.

1.5.3 Social

Se justifica socialmente porque se mejora el servicio para el cliente, a la vez que facilita el trabajo de los cargos de plataforma, cajas, encargado operativo y sub gerente. Aumentando la eficiencia de la institución dando más tiempo para mejorar la atención.

1.6 METODOLOGÍA

1.6.1 Proceso Racional Unificado (*Rational Unified Process*)

Es una metodología de desarrollo de software formal desarrollado por la empresa Rational Software, actualmente propiedad de IBM, El RUP es un conjunto de metodologías adaptables al contexto y necesidades de cada organización.

También se conoce por este nombre al software, también desarrollado por Rational, que incluye información entrelazada de diversos artefactos y descripciones de las diversas actividades. Está incluido en el Rational Method Composer (RMC), que permite la personalización de acuerdo con las necesidades. (wikipedia, 2018).

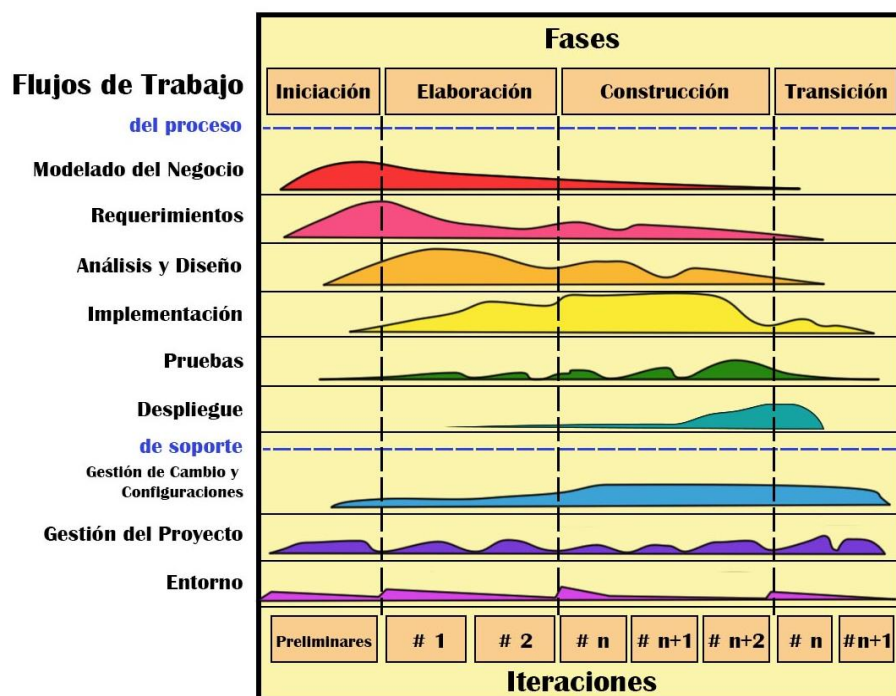
La metodología RUP (*Rational Unified Process*), como herramienta principal de modelado. Así mismo para implementar esta metodología, RUP divide el proceso en cuatro fases dentro de las cuales se realizan varias iteraciones en número variable y en las que se hace un mayor o menor hincapié en las actividades que se realizarán en el proyecto (Sommerville, 2011).

- En la **fase de inicio** las iteraciones hacen mayor énfasis en actividades de modelado del negocio y de requisitos.
- En la **fase de elaboración**, las iteraciones se orientan al desarrollo de la arquitectura, abarcan más los flujos de trabajo de requisitos, modelo de negocios

(refinamiento), análisis, diseño y una parte de implementación orientado a la arquitectura.

- En la **fase de desarrollo**, se lleva a cabo la construcción del producto por medio de una serie de iteraciones.
- En la **fase de cierre** se pretende garantizar que se tiene un producto preparado para su entrega a la comunidad de usuarios.

Figura 1.1 Ciclo de vida de RUP



Fuente: El proceso unificado de desarrollo de software (Booch, Jacobson & Rumbaugh, 2000)
ELABORACIÓN: PROPIA

1.6.2 Método de Ingeniería

El UML (lenguaje unificado de modelado) es un lenguaje gráfico que sirve para visualizar, especificar, construir y documentar un sistema, un estándar utilizado para describir un sistema se divide en dos tipos de diagramas los cuales son dos diagramas de estructura y diagramas de comportamiento

ANÁLISIS

- Diagrama de casos de uso, es un diagrama que muestra el comportamiento del sistema en diferentes escenarios.
- Diagrama de estados, utilizado para identificar cada una de las rutas o caminos que puede tomar un flujo de información luego de ejecutarse cada proceso.
- Diagrama de secuencia, muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada método de la clase.
- Diagrama de comunicación, modela las interacciones entre objetos o partes en términos de mensajes en secuencia. Los diagramas de comunicación representan una combinación de información tomada desde el diagrama de clases, secuencia, y diagrama de casos de uso describiendo tanto la estructura estática como el comportamiento dinámico de un sistema.
- Diagrama de tiempos, son una representación especial de interacción que se enfoca en el tiempo de los mensajes enviados entre objetos. Se pueden usar estos diagramas para mostrar restricciones detalladas sobre el tiempo, o para mostrar los cambios con líneas de vida respecto al tiempo.
- Diagrama global de interacciones, muestra una cierta vista sobre los aspectos dinámicos de los sistemas modelados. Aunque un diagrama global de las interacciones es una representación gráfica de una interacción, éste se distingue fuertemente de los diagramas de secuencia y de comunicación.
- Diagrama de clases, es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos.

DISEÑO

- Diagrama de componentes, representa cómo un sistema de software es dividido en componentes y muestra las dependencias entre estos componentes.
- Diagrama de objetos, utilizan un subconjunto de los elementos de un diagrama de clase. Los diagramas de objetos no muestran la multiplicidad ni los roles,

aunque su notación es similar a los diagramas de clase.

- Diagrama de estructura compuesta, muestra la estructura interna de una clase y las colaboraciones que esta estructura hace posibles.
- Diagrama de despliegue, se utiliza para modelar el hardware utilizado en las implementaciones de sistemas y las relaciones entre sus componentes.
- Diagrama de paquetes, muestra cómo un sistema está dividido en agrupaciones lógicas mostrando las dependencias entre esas agrupaciones. Dado que normalmente un paquete está pensado como un directorio, los diagramas de paquetes suministran una descomposición de la jerarquía lógica de un sistema.
- Diagrama de actividades, representa los flujos de trabajo paso a paso de negocio y operacionales de los componentes en un sistema.

1.7 HERRAMIENTAS

Existen diferentes tipos de necesidades en las plataformas web que han ido surgiendo debido a las tendencias y se crearon diferentes tipos de lenguajes de programación para la web se decidió elegir como lenguaje de desarrollo a PHP por las características que tiene y las ventajas que ofrece también porque tiene documentación que servirá como apoyo en desarrollo. (Sergio A Rosado, desarrollador web) con ayuda de un framework conocido como Laravel.

- **LARAVEL:** Es un framework de php de código abierto, es flexible, adaptable la información se maneja con ORM, la documentación es abundante aparte de poseer una comunidad amplia y activa lo que lo convierte en una poderosa herramienta de PHP.

De los diferentes tipos de gestores de base de datos se seleccionó a Mariadb en su versión libre muestra las ventajas más significativas, siendo la adecuada para trabajar junto con PHP. (Sergio A Rosado, desarrollador web)

1.8 LÍMITES Y ALCANCES

1.8.1 Límites

Bajo requerimiento del gerente general el presente trabajo se limita a la recolección de información de los procesos, seguimiento, control de pagos de préstamos de libre disponibilidad y no tomando en cuenta los préstamos hipotecarios.

1.8.2 Alcances

Los alcances del presente proyecto – expresados como módulos – se describen a continuación:

- Módulo de roles: los diferentes tipos de roles para gestionar y dar niveles de acceso a los usuarios que interactúan con el sistema la creación de los mismos asignación de roles y acceso al sistema.
- Módulo de aprobación: las fases por las que se obtiene el préstamo el registro de solicitud, verificación de requisitos, aprobación de monto o eliminación de la solicitud.
- Módulo de desembolso: la programación del desembolso de dinero pasando una nueva etapa de la solicitud.
- Módulo de seguimiento de pagos: se formar un historial para cada cliente para tomar la información como referencia institucional.
- Módulo de cálculo de mora: los pagos con retrasos tienen un monto extra para cancelar por el interés devengado.
- Módulo de búsqueda: para la búsqueda para la cancelación de pagos programados, adelantados y retrasados.
- Módulo de liquidación: para poner una calificación y guardar como antecedente para futuros créditos.
- Módulo de reportes (para ver la información pertinente a las solicitudes).

- Módulo de inversión (El módulo de inversión servirá para el control de ingresos a la empresa mostrando cuanto ganará por cada prestamos).

1.9 APORTES

Se generarán herramientas para facilitar la administración de los préstamos y gestión de pagos para la institución así facilitando la generación de estados de préstamos y pagos de operaciones. Se otorgar un control más preciso de su información y así facilitando el uso de la misma como referencia para futuras operaciones.

CAPÍTULO II

MARCO TEÓRICO

2 MARCO TEÓRICO

2.1 CAJA POPULAR BOLIVIANA SRL

La CAJA POPULAR BOLIVIANA SRL, está ubicada en avenida 6 de marzo n. ° 450 edificio: galería león piso: pb oficina: 1-a zona: 12 de octubre, es una institución dedica a otorgar créditos prendarios mediante préstamos de dinero con garantía de joyas de oro, objetos de oro, así como con garantía real hipotecaria de inmuebles, vehículos y demás bienes sujetos a registro, garantía prendaria de otra.

Hoy en día es una gran empresa y se encuentra en un desarrollo, sin embargo, junto con este crecimiento se ha generado la necesidad de contar con herramientas que ayuden a realizar control y seguimiento de los préstamos.

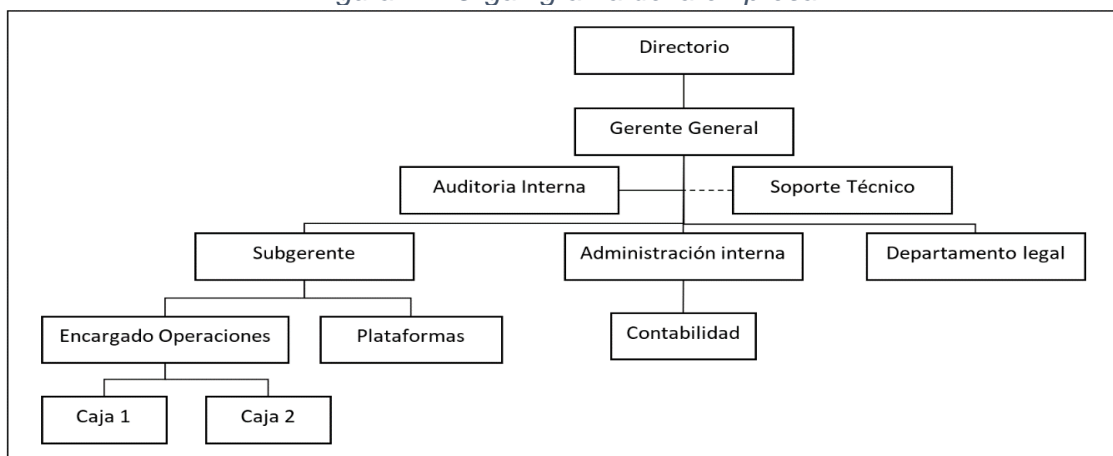
- **Misión**

Somos una empresa que ofrece a nuestros clientes los prestamos competitivos en el mercado y de manera ágiles, accesibles y seguros.

- **Visión**

Proporcionar siempre el mejor servicio para alcanzar rentabilidad permanencia y crecimiento con nuestros clientes.

Figura 2.1 Organigrama de la empresa



Fuente: (jimenez, 2012).

2.2 TEORÍA SOBRE FINANZAS

El estudio académico de las finanzas se divide principalmente en dos ramas, que reflejan las posiciones respectivas de aquel que necesita fondos o dinero para realizar una inversión, llamada finanzas corporativas, y de aquel que quiere invertir su dinero dándoselo a alguien que lo quiera usar para invertir, llamada valuación de activos. El área de finanzas corporativas estudia cómo le conviene más a un inversionista conseguir dinero, por ejemplo, si vendiendo acciones, pidiendo prestado a un banco o vendiendo deuda en el mercado.

El área de valuación de activos estudia cómo le conviene más a un inversionista invertir su dinero, por ejemplo, si comprando acciones, prestando/comprando deuda, o acumulado dinero en efectivo. (Gonzalez, 2018)

Estas dos ramas de las finanzas se dividen en otras más. Algunas de las áreas más populares dentro del estudio de las finanzas son: Intermediación Financiera, Finanzas Conductistas, Microestructura de los Mercados Financieros, Desarrollo Financiero, Finanzas Internacionales, y Finanzas de Consumidor. Una disciplina recientemente creada son las neuro finanzas, rama de la neuroeconomía, encargada del estudio de los sesgos relacionados con el manejo de la economía.

Las finanzas personales son la aplicación de las finanzas y sus principios de una persona o familia en su deseo de realizar sus actividades con la mejor distribución de dinero para ello. Así, deben reconocer cómo ocupar sus ingresos en educación, salud, alimentación, vestimenta, seguros, lujos, transporte, etc. Se deben tener en cuenta los ingresos, los gastos, los ahorros y siempre estableciendo los riesgos y los eventos futuros. Parte de las finanzas personales son los cheques, las cuentas de ahorro, las tarjetas de crédito, los préstamos, las inversiones en el mercado de valores, los planes de jubilación, los impuestos, etc.

2.2.1 Conceptos clave finanzas

Riesgo y beneficio: Los inversores actúan en los diferentes mercados intentando obtener el mayor rendimiento para su dinero a la vez que pretenden minimizar el riesgo de su inversión. El mercado de capitales ofrece en cada momento una frontera eficiente, que relaciona una determinada rentabilidad con un determinado nivel de riesgo o volatilidad. El inversor obtiene una mayor rentabilidad esperada a cambio de soportar una mayor incertidumbre. El precio de la incertidumbre es la diferencia entre la rentabilidad de la inversión y el tipo de interés de aquellos valores que se consideran seguros. A esta diferencia la conocemos como prima de riesgo.

El valor del dinero en el tiempo: Ante la misma cantidad de dinero, un inversor prefiere disponer de ella en el presente que en el futuro. Por ello, el transvase intertemporal de dinero cuenta con un factor de descuento (si intercambiamos renta futura por capital presente, por ejemplo, en un préstamo hipotecario), o con una rentabilidad (si ínter cambiamos renta presente por renta futura, por ejemplo, en un plan de pensiones).

Tasa de interés: Se define como el precio que se paga por los fondos solicitados en préstamo, en un periodo de tiempo. Por lo general es expresado en porcentaje y representa una tasa de intercambio entre el precio del dinero al día de hoy en términos del dinero futuro.

La tasa de interés afecta directamente el consumo, el comercio y la inversión, pues parte del consumo se paga mediante tarjetas de crédito, parte de la mercancía comprada y vendida por los comercios es comprada a crédito, y las inversiones siempre se apoyan con préstamos bancarios o emisión de deuda mediante bonos: Al subir la tasa de interés el consumo y la inversión disminuyen, pues individuos y empresas encuentran más difícil pagar sus deudas; al bajar la tasa de interés el consumo y la inversión aumentan por el estímulo que representa pagar menos intereses.

Esta relación entre tasa de interés, consumo e inversión es utilizada por los diseñadores de políticas macroeconómicas para manipularlos para afectar crecimiento económico, empleo e inflación: Para disminuir la inflación inducen un aumento en las tasas de interés; cuando quieren aumentar el empleo, inducen su disminución. Tales acciones usualmente son llevadas a cabo mediante el Banco Central de cada país al establecer la llamada tasa de descuento – la tasa de interés a la cual presta dinero a los bancos. Este cambio en la tasa de descuento impacta los costos del dinero en los bancos y, por tanto, impacta la tasa de interés que tales bancos cobran por sus préstamos.

La relación entre liquidez e inversión: La necesidad de contar con dinero líquido tanto para el intercambio por bienes y servicios como para realizar una inversión hace que el mercado de la mercancía-dinero tenga su propia oferta y demanda, y sus propios costes y precios.

Costes de oportunidad: Hace referencia al sacrificio que debe hacer cualquier agente que participe en un mercado al decidir prescindir de un consumo o de una inversión para emplear sus recursos, siendo estos por definición escasos, en otro proyecto.

Inflación: Proceso económico que consiste en una subida continuada de los precios de la mayor parte de los productos y servicios, y, por lo tanto, de una pérdida del valor del dinero para poder adquirir esos bienes y servicios, la inflación impacta principalmente a la economía de un país, es el nuevo costo de vida. La inflación tiene un impacto directo sobre la tasa de interés: puesto que la inflación reduce el valor del dinero, a mayor inflación, mayor es la tasa de interés necesaria para compensar a un ahorrador por prestar dinero al vecino. (García, 2018)

2.2.2 Cartera

Monto total de créditos otorgados por una entidad de intermediación financiera.

- Cartera vigente. monto total de créditos que tienen sus amortizaciones de capital e intereses al día, conforme al plan de pagos establecido en los contratos de crédito.
- Cartera con atraso hasta 30 días. monto total de créditos cuyo capital, cuotas de amortización o intereses registran atraso (mora) de hasta 30 días, contados desde el día de su vencimiento.
- Cartera vencida. monto total de créditos cuyo capital, cuotas de amortización o intereses no han sido cancelados íntegramente a la entidad transcurridos los 30 días contados desde la fecha de vencimiento. la cartera pasa de atraso a vencida a partir del día 30 desde su vencimiento.
- Cartera en ejecución. monto total de créditos por los cuales la entidad financiera ha iniciado las acciones judiciales para el cobro.
- Cartera en mora. monto total de créditos que resulta de sumar la cartera vencida más la cartera en ejecución. es el total de cartera cuyo capital, cuotas de amortización o intereses no han sido cancelados íntegramente a la entidad transcurridos los 30 días contados desde la fecha de vencimiento o para la cual se iniciaron acciones judiciales para su cobro.
- Crédito. activo de riesgo asumido por una entidad financiera autorizada con un prestatario.

2.2.3 Tipos Amortización financiera

Desde el punto de vista financiero, se entiende por amortización, el reembolso gradual de una deuda. La obligación de devolver un préstamo recibido de un banco es un pasivo, cuyo importe se va reintegrando en varios pagos diferidos en el tiempo. La parte del capital prestado (o principal) que se cancela en cada uno de esos pagos es una amortización. Los métodos más frecuentes para repartir el

importe en el tiempo y segregar principal de intereses son el sistema francés, alemán y el americano.

Todos estos métodos son correctos desde el punto de vista contable y están basados en el concepto de interés compuesto. Las condiciones pactadas al momento de acordar el préstamo determinan cuál de los sistemas se utilizará.

- El sistema francés consiste en determinar una cuota fija. Mediante el cálculo apropiado del interés compuesto se segrega el principal (que será creciente) de los intereses (decrecientes).
- En el sistema alemán, o sistema de cuota de amortización fija, la amortización de capital es fija, por lo tanto, los intereses y la cuota total serán decrecientes. Se caracteriza porque el interés se paga de forma anticipada en cada anualidad.
- El sistema americano establece una sola amortización única al final de la vida del préstamo. A lo largo de la vida del préstamo solo se pagan intereses. Al no haber pagos intermedios de capital, los intereses anuales son fijos. En si son el contrario de la depreciación.

2.2.3.1 Sistema de Amortización Frances

El sistema de amortización francés, se calcula mediante una equivalencia financiera. De esta forma, lo que el banco nos entrega es el préstamo solicitado y lo que nosotros nos comprometemos a pagar son las cuotas constantes. Su fórmula matemática es relativamente sencilla y se pueden hacer los cálculos utilizando una hoja de cálculo. La ecuación de la anualidad (a) sería esta.

Ecuación 2.1 Cuota fija mensual

$$R = P * \frac{(1 + i)^n * i}{(1 + i)^n - 1}$$

Fuente: (RAMIREZ MOLINARES, GARCIA BARBOZA, PANTOJA ALGARIN, & ZAMBRANO MEZA, 2009)

Donde:

R es cuota fija mensual a pagar sin gastos

P es capital prestado

i es tasa de interés efectiva mensual

n número de cuotas del crédito

Una vez tenemos el importe de la cuota a pagar, se puede preparar el cuadro de amortización, aunque el banco nos facilita uno detallado al solicitar cualquier tipo de préstamo. Aún así, podemos realizar un cálculo aproximado nosotros mismos (que nos será muy útil) teniendo en cuenta estas dos expresiones:

Tabla 2.1 Tabla de amortización a cuota fija

Mes	Saldo al inicio	Interés	Amortización	Cuota fija
n				
0	capital			Cuota fija
1	Capital- amortización	Cuota fija*interés	Cuota fija- interés	

Fuente: (RAMIREZ MOLINARES, GARCIA BARBOZA, PANTOJA ALGARIN, & ZAMBRANO MEZA, 2009)

2.2.4 Tipos de préstamos

La CAJA POPULAR BOLIVIANA se centra en el tipo de préstamo de libre disponibilidad.

2.2.4.1 Préstamo personal

Esta clase de préstamos se utiliza generalmente para financiar necesidades específicas en un momento determinado y de un importe pequeño. Es muy parecido al préstamo al consumo, aunque en este caso se usan para costearse viajes, una boda, etc. Es decir, bienes intangibles o perecederos.

2.2.4.2 Préstamos de libre disponibilidad

Crédito destinado a la adquisición de bienes y servicios de uso personal; gastos personales, médicos, educativos, viajes, de equipamiento y otros.

2.2.4.3 Préstamos de estudios

Se trata de un tipo de préstamos muy utilizados en el Reino Unido y Estados Unidos que se están globalizando cada vez más debido a que las entidades financieras intentan mejorar sus mercados anualmente. Se trata de préstamos dirigidos a estudiantes para financiar matrículas universitarias, estudios de postgrado o viajes al extranjero. Suelen tener un tipo de interés más barato que los préstamos personales.

2.2.5 Requisitos para préstamo

Las clases de préstamos que existen en la actualidad y lo que te exigen para concedértelos, requisitos y condiciones, todo con el objetivo de informarles, de una forma amena, clara y concisa, sobre cuáles son los productos financieros de este tipo que actualmente ofrece nuestro sistema bancario tradicional a sus potenciales clientes, los requisitos son los siguientes:

- Fotocopia cédula de identidad del Titular, cónyuge y garantes (si aplica).
- Fuente de pago: ingresos familiares (sueldos y salarios, utilidades y/o otros ingresos respaldados).
- Estabilidad Laboral: 1 año.
- Carga Financiera Total/Renta Total: Según Manual de políticas de crédito.
- Para créditos de construcción, remodelación o ampliación de vivienda se debe presentar presupuesto y cronograma de avance de obra firmado por un profesional competente.

- Documentación de respaldo de deudas vigentes en el sistema financiero del titular, cónyuge y garantes (si aplica).
- Documentación de respaldo de cajas de ahorro o cuentas corrientes del Titular, cónyuge y garantes (si aplica).
- Llenado de Formulario de Solicitud de Crédito.
- Llenado de Formulario de Declaración Patrimonial (adjuntar respaldo de activos declarados).
- Certificado de no propiedad de DD RR (Si aplica).

2.2.6 Ente regulatorio

La Autoridad de Supervisión del Sistema Financiero (ASFI) es el ente regulador boliviano encargado de supervisar a las instituciones pertenecientes a los sectores de banca, seguros y valores. Es responsable de identificar y monitorear a las instituciones financieras. Además de proporciona estadísticas y estudios relacionados con el sistema bancario local.

2.2.6.1 Autoridad de supervisión del sistema financiero

ASFI es una institución de derecho público y de duración indefinida, con personalidad jurídica, patrimonio propio y autonomía de gestión administrativa, financiera, legal y técnica, con jurisdicción, competencia y estructura de alcance nacional, bajo tuición del Ministerio de Economía y Finanzas Públicas y sujeta a control social.

El objeto de ASFI es regular, controlar y supervisar los servicios financieros en el marco de la Constitución Política del Estado, la Ley N. ° 393 de Servicios Financieros y los Decretos Supremos reglamentarios, así como la actividad del mercado de valores, los intermediarios y sus entidades auxiliares.

Las actividades financieras y la presentación de servicios financieros deben ser realizadas únicamente por entidades que tengan autorización de ASFI, según los tipos de entidades definidos en la Ley N. ° 393. (ASFI, 2018)

En el reglamento para la gestión de seguridad de la información:

Se estipula que las entidades supervisadas deben desarrollar un plan estratégico de tecnologías de la información (TI), que considere su naturaleza, tamaño, complejidad de sus operaciones, procesos, estructura, además de un análisis y evaluación de riesgos en seguridad de la información. Así mismo, se establece que la estrategia en seguridad de la información formara parte del plan estratégico de TI. (ASFI, 2018)

2.3 INTRANET

Es un esquema de red donde se utiliza, sobre una plataforma de red organizacional o interorganizacional existente, con el medio de transporte entre sus unidades, el protocolo de internet, el TCP/IP, aquellas tecnologías avanzadas de publicación electrónica Web (http), correo electrónico (email), archivos y recursos compartidos (file server, printer server, backup server, ftp server), acceso remoto (vpn, ssh), etc. brindando así todo el potencial del esquema cliente / servidor en un ámbito acotado, seguro y racionalizado de acuerdo a los objetivos de eficacia y eficiencia estimados en dicha organización. (Mogetta, Audisio & Casco, 2014)

En la Intranet solo acceden los empleados y las áreas internas de la empresa y permiten el intercambio de información entre los trabajadores.

Tabla 2.2 Comparación de internet, intranet y extranet

	Internet	Intranet	Extranet
Acceso	Publico	Privado	Semi publico

Usuario	Cualquiera	Miembros de una compañía	Grupo de empresas estrechamente relacionado
Información	Fragmentada	Propietaria	Compartida dentro de un círculo de empresas

Fuente: mogetta, audisio & Casco, 2014

2.3.1 Elementos de la intranet

- **SEGURIDAD:** define el ámbito y normativa para asegurarlo de virus, violación de acceso, fuga de información, etc.
- **INTEGRIDAD:** protege los cambios o eliminación de datos no queridos.
AUTENTICACIÓN: control de ingreso a la red de acuerdo al estándar de verificación de identidades.
- **AUDITORÍA:** control de la actividad en tiempo real y posterior, con posibilidad de retroalimentación del sistema (autoaprendizaje).
- **DISPONIBILIDAD Y OPORTUNIDAD:** asegura la presencia de los servicios y datos requeridos en tiempo y forma predefinidos.

2.3.2 Ventajas

- Eficiencia en la administración de las it3. es una forma muy eficiente y económica de distribuir la información interna.
- **Adaptabilidad:** facilita la aplicación de las distintas soluciones en la infraestructura tecnológica de la organización.
- **Configurabilidad:** moldeable a los requerimientos de diferentes niveles: empresa, departamento, área de negocio, etc. puede planearse centralizando o descentralizando el acceso a la información.
- **Mayor integración, consistencia y normalización de datos:** posibilidad de compartir bases de datos organizacionales o interorganizacionales.

- Uniformidad de capacitación de los agentes: el personal puede desenvolverse en cualquier sector con las mismas herramientas.
- Control de acceso: verificación confiable de los usuarios.

2.3.3 Desventajas

- Planificación: se debe realizar con una visión holística de política, objetivos, planes, y metas de la organización.
- Seguridad: debe plantearse con firmeza la política de seguridad para no ser blanco de los mismos peligros que se quieren evitar en internet.

La arquitectura de software, se define como la organización fundamental de un sistema incorporando sus componentes, las relaciones entre ellos y el ambiente y los principios que orientan su diseño y evolución. están relacionados con la interacción de los objetos. Los patrones de arquitectura, al igual que los patrones de diseño, representan llamadas entre objetos, decisiones y criterios arquitectónicos, así como la manera de empaquetado de funcionalidades de una aplicación. Sin embargo, los patrones de arquitectura representan un nivel más alto en el sistema. (Buschmann, Meurier, Rohnert, Sommerlad, & Stal, 1996)

2.4 INGENIERÍA DE REQUERIMIENTOS

¿Qué son Requerimientos?

Normalmente, un tema de la Ingeniería de Software tiene diferentes significados. De las muchas definiciones que existen para requerimiento, a continuación, se presenta la definición que aparece en el glosario de la IEEE.

- (1) Una condición o necesidad de un usuario para resolver un problema o alcanzar un objetivo.
- (2) Una condición o capacidad que debe estar presente

en un sistema o componentes de sistema para satisfacer un contrato, estándar, especificación u otro documento formal. (3) Una representación documentada de una condición o capacidad como en (1) o (2).

La ingeniería de software aplica diferentes normas y métodos que permiten obtener mejores resultados, en cuanto al desarrollo y uso del software, mediante la aplicación correcta de estos procedimientos se puede llegar a cumplir de manera satisfactoria con los objetivos fundamentales de la ingeniería de software.

El establecimiento y uso de principios sólidos de la ingeniería para obtener económicamente un software confiable y que funcione de modo eficiente, entre los objetivos están:

- Mejorar el diseño de aplicaciones o software de tal modo que se adapten de mejor manera a las necesidades de las organizaciones o finalidades para las cuales fueron creadas.
- Promover mayor calidad al desarrollar aplicaciones complejas.
- Brindar mayor exactitud en los costos de proyectos y tiempo de desarrollo de los mismos.
- Aumentar la eficiencia de los sistemas al introducir procesos que permitan medir mediante normas específicas, la calidad del software desarrollado, buscando siempre la mejor calidad posible según las necesidades y resultados que se quieren generar.
- Una mejor organización de equipos de trabajo, en el área de desarrollo y mantenimiento de software.
- Detectar a través de pruebas, posibles mejoras para un mejor funcionamiento del software desarrollado.

2.4.1 La ingeniería de requerimientos y sus principales actividades

Los requerimientos pueden dividirse en requerimientos funcionales y requerimientos no funcionales.

Los requerimientos funcionales definen las funciones que el sistema será capaz de realizar. Describen las transformaciones que el sistema realiza sobre las entradas para producir salidas.

Los requerimientos no funcionales tienen que ver con características que de una u otra forma puedan limitar el sistema, como, por ejemplo, el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad (robustez del sistema, disponibilidad de equipo), mantenimiento, seguridad, portabilidad, estándares, etc. (Sommerville, 2011)

2.4.1.1 Características de los requerimientos

Las características de un requerimiento son sus propiedades principales. Un conjunto de requerimientos en estado de madurez, deben presentar una serie de características tanto individualmente como en grupo. A continuación, se presentan las más importantes. (Sommerville, 2011)

- **Necesario:** Un requerimiento es necesario si su omisión provoca una deficiencia en el sistema a construir, y además su capacidad, características físicas o factor de calidad no pueden ser reemplazados por otras capacidades del producto o del proceso.
- **Conciso:** Un requerimiento es conciso si es fácil de leer y entender. Su redacción debe ser simple y clara para aquellos que vayan a consultarlo en un futuro.
- **Completo:** Un requerimiento está completo si no necesita ampliar detalles en su redacción, es decir, si se proporciona la información suficiente para su comprensión.

- Consistente: Un requerimiento es consistente si no es contradictorio con otro requerimiento.
- No ambiguo: Un requerimiento no es ambiguo cuando tiene una sola interpretación. El lenguaje usado en su definición, no debe causar confusiones al lector.
- Verificable: Un requerimiento es verificable cuando puede ser cuantificado de manera que permita hacer uso de los siguientes métodos de verificación: inspección, análisis, demostración o pruebas.

Dificultades para definir los requerimientos

- Los requerimientos no son obvios y vienen de muchas fuentes.
- Son difíciles de expresar en palabras (el lenguaje es ambiguo).
- Existen muchos tipos de requerimientos y diferentes niveles de detalle.
- La cantidad de requerimientos en un proyecto puede ser difícil de manejar.
- Nunca son iguales. Algunos son más difíciles, más riesgosos, más importantes o más estables que otros.
- Los requerimientos están relacionados unos con otros, y a su vez se relacionan con otras partes del proceso.
- Cada requerimiento tiene propiedades únicas y abarcan áreas funcionales específicas.
- Un requerimiento puede cambiar a lo largo del ciclo de desarrollo.
- Son difíciles de cuantificar, ya que cada conjunto de requerimientos es particular para cada proyecto.

2.4.2 Técnicas utilizadas en la ingeniería de requerimientos

2.4.2.1 Entrevistas y Cuestionarios

Las entrevistas y cuestionarios se emplean para reunir información proveniente de personas o de grupos. Durante la entrevista, el analista conversa con el encuestado; el cuestionario consiste en una serie de preguntas relacionadas con varios aspectos de un sistema. Por lo común, los encuestados son usuarios de los sistemas existentes o usuarios en potencia del sistema propuesto. En algunos casos, son gerentes o empleados que proporcionan datos para el sistema propuesto o que serán afectados por él. Las preguntas que deben realizarse en esta técnica, deben ser preguntas de alto nivel y abstractas que pueden realizarse al inicio del proyecto para obtener información sobre aspectos globales del problema del usuario y soluciones potenciales. Con frecuencia, se utilizan preguntas abiertas para descubrir sentimientos, opiniones y experiencias generales, o para explorar un proceso o problema. Este tipo de preguntas son siempre apropiadas, además que ayudan a entender la perspectiva del afectado y no están influenciadas por el conocimiento de la solución. Las preguntas pueden ser enfocadas a un elemento del sistema, tales como usuarios, procesos, etc. El siguiente ejemplo muestra algunos tipos de preguntas abiertas. Del Usuario. (Sommerville, 2011)

- ¿Quién es el cliente?
- ¿Quién es el usuario?
- ¿Son sus necesidades diferentes?
- ¿Cuáles son sus habilidades, capacidades, ambiente?

Del Proceso:

- ¿Cuál es la razón por la que se quiere resolver este problema?

- ¿Cuál es el valor de una solución exitosa?
- ¿Cómo usted resuelve el problema actualmente?
- ¿Qué retrasos ocurren o pueden ocurrir?

Del Producto:

- ¿Qué problemas podría causar este producto en el negocio?
- ¿En qué ambiente se usará el producto?
- ¿Cuáles son sus expectativas para los conceptos fácil de usar, confiable, rendimiento?
- ¿Qué obstáculos afectan la eficiencia del sistema?

2.4.2.2 Lluvia de Ideas (*Brainstorm*)

Este método comenzó en el ámbito de las empresas, aplicándose a temas tan variados como la productividad, la necesidad de encontrar nuevas ideas y soluciones para los productos del mercado, encontrar nuevos métodos que desarrollen el pensamiento creativo a todos los niveles, etc. Pero pronto se extendió a otros ámbitos, incluyendo el mundo de desarrollo de sistemas; básicamente se busca que los involucrados en un proyecto desarrollen su creatividad, promoviendo la introducción de los principios creativos. A esta técnica se le conoce también como torbellino de ideas, tormenta de ideas, desencadenamiento de ideas, movilización verbal, bombardeo de ideas, sacudidas de cerebros, promoción de ideas, tormenta cerebral, avalancha de ideas, tempestad en el cerebro y tempestad de ideas, entre otras. Principios de la lluvia de ideas

- Aplazar el juicio y no realizar críticas, hasta que no agoten las ideas, ya que actuaría como un inhibidor. Se ha de crear una atmósfera de trabajo en la que nadie se sienta amenazado.

- Cuantas más ideas se sugieren, mejores resultados se conseguirán: “la cantidad produce la calidad”. Las mejores ideas aparecen tarde en el periodo de producción de ideas, será más fácil que encontremos las soluciones y tendremos más variedad sobre la que elegir.
- La producción de ideas en grupos puede ser más efectiva que la individual.
- Tampoco debemos olvidar que durante las sesiones, las ideas de una persona, serán asociadas de manera distinta por cada miembro, y hará que aparezcan otras por contacto.

Dice que el análisis da como resultado un modelo del sistema que pretender ser correcto, completo, consistente y verificable. Los desarrolladores formalizan la especificación del sistema producida durante la obtención de requerimientos y examinan con mayor detalle las condiciones de frontera y los casos excepcionales. El cliente y el usuario están involucrados, por lo general, en esta actividad, en especial cuando se necesita cambiar la especificación del sistema y cuando se necesita recopilar la información adicional. (Markus & Wirkungsforchung, 2007)

2.5 RATIONAL UNIFIED PROCESS (RUP)

El Proceso Racional Unificado o RUP (por sus siglas en inglés de *Rational Unified Process*) es un proceso de desarrollo de software desarrollado por la empresa Rational Software, actualmente propiedad de IBM. El RUP no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización. También se conoce por este nombre al software, también desarrollado por Rational, que incluye información entrelazada de diversos artefactos y descripciones de las diversas actividades. Está incluido en el *Rational Method Composer* (RMC), que permite la personalización de acuerdo con las necesidades.

Originalmente se diseñó un proceso genérico y de dominio público, el Proceso Unificado, y una especificación más detallada, el *Rational Unified Process*, que se vendiera como producto independiente. (andrew, 2020)

2.5.1 Principios de desarrollo

La Filosofía del RUP está basado en 6 principios clave que son los siguientes:

2.5.1.1 Adaptar el proceso

El proceso deberá adaptarse a las necesidades del cliente ya que es muy importante interactuar con él. Las características propias del proyecto, el tamaño del mismo, así como su tipo o las regulaciones que lo condicionen, influirán en su diseño específico. También se deberá tener en cuenta el alcance del proyecto.

2.5.1.2 Equilibrar prioridades

Los requisitos de los diversos participantes pueden ser diferentes, contradictorios o disputarse recursos limitados.

Debe poder encontrarse un equilibrio que satisfaga los deseos de todos. Gracias a este equilibrio se podrán corregir desacuerdos que surjan en el futuro.

2.5.1.3 Demostrar valor iterativamente

Los proyectos se entregan, aunque sea de un modo interno, en etapas iteradas. En cada iteración se analiza la opinión de los inversores, la estabilidad y calidad del producto, y se refina la dirección del proyecto, así como también los riesgos involucrados.

2.5.1.4 Colaboración entre equipos

El desarrollo de software no lo hace una única persona sino múltiples equipos. Debe haber una comunicación fluida para coordinar requisitos, desarrollo, evaluaciones, planes, resultados, etc.

2.5.1.5 Enfocarse en la calidad

El control de calidad no debe realizarse al final de cada iteración, sino en todos los aspectos de la producción. El aseguramiento de la calidad forma parte del proceso de desarrollo y no de un grupo independiente, también es una estrategia de desarrollo de software.

2.5.1.6 Elevar el Nivel de Abstracción

Este principio dominante motiva el uso de conceptos reutilizables tales como patrones de diseño del software, lenguajes 4GL o esquemas (*frameworks*) por nombrar algunos. Estos se pueden acompañar por las representaciones visuales de la arquitectura, por ejemplo, con UML.

2.5.2 Ciclo de vida

Esfuerzo en actividades según fase del proyecto. El ciclo de vida RUP es una implementación del desarrollo en espiral. Fue creado ensamblando los elementos en secuencias semiordenadas. El ciclo de vida organiza las tareas en fases e iteraciones. RUP divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en las distintas actividades. En la Figura muestra cómo varía el esfuerzo asociado a las disciplinas según la fase en la que se encuentre el proyecto RUP.

Las primeras iteraciones (en las fases de Inicio y Elaboración) se enfocan hacia la comprensión del problema y la tecnología, la delimitación del ámbito del proyecto, la eliminación de los riesgos críticos, y al establecimiento de una *base line* (Línea Base) de la arquitectura. Durante la fase de inicio las iteraciones hacen mayor énfasis en actividades de modelado del negocio y de requisitos.

En la fase de elaboración, las iteraciones se orientan al desarrollo de la *base line* de la arquitectura, abarcan más los flujos de trabajo de requisitos, modelo de negocios (refinamiento), análisis, diseño y una parte de implementación orientado a la *base line* de la arquitectura.

En la fase de construcción, se lleva a cabo la construcción del producto por medio de una serie de iteraciones. Para cada iteración se seleccionan algunos Casos de Uso, se refinan su análisis y diseño y se procede a su implementación y pruebas. Se realiza una pequeña cascada para cada ciclo. Se realizan iteraciones hasta que se termine la implementación de la nueva versión del producto.

En la fase de transición se pretende garantizar que se tiene un producto preparado para su entrega a la comunidad de usuarios. Como se puede observar en cada fase participan todas las disciplinas, pero dependiendo de la fase el esfuerzo dedicado a una disciplina varía.

2.5.3 Principales características

El RUP es un producto de Rational (IBM). Se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Incluye artefactos (que son los productos tangibles del proceso como, por ejemplo, el modelo de casos de uso, el código fuente, etc.) y roles (papel que desempeña una persona en un determinado momento, una persona puede desempeñar distintos roles a lo largo del proceso).

- Desarrollo iterativo.

- Administración de requisitos.
- Uso de arquitectura basada en componentes.
- Control de cambios.
- Modelado visual del software.
- Verificación de la calidad del software.
- Pretende implementar las mejores prácticas en Ingeniería de Software, de forma que se adapte a cualquier proyecto.

2.5.4 Fases

- Establece oportunidad y alcance
- Identifica las entidades externas o actores con las que se trata
- Identifica los casos de uso

RUP comprende 2 aspectos importantes por los cuales se establecen las disciplinas: proceso y soporte.

2.5.4.1 Proceso

Las etapas de esta sección son:

- Modelado de negocio.
- Requisitos.
- Análisis y Diseño.
- Implementación.
- Pruebas.
- Despliegue.

2.5.4.2 Soporte

En esta parte nos encontramos con las siguientes etapas:

- Gestión del cambio y configuraciones.
- Gestión del proyecto.
- Entorno.

2.5.5 Dinámica de RUP

La estructura dinámica de RUP es la que permite que éste sea un proceso de desarrollo fundamentalmente iterativo, y en esta parte se ven inmersas las 4 fases descritas anteriormente:

2.5.5.1 Fase de Inicio

Esta fase tiene como propósito definir y acordar el alcance del proyecto con los patrocinadores, identificar los riesgos asociados al proyecto, proponer una visión muy general de la arquitectura de software y producir el plan de las fases y el de iteraciones posteriores.

2.5.5.2 Fase de elaboración

En la fase de elaboración se seleccionan los casos de uso que permiten definir la arquitectura base del sistema y se desarrollaran en esta fase, se realiza la especificación de los casos de uso seleccionados y el primer análisis del dominio del problema, se diseña la solución preliminar.

2.5.5.3 Fase de Desarrollo

El propósito de esta fase es completar la funcionalidad del sistema, para ello se deben clarificar los requisitos pendientes, administrar los cambios de acuerdo a las evaluaciones realizados por los usuarios y se realizan las mejoras para el proyecto.

2.5.5.4 Fase de Transición

El propósito de esta fase es asegurar que el software esté disponible para los usuarios finales, ajustar los errores y defectos encontrados en las pruebas de aceptación, capacitar a los usuarios y proveer el soporte técnico necesario. Se debe verificar que el producto cumpla con las especificaciones entregadas por las personas involucradas en el proyecto.

2.5.6 Artefactos

RUP en cada una de sus fases (pertenecientes a la estructura dinámica) realiza una serie de artefactos que sirven para comprender mejor tanto el análisis como el diseño del sistema. Estos artefactos (entre otros) son los siguientes:

Inicio:

- Documento Visión.
- Diagramas de caso de uso.
- Especificación de Requisitos.
- Diagrama de Requisitos.

Elaboración: Documento Arquitectura que trabaja con las siguientes vistas:

- Vista Lógica como Diagrama de clases y Modelo E-R.

- Vista de Implementación los Diagrama de Secuencia, Diagrama de estados y Diagrama de Colaboración.
- Vista Conceptual Modelo de dominio.
- Diseño y desarrollo de casos de uso, o flujos de casos de uso arquitectónicos.
- Pruebas de los casos de uso desarrollados, que demuestran que la arquitectura documentada responde adecuadamente a requerimientos funcionales y no funcionales.

Construcción:

- Especificación de requisitos faltantes.
- Diseño y desarrollo de casos de uso y/o flujos de acuerdo con la planeación iterativa.
- Pruebas de los casos de uso desarrollados, y pruebas de regresión según sea el caso.

Transición:

- Pruebas finales de aceptación.
- Puesta en producción.
- Estabilización.

linteractúa a este desarrollo brindando una forma disciplinada al asignar tareas y responsabilidades en una empresa de desarrollo (quién hace qué, cuándo y cómo). Es allí donde entran a tallar los diagramas de casos de uso, que es un esquema(modelo) para abstraer una vista del sistema del mundo real, considerando un cierto propósito, y describir el comportamiento de un actor y su interacción con el sistema. (Guadaña Quiroz & Pachamora Pinedo, 2016)

2.6 UNIFIED MODELING LANGUAGE (UML)

En uno de los párrafos más citados del artículo por lejos más citado en la bibliografía de la Ingeniería del Software, (Frederick & Brooks, 1987), dice:

“La parte más difícil de construir un sistema es precisamente saber qué construir. Ninguna otra parte del trabajo conceptual es tan difícil como establecer los requerimientos técnicos detallados, incluyendo todas las interfaces con gente, máquinas, y otros sistemas. Ninguna otra parte del trabajo afecta tanto al sistema si es hecha mal. Ninguna es tan difícil de corregir más adelante entonces, la tarea más importante que el ingeniero de software hace para el cliente es la extracción iterativa y el refinamiento de los requerimientos del producto”. (Frederick & Brooks, 1987)

Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML (Lenguaje unificado de modelado) ofrece un estándar para describir un “plano” del sistema (modelo), incluyendo aspectos conceptuales tales como procesos, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y compuestos reciclados.

Es importante remarcar que UML es un “lenguaje de modelado” para especificar o para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo.

UML no puede compararse con la programación estructurada, pues UML significa Lenguaje Unificado de Modelado, no es programación, solo se diagrama la realidad de una utilización en un requerimiento.

La programación orientada a objetos viene siendo un complemento perfecto de UML, pero no por eso se toma UML sólo para lenguajes orientados a objetos.

UML cuenta con varios tipos de diagramas para el análisis y diseño, muestran diferentes aspectos de las entidades representadas, los cuales son.

2.6.1 ANÁLISIS

2.6.1.1 Diagrama de casos de uso

Los diagramas de caso de uso brindan un panorama bastante sencillo de una interacción, de modo que es necesario ofrecer más detalle para entender lo que está implicado. Este detalle puede ser una simple descripción textual, o una descripción estructurada en una tabla o un diagrama de secuencia. Es posible elegir el formato más adecuado, dependiendo del caso de uso y del nivel de detalle que se considere que requiera el modelo. El formato más útil es un formato tabular estándar. (Frederick & Brooks, 1987)

Los diagramas de caso de uso compuestos indican un número de casos de uso diferentes. En ocasiones, se incluyen todas las interacciones posibles con un sistema en un solo diagrama de caso de uso compuesto. Sin embargo, esto quizá sea imposible debido a la cantidad de casos de uso. En tales situaciones, se pueden desarrollar varios diagramas, cada uno de los cuales exponga casos de uso relacionados.

2.6.1.2 Diagrama de estados

El comportamiento de un objeto en un punto particular en el tiempo con frecuencia depende del estado del objeto; es decir, de los valores de sus variables en dicho momento. Cuando se pide realizar una operación, un objeto puede hacer una cosa si dicha variable es verdadera y hacer algo más si es falsa. Un diagrama de estado UML modela los estados de un objeto, las acciones que se realizan dependiendo

de dichos estados y las transiciones entre los estados del objeto. (Frederick & Brooks, 1987)

El símbolo del estado contiene el nombre del mismo y puede tener variables y actividades del estado. Una transición puede suceder como respuesta a un suceso desencadenado, e implicar una respuesta o acción. Una transición también puede ocurrir por la actividad en un estado: una transición que ocurre de esta forma se conoce como transición no desencadenada. Finalmente, una transición puede ocurrir cuando se cumple una condición particular, o condición de seguridad.

En ocasiones, un estado consta de subestados. Los subestados pueden ser secuenciales (ocurrir uno después del otro) o concurrentes (ocurrir al mismo tiempo). Un estado que consta de subestados se conoce como estado compuesto. Un estado histórico indica que un estado compuesto recordará su subestado cuando el objeto trascienda de este estado compuesto. Un estado histórico puede ser superficial o profundo. Tales términos son propios de los subestados anidados. Un estado histórico superficial recuerda sólo el subestado principal. Un estado histórico profundo recuerda todos los niveles de los subestados.

Es necesario contar con los diagramas de estados porque facilitan la comprensión de los objetos de un sistema a los analistas, diseñadores y desarrolladores.

Los desarrolladores, en particular, deben saber cómo se supone que se comportarán los objetos, dado que serán quienes tengan que establecer estos comportamientos en el software. No es suficiente implementar un objeto: los desarrolladores tienen que hacer que tal objeto haga algo.

2.6.1.3 Diagrama de secuencia

Los diagramas de secuencia en el UML se usan principalmente para modelar las interacciones entre los actores y los objetos en un sistema, así como las interacciones entre los objetos en sí. (Frederick & Brooks, 1987)

El UML tiene una amplia sintaxis para diagramas de secuencia, lo cual permite muchos tipos diferentes de interacción a modelar. Típicamente se examina la descripción de un caso de uso para determinar qué objetos son necesarios para la implementación del escenario. Si se dispone de la descripción de cada caso de uso como una secuencia de varios pasos, entonces se puede “caminar sobre” esos pasos para descubrir qué objetos son necesarios para que se puedan seguir los pasos. Un diagrama de secuencia muestra los objetos que intervienen en el escenario con líneas discontinuas verticales, y los mensajes pasados entre los objetos como flechas horizontales.

Dentro de la sintaxis para modelar diagramas de secuencias existen dos tipos de mensajes: sincrónicos y asincrónicos. Los mensajes sincrónicos se corresponden con llamadas a métodos del objeto que recibe el mensaje. El objeto que envía el mensaje queda bloqueado hasta que termina la llamada. Este tipo de mensajes se representan con flechas con la cabeza llena. Los mensajes asincrónicos terminan inmediatamente, y crean un nuevo hilo de ejecución dentro de la secuencia. Se representan con flechas con la cabeza abierta.

Los mensajes se dibujan cronológicamente desde la parte superior del diagrama a la parte inferior; la distribución horizontal de los objetos es arbitraria.

Durante el análisis inicial, el modelador típicamente coloca el nombre *'business'* de un mensaje en la línea del mensaje. Más tarde, durante el diseño, el nombre *'business'* es reemplazado con el nombre del método que está siendo llamado por un objeto en el otro. El método llamado o invocado pertenece al objeto receptor del mensaje.

2.6.1.4 Diagrama de comunicación

Un diagrama de colaboraciones es otra forma de presentar la información en un diagrama de secuencias. Ambos tipos de diagramas son semánticamente

equivalentes y se recomienda usar ambos cuando construya el modelo de un sistema. El diagrama de secuencias se organiza de acuerdo al tiempo, y el de colaboración de acuerdo al espacio. (Frederick & Brooks, 1987)

El diagrama de colaboraciones muestra las asociaciones entre objetos, así como los mensajes que pasan de un objeto a otro. El mensaje se representa con una flecha junto a la línea de asociación, y una etiqueta numerada que muestra el contenido del mensaje. El número representa el turno del mensaje en la secuencia.

Las condicionales se representan como antes, mediante la colocación de la instrucción condicional entre corchetes. Algunos mensajes provienen de otros. El esquema de numeración de las etiquetas representa esto de forma muy similar a los manuales técnicos que muestran sus encabezados y subtítulos: con un sistema de numeración que utiliza puntos decimales para representar los niveles del anidamiento.

Los diagramas de colaboraciones permiten modelar varios objetos receptores en una clase, ya sea que los objetos reciban o no los mensajes en un orden específico. También es posible representar objetos activos que controlen el flujo de los mensajes, así como los mensajes que se sincronizan con otros.

2.6.1.5 Diagrama de clases

Para modelar clases, incluidos sus atributos, operaciones, relaciones y asociaciones con otras clases, el UML proporciona un diagrama de clase, que aporta una visión estática o de estructura de un sistema, sin mostrar la naturaleza dinámica de las comunicaciones entre los objetos de las clases. Los elementos principales de un diagrama de clase son cajas, que son los íconos utilizados para representar clases e interfaces. Cada caja se divide en partes horizontales. La parte superior contiene el nombre de la clase. La sección media menciona sus atributos. (Frederick & Brooks, 1987)

Un atributo es algo que un objeto de dicha clase conoce o puede proporcionar todo el tiempo. Por lo general, los atributos se implementan como campos de la clase, pero no necesitan serlo. Podrían ser valores que la clase puede calcular a partir de sus variables o valores instancia y que puede obtener de otros objetos de los cuales está compuesto. La tercera sección del diagrama de clase contiene las operaciones o comportamientos de la clase. Una operación es lo que pueden hacer los objetos de la clase. Por lo general, se implementa como un método de la clase. Los diagramas de clase también pueden mostrar relaciones entre clases. Una clase que sea una subclase de otra clase se conecta con ella mediante una flecha con una línea sólida y con una punta triangular hueca, tal relación se llama generalización.

Una asociación entre dos clases significa que existe una relación estructural entre ellas. Las asociaciones se representan mediante líneas sólidas. Una asociación tiene muchas partes opcionales. Puede etiquetarse, así como cada una de sus terminaciones, para indicar el papel de cada clase en la asociación.

Una relación de dependencia representa otra conexión entre clases y se indica mediante una línea punteada (con flechas opcionales en los extremos y con etiquetas opcionales). Una clase depende de otra si los cambios en la segunda clase pueden requerir cambios en la primera. Una asociación de una clase con otra automáticamente indica una dependencia. No se necesitan líneas punteadas entre clases si ya existe una asociación entre ellas. Sin embargo, para una relación transitoria (es decir, una clase que no mantiene alguna conexión de largo plazo con otra, sino que usa dicha clase de manera ocasional), debe dibujarse una línea punteada desde la primera clase hasta la segunda.

2.6.2 DISEÑO

2.6.2.1 Diagrama de componentes

El diagrama de componentes UML es un conglomerado de figuras de los diagramas de casos de uso, estados, secuencias, colaboraciones y actividades. En lugar de representar una entidad conceptual como una clase o estado, un diagrama de componentes representa a un elemento real: un componente de software. Estos componentes se encuentran en las computadoras, no en la mente del analista. (Frederick & Brooks, 1987)

Un componente puede accederse a través de su interfaz, una colección de operaciones. La relación entre un componente y su interfaz se llama realización. Un componente puede acceder los servicios de otro. Cuando se hace, utiliza una interfaz de importación. El componente que realiza la interfaz con tales servicios proporciona una interfaz de exportación.

La representación de un componente es un rectángulo con otros dos rectángulos pequeños sobrepuestos en su lado izquierdo. Se puede representar una interfaz de dos formas: la primera es un rectángulo que contiene información de la interfaz y se conecta con el componente mediante una línea discontinua con una punta de flecha representada por triángulo sin relleno. La otra es un pequeño círculo conectado al componente con una línea continua. Ambos tipos de conexión pretenden mostrar una relación de realización.

2.6.2.2 Diagrama de objetos

Los Diagramas de Objetos están vinculados con los Diagramas de Clases. Un objeto es una instancia de una clase, por lo que un diagrama de objetos puede ser visto como una instancia de un diagrama de clases. Los diagramas de objetos describen la estructura estática de un sistema en un momento particular y son

usados para probar la precisión de los diagramas de clases. (Frederick & Brooks, 1987)

Los diagramas de objetos se generan en las disciplinas de Arquitectura y diseño. Se utilizan para mostrar estructuras de datos y las interacciones que existen entre objetos en tiempo de ejecución.

2.6.2.3 Diagrama de actividades

El diagrama de actividades es muy parecido a un diagrama de flujo. Muestra los pasos, puntos de decisión y bifurcaciones. Este tipo de diagrama es útil para representar las operaciones de un objeto y los procesos de negocios. (Frederick & Brooks, 1987)

El diagrama de actividades es una extensión del diagrama de estados. Los diagramas de estados destacan los estados y representan actividades como flechas entre los estados. Los de actividad se enfocan en las actividades. Cada actividad se representa como un rectángulo con esquinas redondeadas, más ovalados en apariencia que la representación de un estado. El diagrama de actividades utiliza los mismos símbolos que el de estados para los puntos de inicio y final.

En un diagrama de actividades, puede representar las actividades de acuerdo con la responsabilidad asignada. Esto lo haría con marcos de responsabilidad, mismos que son segmentos paralelos que corresponden a los responsables de realizar cada tarea. Es posible combinar al diagrama de actividades con símbolos de otros diagramas con lo que se producirán diagramas híbridos.

2.7 ESTANDAR DE CALIDAD ISO 9126

El estándar ISO 9126 se desarrolló con la intención de identificar los atributos claves

del software de cómputo. La tabla siguiente muestra las preguntas centrales que atiende cada una de estas características. Confiabilidad, Grado en el que se espera que un programa cumpla con su función y con la precisión requerida, se pueden usar las siguientes ecuaciones:

Tabla 2.3 Características de la Normas ISO-9126

Características	Preguntas
Funcionabilidad	¿Las funciones y propiedades satisfacen las necesidades explícitas e implícitas?
Confiabilidad	¿Puede mantener el nivel de rendimiento, bajo ciertas condiciones y por cierto tiempo?
Usabilidad	¿El software es fácil de usar y de aprender?
Eficiencia	¿Es rápido y minimalista en cuanto al uso de recursos?
Mantenibilidad	¿Es fácil de modificar y verifica?
Portabilidad	¿Es fácil de transferir de un ambiente a otro?

Fuente: Abud Figueroa, s.f.

El estándar ISO-9126 establece que cualquier componente de la calidad del software puede ser descrito en términos de seis atributos cada una de las cuales se detalla a través de un conjunto de sub-atributos que permiten analizar y profundizar en la evaluación de la calidad de productos de software. A continuación, se explican estos atributos detalladamente:

Funcionalidad: En este grupo se conjunta una serie de atributos que permiten calificar si un producto de software maneja en forma adecuada el conjunto de funciones que satisfagan las necesidades para las cuales fue diseñado. Para este propósito se establecen los siguientes atributos:

- **Adaptabilidad.** Se enfoca a evaluar si el software cuenta con un conjunto de funciones apropiadas para efectuar las tareas que fueron especificadas en su definición.

- **Exactitud.** Este atributo permite evaluar si el software presenta resultados o efectos acordes a las necesidades para las cuales fue creado.
- **Interoperabilidad.** Permite evaluar la habilidad del software de interactuar con otros sistemas previamente especificados.
- **Conformidad.** Evalúa si el software se adhiere a estándares, convenciones o regulaciones en leyes y prescripciones similares.
- **Seguridad.** Se refiere a la habilidad de prevenir el acceso no autorizado, ya sea accidental o premeditado, a los programas y datos.

Confiabilidad: Aquí se agrupan un conjunto de atributos que se refieren a la capacidad del software de mantener su nivel de ejecución bajo condiciones normales en un periodo de tiempo establecido. Las sub características que el estándar sugiere son:

- **Nivel de Madurez.** Permite medir la frecuencia de falla por errores en el software.
- **Tolerancia a fallas.** Se refiere a la habilidad de mantener un nivel específico de funcionamiento en caso de fallas del software o de cometer infracciones de su interfaz específica.
- **Recuperación.** Se refiere a la capacidad de restablecer el nivel de operación y recobrar los datos que hayan sido afectados directamente por una falla, así como al tiempo y el esfuerzo necesarios para lograrlo.

Usabilidad: Consiste de un conjunto de atributos que permiten evaluar el esfuerzo necesario que deberá invertir el usuario para utilizar el sistema.

- **Comprensibilidad.** Se refiere al esfuerzo requerido por los usuarios para reconocer la estructura lógica del sistema y los conceptos relativos a la aplicación del software.
- **Facilidad de Aprender.** Establece atributos del software relativos al esfuerzo que los usuarios deben hacer para aprender a usar la aplicación.

- **Operabilidad.** Agrupa los conceptos que evalúan la operación y el control del sistema.

Eficiencia: Esta característica permite evaluar la relación entre el nivel de funcionamiento del software y la cantidad de recursos usados. Los aspectos a evaluar son:

- **Comportamiento con respecto al Tiempo.** Atributos del software relativos a los tiempos de respuesta y de procesamiento de los datos.
- **Comportamiento con respecto a Recursos.** Atributos del software relativos a la cantidad de recursos usados y la duración de su uso en la realización de sus funciones.

Mantenibilidad: Se refiere a los atributos que permiten medir el esfuerzo necesario para realizar modificaciones al software, ya sea por la corrección de errores o por el incremento de funcionalidad. En este caso, se tienen los siguientes factores:

- **Capacidad de análisis.** Relativo al esfuerzo necesario para diagnosticar las deficiencias o causas de fallas, o para identificar las partes que deberán ser modificadas.
- **Capacidad de modificación.** Mide el esfuerzo necesario para modificar aspectos del software, remover fallas o adaptar el software para que funcione en un ambiente diferente.
- **Estabilidad.** Permite evaluar los riesgos de efectos inesperados debidos a las modificaciones realizadas al software.
- **Facilidad de Prueba.** Se refiere al esfuerzo necesario para validar el software una vez que fue modificado.

Portabilidad: En este caso, se refiere a la habilidad del software de ser transferido de un ambiente a otro, y considera los siguientes aspectos:

- **Adaptabilidad.** Evalúa la oportunidad para adaptar el software a diferentes ambientes sin necesidad de aplicarle modificaciones.
- **Facilidad de Instalación.** Es el esfuerzo necesario para instalar el software en un ambiente determinado.
- **Conformidad.** Permite evaluar si el software se adhiere a estándares o convenciones relativas a portabilidad.
- **Capacidad de reemplazo.** Se refiere a la oportunidad y el esfuerzo usado en sustituir el software por otro producto con funciones similares. (Abud Figueroa, sin fecha)

2.8 MODELO CONSTRUCTIVO DE COSTOS (COCOMO II)

El Modelo Constructivo de Costes (*CO*nstructive *CO*st *MO*del) fue desarrollado por B. W. Boehm a finales de los 70 y comienzos de los 80. En su libro clásico acerca de “economía de la ingeniería de software”, Barry Boehm introdujo una jerarquía de modelos de estimación de software que llevan el nombre COCOMO, por *Constructive Cost Model*: modelo constructivo de costos. (Gómez, 2006)

2.8.1 COCOMO II

Los objetivos principales que se tuvieron en cuenta para construir el modelo COCOMO II fueron:

- Desarrollar un modelo de estimación de costo y cronograma de proyectos de software que se adaptara tanto a las prácticas de desarrollo de la década del 90 como a las futuras.
- Construir una base de datos de proyectos de software que permitiera la calibración.
- continuar del modelo, y así incrementar la precisión en la estimación.
- Implementar una herramienta de software que soportara el modelo.

- Proveer un marco analítico cuantitativo y un conjunto de herramientas y técnicas que evaluaran el impacto de las mejoras tecnológicas de software sobre los costos y tiempos en las diferentes etapas del ciclo de vida de desarrollo.

Figura 2.2 Distribución del mercado de software actual y futuro

Aplicaciones desarrolladas por usuarios finales		
Generadores de Aplicaciones	Aplicaciones con Componentes	Sistemas Integrados
Infraestructura		

Fuente: Distribucion del mercado de software actual y futuro (Gómez, López, Migani & Otazú, s. f.),

Aplicaciones desarrolladas por Usuarios Finales: En este sector se encuentran las aplicaciones de procesamiento de información generadas directamente por usuarios finales, mediante la utilización de generadores de aplicaciones tales como planillas de cálculo, sistemas de consultas, etc. Estas aplicaciones surgen debido al uso masivo de estas herramientas, conjuntamente con la presión actual para obtener soluciones rápidas y flexibles.

Generadores de Aplicaciones: En este sector operan firmas como Lotus, Microsoft, Novell, Borland con el objetivo de crear módulos pre-empaquetados que serán usados por usuarios finales y programadores.

Aplicaciones con Componentes: Sector en el que se encuentran aquellas aplicaciones que son específicas para ser resueltas por soluciones pre empaquetadas, pero son lo suficientemente simples para ser construidas a partir de componentes interoperables. Componentes típicas son constructores de interfases

gráficas, administradores de bases de datos, buscadores inteligentes de datos, componentes de dominio-específico (medicina, finanzas, procesos industriales, etc.). Estas aplicaciones son generadas por un equipo reducido de personas, en pocas semanas o meses.

Sistemas Integrados: Sistemas de gran escala, con un alto grado de integración entre sus componentes, sin antecedentes en el mercado que se puedan tomar como base. Porciones de estos sistemas pueden ser desarrolladas a través de la composición de aplicaciones. Entre las empresas que desarrollan software representativo de este sector, se encuentran grandes firmas que desarrollan software de telecomunicaciones, sistemas de información corporativos, sistemas de control de fabricación, etc.

Infraestructura: Área que comprende el desarrollo de sistemas operativos, protocolos de redes, sistemas administradores de bases de datos, etc. Incrementalmente este sector direccionará sus soluciones, hacia problemas genéricos de procesamiento distribuido y procesamiento de transacciones, a soluciones middleware. Firmas representativas son Microsoft, Oracle, SyBase, Novell y NeXT.

COCOMO II está compuesto por tres modelos denominados: Composición de Aplicación, Diseño Temprano y Post-Arquitectura. Éstos surgen en respuesta a la diversidad del mercado actual y futuro de desarrollo de software, cada uno de los cuales ofrece mayor fidelidad a medida que uno avanza en la planificación del proyecto y en el proceso de diseño. Estos tres modelos se denominan:

El modelo Composición de Aplicación se emplea en desarrollos de software durante la etapa de prototipación.

El modelo Diseño Temprano se utiliza en las primeras etapas del desarrollo en las cuales se evalúan las alternativas de hardware y software de un proyecto. En estas

etapas se tiene poca información, lo que concuerda con el uso de Puntos Función, para estimar tamaño y el uso de un número reducido de factores de costo.

El modelo Post-Arquitectura se aplica en la etapa de desarrollo propiamente dicho, después que se define la arquitectura del sistema, y en la etapa de mantenimiento. Este modelo utiliza:

- Puntos Función y/o Líneas de Código Fuente para estimar tamaño, con modificadores que contemplan el reusó, con y sin traducción automática, y el “desperdicio” (*breakage*).
- Un conjunto de 17 atributos, denominados factores de costo, que permiten considerar características del proyecto referentes al personal, plataforma de desarrollo, etc., que tienen injerencia en los costos.
- Cinco factores que determinan un exponente, que incorpora al modelo el concepto de deseconomía y economía de escala. Estos factores reemplazan los modos Orgánico, Semiacoplado y Empotrado del modelo COCOMO 81.

2.8.1.1 Estimación del esfuerzo

El esfuerzo necesario para concretar un proyecto de desarrollo de software, cualquiera sea el modelo empleado, se expresa en meses/persona (PM) y representa los meses de trabajo de una persona *full time*, requeridos para desarrollar el proyecto. (Gómez, 2006)

En el modelo de diseño temprano y post-arquitectura se basan en la en el desarrollo de la misma fórmula para calcular la estimación del esfuerzo requerido. A continuación, se detalla dicha formula.

$$PM_{nominal} = A(Size)^B$$

Donde:

PMnominal es el esfuerzo expresados en meses persona.

A es una constante que captura los efectos lineales sobre el esfuerzo de acuerdo a la variación del tamaño. Provisionalmente se le ha estimado un valor de 2.45

Size, Donde:

$$Size = Size \left(1 + \frac{BRAK}{100} \right)$$

Donde:

- Size, cantidad de instrucciones del proyecto estimado. Si se opta por puntos función para calcular el tamaño del proyecto, estos deben convertirse en líneas de código de fuente en el lenguaje de implementación.
- BRAK, porcentaje de código desperdiciado debido a la volatilidad de los requisitos.

B es el factor exponencial de escala, toma en cuenta las características relacionadas con las economías y diseconomías de escala producidas cuando un proyecto de software incrementa su tamaño. El exponente B se usa para capturar estos efectos.

- Si $B < 1.0$, El proyecto presenta ahorros de escala. Si el tamaño del producto se dobla, el esfuerzo del proyecto es menor que el doble. La productividad del proyecto aumenta a medida que aumenta el tamaño del producto.
- Si $B = 1.0$, los ahorros y gastos de escala están equilibrados. Este modelo lineal se usa a menudo para la estimación de costo de proyectos pequeños. Se usa para el modelo COCOMO II: composición de aplicaciones.
- Si $B > 1.0$, el proyecto presenta gastos de escala. Esto se debe normalmente a dos factores principales: El crecimiento del gasto en comunicaciones y el gasto

en crecimiento de la integración de un gran sistema. Los proyectos más grandes tendrán más personal y por lo tanto más vías de comunicación interpersonales produciendo gasto.

A continuación, se muestra la formula del cálculo de la variable B.

$$B = 0.91 + 0.01 \sum_{j=1}^S SFj$$

El exponente B se obtiene mediante los denominados factores de escala

Tabla 2.4 Valores de escala

Factores de escala	Muy bajo	Bajo	nominal	alto	Muy alto	Extra alto
PREC (Precedencia de Desarrollo)	6.2	4.96	3.72	2.48	1.24	0
FLEX (Flexibilidad de Desarrollo)	5.07	4.05	3.04	2.03	1.01	0
RESL (Arquitectura/Resolución de Riesgo)	7.07	5.65	4.24	2.83	1.41	0
TEAM (Cohesión de equipo)	5.48	4.38	3.29	2.19	1.1	0
PMAT (Madurez de proceso)	7.8	6.24	4.68	3.12	1.56	0

Fuente: Valores de escala (Gómez, López, Migani & Otazú, s. f.),

La selección de los factores de escala se basa en la razón de que ellos son un recurso significante de variación exponencial en un esfuerzo o variación de la productividad del proyecto.

2.8.1.2 Ajuste mediante los atributos de costo

Los atributos de costos se usan para capturar características de desarrollo del software que afectan al esfuerzo para completar el proyecto. Los atributos de costo

tienen un nivel de medida que expresa el impacto del atributo en el desarrollo. Estos valores pueden ir desde extra bajo hasta extra alto.

Para el propósito del análisis cuantitativo, cada nivel de medida de cada atributo de costo tiene un peso asociado. El peso se llama multiplicador de esfuerzo (EM). La medida asignada a un atributo de costo es 1.0 y el nivel de medida asociado con ese peso se llama nominal. Si un nivel de medida produce más esfuerzo de desarrollo de software, entonces su correspondiente EM está por encima de 1.0.

Recíprocamente si el nivel de medida reduce el esfuerzo entonces el correspondiente EM es menor que 1.0. La selección de multiplicadores de esfuerzo de proyecto o variación de la productividad independientemente. Los EM se usan para ajustar el esfuerzo Meses-persona nominal.

La fórmula para calcular el esfuerzo ajustado es la siguiente:

$$MM = A(Size)^B \prod EM$$

Hay 17 multiplicadores de esfuerzo para el modelo de Post-Arquitectura.

Tabla 2.5 Atributos de costo para el modelo Post-Arquitectura

Grupo	Atributo	Descripción
Producto	RELY	Fiabilidad requerida del software.
	DATA	Medida del volumen de datos.
	CPLX	Complejidad del producto.
	RUSE	Reutilización requerida.
	DOCU	Documentación asociada a los ciclos de vida.
Hardware	TIME	Restricción de tiempo de ejecución.
	STOR	Restricción del almacenamiento principal.
	PVOL	Volatilidad de la máquina virtual.
	ACAP	Capacidad del analista.

Personal	PCAP	Habilidad del programador.
	AEXP	Experiencia en aplicaciones.
	PEXP	Experiencia en plataforma.
	LTEX	Experiencia en la herramienta y el lenguaje.
	PCON	Continuidad del personal.
Proyecto	SITE	Desarrollo multilugar.
	TOOL	Utilización de herramientas software.
	SCED	Calendario de desarrollo requerido.

Fuente: Atributos de costo para el modelo Post-Arquitectura (Gómez, López, Migani & Otazú, s. f.)

La siguiente tabla muestra los valores de los atributos de costos para el modelo Post-Arquitectura.

Tabla 2.6 Valores de atributos de costos

	Muy bajo	Bajo	Nominal	Alto	Muy alto	Extra alto
RELY	0.82	0.92	1.00	1.10	1.26	0.00
DATA	0.00	0.90	1.00	1.14	1.28	0.00
CPLX	0.73	0.87	1.00	1.17	1.34	1.74
RUSE	0.00	0.95	1.00	1.07	1.15	1.24
DOCU	0.81	0.91	1.00	1.11	1.23	0.00
TIME	0.00	0.00	1.00	1.11	1.29	1.63
STOR	0.00	0.00	1.00	1.05	1.17	1.46
PVOL	0.00	0.87	1.00	1.15	1.30	0.00
ACAP	1.42	1.19	1.00	0.85	0.71	0.00
PCAP	1.34	1.15	1.00	0.88	0.76	0.00
PCON	1.29	1.12	1.00	0.90	0.81	0.00
AEXP	1.22	1.10	1.00	0.88	0.81	0.00
PEXP	1.19	1.09	1.00	0.91	0.85	0.00
LTEX	1.20	1.09	1.00	0.91	0.84	0.00
TOOL	1.17	1.09	1.00	0.90	0.78	0.00
SITE	1.22	1.09	1.00	0.93	0.86	0.80
SCED	1.43	1.14	1.00	1.00	1.00	0.00

Fuente: Valores de atributos de costos (Gómez, López, Migani & Otazú, s. f.)

2.8.1.3 Tiempo de desarrollo

La versión inicial de COCOMO II proporcionaba una capacidad de estimación de tiempo simplemente similar a las de COCOMO. La fórmula para determinar el tiempo de desarrollo es la siguiente:

$$TDEV = (3.67(PM)^{(0.28+0.2(B-1.01))}) \frac{SCED\%}{100}$$

Donde:

- TDEV es el tiempo calendario en meses que transcurre desde la determinación de los requerimientos a la culminación de una actividad que certifique que el producto cumple con las especificaciones.
- PM es el esfuerzo expresado en meses personas, calculado sin tener en cuenta el multiplicador de esfuerzo SCED.
- B es la suma del Factor de Escala.

SCED% es el porcentaje de compresión/expansión del cronograma.

CAPÍTULO III

MARCO APLICATIVO

3 MARCO APLICATIVO

3.1 PLAN DE DESARROLLO

Para este proyecto se ha establecido en siguiente plan de desarrollo:

Tabla 3.1 Plan de desarrollo

Artefactos generados y modificados durante la fase de inicio	Comienzo	Aprobación
Modelado de negocio	Semana 1	Semana 2
Modelado de casos de uso del negocio	Semana 1	Semana 2
Requerimientos funcionales	Semana 2	Semana 3
Requerimientos no funcionales	Semana 2	Semana 3
Requisitos		
Visión	Semana 1	Semana 2
diagrama de casos de uso de estado actual del sistema	Semana 2	Siguiente fase
Especificaciones de casos de uso	Semana 3	Siguiente fase
Especificaciones y adicionales	Semana 4	Siguiente fase
Análisis y diseño		
Modelado de análisis	Semana 7	Siguiente fase
Diagrama de casos de uso del nuevo sistema	Semana 2	Semana 3
Diagrama de secuencias	Semana 3	Semana 4
Diagrama de actividades	Semana 4	Semana 5
Diagrama Entidad Relación	Semana 5	Semana 7
Diagrama de clases	Semana 7	Siguiente fase
Modelado de diseño	Semana 4	Siguiente fase
Diagrama navegacional	Semana 1	Semana 2
Diagrama de interfaz	Semana 2	Semana 4
Pruebas		
Casos funcionales	Semana 1	Semana 2

Elaboración: propia

3.2 ANALISIS DE LA SITUACIÓN ACTUAL DEL SISTEMA.

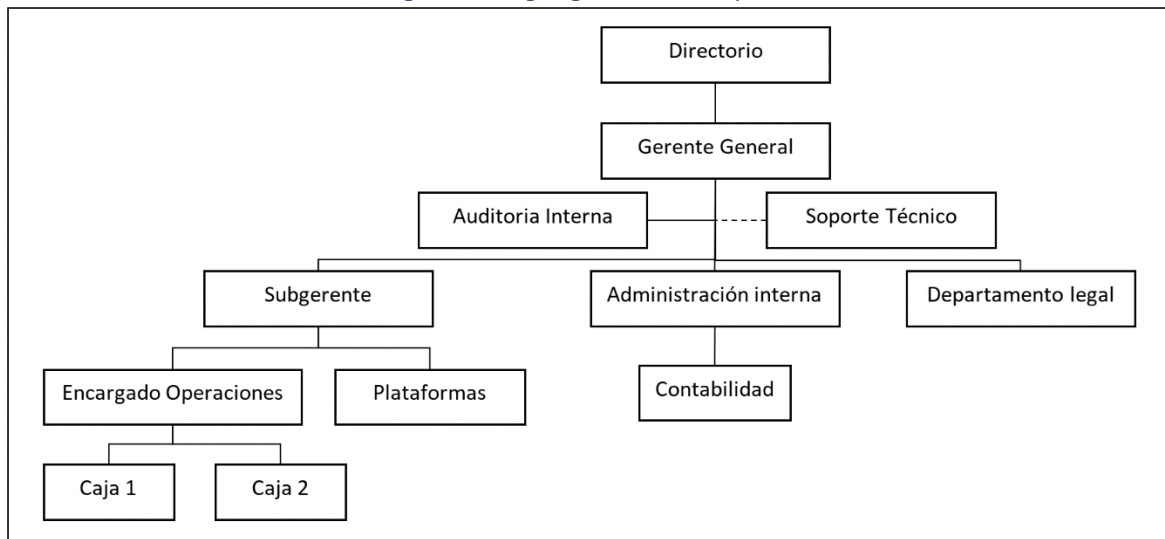
La Caja Popular Boliviana S.R.L. es una institución financiera que se dedica a prestar estos servicios pagos de bono Juana Azurduy, Renta dignidad, cambio de moneda, créditos personales y créditos hipotecarios, para los cuales utilizan garantías personales o hasta garantías de hipotecarias de folio real.

Los clientes tienen que cumplir los requisitos para solicitar un prestamos en la caja popular boliviana que son los siguientes:

- Fotocopia cédula de identidad.
- Fuente de pago (sueldos y salarios, utilidades y/o otros ingresos respaldados).
- Estabilidad Laboral: 1 año.
- Carga Financiera Total/Renta Total: Según Manual de políticas de crédito.
- Documentación de respaldo de deudas vigentes en el sistema financiero del titular, cónyuge y garantes (si aplica).
- Documentación de respaldo de cajas de ahorro o cuentas corrientes del Titular, cónyuge y garantes (si aplica).
- Llenado de Formulario de Solicitud de Crédito.
- Llenado de Formulario de Declaración Patrimonial (adjuntar respaldo de activos declarados).

La jerarquía organizacional de la empresa se muestra en la siguiente figura:

Figura 3.1 Organigrama de la empresa

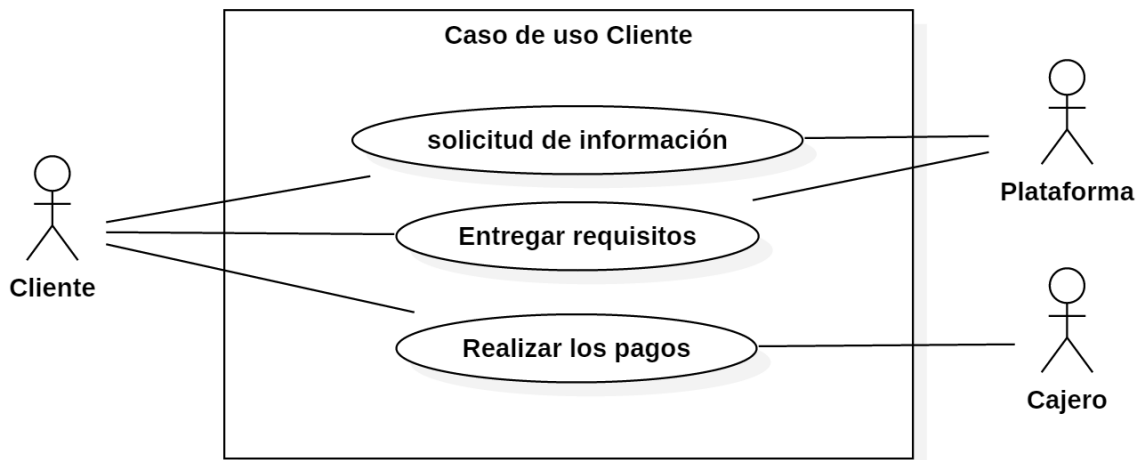


Fuente y elaboración: Caja Popular Boliviana SRL (S. Jimenez, 2012).

Identificación de actores

La descripción del funcionamiento que se lleva en actual puede apreciarse en las siguientes figuras:

Figura 3.2 Diagrama caso de uso cliente

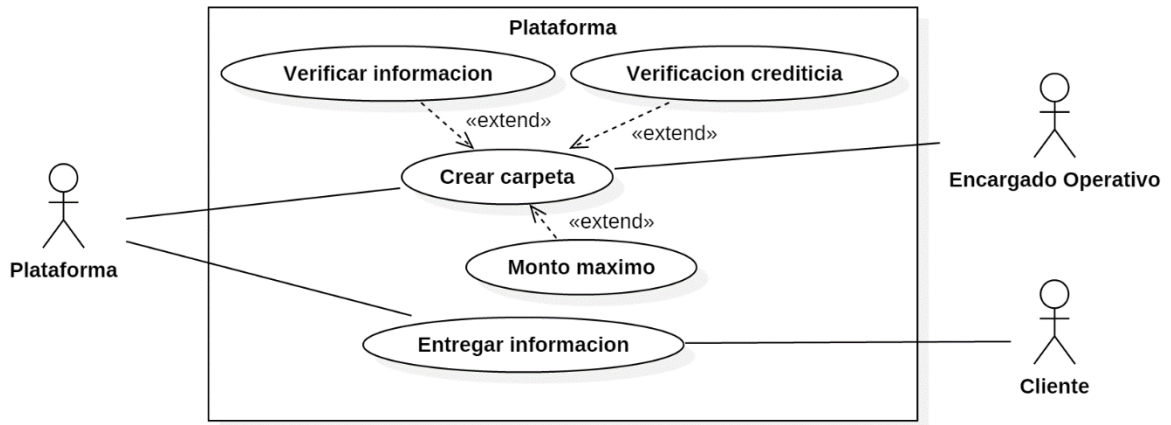


Fuente y elaboración: Propia

Casos de uso del cliente: el cliente solicita la información de los requisitos de los préstamos que otorga la institución, escoge el tipo de préstamo que aspira a sacar

recaba los requisitos solicitados, para presentar la solicitud del préstamo. Una vez otorgado el préstamo el cliente empieza a pagar según el plan de pagos asignado.

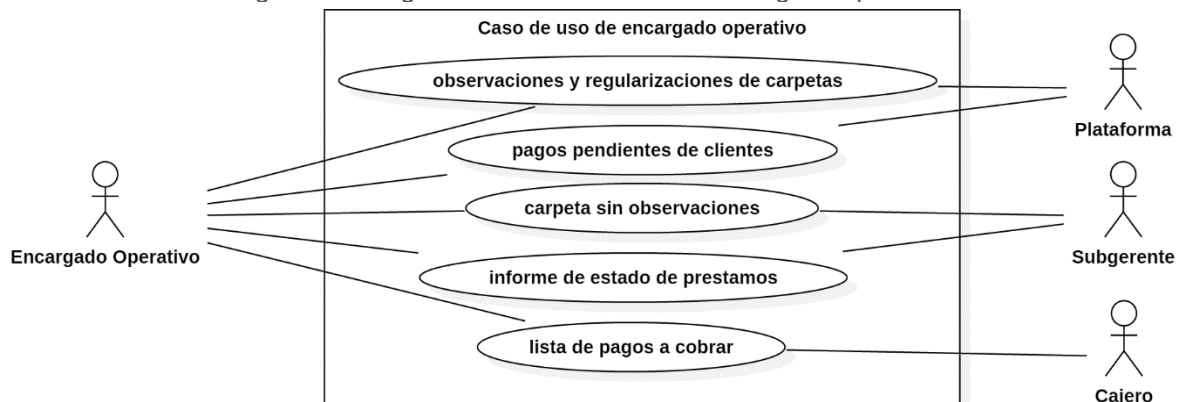
Figura 3.3 Diagrama de caso de uso de plataforma



Fuente y elaboración: Propia

Casos de uso plataforma: Plataforma proporciona la información de los tipos de préstamos que otorga la institución, los requisitos que son necesarios para cada prestamos, crea una carpeta para realizar la verificación de la información también la comprobación crediticia de él o de los clientes y otorga el monto máximo que puede optar a dar la institución, realiza el registro de los clientes con el crédito y realizan el cobro de mora.

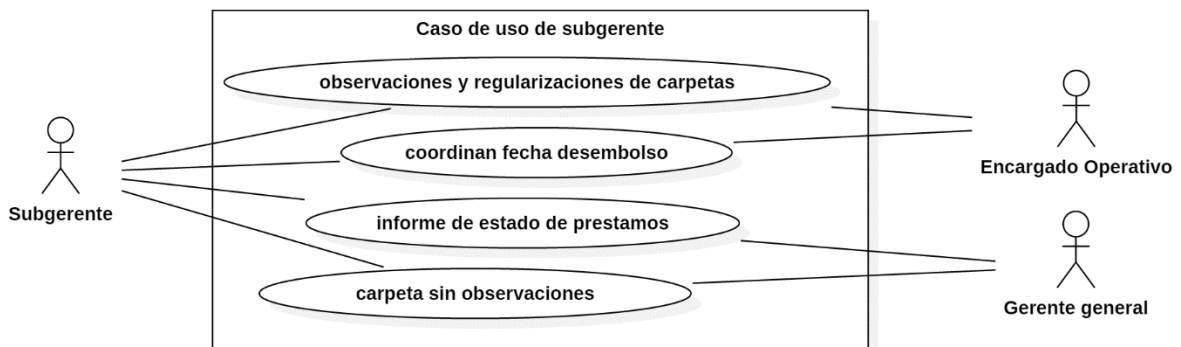
Figura 3.4 Diagrama de caso de uso de encargado operativo.



Fuente y elaboración: Propia

Casos de uso del encargado operativo: Realiza la verificación de la carpeta en caso de presentar observaciones y regularizaciones se envía a plataforma para corregir la información o completarla para que todos los datos estén en orden para proceder él envió a subgerencia coordina el desembolso, realiza un informe de los estados de préstamos y presenta información al cajero de nuevos pagos por realizar.

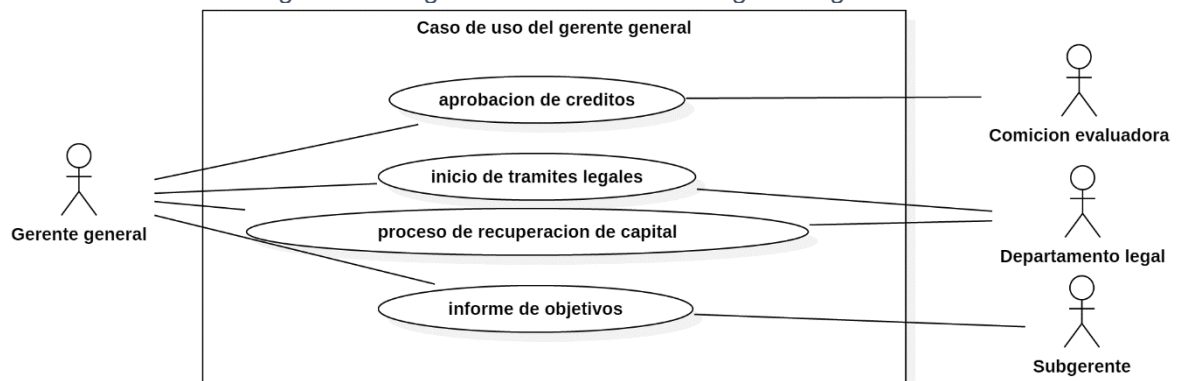
Figura 3.5 Diagrama de caso de uso de encargado operativo.



Fuente y elaboración: Propia

Casos de uso de subgerente: Realiza la verificación de la carpeta en caso de presentar observaciones y regularizaciones se envía a encargado operativo para corregir plan de créditos, monto de préstamo o seguro, poniendo se ha coordinación programan una fecha desembolso, presenta un informe mostrando el estado de los préstamos de cada trimestre al gerente general y presenta nuevos créditos posibles.

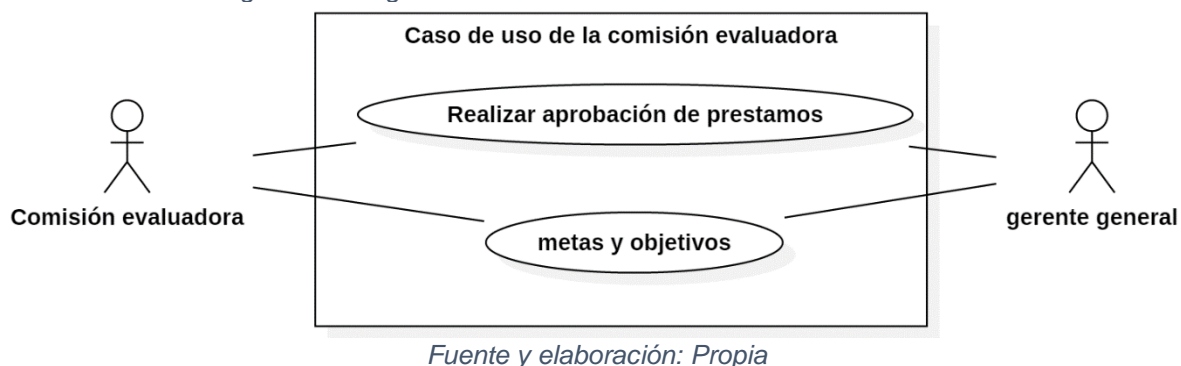
Figura 3.6 Diagrama de caso de uso de gerente general.



Fuente y elaboración: Propia

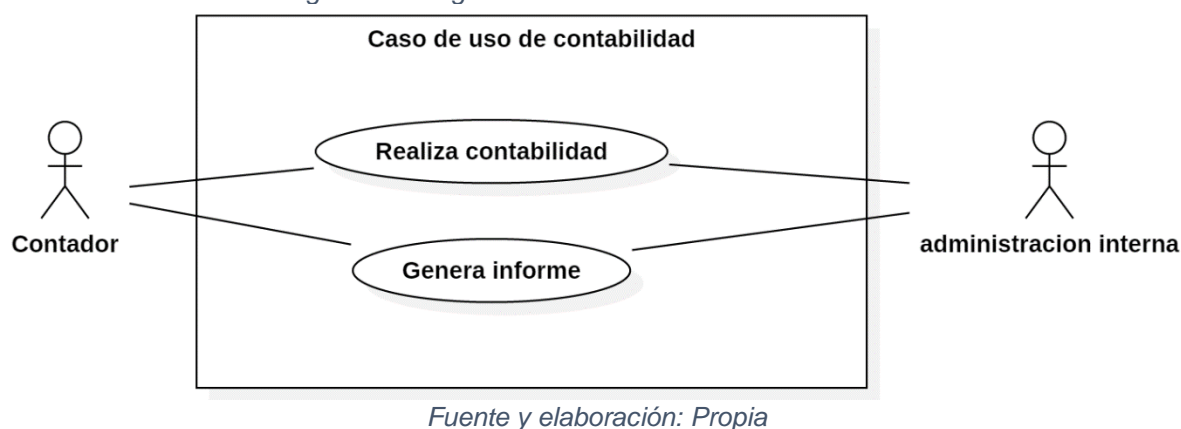
Casos de uso del gerente general: procede a la votación con la comisión evaluadora depende de la votación si el préstamo procede al inicio de trámite legal (reconocimiento de firmas, elaboración de contrato y/o registro de hipotecario en derechos reales), los préstamos que exceden el límite de mora empiezan el proceso de recuperación de capital embargo y remate de los bienes que se tomaron como garantía.

Figura 3.7 Diagrama de caso de uso de la comisión evaluadora.



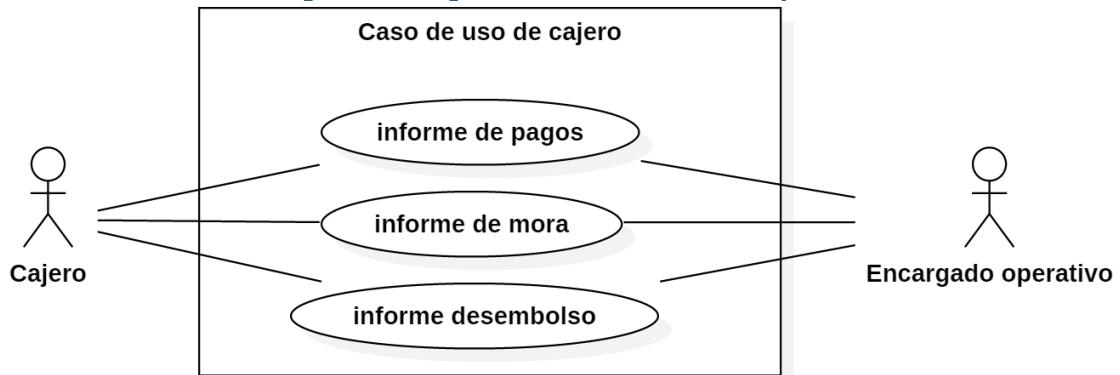
Casos de uso de la comisión evaluadora: se realiza una votación y la verificación mediante parámetros de posibilidad de devolución del capital de los clientes. Presentas las metas y objetivos con el capital disponible para el siguiente trimestre.

Figura 3.8 Diagrama de caso de uso de contabilidad.



Caso de uso de contabilidad: Se lleva acabo la contabilidad cada trimestre y se generan los informes periódicos presentado los al administrador.

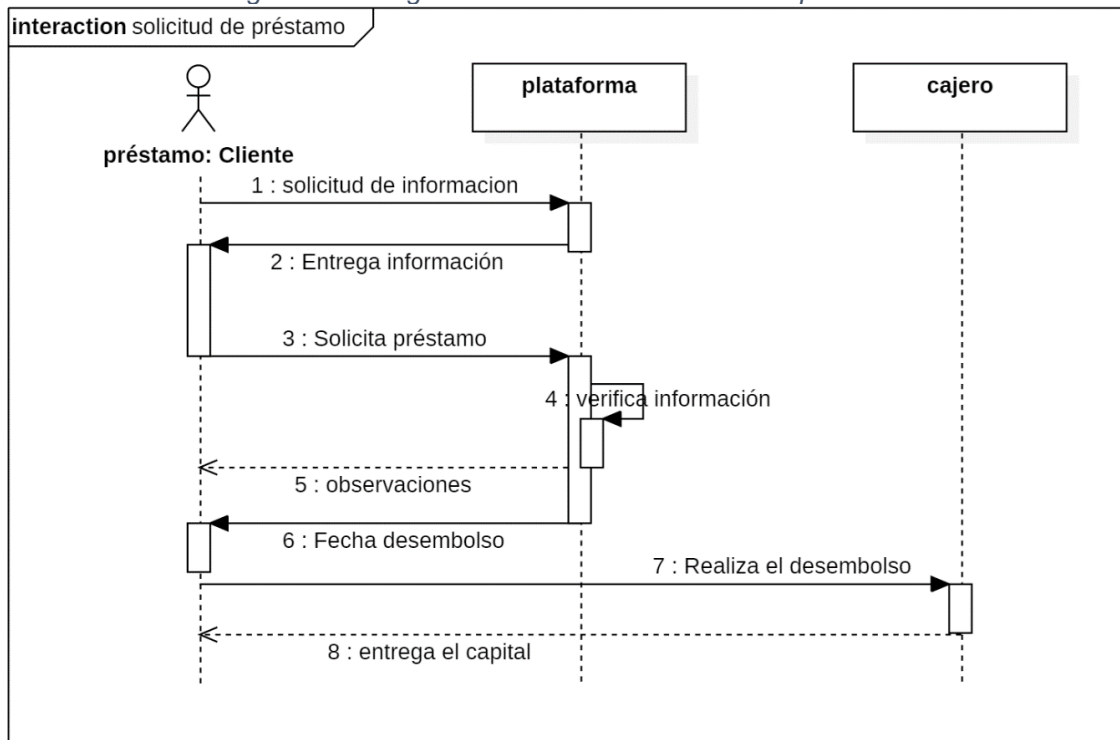
Figura 3.9 Diagrama de caso de uso de cajero.



Fuente y elaboración: Propia

Casos de uso del cajero: el cajero realiza los desembolsos programados, lo que es el cobro a los clientes y realiza el informe de créditos no cancelados, cancelados y desembolsos realizados.

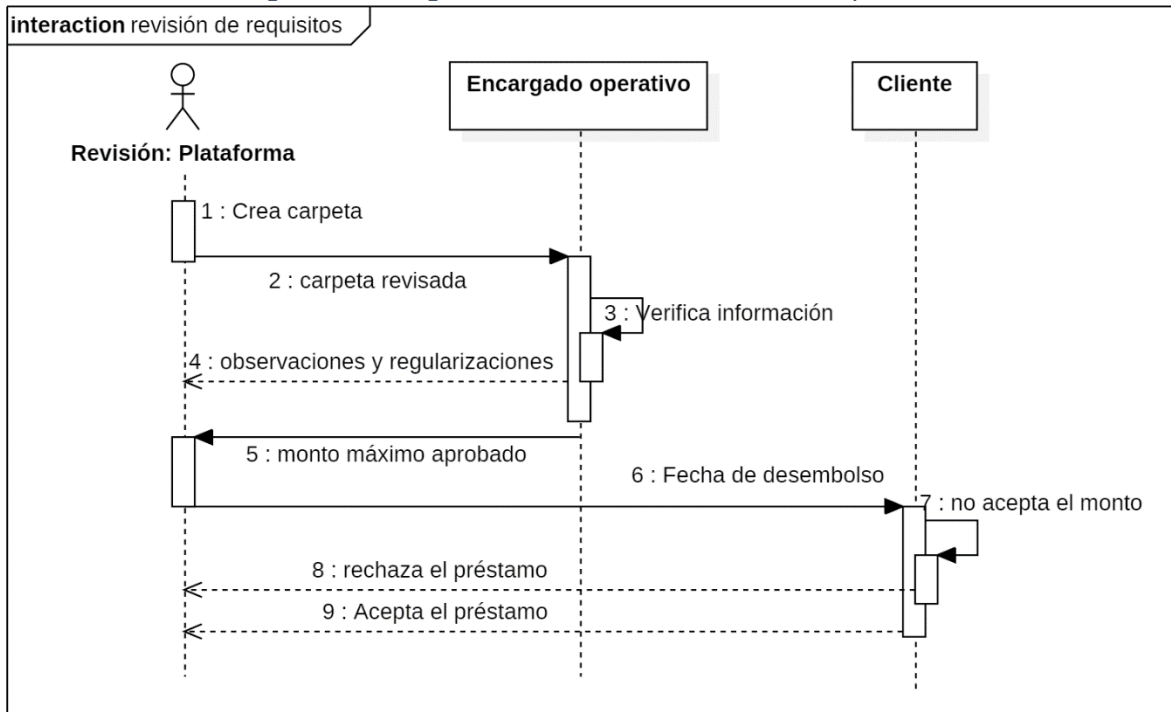
Figura 3.10 Diagrama de secuencia Solicitud de préstamo



Fuente y elaboración: Propia

El cliente solicita información a plataforma se entrega de forma manual, luego el cliente recauda los requisitos. Se solicita un préstamo si el cliente cumple con todos los requisitos, de lo contrario se da la respuesta al cliente de las observaciones. Se programa el desembolso y se entrega el capital.

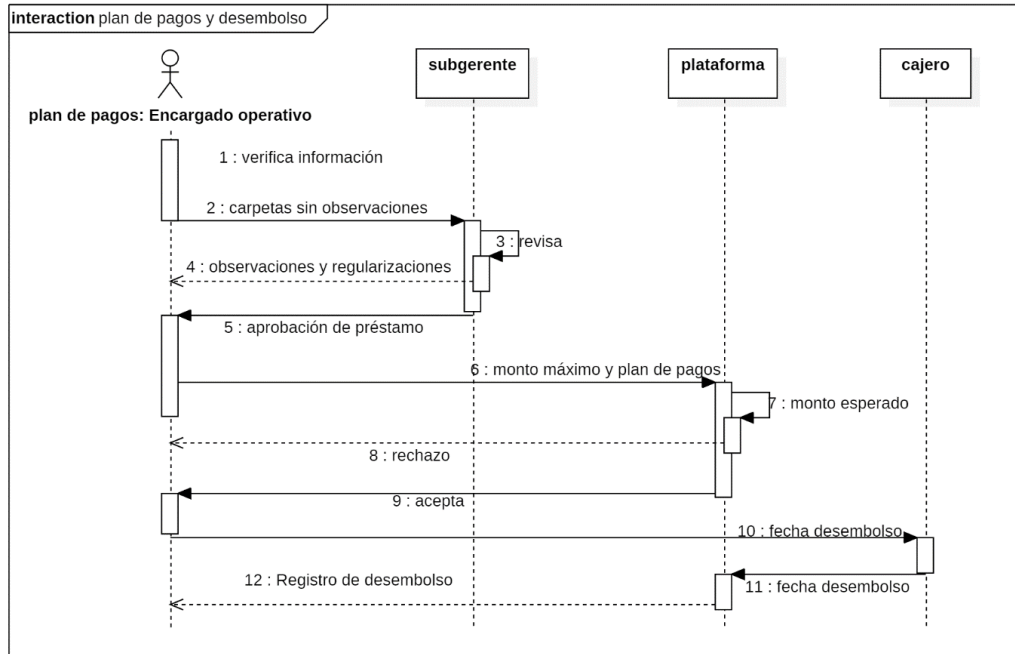
Figura 3.11 Diagrama de secuencia Revisión de requisitos



Fuente y elaboración: Propia

Plataforma empieza por la creación de la carpeta y revisa que los requisitos estén completos y la información crediticia, envía la carpeta revisada. El encargado operativo verifica que toda la información sea correcta y el monto máximo aprobado con el plan de pago. Plataforma muestra la oferta de crédito de la institución el cliente toma la decisión de aceptar o rechazar el crédito.

Figura 3.12 Diagrama de secuencia Plan de pagos y desembolso

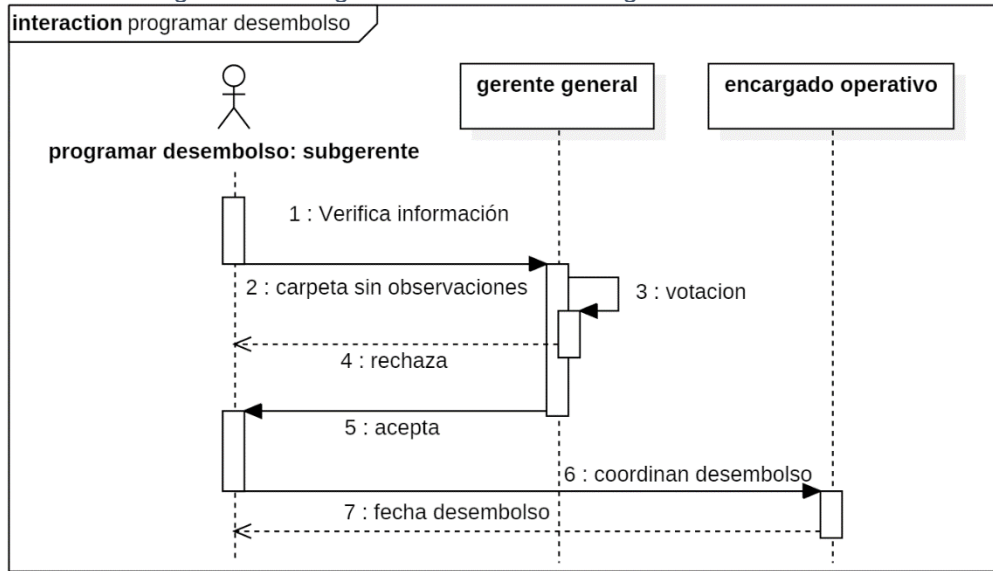


Fuente y

elaboración: Propia

El encargado operativo verifica información, envía la carpeta sin observaciones y revisa la información si presenta observaciones o regularizaciones se envía al encargado operativo para corregir la información. La aprobación de préstamo presenta el monto máximo disponible para el cliente si rechaza el monto no se hace nada más. Si acepta se programa una fecha desembolso, se notifica a plataforma y registra el desembolso.

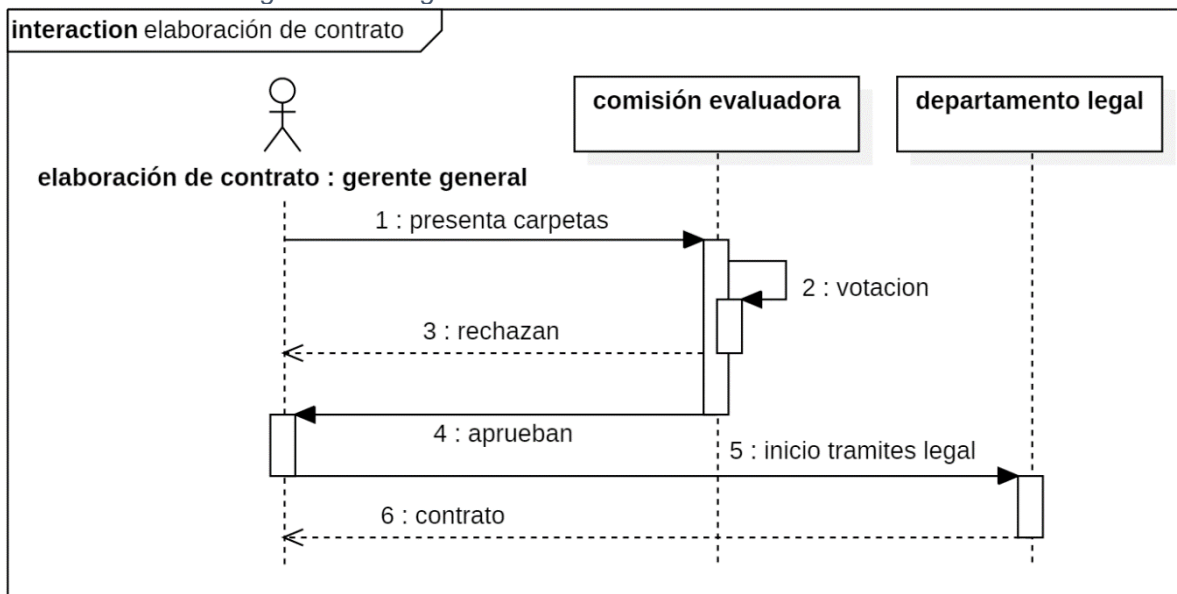
Figura 3.13 Diagrama de secuencia Programar desembolso



Fuente y elaboración: Propia

El subgerente verifica la información envía la carpeta sin observación a la gerencia si la respuesta es positiva envía el subgerente y el encargado operativo coordina desembolso, si la respuesta es negativa se rechaza el préstamo.

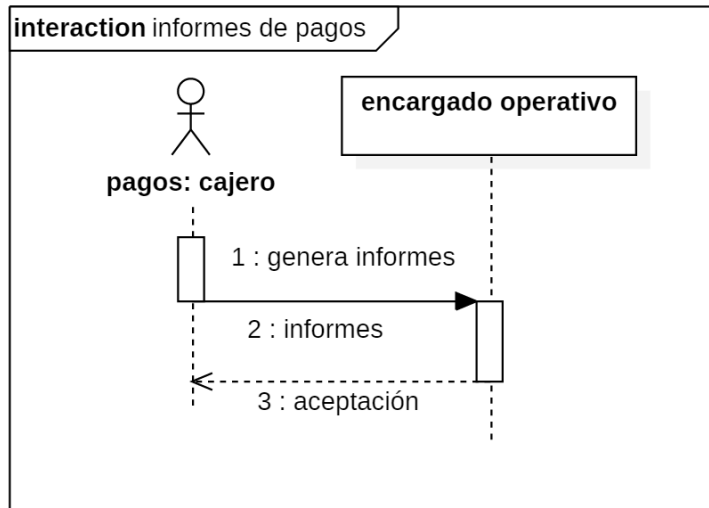
Figura 3.14 Diagrama de secuencia Elaboración de contrato



Fuente y elaboración: Propia

El gerente general presenta las carpetas a la comisión evaluadora da una respuesta de positiva se contacta con el departamento legal para iniciar los trámites legales, si rechazan el préstamo el proceso finaliza.

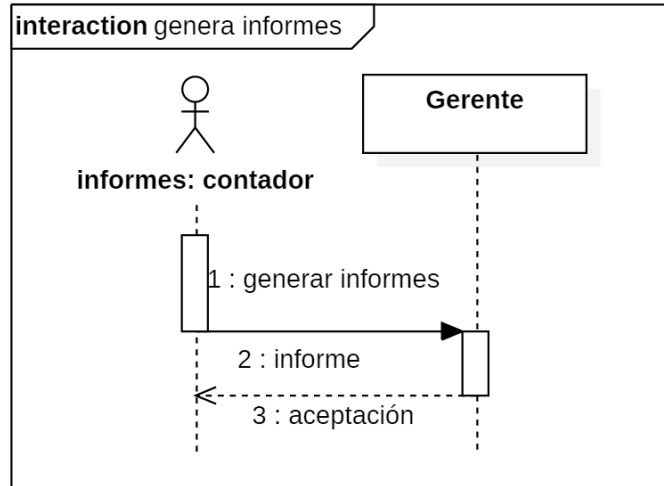
Figura 3.17 Diagrama de secuencia Informes de pagos



Fuente y elaboración: Propia

El cajero genera los informes de pagos cancelados, pagos no cancelados, desembolso y pagos en mora.

Figura 3.18 Diagrama de secuencia Genera informes



Fuente y elaboración: Propia

El contador realiza informa periódicamente que se entregan al gerente para su revisión.

De acuerdo a la información otorgada por el gerente y el resultado del diagnóstico de la situación actual de la empresa se evidencian falencias y mejoras al sistema actual, generando dos tipos de requisitos.

3.2.1 Requisitos funcionales y no funcionales

Los requisitos no funcionales:

- Rol de usuarios, Se crearán roles específicos para el acceso al sistema.
- Módulo de aprobación, para una mejor administración de los créditos la aprobación ayudara para ver en qué punto se encuentra cada crédito.
- Módulo de desembolso, el sistema de tiene la necesidad de programar los desembolsos para tener el dinero necesario para poder trabajar normal mente.
- Módulo de seguimiento de pagos, se llevará un control de pagos por fechas mostrando los pagos que se realizaran por día y llevar un historial de cada préstamo.
- Módulo de cálculo de mora, los pagos que están retrasados a la fecha de pago se le cobra un porcentaje extra sumado al pago mensual.
- Módulo de búsqueda, para realizar la búsqueda al momento de pago y así agilizar el proceso.
- Módulo de liquidación, se calificará el cumplimiento de pagos del cliente, para guardar un antecedente crediticio en la institución.
- Módulo de reportes, el sistema debe realizar el registro de todas las acciones como es el registro, desembolso, pagos retrasados, liquidación y seguimiento de pagos.
- Módulo de inversión, para el control de ingresos a la empresa mostrando cuanto ganará por cada préstamo.

Los requisitos no funcionales:

- Se requiere un servidor para la ejecución del proyecto.
- El sistema debe asegurar que los datos estén protegidos contra acceso no autorizados.
- El sistema debe proporcionar mensajes de error que sean informativos y orientados a usuario final.
- El nuevo sistema debe desarrollarse aplicando patrones y recomendaciones de programación que incrementen la seguridad de datos.
- El sistema debe contar con manuales técnicos que describan los módulos y procesos para una revisión, mejoramiento y/o modificación del mismo.

3.3 ANÁLISIS DEL SISTEMA FUTURO

El análisis del sistema futuro comprende varios procesos que proporcionarán la funcionalidad de los componentes del sistema a través de modelos y diagramas, los cuales serán ilustrados gráficamente consolidando una arquitectura y organización del contexto.

3.3.1 Descripción de funciones

A continuación, se describe cada una de las funciones que desempeñan los actores en la Caja Popular Boliviana srl.

Tabla 3.2 Descripción de funciones

Actor	Descripción
Cajero	Sus funciones son: <ul style="list-style-type: none"> • De registrar los pagos que ingresan. • Imprimir comprobante
Plataforma	Sus funciones son: <ul style="list-style-type: none"> • De registrar nuevas solicitudes • Imprimir plan de pagos • Imprimir contrato de prestamos

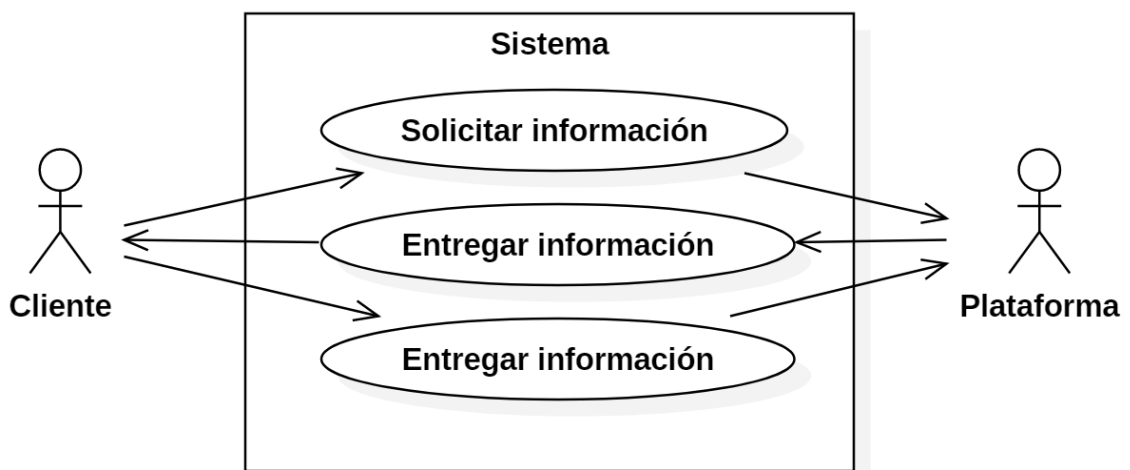
	<ul style="list-style-type: none"> • Imprimir comprobante de desembolso
Encargado Operativo	Sus funciones son: <ul style="list-style-type: none"> • Verificar datos de solicitudes
Sub gerente	Sus funciones son: <ul style="list-style-type: none"> • Aprobar montos
Gerente general	Sus funciones son: <ul style="list-style-type: none"> • Añadir productos. • Añadir moneda, amortización y interés.
Administrador	Sus funciones son: <ul style="list-style-type: none"> • Añadir usuario

Fuente y elaboración: Propia

Usando UML, es posible definir gráficamente los procesos que realizará el nuevo sistema, definiendo el entorno y las interacciones que suceden en éste. La generación de diagramas de casos de uso, secuencias, actividades, clases y base de datos son esenciales para el análisis y de esta manera formar la base principal para la etapa de diseño.

3.3.2 Diagrama de casos de uso

Figura 3.19 Diagrama de caso de uso solicitud de información y entrega de requisitos



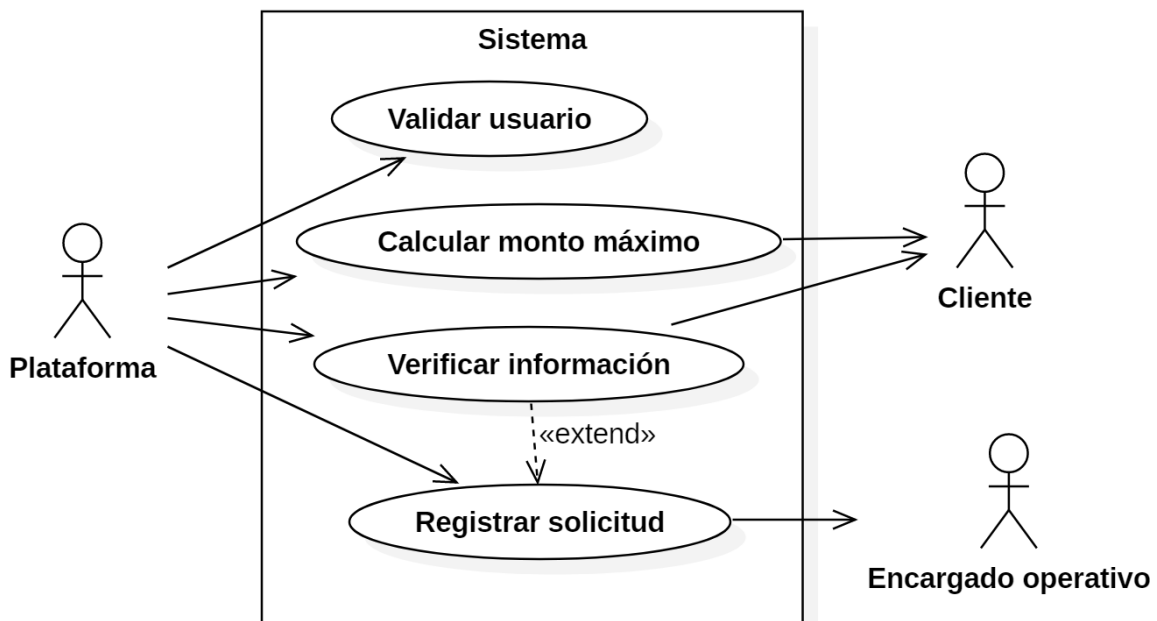
Fuente y elaboración: Propia

Tabla 3.3 Especificación de los casos de uso Cliente

Nombre	Solicitud de información y entrega de requisitos	
Actor	Cliente y plataforma	
Descripción	Describe el proceso para entregar los requisitos de un préstamo.	
Flujo principal	Eventos Actor	Eventos Sistema
	Plataforma ingresa al sistema	Muestra la información solicitada.
	Se consigue los requisitos.	
	Se presentación los requisitos.	Se verifica que tenga todos los requisitos.
	Se le informara el día para el desembolso	Registra información
Flujo alternativo	Revisar información	Revisan los requisitos si presenta observaciones y regularizaciones.
	Corrige las observaciones y regularizaciones.	Presenta los requisitos.

Fuente y elaboración: Propia

Figura 3.20 Diagrama de caso de uso registro de solicitud



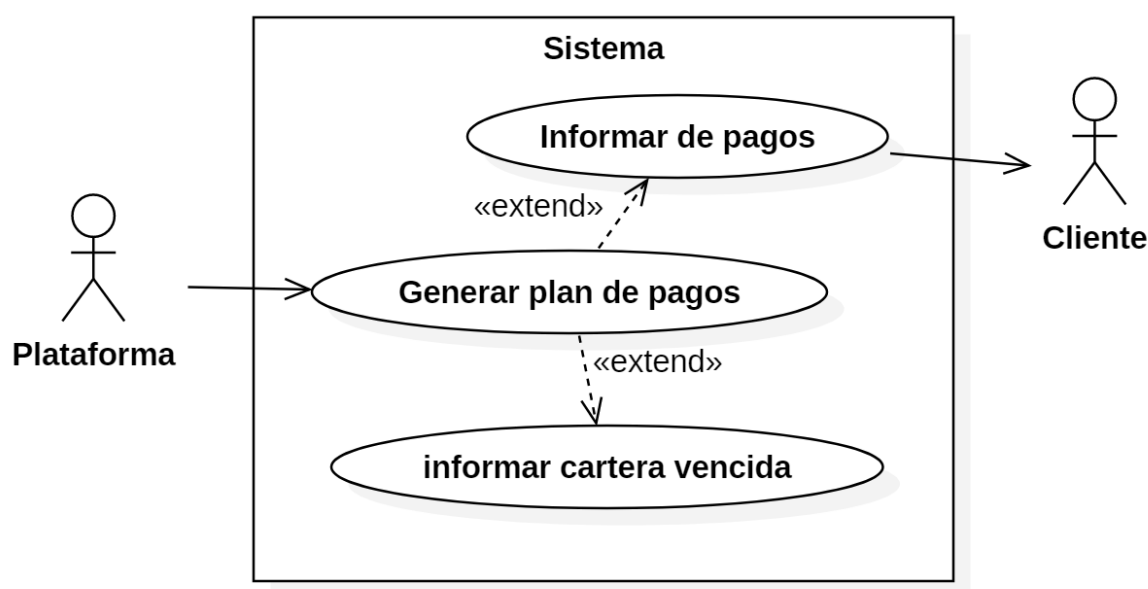
Fuente y elaboración: Propia

Tabla 3.4 Especificación de caso de uso registro solicitud

Nombre	Registro de solicitud	
Actor	Plataforma, cliente y encargado operativo.	
Descripción	Describe el proceso para el registro de préstamos solicitados por la CAJA POPULAR BOLIVIANA SRL.	
Flujo principal	Eventos Actor	Eventos Sistema
	Validar usuario	Calcular el Monto máximo
	Verificar información	Llenar el checklist
	Llenar formularios	Registrar solicitud
Flujo alternativo	Observaciones de documentación	Registra la documentación

Fuente y elaboración: Propia

Figura 3.21 Diagrama de casos de reporte pagos y cartera vencida



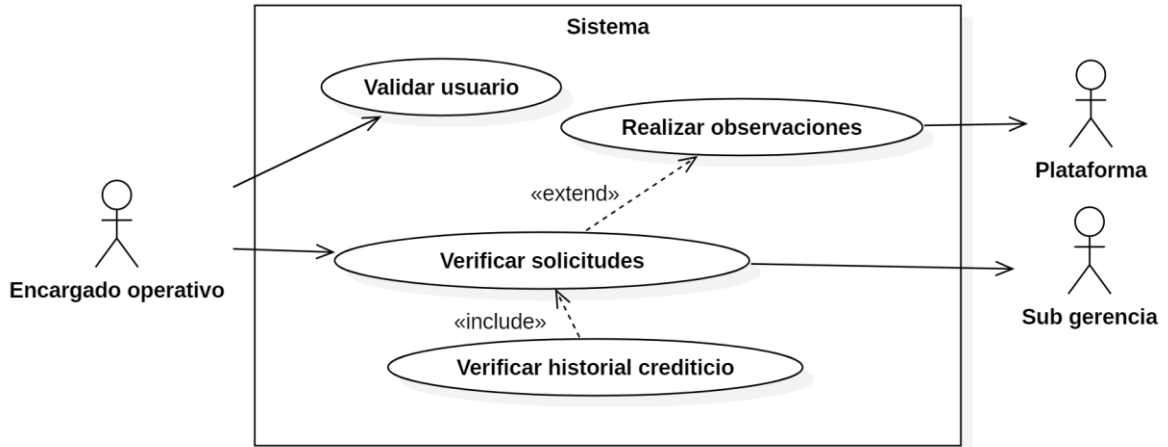
Fuente y elaboración: Propia

Tabla 3.5 Especificación de caso de uso reporte de pagos y cartera vencida

Nombre	Reporte pagos y cartera vencida	
Actor	Plataforma y cliente.	
Descripción	Describe el proceso para el reporte de pagos y reporte de cartera vencida en la CAJA POPULAR BOLIVIANA SRL.	
Flujo principal	Eventos Actor	Eventos Sistema
	Petición de reporte de estado	Generar reporte de estado de cliente
	Verificar estado de préstamos	Reporte de cartera vencida
Flujo alternativo	Documento de identidad del	

Fuente y elaboración: Propia

Figura 3.22 Diagrama de caso de uso verificación de documentación



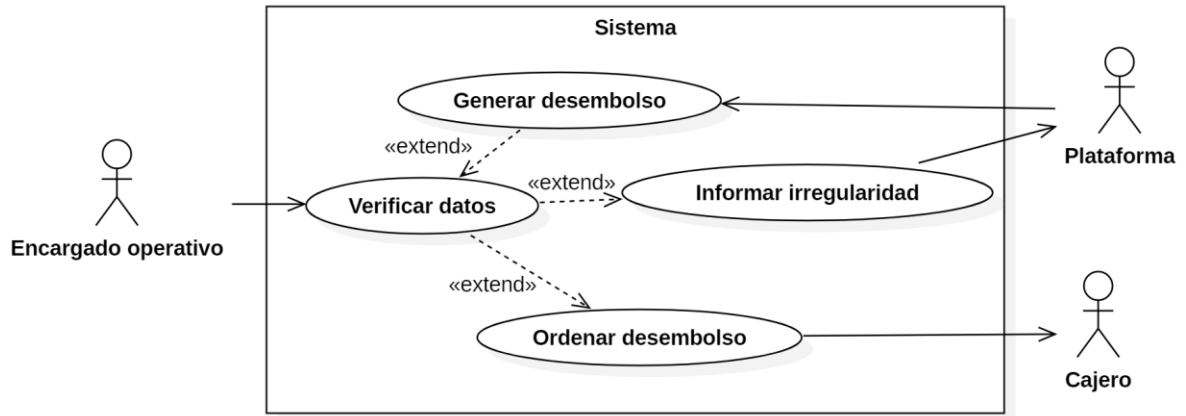
Fuente y elaboración: Propia

Tabla 3.6 Especificación de caso de uso verificación de documentación

Nombre	Gestión de solicitudes	
Actor	Encargado operativo, plataforma y subgerente.	
Descripción	Describe el proceso de la gestión de solicitudes	
Flujo principal	Eventos Actor	Eventos Sistema
	Iniciar sesión	Validar datos
	Verifica el historial institucional	Ve si el cliente tiene un historial crediticio en la institución
	Verifica los datos	Aprueba la verificación
Flujo alternativo	No cumple los requisitos	Se deniega el préstamo

Fuente y elaboración: Propia

Figura 3.23 Diagrama de caso de uso de programación de desembolso



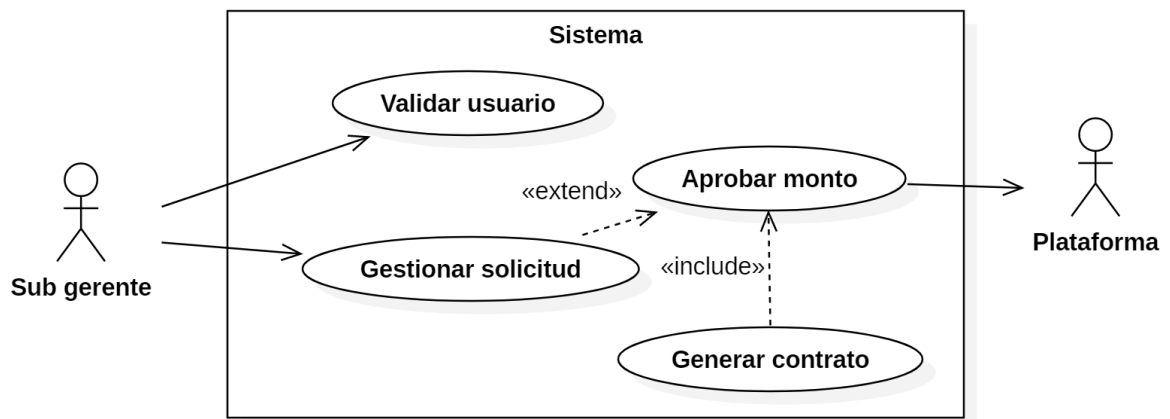
Fuente y elaboración: Propia

Tabla 3.7 Diagrama de caso de uso Subgerente

Nombre	Programar desembolso	
Actor	Encargado operativo, plataforma y cajero.	
Descripción	Describe el proceso para realizar un desembolso.	
Flujo principal	Eventos Actor	Eventos Sistema
	Ingrese desembolso	Generar Comprobante de desembolso
		Verificar datos
Flujo alternativo	Error al validar los datos	Informar irregularidades

Fuente y elaboración: Propia

Figura 3.24 Diagrama de casos de uso aprobación de prestamos



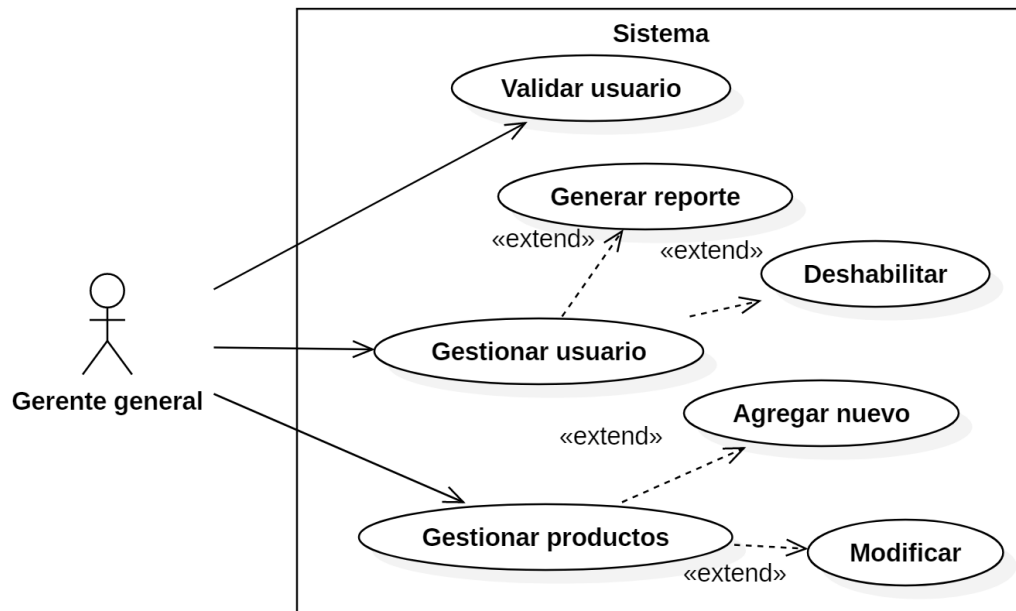
Fuente y elaboración: Propia

Tabla 3.8 Especificación de caso de uso aprobación de un préstamo

Nombre	Aprobación de un préstamo.	
Actor	Subgerente y plataforma.	
Descripción	Describe el proceso de aprobación por la CAJA POPULAR BOLIVIANA SRL.	
Flujo principal	Eventos Actor	Eventos Sistema
	Carpetas sin observaciones.	Aprobar monto y generar contrato. Registra el detalle de la operación.
Flujo alternativo	Recibe mensaje de error al realizar una operación.	Emite un mensaje de error y registra en los logs.

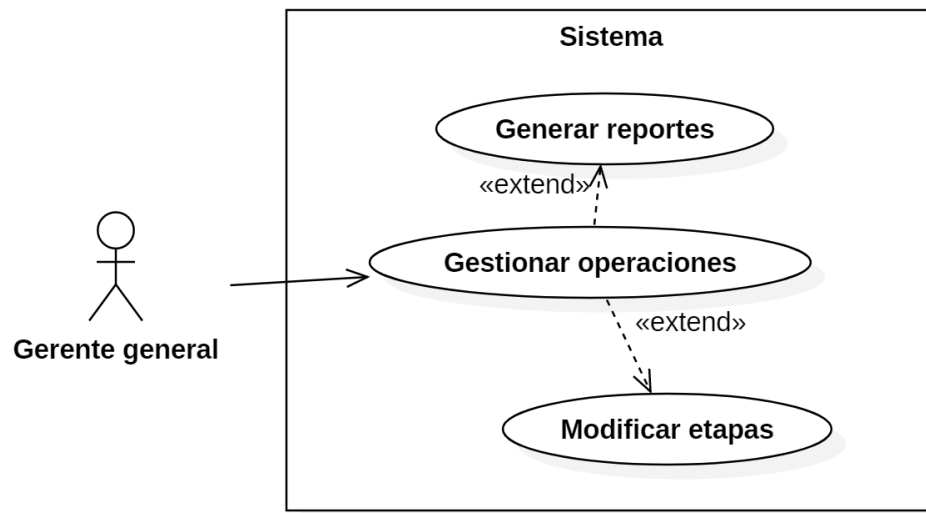
Fuente y elaboración: Propia

Figura 3.25 Diagrama de caso de uso Gerente general



Fuente y elaboración: Propia

Figura 3.26 Diagrama de casos de uso gestionar préstamos



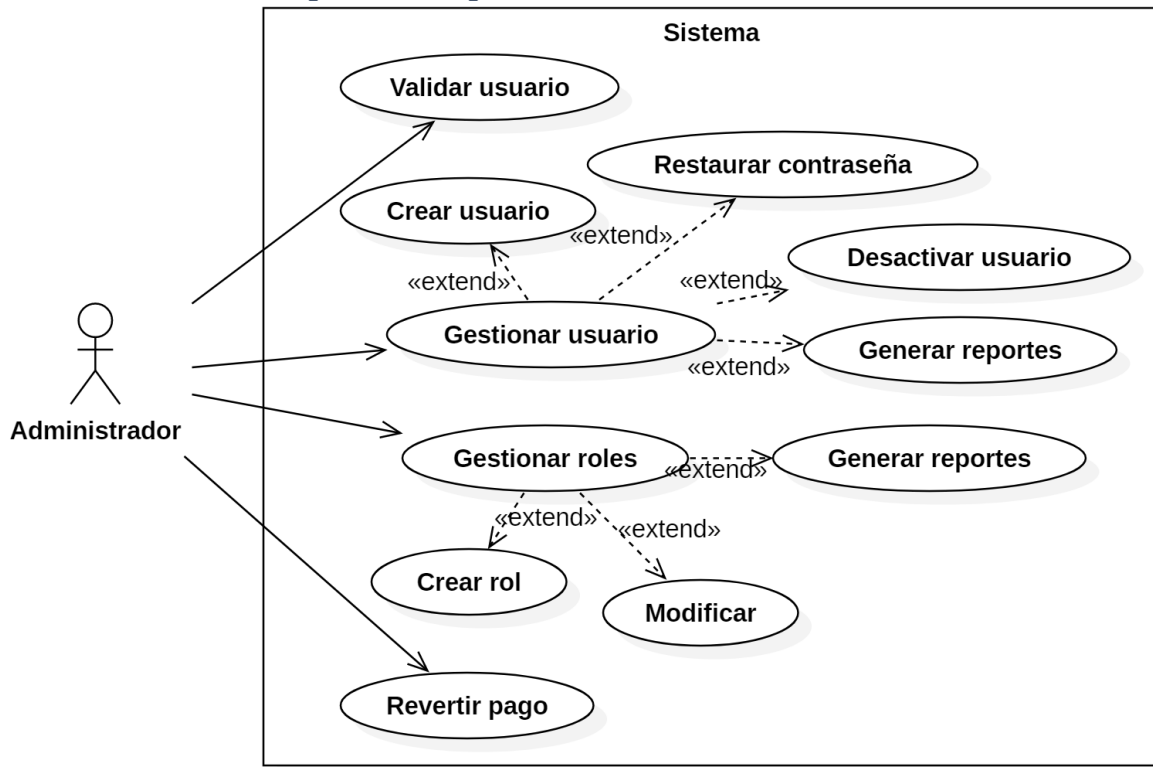
Fuente y elaboración: Propia

Tabla 3.9 Especificación de caso de uso Gestionar préstamo

Nombre	Gestionar préstamo.	
Actor	Gerente general, comisión evaluadora, administrador y departamento legal.	
Descripción	Describe del proceso para gestionar préstamos.	
Flujo principal	Eventos Actor	Eventos Sistema
	Se realiza alguna de las operaciones de gestión (Generar reportes y modificar etapas).	Muestra la vista de gestión de préstamos. Realizar los detalles de la operación.
Flujo alternativo	Recibe mensaje de error al realizar una operación.	Emite un mensaje de error y registra en los logs.

Fuente y elaboración: Propia

Figura 3.27 Diagrama de caso de uso Administrador



Fuente y elaboración: Propia

Tabla 3.10 Especificación de caso de uso gestionar usuarios

Nombre	Gestionar los usuarios.	
Actor	Administrador, gerente y subgerente.	
Descripción	Describe el proceso para la gestión de cuentas de usuarios	
Flujo principal	Eventos Actor	Eventos Sistema
	Se realiza alguna de las operaciones de gestión (Crear, modificación, configurar nivel de acceso consultar y eliminar)	Muestra la vista de gestión de cuentas de usuario.
		Registra el detalle de la operación.
Flujo alternativo	Recibe mensaje de error al realizar una operación.	Emite un mensaje de error y registra en los logs.

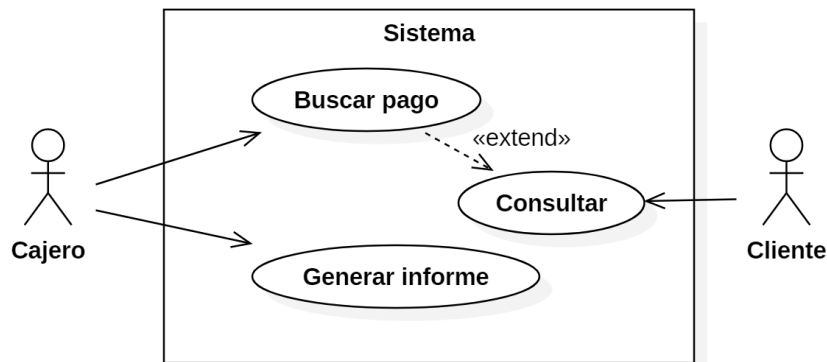
Fuente y elaboración: Propia

Tabla 3.11 Especificación de los casos de uso Generar reportes

Nombre	Generar reportes	
Actor	Administrador, gerente	
Descripción	Describe el proceso para la generación de reportes	
Flujo principal	Eventos Actor	Eventos Sistema
	Ingresa a la sección de reportes.	Muestra la vista de generación de reportes.
	Busca el tipo de reporte y proporciona la información requerida.	Muestra el formulario para la emisión del reporte elegido.
		Genera un reporte bajo los parámetros asignados y su posible exportación.
Flujo alternativo	Recibe mensaje de error al realizar una operación.	Emite un mensaje de error y registra en los logs.

Fuente y elaboración: Propia

Figura 3.28 Diagrama de caso de uso Cajero



Fuente y elaboración: Propia

Tabla 3.12 Especificación de casos de uso Registro de pagos

Nombre	Registro de pagos	
Actor	Cajero, cliente y encargado operativo.	
Descripción	Describe el proceso para el cobro de pagos por la CAJA POPULAR BOLIVIANA SRL.	
Flujo principal	Eventos Actor	Eventos Sistema
	Se realiza la búsqueda de pago.	Muestra la vista de pago.
		Registra el detalle de la operación.
Flujo alternativo	Recibe mensaje de error al realizar una operación.	Emite un mensaje de error y registra en los logs.

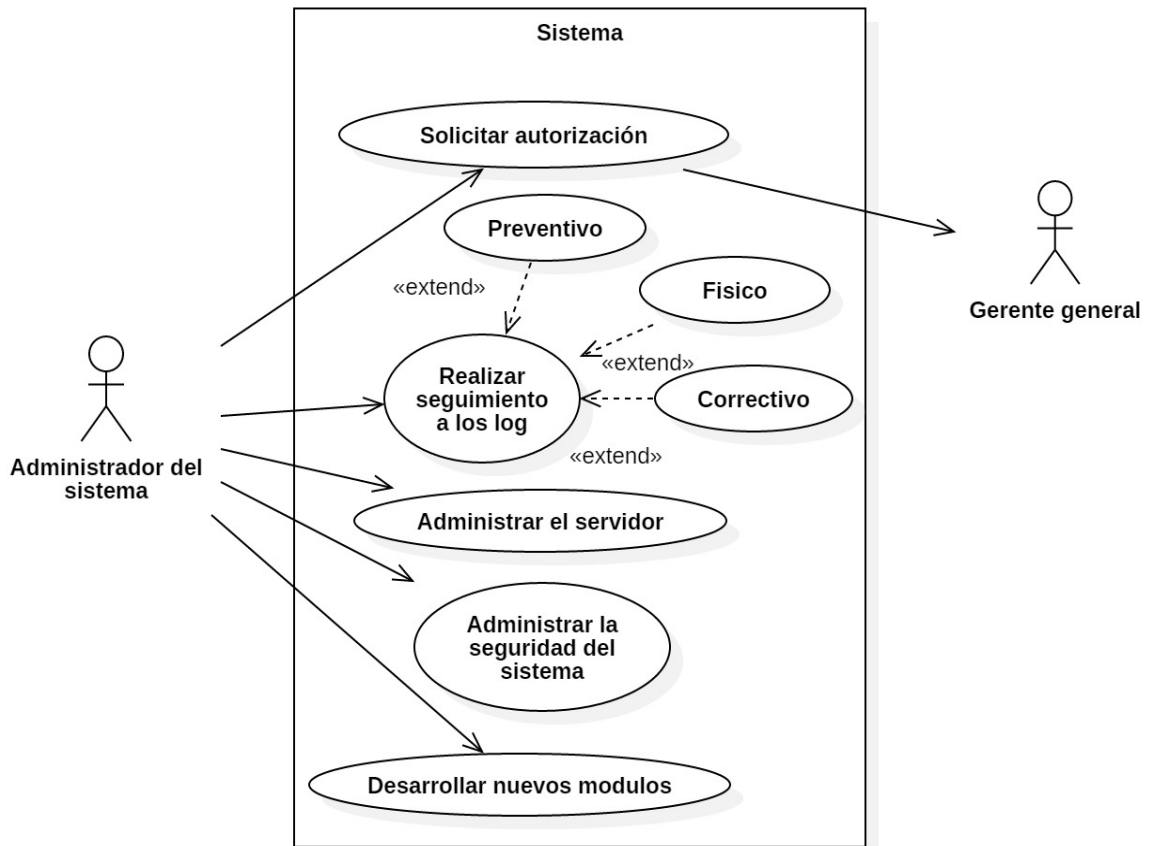
Fuente y elaboración: Propia

Tabla 3.13 Especificación de casos de uso Generar reportes de pagos

Nombre	Generar reportes de pagos	
Actor	Cajero y encargado operativo.	
Descripción	Describe el proceso para la generación de reportes	
Flujo principal	Eventos Actor	Eventos Sistema
	Ingresa a la sección de reportes.	Muestra la vista de generación de reportes.
	Busca el tipo de reporte y proporciona la información requerida.	Muestra el formulario para la emisión del reporte elegido.
		Genera un reporte bajo los parámetros asignados y su posible exportación.
Flujo alternativo	Recibe mensaje de error al realizar una operación.	Emite un mensaje de error y registra en los logs.

Fuente y elaboración: Propia

Figura 3.29 Diagrama de caso de uso Administrador del sistema

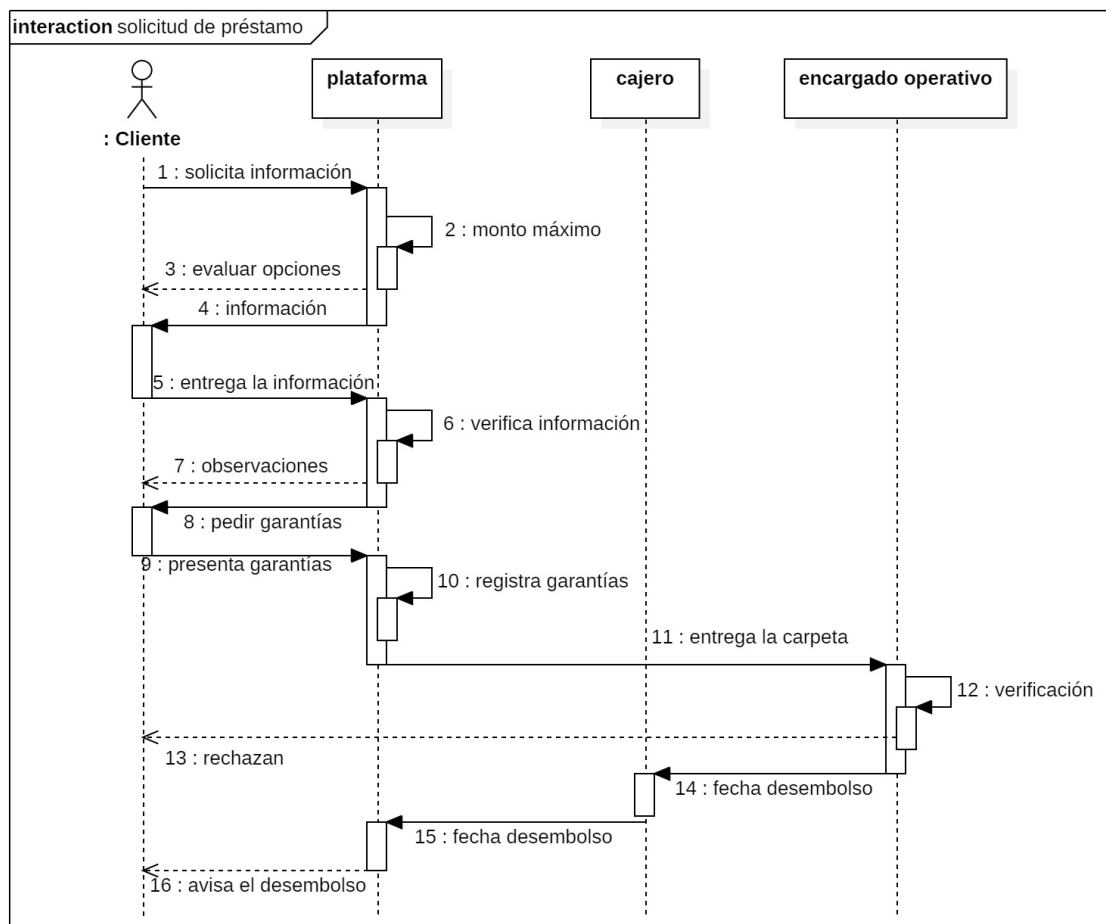


Fuente y elaboración: Propia

El administrador del sistema posee las facultades para realizar el correcto seguimiento de los registros (*logs*), realizar diferentes tipos de mantenimiento, gestionar el servidor y la seguridad e implementar nuevos módulos o características al sistema. Para cada tarea asignada el administrador del sistema, es necesaria la autorización previa del gerente.

3.3.3 Diagrama de secuencias

Figura 3.30 Diagrama de secuencia Solicitud de préstamo

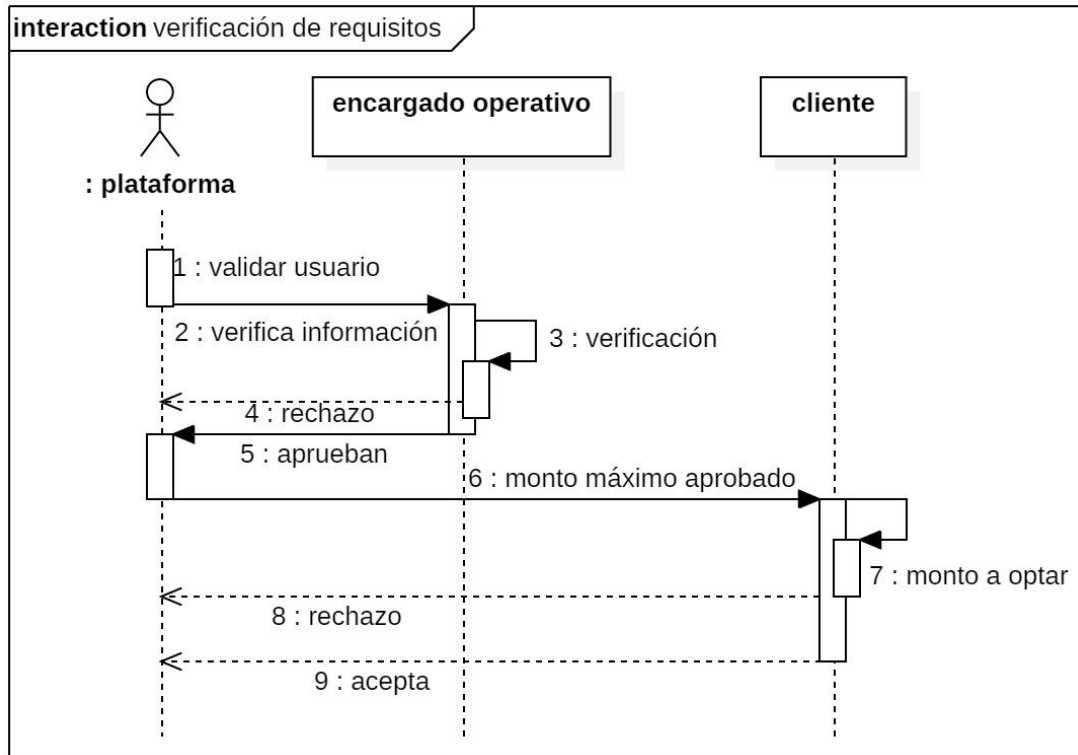


Fuente y elaboración: Propia

El cliente solicita la información de un préstamo, luego plataforma consulta datos otorgados por el cliente la información de los montos máximos y mínimo, si el cliente acepta el monto disponible recolecta los requisitos entrega la información, Si hay alguna observación de corrige se llama al cliente para adjuntar la información

faltante el registro de garantías cerciorados por plataforma, Se realiza el registro de garantías, el encargado operativo verifica una vez más la información da una fecha de desembolso para el préstamo notifica a cajero y a plataforma y se realiza el desembolso.

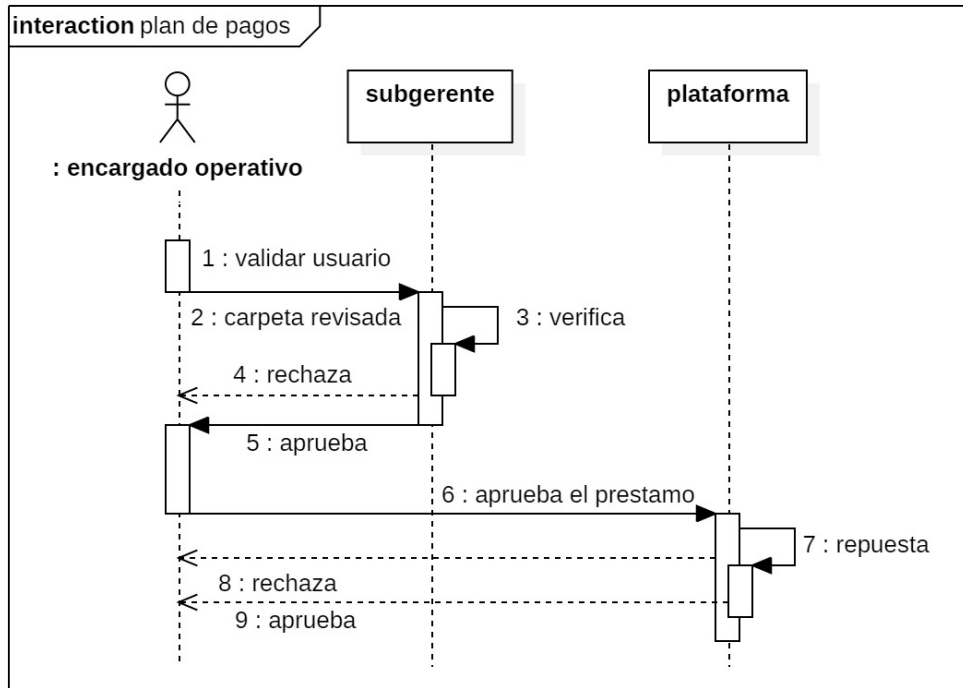
Figura 3.31 Diagrama de secuencia Verificación de requisitos



Fuente y elaboración: Propia

Se realiza el inicio de sesion verifica la informacion envia la informacion a encargado operativo revisa la informacion da el visto bueno da la aprobacion para el monto otorgado.

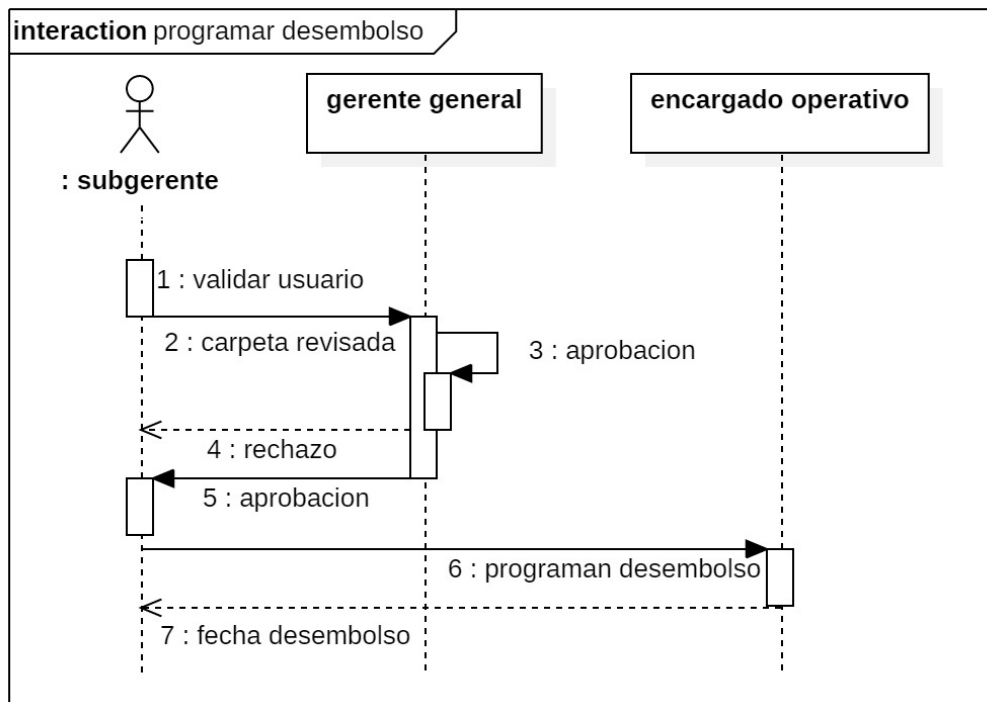
Figura 3.32 Diagrama de secuencia Plan de pagos



Fuente y elaboración: Propia

El encargado operativo inicia sesión verifica la carpeta crea el plan de pagos.

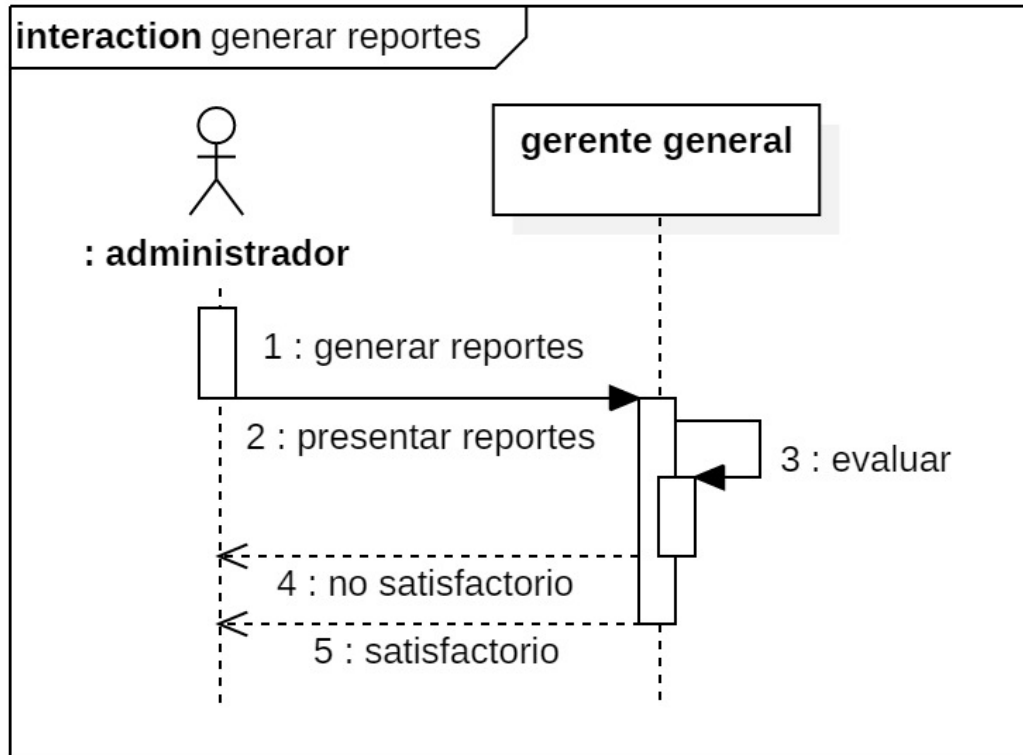
Figura 3.33 Diagrama de secuencia Programar desembolso



Fuente y elaboración: Propia

El subgerente coordina con el encargado operativo para realizar el desembolso una vez que lo aprueba el gerente.

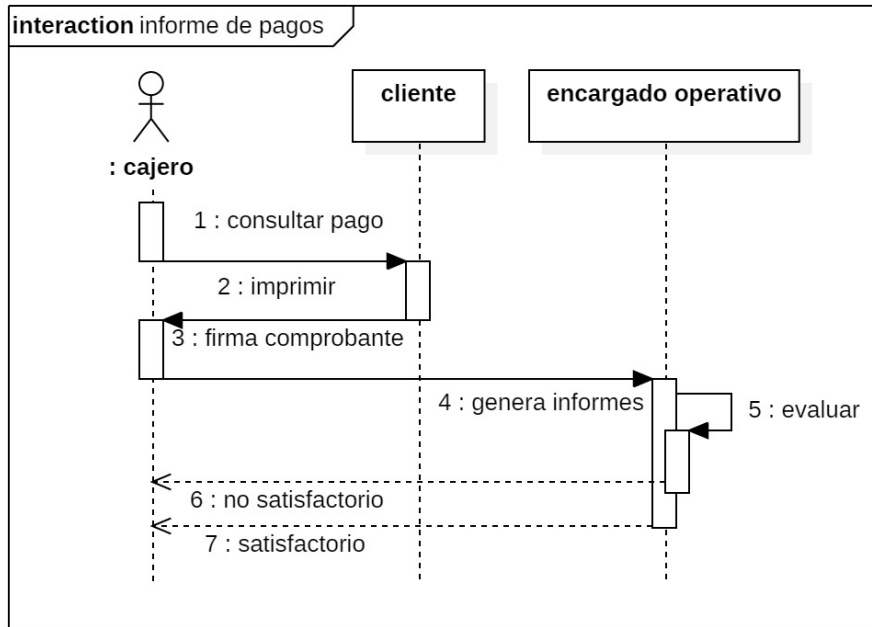
Figura 3.34 Diagrama de secuencia Generar reportes



Fuente y elaboración: Propia

Los reportes requeridos por el sistema son clientes, plataforma, encargado operativo, subgerente, gerente general, registro de actividades y estado de préstamos. Sólo los roles de gerente y administrador tienen acceso al módulo para su futura evaluación.

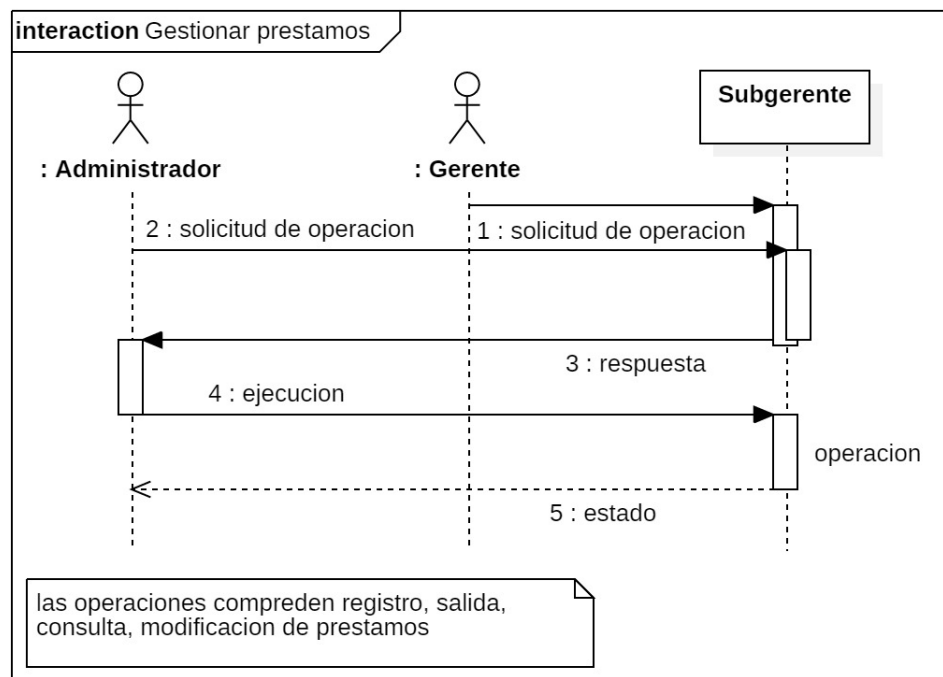
Figura 3.35 Diagrama de secuencia Informe de pagos



Fuente y elaboración: Propia

El cajero realiza la búsqueda del cliente para realizar los pagos imprime el comprobante y genera informe de las personas que pagaron y que no pagaron.

Figura 3.36 Diagrama de secuencia Gestionar prestamos

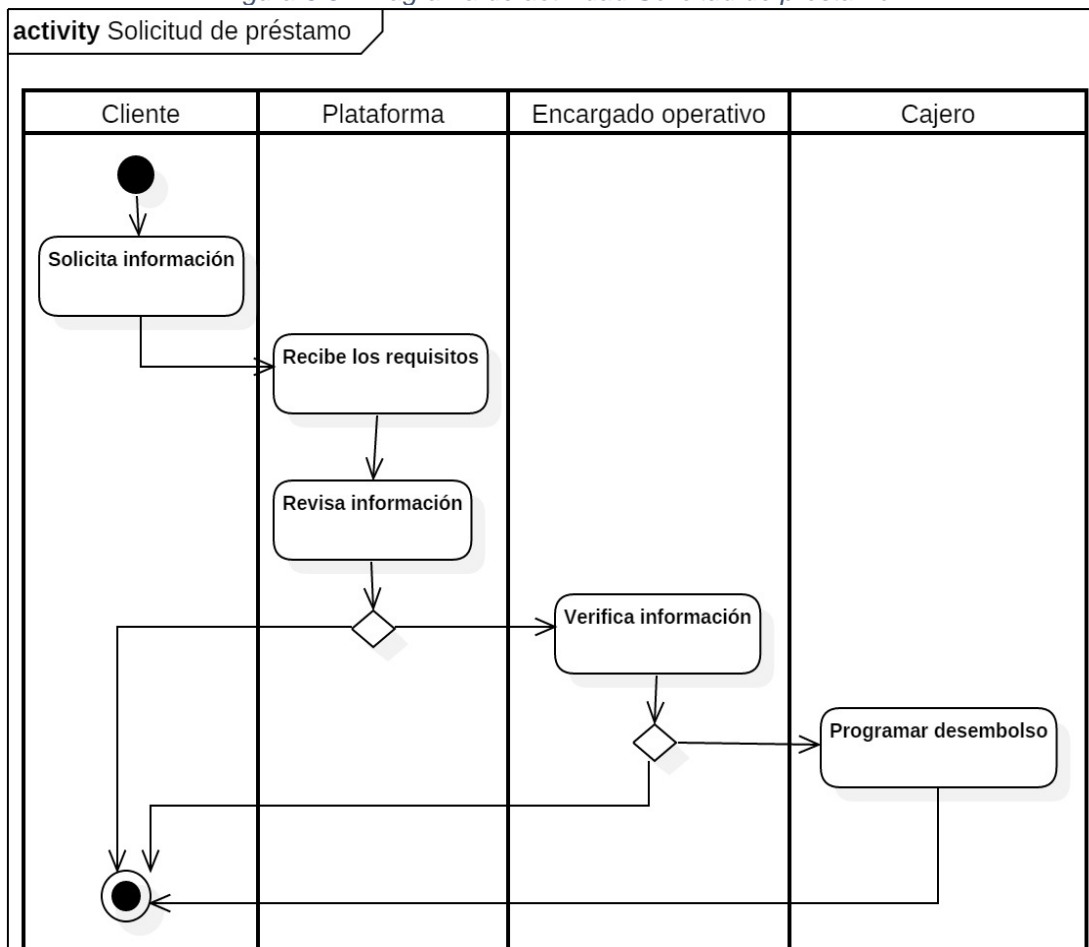


Fuente y elaboración: Propia

Las operaciones de registro, salida, consulta, modificación solo pueden ser realizadas por los roles de administrador y gerente, estos procesos son registrados para la evaluación en los reportes.

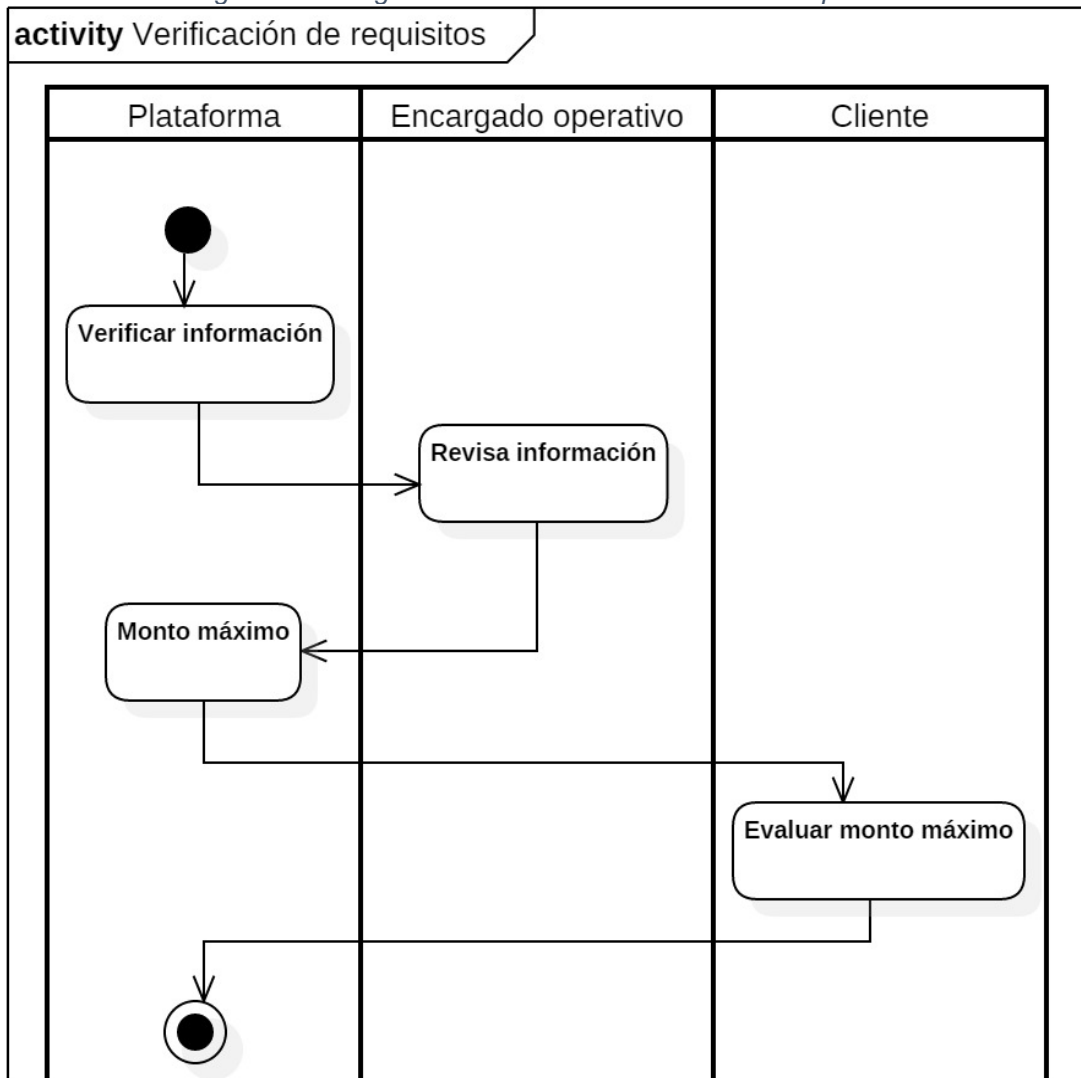
3.3.4 Diagramas de actividades

Figura 3.37 Diagrama de actividad Solicitud de préstamo



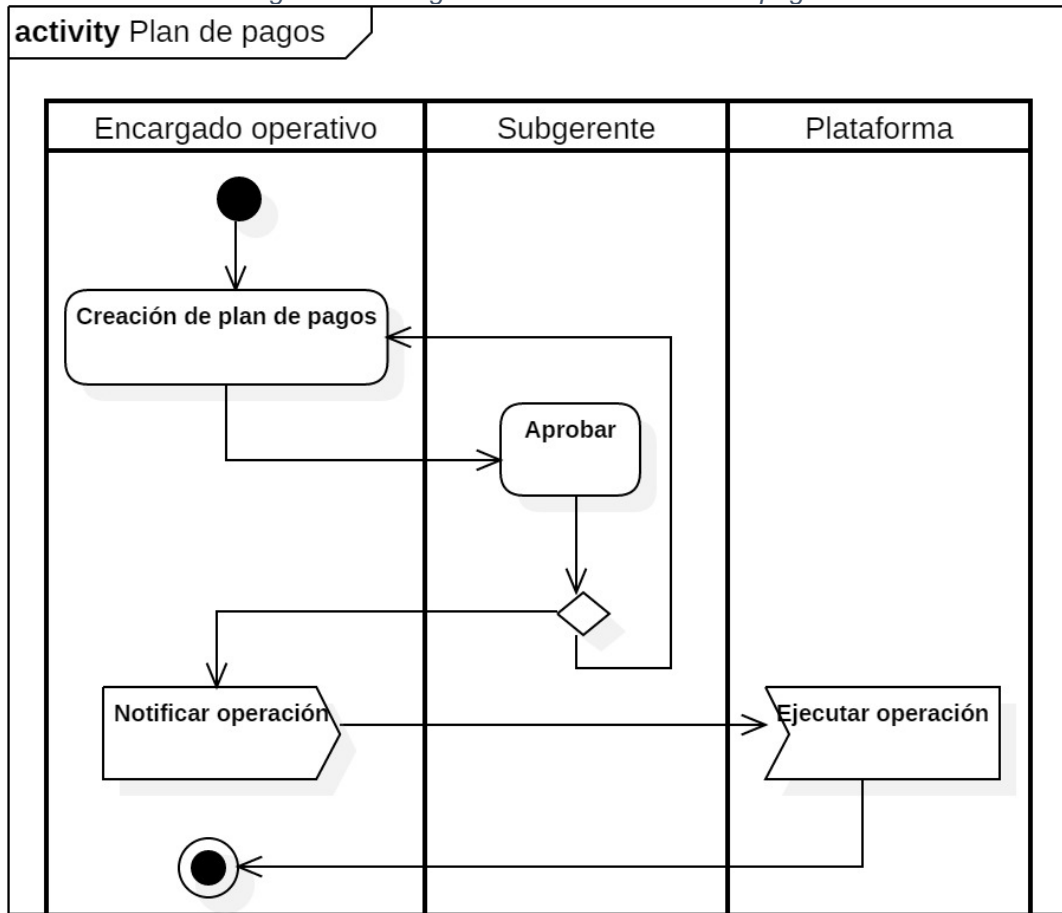
Fuente y elaboración: Propia

Figura 3.38 Diagrama de actividad Verificación de requisitos



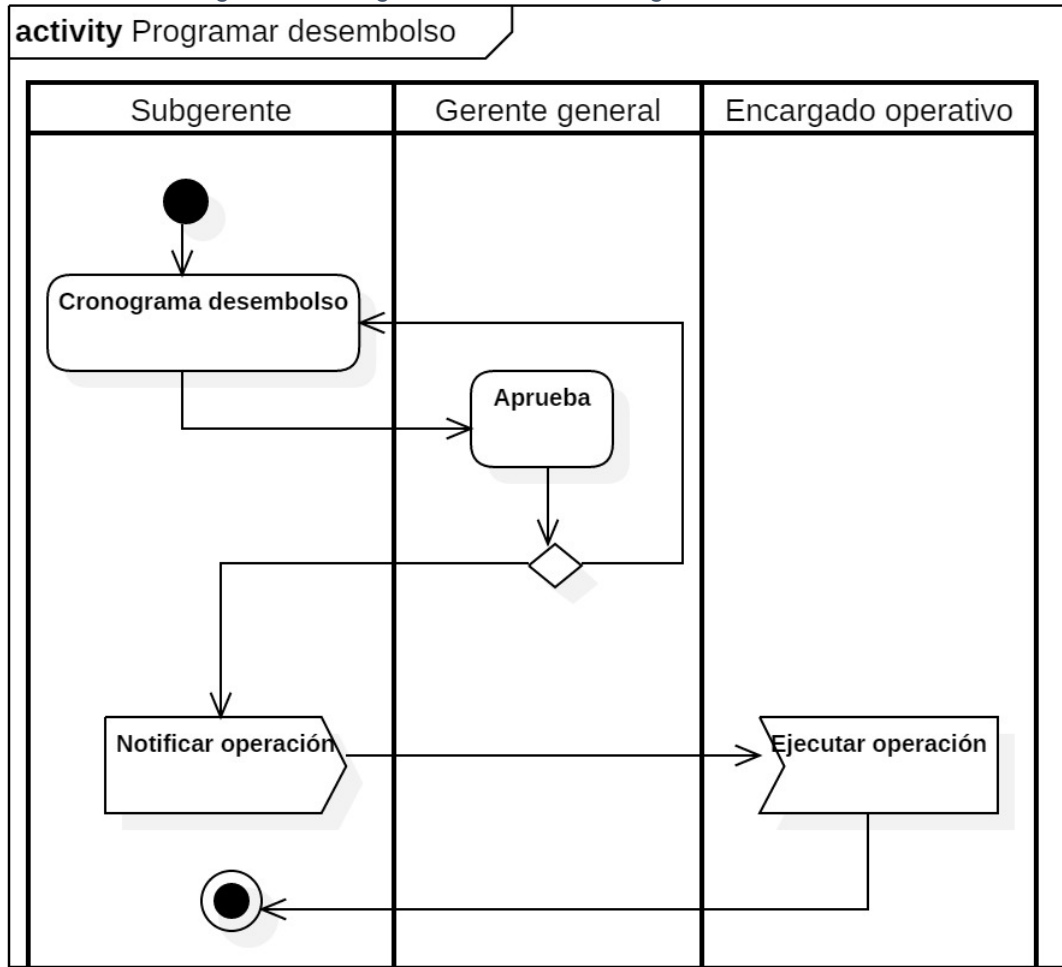
Fuente y elaboración: Propia

Figura 3.39 Diagrama de actividad Plan de pagos



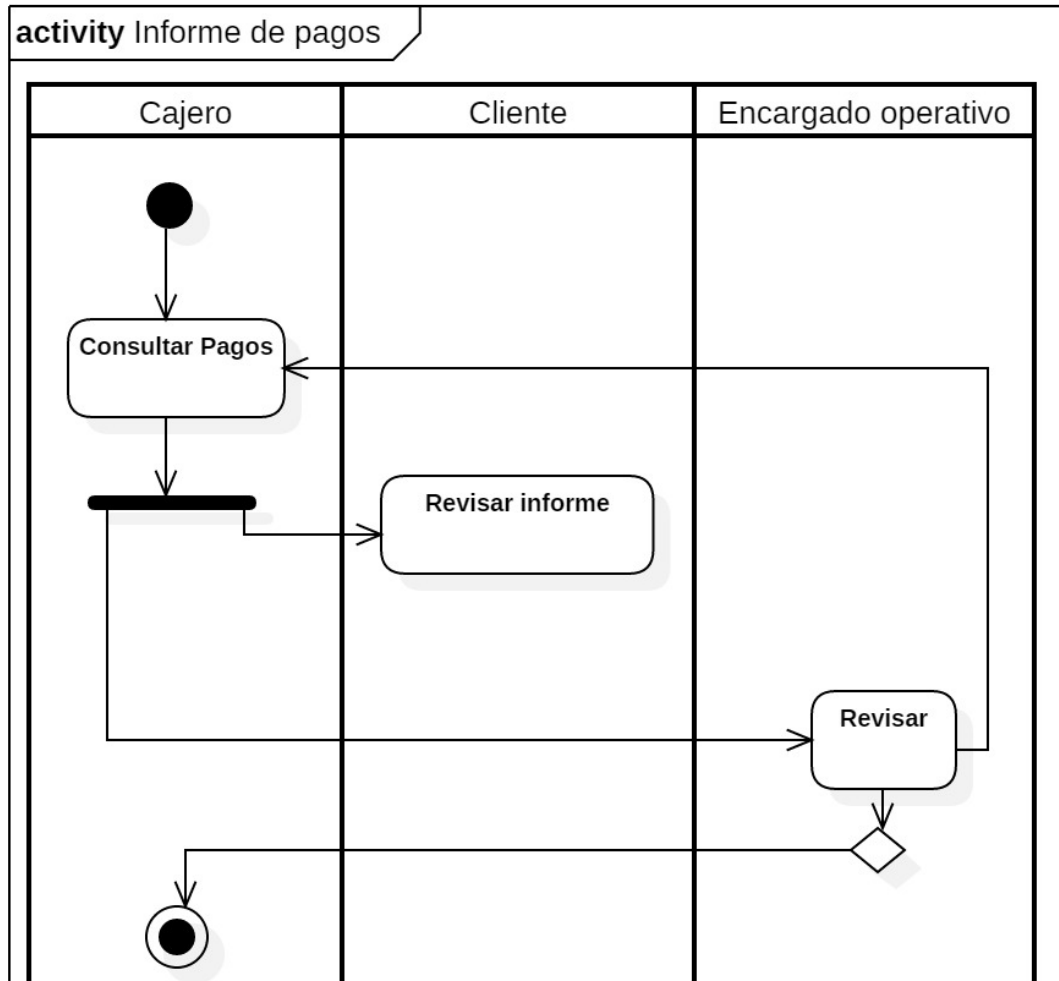
Fuente y elaboración: Propia

Figura 3.40 Diagrama de actividad Programar desembolso



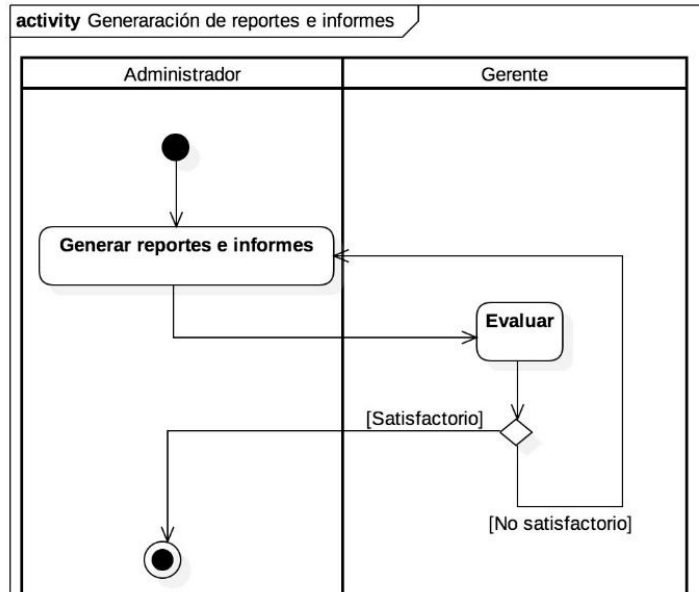
Fuente y elaboración: Propia

Figura 3.41 Diagrama de actividad Informe de pagos



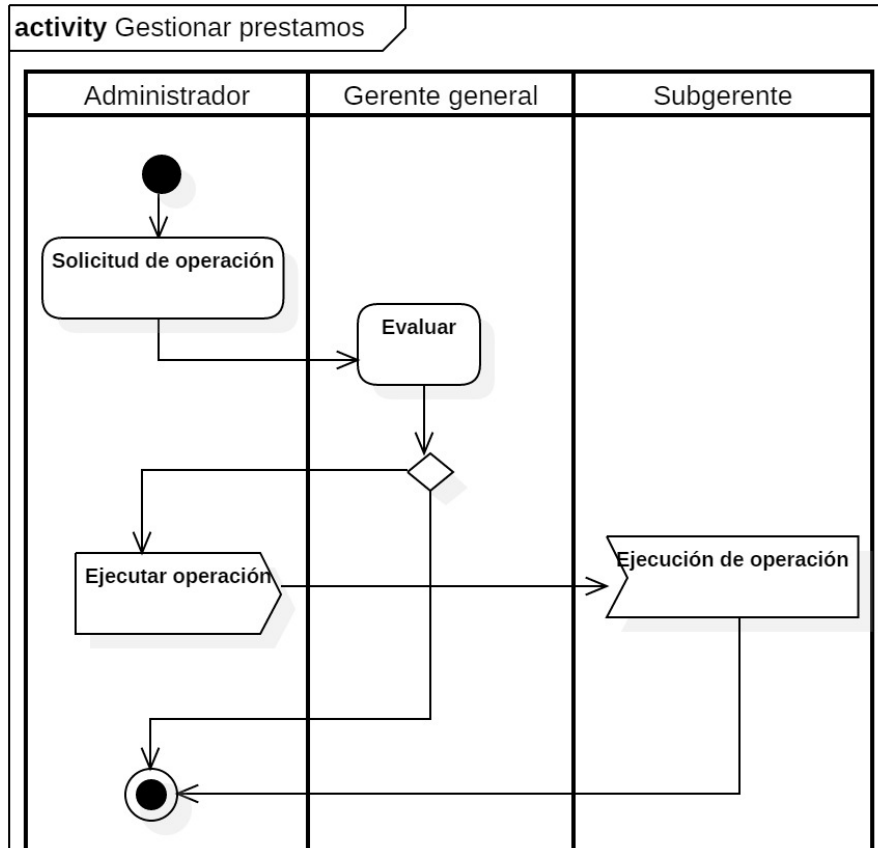
Fuente y elaboración: Propia

Figura 3.42 Diagrama de actividad Generación de reportes e informes



Fuente y elaboración: Propia

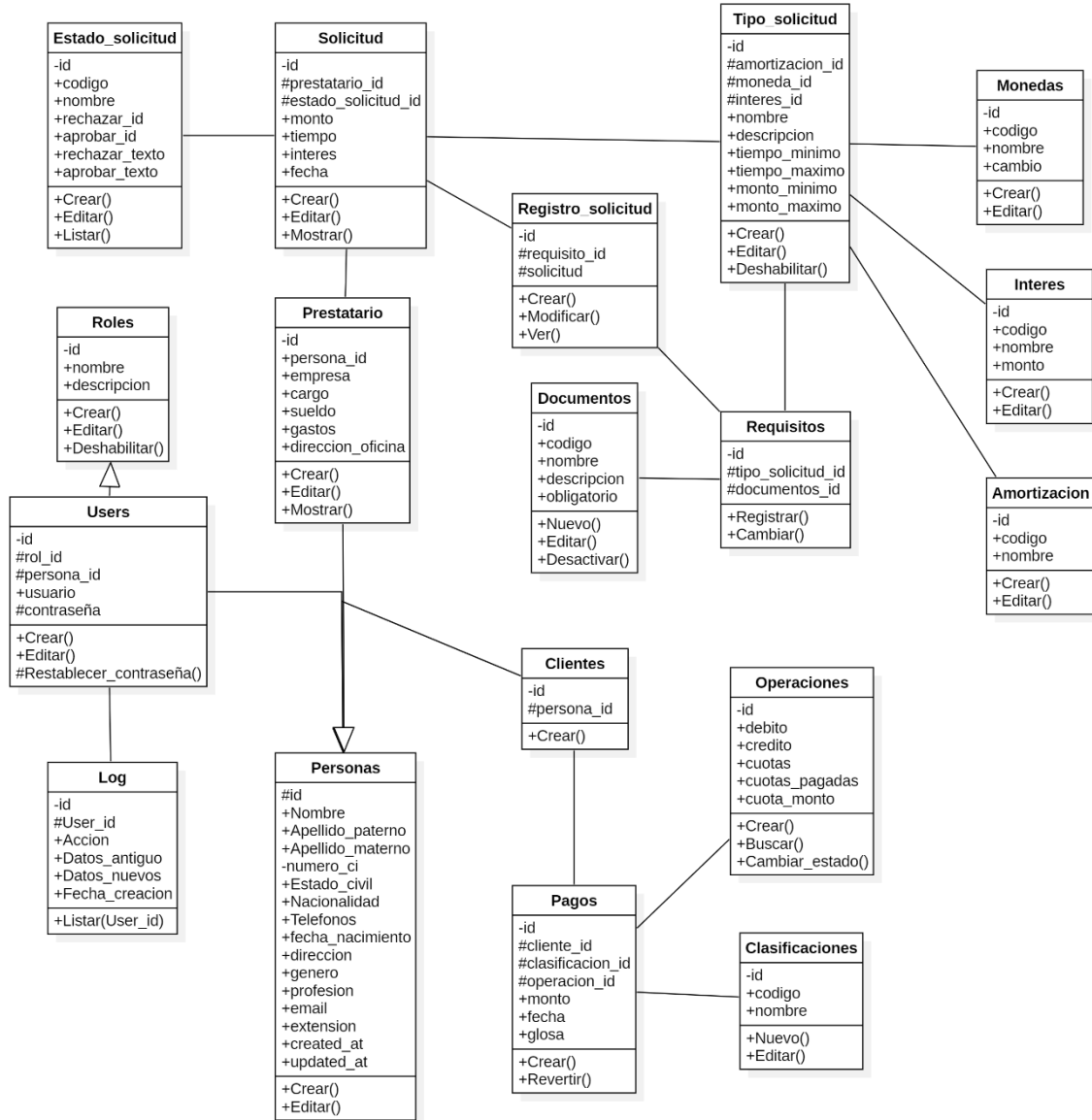
Figura 3.43 Diagrama de actividad Gestionar prestamos



Fuente y elaboración: Propia

3.3.5 Diagramas de clases

Figura 3.44 Diagrama de clases del sistema



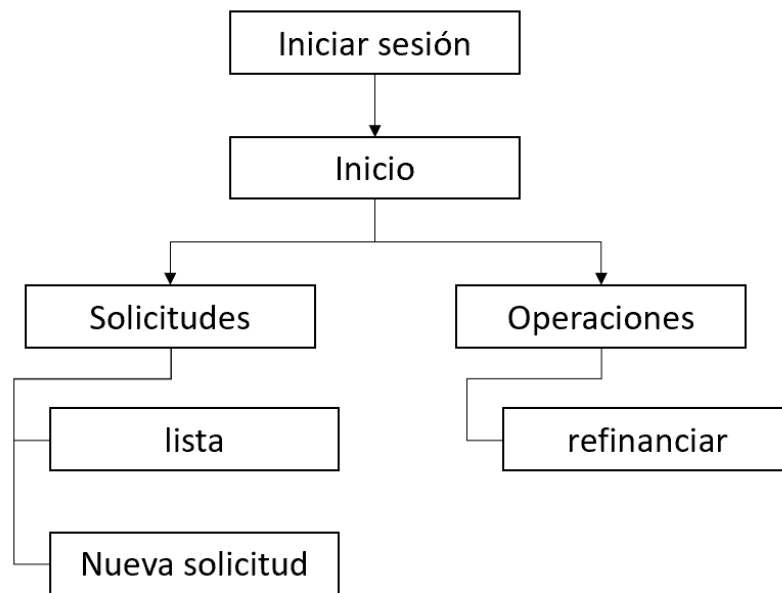
Fuente y elaboración: Propia

3.4 DISEÑO EL SISTEMA.

La etapa de diseño del sistema comprende la especificación de los modelos de navegación, interfaz y reportes.

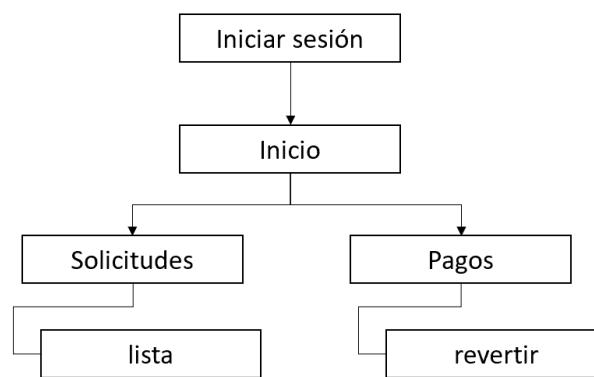
3.4.1 Diagrama de navegación

Figura 3.46 Diagrama navegacional de plataforma



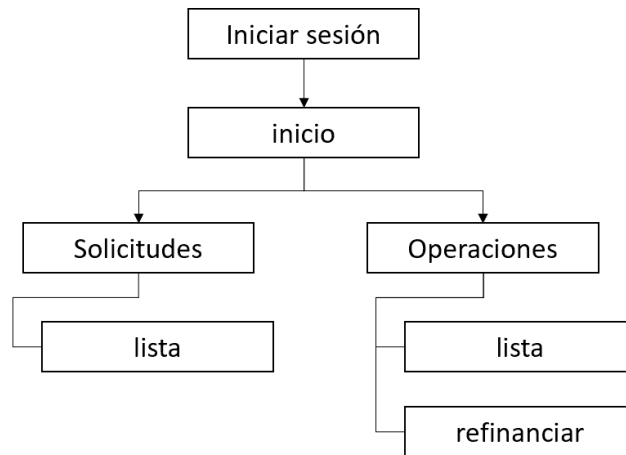
Fuente y elaboración: Propia

Figura 3.47 Diagrama navegacional de encargado operativo



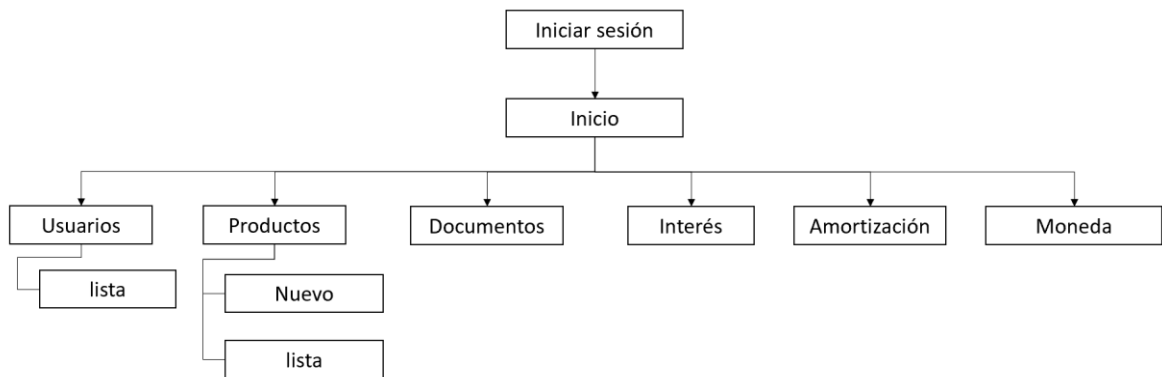
Fuente y elaboración: Propia

Figura 3.48 Diagrama navegacional de sub gerente



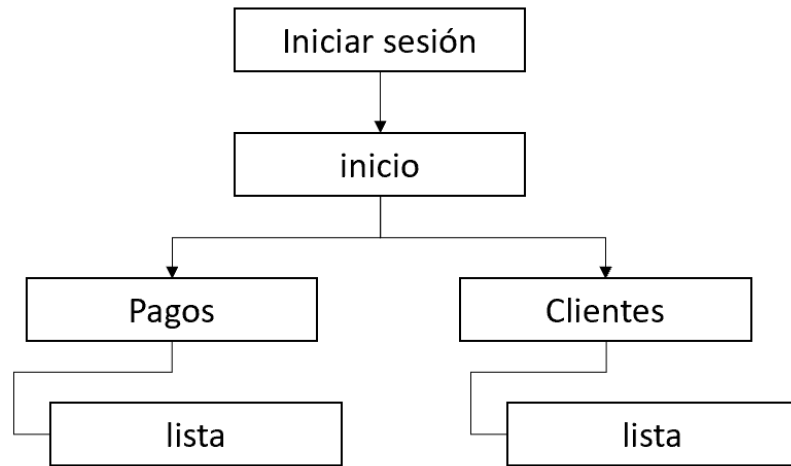
Fuente y elaboración: Propia

Figura 3.49 Diagrama de navegacional de gerente general



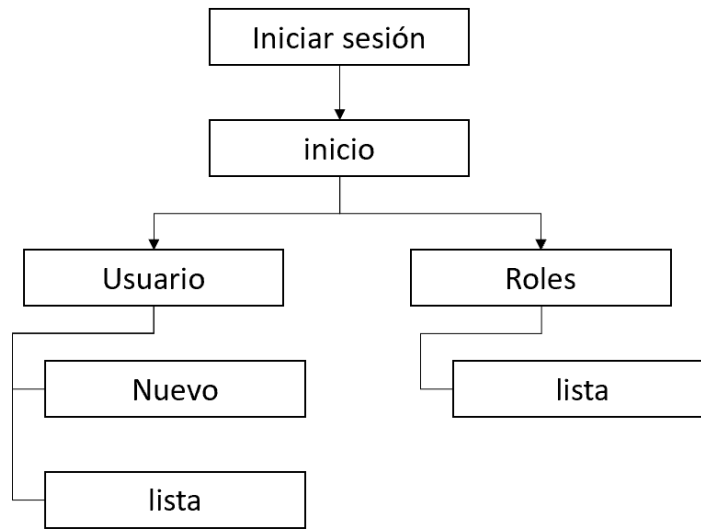
Fuente y elaboración: Propia

Figura 3.50 Diagrama navegacional de cajas



Fuente y elaboración: Propia

Figura 3.51 Diagrama navegacional del administrador



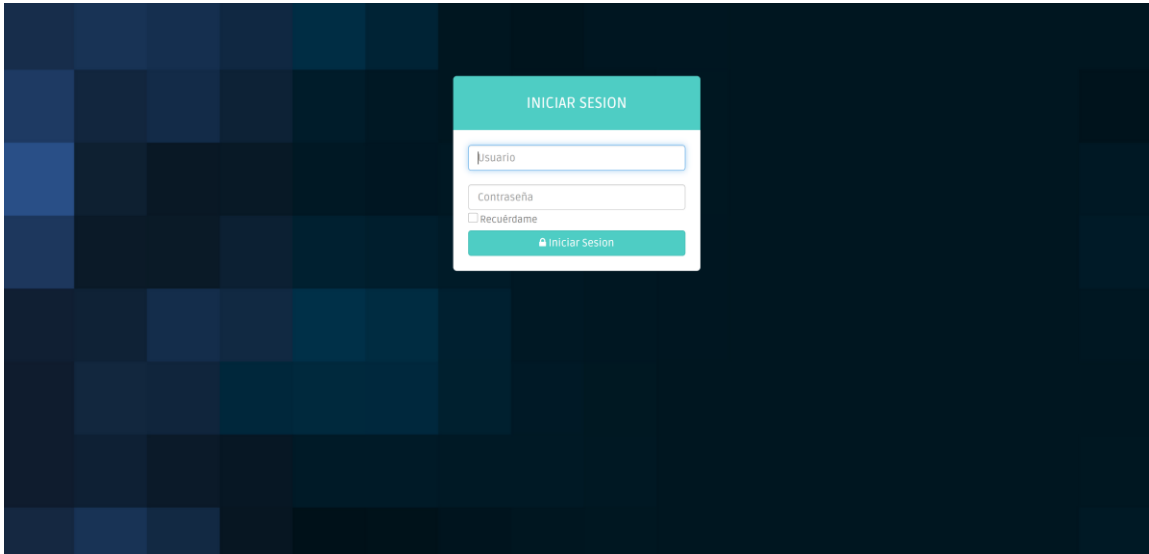
Fuente y elaboración: Propia

3.4.2 Diagrama de interfaz

Sistema web para la administración de préstamos Caja Popular Boliviana

Estas son las vistas que tiene el sistema

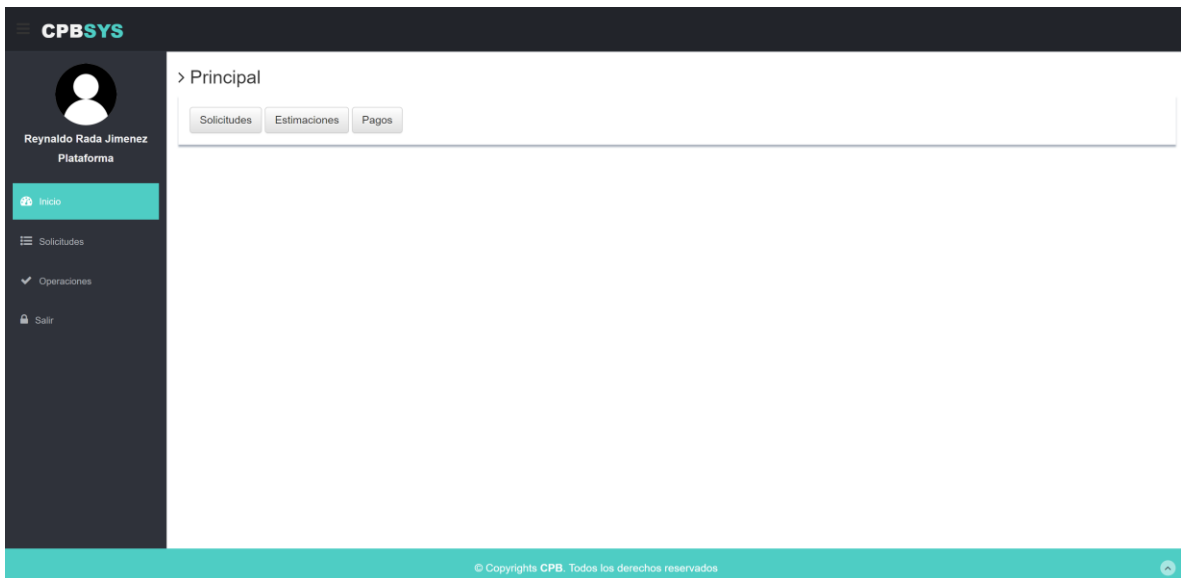
Figura 3.52 Interfaz de inicio de sesión



Fuente y elaboración: Propia

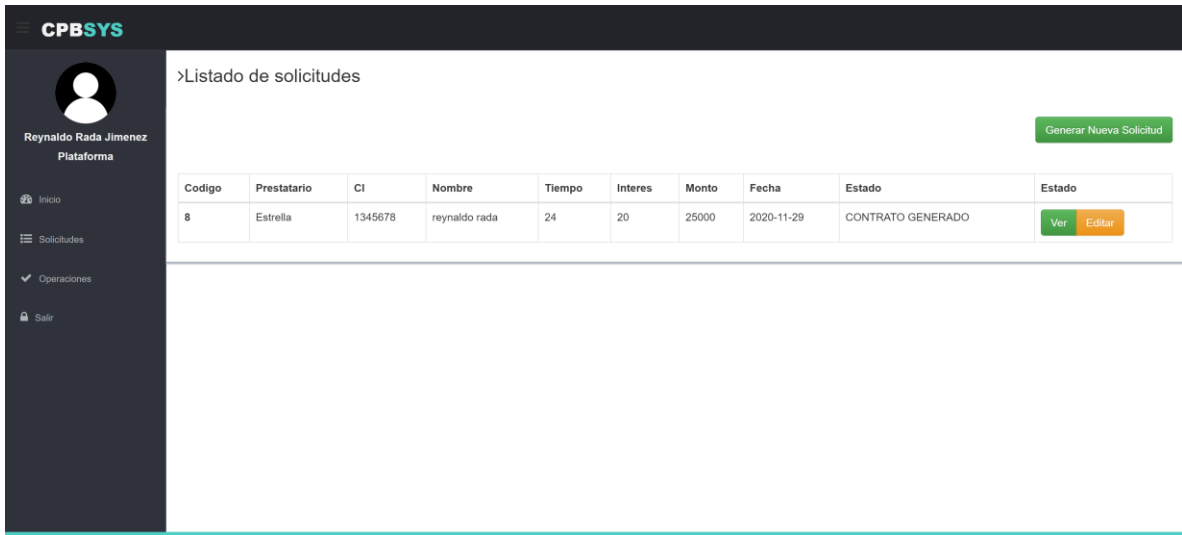
En el sistema hay 6 tipos de usuarios cada uno interactúa de distinta manera con el sistema, vistas plataforma.

Figura 3.53 Interfaz principal de plataforma



Fuente y elaboración: Propia

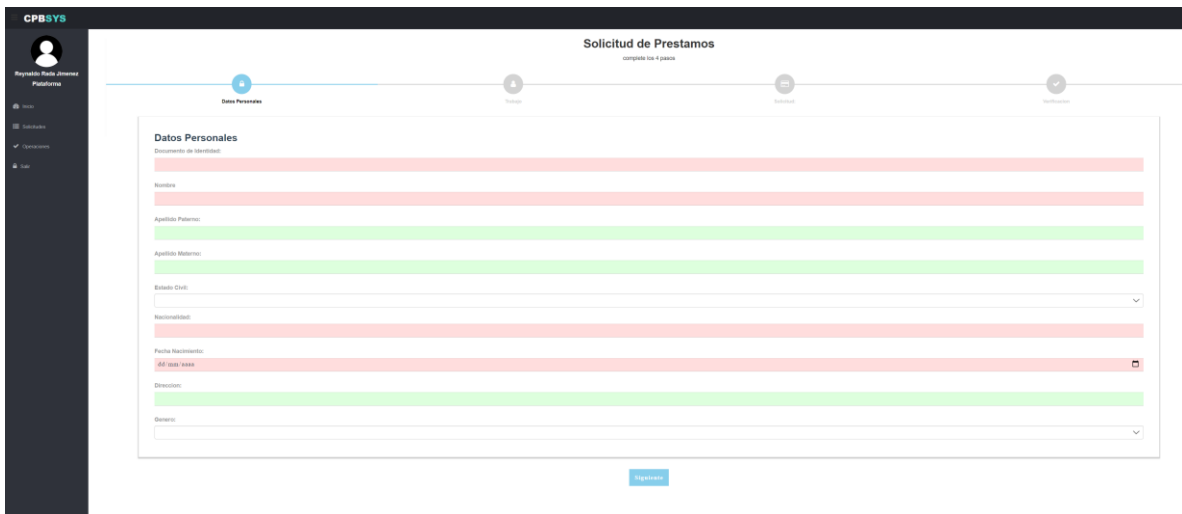
Figura 3.54 Interfaz de lista de solicitudes plataforma



Fuente y elaboración: Propia

Las nuevas solicitudes se registran en un formulario paso a paso es para capturar el primer paso realiza el registro de los datos personales.

Figura 3.55 Interfaz registro de datos persona



Fuente y elaboración: Propia

El segundo paso se realiza la captura de la información de su fuente de ingresos

Figura 3.56 Interfaz registro de datos de trabajo

CPBSYS

Reynaldo Rada Jimenez
Plataforma

Solicitud de Prestamos
completo los 3 pasos

Datos Personales Trabajo Solicitud Verificación

Trabajo

Empresa

Cargo

Sueldo

Gastos

Direccion

Anterior Siguiente

Fuente y elaboración: Propia

El tercer paso se utiliza para guardar el monto solicitado y el producto seleccionado

Figura 3.57 Interfaz registro de solicitud

CPBSYS

Reynaldo Rada Jimenez
Plataforma

Solicitud de Prestamos
completo los 4 pasos

Datos Personales Trabajo Solicitud Verificación

Solicitud:

Producto
prestamo de libre disponibilidad

Tiempo
Estrategia adicional 20 a 60 meses

Interes
2%

monto

Previous Siguiente

Fuente y elaboración: Propia

Y el paso final se utiliza como una verificación de los requisitos que tiene.

Figura 3.58 Interfaz verificación de requisitos

CPBSYS

Reynaldo Rada Jimenez
Plataforma

Solicitud de Prestamos
completo los 4 pasos

Datos Personales Trabajo Solicitud Verificación

Verificación:

carnet de identidad:

Fuente de ingresos:

Documentacion declaracion patrimonial:

Estabilidad laboral:

Carga Financiera Total/Renta Total:

Documentación de respaldo de deudas vigentes:

Previous Generar Sol

Fuente y elaboración: Propia

Figura 3.59 Interfaz lista de las operaciones que pueden ser refinanciadas

CPBSYS

Reynaldo Rada Jimenez
Plataforma

Inicio
Solicitudes
Operaciones
Salir

>Listado de operacion

Codigo	Prestatario	CI	Nombre	Couta restante	Estado	
8	Estrella	1345678	reynaldo rada	5	Cartera Activa	Refinanciar

Fuente y elaboración: Propia

Figura 3.60 Interfaz mostrar los datos de la solicitud

CPBSYS

Reynaldo Rada Jimenez
Plataforma

Inicio
Solicitudes
Operaciones
Salir

DATOS PERSONALES		DATOS TRABAJO		DATOS SOLICITUD	
Numero documento:	1345678	Empresa:	Estrella	Tipo Solicitud:	préstamo de libre disponibilidad
Nombre:	reynaldo	Cargo:	Tecnico	Tiempo:	24
Apellido Paterno:	rada	Sueldo:	3500	Interes:	20
Apellido Materno:	jimenez	Gastos:	500	Monto:	25000
Estado civil:	soltero	Direccion:	prueba	DOCUMENTOS	
Estado civil:	soltero			carne de identidad:	PRESENTADO
Nacionalidad:	Boliviano			Fuente de ingresos:	PRESENTADO
Fecha Nacimiento:	1995-11-27			Documentacion declaracion patrimonial:	PRESENTADO
Direccion:	av lineares				
Genero:	masculino				

DOCUMENTOS

Plan de pagos: [Imprimir](#)

Contrato: [Imprimir](#)

Fuente y elaboración: Propia

Figura 3.61 Interfaz principal de encargado operativo

CPBSYS

Dario fernandez lopez
Encargado de operaciones

Inicio
Solicitudes
Pago
Salir

> Dashboard

[Solicitudes](#)

Fuente y elaboración: Propia

Figura 3.62 Interfaz lista de solicitudes

>Listado de solicitudes

Codigo	Prestatario	CI	Nombre	Tiempo	Interes	Monto	Fecha	Estado	Estado
3	emstel	9999999	Yuri nina	24	20	16600	2020-11-23	Monto Observado	Ver Editar

Fuente y elaboración: Propia

Figura 3.63 Interfaz de pagos cliente

> Lista de Pagos de Cliente:

Codigo	CI	Nombre	Monto	Mora	Fecha limite	Fecha Pago	Glosa	Pagado	Accion
1	3333333	Rodolfo ubrinovich	414.53	0	2020-12-26	2020-11-30		PAGADO	Comprobante Editar
2	3333333	Rodolfo ubrinovich	414.52728427164	0	2021-01-26			PENDIENTE	Editar
3	3333333	Rodolfo ubrinovich	414.52728427164	0	2021-02-26			PENDIENTE	Editar
4	3333333	Rodolfo ubrinovich	414.52728427164	0	2021-03-26			PENDIENTE	Editar
5	3333333	Rodolfo ubrinovich	414.52728427164	0	2021-04-26			PENDIENTE	Editar

Fuente y elaboración: Propia

Figura 3.64 Interfaz principal de sub gerente

> Dashboard

Solicitudes Clientes Pagos

Fuente y elaboración: Propia

Figura 3.65 Interfaz de solicitud para la aprobación de monto

alejandra perez mollinedo
Subgerente

Inicio

Solicitudes

Lista

Operaciones

Salir

>Listado de solicitudes

Codigo	Prestatario	CI	Nombre	Tiempo	Interes	Monto	Fecha	Estado	Estado
9	Estrella	5456694LP	daniel gomez	24	20	12200	2020-11-30	Validacion Solicitud	Ver Editar

Fuente y elaboración: Propia

Figura 3.66 Interfaz para ver los datos de solicitudes

alejandra perez mollinedo
Subgerente

Inicio

Solicitudes

Operaciones

Salir

DATOS PERSONALES

Numero documento: 5456694LP
Nombre: daniel
Apellido Paterno: gomez
Apellido Materno:
Estado civil: SOLTERO
Estado civil: SOLTERO
Nacionalidad: Boliviano
Fecha Nacimiento: 1945-02-18
Direccion: av linares
Genero:

DATOS TRABAJO

Empresa: Estrella
Cargo: Tecnico
Sueldo: 3500
Gastos: 1500
Direccion: sopocachi

DATOS SOLICITUD

Tipo Solicitud: préstamo de libre disponibilidad
Tiempo: 24
Interes: 20
Monto: 12200

DOCUMENTOS

carne de identidad: PRESENTADO
Fuente de ingresos: PRESENTADO
Documentacion declaracion patrimonial: PRESENTADO

DOCUMENTOS

Plan de pagos: Imprimir

ACCIONES A REALIZAR

Aprobar monto solicitado

Observar monto solicitado

Fuente y elaboración: Propia

Figura 3.67 Interfaz principal de cajero

Brenda Olmos Ibaros
Cajero

Inicio

Pago

Cuentas

Salir

Buscar pagos

Buscar por CI: Ingrese numero de CI

Buscar Nombres: Nombre Apellido paterno Apellido Materno

Buscar

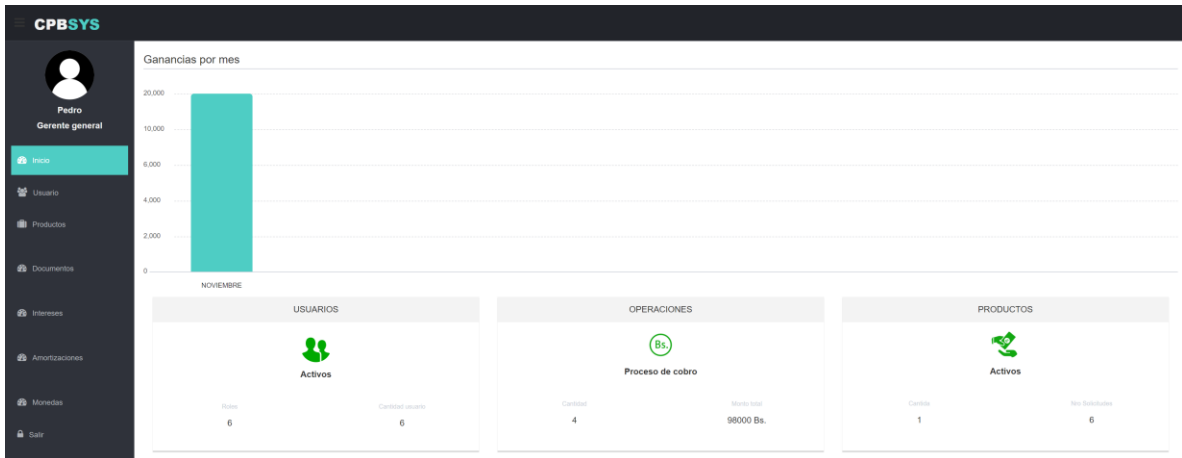
Fuente y elaboración: Propia

Figura 3.68 Interfaz lista de clientes y pagos

Codigo	CI	Nombre	Apellido Paterno	Apellido Materno	Accion
1	1111111	Belen	Vargas	linares	Ver Editar
2	2222222	Daniela	Romero	Alvarez	Ver Editar
3	8888888	daniel	fernandes	mollinedo	Ver Editar
4	3333333	Rodolfo	ubrinovich	blanco	Ver Editar
5	8326396	israel	rivadeneira	lopez	Ver Editar
6	8654569	trunk	capsula	fernandez	Ver Editar
7	1345678	reynaldo	rada	jimenez	Ver Editar

Fuente y elaboración: Propia

Figura 3.69 Interfaz principal gerente general



Fuente y elaboración: Propia

Figura 3.70 Interfaz de los usuarios y sus roles

> Lista de usuarios

Agregar nuevo usuario

Codigo	Nombre	Correo	Rol
1	Reynaldo Rada Jimenez	daniel.rada2010@hotmail.com	Plataforma
2	Dario fernandez lopez	dario@gmail.com	Encargado de operaciones
3	alejandra perez molinedo	alejandra@hotmail.com	Subgerente
4	Brenda Olmos larios	brenda@hotmail.com	Cajero
5	Pedro	pedro@hotmail.com	Gerente general
6	Miguel	miguel@hotmail.com	Administrador

Fuente y elaboración: Propia

Figura 3.71 Interfaz de los productos

> Lista de Productos

Generar Nuevo Producto

Nombre	Descripcion	Amortizacion	Moneda	Interes	Tiempo minimo	Tiempo maximo	Monto minimo	Monto maximo	Acciones
préstamo de libre disponibilidad		tasa fija	Bolivianos	interes1	24	48	10000	70000	Ver Editar

Fuente y elaboración: Propia

Figura 3.72 Interfaz de creación de productos y asignación de requisitos

CPBSYS

Pedro
Gerente general

Inicio
Usuario
Productos
Documentos
Intereses
Amortizaciones
Monedas
Salir

> Registro de Productos

Nombre

Descripción

Amortización: tasa fija

Interés: interés

Moneda: Bolivianos

Tiempo mínimo

Tiempo máximo

Monto mínimo

Monto Máximo

Requisitos

carnet de identidad

Fuente de ingresos

Documentación declaración patrimonial

Estabilidad laboral

Carga Financiera Total/Renta Total

Documentación de respaldo de deudas vigentes

Registrar

Fuente y elaboración: Propia

Figura 3.73 Interfaz vista de documentos para los requisitos

CPBSYS

Pedro
Gerente general

Inicio
Usuario
Productos
Documentos
Intereses
Amortizaciones
Monedas
Salir

> Lista de Documentos

Generar Nueva Documento

Codigo	Nombre	Descripcion	Obligatorio	
1	carnet de identidad		1	Ver Editar
2	Fuente de ingresos		1	Ver Editar
3	Documentación declaración patrimonial		1	Ver Editar
4	Estabilidad laboral		1	Ver Editar
5	Carga Financiera Total/Renta Total		1	Ver Editar
6	Documentación de respaldo de deudas vigentes		1	Ver Editar

Fuente y elaboración: Propia

Figura 3.74 Interfaz valor de interés para los productos

Codigo	Nombre	Monto	Acciones
1	interes1	20	Ver Editar
2	interes2	20	Ver Editar

Fuente y elaboración: Propia

Figura 3.75 Interfaz de tipo de amortización

Codigo	Nombre	Monto	Acciones
1	tasa fija	0	Ver Editar

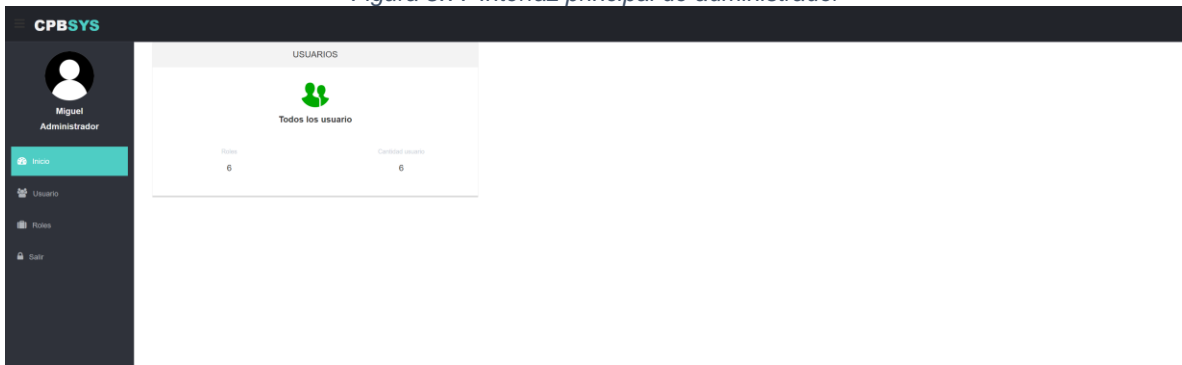
Fuente y elaboración: Propia

Figura 3.76 Interfaz tipo de monedas

Codigo	Nombre	Cambio	Acciones
BS	Bolivianos	1	Ver Editar

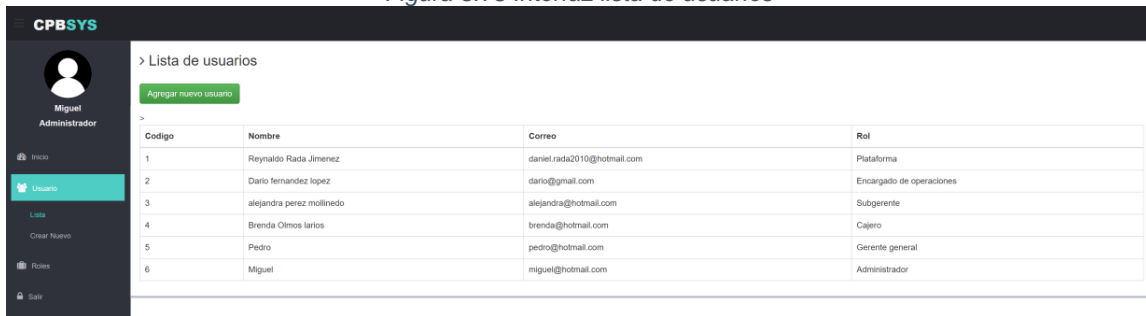
Fuente y elaboración: Propia

Figura 3.77 Interfaz principal de administrador



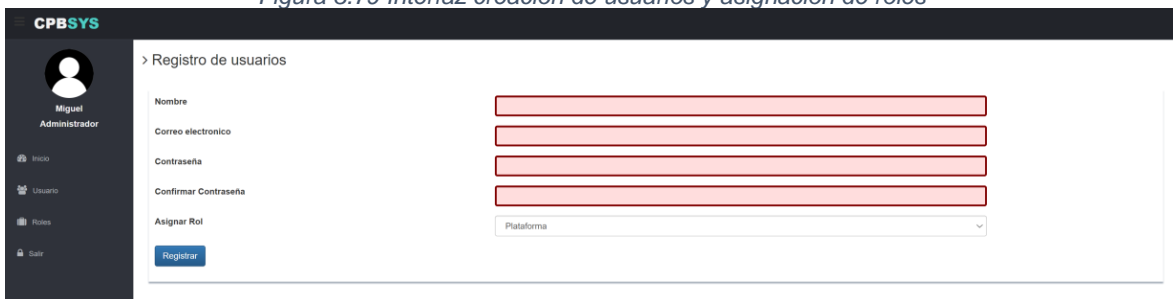
Fuente y elaboración: Propia

Figura 3.78 Interfaz lista de usuarios



Fuente y elaboración: Propia

Figura 3.79 Interfaz creación de usuarios y asignación de roles



Fuente y elaboración: Propia

Figura 3.80 Interfaz de lista de roles



Fuente y elaboración: Propia

CAPÍTULO IV

CALIDAD, PRUEBAS Y SEGURIDAD

4 CALIDAD, PRUEBAS Y SEGURIDAD.

Habiendo identificado las características y requisitos de calidad de software del proyecto, como el ciclo de vida, flexibilidad, portabilidad, etc. Se optó por utilizar el estándar de la ISO 9126, que organiza los factores de calidad en tres ejes o puntos de vista desde los cuales el usuario puede contemplar la calidad de un producto.

Cada factor determinante de la calidad se descompone, a su vez, en una serie de criterios o propiedades que determinan su calidad. Los criterios pueden ser evaluados mediante un conjunto de métricas. Para cada criterio deben fijarse unos valores máximos y mínimos aceptables para cada criterio.

4.1 MÉTRICAS DE CALIDAD

Funcionalidad

La funcionalidad no se puede medir directamente, por esta razón corresponde derivar medidas directas como es el punto función que cuantifica el tamaño y la complejidad del sistema en términos de las funciones del usuario. Los puntos de función se derivan usando una relación empírica basada en medidas contables (directas) del dominio de información del software y en valoraciones cualitativas de la complejidad del software.

Tabla 4.1 Factores de ponderación

Valor de dominio de información	Conteo	Factor ponderado			Total
		Simple	Promedio	Complejo	
Entradas externas (EE)	20	3	4	6	80
Salidas externas (SE)	40	4	5	7	200
Consultas externas (CE)	10	3	4	6	40
Archivos lógicos internos (ALI)	21	7	10	15	210
Archivos de interfaz (AI)	0	5	7	10	0
Conteo total					530

Fuente: Ingeniería de Software (R. Pressman, 2010). Elaboración: Propia

Tabla 4.2 Valores de ajuste de complejidad

Factor de complejidad	Valoración del factor						Total
	No influye	Incidental	Moderado	Medio	Significativo	Esencial	
¿Requiere el Sistema copias de Seguridad y de recuperación fiables?	0	1	2	3	4	5	5
¿Se requiere comunicación de datos?	0	1	2	3	4	5	3
¿Existen funciones de procesamiento distribuido?	0	1	2	3	4	5	4
¿Es crítico el rendimiento?	0	1	2	3	4	5	4
¿Será ejecutado el sistema en un entorno operativo existente y fuertemente utilizado?	0	1	2	3	4	5	3
¿Requiere el sistema entrada de datos interactiva?	0	1	2	3	4	5	4
¿Requiere la entrada de datos interactiva que las transacciones se lleven a cabo en múltiples pantallas?	0	1	2	3	4	5	3
¿Se actualizan los archivos maestros de forma interactiva?	0	1	2	3	4	5	4
¿Son complejas las entradas, las salidas, los archivos o las peticiones?	0	1	2	3	4	5	3
¿Es complejo el procesamiento interno?	0	1	2	3	4	5	4
¿Se ha diseñado el código para ser utilizado?	0	1	2	3	4	5	4
¿Están incluidas en el diseño la conversión y la instalación?	0	1	2	3	4	5	3
¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	0	1	2	3	4	5	2
¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente usada por el usuario?	0	1	2	3	4	5	5
Total							51

Fuente: Ingeniería de Software (R. Pressman, 2010). Elaboración: Propia

Para calcular los puntos de función se utiliza la siguiente ecuación:

$$PT = \text{Conteo total} * (0,65 + 0,01 * \sum F_i)$$

Reemplazando los valores se tiene:

$$PT = 530 * (0,65 + 0,01 * 51) = 614,8$$

Observando la siguiente tabla, podemos observar que el sistema tiene una

Tabla 4.3 Escala de punto función

Escala	Observación
PF > 300	Optimo
200 < PF < 300	Bueno
100 < PF < 200	Suficiente
PF < 100	Deficiente

Fuente y elaboración: Ingeniería de Software (R. Pressman, 2010).

Si calculamos al 100% el nivel de confianza considerando $\sum F_i = 51$ como el máximo valor de ajuste de complejidad entonces se tiene:

$$PT_{max} = \text{Conteo total} * (0,65 + 0,01 * \sum F_i)$$

Reemplazando valores:

$$PT_{max} = 614,8 * (0,65 + 0,01 * 51)$$

$$PT_{max} = 713.19$$

Con el máximo valor de ajuste de complejidad se tiene la funcionalidad real:

$$\text{Funcionalidad} = \frac{614.8}{713.19} = 0,86$$

Entonces la funcionalidad del sistema es de 86%, esto quiere decir que el sistema tiene 86% de probabilidad de funcionamiento sin riesgo de fallo y con operatividad constante y 14% de colapso de sistema.

Usabilidad

La usabilidad es la facilidad de uso, esta métrica nos muestra el costo de aprender

el producto, lo cual se calcula con la siguiente formula:

$$FU = \left(\frac{\sum x_i}{n} \right) * 100$$

Tabla 4.4 Ajuste de preguntas

Preguntas	Respuestas		Ponderación
	Si	No	
¿Puede utilizar con facilidad el sistema?	2	1	0.67
¿Puede controlar operaciones que el sistema solicite?	2	1	0.67
¿Las respuestas del sistema son complicadas?	3	0	1
¿El sistema permitió la retroalimentación de información?	2	1	0.67
¿El sistema cuenta con interfaz agradable a la vista?	2	1	0.67
¿La respuesta del sistema es satisfactoria?	3	0	1
¿Le parece complicadas las funciones del sistema?	1	2	0.33
¿Se hace difícil aprender a manejar el sistema?	2	1	0.67
¿Los resultados que proporciona el sistema facilitan el trabajo?	1	2	0.33
¿Durante el uso del sistema se produjo errores?	2	1	0.67
Total			7.69

Fuente: Ingeniería de Software (R. Pressman, 2010). Elaboración: Propia

Reemplazando en la fórmula:

$$FU = \left(\frac{\sum x_i}{n} \right) * 100$$

$$FU = \left(\frac{7.69}{10} \right) * 100$$

$$FU = 76.9$$

Existe un 76.9% de comprensión o entendimiento de los usuarios con respecto a la capacidad del sistema.

Confiabilidad

La confiabilidad se define como la probabilidad de que un sistema, activo o componente lleve a cabo su función adecuadamente durante un periodo bajo condiciones operacionales previamente definidas y constantes.

Para calcular la confiabilidad del sistema se toma en cuenta el periodo de tiempo en el cual se ejecuta y se obtiene muestras, se utilizan las siguientes ecuaciones:

$$\text{Confiabilidad} = \frac{MTBF}{MTBF + MTTR} * 100$$

$$MTBF = \left(\frac{h_t}{p}\right) * 100$$

$$MTTR = \left(\frac{h_p}{p}\right) * 100$$

Donde:

MTBF, tiempo total que funciona el sistema sin fallar.

MTTR, tiempo total de reparación del sistema.

h_t, horas trabajadas o de marcha durante el periodo de evaluación.

h_p, horas de paro durante el periodo de evaluación.

p, numero de fallas durante la evaluación.

Calculando *MTBF* y *MTTR*:

$$MTBF = \left(\frac{360}{6}\right) * 100$$

$$MTTR = \left(\frac{48}{6}\right) * 100$$

Reemplazando en la ecuación de la confiabilidad:

$$\text{Confiabilidad} = \frac{6000}{6000 + 800} * 100$$

En un periodo de prueba de 15 días (360 horas), el sistema muestra un índice de confiabilidad de 88.23% Confiabilidad.

Mantenibilidad

El mantenimiento se desarrolla para mejorar el sistema en respuesta a los nuevos requerimientos que la empresa tenga y los reglamentos que está regida por la misma. El estándar IEE94 sugiere un índice de madurez del software (IMS) que proporciona un indicador en la estabilidad de un producto, se lo determina con la siguiente formula:

$$IMS = \frac{M_T - (F_a + F_c + F_d)}{M_T}$$

Donde:

M_T , número de módulos de la versión actual.

F_c , números de módulos en la versión actual que se han cambiado.

F_a , número de módulos en la versión actual que se han añadido.

F_d , número de módulos de la versión anterior que se han borrado en la versión actual.

Reemplazando los valores:

$$IMS = \frac{6 - (0 + 1 + 0)}{6}$$

Con lo que se puede decir que el nuevo sistema tiene una estabilidad de 0.83 que indica la facilidad de mantenimiento, el 83% restante es el margen de error correspondiente a los cambios y modificaciones.

4.2 PRUEBA DEL SISTEMA

Las pruebas de seguridad sirven para encontrar la presencia de errores en el sistema que pueden generar las entradas, salida de los datos o comportamientos inapropiados durante su uso.

4.2.1 Pruebas de caja negra

Se realizará la prueba en el acceso al sistema para ver si cumple la función asignada.

Figura 4.1 Prueba de caja negra del inicio de sesión

La imagen muestra una interfaz de usuario para iniciar sesión. El encabezado es un recuadro de color verde claro con el texto "INICIAR SESION" en mayúsculas blancas. Debajo del encabezado, hay un formulario con tres campos de entrada: "Usuario", "Contraseña" y "Recuérdame" (con un cuadro de verificación). Al final del formulario, hay un botón de color verde claro con el texto "Iniciar Sesion" y un ícono de candado a la izquierda.

Fuente y elaboración: Propia

Tabla 4.5 Descripción de prueba de caja negra inicio sesión

Descripción prueba de caja negra				
Datos de entrada			Resultado	
Usuario & Contraseña			Acceso al sistema	
Tipo de flujo de datos				
Archivo	Pantalla	Informe	Formulario	interno
La estructura de datos viaja con el flujo de datos				
Iniciar Sesión - Usuario & Contraseña				
Comentarios: El usuario digita la información requerida, El sistema verifica la información, si los datos son correctos ingresa al sistema.				

Fuente y elaboración: Propia

Tabla 4.6 Prueba de caja negra registro usuario

The image shows a user registration form with the following elements:

- Nombre: Text input field
- Correo electrónico: Text input field
- Contraseña: Text input field
- Confirmar Contraseña: Text input field
- Asignar Rol: Dropdown menu with "Plataforma" selected
- Registrar: Blue button

Fuente y elaboración: Propia

Tabla 4.7 Descripción de prueba de caja negra de registro usuario

Descripción prueba de caja negra				
Datos de entrada			Resultado	
Nombre, Usuario, Contraseña,			Registro de usuario	
Confirmar contraseña y asignar rol				
Tipo de flujo de datos				
Archivo	Pantalla	Informe	Formulario	interno
La estructura de datos viaja con el flujo de datos				

Registrar – Nombre, Usuario, Contraseña, Confirmar contraseña y asignar rol

Comentarios: **El administrador digita la información requerida, El sistema verifica la información, si los datos son correctos se registra el nuevo usuario.**

Fuente y elaboración: Propia

4.3 SEGURIDAD

4.3.1 Seguridad a nivel de base de datos

Se hace uso del gestor de base de datos Mariadb proporciona encriptación y autenticación a todos los niveles, además brinda extensiones para distintas funcionalidades como ser encriptación de datos.

4.3.2 Seguridad a nivel de aplicación

Tomando en cuenta las recomendaciones más relevantes especificadas en la norma ISO 27002 con respecto a la presentación de las características de confiabilidad, integridad y disponibilidad de la información se incorpora las siguientes medidas de seguridad del sistema.

Tabla 4.8 Recomendaciones ISO 27002

Recomendaciones ISO medidas de seguridad incorporadas en el sistema 27002	
Control de accesos	Se implemento como elemento importante la autenticación de usuario que consta usuario y contraseña el usuario deberá estar previamente autenticado para realizar cualquier acción, caso contrario será restringido

Controles criptográficos	Se implemento la encriptación de la contraseña de los usuarios con el uso de algoritmo de cifrado AES 256.
Registro de actividad y supervisión	Se controla los registros de información mediante la validación de datos

Fuente y elaboración: Propia

CAPÍTULO V

ANÁLISIS DE COSTOS

5 ANÁLISIS DE COSTOS.

La estimación del costo y los beneficios que se espera obtener con el desarrollo e implementación del sistema. Haciendo uso del modelo de estimación de costos COCOMO II que permite realizar estimaciones en función del tamaño del software, y de un conjunto de factores de costos

5.1 CALCULO DE COSTOS

Para el cálculo del costo total del software desarrollado, se utilizó la herramienta “USC-COCOMO II.2000.4” que pertenece al Centro de Sistemas e Ingeniería de software de la Universidad de California del Sur.

Figura 5.1 Herramienta: USC-COCOMO II.2000.4

The screenshot shows the USC-COCOMO II.2000.4 software interface. The main window displays a table with the following data:

Module Name	Module Size	LINKR Rate (\$/month)	EAF	Language	MM Effort DEV	EST Effort DEV	PRCD	COST	INST COST	Staff	RISK
Sistema	F:12230	300.00	1.00	Object-Orient	46.2	46.2	265.0	13855.55	1.1	3.7	0.0

At the bottom of the window, there is a summary table:

Estimated	Effort	Sched	PRCD	COST	INST	Staff	RISK
Optimistic	36.9	11.6	331.2	11084.44	0.9	3.2	
Most Likely	46.2	12.4	265.0	13855.55	1.1	3.7	0.0
Pessimistic	57.7	13.3	212.0	17919.44	1.4	4.9	

Additional summary information shown:

- Total Lines of Code: 12230
- Hours/PW: 152.00

Fuente y elaboración: Propia

El cálculo se basó a los siguientes datos introducidos

Tabla 5.1 Datos introducidos en USC-COCOMO II.2000.4

Campo	Valor
Modelo de desarrollo	Post-Arquitectura
Lenguaje de desarrollo	PHP con multiplicador de 20
Salario Mensual	\$300 (Bs 2100)

Fuente y elaboración: Propia

Los parámetros de medida para los puntos de función fueron los siguientes:

Tabla 5.2 Parámetros de medida

Parámetro de medidas	Cantidad
Nro. De entradas de usuario	7
Nro. De salidas de usuario	18
Nro. Peticiones de usuario	7
Nro. De archivos	0
Nro. De interfaces externas	2

Fuente y elaboración: Propia

El resultado obtenido fue los siguientes:

Figura 5.2 Resultados de USC-COCOMO II.2000.4

Project Name: CPBSYS		Scale Factor: 18.97		Schedule								
Project Notes		Development Model: Post Architecture										
X	Module Name	Module Size	LABOR Rate (\$/month)	EAF	Language	NCM Effort DEV	EST Effort DEV	PROD	COST	INST COST	Staff	RISK
	Sistema	F:11948	300.00	1.00	Object-Orient	45.0	45.0	265.6	13494.91	1.1	3.7	0.0
Total Lines of Code: 11948						Estimated	Effort Sched	PROD	COST	INST	Staff	RISK
Hours/PM: 152.00						Optimistic	36.0	11.5	332.0	10795.93	0.9	3.1
						Most Likely	45.0	12.3	265.6	13494.91	1.1	3.7
						Pessimistic	56.2	13.2	212.5	16868.64	1.4	4.3

Fuente y elaboración: Propia

Los datos obtenidos son:

- Costos de la estimación promedio del proyecto: Bs. 13494.91.
- Tiempo de desarrollo 14 meses.
- Numero de programadores 5.
- Horas/personas/mes 152.

CAPÍTULO VI

CONCLUSIONES Y RECOMENDACIONES

6 CONCLUSIONES Y RECOMENDACIONES

6.1 CONCLUSIONES

Habiendo realizado el análisis, diseño y construcción del sistema, se concluyó lo siguiente:

- Se desarrollaron los módulos de rol de usuario, aprobación, desembolso, seguimientos de pagos, cálculo de mora, búsqueda, liquidación y reportes satisfactoriamente.
- Se aplicó normas de calidad, y herramientas de programación para que el sistema tenga alta usabilidad, funcionalidad, confiabilidad, seguridad, integridad y portabilidad.
- Se logró acortar el tiempo de búsqueda y acceso a los pagos de los clientes, registro de solicitudes, verificación de los requisitos, mejorando la generación de comprobante de pagos, elaboración de contratos y plan de pagos.
- El manejo de la información relacionada a las solicitudes de préstamos de la empresa ahora se realiza con mayor velocidad, precisión y confiabilidad.
- Ahora se lleva un registro detallado de todas las transacciones de las solicitudes y pagos, mejorando el control y evitando búsquedas en los archivos.
- El módulo de reportes ayuda a la elaboración de informes periódicos y muestra información vital para la toma de decisiones.

En conclusión, se llegó a solucionar los problemas específicos y consecuentemente la problemática central y los objetivos planteados fueron cumplidos de manera satisfactoria.

6.2 RECOMENDACIONES

En base a las observaciones realizadas en la construcción durante la prueba del sistema se elaboraron las siguientes recomendaciones:

- Cambiar las contraseñas de acceso al sistema periódicamente, para mejorar la seguridad y evitar accesos no autorizados.
- Realizar copias de seguridad de la base de datos en lo posible diariamente, con el fin de evitar pérdida de información.
- Llevar un registro detallado de los errores del sistema y notificarlos al administrador.
- Revisar el manual de usuario si no se tiene claro cómo realizar una operación específica.

BIBLIOGRAFÍA

- andrew, s. C. (25 de 05 de 2020). *RUP, "Rational Unified Process?: A Best Practices Approach Topics What is RUP?? RUP best practices Software economics Adapt the process.* Obtenido de <http://www.andrew.cmu.edu/course/90-754/umlucdfaq.html>
- arlow, J., & Neustadt, I. (2005). *Uml 2 y el proceso unificado: análisis y diseño práctico orientado a objetos.* booch jacobson rumbaugh.
- ASFI. (21 de 08 de 2018). *asfi.gob.bo.* Obtenido de Autoridad de Supervisión del Sistema Financiero: http://servdmzw.asfi.gob.bo/CircularValores/circulares/ASFI_336.pdf
- Bruegge, B., & H.Dutoit, A. (s.f.). *Object-Oriented Software Engineering.* En B. B. H.Dutoit..
- Budgen, D. (2003). *Diseño de Software.* Mexico: Addison-Wesley.
- Burch, J. G. (1992). *Diseño de sistemas de información: Teoría y práctica/por John G. Burch.* España: Editorial Limusa.
- Buschmann, F., Meurier, R., Rohnert, H., Sommerlad, P., & Stal, M. (1996). *Pattern-Oriented Software Architecture A System of Patterns.* . New York: John Wiley & Sons.
- epportal. (12 de 07 de 2018). *epportal.* Obtenido de epportal: epportal.com/resto/productos/uclub/presta.html
- Garcia, R. A. (24 de 08 de 2018). *Aula finanzas.* Obtenido de Aula finanzas facil: www.aulafinanzasfacil.com/
- Gómez, A. L. (2006). *COCOMO UN MODELO DE ESTIMACION DE PROYECTOS DE SOFTWARE.*
- Gonzalez, M. A. (24 de 08 de 2018). *conceptosdefinanza.* Obtenido de Generalidades de finanzas: conceptosdefinanza.blogspot.com/2013/06/generalidades-de-la-finanzas.html
- Guadaña Quiroz, I., & Pachamora Pinedo, J. D. (2016). *RUP y UML: Un estudio sobre ¿qué es?, ¿para qué se usa? y ¿cómo se desarrolla? Un Diagrama de Caso de Uso.* Peru.

informática, D. (15 de 07 de 2018). *divinf*. Obtenido de Division de informatica:
<http://www.divinf.com.ar/software-internet-loan.html>

Paez, S. d. (15 de 07 de 2018). *sistemaspaetz*. Obtenido de Sistemas de información
paez: <http://sistemaspaetz.com/prestacobpro-lan/>

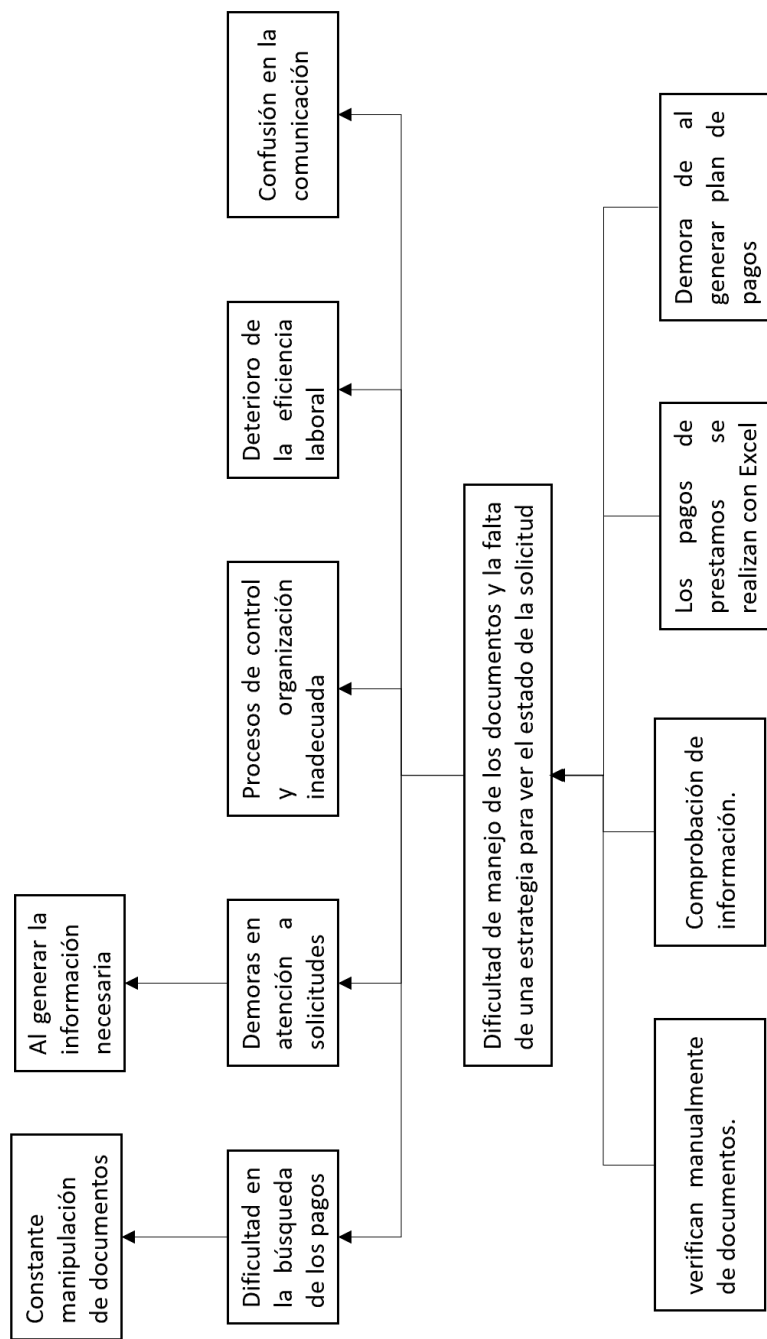
RAMIREZ MOLINARES, C., GARCIA BARBOZA, M., PANTOJA ALGARIN, C., &
ZAMBRANO MEZA, A. (2009). *FUNDAMENTOS DE MATEMÁTICAS
FINANCIERAS*. Cartagena: Universidad libre colombia.

Sommerville, I. (2011). *Ingeniería de Software*. Mexico: Pearson Education.

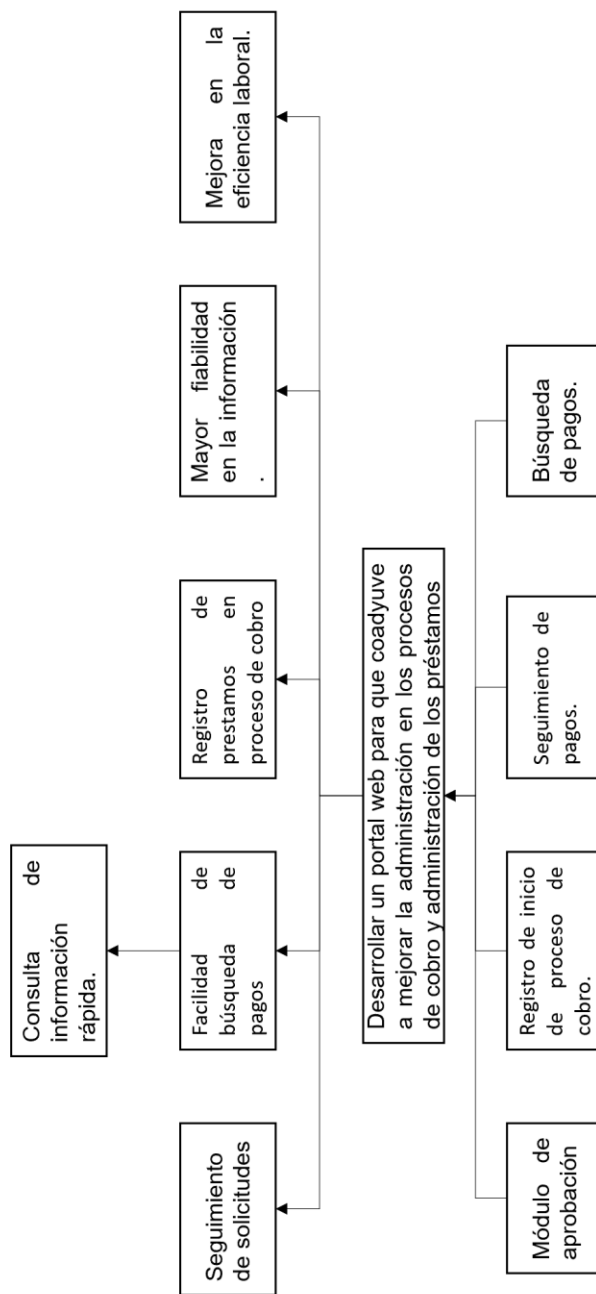
wikipedia. (15 de 07 de 2018). Obtenido de wikipedia enciclopedia libre:
https://es.wikipedia.org/wiki/Proceso_Unificado_de_Rational

ANEXOS

ANEXO A. ARBOL DE PROBLEMAS



ANEXO B. ARBOL DE OBJETIVOS



ANEXO C. COMPROBANTE DE PAGO

CAJA POPULAR BOLIVIANA SRL
Préstamo Bs. : 9010653434
COMPROBANTE DE PAGO
Cuota 3
fecha de proceso 04-01-2012
Ximena

Tercer pago 02/01/2012

capital	497,68
valor al interés	871,43
mora	28
formularios	15
	1412,11

Saldo 28037,42
proximo pago martes 01/02/2012

nota el saldo solo es del capital


Firma del cliente



ANEXO D. PLAN DE PAGOS

PLAN DE PAGOS PARA EL CREDITO DE XIMENA MACHICADO PLATA CON C.I. 8305001 LP
 ESTE CREDITO ESTA SIENDO DESEMBOLSADO PARA EL PAGO EN 36 MESES PLAZO
 AL PIE DEL PRESENTE FIRMA COMO CONSTANCIA DEL DESEMBOLSO O Y RECEPCION DEL EFECTIVO
 DE FECHA 04 DE OCTUBRE DEL 2011 A HRS.13,00

CUOTA	FECHA	SALDO	CAPITAL	INTERES	FORMULARIO	CUOTA
1	03/11/2011	29.520,89	469,11	900	15	1394,11
2	03/12/2011	29.047,71	483,18	885,93	15	1394,11
3	02/01/2012	28.550,03	497,68	871,43	15	1394,11
4	01/02/2012	28.037,42	512,61	856,5	15	1394,11
5	02/03/2012	27.509,43	527,99	841,12	15	1394,11
6	01/04/2012	26.965,60	543,83	825,28	15	1394,11
7	01/05/2012	26.405,46	560,14	808,97	15	1394,11
8	31/05/2012	25.828,51	576,95	792,16	15	1394,11
9	30/06/2012	25.234,26	594,25	774,86	15	1394,11
10	30/07/2012	24.622,18	612,08	757,03	15	1394,11
11	29/08/2012	23.991,74	630,44	738,67	15	1394,11
12	28/09/2012	23.342,38	649,36	719,75	15	1394,11
13	28/10/2012	22.673,54	668,84	700,27	15	1394,11
14	27/11/2012	21.984,64	688,9	680,21	15	1394,11
15	27/12/2012	21.275,07	709,57	659,54	15	1394,11
16	26/01/2013	20.544,21	730,86	638,25	15	1394,11
17	25/02/2013	19.791,43	752,78	616,33	15	1394,11
18	27/03/2013	19.016,06	775,37	593,74	15	1394,11
19	26/04/2013	18.217,43	798,63	570,48	15	1394,11
20	26/05/2013	17.394,84	822,59	546,52	15	1394,11
21	25/06/2013	16.547,58	847,26	521,85	15	1394,11
22	25/07/2013	15.674,90	872,68	496,43	15	1394,11
23	24/08/2013	14.776,04	898,86	470,25	15	1394,11
24	23/09/2013	13.850,21	925,83	443,28	15	1394,11
25	23/10/2013	12.896,61	953,6	415,51	15	1394,11
26	22/11/2013	11.914,40	982,21	386,9	15	1394,11
27	22/12/2013	10.902,72	1.011,68	357,43	15	1394,11
28	21/01/2014	9.860,69	1.042,03	327,08	15	1394,11
29	20/02/2014	8.787,40	1.073,29	295,82	15	1394,11
30	22/03/2014	7.681,91	1.105,49	263,62	15	1394,11
31	21/04/2014	6.543,26	1.138,65	230,46	15	1394,11
32	21/05/2014	5.370,45	1.172,81	196,3	15	1394,11
33	20/06/2014	4.162,45	1.208,00	161,11	15	1394,11
34	20/07/2014	2.918,21	1.244,24	124,87	15	1394,11
35	19/08/2014	1.636,65	1.281,56	87,55	15	1394,11
36	18/09/2014	316,64	1.320,01	49,1	15	1394,11

ANEXO E. MANUAL DE USUARIO

el inicio de sesión este inicio de sesión ingrese su usuario y contraseña



INICIAR SESION

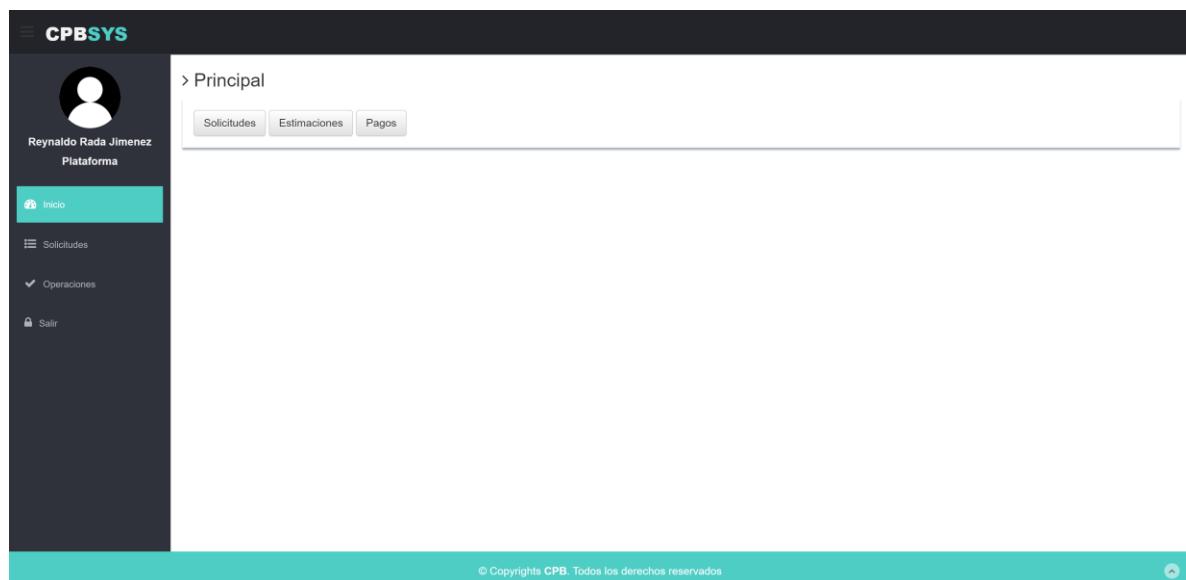
Usuario

Contraseña

Recuérdame

Iniciar Sesión

Plataforma



CPBSYS

Reynaldo Rada Jimenez
Plataforma

Inicio

Solicitudes

Operaciones

Salir

> Principal

Solicitudes Estimaciones Pagos

© Copyrights CPB. Todos los derechos reservados

CPBSYS

Reynaldo Rada Jimenez
Plataforma

Inicio
Solicitudes
Operaciones
Salir

>Listado de solicitudes

Generar Nueva Solicitud

Codigo	Prestatario	CI	Nombre	Tiempo	Interes	Monto	Fecha	Estado	Estado
8	Estrella	1345678	reynaldo rada	24	20	25000	2020-11-29	CONTRATO GENERADO	Ver Editar

CPBSYS

Reynaldo Rada Jimenez
Plataforma

Inicio
Solicitudes
Operaciones
Salir

Solicitud de Prestamos
completa los 4 pasos

1 Datos Personales

2 Trabajo

3 Ingresos

4 Referencias

Datos Personales
Documento de Identidad

Nombre

Apellido Paterno

Apellido Materno

Estado Civil

Nacionalidad

Fecha Nacimiento

Dirección

Genero

Guardar

CPBSYS

Reynaldo Rada Jimenez
Plataforma

Inicio
Solicitudes
Operaciones
Salir

Solicitud de Prestamos
completa los 4 pasos

1 Datos Personales

2 Trabajo

3 Ingresos

4 Referencias

Trabajo

Empresa

Cargo

Salario

Salario

Dirección

Autenticar Registrar

CPBSYS

Reynaldo Rada Jimenez
Plataforma

Inicio
Solicitudes
Operaciones
Salir

Solicitud de Prestamos

complete los 4 pasos

1 Datos Personales 2 Trabajo 3 Ingresos 4 Verificación

Solicitud:

Producto:

Tiempo:

Interes:

Monto:

[Previaos](#) [Siguiente](#)

CPBSYS

Reynaldo Rada Jimenez
Plataforma

Inicio
Solicitudes
Operaciones
Salir

Solicitud de Prestamos

complete los 4 pasos

1 Datos Personales 2 Trabajo 3 Ingresos 4 Verificación

Verificacion:

carnet de identidad:

Fuente de ingresos:

Documentacion declaracion patrimonial:

Estabilidad laboral:

Carga Financiera Total/Renta Total:

Documentacion de respaldo de deudas vigentes:

[Previaos](#) [Generar Sol](#)

CPBSYS

Reynaldo Rada Jimenez
Plataforma

Inicio
Solicitudes
Operaciones
Salir

>Listado de operacion

Codigo	Prestatario	CI	Nombre	Couta restante	Estado	
8	Estrella	1345678	reynaldo rada	5	Cartera Activa	Refinanciar

CPBSYS

Reynaldo Rada Jimenez
Plataforma

- Inicio
- Solicitudes
- Operaciones
- Salir

DATOS PERSONALES		DATOS TRABAJO		DATOS SOLICITUD	
Numero documento:	1345678	Empresa:	Estrella	Tipo Solicitud:	préstamo de libre disponibilidad
Nombre:	reynaldo	Cargo:	Tecnico	Tiempo:	24
Apellido Paterno:	rada	Sueldo:	3500	Interes:	20
Apellido Materno:	jimenez	Gastos:	500	Monto:	25000
Estado civil:	soltero	Direccion:	prueba	DOCUMENTOS	
Estado civil:	soltero			carnet de identidad:	PRESENTADO
Nacionalidad:	Boliviano			Fuente de ingresos:	PRESENTADO
Fecha Nacimiento:	1995-11-27			Documentacion declaracion patrimonial:	PRESENTADO
Direccion:	av lineares				
Genero:	masculino				

DOCUMENTOS

Plan de pagos: [Imprimir](#)

Contrato: [Imprimir](#)

Encargado Operativo

CPBSYS

Dario fernandez lopez
Encargado de operaciones

- Inicio
- Solicitudes
- Pago
- Salir

> Dashboard

[Solicitudes](#)

CPBSYS

Dario fernandez lopez
Encargado de operaciones

- Inicio
- Solicitudes
- Lista
- Pago
- Salir

>Listado de solicitudes

Codigo	Prestatario	Ci	Nombre	Tiempo	Interes	Monto	Fecha	Estado	Estado
3	emsitel	9999999	Yuri nina	24	20	16600	2020-11-23	Monto Observado	Ver Editar

CPBSYS

Dario fernandez lopez
Encargado de operaciones

- Inicio
- Solicitudes
- Pago
- Salir

> Lista de Pagos de Cliente:

Codigo	CI	Nombre	Monto	Mora	Fecha limite	Fecha Pago	Glosa	Pagado	Accion
1	3333333	Rodolfo ubrinovich	414.53	0	2020-12-26	2020-11-30		PAGADO	Comprobante Editar
2	3333333	Rodolfo ubrinovich	414.52728427164	0	2021-01-26			PENDIENTE	Editar
3	3333333	Rodolfo ubrinovich	414.52728427164	0	2021-02-26			PENDIENTE	Editar
4	3333333	Rodolfo ubrinovich	414.52728427164	0	2021-03-26			PENDIENTE	Editar
5	3333333	Rodolfo ubrinovich	414.52728427164	0	2021-04-26			PENDIENTE	Editar

Subgerencia

CPBSYS

alejandra perez mollinedo
Subgerente

- Inicio
- Solicitudes
- Operaciones
- Salir

> Dashboard

Solicitudes Cientes Pagos

CPBSYS

alejandra perez mollinedo
Subgerente

- Inicio
- Solicitudes
- Lista
- Operaciones
- Salir

> Listado de solicitudes

Codigo	Prestatario	CI	Nombre	Tiempo	Interes	Monto	Fecha	Estado	Estado
9	Estrella	5456694LP	daniel gomez	24	20	12200	2020-11-30	Validacion Solicitud	Ver Editar

CPBSYS

alejandra perez mollinedo
Subgerente

- Inicio
- Solicitudes
- Operaciones
- Salir

DATOS PERSONALES		DATOS TRABAJO		DATOS SOLICITUD	
Numero documento:	5456094LP	Empresa:	Estrella	Tipo Solicitud:	préstamo de libre disponibilidad
Nombre:	daniel gomez	Cargo:	Tecnico	Tiempo:	24
Apellido Paterno:		Sueldo:	3500	Interes:	20
Apellido Materno:		Gastos:	1500	Monto:	12200
Estado civil:	SOLTERO	Direccion:	sopocachi	DOCUMENTOS	
Estado civil:	SOLTERO			carnet de identidad:	PRESENTADO
Nacionalidad:	Boliviano			Fuente de ingresos:	PRESENTADO
Fecha Nacimiento:	1945-02-18			Documentacion declaracion patrimonial:	PRESENTADO
Direccion:	av lineas				
Genero:					

DOCUMENTOS

Plan de pagos: Imprimir

ACCIONES A REALIZAR

Aprobar monto solicitado
Observar monto solicitado

Cajas

CPBSYS

Brenda Olmos Iarios
Cajero

- Inicio
- Pago
- Cuentas
- Salir

Buscar pagos

Buscar por CI:

Buscar Nombres:

CPBSYS

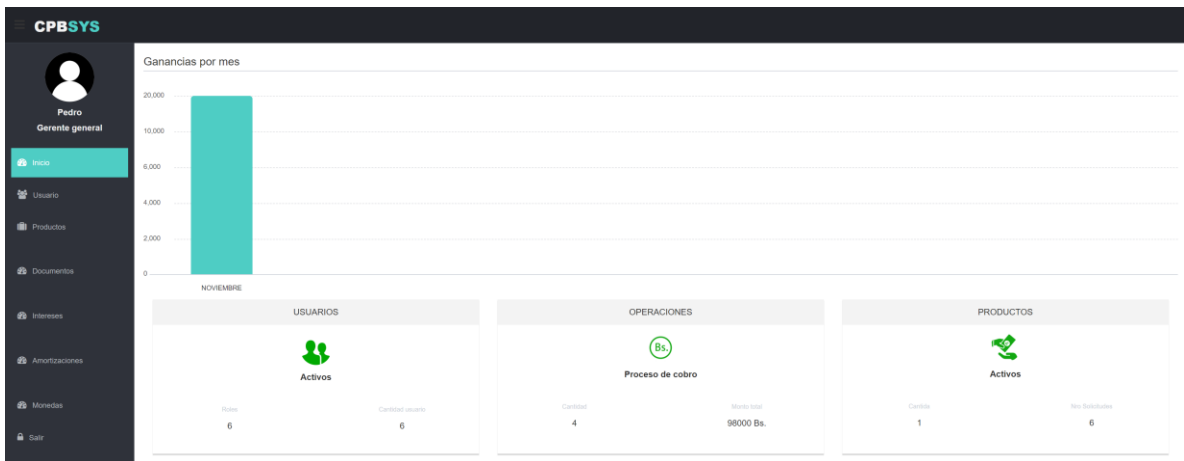
Brenda Olmos Iarios
Cajero

- Inicio
- Pago
- Cuentas
- Salir

> Lista de Clientes

Codigo	CI	Nombre	Apellido Paterno	Apellido Materno	Accion
1	1111111	Belen	Vargas	lineas	Ver Editar
2	2222222	Daniela	Romero	Alvarez	Ver Editar
3	8888888	daniel	fernandes	mollinedo	Ver Editar
4	3333333	Rodolfo	ubrinovich	blanco	Ver Editar
5	8326396	Israel	rivadeneira	lopez	Ver Editar
6	8654589	trunk	capsula	fernandez	Ver Editar
7	1345678	reynaldo	rada	jimenez	Ver Editar

Gerente General



CPBSYS

Pedro
Gerente general

Inicio
Usuario
Lista
Productos
Documentos
Intereses
Amortizaciones
Monedas
Salir

> Lista de usuarios

Agregar nuevo usuario

Codigo	Nombre	Correo	Rol
1	Reynaldo Rada Jimenez	daniel.rada2010@hotmail.com	Plataforma
2	Dario fernandez lopez	dario@gmail.com	Encargado de operaciones
3	alejandra perez molinedo	alejandra@hotmail.com	Subgerente
4	Brenda Olmos larios	brenda@hotmail.com	Cajero
5	Pedro	pedro@hotmail.com	Gerente general
6	Miguel	miguel@hotmail.com	Administrador

CPBSYS

Pedro
Gerente general

Inicio
Usuario
Productos
Documentos
Intereses
Amortizaciones
Monedas
Salir

> Lista de Productos

Generar Nuevo Producto

Nombre	Descripcion	Amortizacion	Moneda	Interes	Tiempo minimo	Tiempo maximo	Monto minimo	Monto maximo	Acciones
préstamo de libre disponibilidad		tasa fija	Bolivianos	Interes1	24	48	10000	70000	Ver Editar

CPBSYS

Pedro
Gerente general

Inicio
Usuario
Productos
Documentos
Intereses
Amortizaciones
Monedas
Salir

Registro de Productos

Nombre:

Descripcion:

Amortizacion: tasa fija

Interes: interes1

Moneda: Bolivianos

Tiempo minimo:

Tiempo maximo:

Monto minimo:

Monto Maximo:

Requisitos

carnet de identidad	<input type="checkbox"/>
Fuente de ingresos	<input type="checkbox"/>
Documentacion declaracion patrimonial	<input type="checkbox"/>
Estabilidad laboral	<input type="checkbox"/>
Carga Financiera Total/Renta Total	<input type="checkbox"/>
Documentación de respaldo de deudas vigentes	<input type="checkbox"/>

[Registrar](#)

CPBSYS

Pedro
Gerente general

Inicio
Usuario
Productos
Documentos
Intereses
Amortizaciones
Monedas
Salir

Lista de Documentos

[Generar Nueva Documento](#)

Codigo	Nombre	Descripcion	Obligatorio	
1	carnet de identidad		1	Ver Editar
2	Fuente de ingresos		1	Ver Editar
3	Documentacion declaracion patrimonial		1	Ver Editar
4	Estabilidad laboral		1	Ver Editar
5	Carga Financiera Total/Renta Total		1	Ver Editar
6	Documentación de respaldo de deudas vigentes		1	Ver Editar

CPBSYS

Pedro
Gerente general

Inicio
Usuario
Productos
Documentos
Intereses
Amortizaciones
Monedas
Salir

Lista de Intereses

[Generar Nuevo Interes](#)

Codigo	Nombre	Monto	Acciones
1	interes1	20	Ver Editar
2	interes2	20	Ver Editar

CPBSYS

Pedro
Gerente general

> Lista de Amortizaciones Generar Nueva Amortizacion

Codigo	Nombre	Monto	Acciones
1	tasa fija	0	Ver Editar

CPBSYS

Pedro
Gerente general

> Lista de Monedas Generar Nueva Moneda

Codigo	Nombre	Cambio	Acciones
BS	Bolivianos	1	Ver Editar

Administrador

CPBSYS

Miguel
Administrador

Inicio

USUARIOS

Todos los usuario

Usuarios	Cantidad usuarios
6	6

CPBSYS

Miguel
Administrador

Inicio

Usuarios

Lista

Crear Nuevo

Roles

Salir

> Lista de usuarios Agregar nuevo usuario

Codigo	Nombre	Correo	Rol
1	Reynaldo Rada Jimenez	daniel.rada2010@hotmail.com	Plataforma
2	Dario fernandez lopez	dario@gmail.com	Encargado de operaciones
3	alejandra perez molinedo	alejandra@hotmail.com	Subgerente
4	Brenda Olmos larios	brenda@hotmail.com	Cajero
5	Pedro	pedro@hotmail.com	Gerente general
6	Miguel	miguel@hotmail.com	Administrador

CPBSYS

Miguel
Administrador

Inicio
Usuario
Roles
Salir

> Registro de usuarios

Nombre

Correo electronico

Contraseña

Confirmar Contraseña

Asignar Rol

CPBSYS

Miguel
Administrador

Inicio
Usuario
Roles
Salir

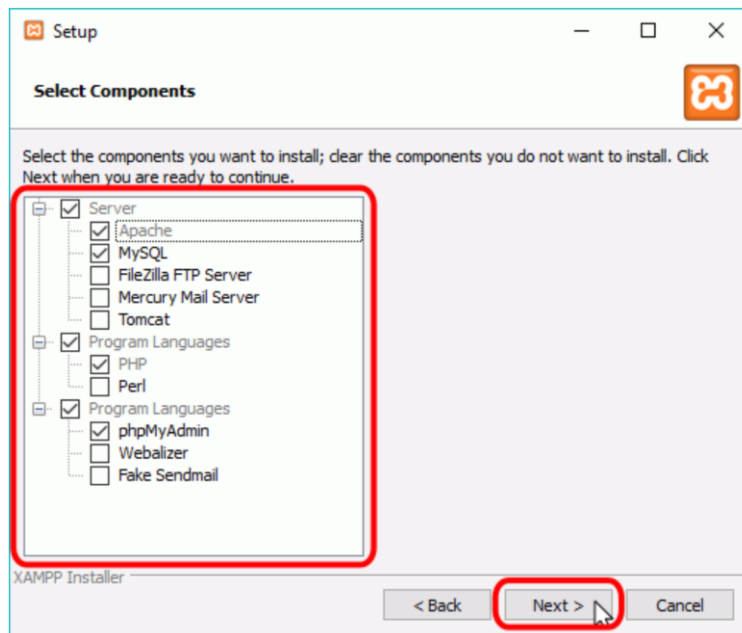
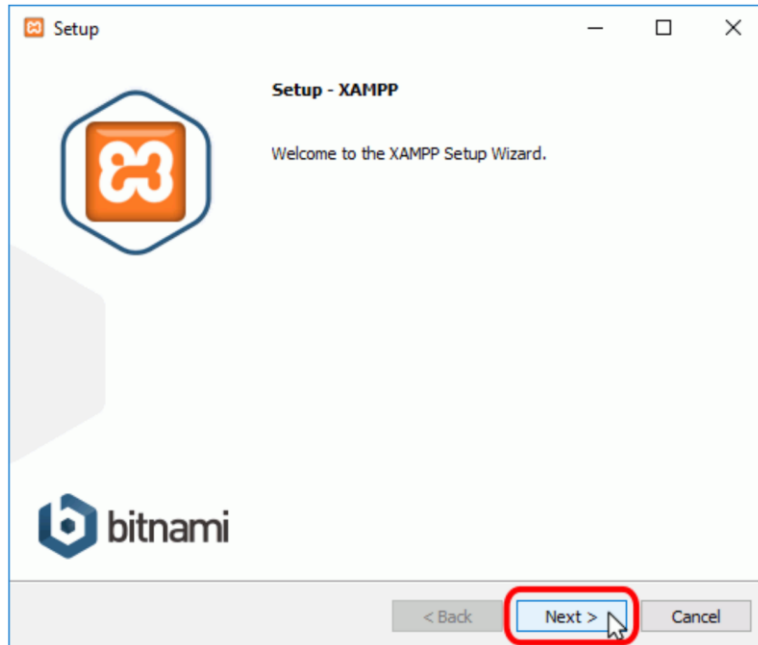
> Lista de Roles

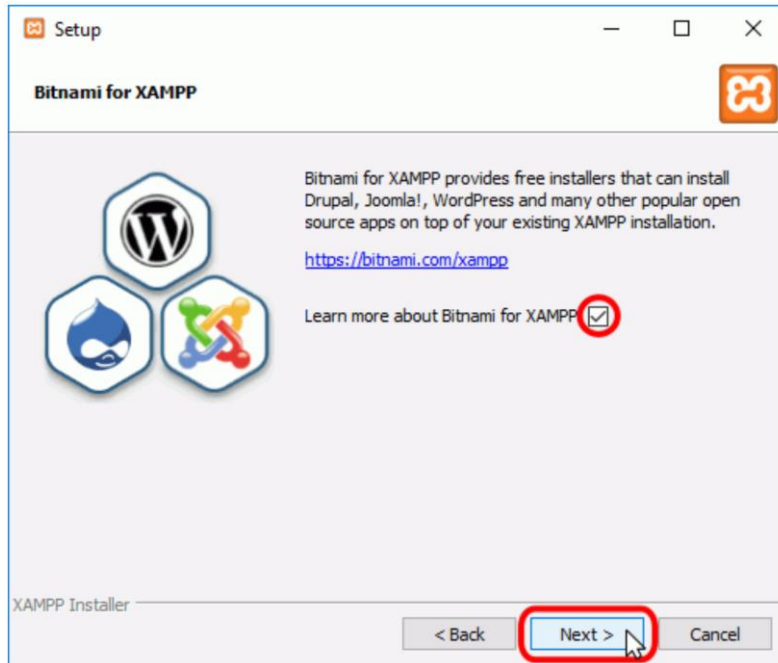
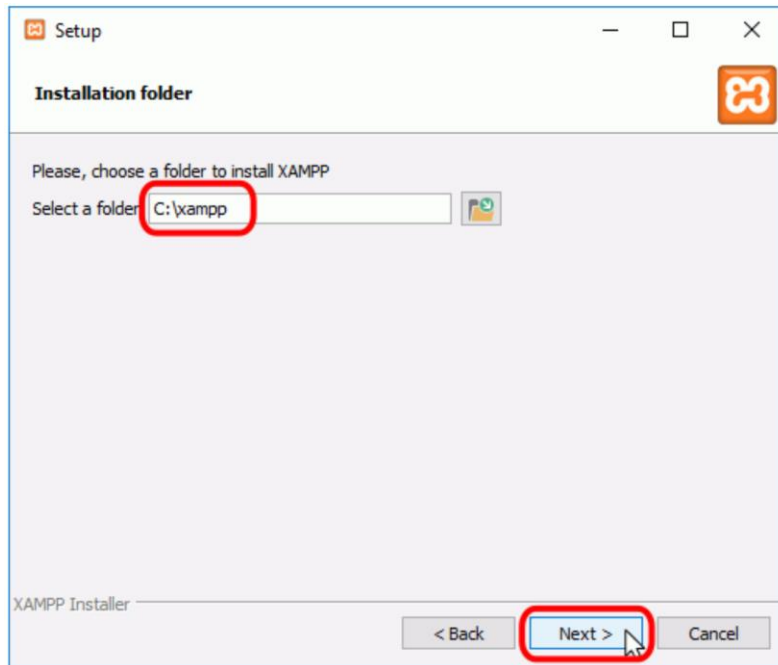
ID	Codigo	Descripcion
1	1	Plataforma
2	2	Encargado de operaciones
3	3	Subgerente
4	4	Cajero
5	5	Gerente general
6	6	Administrador

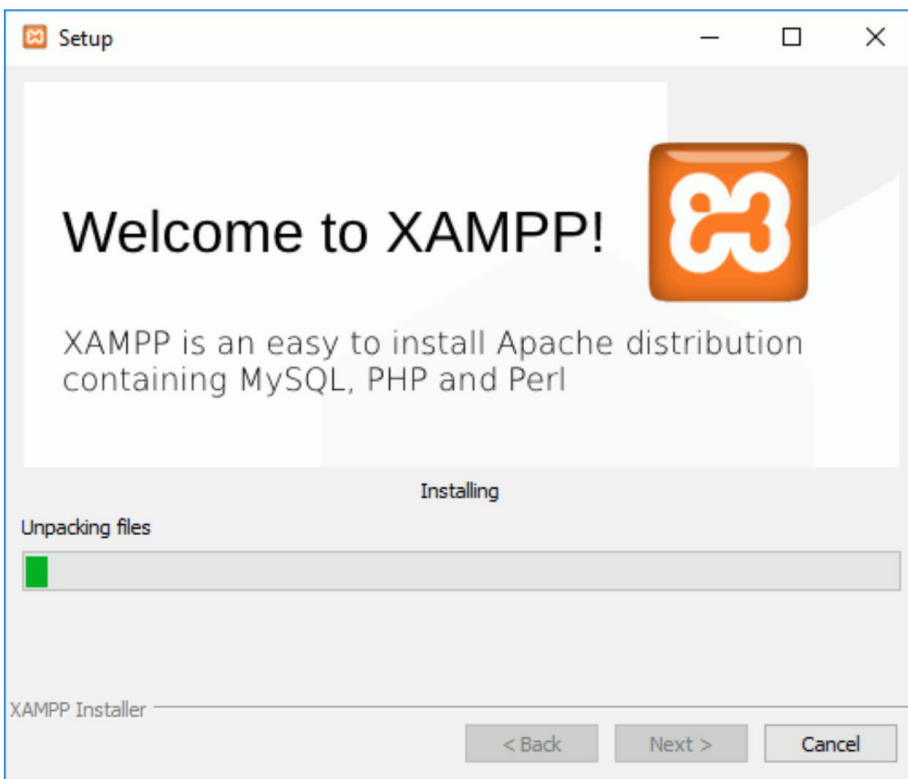
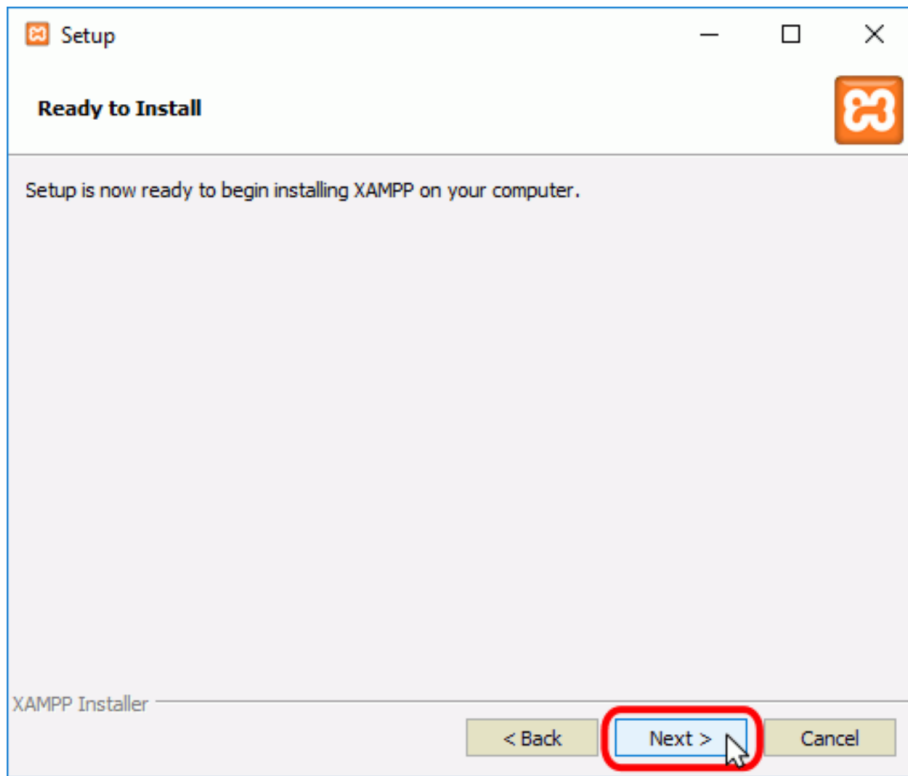
ANEXO F. MANUAL TÉCNICO

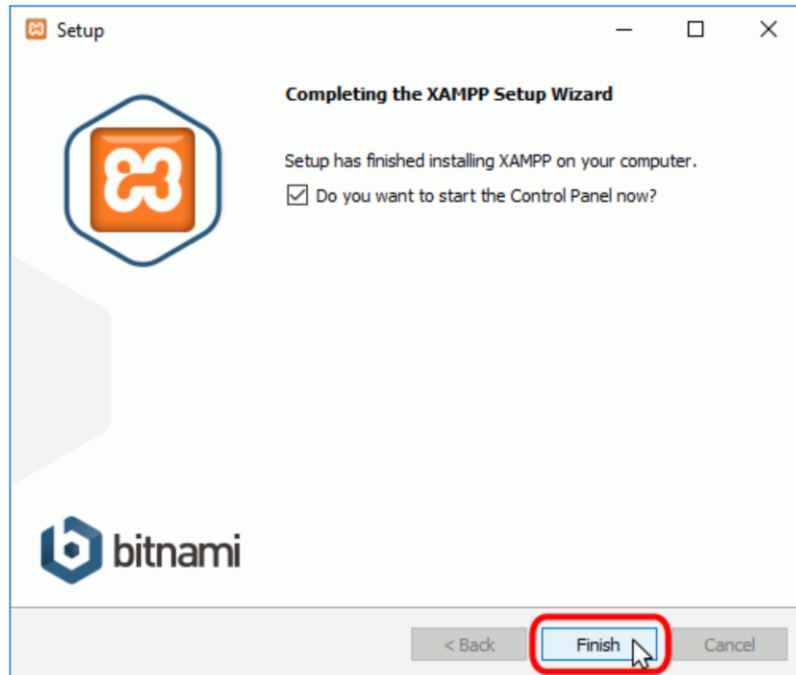
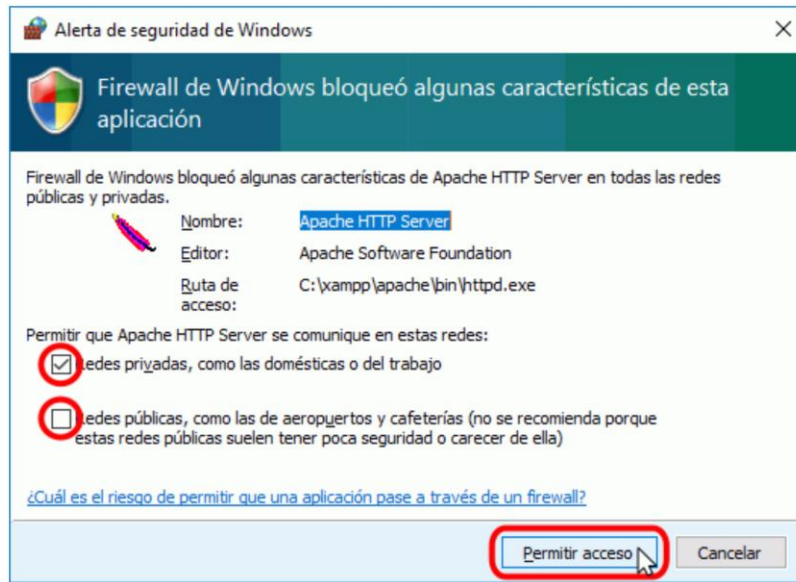
Manual de técnico

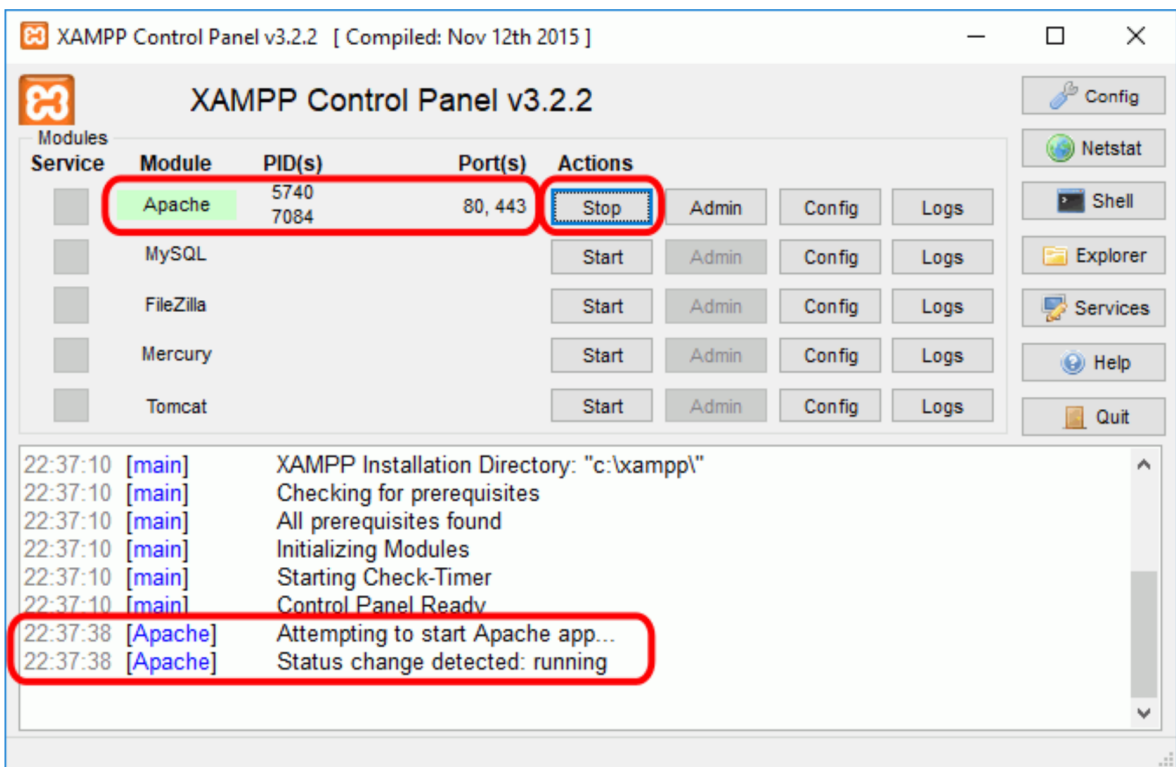
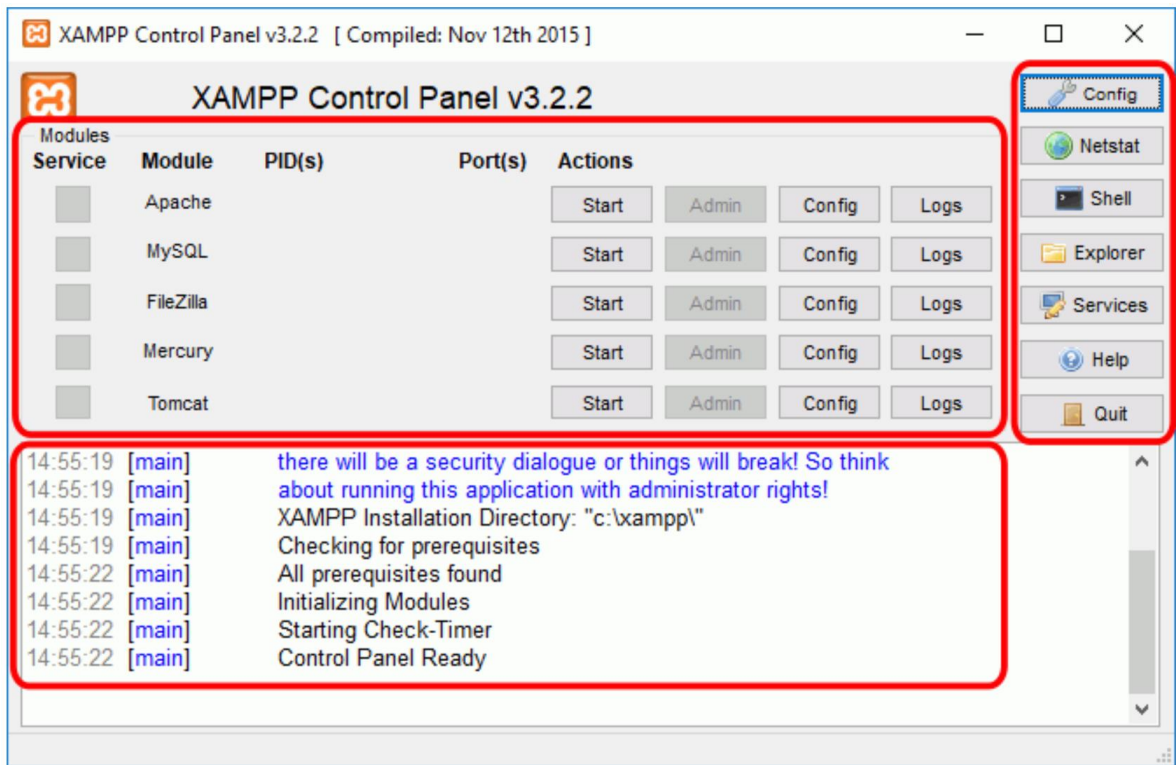
1. Instalación de Xampp en windows

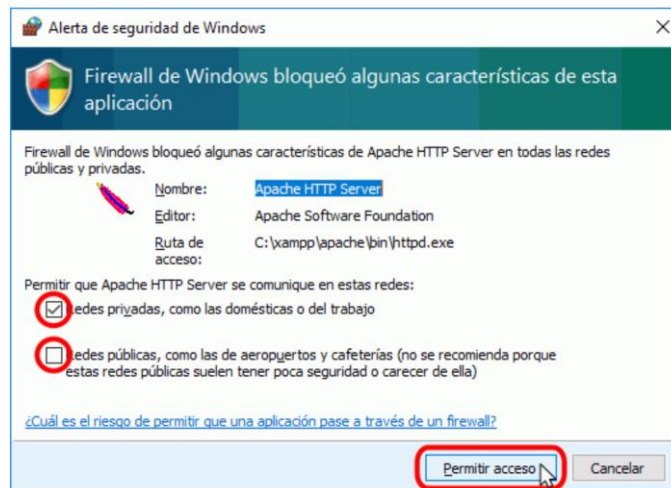
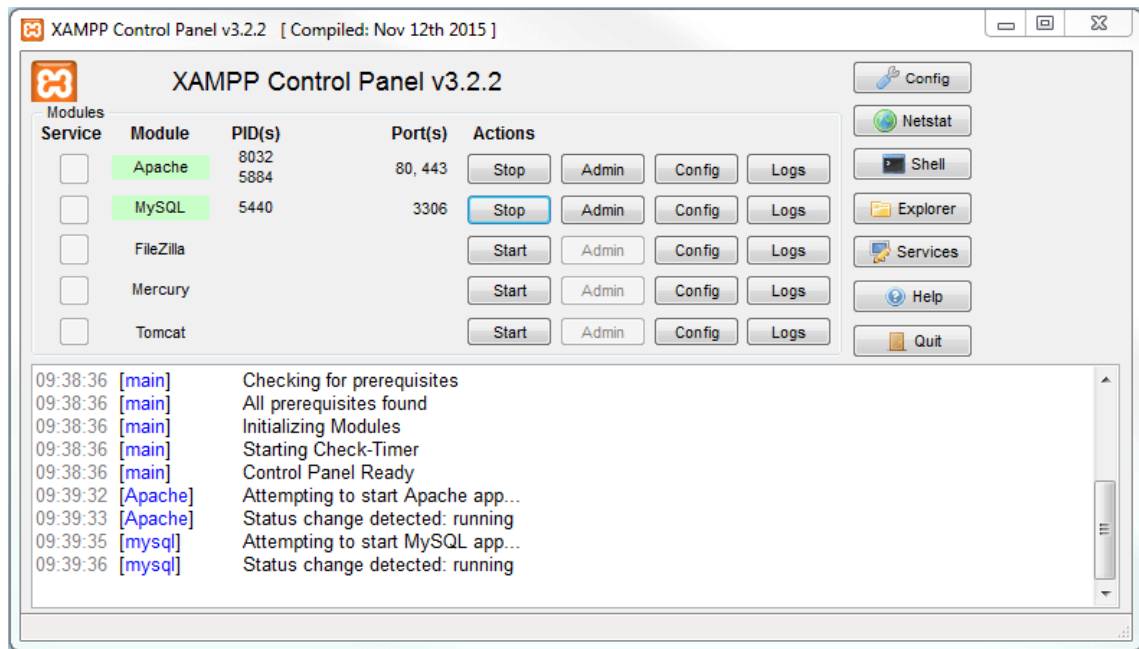






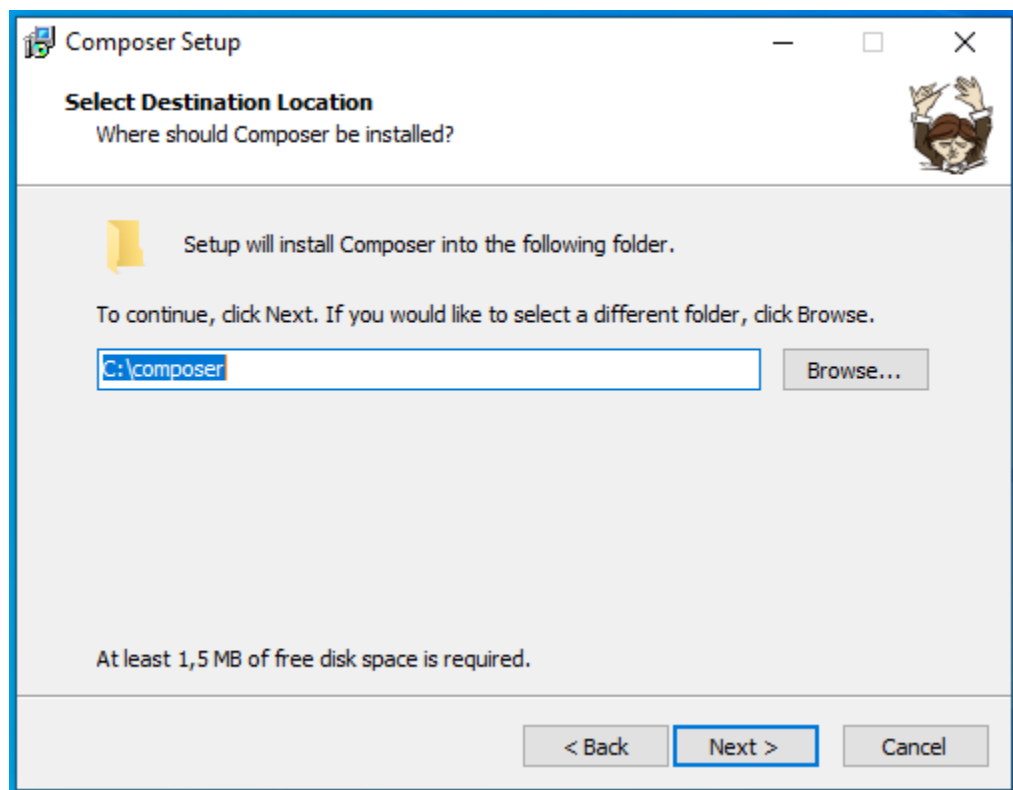
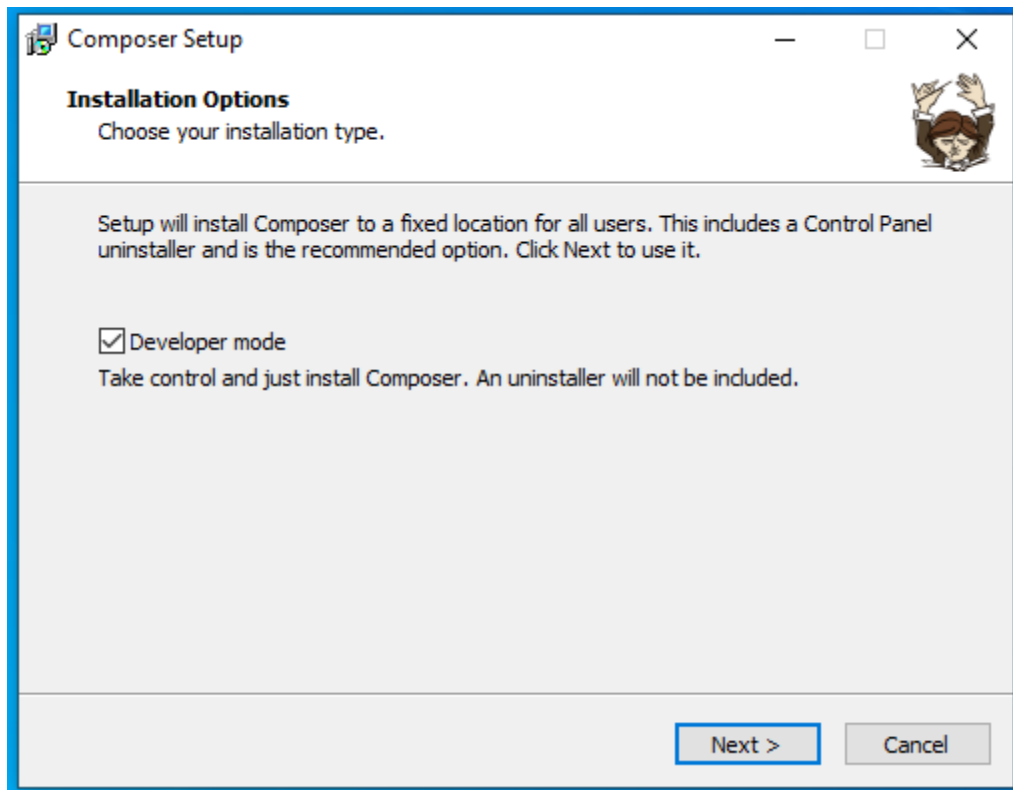


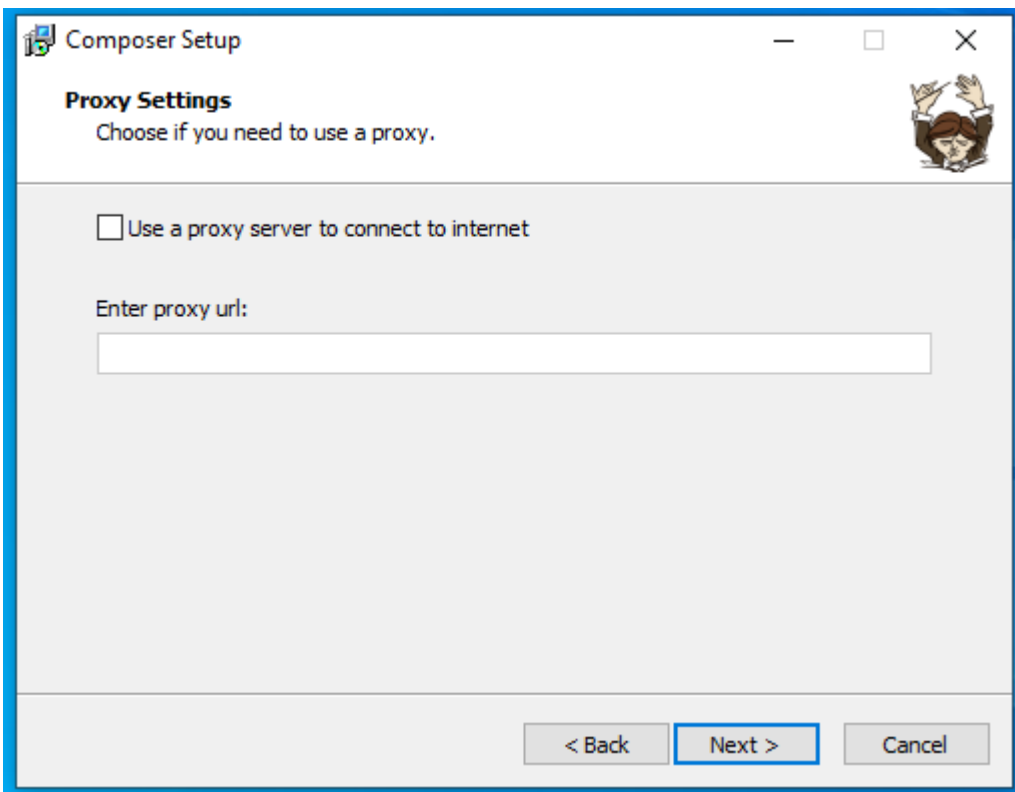
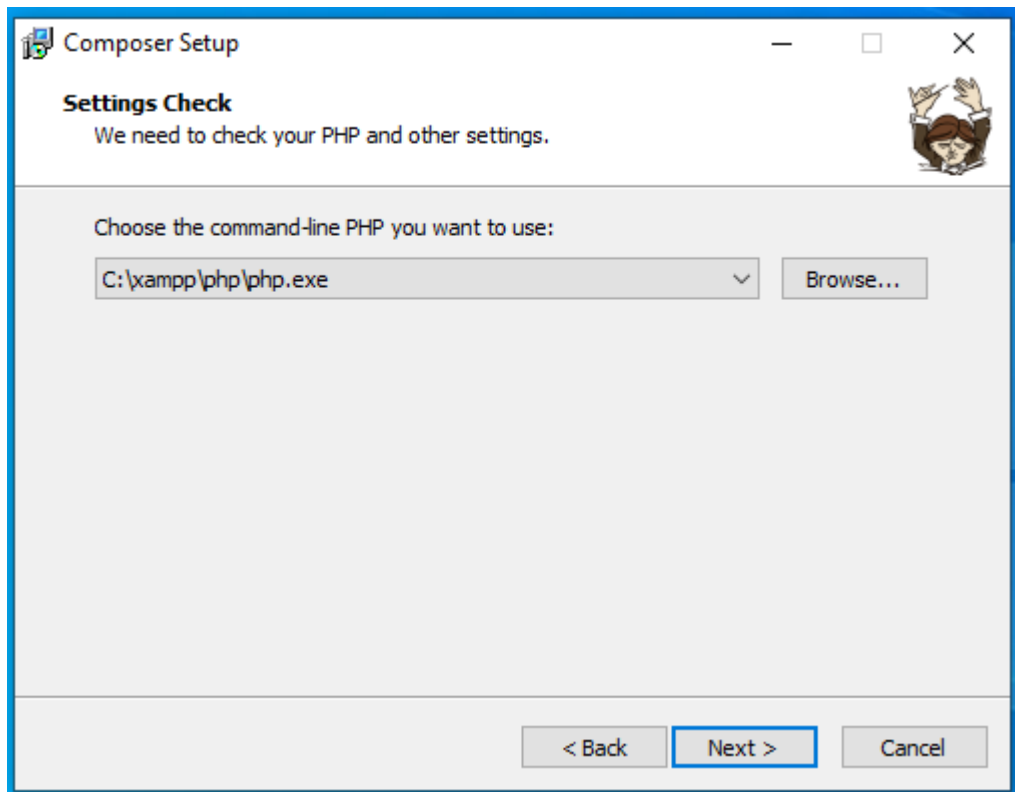


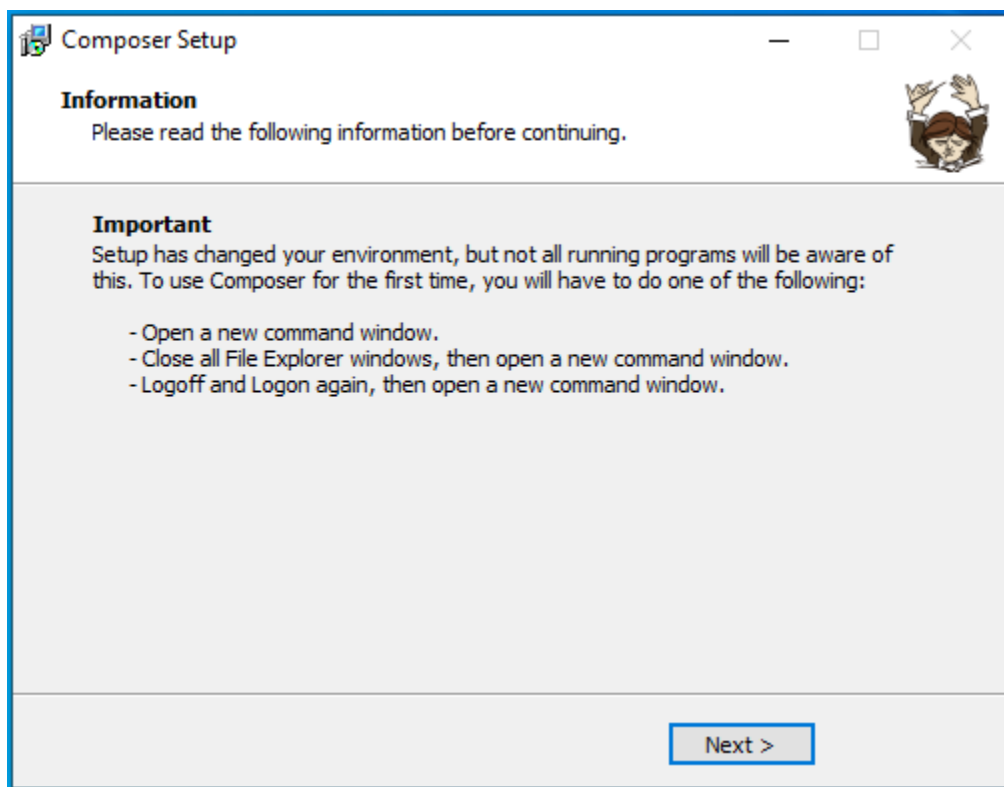
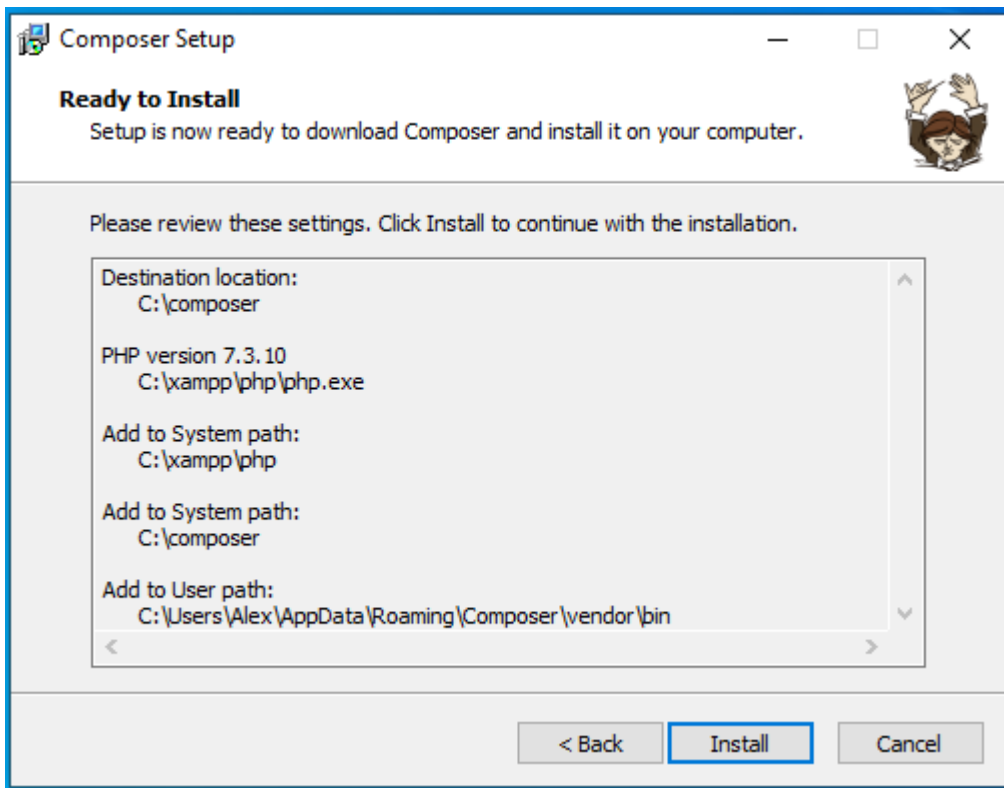


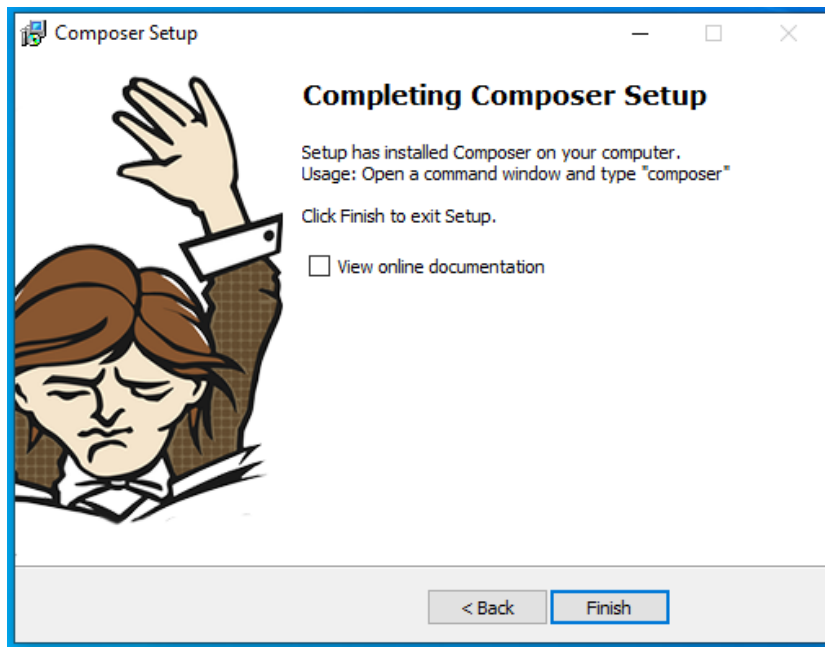
2. Instalación de composer

Una vez dentro de la web, nos dirigiremos al apartado donde pone Windows Installer y clickaremos sobre Composer-Setupo.exe.









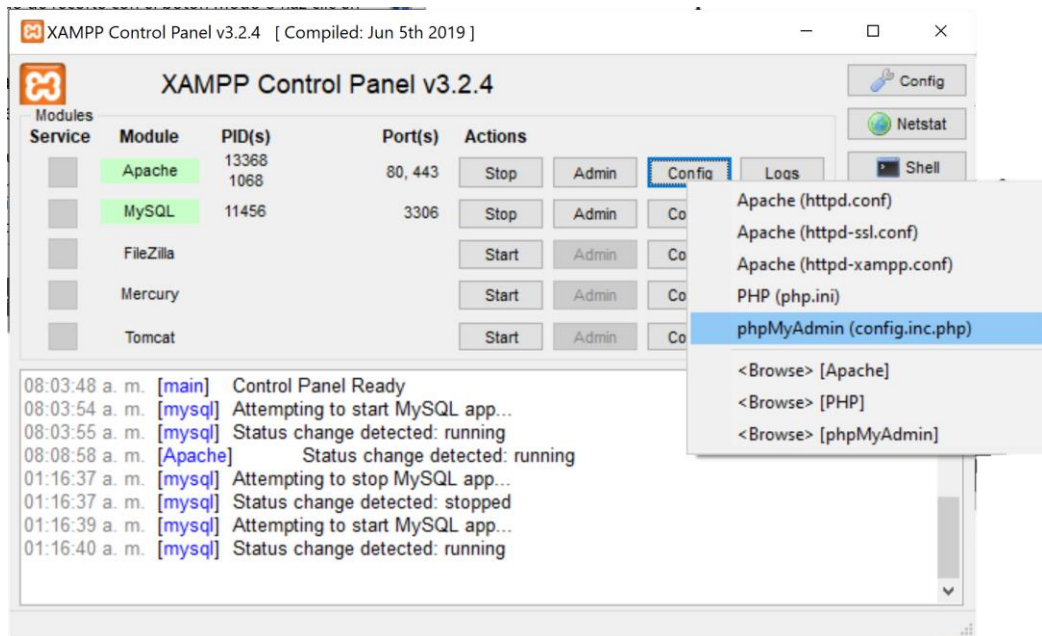
```
C:\Users\Daniel Rada>composer

Composer version 2.0.7 2020-11-13 17:31:06

Usage:
  command [options] [arguments]

Options:
  -h, --help                Display this help message
  -q, --quiet               Do not output any message
  -V, --version             Display this application version
  --ansi                    Force ANSI output
  --no-ansi                 Disable ANSI output
  -n, --no-interaction     Do not ask any interactive question
  --profile                 Display timing and memory usage information
  --no-plugins              Whether to disable plugins.
  -d, --working-dir=WORKING-DIR If specified, use the given directory as working directory.
  --no-cache                Prevent use of the cache
  -v|vv|vvv, --verbose     Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and
                             3 for debug
```

3. Creación de base de datos



```

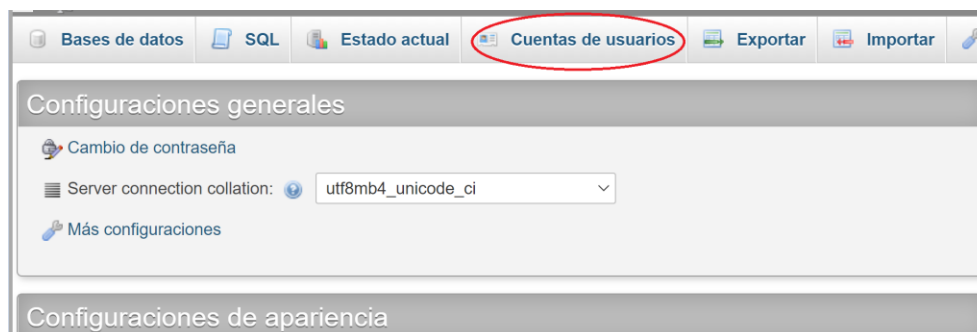
/* Authentication type and info */
$cfg['Servers'][$i]['auth_type'] = 'config';
$cfg['Servers'][$i]['user'] = 'root';
$cfg['Servers'][$i]['password'] = '';
$cfg['Servers'][$i]['extension'] = 'mysqli';
$cfg['Servers'][$i]['AllowNoPassword'] = true;
$cfg['Lang'] = '';

```

Establecer una contraseña para el usuario de la contraseña

Inicie su navegador y escriba en url <http://localhost/phpmyadmin>

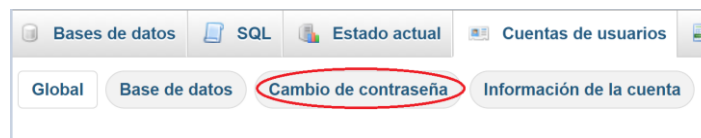
Entre a cuentas de usuarios



Editar privilegios

	Nombre de usuario	Nombre del servidor	Contraseña	Privilegios globales	Grupo de usuario	Conceder	Acción
<input type="checkbox"/>	cualquiera	%	No	USAGE		No	Editar privilegios Exportar
<input type="checkbox"/>	pma	localhost	No	USAGE		No	Editar privilegios Exportar
<input type="checkbox"/>	root	127.0.0.1	No	ALL PRIVILEGES		Si	Editar privilegios Exportar
<input type="checkbox"/>	root	:::1	No	ALL PRIVILEGES		Si	Editar privilegios Exportar
<input type="checkbox"/>	root	localhost	No	ALL PRIVILEGES		Si	Editar privilegios Exportar

Cambiar contraseña



Es recomendado generar la contraseña y guardarla en algún lugar seguro si ingresa un contraseña manual mente es recomendable usar contraseñas alfanuméricas

Cambio de contraseña

Sin contraseña

Contraseña:

Ingresar: Strength: Good

Debe volver a escribir:

Hashing de la contraseña: Autenticación de MySQL nativo

Generar contraseña

Cambiar la palabra config por cookie

```

/* Authentication type and info */
$cfg['Servers'][$i]['auth_type'] = 'cookie';
$cfg['Servers'][$i]['user'] = 'root';
$cfg['Servers'][$i]['password'] = '';
$cfg['Servers'][$i]['extension'] = 'mysqli';
$cfg['Servers'][$i]['AllowNoPassword'] = true;
$cfg['Lang'] = '';

```

Ingrese el usuario root y la contraseña asignada



4. Crear base de datos



Ingrese en el campo Nombre de la base de datos el nombre cpb cambiar el tipo de cotejamiento **utf32_unicode_520_ci** para que acepte acentos y ñ.



5. Copiar proyecto

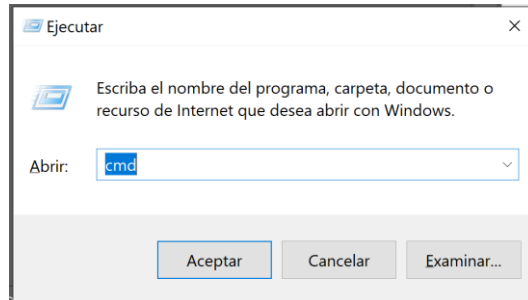
Copiar el proyecto en la ruta donde se instalo el xampp (D:\xampp\htdocs) copiar el proyecto en la ruta especificada

Configurar el archivo .env ingresar los datos de la base de datos username, password, database, host y port

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=cpb
DB_USERNAME=root
DB_PASSWORD=
```

Solo cambiar database y password por los datos configurados de el nombre de la base de datos y contraseña

Presionar inicio + r



Escribir cmd y aceptar

Escribir cd / para retroceder a la raíz si esta instalado en la unidad principal caso contrario ingresar la letra de la unidad mas : ejemplo D:

```
C:\Users\Daniel Rada>cd /  
C:\>D:  
D:\>
```

Ingresar a la ruta cd xampp\htdocs\prestamos_cpb

```
\>cd xampp\htdocs\prestamos_cpb  
\xampp\htdocs\prestamos_cpb>
```

Iniciar la migración de la base de datos con el comando

Php artisan migrate

Php artisan db:seed

Abra su navegador y escriba la siguiente dirección
http://localhost/prestamos_cpb/public

ANEXO G. AVALES

- Aval de aceptación
- Aval de conclusión de proyecto
- Aval especialista
- Aval revisor
- Aval metodológico