



**FACULTAD DE INGENIERIA EN CIENCIAS DE
LA COMPUTACIÓN Y TELECOMUNICACIONES**

UNIVERSIDAD AUTONOMA GABRIEL RENE MORENO

**Software Web y Móvil con Inteligencia Artificial
para el Scoring Crediticio Predictivo y la Evaluación
de Solvencia en Entidades Financieras**



GRUPO 10

INTEGRANTES:

BANEGAS ROCA LUIS FERNANDO	221043780
FLORES QUISPE NATHALY MADELINE	218020139
MELGAR ARTEAGA ALEJANDRO	220154473
MENDOZA RODRIGUEZ LUIS PABLO	219033463
MOLINA CORTEZ MAIKOL ANTHONY	222056118
PEREZ GOMEZ DANIELA	221180575

MATERIA: SISTEMAS DE INFORMACION (INF412 - SA)

DOCENTE: GARZON CUELLAR ANGELICA

Tabla de contenido

PERFIL	1
CAPITULO 1 MARCO TEORICO	20
1.1) Marco Referencial.....	20
Fundamentos	20
1.1.2 Generalidades.....	20
1.1.3 Características	21
1.1.4 Conceptos Específicos	21
1.2) Marco de trabajo ágil SCRUM.....	22
1.2.1 Metodologías ágiles	22
1.2.2 Scrum	22
1.2.3 Equipo Scrum (Scrum Team)	23
1.2.4 Eventos de Scrum	24
1.2.5 Revisión de Sprint.....	25
1.2.6 Artefactos de Scrum.....	26
CAPITULO 2 Herramientas tecnológicas para el desarrollo	28
2.1) Lenguaje de Programación.....	28
2.2) Frameworks y entornos de ejecución	31
Django.....	31
2.3) Sistema de gestor de base de datos.....	34
2.4) Lenguaje de modelado de software (Modelo C4)	35
2.5) Herramienta de diseño y modelado (Enterprise Architect)	37
2.6) Entorno de Desarrollo	39
2.7) Infraestructura de software.....	41
2.7.1 Servicio en la nube (Cloud Computing)	41
2.7.2 SAAS (Software as a Service, SaaS)	43
2.8) Herramientas Colaborativas para Seguimientos de Proyectos	45
2.9) Sistema de control de Versiones de código (Git)	47
2.10) Herramientas de Gestión de Código en la Nube (Bitbucket)	49
CAPITULO 3 REQUERIMIENTOS	1
3.1) Propósito	1
3.2) Ámbito de Sistema.....	2
3.3) Equipo SCRUM.....	4
3.4) Definiciones, Acrónimos y Abreviaturas.....	6
3.5) Funciones del Producto.....	7

3.6)	Product Backlog.....	10
3.7)	Requisitos Funcionales	13
3.8)	Requisitos no funcionales	14
3.9)	Lista de casos de uso.....	17
3.10)	Paquetes y casos de Uso.....	19
3.11)	Planificación Sprint.....	22
CAPITULO 4 PROCESO DE DESARROLLO DE SOFTWARE.....		25
SPRINT 1.....		25
1.	Sprint Planning.....	25
1.1.	Objetivos del Sprint	25
1.2.	Historias de usuario (tarjetas 3C, Planning Póker, prototipos).....	26
1.3.	Contexto del sistema.....	40
1.4.	Sprint backlog	41
1.5.	Equipo SCRUM.....	42
2.	Proceso/patrón de desarrollo por Historia de Usuario	43
2.1.	Diseño	43
2.1.1.	Diseñar de la arquitectura.....	43
Diagrama de Despliegue		43
Diagrama Organizado por Capas		44
2.1.2.	Diseño de datos	44
Mapeo		45
Script.....		45
2.1.3.	Diseño de la lógica de negocio.....	47
2.2.	Implementación.....	50
2.2.1.	Componentes y artefactos generados	50
2.3.	Pruebas.....	51
2.3.1.	Plan de pruebas (criterios de aceptación).....	51
2.3.2.	Reporte de prueba.....	55
3.	Daily Scrum (o Scrum diario).....	59
4.	Sprint Review.....	65
5.	Sprint Retrospective.....	66
6.	Burndown y BurnUp (Grafica de tareas y Datos de tareas).....	67
7.	Grafica de esfuerzo y Datos de esfuerzo.....	68
8.	Scrum TaskBoard (Backlog, to do, doing, done).....	68
SPRINT 2.....		69

1	Sprint Planning.....	69
1.1	Objetivos del Sprint.....	69
1.2	Historias de usuario (tarjetas 3C, Planning Póker, prototipos)	71
1.3	Contexto del sistema	84
1.4	Sprint backlog	85
1.5	Equipo SCRUM	86
1.6	UML 2.5+.....	87
1.6.1	Diagrama de Navegación	87
1.6.2	Diagrama de tiempo	88
1.6.3	Diagrama de estado	90
1.6.4	Características generales	91
2	Proceso/patrón de desarrollo por Historia de Usuario	92
2.1	Diseño.....	92
2.1.1	Diseñar de la arquitectura C4.....	92
	Nivel 1: Context.....	92
	Nivel 2: Contenedor.....	93
	Nivel 3: Diagrama de Componente.....	94
	Nivel 4: Código.....	94
2.2	Diseño de datos	95
	Mapeo	95
2.2.1	Diseño de la lógica de negocio	100
2.3	Implementación.....	110
2.3.1	Componentes y artefactos generados.....	110
2.3.2	Implementación de la arquitectura del sistema	110
2.3.3	Implementación de la arquitectura del subsistema	111
2.4	Pruebas	114
2.4.1	Plan de pruebas (criterios de aceptación).....	114
2.4.2	Reporte de prueba	119
3	Daily Scrum (o Scrum diario).....	127
4	Sprint Review.....	140
5	Sprint Retrospective.....	142
6	Burndown y BurnUp (Gráfica de tareas y Datos de tareas).....	144
7	Gráfica de esfuerzo y Datos de esfuerzo.....	145
8	Scrum TaskBoard (Backlog, to do, doing, done).....	145
	Sprint 3.....	145

Sprint 4.....	145
Bibliografía	146
Anexos	149
Caso de Estudio 1 – Banco Prodem de Bolivia	149
Caso de Estudio 2 – Banco Economico	154
Caso de estudio 3 - Banco BCB.....	158

PERFIL

1) Introducción

El sistema financiero boliviano experimenta una transformación digital acelerada, impulsada por la necesidad de ampliar la inclusión financiera y optimizar la gestión de riesgos. Sin embargo, persiste un desafío crítico: la evaluación de solvencia en segmentos poblacionales con ingresos no formales, como emprendedores, comerciantes y trabajadores independientes. Estos grupos, aunque económicamente activos, frecuentemente quedan excluidos de los créditos tradicionales debido a la falta de historial crediticio formal o aportes a sistemas de pensiones (AFP).

Este proyecto propone el desarrollo de un **sistema de información inteligente** (web y móvil) que integra **inteligencia artificial (IA)** en el workflow de evaluación crediticia. El sistema no solo automatiza procesos, sino que también incorpora el análisis de datos alternativos (transacciones comerciales, facturación electrónica, flujos de caja) para generar una evaluación integral de solvencia. Al hacerlo, se democratiza el acceso al crédito, se reducen tiempos de respuesta y se fortalece la capacidad de las entidades financieras para operar en un entorno competitivo y regulado.

2) Antecedentes

a) Fundamentación teórica

La evaluación crediticia ha evolucionado desde modelos basados exclusivamente en historial crediticio y garantías tangibles hacia enfoques que incorporan **análisis conductual y predictivo**. La teoría de la inclusión financiera sostiene que el acceso a servicios crediticios es un motor de desarrollo económico, pero requiere mecanismos adaptados a realidades socioeconómicas diversas.

La **inteligencia artificial (IA)** y el **machine learning (ML)** representan un paradigma teórico-tecnológico clave. Los algoritmos de ML permiten identificar patrones complejos en grandes volúmenes de datos, tanto estructurados (estados de cuenta, aportes AFP) como no estructurados (historial de transacciones informales, actividad en redes de comercio). La **IA explicativa (XAI)** agrega una capa de transparencia al generar razones interpretables para cada decisión, lo que es crucial para cumplir con principios de equidad y normativas financieras.

En Bolivia, el marco regulatorio de la **ASFI** y el **BCB** ha incorporado directrices para el uso de tecnologías financieras (Fintech), incentivando la innovación bajo principios de seguridad, interoperabilidad y protección al consumidor. Esto crea un sustento legal para sistemas que utilicen IA en la evaluación crediticia, siempre que se garantice la trazabilidad y auditabilidad de las decisiones.

b) Sistemas similares

A nivel global, existen plataformas como **Kabbage** (EE.UU.) y **Nubank** (Brasil) que utilizan algoritmos de scoring alternativo para otorgar créditos a pequeñas empresas y personas no bancarizadas. Estos sistemas analizan datos como movimientos en cuentas bancarias, facturas de servicios e incluso actividad en redes sociales para predecir capacidad de pago.

En el contexto boliviano, si bien la banca tradicional ha avanzado en digitalización (banca en línea, apps móviles), la integración de IA en el núcleo del proceso de crédito es

incipiente. Algunas entidades han implementado sistemas de automatización de workflows, pero sin un componente robusto de IA para el análisis predictivo de riesgo. La mayoría se limita a la digitalización de formularios y flujos de aprobación manuales asistidos por tecnología, sin una capacidad analítica avanzada.

c) Casos de estudio

1. **Banco Prodem: Plataforma digital para microfinanzas**

Enfoque: Digitalización integral de servicios microfinancieros con foco en inclusión.

***Solución implementada:** Desarrollo de una plataforma web y móvil (ProdemNet) que permite a microempresarios y población rural acceder a productos financieros (cuentas de ahorro, microcréditos), realizar pagos, transferencias y consultar su información.

Tecnología utilizada: Plataforma omnicanal con integraciones a pasarelas de pago QR y billeteras móviles. Su prototipo prioriza la usabilidad y accesibilidad para usuarios con baja alfabetización digital.

Lecciones aprendidas: La inclusión financiera requiere no solo tecnología, sino también educación y canales de atención adaptados. La confianza del usuario se construye con interfaces simples y procesos transparentes.

[Para visualizar el caso de estudio 1 click aquí](#)

2. **Banco Económico: Workflow de aprobación de créditos con IA**

Enfoque: Automatización inteligente del proceso de evaluación y aprobación de créditos.

Solución implementada: Implementación de un sistema de información con un motor de IA que automatiza el flujo de crédito. Los clientes solicitan crédito escaneando un QR o mediante la app, y el sistema valida automáticamente su identidad, historial crediticio (vía ASFI/INFOCRED), antigüedad laboral e ingresos.

Tecnología utilizada: Algoritmos de scoring crediticio para predecir riesgo y detectar inconsistencias. El sistema clasifica las solicitudes en aprobadas, rechazadas o para revisión manual, reduciendo significativamente los tiempos de espera.

Lecciones aprendidas: La automatización con IA es viable en el contexto boliviano, pero requiere integraciones robustas con bases de datos externas (AFP, impuestos). La transparencia en la decisión es clave para la aceptación del cliente.

[Para visualizar el caso de estudio 2 click aquí](#)

3. **Banco BCB: Digitalización de trámites financieros**

Enfoque: Modernización y seguridad en la gestión de solicitudes de crédito institucional.

Solución implementada: Desarrollo de un sistema web para la recepción, evaluación y seguimiento digital de solicitudes de crédito. Incluye carga segura de documentos, firma electrónica con certificados digitales y paneles de seguimiento para clientes y administradores.

Tecnología utilizada: Plataforma con enrutamiento inteligente de solicitudes a oficiales de crédito, integración con sistemas de verificación de identidad y emphasis en la seguridad digital y el cumplimiento normativo.

Lecciones aprendidas: La digitalización de procesos burocráticos exige un fuerte componente de seguridad y validez legal (firma electrónica). La trazabilidad completa del trámite mejora la experiencia del usuario y la eficiencia operativa.

[Para visualizar el caso de estudio 3 click aquí](#)

3) Descripción del problema

El sistema financiero boliviano ha mostrado un crecimiento sostenido en los últimos años, con más de 17 millones de cuentas de depósito y alrededor de 1,9 millones de prestatarios al año 2025. Estos datos evidencian un proceso de inclusión financiera en expansión, aunque acompañado de importantes retos para bancos, cooperativas y entidades microfinancieras. En la práctica, uno de los problemas más relevantes es la evaluación de la solvencia de los clientes que solicitan créditos. La metodología tradicional se enfoca principalmente en trabajadores dependientes con aportes a fondos de pensiones (AFP), dejando fuera a emprendedores, comerciantes y trabajadores independientes. Estos sectores, aunque poseen ingresos estables, enfrentan barreras para acceder a préstamos debido a la ausencia de historial formal. La situación genera exclusión financiera y pérdida de oportunidades económicas.

Otro aspecto crítico son los procesos manuales y burocráticos que caracterizan a muchas instituciones. La solicitud y evaluación de créditos requieren trámites presenciales, duplicación de documentos y tiempos de espera elevados. Esto no solo eleva los costos operativos, sino que también afecta la experiencia del cliente, quien demanda respuestas rápidas y accesibilidad digital.

La mora del sistema financiero, que ronda el 3,2%, representa un riesgo latente. La ausencia de herramientas modernas de análisis predictivo impide anticipar el comportamiento de pago de los clientes. Esto limita la capacidad de gestionar riesgos de manera proactiva y pone en juego la sostenibilidad de la cartera crediticia.

La infraestructura tecnológica constituye otro obstáculo. Muchos sistemas centrales bancarios se basan en plataformas obsoletas que dificultan la integración con nuevas tecnologías. En un contexto en el que la innovación es un factor crítico de competitividad, esta rigidez tecnológica limita la modernización.

Finalmente, la presión regulatoria ha incrementado. Normas recientes de la Autoridad de Supervisión del Sistema Financiero (ASFI) y el Banco Central de Bolivia (BCB) exigen mayor seguridad, trazabilidad e interoperabilidad en operaciones financieras. El marco regulatorio Fintech impulsa además la incorporación de inteligencia artificial y automatización para cumplir con requisitos normativos y de ciberseguridad.

En este contexto, la problemática central es la carencia de un mecanismo moderno, inclusivo y seguro para evaluar la solvencia crediticia de los clientes. Se plantea como solución el desarrollo de un software web y móvil con soporte de inteligencia artificial, que permita analizar tanto información tradicional como datos alternativos (facturación, flujos de caja, historial de transacciones). De este modo, se busca reducir tiempos de respuesta, ampliar la inclusión financiera, optimizar la experiencia del cliente y asegurar la sostenibilidad del sistema financiero.

4) Objetivos

a) General

Desarrollar un sistema de información web y móvil, que permita a entidades financieras evaluar de manera automatizada la solvencia crediticia de clientes, optimizando la gestión de solicitudes y fomentando la inclusión financiera en Bolivia.

b) Específicos

1. Automatizar el workflow de solicitudes de crédito para reducir tiempos de respuesta y costos operativos.

2. Implementar modelos de IA que analicen datos financieros y no financieros, prediciendo la capacidad de pago.

3. Integrar fuentes de información tradicionales (AFP, estados de cuenta) con datos alternativos (facturas electrónicas, flujos de ventas).

4. Diseñar dashboards para oficiales de crédito con indicadores de riesgo y justificación de decisiones.

5. Asegurar el cumplimiento de normativas de ASFI y BCB, incluyendo seguridad digital y trazabilidad.

6. Ofrecer una experiencia digital accesible para clientes mediante aplicaciones web y móviles.

5) Alcance

El proyecto contempla el diseño, desarrollo e implementación de un **software integral** compuesto por una aplicación web y una aplicación móvil, con módulos transversales de seguridad, analítica e integración con inteligencia artificial.

Sistema Web (Oficiales y Administradores)

Gestión de Usuarios y Acceso

- Gestionar usuarios: Registrar, modificar, suspender y eliminar usuarios internos.
- Gestionar roles: Definir y asignar roles (administrador, oficial de crédito, analista de riesgo, auditor).
- Gestionar permisos: Crear y restringir permisos específicos según el rol.
- Autenticación de usuario: Validar credenciales, MFA y políticas de seguridad.
- Auditoría de accesos: Registro de eventos y actividades por usuario.

Gestión de Solicitudes de Crédito

- Registrar solicitudes: Alta de nuevas solicitudes con datos personales y financieros.
- Validar información: Verificación de documentos (cédula, extractos, NIT, facturas).
- Workflow crediticio: Seguimiento del estado (en revisión, en análisis, aprobado, rechazado, desembolsado).
- Evaluación automática con IA: Scoring crediticio en tiempo real.
- Aprobación/Rechazo: Manual, automático o mixto (IA + oficial).

Gestión de Productos Financieros

- Configuración de tipos de crédito: microcrédito, vivienda, PyME, consumo.
- Definir tasas de interés y condiciones.
- Reglas de negocio configurables: límites de monto según solvencia.

2. Gestión de Pagos y Cobros

- Generación de plan de pagos.
- Registro de pagos realizados (manual o automático).
- Integración con pasarelas de pago y sistemas QR interoperables.
- Gestión de cuotas atrasadas y alertas de mora.

Gestión de Reportes y Cumplimiento Regulatorio

- Reportes para ASFI y BCB (exportables en formatos oficiales).
- Indicadores de cartera: mora, cobertura, crecimiento.
- Informes de inclusión financiera (segmentos atendidos, prestatarios nuevos).
- Historial de auditoría de operaciones.

Análisis de Datos y Dashboards

- Panel de control para oficiales de crédito.
- Visualización de métricas de riesgo en tiempo real.
- Segmentación de clientes según scoring y comportamiento.
- Proyecciones de mora y predicciones de solvencia usando IA.

Aplicación Móvil (Clientes)

Gestión de Registro y Acceso

- Registro digital con verificación KYC (cédula, foto, biometría).
- Autenticación segura con MFA.(**Autenticación Multifactor.**)
- Perfil de usuario con datos básicos y financieros.

Gestión de Créditos

- Solicitud de créditos: envío de formularios y documentos desde la app.
- Simulador de préstamos: cálculo de cuotas y plazos.
- Seguimiento de estado: notificación de revisión, aprobación o rechazo.
- Visualización de plan de pagos.

2. Pagos y Transacciones

- Pago de cuotas desde la app (tarjeta, QR interoperable, transferencia).
- Registro de pagos realizados y pendientes.
- Alertas de vencimiento de cuotas.

3. Atención al Cliente y Reclamos

- Registro de reclamos o consultas.
- Seguimiento del estado de un reclamo.

Módulos Transversales (Web y Móvil)

1. Seguridad de la Información

- MFA (Multi-Factor Authentication).
- Cifrado extremo a extremo en datos sensibles.
- Registro de auditoría inviolable.
- Cumplimiento con normativas ASFI y BCB.

2. Integración y APIs

- Conexión con sistemas core bancarios.
- Integración con AFP para verificación de aportes.
- Integración con burós de crédito.
- API pública para interoperabilidad con Fintech.

3. Motor de IA Explicativa (XAI)

- Scoring crediticio con ML.
- Justificación de decisiones para oficiales y clientes.
- Segmentación dinámica de clientes.
- Detección de fraude en solicitudes.

4. Arquitectura del Sistema

- Diseño escalable basado en microservicios.

- Base de datos relacional (MySQL) con soporte para analítica.
- Infraestructura híbrida (cloud + on-premise).
- Sincronización en tiempo real entre web y móvil.

5. Notificaciones y Comunicación

- Alertas de pagos pendientes, vencimientos y novedades.
- Comunicación oficial entre clientes y oficiales.

Notificaciones push en móviles y correo electrónico.

6) Elementos del SIBC

Servidor

- **Procesador: Intel Xeon Gold / AMD EPYC.**
 - Características: +16 núcleos, 3.0 GHz.
 - Justificación: Manejo de procesos simultáneos de IA, bases de datos y microservicios.
- **Memoria RAM: 64 GB DDR4/DDR5 ECC.**
 - Características: Escalable hasta 128 GB.
 - Justificación: Ejecución estable de múltiples hilos, procesos de IA y sistemas de monitoreo en tiempo real.
- **Disco Duro: SSD NVMe 2 TB (RAID 10).**
 - Características: Alta velocidad de lectura/escritura.
 - Justificación: Evita cuellos de botella en operaciones bancarias críticas.
- **Red: Tarjeta de Red 10GbE dual.**
 - Características: Con redundancia.
 - Justificación: Garantiza baja latencia y alta disponibilidad para transacciones financieras.
- **Seguridad Física: HSM (Hardware Security Module).**
 - Características: Módulo criptográfico.
 - Justificación: Protección de claves de cifrado y autenticación de usuarios.

Cliente (Cajeros y Oficinas)

- **Estaciones de Trabajo:**
 - Procesador: Intel Core i5 (10ª gen) o AMD Ryzen 5.
 - RAM: 8–16 GB DDR4.
 - Almacenamiento: SSD 256 GB.
 - Tarjeta Gráfica: Integrada.

- **Dispositivos Móviles (para gestores):**
 - Smartphones / Tablets con Android 12+ o iOS 15+.
 - 6 GB RAM, 128 GB almacenamiento.

Medios de Comunicación

- Switch administrable de 48 puertos (Cisco/HP Aruba) con soporte a VLANs.
- Firewall perimetral dedicado (FortiGate, Palo Alto).
- Red WiFi corporativa con WPA3 Enterprise.
- VPN bancaria para accesos remotos seguros.

Otros dispositivos

- Impresoras multifuncionales seguras.
- UPS (Uninterruptible Power Supply).

2. Software

Servidor

- Sistema Operativo: Ubuntu Server 22.04 LTS / Windows Server 2022.
 - Justificación: Estabilidad, seguridad y soporte a largo plazo.
- Gestor de BD: MySQL.
 - Justificación: Transacciones ACID, replicación y manejo confiable de datos financieros.
- Workflow Engine + IA: Camunda BPM + TensorFlow/Scikit-Learn.
 - Justificación: Automatización de procesos y predicción de riesgos.
- Middleware: Kafka / RabbitMQ.
 - Justificación: Mensajería asíncrona para grandes volúmenes de transacciones.
- Seguridad: Bitdefender GravityZone / Microsoft Defender ATP.
 - Justificación: Protección avanzada contra malware y ciberataques.

Cliente

- Sistema Operativo: Windows 10 Pro.
- Navegadores: Google Chrome, Microsoft Edge.
- Aplicaciones: Outlook/Exchange, acceso a intranet bancaria.

Software adicional

- Suite Ofimática: Microsoft Office 365 / LibreOffice.
- Control de Versiones: Git + GitHub/Bitbucket.
- Gestión de Proyectos: Jira / Trello / Confluence.

7) Tecnología

a) Para el desarrollo

- **Desarrollo Web:**

- Lenguaje y Framework: Python con Django.
- Base de datos: MySQL.
- IDE: Visual Studio Code.

- **Aplicación móvil:**

- Framework: Flutter con Dart.
- IDE: Android Studio y Visual Studio Code.

- **Control de versiones y colaboración:**

- Git, GitHub/Bitbucket.
- Jira Software (para Scrum).
- Confluence / Trello.

b) Para la puesta en marcha

- **Servidor en la nube:**

- Opciones: AWS (EC2, RDS, SageMaker), GCP (BigQuery, AutoML), Azure.

- **Base de datos en producción:**

- MySQL Enterprise Edition con replicación y clustering.
- Posible uso de Amazon RDS o Google Cloud SQL.

- **Automatización de despliegue:**

- Docker para contenerización.
- CI/CD: GitHub Actions o Jenkins.

8) Costos para la puesta en marcha

a) Hardware

ITEM	CANTIDAD	COSTO PROM. (en \$)
Laptop	6	550
Computadora de escritorio	1	500
Celulares	6	300
Total		1350

b) Software

ITEM	COSTO (en \$)
StarUML	130
Sistema Operativo Windows10 C/Licencia	150
Total	280

c) Comunicaciones

El costo de las comunicaciones tendrá que incluir:

- **Costo de llamadas realizadas a través de redes de telecomunicaciones;** tanto internas (entre sucursales) como externas (atención al cliente, coordinación con entes reguladores como la ASFI).
- El costo de cualquier equipo adicional de comunicaciones, como líneas telefónicas permanentes, teléfonos portátiles, teléfonos celulares, estaciones base;
- Cualquier instalación adicional requerida para redes móviles.

- El tipo de equipo más adecuado y económico dependerá del contexto operativo de cada sucursal o unidad financiera, priorizando eficiencia.

d) Humanos

Las personas que van a estar incorporadas para el manejo de este sistema son de tipo:

- **Administrador:** El Encargado que se hará cargo de organizar asuntos que se encuentran bajo su responsabilidad, así como las funciones relacionadas con la gestión diaria, organización de trámites internos (solicitudes de crédito, apertura de cuentas, y seguimiento de operaciones), elaboración y revisión de documentos oficiales (contratos, reportes financieros), preparación de informes entre otros como la elaboración de documentos.
- **Usuario:** Es la persona que usa el servicio.

e) Logística

Un factor a tomar en cuenta son las capacitaciones que se deberá brindar a los empleados de la empresa que desee ocupar este sistema, de esta manera cada empleado podrá sacar provecho a las funcionalidades, y así también se evitara malas prácticas que podrían entorpecer el desarrollo normal de las actividades. Consideremos la siguiente tabla para el calculo de los costos por capacitación al personal:

	Horas requeridas de capacitación	Costo/hora	Total
Cajero	6	80	480
Oficial de crédito	8	100	800
Analista financiero	10	120	1200
Supervisor de operaciones	5	150	750

9) Beneficios para el cliente

a) Tiempo

- Mayor agilidad en el procesamiento de solicitudes financieras, como créditos, aperturas de cuentas y trámites administrativos.
- Facilidad para acceder a información específica de productos financieros, historial de transacciones o estado de solicitudes.
- Acceso rápido y seguro a servicios financieros desde cualquier lugar, mediante plataformas digitales y banca móvil.

b) Esfuerzo humano

- Automatización de procesos operativos como validación de documentos, gestión de expedientes y seguimiento de casos, reduciendo la carga manual.
- Incremento en la productividad del personal financiero gracias a sistemas integrados que permiten una atención más eficiente y personalizada.
- Reducción de errores humanos al eliminar el uso de formularios físicos y sistemas manuales, mejorando la precisión en la gestión de datos.

c) Costos

- Disminución de costos operativos mediante la digitalización de procesos, lo que reduce la necesidad de infraestructura física para almacenamiento de documentos.
- Optimización de recursos tecnológicos y humanos, permitiendo que las entidades financieras ofrezcan servicios más competitivos y accesibles para el cliente.

CAPITULO 1 MARCO TEORICO

1.1) Marco Referencial

El crédito constituye un instrumento fundamental de inclusión financiera. En Bolivia, la cartera de créditos superó los 226 mil millones de bolivianos en 2025, distribuidos principalmente en microcrédito, vivienda y empresarial. El desafío radica en equilibrar accesibilidad con sostenibilidad, evitando el sobreendeudamiento y garantizando la recuperación de la cartera.

La inteligencia artificial se posiciona como herramienta estratégica para transformar este proceso. Modelos de machine learning permiten identificar patrones de riesgo, segmentar clientes y predecir su comportamiento de pago. El uso de IA explicativa facilita, además, la transparencia al ofrecer justificaciones claras sobre cada decisión tomada.

La normativa local, impulsada por la ASFI y el BCB, establece requisitos de seguridad, interoperabilidad y control que obligan a las entidades a incorporar soluciones digitales confiables. La reciente regulación Fintech en Bolivia abre espacio para que tecnologías emergentes como la IA sean aplicadas en la evaluación crediticia, favoreciendo la inclusión de sectores históricamente marginados.

Fundamentos

La digitalización de procesos financieros responde a la necesidad de optimizar recursos, reducir tiempos y aumentar la precisión en la toma de decisiones. La incorporación de IA en la evaluación crediticia constituye un fundamento técnico y estratégico, ya que permite analizar grandes volúmenes de información con rapidez y fiabilidad.

1.1.2 Generalidades

El sistema propuesto considera a tres actores principales: clientes (usuarios finales), oficiales de crédito (usuarios internos) y reguladores (ASFI, BCB). La plataforma busca satisfacer las necesidades de cada actor: accesibilidad y rapidez para clientes, eficiencia operativa para oficiales, y cumplimiento normativo para reguladores. Asimismo, se

inspira en tendencias globales que apuntan hacia la banca digital y la inclusión de clientes no bancarizados.

1.1.3 Características

Entre las principales características del sistema destacan:

- Organización centralizada de datos.
- Automatización de workflows crediticios.
- Seguridad de la información (confidencialidad, integridad y disponibilidad).
- Análisis predictivo mediante IA.
- Inclusión financiera a través de la consideración de ingresos no formales.
- Transparencia mediante IA explicativa.

1.1.4 Conceptos Específicos

Workflow: flujo estructurado de actividades desde la solicitud hasta el desembolso de un crédito.

Scoring crediticio: puntaje generado por un modelo predictivo que mide la capacidad de pago.

KYC (Know Your Customer): proceso de verificación de identidad de clientes.

IA explicativa (XAI): capacidad de un modelo de IA de ofrecer razones claras sobre sus decisiones.

1.2) Marco de trabajo ágil SCRUM

1.2.1 Metodologías ágiles

Las metodologías ágiles son un conjunto de principios y prácticas que se utilizan en el desarrollo de software para entregar productos de alta calidad de manera rápida y flexible.

Estas metodologías se basan en la idea de que el trabajo en proyectos de software es más efectivo cuando se realiza de manera incremental y colaborativa, en lugar de tratar de planificar y especificar todos los detalles al principio del proyecto. Así, buscan centrarse en la entrega continua de pequeñas porciones de software funcional, lo que permite a los equipos adaptarse y responder rápidamente a los cambios en los requisitos y en el entorno.

Algunas de las metodologías ágiles más comunes son: Scrum, Lean, XP y Kanban

1.2.2 Scrum

Scrum es un marco de trabajo en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos.

En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales

1.2.3 Equipo Scrum (Scrum Team)

Los Equipos Scrum son autoorganizados y multifuncionales. Los equipos autoorganizados eligen la mejor forma de llevar a cabo su trabajo y no son dirigidos por personas externas al equipo. Los equipos multifuncionales tienen todas las competencias necesarias para llevar a cabo el trabajo sin depender de otras personas que no son parte del equipo.

El modelo de equipo en Scrum está diseñado para optimizar la flexibilidad, la creatividad y la productividad. El equipo Scrum está conformado por: un Product Owner, el Equipo de Desarrollo y un Scrum Master. (Schwaber, K. & Sutherland. J., 2016).

Scrum Master

El Scrum Master es el responsable de asegurar que Scrum se entienda y se adopte. Los Scrum Masters hacen esto asegurándose de que el Equipo Scrum trabaja ajustándose a la teoría, prácticas y reglas de Scrum.

El Scrum Master es un líder que está al servicio del Equipo Scrum. El Scrum Master ayuda a las personas externas al Equipo Scrum a entender qué interacciones con el Equipo Scrum pueden ser útiles y cuáles no. El Scrum Master ayuda a todos a modificar estas interacciones para maximizar el valor creado por el Equipo Scrum.

Dueño de Producto (Product Owner)

El Dueño de Producto es el responsable de maximizar el valor del producto y el trabajo del Equipo de Desarrollo. El cómo se lleva a cabo esto podría variar ampliamente entre distintas organizaciones, Equipos Scrum e individuos. El Dueño de Producto es una única persona, no un comité. y es la única persona responsable de gestionar la Lista del Producto (Product Backlog).

Equipo de desarrollo (Development Team)

El Equipo de Desarrollo consiste en los profesionales que realizan el trabajo de

entregar un incremento de producto “Terminado” que potencialmente se pueda poner en producción al final de cada Sprint. Solo los miembros del Equipo de Desarrollo participan en la creación del Incremento.

Los Equipos de Desarrollo tienen las siguientes características:

- Son autoorganizados.
- Son multifuncionales.

El tamaño óptimo del Equipo de Desarrollo es lo suficientemente pequeño como para permanecer ágil y lo suficientemente grande como para completar una cantidad de trabajo significativa

1.2.4 Eventos de Scrum

En Scrum existen eventos predefinidos con el fin de crear regularidad y minimizar la necesidad de reuniones no definidas en Scrum. Todos los eventos son bloques de tiempo (time-boxes), de tal modo que todos tienen una duración máxima. Una vez que comienza un Sprint, su duración es fija y no puede acortarse o alargarse.

Cada uno de los eventos de Scrum constituye una oportunidad formal para la inspección y adaptación de algún aspectSprint

El corazón de Scrum es el Sprint, es un bloque de tiempo (time-box) de un mes o menos durante el cual se crea un incremento de producto “Terminado” utilizable y potencialmente desplegable.

Es más conveniente si la duración de los Sprints es consistente a lo largo del esfuerzo de desarrollo. Cada nuevo Sprint comienza inmediatamente después de la finalización del Sprint anterior. Cada Sprint puede considerarse un proyecto con un horizonte no mayor de un mes

Planificación del Sprint (Sprint Planning)

El trabajo a realizar durante el Sprint se planifica en la Planificación de Sprint. Este

plan se crea mediante el trabajo colaborativo del Equipo Scrum completo.

La Planificación de Sprint tiene un máximo de duración de ocho horas para un Sprint de un mes. Para Sprints más cortos el evento es usualmente más corto. El Scrum Master se asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito. El Scrum Master enseña al Equipo Scrum a mantenerse dentro del bloque de tiempo

Objetivo del Sprint (Sprint Goal)

El Objetivo del Sprint es una meta establecida para el Sprint que puede lograrse mediante la implementación de la Lista de Producto. Proporciona una guía al Equipo de Desarrollo acerca de por qué está construyendo el incremento. Además, brinda al equipo de desarrollo cierta flexibilidad con respecto a la funcionalidad implementada en el Sprint.

Scrum Diario (Daily Scrum)

El Scrum Diario es una reunión con un bloque de tiempo de 15 minutos para que el Equipo de Desarrollo sincronice sus actividades y cree un plan para las siguientes 24 horas. Esto se lleva a cabo inspeccionando el trabajo avanzado desde el último Scrum Diario y haciendo una proyección acerca del trabajo que podría completarse antes del siguiente.

El Equipo de Desarrollo usa el Scrum Diario para evaluar el progreso hacia el Objetivo del Sprint y para evaluar qué tendencia sigue este progreso hacia la finalización del trabajo contenido en la Lista de Pendientes del Sprint.

1.2.5 Revisión de Sprint

Al final del Sprint se lleva a cabo una Revisión de Sprint para inspeccionar el Incremento y adaptar la Lista de Producto si fuese necesario. Durante la Revisión de Sprint, el Equipo Scrum y los interesados colaboran acerca de lo que se hizo durante el Sprint. Basándose en esto y en cualquier cambio a la Lista de Producto

durante el Sprint, los asistentes colaboran para determinar las siguientes cosas que podrían hacerse para optimizar el valor

Retrospectiva de Sprint

La Retrospectiva de Sprint es una oportunidad para el Equipo Scrum de inspeccionarse a sí mismo y de crear un plan de mejoras que sean abordadas durante el siguiente Sprint.

La Retrospectiva de Sprint tiene lugar después de la Revisión de Sprint y antes de la siguiente Planificación de Sprint.

El propósito de la Retrospectiva de Sprint es:

- Inspeccionar cómo fue el último Sprint en cuanto a personas, relaciones, procesos y herramientas;
- Identificar y ordenar los elementos más importantes que salieron bien y las posibles mejoras; y,
- Crear un plan para implementar las mejoras a la forma en la que el Equipo Scrum desempeña su trabajo.

1.2.6 Artefactos de Scrum

Los artefactos de Scrum representan trabajo o valor en diversas formas que son útiles para proporcionar transparencia y oportunidades para la inspección y adaptación. Los artefactos definidos por Scrum están diseñados específicamente para maximizar la transparencia de la información clave, necesaria para asegurar que todos tengan el mismo entendimiento del artefacto.

Lista de Producto (Product Backlog)

La Lista de Producto es una lista ordenada de todo lo que podría ser necesario en el producto y es la única fuente de requisitos para cualquier cambio a realizarse en el producto. El Dueño de Producto (Product Owner) es el responsable de la Lista de Producto, incluyendo su contenido, disponibilidad y ordenación.

La Lista de Producto enumera todas las características, funcionalidades, requisitos, mejoras y correcciones que constituyen cambios a realizarse sobre el producto para

entregas futuras. Los elementos de la Lista de Producto tienen como atributos la descripción, el orden, la estimación y el valor.

Lista de Pendientes del Sprint (Sprint Backlog)

La Lista de Pendientes del Sprint es el conjunto de elementos de la Lista de Productos seleccionados para el Sprint, más un plan para entregar el Incremento de producto y conseguir el Objetivo del Sprint. La Lista de Pendientes del Sprint es una predicción hecha por el Equipo de Desarrollo acerca de qué funcionalidad formará parte del próximo Incremento y del trabajo necesario para entregar esa funcionalidad en un Incremento “Terminado”. La Lista de Pendientes del Sprint hace visible todo el trabajo que el Equipo de Desarrollo identifica como necesario para alcanzar el Objetivo del Sprint

Incremento

El Incremento es la suma de todos los elementos de la Lista de Producto completados durante un Sprint y el valor de los incrementos de todos los Sprints anteriores. Al final de un Sprint el nuevo Incremento debe estar “Terminado”, lo cual significa que está en condiciones de ser utilizado y que cumple la Definición de “Terminado” del Equipo Scrum. El incremento debe estar en condiciones de utilizarse sin importar si el Dueño de Producto decide liberarlo o no.

CAPITULO 2 Herramientas tecnológicas para el desarrollo

2.1) Lenguaje de Programación

WEB: PYTHON



Python es un lenguaje de programación interpretado, de alto nivel y propósito general, ampliamente utilizado en el desarrollo web gracias a su sintaxis clara y su gran ecosistema de frameworks. Su versatilidad permite crear desde sitios web sencillos hasta aplicaciones empresariales complejas.

Uno de los frameworks más destacados para desarrollo web con Python es Django, que facilita la creación rápida de aplicaciones seguras y escalables.

Características destacadas:

- Sintaxis clara y legible: Ideal para desarrolladores principiantes y expertos, facilita el mantenimiento del código.
- Multiplataforma: Compatible con sistemas operativos como Windows, Linux y macOS.
- Frameworks robustos: Django y Flask permiten desarrollar aplicaciones web de forma rápida y estructurada.
- Gran comunidad: Cuenta con una comunidad activa que ofrece soporte, documentación y miles de bibliotecas.
- Escalabilidad: Python permite construir aplicaciones que crecen fácilmente en funcionalidad y rendimiento.
- Integración sencilla: Se integra fácilmente con bases de datos, APIs y otros lenguajes.

MOVIL: FLUTTERDART



Dart es un lenguaje de programación desarrollado por Google y de código abierto. Su objetivo principal es permitir a los desarrolladores crear aplicaciones web y móviles de forma más eficiente. Desde su lanzamiento en 2011, Dart ha evolucionado significativamente.

Dart viene con un conjunto completo de herramientas integradas, como un gestor de paquetes y un analizador de código, lo que facilita el proceso de desarrollo. Puede ejecutarse de forma instantánea gracias a su máquina virtual y compilación Just-in-Time.

En producción, el código Dart se puede compilar en lenguaje nativo, eliminando la necesidad de entornos especiales. Para desarrollo web, se transpila a JavaScript.

La sintaxis de Dart es similar a otros lenguajes populares como JavaScript o Java, lo que facilita su aprendizaje para quienes ya conocen esos lenguajes.

Características destacadas:

- Elegante y eficiente: Dart se caracteriza por su diseño moderno, limpio y fácil de usar, lo que simplifica el desarrollo de aplicaciones complejas. Tipado estático opcional: Dart permite especificar tipos de datos estáticos para detectar errores y mejorar el rendimiento, aunque esta característica es opcional para brindar flexibilidad en el desarrollo.
- Compilado o interpretado: Dart puede compilarse a código nativo para lograr un mejor rendimiento en aplicaciones móviles y de escritorio, o interpretarse en tiempo de ejecución para aplicaciones web.
- Programación orientada a objetos: Dart sigue el paradigma de programación orientada

a objetos, lo que significa que todo en Dart es un objeto y ofrece características como herencia, polimorfismo y encapsulamiento.

- **Asincronía:** Dart cuenta con soporte nativo para programación asíncrona, lo que facilita la creación de aplicaciones que realizan múltiples tareas simultáneamente, como solicitudes de red o procesamiento de datos en segundo plano.
- **Colecciones:** Dart proporciona una variedad de estructuras de datos para manejar colecciones de elementos, como listas, conjuntos y mapas, simplificando el manejo y manipulación de datos.
- **Soporte para interfaces gráficas de usuario (GUI):** Dart incluye un conjunto de bibliotecas para crear interfaces de usuario dinámicas y ricas en múltiples plataformas.

2.2) Frameworks y entornos de ejecución

DJANGO FRAMEWORK



Django es un framework de desarrollo web de alto nivel basado en el lenguaje de programación Python. Fue diseñado para facilitar la creación rápida de aplicaciones web seguras, escalables y mantenibles. Django sigue el patrón arquitectónico **Modelo-Vista-Controlador (MVC)** adaptado como **Modelo-Vista-Template (MVT)**, lo que permite una separación clara entre la lógica de negocio, la presentación y el manejo de datos.

Características destacadas:

- **Desarrollo Rápido y Eficiente:** Django incluye herramientas integradas como panel de administración, sistema de autenticación y manejo de formularios, lo que permite construir aplicaciones completas en menos tiempo.
- **Seguridad Incorporada:** El framework protege automáticamente contra vulnerabilidades comunes como inyección SQL, falsificación de solicitudes (CSRF), y secuestro de sesiones, garantizando aplicaciones más seguras.
- **Escalabilidad y Rendimiento:** Django está preparado para manejar aplicaciones de alto tráfico y grandes volúmenes de datos, siendo utilizado por empresas como Instagram y Pinterest.
- **Sistema de ORM (Object-Relational Mapping):** Permite interactuar con bases de datos mediante objetos Python, sin necesidad de escribir consultas SQL directamente, lo que simplifica el desarrollo y mantenimiento.
- **Estructura Modular y Reutilizable:** Las aplicaciones Django se organizan en módulos reutilizables, lo que facilita la colaboración en equipo y la expansión del proyecto.

- **Amplia Comunidad y Documentación:** Django cuenta con una comunidad activa y extensa documentación oficial, lo que facilita el aprendizaje, la resolución de problemas y el acceso a recursos adicionales.

FLUTTER FRAMEWORK



Flutter es un SDK desarrollado por Google para la creación de aplicaciones móviles y web utilizando el lenguaje de programación Dart. Con Flutter, puedes desarrollar aplicaciones móviles para Android e iOS utilizando un solo conjunto de código fuente.

Características clave de Flutter:

- **Desarrollo Ágil:** Flutter ofrece características como Hot Reload, que permite realizar cambios en el código y ver las actualizaciones de manera instantánea en la aplicación en desarrollo. Esto acelera el proceso de desarrollo y mejora la productividad del equipo.
- **Aplicaciones Nativas con un Solo Código:** A diferencia de otros frameworks donde una gran parte del código se comparte, en Flutter el 100% del código funciona para ambas plataformas (Android e iOS). No es necesario escribir código personalizado para cada plataforma, ya que Flutter compila a aplicaciones nativas, lo que garantiza un rendimiento óptimo y una experiencia de usuario consistente en todas las plataformas.
- **Alto Rendimiento:** Las aplicaciones desarrolladas con Flutter funcionan a 60fps (cuadros por segundo), lo que ofrece una experiencia de usuario fluida y altamente receptiva. Esto se traduce en animaciones suaves, actualizaciones rápidas de la interfaz de usuario y un rendimiento generalmente superior.
- **Componentes de Interfaz Gráfica Personalizables:** Flutter proporciona una amplia variedad de widgets y componentes de interfaz gráfica listos para usar, que se pueden

personalizar según las necesidades del proyecto. Estos widgets respetan los sistemas de diseño tanto de iOS como de Android, incluido Material Design, lo que garantiza que las aplicaciones tengan un aspecto nativo y se integren perfectamente con las plataformas respectivas desde el principio.

2.3) Sistema de gestor de base de datos

POSTGRESQL



PostgreSQL es un sistema de gestión de bases de datos relacional orientado a objetos de código abierto con una amplia gama de características avanzadas. Aquí están algunas de sus **características principales:**

- **Transacciones ACID:** PostgreSQL garantiza la integridad de los datos al cumplir con las propiedades ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad) en todas las operaciones de la base de datos. Esto asegura que las transacciones sean confiables y se mantenga la consistencia de los datos.
- **Tipos de Datos Extensibles:** Además de admitir los tipos de datos estándar como enteros, cadenas de texto y booleanos, PostgreSQL también soporta una amplia variedad de tipos de datos especializados. Esto incluye tipos de datos para fechas, valores monetarios, direcciones IP, elementos gráficos, y permite la creación de tipos de datos personalizados para adaptarse a las necesidades específicas de la aplicación.
- **Herencia entre Tablas:** PostgreSQL es conocido por su capacidad de herencia entre tablas, lo que significa que una tabla puede heredar propiedades y métodos de otra. Esta funcionalidad es útil para modelar relaciones entre conjuntos de datos de manera eficiente y para implementar esquemas de herencia en la base de datos, lo que lo convierte en un sistema de gestión de bases de datos objeto-relacional.

2.4) Lenguaje de modelado de software (Modelo C4)

El **Modelo C4** es una técnica de notación ligera utilizada para **visualizar la arquitectura de software** en distintos niveles de detalle. Su nombre proviene de las **4 vistas** principales que lo componen:

1. Nivel de Contexto

- Proporciona una visión general del sistema y sus interacciones externas.
- Identifica los sistemas externos con los que se comunica nuestro software.
- Define los actores (usuarios) que interactúan con el sistema.
- En este nivel, el sistema se modela como una “caja negra” que solo expone sus relaciones externas.
- En resumen: Muestra el sistema completo como una “caja negra” y cómo interactúa con usuarios externos y otros sistemas.

2. Nivel de Contenedores

- Divide el sistema en grandes bloques ejecutables o de almacenamiento (ej: aplicaciones web, móvil, backend, base de datos).
- En nuestro caso:
 - **Web app** (portal para clientes y oficiales de crédito).
 - **Mobile app** (clientes solicitan créditos desde celular).
 - **Backend/API** (motor de workflow + IA).
 - **Base de datos PostgreSQL** (almacén de solicitudes y expedientes digitales).

3. Nivel de Componentes

Detalla cada contenedor en **módulos internos** que cumplen funciones específicas.

Ejemplo de nuestro sistema:

En el backend:

- **Módulo de autenticación y seguridad.**

- **Módulo de workflow de créditos.**
- **Módulo de IA para scoring crediticio.**
- **Módulo de notificaciones.**

4. Nivel de Código

Representa la implementación a nivel de clases, librerías o microservicios.

Aquí se pueden usar diagramas UML o de clases para modelar cómo se programa cada componente.

Ejemplo: clases para Cliente, SolicitudCredito, EvaluadorIA.

Ejemplo aplicado a nuestro proyecto

- **Contexto:**

Un **cliente independiente** solicita un crédito vía **app móvil**. El sistema valida sus datos con AFP/ASFI, ejecuta el **scoring IA** y genera una decisión.
- **Contenedores:**
 - **Web App (React/Angular):** interfaz para oficiales de crédito y admin.
 - **Mobile App (Flutter):** interfaz para clientes.
 - **API Backend (Node/NestJS):** gestiona workflow, IA y reglas de negocio.
 - **DB PostgreSQL:** guarda expedientes y métricas.
- **Componentes:**
 - **Gestión de solicitudes**
 - **IA Scoring Crediticio**
 - **Notificaciones (SMS/email)**
 - **Gestión documental (expedientes digitales)**
- **Código:**
 - Clases/entidades: Cliente, Solicitud, EvaluacionIA, OficialCredito.

2.5) Herramienta de diseño y modelado (Enterprise Architect)

Enterprise Architect (EA) es una herramienta de modelado y diseño ampliamente utilizada en la ingeniería de software para representar gráficamente **arquitecturas, procesos, requisitos y sistemas**.

Permite trabajar con diferentes lenguajes de modelado y metodologías, entre ellos:

- **UML (Unified Modeling Language):** para casos de uso, clases, secuencias, actividades.
- **BPMN (Business Process Model and Notation):** para modelar procesos de negocio.
- **Modelo C4:** para describir la arquitectura de software en distintos niveles.
- **SysML (Systems Modeling Language):** para sistemas más complejos.

Aplicación a nuestro proyecto

En el desarrollo de “Software Web Y Móvil Para La Gestión De Solicitudes De Crédito, Evaluación De Solvencia En Entidades Financieras”, Enterprise Architect será útil para:

1. Modelado de requisitos y casos de uso

Diagramas de casos de uso para mostrar cómo los clientes, oficiales de crédito y administradores interactúan con el sistema.

2. Modelado de procesos financieros (workflow de aprobación de créditos)

Diagramas BPMN que representen el flujo de:

- Solicitud de crédito.
- Validación de requisitos.
- Evaluación con IA.
- Aprobación/rechazo/notificación.

3. Modelado de la arquitectura (complementando al Modelo C4)

- Diagramas de contenedores y componentes que muestren cómo se conectan la app web, app móvil, backend de IA y base de datos.

4. Gestión de documentación

- EA permite mantener una documentación integrada de todos los modelos y diagramas, lo cual ayuda a la trazabilidad de los requisitos hasta la implementación.

Ventajas de usar Enterprise Architect

- Facilita la **comunicación visual** dentro del equipo de desarrollo.
- Integra **diversos tipos de diagramas** en una sola plataforma.
- Aporta a la **documentación formal** que se requiere en la materia.
- Permite la **trazabilidad** entre requisitos → diseño → implementación.

Conclusión:

En nuestro proyecto, **Enterprise Architect** se utilizará para **modelar los procesos financieros y la arquitectura del sistema**, generando diagramas de casos de uso, BPMN y UML que representen claramente cómo funciona el **workflow de aprobación de créditos con IA**.

2.6) Entorno de Desarrollo

El **entorno de desarrollo** es el conjunto de herramientas y aplicaciones que utilizarán los programadores para **escribir, probar y depurar** el código del sistema. Elegir un entorno adecuado garantiza productividad, organización del proyecto y mejor integración del equipo.

Entorno propuesto para nuestro proyecto (Grupo #10)

1. Editor de Código – Visual Studio Code (VS Code)

- Multiplataforma (Windows, Linux, MacOS).
- Ligero, rápido y con gran soporte de extensiones.
- Extensiones útiles: integración con GitHub/Bitbucket, depuradores para python y PostgreSQL, conectores para Docker/AWS.

2. Entornos de desarrollo para móviles

- **Android Studio** (para pruebas en Android).
- **Xcode (MacOS)** en caso de despliegues iOS.
- Aunque se use **Flutter**, es necesario configurar entornos nativos para emulación y publicación en Play Store/App Store.

3. Entorno Backend / API

Framework: Django + Django REST Framework (DRF)

Paquetes clave (sugeridos):

- Autenticación/Seguridad: django-allauth (social/OTP opcional), django-axes (anti-fuerza bruta), django-otp (2FA).
- Workflow/Estados: django-fsm (máquina de estados) o modelado de estados propio.
- Documentación API: drf-spectacular o drf-yasg (OpenAPI/Swagger).
- CORS: django-cors-headers (para web/app móvil).
- Almacenamiento documental: django-storages (S3/MinIO).

- Auditoría: django-simple-history (bitácoras de cambios).

4. Entorno de Base de Datos

PostgreSQL como gestor principal.

Uso de **PgAdmin 4** o **DBeaver** como clientes gráficos para consultas, pruebas y administración de datos.

5. Contenedores y virtualización

- **Docker** para levantar entornos aislados y reproducibles del backend y la base de datos.
- Facilita el despliegue en la nube (AWS o similares).

6. Flujo de trabajo del equipo

- **Git** (ramas: main, develop, feature/*, hotfix/*).
- **Pull Requests** con revisión por pares.
- **Tablero Scrum** en Jira/Trello (sprints, HU, definición de terminado).
- **Convenciones:** PEP8, black, mensajes de commit claros (convencionales).

Ventajas del entorno elegido

- **VS Code:** liviano y flexible, ideal para proyectos universitarios.
- **Flutter + Android Studio:** permite desarrollar una sola base de código para apps móviles en Android/iOS.
- **Django/DRF** acelera el backend y estandariza la API. **PostgreSQL:** robusto y usado en sistemas financieros reales.
- **Docker:** asegura portabilidad y facilita despliegue en distintos servidores.

Conclusión

El entorno de desarrollo propuesto combina herramientas modernas, accesibles y ampliamente usadas en la industria, lo que permitirá al equipo trabajar de manera ágil, colaborativa y orientada a buenas prácticas.

2.7) Infraestructura de software

El modelo **IaaS (Infrastructure as a Service)** proporciona recursos de hardware y virtualización desde la nube, como **servidores, almacenamiento y redes**, que permiten desplegar aplicaciones sin necesidad de invertir en infraestructura física.

Para nuestro proyecto, IaaS asegura:

- **Escalabilidad:** aumentar capacidad de cómputo o almacenamiento según el número de clientes y solicitudes de crédito.
- **Alta disponibilidad:** servicios distribuidos geográficamente, reduciendo caídas.

Seguridad: capas de firewall, cifrado de datos en tránsito y en reposo

.

2.7.1 Servicio en la nube (Cloud Computing)

Las principales opciones de nube pública son:

- **Amazon Web Services (AWS)**

Más utilizada en proyectos académicos e industriales.

Servicios clave para nuestro proyecto:

- **EC2:** servidores virtuales para desplegar la API Django/DRF.
- **RDS (PostgreSQL):** base de datos gestionada y segura.
- **S3:** almacenamiento de expedientes digitales de clientes (boletas, contratos, documentos de identidad).
- **CloudWatch:** monitoreo de logs y métricas.
- Recomendación: usar **AWS Free Tier** para pruebas del proyecto.

Google Cloud Platform (GCP)

- Orientado a analítica de datos e IA.

- Podría usarse si se desea integrar modelos de **Machine Learning** en la nube con **Vertex AI**.

Microsoft Azure

- Fuerte en integraciones empresariales y seguridad.
- Alternativa viable si la institución ya usa ecosistema Microsoft.

- **IBM Cloud**

Tiene servicios financieros y regulatorios, aunque menos común en entornos académicos.

2.7.2 SAAS (Software as a Service, SaaS)

SaaS (Software as a Service) es un modelo de distribución en el cual la aplicación está alojada en la nube y los usuarios acceden a ella a través de internet, sin necesidad de instalar nada en sus dispositivos.

Aplicación en nuestro proyecto:

- El “Software Web Y Móvil Para La Gestión De Solicitudes De Crédito, Evaluación De Solvencia En Entidades Financieras” se planteará como **SaaS**, lo que significa que:
 - Los clientes acceden desde **un navegador web o app móvil**.
 - No requieren instalaciones locales.
 - El sistema se actualiza y mejora automáticamente en el servidor.
 - Se ofrece **escalabilidad y disponibilidad 24/7**.

Beneficios del modelo SaaS en nuestro caso:

- **Accesibilidad:** clientes, oficiales de crédito y administradores pueden acceder desde cualquier dispositivo conectado a internet.
- **Reducción de costos:** no requiere infraestructura física local en la entidad financiera.
- **Mantenimiento centralizado:** las actualizaciones de seguridad y mejoras funcionales se aplican en la nube.
- **Seguridad:** datos encriptados y respaldados en la nube.

Conclusión:

Nuestro proyecto se desplegará en la nube utilizando un modelo **IaaS + SaaS**:

- **IaaS:** para contar con servidores virtuales, almacenamiento y bases de datos gestionadas (ej: AWS EC2 + RDS + S3).

SaaS: para ofrecer el sistema a clientes y oficiales de crédito vía web y móvil, garantizando accesibilidad, reducción de costos y mantenimiento centralizado.

2.8) Herramientas Colaborativas para Seguimientos de Proyectos

En el desarrollo de software con equipos distribuidos, es fundamental contar con **herramientas colaborativas** que permitan organizar tareas, dar seguimiento al progreso y documentar el proyecto de manera ágil. Para nuestro sistema, se emplearán principalmente **herramientas basadas en la metodología SCRUM**, que faciliten la planificación de sprints y el control del avance.

1. Jira Software

- Plataforma de **gestión ágil de proyectos** desarrollada por Atlassian.
- Ideal para administrar proyectos bajo **SCRUM o Kanban**.
- **Uso en nuestro proyecto:**
 - Creación de **historias de usuario (HU)** con criterios de aceptación.
 - Definición de **sprints** (planificación y seguimiento).
 - Uso de tableros visuales (*to do, in progress, done*).
 - Generación de reportes de **burndown/burnup**.
- Beneficio: permite **trazabilidad total** desde la HU hasta la entrega final.

2. Trello

- Herramienta ligera y visual basada en **tableros Kanban**.
- **Uso en nuestro proyecto:**
 - Organización de tareas rápidas.
 - Seguimiento de pendientes en cada sprint.
 - Ideal para integrar al equipo en tareas no técnicas (documentación, reuniones, investigación).
- Beneficio: curva de aprendizaje baja y acceso multiplataforma.

3. Confluence

- Plataforma de documentación colaborativa (también de Atlassian).

- **Uso en nuestro proyecto:**
 - Redacción de la **documentación técnica y funcional** del sistema.
 - Registro de acuerdos de reuniones y decisiones de diseño.
 - Vinculación con **Jira** para mantener la trazabilidad.
- Beneficio: centraliza la documentación, evitando dispersión en múltiples archivos.

4. Otros complementos (opcionales)

- **Slack / Microsoft Teams:** para comunicación instantánea entre los miembros del grupo.
- **Google Drive / OneDrive:** para almacenamiento y edición compartida de archivos complementarios.

Conclusión

Para el desarrollo del “Software Web Y Móvil Para La Gestión De Solicitudes De Crédito, Evaluación De Solvencia En Entidades Financieras”, el equipo utilizará **Jira Software** como herramienta principal de gestión ágil, complementado con **Trello** para la organización rápida de tareas y **Confluence** para la documentación centralizada.

Esto asegurará un **seguimiento transparente del proyecto**, fomentará la colaboración y garantizará que cada integrante tenga claridad sobre su rol y responsabilidades dentro de cada sprint.

2.9) Sistema de control de Versiones de código (Git)

El **control de versiones** es fundamental en el desarrollo colaborativo de software, ya que permite a los equipos mantener un historial de cambios, coordinar trabajo en paralelo y revertir modificaciones cuando sea necesario.

El sistema elegido para nuestro proyecto es **Git**, la herramienta más utilizada a nivel mundial por equipos de desarrollo de software.

1. ¿Por qué Git?

- **Distribuido:** cada miembro del equipo tiene una copia completa del repositorio, lo que permite trabajar incluso sin conexión.
- **Historial completo:** se mantiene un registro detallado de todos los cambios realizados en el código fuente.
- **Colaboración en equipo:** múltiples desarrolladores pueden trabajar al mismo tiempo en diferentes funcionalidades.
- **Integración con plataformas en la nube:** como Bitbucket, GitHub o GitLab, que facilitan la gestión de ramas, *pull requests* y revisiones de código.

2. Uso de Git en nuestro proyecto

1. Repositorio centralizado en la nube

- Se almacenará el repositorio en **Bitbucket/GitHub** para permitir acceso desde cualquier lugar.
- Cada integrante tendrá permisos según su rol.

2. Flujo de trabajo recomendado (Git Flow simplificado):

- **main** → rama estable, lista para despliegue.
- **develop** → rama de integración, donde se combinan nuevas funcionalidades.
- **feature/** → ramas temporales para el desarrollo de cada Historia de Usuario (ejemplo: feature/solicitud-credito).

- **hotfix/** → ramas de corrección de errores críticos detectados en producción.

3. Buenas prácticas del equipo:

- Commits descriptivos y consistentes (ejemplo: feat: agregar validación de CI en solicitudes).
- Revisión por pares antes de integrar cambios a develop o main.
- Uso de *pull requests* para mantener la calidad del código.

3. Beneficios esperados

- Mayor **organización y control** sobre las versiones del sistema.
- **Trazabilidad completa** de qué cambios se hicieron, cuándo y por quién.
- Facilidad para **integrar y probar nuevas funcionalidades** sin afectar la rama principal.
- Reducción de errores y conflictos gracias a la revisión por pares.

Conclusión

En nuestro proyecto, **Git será la herramienta de control de versiones** utilizada para coordinar el trabajo del equipo. Combinado con una plataforma en la nube (Bitbucket o GitHub), garantizará un desarrollo ágil, seguro y ordenado del “Software Web Y Móvil Para La Gestión De Solicitudes De Crédito, Evaluación De Solvencia En Entidades Financieras”

2.10) Herramientas de Gestión de Código en la Nube (Bitbucket)

En proyectos colaborativos, no basta con usar Git de manera local: es necesario contar con una plataforma en la nube que permita centralizar el repositorio, coordinar cambios, revisar el código y desplegar versiones. Para este proyecto, el equipo utilizará **Bitbucket**.

1. ¿Qué es Bitbucket?

- Plataforma de **gestión de repositorios Git en la nube**, desarrollada por Atlassian.
- Permite **almacenar, versionar y compartir código fuente** de manera segura entre los miembros del equipo.
- Se integra de forma nativa con otras herramientas ágiles como **Jira** y **Confluence**, lo cual facilita la trazabilidad entre tareas, documentación y código.

2. Uso de Bitbucket en nuestro proyecto

1. Repositorio centralizado

- El código del **Sistema Inteligente de Aprobación de Créditos con IA** se alojará en un repositorio privado de Bitbucket.
- Cada integrante tendrá acceso según su rol (administrador, desarrollador, revisor).

2. Integración con Git

- Todo el trabajo en ramas (main, develop, feature/*, hotfix/*) se sincronizará automáticamente con Bitbucket.
- El repositorio en la nube será la **fuentes de verdad** del proyecto.

3. Pull Requests y Code Review

- Cada cambio significativo deberá subirse mediante un **pull request**, que será revisado por al menos otro miembro antes de fusionarse.
- Esto asegura la calidad del código y la detección temprana de errores.

4. Pipelines (CI/CD) (*opcional, pero recomendable*)

- Bitbucket Pipelines puede configurarse para automatizar pruebas, compilación y despliegue del sistema en servidores de prueba o en la nube (ej: AWS EC2).
- Esto refuerza la disciplina de **integración continua (CI)** y **despliegue continuo (CD)**.

3. Beneficios esperados

- **Colaboración centralizada:** todos los integrantes trabajan en un único repositorio accesible desde cualquier lugar.
- **Seguridad:** repositorios privados con control de accesos.
- **Trazabilidad:** integración con Jira/Confluence para vincular commits y tareas.
- **Automatización:** posibilidad de usar Pipelines para pruebas y despliegues.

Conclusión

El uso de **Bitbucket** como plataforma de gestión de código en la nube complementa el sistema de control de versiones **Git**, ofreciendo un entorno seguro y colaborativo para el desarrollo del proyecto. Gracias a su integración con **Jira y Confluence**, el equipo podrá mantener un flujo de trabajo ordenado, ágil y con trazabilidad completa desde la planificación de historias de usuario hasta la entrega final del software

CAPITULO 3 REQUERIMIENTOS

3.1) Propósito

El propósito de este proyecto es desarrollar un sistema integral de información web y móvil para la gestión automatizada de solicitudes de crédito y la evaluación de solvencia en entidades financieras de Bolivia. Actualmente, el sistema financiero boliviano opera con metodologías tradicionales que se enfocan principalmente en trabajadores con ingresos formales y aportes a AFP, dejando excluidos a emprendedores, comerciantes y trabajadores independientes. Esto, sumado a procesos manuales y burocráticos, genera ineficiencias, elevados costos operativos, tiempos de respuesta prolongados y limita la capacidad de gestionar riesgos de manera proactiva.

Este sistema permitirá optimizar el workflow crediticio completo, integrando inteligencia artificial (IA) para analizar tanto información tradicional (historial crediticio, aportes AFP) como datos alternativos (facturación electrónica, flujos de caja, transacciones comerciales). La plataforma facilitará a los clientes solicitar créditos de manera digital, realizar simulaciones y realizar un seguimiento en tiempo real de sus solicitudes. Por otro lado, los oficiales de crédito y administradores dispondrán de dashboards intuitivos con indicadores de riesgo, herramientas de análisis predictivo y un sistema de aprobación/rechazo asistido por IA, todo ello dentro de un marco de seguridad y cumplimiento normativo (ASFI, BCB).

La automatización de estos procesos no solo incrementará la eficiencia operativa y reducirá los tiempos de espera, sino que también democratizará el acceso al crédito, fomentará la inclusión financiera y garantizará una experiencia de usuario ágil, segura y transparente para todos los actores involucrados.

3.2) Ámbito de Sistema

Para definir el ámbito del sistema, se realizó un análisis de las necesidades operativas de las entidades financieras y de las exigencias regulatorias vigentes en Bolivia. En la actualidad, gran parte de los procesos de solicitud y evaluación crediticia son manuales y burocráticos, lo que incrementa costos, genera demoras en las respuestas y dificulta la inclusión de segmentos sin historial crediticio formal.

El sistema propuesto tiene como objetivo central automatizar el workflow de aprobación de créditos mediante la integración de un motor de inteligencia artificial explicativa que permita predecir la solvencia de los solicitantes a partir de datos tradicionales (AFP, estados de cuenta) y datos alternativos (facturación electrónica, flujos de caja, historial de transacciones). De este modo, se busca reducir los tiempos de respuesta, optimizar la gestión de riesgos y fortalecer la inclusión financiera en el país.

El sistema abarcará tanto una plataforma web como una aplicación móvil, garantizando accesibilidad a clientes y oficiales de crédito. Entre sus funciones se incluyen:

Registro y validación digital de solicitudes.

Motor de workflow para el seguimiento del estado de cada solicitud (en revisión, aprobado, rechazado, desembolsado).

Evaluación crediticia asistida por IA con justificación de decisiones.

Gestión de productos financieros y condiciones crediticias configurables.

Integración con pasarelas de pago y sistemas QR interoperables.

Asimismo, el sistema deberá garantizar seguridad, trazabilidad y cumplimiento normativo, mediante autenticación multifactor (MFA), cifrado de datos y auditoría de operaciones.

En suma, el ámbito del sistema se centra en ofrecer una solución digital integral que automatice y optimice la gestión de créditos, incremente la eficiencia operativa de las entidades financieras y mejore la experiencia del cliente, sin dejar de cumplir con los estándares regulatorios y de seguridad exigidos en el sector.

3.3) Equipo SCRUM

ROL	TAREA
Product Owner	Encargado de la coordinación y acuerdo con el cliente, es el mediador entre el cliente y el resto del equipo.
Scrum Master	Encargado de asegurar que el equipo siga el plan SCRUM y encargado de hacer posible el desarrollo del proyecto bajo la metodología.
Team Developer	Encargado del desarrollo de los artefactos que se necesiten para el avance del proyecto y desarrolladores del software.

SPRINT 0	
Perfil	<ul style="list-style-type: none"> ❖ Banegas Roca Luis Fernando ❖ Flores Quispe Nathaly Madeline ❖ Melgar Arteaga Alejandro ❖ Mendoza Rodriguez Luis Pablo ❖ Molina Cortez Maikol Anthony ❖ Perez Gomez Daniela
SPRINT 1	
Product Owner	❖ Banegas Roca Luis Fernando
Scrum Master	❖ Melgar Arteaga Alejandro
Team Developer	<ul style="list-style-type: none"> ❖ Mendoza Rodríguez Luis Pablo ❖ Flores Quispe Nathaly Madeline ❖ Molina Cortez Maikol Anthony ❖ Perez Gomez Daniela

SPRINT 2	
Product Owner	❖ Molina Cortez Maikol Anthony
Scrum Master	❖ Perez Gomez Daniela
Team Developer	<ul style="list-style-type: none"> ❖ Banegas Roca Luis Fernando ❖ Melgar Arteaga Alejandro ❖ Mendoza Rodríguez Luis Pablo ❖ Flores Quispe Nathaly Madeline

SPRINT 3	
Product Owner	❖ Mendoza Rodríguez Luis Pablo
Scrum Master	❖ Flores Quispe Nathaly Madeline

Team Developer	<ul style="list-style-type: none"> ❖ Banegas Roca Luis Fernando ❖ Melgar Arteaga Alejandro ❖ Molina Cortez Maikol Anthony ❖ Perez Gomez Daniela
-----------------------	---

SPRINT 4	
Product Owner	❖ Banegas Roca Luis Fernando
Scrum Master	❖ Melgar Arteaga Alejandro
Team Developer	<ul style="list-style-type: none"> ❖ Banegas Roca Luis Fernando ❖ Flores Quispe Nathaly Madeline ❖ Melgar Arteaga Alejandro ❖ Mendoza Rodríguez Luis Pablo ❖ Molina Cortez Maikol Anthony ❖ Perez Gomez Daniela

3.4) Definiciones, Acrónimos y Abreviaturas

Administrador: Usuario con los mayores privilegios dentro del sistema. Tiene acceso total a la plataforma, permitiéndole gestionar usuarios, solicitudes de crédito, productos financieros, reportes, pagos, supervisa el motor de IA.

oficial de credito: Profesional encargado en evaluar solicitudes crediticias. Analiza datos tradicionales y alternativos, y emite recomendaciones basadas en el scoring generado por el sistema.

Cliente: Usuario final que accede a los servicios ofrecidos por la entidad. Puede registrarse, realizar solicitudes, realizar pagos, consultar su historial financiero y comunicarse con ejecutivos.

Solicitud: Proceso mediante el cual un cliente inicia una petición de crédito, indicando monto, plazo, tipo de producto y documentación requerida.

Producto Financiero: Conjunto de servicios o beneficios ofrecidos por la entidad financiera, tales como (tipo de crédito, tasa de interés, plazo, garantías), agrupadas según perfil del cliente.

CU: Caso de Uso

DB: Base de Datos

HU: Historia de Usuario

RF: Requisito funcional

RNF: requisito No Funcional

S: Sprint (iteración de desarrollo en Scrum)

3.5) Funciones del Producto

El Software web y móvil para la gestión de solicitudes de crédito, evaluación de solvencia en entidades financieras tendrá como propósito principal **automatizar y optimizar el proceso de otorgamiento de créditos** en entidades financieras, integrando reglas de negocio, validación documental y análisis predictivo mediante inteligencia artificial.

Las **funciones del producto** se organizan en tres grandes bloques: **aplicación web, aplicación móvil y módulos transversales.**

Módulos Aplicación Web (Oficiales y Administradores)

- **Gestión de usuarios y roles:** registro, modificación, suspensión de usuarios internos, asignación de roles (administrador, oficial de crédito, analista de riesgo, auditor).
- **Gestión de clientes internos:** base centralizada de clientes, con historial crediticio y documentos asociados.
- **Gestión de solicitudes de crédito:** recepción digital, validación automática de requisitos, control de estados del workflow (en revisión, en análisis, aprobado, rechazado, desembolsado).
- **Evaluación automática con IA:** cálculo de scoring crediticio en tiempo real, con opción de revisión manual por un oficial de crédito.
- **Gestión de productos financieros:** configuración de tipos de crédito (consumo, PyME, vivienda), tasas de interés, límites según solvencia.
- **Gestión de pagos, cobros y cartera post-credito:** generación de plan de pagos, registro de pagos, alertas de mora, integración con pasarelas de pago y sistemas QR, además de realizar seguimiento de pagos.
- **Reportes y cumplimiento regulatorio:** exportación de informes PDF/Excel para ASFI/BCB, indicadores de inclusión financiera, métricas de aprobación/rechazo.

- **Dashboards analíticos:** visualización en tiempo real de indicadores clave: solicitudes procesadas, niveles de riesgo, segmentación de clientes.
- **Auditoría avanzada:** bitácora completa de todas las acciones realizadas dentro del sistema (web y móvil) para fines de seguridad y control.
- **Soporte interno:** módulo para que oficiales registren consultas o reclamos recibidos por ventanilla, teléfono o medios presenciales.

Módulos Aplicación Móvil (Clientes)

- **Gestión de registro y acceso:** creación de cuenta con validación de identidad (KYC) y autenticación multifactor (MFA).
- **Solicitud de créditos:** formulario digital con carga de documentos (CI, boletas de pago, extractos bancarios).
- **Simulador de préstamos:** cálculo de cuotas, tasas de interés y condiciones de crédito.
- **Seguimiento de solicitudes:** consulta en tiempo real del estado de aprobación/rechazo.
- **Gestión de pagos:** opción de pago de cuotas mediante la app (QR interoperable, transferencia bancaria, tarjeta).
- **Historial de créditos y pagos:** consulta de solicitudes pasadas, créditos activos y pagos realizados.
- **Notificaciones:** alertas automáticas sobre aprobación, rechazo, vencimiento de cuotas y recordatorios de pago.
- **Atención al cliente en línea:** módulo de soporte (chat en línea y tickets) para consultas o reclamos.

Módulos Transversales (Web y Móvil)

- **Seguridad:** cifrado extremo a extremo, autenticación multifactor, control de accesos y cumplimiento normativo (ASFI/BCB).
- **Motor de IA explicativa (XAI):**

- Scoring crediticio predictivo.
- Justificación clara de las decisiones (para clientes y oficiales).
- Identificación de inconsistencias en los documentos cargados.
- **Integración y APIs:** operaciones con sistemas externos (AFP, burós de crédito, sistemas core bancarios) y APIs públicas para conexión con Fintechs.
- **Gestión documental avanzada:** expediente digital seguro, con validación de formatos y almacenamiento estructurado de documentos.
- **Respaldo y recuperación:** backups automáticos (semanal) de la base de datos y documentos para garantizar disponibilidad.
- **Comunicación:** envío de notificaciones push, correos electrónicos y mensajes SMS para mantener informado al cliente.

3.6) Product Backlog

Product Backlog							
Proyecto	Software web y móvil para la gestión de solicitudes de crédito, evaluación de solvencia en entidades financieras						
Product Owner	Banegas Roca Luis Fernando						
Versión	Sprint 0 – Documentación inicial (2 semanas)						
ID	Rol	Encargado	Característica/ Funcionalidad	Tiempo	Prioridad	Estado	Plataforma
SP0-01	Equipo	Banegas Luis	Elaboración del perfil del proyecto (Cap. 1 y 2)	Alta	Pendiente	Documentación	SP0-01
SP0-02	Equipo	Flores Nathaly	Definición del alcance y funciones del producto (3.5a)	Alta	Pendiente	Documentación	SP0-02
SP0-03	Equipo	Melgar Alejandro	Redacción del Product Backlog (3.5b)	Alta	Pendiente	Documentación	SP0-03
SP0-04	Equipo	Mendoza Luis	Definición de requisitos funcionales y no funcionales (3.6, 3.7)	Alta	Pendiente	Documentación	SP0-04
SP0-05	Equipo	Molina Maikol	Elaboración de lista y diagramas de casos de uso (3.8, 3.9)	Alta	Pendiente	Documentación	SP0-05
SP0-06	Equipo	Pérez Daniela	Planificación de sprints y diagrama de Gantt (3.10)	Alta	Pendiente	Documentación	SP0-06

Product Backlog							
Proyecto	Software web y móvil para la gestión de solicitudes de crédito, evaluación de solvencia en entidades financieras						
Product Owner	Banegas Roca Luis Fernando						
Scrum Master	Melgar Arteaga Alejandro						
Sprint 1 – Registro, seguridad y base del sistema (2 semanas)							
ID	Rol	Encargado	Funcionalidad	Tiempo	Prioridad	Estado	Plataforma
SP1-01	Scrum Master	Melgar Alejandro	Iniciar sesión (CU01)	Alta	Pendiente	Web/Móvil	SP1-01
SP1-02	Team Developer	Flores Madeline	Cerrar sesión (CU02)	Alta	Pendiente	Web/Móvil	SP1-02
SP1-03	Scrum Master	Melgar Alejandro	Recuperar contraseña (CU03)	Alta	Pendiente	Web/Móvil	SP1-03
SP1-04	Product Owner	Banegas Luis	Gestionar usuarios (CU04)	Alta	Pendiente	Web	SP1-04
SP1-05	Team Developer	Pérez Daniela	Gestionar roles y permisos (CU05)	Alta	Pendiente	Web	SP1-05
SP1-06	Product Owner	Banegas Luis	Gestionar cliente (CU06)	Alta	Pendiente	Web/Móvil	SP1-06
SP1-07	Team Developer	Molina Maikol	Gestionar personal (CU07)	Alta	Pendiente	Web/Móvil	SP1-07
SP1-08	Team Developer	Mendoza Luis	Consentimiento de tratamiento de datos (CU08)	Alta	Pendiente	Web/Móvil	SP1-08

SP1-09	Team Developer	Mendoza Luis	Gestionar bitácora (CU09)	Alta	Pendiente	Web/Móvil	SP1-09
--------	----------------	--------------	---------------------------	------	-----------	-----------	--------

Product Backlog							
Proyecto	Software web y móvil para la gestión de solicitudes de crédito, evaluación de solvencia en entidades financieras						
Product Owner	Molina Cortez Maikol Anthony						
Scrum Master	Perez Gomez Daniela						
Sprint 2 – Solicitudes de crédito e IA inicial (2 semanas)							
ID	Rol	Encargado	Funcionalidad	Tiempo	Prioridad	Estado	Plataforma
SP2-01	Team Developer	Melgar & Flores	Solicitar crédito (CU10)	Alta	Pendiente	Web/Móvil	SP2-01
SP2-02	Team Developer	Flores Madeline	Simular crédito (CU11)	Alta	Pendiente	Web/Móvil	SP2-02
SP2-03	Product Owner	Molina Maikol	Registrar solicitud de crédito (CU12)	Alta	Pendiente	Web	SP2-03
SP2-04	Team Developer	Molina & Mendoza	Validar información (CU13)	Alta	Pendiente	Web	SP2-04
SP2-05	Scrum Master	Pérez Daniela	Aprobar / rechazar solicitud (CU14)	Alta	Pendiente	Web	SP2-05
SP2-06	Team Developer	Mendoza Luis	Generar plan de pagos (CU15)	Alta	Pendiente	Web	SP2-06
SP2-07	Team Developer	Banegas Luis	Seguir estado de solicitud (CU16)	Alta	Pendiente	Web/Móvil	SP2-07
SP2-08	Team Developer	Pérez & Banegas	Ver historial de créditos (CU17)	Media	Pendiente	Web/Móvil	SP2-08
SP2-09	Team Developer	Pérez Daniela	Gestionar productos financieros (CU18)	Media	Pendiente	Web	SP2-09
SP2-10	Team Developer	Molina & Mendoza	Cargar y gestionar documentos (CU19)	Alta	Pendiente	Web/Móvil	SP2-10

Product Backlog							
Proyecto	Software web y móvil para la gestión de solicitudes de crédito, evaluación de solvencia en entidades financieras						
Product Owner	Mendoza Rodríguez Luis Pablo						
Scrum Master	Flores Quispe Nathaly Madeline						
Sprint 3 – Pagos, seguimiento y comunicación (2 semanas)							
ID	Rol	Encargado	Funcionalidad	Tiempo	Prioridad	Estado	Plataforma
SP3-01	Product Owner	Mendoza Luis	Evaluar solicitud con IA (CU20)	Alta	Pendiente	Web	SP3-01
SP3-02	Team Developer	Pérez Daniela	Pagar cuotas (CU21)	Alta	Pendiente	Web/Móvil	SP3-02

SP3-03	Team Developer	Pérez & Banegas	Registrar pagos (CU22)	Alta	Pendiente	Web	SP3-03
SP3-04	Scrum Master	Flores Madeline	Generar comprobante digital (CU23)	Alta	Pendiente	Web/Móvil	SP3-04
SP3-05	Team Developer	Melgar Alejandro	Consultar dashboard (CU24)	Media	Pendiente	Web/Móvil	SP3-05
SP3-06	Team Developer	Melgar & Flores	Generar reportes (CU25)	Alta	Pendiente	Web	SP3-06
SP3-07	Team Developer	Banegas Luis	Gestionar notificaciones (CU26)	Media	Pendiente	Web/Móvil	SP3-07
SP3-08	Team Developer	Molina & Mendoza	Gestionar reclamos (CU27)	Media	Pendiente	Web/Móvil	SP3-08
SP3-09	Team Developer	Banegas Luis	Generar y aceptar términos/contrato (CU28)	Alta	Pendiente	Web/Móvil	SP3-09
SP3-10	Team Developer	Molina Maikol	Integrar pasarela / QR interoperable (CU29)	Alta	Pendiente	Web/Móvil	SP3-10

Product Backlog							
Proyecto	Software web y móvil para la gestión de solicitudes de crédito, evaluación de solvencia en entidades financieras						
Product Owner	Banegas Roca Luis Fernando						
Scrum Master	Melgar Arteaga Alejandro						
Sprint 4 – Reportes, dashboards y cierre (2 semanas)							
ID	Rol	Encargado	Funcionalidad	Tiempo	Prioridad	Estado	Plataforma
SP4-01	Team Developer	Flores Madeline	Reprogramación / Refinanciamiento (CU30)	Media	Pendiente	Web/Móvil	SP4-01
SP4-02	Product Owner	Banegas & Pérez	Pago anticipado / Cancelación total (CU31)	Media	Pendiente	Web/Móvil	SP4-02
SP4-03	Scrum Master	Melgar Alejandro	Conciliación automática (CU32)	Alta	Pendiente	Web	SP4-03
SP4-04	Team Developer	Pérez Daniela	Acceso biométrico (CU33)	Media	Pendiente	Móvil	SP4-04
SP4-05	Team Developer	Molina & Mendoza	Gestionar formularios (CU34)	Baja	Pendiente	Web/Móvil	SP4-05
SP4-06	Team Developer	Melgar & Flores	Gestión de catálogos (CU35)	Baja	Pendiente	Web	SP4-06

3.7) Requisitos Funcionales

ID	FUNCIONALIDAD	DESCRIPCION	PLATAFORMA
RF 01	Registro de usuarios	Permite crear cuentas con datos personales, rol asignado y credenciales seguras	Web y Móvil
RF 02	Inicio de sesión seguro	Autenticación mediante usuario y contraseña con validación de rol	Web y Móvil
RF 03	Solicitud de créditos	El usuario puede generar/crear solicitudes para ingresar monto, plazo, tipo de producto y garantías	Móvil
RF 04	Simulador de crédito	Calcula cuotas, tasas y plazos según parámetros ingresados por el usuario	Móvil
RF 05	Seguimiento de estado de solicitud	Visualización del estado actual y trazabilidad del proceso	Web y Móvil
RF 06	Carga de documentos adjuntos	Subida de archivos requeridos (CI, extractos, etc.)	Web y Móvil
RF 07	Gestión de perfil de usuario	Edición de datos personales, contraseña y preferencias	Web y Móvil
RF 08	Evaluación automática con IA	Cálculo de scoring crediticio en tiempo real con opción de revisión manual	Web
RF 09	Evaluación y aprobación de solicitudes	Funcionalidad para que el evaluador revise, apruebe o rechace solicitudes	Web
RF 10	Gestionar historial de solicitudes	Registro completo y ordenado de todas las solicitudes realizadas, créditos activos y pagos realizados	Web y Móvil
RF 11	Gestión de pagos y cartera post-crédito	Generación de plan de pagos, registro de pagos, alertas de mora, integración QR	Web y Móvil
RF 12	Gestión de productos financieros	Configuración de tipos de crédito, tasas y límites según perfil de solvencia	Web
RF 13	Generar reportes	Exportación de informes PDF/Excel para ASFI/BCB por rango de fechas, tipo de producto o estado	Web
RF 14	Notificaciones automáticas	Alertas sobre aprobación, rechazo, vencimientos y recordatorios	Web y Móvil
RF 15	Seguridad y cumplimiento normativo	Cifrado de datos, autenticación multifactor y	Web y Móvil

		control de accesos conforme normativa local	
RF 16	Registro de bitácora	El sistema debe registrar todas las acciones realizadas por los usuarios (crear, modificar, eliminar, consultar).	Web
RF 17	Gestión de formularios	El sistema debe permitir al administrador crear, modificar o eliminar formularios utilizados en las solicitudes de crédito.	Web

3.8) Requisitos no funcionales

ID	REQUISITO NO FUNCIONAL	DESCRIPCIÓN	PLATAFORMA	PRIORIDAD
RNF 01	Seguridad y cifrado	Todo dato sensible (credenciales, datos personales, financieros) debe cifrarse en tránsito y reposo. Cumplir normativas ASFI/BCB.	Web y Móvil	Alta
RNF 02	Autenticación robusta	Implementar autenticación con múltiples factores (MFA/2FA) para usuarios con privilegios elevados.	Web y Móvil	Alta
RNF 03	Rendimiento	El sistema debe responder en menos de 2 segundos en operaciones comunes (simulaciones, login, seguimiento).	Web y Móvil	Alta
RNF 04	Alta disponibilidad	El sistema debe estar disponible el 99.9% del tiempo, especialmente durante horario bancario.	Web y Móvil	Alta
RNF 05	Escalabilidad	La arquitectura debe escalar horizontalmente para soportar crecimiento (nuevas sucursales, 5.000+ usuarios concurrentes).	Web y Móvil	Media

RNF 06	Usabilidad	Interfaces intuitivas, amigables y adaptadas para personas con conocimientos básicos. Tiempo de aprendizaje < 30 minutos.	Web y Móvil	Alta
RNF 07	Compatibilidad multiplataforma	La web debe funcionar en Chrome, Firefox, Safari, Edge. La app móvil debe funcionar en Android 8+ y iOS 12+.	Web y Móvil	Alta
RNF 08	Tiempo de respuesta crítico	Operaciones críticas deben responder en máximo 3 segundos bajo carga normal.	Web y Móvil	Alta
RNF 09	Registro de auditoría	Toda acción de login, evaluación, aprobación o rechazo debe registrarse con usuario, fecha, IP y acción.	Web y Móvil	Alta
RNF 10	Mantenibilidad y pruebas	El código debe tener documentación clara, pruebas unitarias (>80%) y facilitar el mantenimiento modularizado.	Web	Media
RNF 11	Cumplimiento regulatorio	El sistema debe generar alertas, reportes e interfaces compatibles con exigencias de ASFI, BCB y normativas locales.	Web y Móvil	Alta
RNF 12	Integración externa segura	Toda integración con APIs externas debe usar OAuth2 o equivalentes, con logs de acceso.	Web y Móvil	Alta
RNF 13	Trazabilidad	Cada solicitud, evaluación y modificación debe poder ser trazada de forma íntegra desde su creación hasta su cierre o pago.	Web y Móvil	Alta

RNF 14	Respaldo y recuperación	Backups automáticos diarios y recuperación ante desastres en menos de 4 horas.	Web	Alta
RNF 15	Internacionalización	El sistema debe permitir fácilmente el cambio de idioma (Español/Inglés) y formatos de moneda/fecha adaptables.	Web y Móvil	Baja

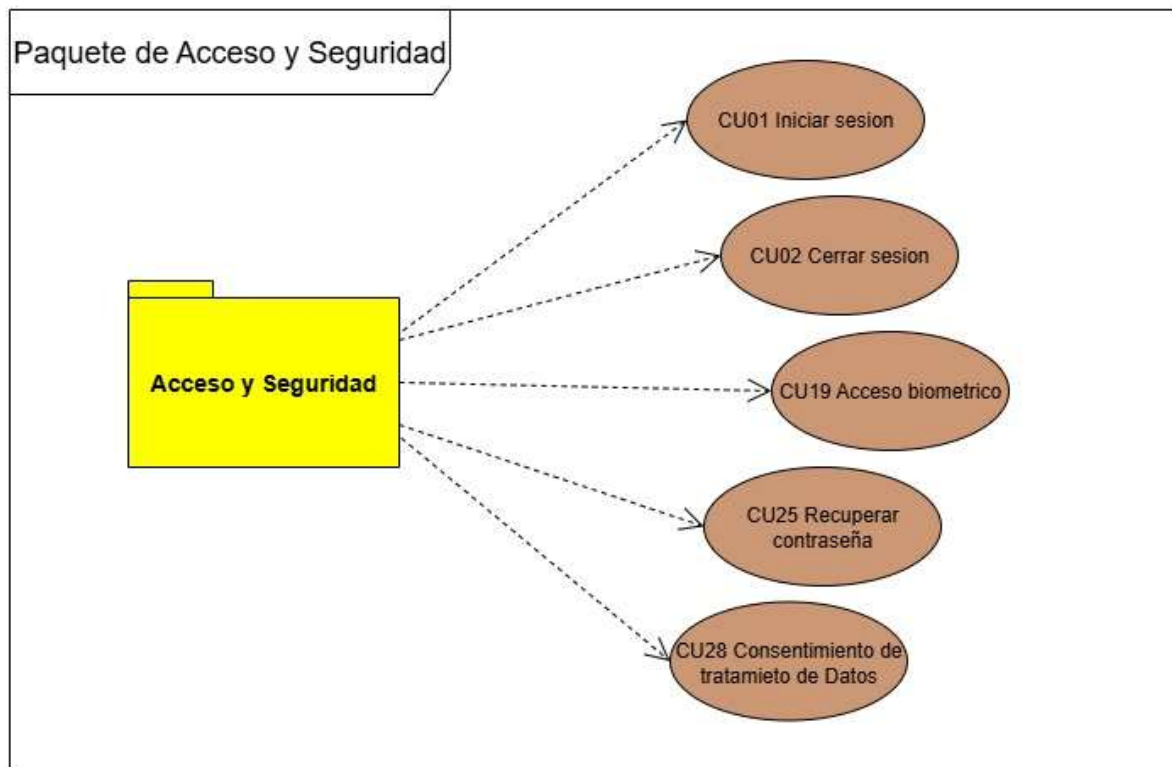
3.9) Lista de casos de uso

ID	Nombre de caso de uso	Sprint asignado	Descripción
CU01	Iniciar sesión (Web / Móvil)	Sprint 1	Acceso al sistema con credenciales seguras.
CU02	Cerrar sesión (Web / Móvil)	Sprint 1	Cierre seguro de la sesión.
CU03	Recuperar contraseña (Web / Móvil)	Sprint 1	Recuperación de acceso vía correo o MFA.
CU04	Gestionar usuarios (Web)	Sprint 1	Registro, modificación, suspensión y eliminación de usuarios internos.
CU05	Gestionar roles y permisos (Web)	Sprint 1	Definir y asignar roles con permisos específicos.
CU06	Gestionar cliente (Web / Móvil)	Sprint 1	Alta, modificación, eliminación y listado de clientes.
CU07	Gestionar personal (Web / Móvil)	Sprint 1	Alta, modificación, eliminación y listado de personal.
CU08	Consentimiento de tratamiento de datos (Web / Móvil)	Sprint 1	Aceptación de cláusulas de uso de datos (LOPD/ASFI).
CU09	Gestionar bitácora (Web / Móvil)	Sprint 1	Registro de todas las acciones y usuarios en el sistema.
CU10	Solicitar crédito (Móvil / Web)	Sprint 2	Envío de solicitud con formularios y documentos.
CU11	Simular crédito (Móvil / Web)	Sprint 2	Cálculo de cuotas y plazos antes de solicitar.
CU12	Registrar solicitud de crédito (Web)	Sprint 2	Alta de solicitud por oficial con datos del cliente.
CU13	Validar información (Web)	Sprint 2	Verificación de documentos y cruces con sistemas externos.
CU14	Aprobar / rechazar solicitud (Web)	Sprint 2	Decisión de aprobación manual, automática o mixta.
CU15	Generar plan de pagos (Web)	Sprint 2	Cálculo del cronograma de pagos.
CU16	Seguir estado de solicitud (Móvil / Web)	Sprint 2	Consultar y recibir notificaciones del estado de la solicitud.
CU17	Ver historial de créditos (Web / Móvil)	Sprint 2	Consulta de créditos anteriores y su estado.
CU18	Gestionar productos financieros (Web)	Sprint 2	Configuración de tipos de crédito, tasas y condiciones.

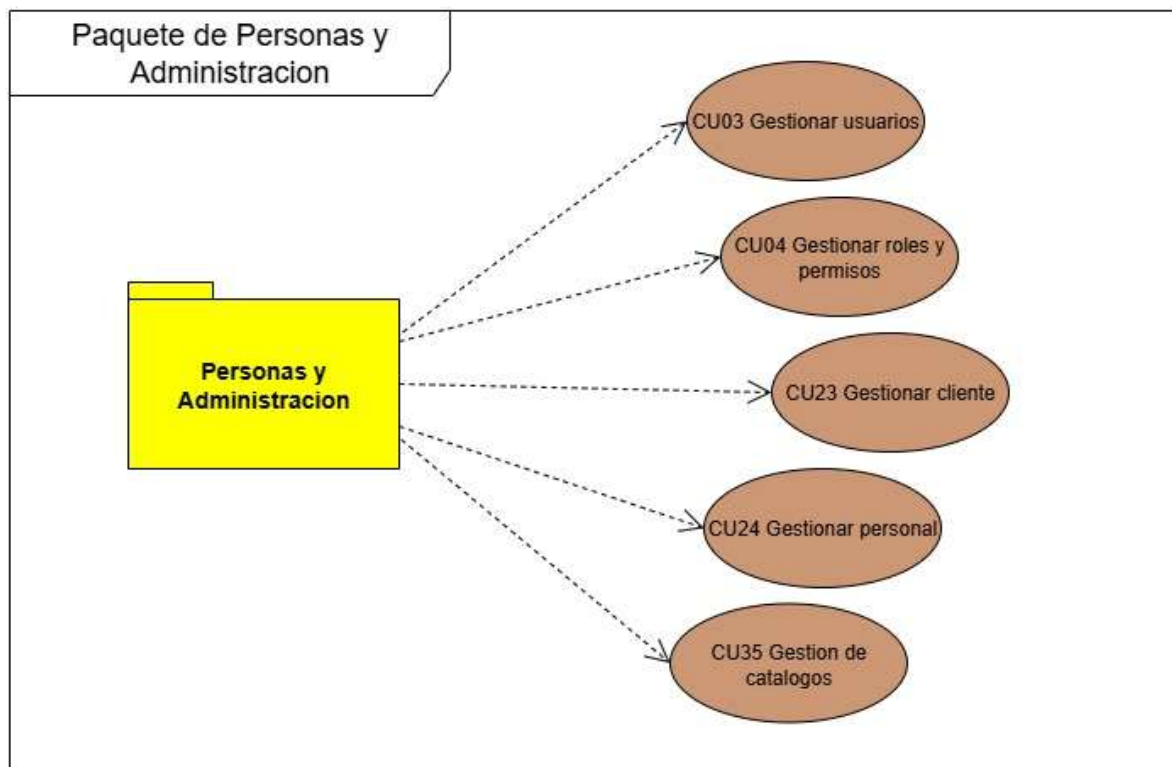
CU19	Cargar y gestionar documentos (Móvil / Web)	Sprint 2	Adjuntar archivos con validación de formato y vigencia.
CU20	Evaluar solicitud con IA (Web)	Sprint 3	Obtención de scoring crediticio con justificación.
CU21	Pagar cuotas (Móvil / Web)	Sprint 3	Pago de cuotas mediante tarjeta, QR o transferencia.
CU22	Registrar pagos (Web)	Sprint 3	Registro manual o verificación de pagos automáticos.
CU23	Generar comprobante digital (Web / Móvil)	Sprint 3	Emisión de comprobantes en PDF/digital por pago realizado.
CU24	Consultar dashboard (Móvil / Web)	Sprint 3	Visualización de métricas de riesgo y proyecciones.
CU25	Generar reportes (Web)	Sprint 3	Creación y exportación de reportes regulatorios y de gestión.
CU26	Gestionar notificaciones (Web / Móvil)	Sprint 3	Envío de alertas y novedades por push o email.
CU27	Gestionar reclamos (Móvil / Web)	Sprint 3	Registro y seguimiento de reclamos o consultas.
CU28	Generar y aceptar términos/contrato (Web / Móvil)	Sprint 3	Creación del contrato y firma electrónica.
CU29	Integrar pasarela / QR interoperable (Web / Móvil)	Sprint 3	Conexión con pasarela bancaria o QR para procesar pagos.
CU30	Reprogramación / Refinanciamiento (Web / Móvil)	Sprint 4	Generar nuevos planes de pago en caso de dificultad.
CU31	Pago anticipado / Cancelación total (Web / Móvil)	Sprint 4	Liquidación anticipada con cálculo de intereses corridos.
CU32	Conciliación automática (Web)	Sprint 4	Comparación de pagos recibidos vs. cuotas para ajustar saldos.
CU33	Acceso biométrico (Móvil)	Sprint 4	Añadir o quitar métodos biométricos de acceso.
CU34	Gestionar formularios (Web / Móvil)	Sprint 4	Modificación, creación o eliminación de campos de formularios.
CU35	Gestión de catálogos (Web)	Sprint 4	Administración de listas de actividades, ocupaciones y unidades.

3.10) Paquetes y casos de Uso

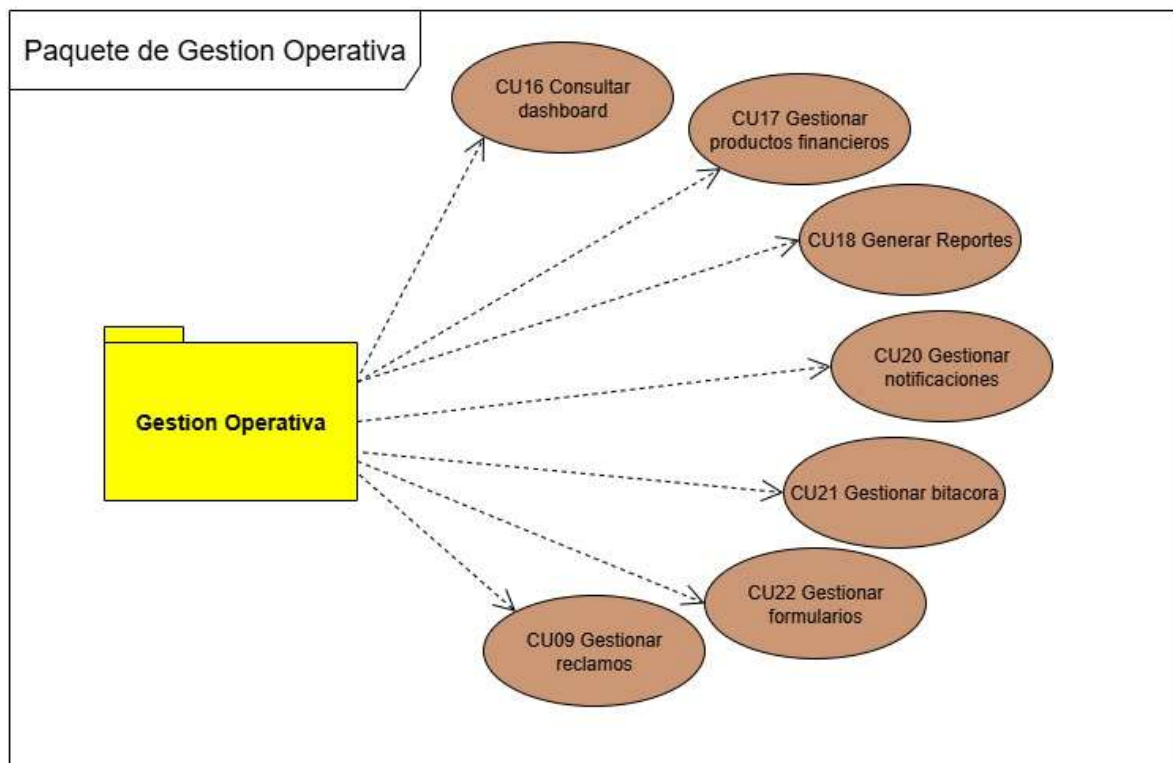
Paquete de Acceso y Seguridad



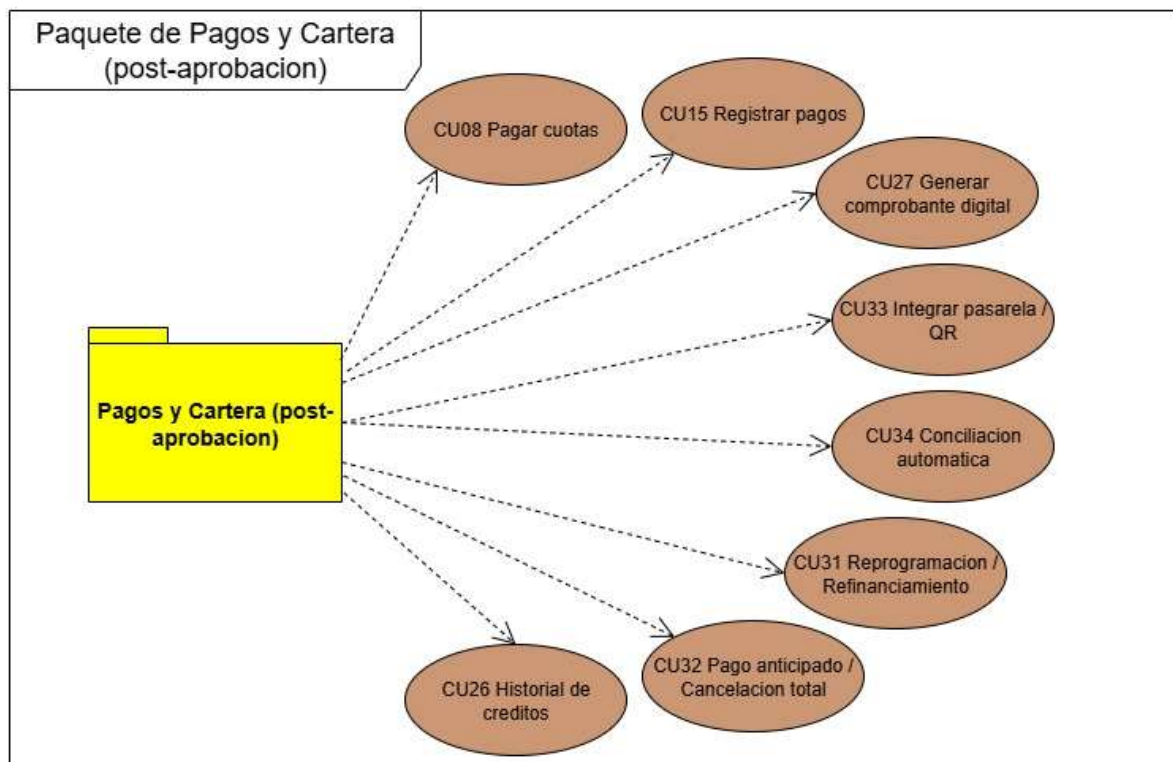
Paquete de Personal y Administración



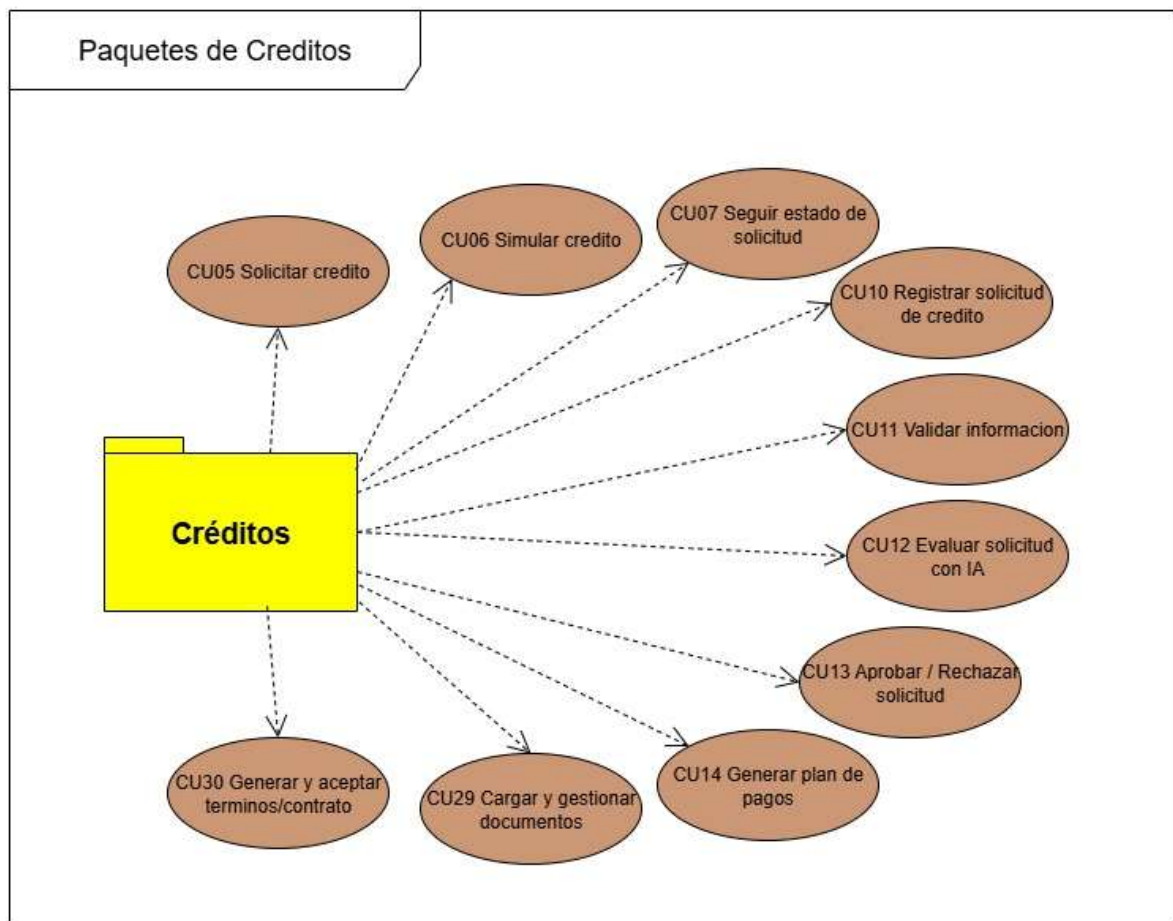
Paquete de Gestión Operativa



Paquete de Pagos y Cartera



Paquete de Creditos



3.11) Planificación Sprint

ID	Descripción	SPRINT 1 (02/09/2025 hasta 14/09/2025)												
		2/09	3/09	4/09	5/09	6/09	7/09	8/09	9/09	10/09	11/09	12/09	13/09	14/09
Tarea 1	Crear repositorio en GitHub													
Tarea 2	Instalar Django en entorno local													
Tarea 3	Crear proyecto en Django													
Tarea 4	Hacer el diseño de la base de datos													
Tarea 5	Configurar conexión con la base de datos													
Tarea 6	Crear migraciones a la base de datos													
Tarea 7	Crear formulario de Registro web													
Tarea 8	Crear formulario de login web													
Tarea 9	Definir roles													
Tarea 10	Validar inicio de sesión													
Tarea 11	Crear dashboard													
Tarea 12	Crear el proyecto móvil													
Tarea 13	Diseñar pantalla de registro en flutter													
Tarea 14	Mostrar mensaje de "login exitoso" en móvil													
Tarea 15	Configurar despliegue en la nube													
Tarea 16	Subir el backend en la nube													
Tarea 17	Probar el login y registro desde web y móvil													

SPRINT 2

ID	Descripción	SPRINT 2 (19/09/2025 hasta 12/10/2025)												
		19-sep	20-sep	21-sep	22-sep	23-sep	24-sep	25-sep	26-sep	27-sep	28-sep	29-sep	30-sep	01-oct
Tarea 1	Diseñar formulario solicitud de credito													
Tarea 2	Implementar formulario													
Tarea 3	Subir documento para la solicitud													
Tarea 4	Validar tipo y vigencia de documentos													
Tarea 5	Crear pantalla de simulación de crédito													
Tarea 6	Programar cálculo de cuotas y plazos													
Tarea 7	Registrar solicitud por parte del oficial (Web)													
Tarea 8	Validar campos obligatorios en solicitud del oficial													
Tarea 9	Verificar datos con fuentes externas													
Tarea 10	Diseñar lógica de validación documental													
Tarea 11	Crear módulo de aprobación manual													
Tarea 12	Configurar aprobación automática y reglas mixtas													
Tarea 13	Generar plan de pagos en formato tabla													
Tarea 14	Diseñar cronograma de pagos visual													
Tarea 15	Ver estado de solicitud (Web/Móvil)													
Tarea 16	Notificar al usuario cambios de estado													
Tarea 17	Consultar historial de créditos anteriores													
Tarea 18	Diseñar vista historial de créditos													

Tarea 19	Diseñar panel de administración de productos financieros												
Tarea 20	Registrar nuevos tipos de crédito Cargar condiciones de tasa y plazos												
Tarea 21	Configurar sistema para adjuntar y validar documentos												
Tarea 22	Diseñar vista historial de créditos												

CAPITULO 4 PROCESO DE DESARROLLO DE SOFTWARE

SPRINT 1

1. Sprint Planning

1.1.Objetivos del Sprint

El objetivo principal de este sprint es sentar las bases técnicas y funcionales del sistema, implementando los módulos esenciales de acceso, seguridad, gestión de usuarios y cumplimiento normativo. Este sprint se enfoca en construir un sistema seguro, escalable y preparado para integraciones futuras. Durante este periodo, el equipo se concentrará en desarrollar las siguientes funcionalidades clave:

CU01: Iniciar sesión (Web / Móvil)

CU02: Cerrar sesión (Web / Móvil)

CU03: Recuperar contraseña (Web / Móvil)

CU04: Gestionar usuarios (Web)

CU05: Gestionar roles y permisos (Web)

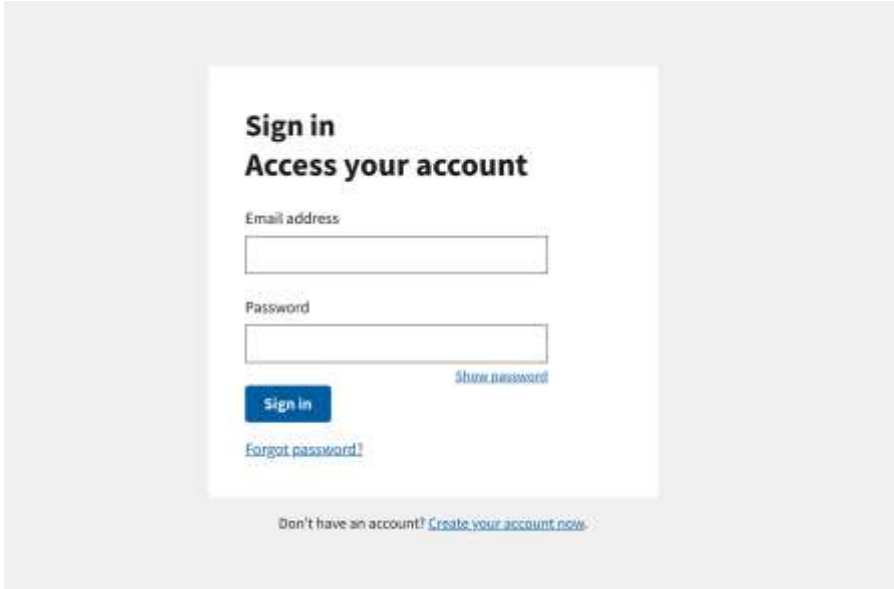
CU06: Gestionar cliente (Web / Móvil)


CU07: Gestionar personal (Web / Móvil)


CU08: Consentimiento de tratamiento de datos (Web / Móvil)


CU09: Gestionar bitácora (Web / Móvil)

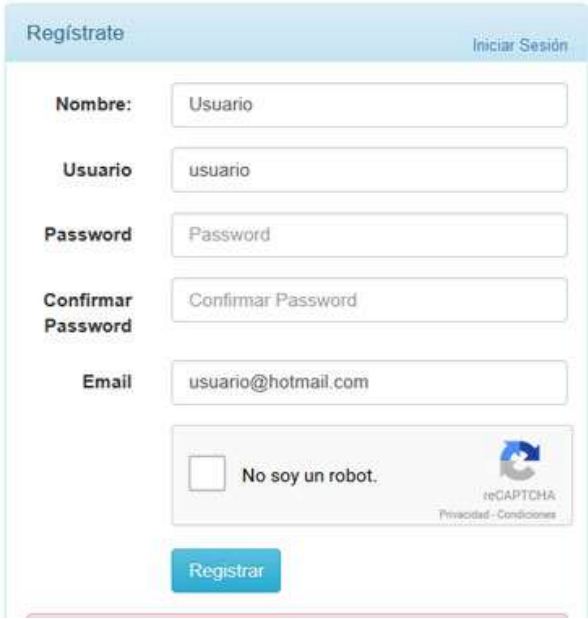
1.2. Historias de usuario (tarjetas 3C, Planning Póker, prototipos)

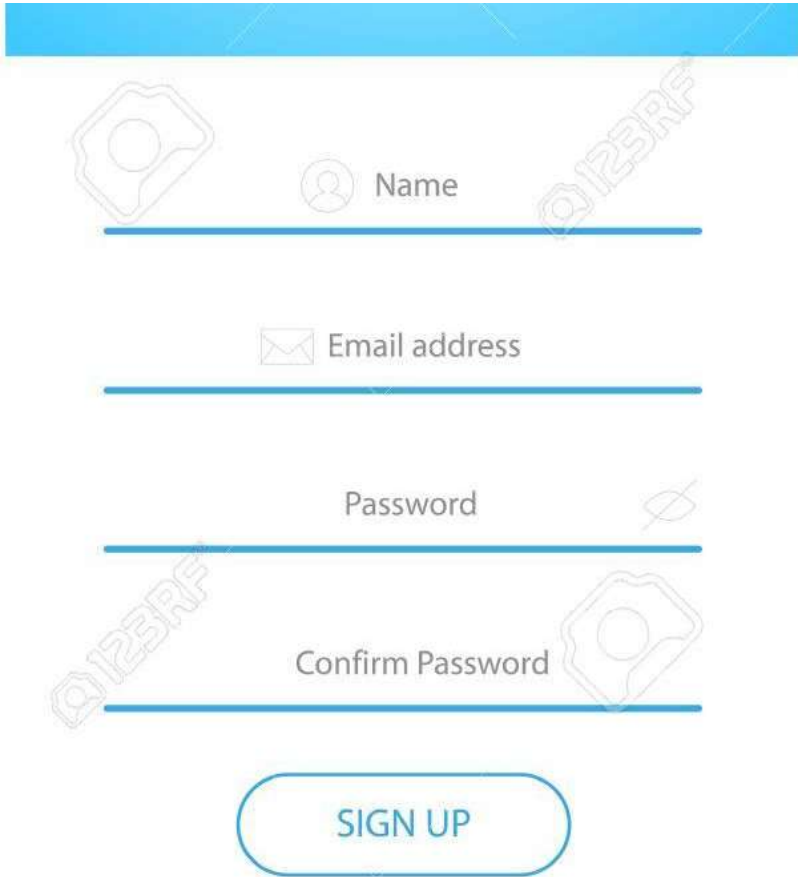
HU01- Inicio de sesión Web	
Nombre de HU	Inicio de sesión Web
Prioridad	Alta
HUEstado	Terminada
Criterios de aceptación:	<ul style="list-style-type: none">• El sistema debe validar correctamente el nombre de usuario y la contraseña.• Si las credenciales son válidas, debe redirigir al usuario al dashboard o pantalla principal.• En caso de error, se debe mostrar un mensaje claro e informativo.
Desarrollado por:	Alejandro Melgar
Prototipo:	
	

HU02- Inicio de sesión Móvil	
Nombre de HU	Inicio de sesión Móvil
Prioridad	Alta
HUEstado	Terminada
<ul style="list-style-type: none"> • Criterios de aceptación: 	<ul style="list-style-type: none"> • El sistema debe validar correctamente el nombre de usuario y la contraseña. • Si las credenciales son válidas, debe redirigir al usuario al dashboard o pantalla principal. • En caso de error, se debe mostrar un mensaje claro e informativo.
Desarrollado por:	Madeline Nathaly Flores Quispe
Prototipo: <div>  </div>	

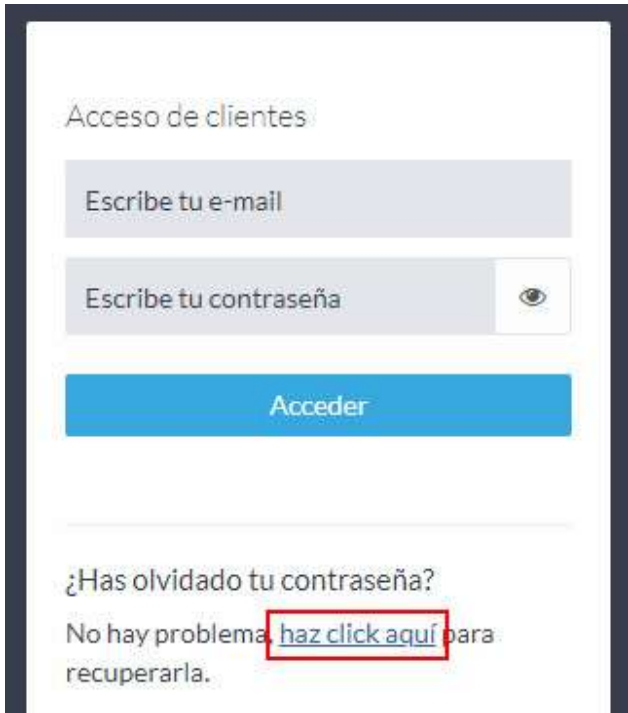
HU03- Cerrar Sesion Web	
Nombre de HU	Cerrar Sesión Web
Prioridad	Alta
HUEstado	Terrminada
<ul style="list-style-type: none"> Criterios de aceptación: 	<ul style="list-style-type: none"> El sistema debe cerrar la sesión activa del usuario de forma inmediata al hacer clic en "Cerrar sesión". El token o sesión del usuario debe invalidarse en el backend. El usuario debe ser redirigido automáticamente a la página de inicio o login. No debe permitir acceder a rutas protegidas tras cerrar sesión (aun si se usa el botón "Atrás" del navegador).
Desarrollado por:	Madeline Nathaly Flores Quispe
Prototipo: 	

HU04- Cerrar Sesión Movil	
Nombre de HU	Cerrar Sesión Web
Prioridad	Alta
HUEstado	Terminada
<ul style="list-style-type: none"> • Criterios de aceptación: 	<ul style="list-style-type: none"> • Al presionar el botón de cerrar sesión, se debe eliminar cualquier token o sesión almacenada localmente. • El usuario debe ser redirigido a la pantalla de login. • Las rutas protegidas de la app deben quedar bloqueadas tras cerrar sesión. • Se debe mostrar una notificación tipo toast confirmando la salida exitosa.
Desarrollado por:	Alejandro Melgar Arteaga
Prototipo: 	


HU05- Registrar usuario Web	
Nombre de HU	Registrar usuario Web
Prioridad	Alta
HUEstado	Terminada
<ul style="list-style-type: none"> Criterios de aceptación: 	<ul style="list-style-type: none"> El formulario de registro debe validar que todos los campos requeridos estén completos y sean correctos (email, contraseña, nombre, etc.). <input type="checkbox"/> No se debe permitir registros con correos ya existentes. Los datos deben almacenarse correctamente en la base de datos. El sistema debe notificar que el registro fue exitoso y redirigir al login o dashboard según el flujo. La contraseña debe almacenarse cifrada. quedar bloqueadas tras cerrar sesión. Se debe mostrar una notificación tipo toast confirmando la salida exitosa.
Desarrollado por:	Madeline Nathaly Flores Quispe
Prototipo: 	

HU06- Registrar usuario Movil	
Nombre de HU	Registrar usuario Movil
Prioridad	Alta
HUEstado	Terminada
<ul style="list-style-type: none"> Criterios de aceptación: 	<ul style="list-style-type: none"> El formulario debe permitir registrar nuevos usuarios desde la app móvil con validaciones adecuadas. Debe mostrar errores claros si hay problemas con los datos ingresados. El sistema debe conectarse con el backend y registrar los datos correctamente. El usuario debe recibir confirmación visual del registro exitoso (pantalla, toast o alerta). Se debe evitar duplicidad de registros con correos existentes.
Desarrollado por:	Alejandro Melgar Arteaga
Prototipo: 	


HU07-Recuperar Contraseña web	
Nombre de HU	Recuperar Contraseña web
Prioridad	Alta
HUEstado	Terminada
<ul style="list-style-type: none"> Criterios de aceptación: 	<ul style="list-style-type: none"> El usuario debe poder ingresar su correo electrónico registrado en un formulario de recuperación. El sistema debe validar que el correo ingresado exista en la base de datos. Si el correo es válido, se debe enviar un correo electrónico con un enlace o código para restablecer la contraseña. El enlace debe tener un tiempo de expiración definido (ej. 15 o 30 minutos). Al acceder al enlace, el usuario debe poder definir una nueva contraseña cumpliendo criterios de seguridad (longitud mínima, combinación de caracteres, etc.). El sistema debe confirmar que la contraseña fue actualizada correctamente y redirigir al login.
Desarrollado por:	Nathaly Madeline Flores Quispe
Prototipo: <div data-bbox="569 1294 989 1588" data-label="Image"> </div>	

HU08- Recuperar Contraseña movil	
Nombre de HU	Recuperar Contraseña Movil
Prioridad	Alta
HUEstado	Terminada
<ul style="list-style-type: none"> Criterios de aceptación: 	<ul style="list-style-type: none"> El usuario debe poder ingresar su correo en un campo específico desde la app móvil. El sistema debe validar si el correo existe en la base de datos y mostrar un mensaje adecuado si no existe. En caso válido, se debe enviar un correo o mensaje con un enlace o código para restablecer la contraseña. Desde la app, el usuario debe poder ingresar un nuevo password tras validar el código o acceder al enlace. La nueva contraseña debe ser enviada y actualizada correctamente en el backend. El usuario debe ser notificado del cambio exitoso y redirigido al login. Se debe evitar duplicidad de registros con correos existentes.
Desarrollado por:	Alejandro Melgar Arteaga
Prototipo: 	


HU09 – Gestionar usuarios (Web) – CU04

Nombre de HU	Gestionar usuarios Web																				
Prioridad	Alta																				
HU Estado	Terminada																				
Criterios de aceptación	<ul style="list-style-type: none">• Registrar, modificar, suspender y eliminar usuarios internos.• Validar unicidad de correo y nombre de usuario.• Solo roles autorizados pueden acceder.• Los cambios deben registrarse en la bitácora.• Notificación de creación o suspensión de usuarios.																				
Desarrollado por	Luis Fernando Banegas																				
Prototipo	 <p>El prototipo muestra una interfaz web titulada "Gestionar usuarios". En la parte superior, hay un botón azul con el texto "+ Agregar usuario". Debajo, se encuentra una tabla con las siguientes columnas: "Nombre", "Correo electrónico", "Rol" y "Estado". La tabla contiene cuatro filas de datos:</p> <table><tr><th>Nombre</th><th>Correo electrónico</th><th>Rol</th><th>Estado</th></tr><tr><td>Juan Pérez</td><td>juan.perez@correo.com</td><td>Administrador</td><td>Activo Editar</td></tr><tr><td>María González</td><td>maria.gonzalez@correo.com</td><td>Editor</td><td>Activo Editar</td></tr><tr><td>Carlos Rodríguez</td><td>carlos.rodriguez@correo.com</td><td>Editor</td><td>Activo Editar</td></tr><tr><td>Ana López</td><td>ana.lopez@correo.com</td><td>Inactivo</td><td>Inactivo Editar</td></tr></table>	Nombre	Correo electrónico	Rol	Estado	Juan Pérez	juan.perez@correo.com	Administrador	Activo Editar	María González	maria.gonzalez@correo.com	Editor	Activo Editar	Carlos Rodríguez	carlos.rodriguez@correo.com	Editor	Activo Editar	Ana López	ana.lopez@correo.com	Inactivo	Inactivo Editar
Nombre	Correo electrónico	Rol	Estado																		
Juan Pérez	juan.perez@correo.com	Administrador	Activo Editar																		
María González	maria.gonzalez@correo.com	Editor	Activo Editar																		
Carlos Rodríguez	carlos.rodriguez@correo.com	Editor	Activo Editar																		
Ana López	ana.lopez@correo.com	Inactivo	Inactivo Editar																		

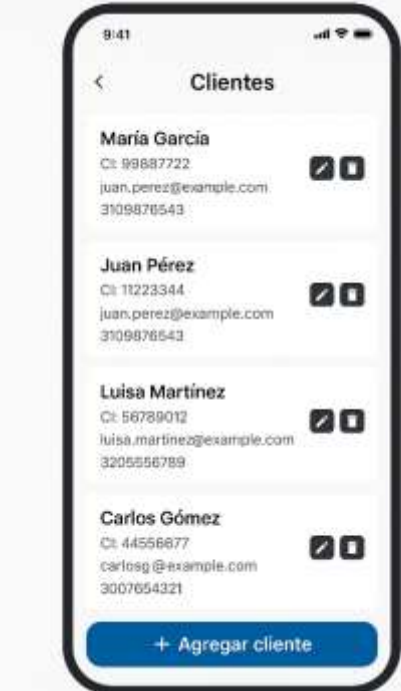
HU10 – Gestionar roles y permisos (Web) – CU05

Nombre de HU	Gestionar roles y permisos Web
Prioridad	Alta
HU Estado	Terminada
Criterios de aceptación	<ul style="list-style-type: none"> • Crear, editar y eliminar roles. • Asignar permisos específicos a cada rol. • No permitir eliminar el último rol administrador. • Cambios se reflejan de inmediato en accesos. • Registrar todas las modificaciones en la bitácora.
Desarrollado por	Daniela Pérez Gómez
Prototipo	 <p>El prototipo de la interfaz de usuario para 'Gestionar roles y permisos' se muestra en un recuadro gris. El título principal es 'Gestionar roles y permisos'. Debajo, hay una tabla con dos columnas: 'Rol' y 'Acciones'. La tabla contiene tres filas: 'Administrador', 'Editor' y 'Usuario'. Cada fila tiene un icono de lápiz y un icono de basurero. Debajo de la tabla, hay un botón azul que dice 'Agregar rol'. A la derecha de la tabla, hay una sección titulada 'Administrador' que contiene una subsección 'Permisos'. Esta sección tiene una lista de permisos con casillas de verificación: 'Ver usuarios' (marcada), 'Editar usuarios' (marcada), 'Ver roles' (marcada), 'Editar roles' (marcada), 'Ver clientes' (no marcada) y 'Editar clientes' (no marcada). Al final de esta sección, hay un botón azul que dice 'Guardar cambios'.</p>

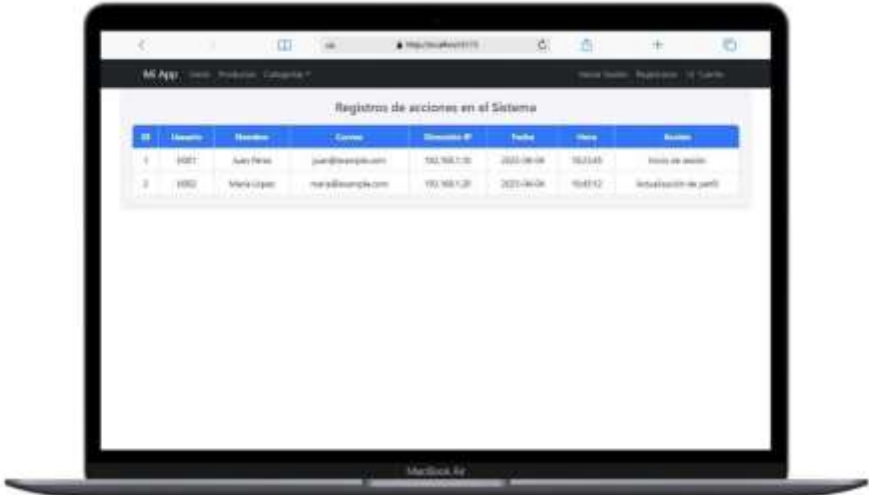
HU11 – Gestionar cliente (Web) – CU06

Nombre de HU	Gestionar cliente Web
Prioridad	Alta
HU Estado	Terminada
Criterios de aceptación	<ul style="list-style-type: none"> • Alta, modificación, eliminación y listado de clientes. • Validación de CI, correo y teléfono. • Acceso restringido por rol. • Cambios registrados en la bitácora. • Opción de búsqueda y filtros en la lista.
Desarrollado por	Luis Fernando Banegas
Prototipo	

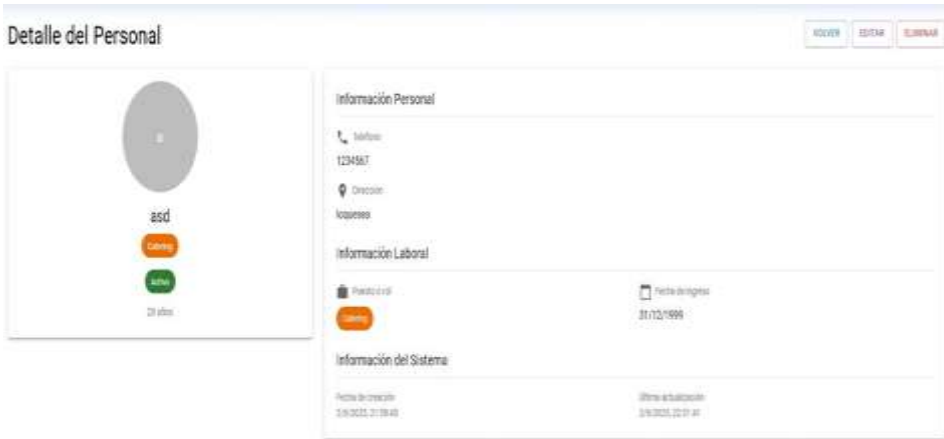
HU12 – Gestionar cliente (Móvil) – CU06

Nombre de HU	Gestionar cliente Móvil
Prioridad	Alta
HU Estado	Terminada
Criterios de aceptación	<ul style="list-style-type: none"> • Registro, edición y eliminación de clientes desde la app. • Formularios adaptados a móvil con validaciones. • Mostrar mensajes de error claros. • Cada cliente solo accede a su información. • Confirmación visual (toast o alerta) en acciones exitosas.
Desarrollado por	Daniela Pérez Gómez
Prototipo	 <p>El prototipo muestra una interfaz de usuario para una aplicación móvil. La pantalla principal, titulada 'Clientes', presenta una lista de cuatro clientes. Cada cliente se muestra con su nombre, número de celular, correo electrónico y un número de identificación único. A la derecha de cada entrada hay dos iconos: uno para editar y otro para eliminar. En la parte inferior de la pantalla hay un botón azul con el texto '+ Agregar cliente'.</p>

HU-13: Registrar Bitacora

HU-09 : Registrar Bitácora (Web)			
Caso de Uso:	CU09	Encargado:	Mendoza Luis
Prioridad:	Alta	PHU:	15
Descripción:	Registro de todas las acciones y usuarios en el sistema		
Características:	<ul style="list-style-type: none"> • Protección del sistema • Consultas sobre accesos al sistema • Verificación de las consultas del usuario • Historial de los movimientos 		
Prototipo:			

HU-14: Gestionar Personal

HU-10: Gestionar Personal (Web)			
Caso de Uso:	CU07	Encargado:	Mendoza Luis
Prioridad:	Alta	PHU:	12
Descripción:	Esta funcionalidad permitirá gestionar a los trabajadores proporcionando información al usuario del sistema.		
Características:	<ul style="list-style-type: none"> Se podrá registrar nuevos miembros del personal incluyendo datos como nombre, contacto, rol y disponibilidad. El sistema permitirá editar información del personal y controlar su estado (activo/inactivo). Se podrán eliminar registros, siempre que no estén asignados a tareas o eventos activos. 		
Prototipo:			

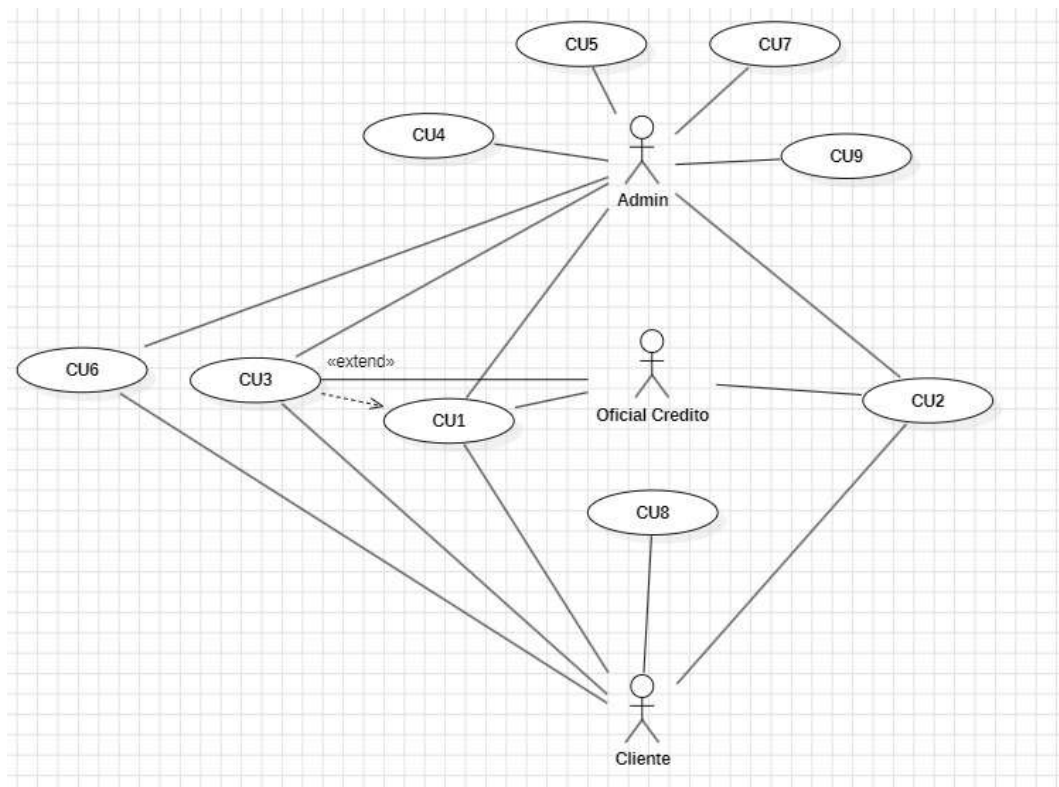
1.3. Contexto del sistema

El sistema de gestión de solicitudes de crédito con evaluación de solvencia mediante IA está diseñado para automatizar y optimizar el proceso de aprobación de créditos en entidades financieras de Bolivia. interactúan tres actores principales:

Clientes: solicitan créditos, cargan documentos y realizan pagos mediante la app móvil o web.

Oficiales de crédito: revisan, aprueban o rechazan solicitudes mediante la plataforma web.

Administradores: gestionan usuarios, roles, productos financieros y reportes.



1.4. Sprint backlog

Sprint Backlog	
Número de Sprint : 1	Tiempo de programación: 60h
Objetivo: Implementar funcionalidades principales de gestionar usuario, gestionar roles, inicio y cierre de sesión, diseñar la base de datos	
Fecha inicio: 02/09/2025	Fecha final: 14/09/2025

ID	Tarea	Tipo	Duración	Responsable	Estado
TP01	Crear repositorio en GitHub	gestión	1h	Luis Fernando Banegas	terminado
TP02	Instalar Django en entorno local	configuración	1h	Maikol Molina Cortez	terminado
TP03	Crear proyecto en Django	desarrollo	2h	Alejandro Melgar Arteaga	terminado
TP04	Hacer el diseño de la base de datos	diseño	2h	Madeline Flores Quispe	terminado
TP05	Configurar conexión con la base de datos	configuración	2h	Luis Pablo Mendoza	terminado
TP06	Crear migraciones a la base de datos	desarrollo	2h	Daniela Perez Gomez	terminado
TP07	Crear formulario de Registro web	frontend	3h	Luis Fernando Banegas	terminado
TP08	Crear formulario de login web	frontend	2h	Maikol Molina Cortez	terminado
TP09	Definir roles	diseño	1h	Madeline Flores Quispe	terminado

TP10	Validar inicio de sesión	desarrollo	2h	Alejandro Melgar Arteaga	terminado
TP11	Crear dashboard	frontend	2h	Daniela Perez Gomez	terminado
TP12	Crear el proyecto móvil	configuración	2h	Luis Pablo Mendoza	terminado
TP13	Diseñar pantalla de registro en Flutter	frontend	2h	Madeline Flores Quispe	terminado
TP14	Mostrar mensaje de 'login exitoso' en móvil	frontend	1h	Maikol Molina Cortez	terminado
TP15	Configurar despliegue en la nube	configuración	3h	Luis Fernando Banegas	terminado
TP16	Subir el backend en la nube	configuración	3h	Luis Pablo Mendoza	terminado
TP17	Probar el login y registro desde web y móvil	pruebas	2h	Daniela Perez Gomez	terminado

1.5. Equipo SCRUM

SPRINT 1	
Product Owner	❖ Banegas Roca Luis Fernando
Scrum Master	❖ Melgar Arteaga Alejandro
Team Developer	❖ Mendoza Rodríguez Luis Pablo ❖ Flores Quispe Nathaly Madeline ❖ Molina Cortez Maikol Anthony ❖ Perez Gomez Daniela

2. Proceso/patrón de desarrollo por Historia de Usuario

Sprint 1	
Product Owner	Banegas Roca Luis Fernando
Scrum Master	Melgar Arteaga Alejandro
Team Developer	<ul style="list-style-type: none">❖ Mendoza Rodríguez Luis Pablo❖ Flores Quispe Nathaly Madeline❖ Molina Cortez Maikol Anthony❖ Perez Gomez Daniela

2.1.Diseño

2.1.1. Diseñar de la arquitectura

Diagrama de Despliegue

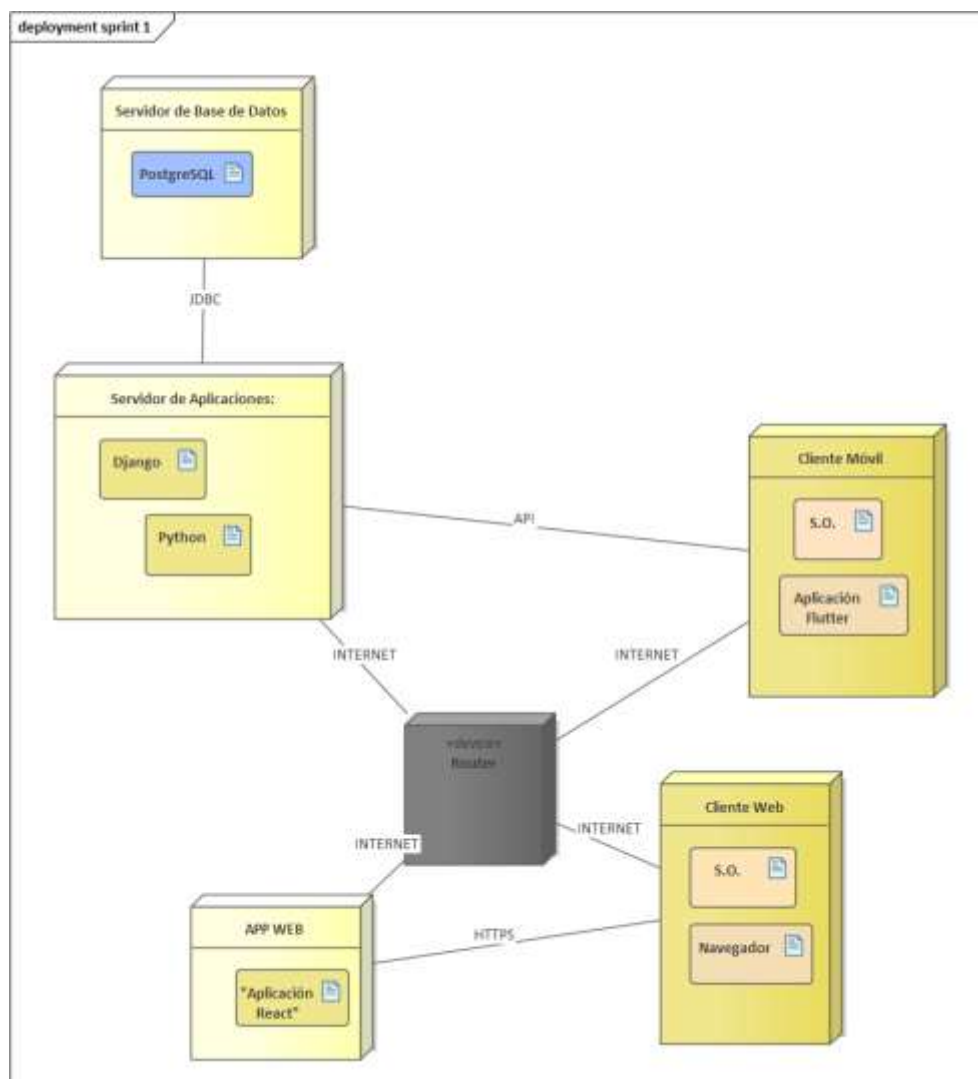
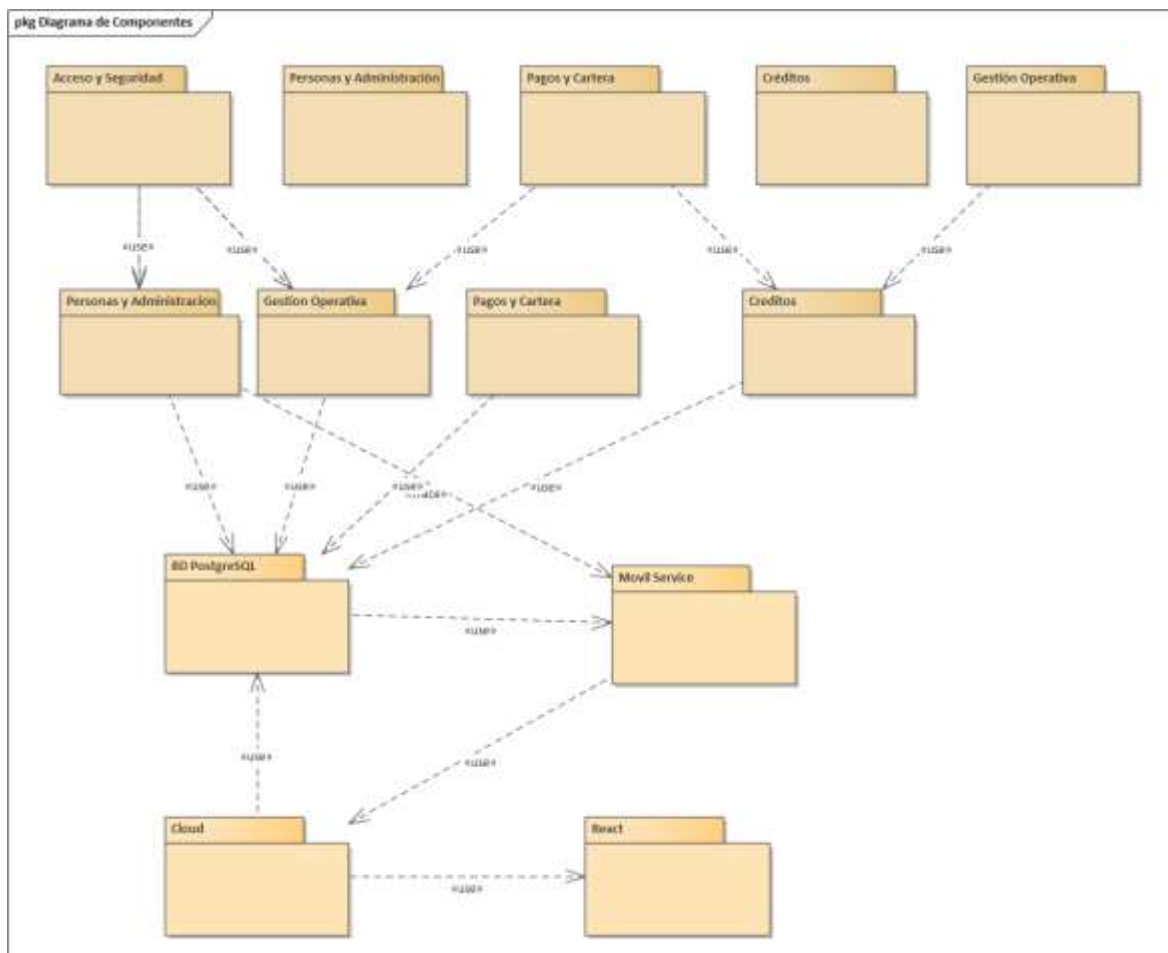
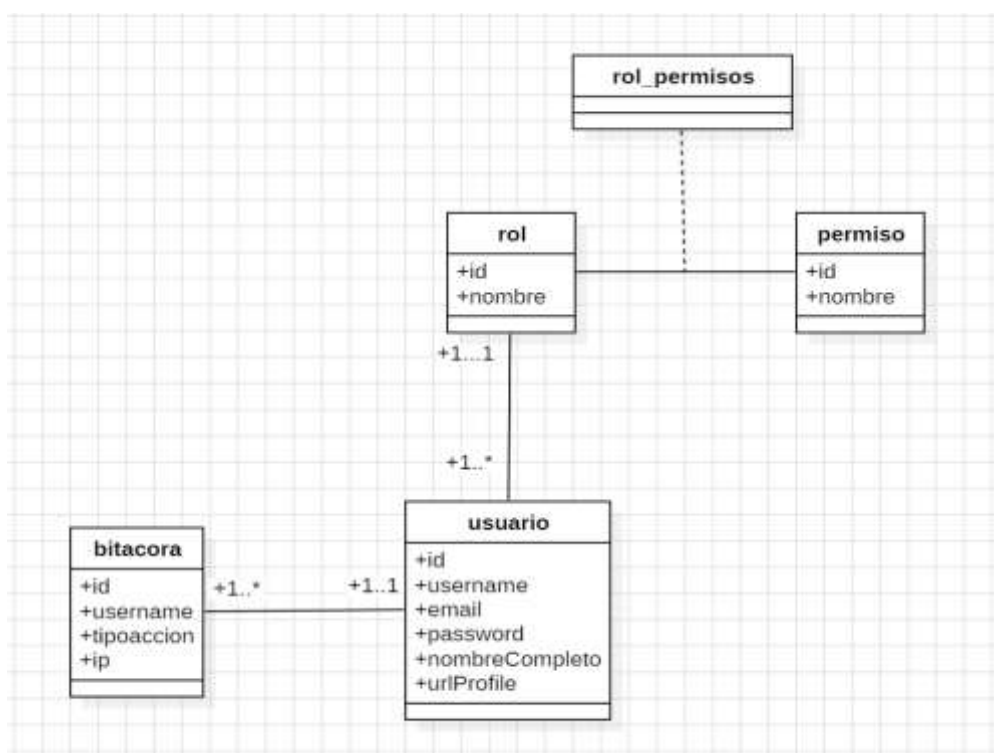


Diagrama Organizado por Capas



2.1.2. Diseño de datos



Mapeo

TABLA USUARIO							
PK				FK			
id	username	email	password	rol_id	created_id	updated_id	is_deleted

TABLA ROLES				
PK				
id	nombre	created_id	updated_id	is_deleted

TABLA PERMISOS				
PK				FK
id	nombre	created_id	updated_id	is_deleted

TABLA ROL_PERMISOS			
PK/FK	PK/FK		
rol_id	permiso_id	created_id	updated_id

TABLA BITACORA_USUARIOS				
PK				
id	ip	tipo_accion	created_at	update_at

Script

```

CREATE TABLE roles (
  id          SERIAL NOT NULL,
  nombre      VARCHAR(80) NOT NULL,
  created_at  TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),
  updated_at  TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),
  is_deleted  BOOLEAN NOT NULL DEFAULT FALSE,
  CONSTRAINT pk_roles PRIMARY KEY (id),
  CONSTRAINT uq_roles_nombre UNIQUE (nombre)
);

CREATE TABLE permisos (
  id          SERIAL NOT NULL,
  nombre      VARCHAR(80) NOT NULL,
  created_at  TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),
  updated_at  TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),
  is_deleted  BOOLEAN NOT NULL DEFAULT FALSE,
  CONSTRAINT pk_permisos PRIMARY KEY (id),
  CONSTRAINT uq_permisos_nombre UNIQUE (nombre)
);

CREATE TABLE usuarios (
  id          SERIAL NOT NULL,
  username    VARCHAR(80) NOT NULL,
  nombre      VARCHAR(80) NOT NULL,
  email       VARCHAR(120) NOT NULL,
  password    VARCHAR(200) NOT NULL,
  rol_id      INTEGER NOT NULL,
  created_at  TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),
  updated_at  TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),
  is_deleted  BOOLEAN NOT NULL DEFAULT FALSE,
  CONSTRAINT pk_usuarios PRIMARY KEY (id),
  CONSTRAINT uq_usuarios_username UNIQUE (username),
  CONSTRAINT uq_usuarios_email UNIQUE (email),

```

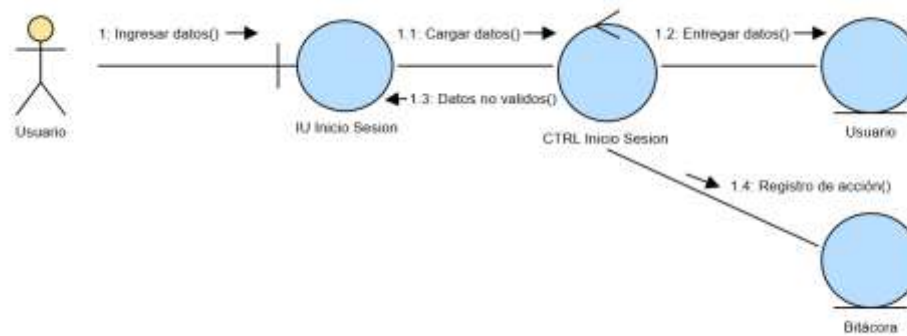
```

    CONSTRAINT fk_usuarios_rols
        FOREIGN KEY (rol_id) REFERENCES roles (id)
        ON UPDATE CASCADE ON DELETE RESTRICT
);
CREATE TABLE rol_permisos (
    rol_id          INTEGER NOT NULL,
    permiso_id      INTEGER NOT NULL,
    created_at      TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),
    updated_at      TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),
    is_deleted      BOOLEAN NOT NULL DEFAULT FALSE,
    CONSTRAINT pk_rol_permisos PRIMARY KEY (rol_id, permiso_id),
    CONSTRAINT fk_rol_permisos_rol
        FOREIGN KEY (rol_id) REFERENCES roles(id)
        ON UPDATE CASCADE ON DELETE CASCADE,
    CONSTRAINT fk_rol_permisos_permiso
        FOREIGN KEY (permiso_id) REFERENCES permisos(id)
        ON UPDATE CASCADE ON DELETE CASCADE
);
CREATE TABLE bitacora (
    id              SERIAL NOT NULL,
    usuario_id      INTEGER NOT NULL,
    tipoaccion      VARCHAR(50) NOT NULL,
    ip              VARCHAR(45),
    created_at      TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),
    updated_at      TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),
    is_deleted      BOOLEAN NOT NULL DEFAULT FALSE,
    CONSTRAINT pk_bitacora PRIMARY KEY (id),
    CONSTRAINT fk_bitacora_usuario
        FOREIGN KEY (usuario_id) REFERENCES usuarios(id)
        ON UPDATE CASCADE ON DELETE RESTRICT
);

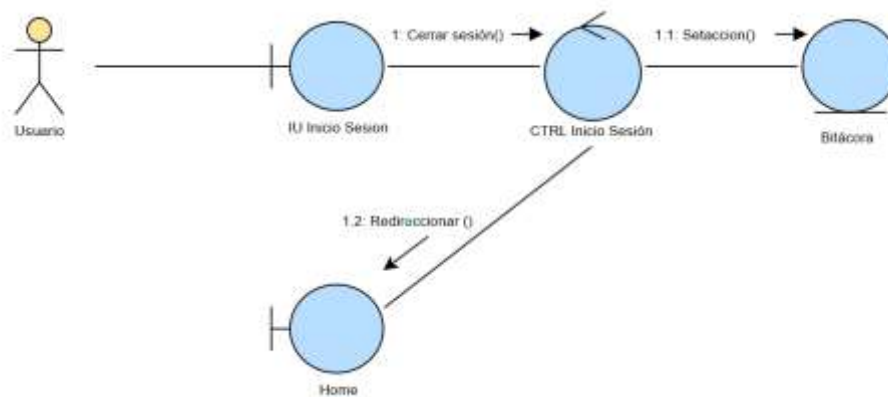
```

2.1.3. Diseño de la lógica de negocio

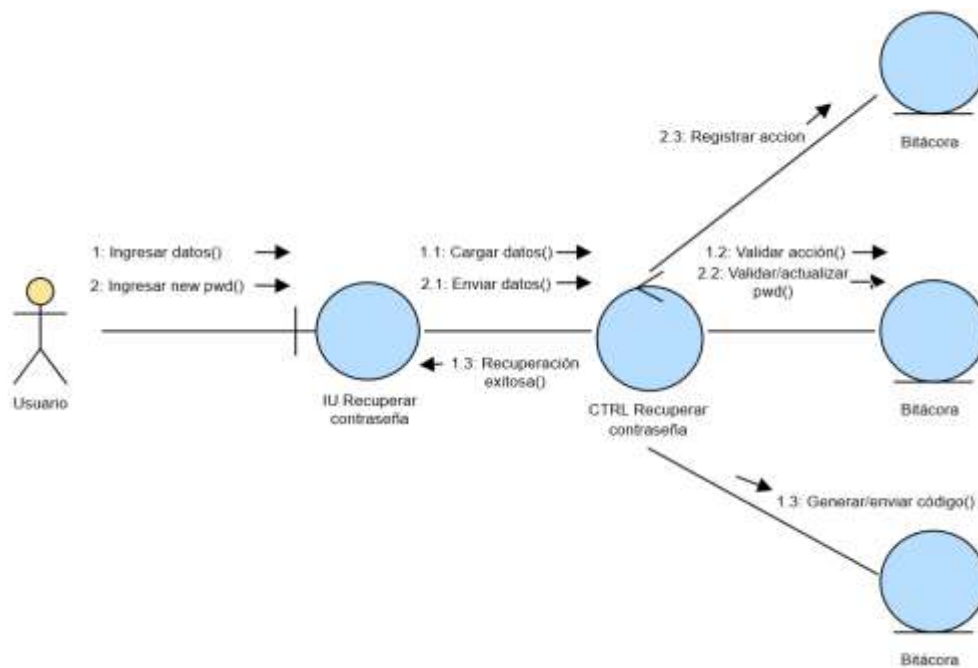
CU1 INICIO DE SESIÓN



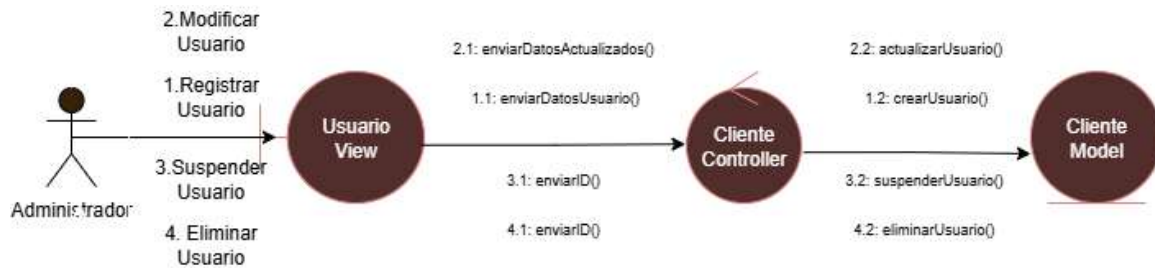
CU2 CERRAR SESIÓN



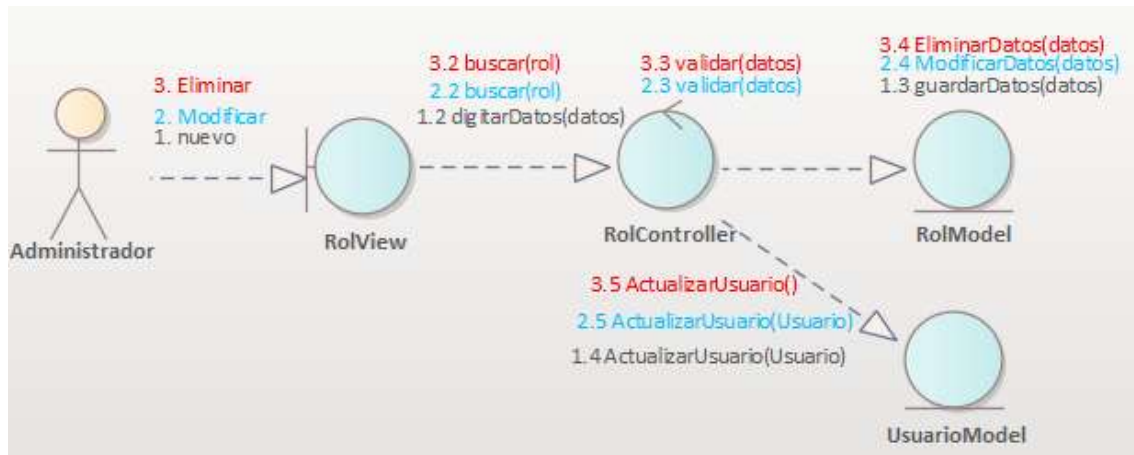
CU3 RECUPERAR CONTRASEÑA



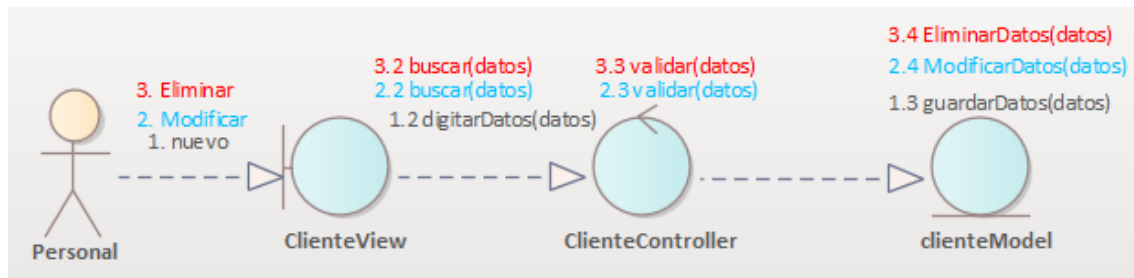
CU4 GESTIONAR USUARIOS



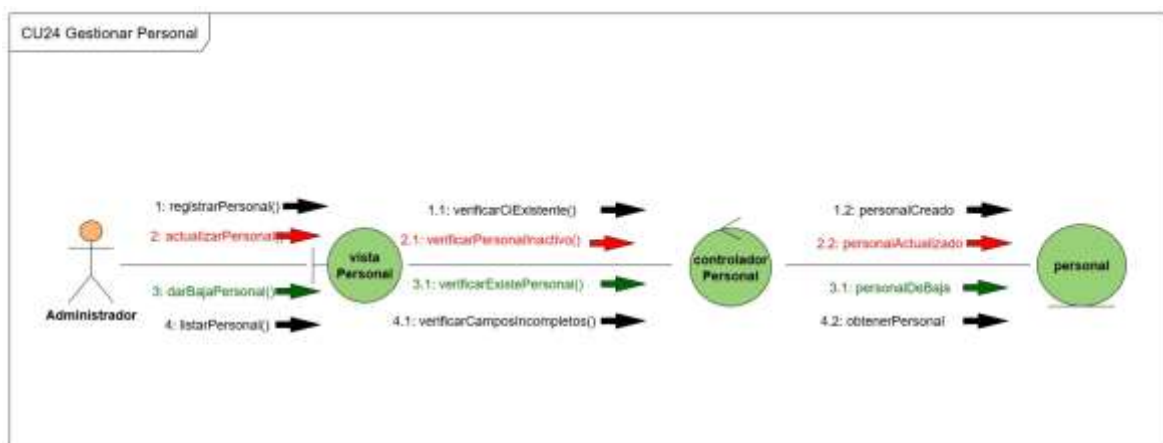
CU5 GESTIONAR ROLES Y PERMISOS



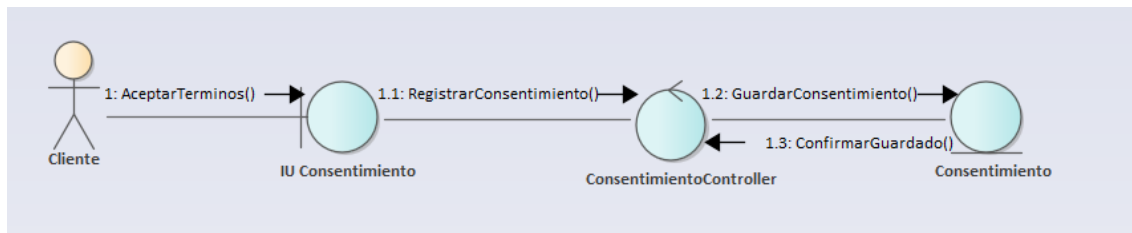
CU6 GESTIONAR CLIENTE



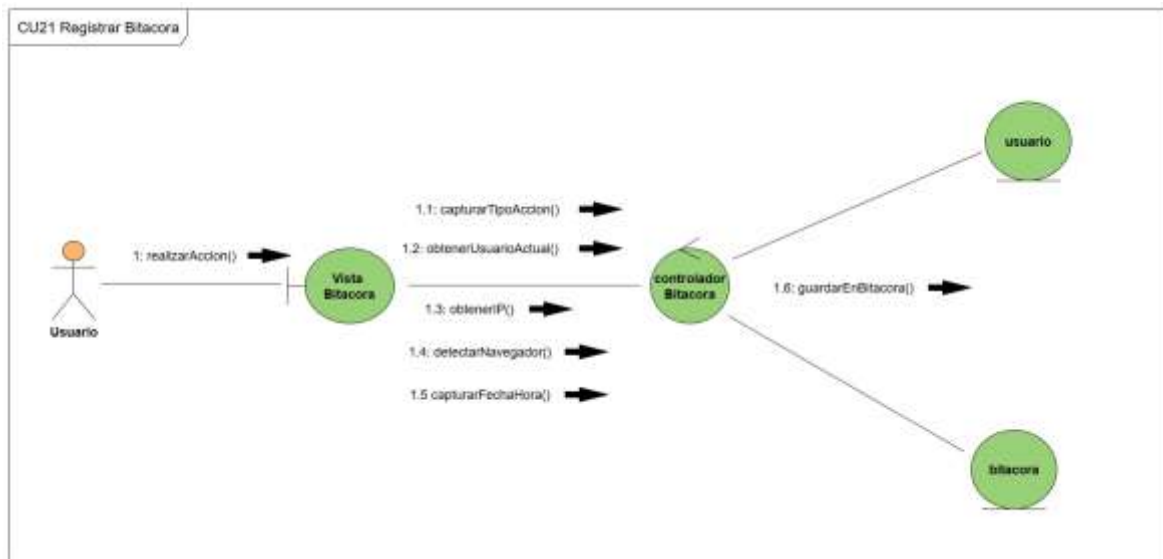
CU7 GESTIONAR PERSONAL



CU8: Consentimiento de tratamiento de datos (Web / Móvil)

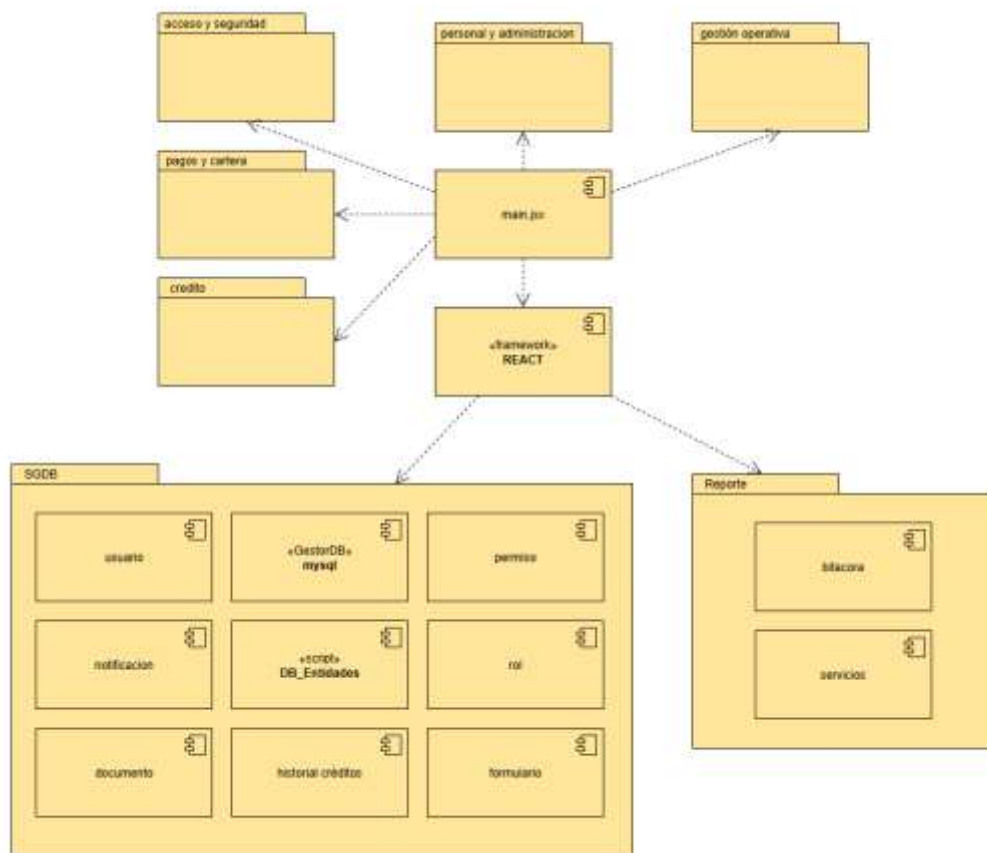


CU9 GESTIONAR BITACORA



2.2.Implementación

2.2.1. Componentes y artefactos generados



2.3.Pruebas

2.3.1. Plan de pruebas (criterios de aceptación)

CU01 Iniciar sesión (Web / Móvil)			
Nº	Criterio de aceptación	Tipo	Evidencia sugerida
1	El usuario con credenciales válidas accede al sistema y es redirigido al panel correspondiente a su rol.	Funcional	Captura pantalla / log de autenticación
2	El usuario con credenciales inválidas ve un mensaje claro y no accede.	UX/Funcional	Captura mensaje
3	Cuentas inactivas/bloqueadas no pueden iniciar sesión (mensaje específico).	Seguridad/Funcional	Captura mensaje / log
4	Tras N intentos fallidos consecutivos se bloquea la cuenta por T minutos.	Seguridad	Log de bloqueo / captura
5	Se registra en bitácora: intento exitoso/ fallido con timestamp y origen.	Auditoría	Registro en tabla/log

CU02 – Cerrar sesión (Web / Móvil)			
Nº	Criterio	Tipo	Evidencia
1	Al cerrar sesión se invalidan tokens/sesión y se redirige a pantalla pública.	Seguridad/Funcional	Captura flujo
2	No se puede acceder a áreas privadas usando el botón “Atrás” del navegador.	Seguridad	Comportamiento verificado
3	Se registra el evento de logout en bitácora.	Auditoría	Registro en tabla/log
4	En móvil, la app no mantiene sesión si el usuario cerró manualmente.	Seguridad/Funcional	Prueba en dispositivo

CU03 – Recuperar contraseña (Web / Móvil)			
Nº	Criterio	Tipo	Evidencia
1	El usuario puede solicitar recuperación ingresando email/identificador válido.	Funcional	Captura formulario
2	Se envía enlace/código de recuperación con expiración.	Seguridad	Log de envío / correo

3	El enlace/código expira correctamente o es de un solo uso.	Seguridad	Prueba de expiración
4	Tras restablecer, la nueva contraseña cumple políticas (longitud, complejidad).	Seguridad	Validaciones visibles
5	Se registra en bitácora solicitud y restablecimiento (sin exponer credenciales).	Auditoría	Registro en tabla/log

CU04 – Gestionar usuarios (Web)

Nº	Criterio	Tipo	Evidencia
1	Se puede crear, modificar, suspender y eliminar usuarios internos según permisos.	Funcional	Capturas CRUD
2	Validación de unicidad (email/usuario) y formatos.	Funcional	Mensajes de validación
3	Solo roles autorizados pueden acceder a este módulo.	Seguridad	Prueba con rol no autorizado
4	Cambios quedan auditados (quién, qué, cuándo, antes/después).	Auditoría	Registro histórico
5	Notificación opcional al usuario creado/suspendido.	Comunicación	Correo/log de envío

CU05 – Gestionar roles y permisos (Web)

Nº	Criterio	Tipo	Evidencia
1	Crear/editar/eliminar roles y asignar permisos granulares.	Funcional	Capturas
2	No se puede eliminar el último rol con permisos de administración.	Seguridad	Mensaje de bloqueo
3	Los cambios impactan inmediatamente en autorizaciones.	Funcional	Prueba de acceso antes/después
4	Toda modificación de permisos queda en bitácora.	Auditoría	Registro en tabla/log
5	La UI lista permisos de forma clara (legible y filtrable).	UX	Captura listados

CU06 – Gestionar cliente (Web / Móvil)

N ^o	Criterio	Tipo	Evidencia
1	Alta, modificación, eliminación y listado de clientes con validaciones (CI, email, teléfono).	Funcional	Capturas CRUD
2	En móvil, el alta/modificación respeta restricciones de campos y formatos.	UX/Funcional	Pruebas en app
3	Acceso restringido por rol; el cliente no ve datos de otros clientes.	Seguridad	Prueba con cliente/usuario
4	Cambios quedan auditados (quién, cuándo, qué).	Auditoría	Registros
5	Búsqueda y filtros por campos clave (CI, nombre, estado).	UX	Capturas filtros

CU07 – Gestionar personal (Web / Móvil)

N ^o	Criterio	Tipo	Evidencia
1	Alta, modificación, eliminación y listado de personal con validaciones.	Funcional	Capturas CRUD
2	Solo roles autorizados (admin/recursos) pueden gestionar personal.	Seguridad	Prueba acceso negado
3	Asignación de rol/perfil al personal se refleja en autorizaciones.	Funcional	Verificación inmediata
4	Historial de cambios en bitácora (antes/después).	Auditoría	Registro
5	Importación/exportación (opcional) con validación de formato.	Funcional	Archivo muestra

CU08 – Consentimiento de tratamiento de datos (Web / Móvil)

N ^o	Criterio	Tipo	Evidencia
1	Se muestra el texto de consentimiento (ASFI/LOPD) antes de usar datos personales.	Legal/UX	Captura
2	El usuario debe aceptar explícitamente (checkbox/firmar) para continuar.	Legal/Funcional	Captura acción
3	Se registra consentimiento (usuario, versión del texto, timestamp, canal).	Auditoría/Legal	Registro en BD
4	El usuario puede consultar la última versión aceptada.	UX	Captura
5	Si cambia la política, se solicita nueva aceptación en el siguiente ingreso.	Legal/Funcional	Flujo verificado

CU09 – Gestionar bitácora (Web / Móvil)

Nº	Criterio	Tipo	Evidencia
1	Se registran eventos clave: login/logout, CRUD usuarios/clientes/personal, cambios de permisos, consentimientos.	Auditoría	Vistas/logs
2	Cada registro contiene: actor, acción, entidad, valores relevantes, timestamp, IP/Device (si aplica).	Auditoría/Seguridad	Estructura de registro
3	La bitácora es consultable con filtros por fecha, usuario, acción y módulo.	UX/Funcional	Capturas filtros
4	La bitácora es solo lectura para roles no autorizados; no editable/borrable.	Seguridad	Prueba con rol no admin
5	Exportación de registros (CSV/PDF) preserva integridad y orden cronológico.	Funcional	Archivo exportado

Notas transversales de prueba (Sprint 1)

Nº	Nota
1	Acceso/Autorización: probar con cuentas de distintos roles (admin, oficial, auditor, cliente).
2	Seguridad: probar intentos de fuerza bruta (bloqueo), reutilización de tokens tras logout, acceso directo a URLs sin sesión.
3	UX: mensajes claros, validaciones en formularios, campos obligatorios, foco y navegabilidad.
4	Auditoría: verificar que <i>todas</i> las acciones relevantes quedan en bitácora con datos mínimos (quién, qué, cuándo).
5	Datos de prueba sugeridos: 3 usuarios internos (admin/oficial/auditor) + 2 clientes; emails válidos/ inválidos; contraseñas en el límite de políticas; CI correcto/incorrecto.
	Acceso/Autorización: probar con cuentas de distintos roles (admin, oficial, auditor, cliente).

2.3.2. Reporte de prueba

CU01 – Iniciar sesión (Web / Móvil)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-01	Login con credenciales válidas (admin)	Acceso exitoso y redirección al panel principal	Acceso permitido, panel admin cargado	Pass	Captura pantalla login
TC-02	Login con contraseña incorrecta	Mensaje “Credenciales inválidas” y denegación de acceso	Mensaje correcto mostrado, sin acceso	Pass	Captura mensaje
TC-03	3 intentos fallidos consecutivos	Cuenta bloqueada 15 min	Bloqueo correcto, aviso mostrado	Pass	Log de bloqueo
TC-04	Usuario inactivo intenta login	Mensaje “Cuenta inactiva”	Mensaje mostrado, acceso denegado	Pass	Captura mensaje

CU02 – Cerrar sesión (Web / Móvil)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-05	Usuario hace logout desde menú	Sesión invalidada, redirige a login	Logout correcto, redirigido a login	Pass	Captura flujo
TC-06	Intento de volver con botón “Atrás”	No accede a páginas privadas	Sistema pide reautenticación	Pass	Captura navegador
TC-07	Logout registrado en bitácora	Evento guardado con timestamp y usuario	Registro correcto en BD	Pass	Captura bitácora

CU03 – Recuperar contraseña (Web / Móvil)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-08	Solicitud de recuperación con email válido	Se envía enlace de recuperación	Correo recibido con link válido	Pass	Captura correo
TC-09	Link usado después de expirar	Mensaje “Enlace caducado”	Mensaje mostrado	Pass	Captura error

TC-10	Restablecimiento con nueva contraseña válida	Usuario puede loguear con nueva contraseña	Login exitoso con nueva clave	Pass	Captura login
-------	--	--	-------------------------------	------	---------------

CU04 – Gestionar usuarios (Web)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-11	Crear nuevo usuario interno	Usuario creado, aparece en listado	Registro correcto	Pass	Captura CRUD
TC-12	Editar datos de usuario	Datos actualizados en BD	Cambios reflejados en vista	Pass	Captura edición
TC-13	Eliminar usuario	Registro marcado como eliminado/inactivo	Usuario eliminado correctamente	Pass	Captura listado
TC-14	Acceso con rol no autorizado	Denegación de acceso	Acceso bloqueado	Pass	Mensaje de error

CU05 – Gestionar roles y permisos (Web)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-15	Crear nuevo rol con permisos	Rol guardado, permisos aplicables	Correcto, permisos reflejados	Pass	Captura roles
TC-16	Intentar eliminar rol admin único	Sistema impide acción	Mensaje de error mostrado	Pass	Captura validación
TC-17	Cambio de permisos a rol existente	Cambios aplicados a usuarios asignados	Accesos actualizados inmediatamente	Pass	Captura prueba

CU06 – Gestionar cliente (Web / Móvil)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-18	Registrar nuevo cliente (datos válidos)	Cliente guardado en BD	Correcto, aparece en listado	Pass	Captura alta
TC-19	Intentar registrar cliente con CI duplicado	Mensaje de error “CI ya registrado”	Validación correcta	Pass	Captura error
TC-20	Editar cliente existente	Cambios guardados en BD	Datos actualizados correctamente	Pass	Captura edición

CU07 – Gestionar personal (Web / Móvil)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-21	Alta de nuevo personal (rol asignado)	Personal creado y rol asignado	Registro correcto	Pass	Captura alta
TC-22	Eliminar personal	Personal marcado como inactivo	Correcto, aparece estado inactivo	Pass	Captura listado
TC-23	Cambiar rol a personal existente	Nuevo rol aplicado y reflejado en accesos	Correcto	Pass	Captura permisos

CU08 – Consentimiento tratamiento de datos (Web / Móvil)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-24	Mostrar texto de consentimiento en primer login	Usuario debe aceptar para continuar	Pantalla mostrada, aceptación registrada	Pass	Captura pantalla
TC-25	Usuario rechaza consentimiento	Acceso denegado hasta aceptar	Correcto, acceso bloqueado	Pass	Captura flujo
TC-26	Usuario revisa consentimiento aceptado	Se muestra versión y fecha aceptada	Información visible	Pass	Captura historial

CU09 – Gestionar bitácora (Web / Móvil)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-27	Registrar login/logout de usuarios	Evento guardado con datos básicos	Correcto, aparece en bitácora	Pass	Captura log
TC-28	Filtrar bitácora por fecha y usuario	Resultados coinciden con filtro	Correcto	Pass	Captura filtros
TC-29	Exportar bitácora en CSV	Archivo generado con registros íntegros	Correcto	Pass	Captura exportación
TC-30	Intentar borrar registro de bitácora	Acción no permitida, error mostrado	Correcto	Pass	Captura error

3. Daily Scrum (o Scrum diario)

Luis Fernando Banegas

Fecha	¿Qué hiciste ayer?	¿Qué hiciste hoy?	¿Qué obstáculo encontraste en el camino?
02/09	Nada	Crear entorno virtual y proyecto Django	Ninguna
03/09	Configurar base de datos PostgreSQL	Definir modelos iniciales	Ninguna
04/09	Definir lógica de login	Crear entorno virtual y proyecto Django	Ninguna
05/09	Diseñar estructura backend	Diseñar estructura backend	Ninguna
06/09	Definir lógica de login	Configurar CORS	Ninguna
07/09	Definir lógica de login	Definir modelos iniciales	Ninguna
08/09	Crear app 'accounts'	Probar conexión a BD	Ninguna
09/09	Probar login en backend	Registrar app en settings.py	Ninguna
10/09	Definir rutas iniciales	Registrar modelo Usuario	Ninguna
11/09	Hacer migraciones	Definir lógica de login	Ninguna
12/09	Crear superusuario	Registrar modelo Usuario	Ninguna
13/09	Crear app 'accounts'	Instalar dependencias necesarias	Ninguna
14/09	Definir modelos iniciales	Iniciar servidor con runserver	Ninguna

Madeline Flores Quispe

Fecha	¿Qué hiciste ayer?	¿Qué hiciste hoy?	¿Qué obstáculo encontraste en el camino?
02/09	Nada	Crear archivo .env	Ninguna
03/09	Probar login en backend	Crear superusuario	Ninguna
04/09	Definir lógica de login	Configurar CORS	Ninguna
05/09	Hacer migraciones	Definir modelos iniciales	Ninguna
06/09	Probar conexión a BD	Probar conexión a BD	Ninguna
07/09	Definir lógica de login	Definir modelos iniciales	Ninguna
08/09	Instalar dependencias necesarias	Definir lógica de login	Ninguna
09/09	Probar conexión a BD	Crear archivo .env	Ninguna
10/09	Registrar app en settings.py	Hacer migraciones	Ninguna
11/09	Instalar dependencias necesarias	Registrar app en settings.py	Ninguna
12/09	Definir lógica de login	Configurar CORS	Ninguna
13/09	Definir lógica de login	Definir modelos iniciales	Ninguna
14/09	Configurar CORS	Definir rutas iniciales	Ninguna

Alejandro Melgar Arteaga

Fecha	¿Qué hiciste ayer?	¿Qué hiciste hoy?	¿Qué obstáculo encontraste en el camino?
02/09	Nada	Definir modelos iniciales	Ninguna
03/09	Iniciar servidor con runserver	Definir lógica de login	Ninguna
04/09	Hacer migraciones	Probar conexión a BD	Ninguna
05/09	Configurar base de datos PostgreSQL	Probar login en backend	Ninguna
06/09	Definir rutas iniciales	Hacer migraciones	Ninguna
07/09	Crear app 'accounts'	Probar login en backend	Ninguna
08/09	Iniciar servidor con runserver	Registrar app en settings.py	Ninguna
09/09	Definir lógica de login	Definir lógica de login	Ninguna
10/09	Definir modelos iniciales	Hacer migraciones	Ninguna
11/09	Hacer migraciones	Crear entorno virtual y proyecto Django	Ninguna
12/09	Registrar app en settings.py	Configurar base de datos PostgreSQL	Ninguna
13/09	Diseñar estructura backend	Crear entorno virtual y proyecto Django	Ninguna
14/09	Registrar app en settings.py	Definir modelos iniciales	Ninguna

Luis Pablo Mendoza

Fecha	¿Qué hiciste ayer?	¿Qué hiciste hoy?	¿Qué obstáculo encontraste en el camino?
02/09	Nada	Hacer migraciones	Ninguna
03/09	Definir lógica de login	Crear app 'accounts'	Ninguna
04/09	Iniciar servidor con runserver	Probar conexión a BD	Ninguna
05/09	Iniciar servidor con runserver	Definir rutas iniciales	Ninguna
06/09	Crear archivo .env	Crear superusuario	Ninguna
07/09	Diseñar estructura backend	Registrar modelo Usuario	Ninguna
08/09	Iniciar servidor con runserver	Crear superusuario	Ninguna
09/09	Probar login en backend	Definir modelos iniciales	Ninguna
10/09	Definir rutas iniciales	Probar login en backend	Ninguna
11/09	Configurar CORS	Configurar base de datos PostgreSQL	Ninguna
12/09	Definir rutas iniciales	Crear app 'accounts'	Ninguna
13/09	Crear superusuario	Probar login en backend	Ninguna
14/09	Configurar CORS	Definir lógica de login	Ninguna

Maikol Molina Cortez

Fecha	¿Qué hiciste ayer?	¿Qué hiciste hoy?	¿Qué obstáculo encontraste en el camino?
02/09	Nada	Registrar modelo Usuario	Ninguna
03/09	Crear app 'accounts'	Crear superusuario	Ninguna
04/09	Diseñar estructura backend	Configurar CORS	Ninguna
05/09	Probar login en backend	Definir lógica de login	Ninguna
06/09	Definir lógica de login	Definir rutas iniciales	Ninguna
07/09	Definir lógica de login	Crear superusuario	Ninguna
08/09	Registrar app en settings.py	Iniciar servidor con runserver	Ninguna
09/09	Registrar app en settings.py	Registrar app en settings.py	Ninguna
10/09	Iniciar servidor con runserver	Definir lógica de login	Ninguna
11/09	Diseñar estructura backend	Probar login en backend	Ninguna
12/09	Registrar app en settings.py	Configurar CORS	Ninguna
13/09	Instalar dependencias necesarias	Crear archivo .env	Ninguna
14/09	Diseñar estructura backend	Crear app 'accounts'	Ninguna

Daniela Perez Gomez

Fecha	¿Qué hiciste ayer?	¿Qué hiciste hoy?	¿Qué obstáculo encontraste en el camino?
02/09	Nada	Probar login en backend	Ninguna
03/09	Definir rutas iniciales	Probar conexión a BD	Ninguna
04/09	Definir lógica de login	Definir modelos iniciales	Ninguna
05/09	Definir lógica de login	Registrar app en settings.py	Ninguna
06/09	Probar login en backend	Configurar base de datos PostgreSQL	Ninguna
07/09	Crear entorno virtual y proyecto Django	Probar conexión a BD	Ninguna
08/09	Definir modelos iniciales	Crear app 'accounts'	Ninguna
09/09	Crear superusuario	Configurar base de datos PostgreSQL	Ninguna
10/09	Definir modelos iniciales	Instalar dependencias necesarias	Ninguna
11/09	Registrar modelo Usuario	Registrar app en settings.py	Ninguna
12/09	Configurar CORS	Probar conexión a BD	Ninguna
13/09	Definir lógica de login	Definir lógica de login	Ninguna
14/09	Probar conexión a BD	Configurar CORS	Ninguna

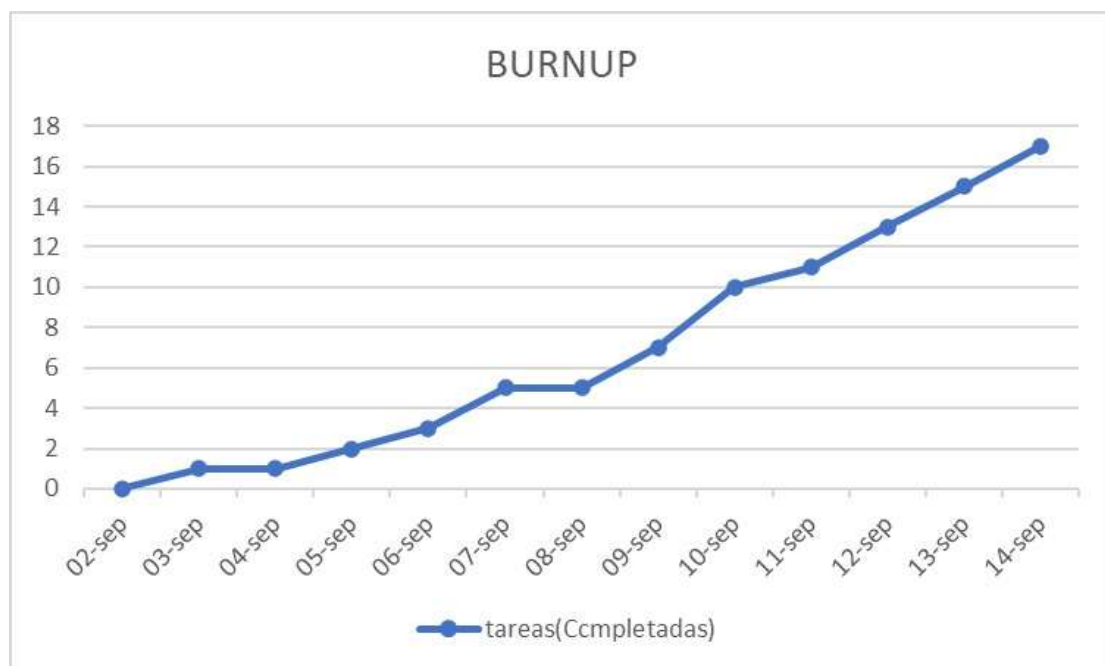
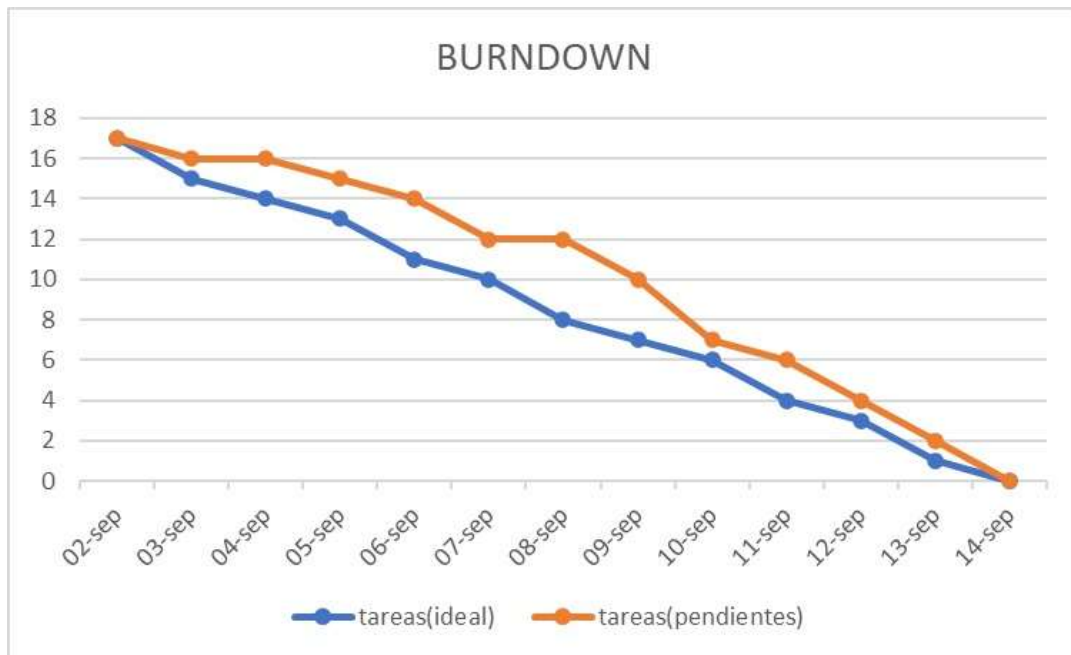
4. Sprint Review

REVISION DEL SPRING 1	
Objetivos del Sprint: <ul style="list-style-type: none"> • Crear la interfaz principal • Crear el acceso para el login de usuarios • Implementar todas las funcionalidades para la gestión de Usuario 	
PARTICIPANTES	
NOMBRE	ROL
Melgar Arteaga Alejandro	Scrum Master
Banegas Roca Luis Fernando	Product owner
Mendoza Rodríguez Luis Pablo	Team Development
Flores Quispe Nathaly Madeline	Team Development
Molina Cortez Maikol Anthony	Team Development
Perez Gomez Daniela	Team Development
PRESENTACION DEL INCREMENTO	
FUNCIONES PRESENTADAS (Elemento de trabajo a presentar)	RETROALIMENTACION
Diseñar la BD	Ninguna
Implementar la BD	Ninguna
Implementar inicio sesión	Ninguna
Implementar cerrar sesión	Ninguna
Implementar recuperar contraseña	Ninguna
Diseñar la interfaz general	Ninguna
Gestionar usuarios	Ninguna
Gestionar roles y permisos	Ninguna
Gestionar cliente	Ninguna
Gestionar personal	Ninguna
Gestionar bitácora	Ninguna

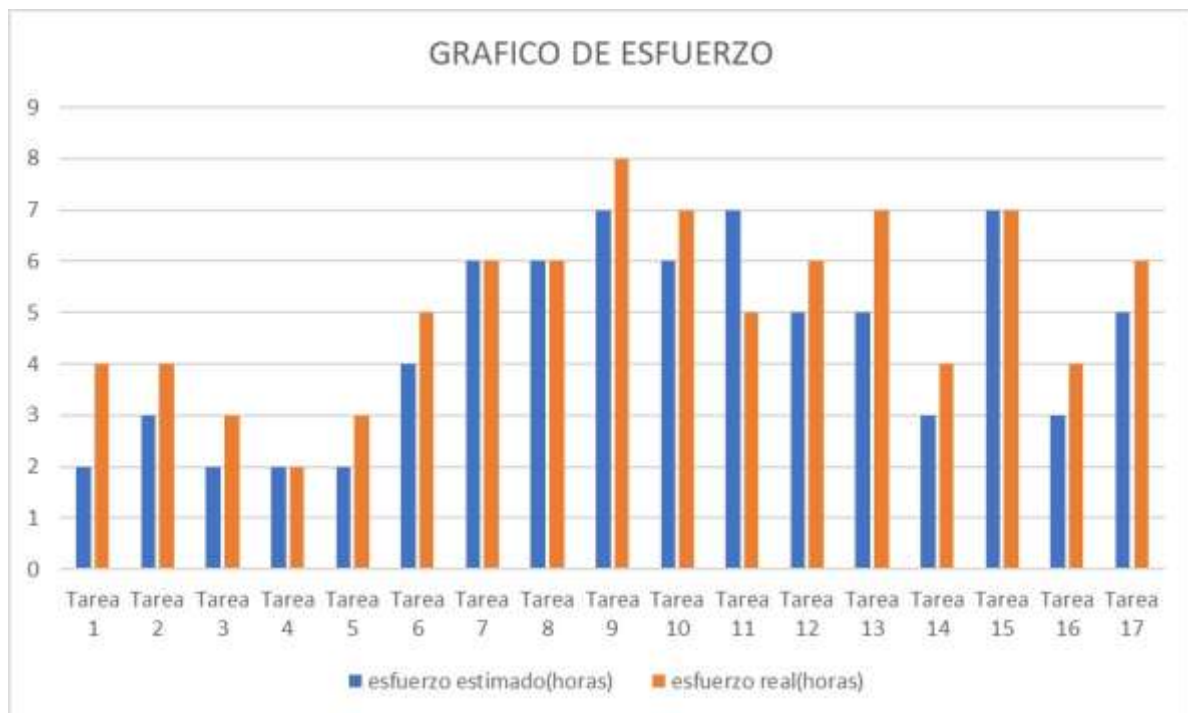
5. Sprint Retrospective

Fecha:			
Facilitador:		Banegas Roca Luis Fernando	
Objetivo		Evaluar el proceso del desarrollo del Sprint 1	
Nombre de Asistentes:		<ul style="list-style-type: none">Flores Quispe Nathaly MadelineMelgar Arteaga AlejandroMendoza Rodríguez Luis PabloMolina Cortez Maikol Anthony Pérez Gómez Daniela	
Temas a Tratar:		<ul style="list-style-type: none">Implementación de módulos de seguridad (login/logout, recuperación).Gestión de usuarios, roles, clientes y personal.Registro de consentimientos y bitácora. <ul style="list-style-type: none">Documentación y pruebas del Sprint 1.	
Discusión		<ul style="list-style-type: none">	
Nro.			
1	Se cumplieron los objetivos del sprint (login, usuarios, bitácora).	1	Se cumplieron los objetivos del sprint (login, usuarios, bitácora).
2	Buena colaboración entre los integrantes, se resolvieron dudas en equipo.	2	Buena colaboración entre los integrantes, se resolvieron dudas en equipo.
3	La documentación de criterios de aceptación y pruebas fue completa.	3	La documentación de criterios de aceptación y pruebas fue completa.
4	Los Daily Scrum fueron útiles para desbloquear tareas.	4	Los Daily Scrum fueron útiles para desbloquear tareas.
5	Se logró implementar recuperación de contraseña y consentimiento de datos.	5	Se logró implementar recuperación de contraseña y consentimiento de datos.
Fecha:	08/09/2025	Fecha:	08/09/2025

6. Burndown y BurnUp (Grafica de tareas y Datos de tareas)



7. Grafica de esfuerzo y Datos de esfuerzo



8. Scrum TaskBoard (Backlog, to do, doing, done)



SPRINT 2

1 Sprint Planning

1.1 Objetivos del Sprint

El objetivo principal de este segundo sprint es **construir y consolidar el núcleo funcional del proceso de solicitud y evaluación de créditos**, integrando los primeros componentes de análisis automático mediante IA. Este ciclo marca la transición del sistema desde la fase de configuración y acceso (Sprint 1) hacia la **operativa crediticia real**, permitiendo a los clientes registrar, simular y dar seguimiento a sus solicitudes dentro de un flujo controlado y transparente.

Durante este sprint, el equipo de desarrollo se enfocará en:

- Implementar el flujo completo de **solicitud de crédito**, desde la simulación hasta la aprobación o rechazo.
- Integrar las primeras **validaciones automáticas de información del cliente** mediante reglas y algoritmos de análisis de solvencia inicial.
- Incorporar la generación dinámica del **plan de pagos**, ajustado al monto, tasa y plazo seleccionado.
- Permitir la **visualización y seguimiento del estado de las solicitudes** y el **historial de créditos previos** del cliente.
- Establecer la gestión de **productos financieros** disponibles y el módulo de **documentación digital**, necesario para respaldar las solicitudes.

Casos de uso cubiertos en el Sprint 2:


- **CU10 – Solicitar crédito (Web / Móvil)**
- **CU11 – Simular crédito (Web / Móvil)**
- **CU12 – Registrar solicitud de crédito (Web)**
- **CU13 – Validar información (Web)**

- **CU14 – Aprobar / rechazar solicitud (Web)**
- **CU15 – Generar plan de pagos (Web)**
- **CU16 – Seguir estado de solicitud (Web / Móvil)**
- **CU17 – Ver historial de créditos (Web / Móvil)**
- **CU18 – Gestionar productos financieros (Web)**
- **CU19 – Cargar y gestionar documentos (Web / Móvil)**

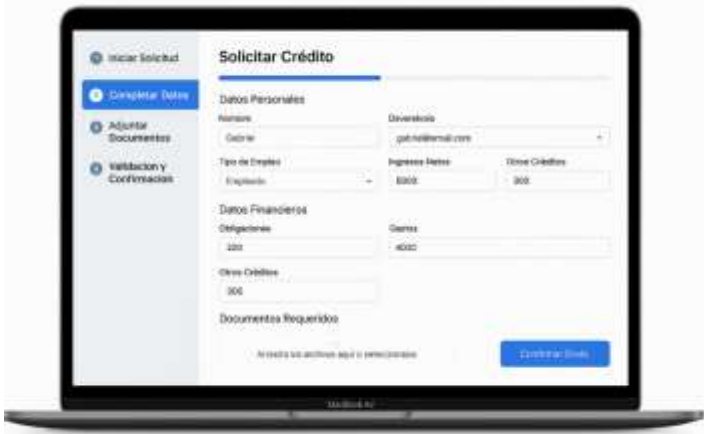
Con estas funcionalidades, se busca ofrecer una **experiencia fluida y confiable al usuario final**, reduciendo tiempos administrativos, mejorando la trazabilidad de cada solicitud y sentando las bases para la integración de modelos predictivos más avanzados en futuros sprints.

1.2 Historias de usuario (tarjetas 3C, Planning Póker, prototipos)


HU10 – Solicitar crédito (Móvil) – CU10

Nombre de HU	Solicitar crédito (Móvil)
Prioridad	Alta
HU Estado	En desarrollo
Criterios de aceptación	<ul style="list-style-type: none"> • El cliente puede iniciar una solicitud de crédito desde la app móvil. • El sistema muestra formularios dinámicos según el producto (monto, plazo, moneda, destino). • Permite registrar datos personales, laborales y financieros con validaciones de formato y rango. • El cliente puede guardar un borrador y completarlo posteriormente. • Se pueden adjuntar documentos requeridos (CI, comprobantes, extractos, etc.) con validación de tipo, tamaño y legibilidad. • El sistema valida la completitud y consistencia antes de permitir el envío. • Se genera un número de solicitud y estado inicial “Recibida” o “En revisión preliminar”. • El cliente recibe confirmación visual y notificación (correo, SMS o push) con el número de solicitud. • En caso de error o falta de conexión, se conserva el borrador local y se permite reintentar el envío. • Los mensajes de error deben ser claros (campos obligatorios, formato inválido, documento ilegible). • El cliente puede consultar el estado posterior de su solicitud y gestionar anexos adicionales según sea necesario.
Desarrollado por	Luis Mendoza
Prototipo	


HU10 – Solicitar crédito (Web) – CU10

Nombre de HU	Solicitar crédito (Web)
Prioridad	Alta
HU Estado	En desarrollo
Criterios de aceptación	<ul style="list-style-type: none"> • El cliente puede iniciar una solicitud de crédito desde la plataforma web. • El sistema muestra formularios dinámicos según el producto (monto, plazo, moneda, destino). • Permite registrar datos personales, laborales y financieros con validaciones de formato y rango. • El cliente puede guardar un borrador y completarlo posteriormente. • Se pueden adjuntar documentos requeridos (CI, comprobantes, extractos, etc.) con validación de tipo, tamaño y legibilidad. • El sistema valida la completitud y consistencia antes de permitir el envío. • Se genera un número de solicitud y estado inicial “Recibida” o “En revisión preliminar”. • El cliente recibe confirmación visual y notificación (correo, SMS o push) con el número de solicitud. • En caso de error o falta de conexión, se conserva el borrador local y se permite reintentar el envío. • Los mensajes de error deben ser claros (campos obligatorios, formato inválido, documento ilegible). • El cliente puede consultar el estado posterior de su solicitud y gestionar anexos adicionales según sea necesario.
Desarrollado por	Luis Mendoza
Prototipo	


HU11 – Simular crédito (Móvil/Web) – CU11

Nombre de HU	Simular crédito (Web)
Prioridad	Alta
HU Estado	En desarrollo
Criterios de aceptación	<ul style="list-style-type: none"> • El cliente puede acceder al módulo “Simular crédito” desde web o móvil. • El sistema permite seleccionar el producto, monto, moneda, plazo y frecuencia de pago. • El sistema obtiene y aplica tasas, topes y políticas vigentes para el producto elegido. • Se validan los rangos y formatos de todos los parámetros antes del cálculo. • El sistema calcula la cuota estimada, tasa efectiva anual (TEA/TNA), costo total (CFT/CAE) y un cronograma preliminar. • Los resultados se presentan de forma clara y visual, indicando que son “no vinculantes”. • El cliente puede modificar los parámetros (monto, plazo, frecuencia) y recalcular ilimitadamente. • Si el cliente está conforme, puede presionar “Continuar con la solicitud” para precargar los datos en el CU10 – Solicitar crédito. • En caso de error o falta de conexión, el sistema muestra mensaje claro y permite reintentar el cálculo. • Se gestionan correctamente las excepciones definidas (tasas no disponibles, parámetros inválidos, políticas no cumplidas, error de cálculo o traspaso fallido).
Desarrollado por	Luis Mendoza
Prototipo	


HU11 – Simular crédito (Móvil) – CU11

Nombre de HU	Simular crédito (Movil)
Prioridad	Alta
HU Estado	En desarrollo
Criterios de aceptación	<ul style="list-style-type: none"> • El cliente puede acceder al módulo “Simular crédito” desde web o móvil. • El sistema permite seleccionar el producto, monto, moneda, plazo y frecuencia de pago. • El sistema obtiene y aplica tasas, topes y políticas vigentes para el producto elegido. • Se validan los rangos y formatos de todos los parámetros antes del cálculo. • El sistema calcula la cuota estimada, tasa efectiva anual (TEA/TNA), costo total (CFT/CAE) y un cronograma preliminar. • Los resultados se presentan de forma clara y visual, indicando que son “no vinculantes”. • El cliente puede modificar los parámetros (monto, plazo, frecuencia) y recalcular ilimitadamente. • Si el cliente está conforme, puede presionar “Continuar con la solicitud” para precargar los datos en el CU10 – Solicitar crédito. • En caso de error o falta de conexión, el sistema muestra mensaje claro y permite reintentar el cálculo. • Se gestionan correctamente las excepciones definidas (tasas no disponibles, parámetros inválidos, políticas no cumplidas, error de cálculo o traspaso fallido).
Desarrollado por	Luis Mendoza
Prototipo	

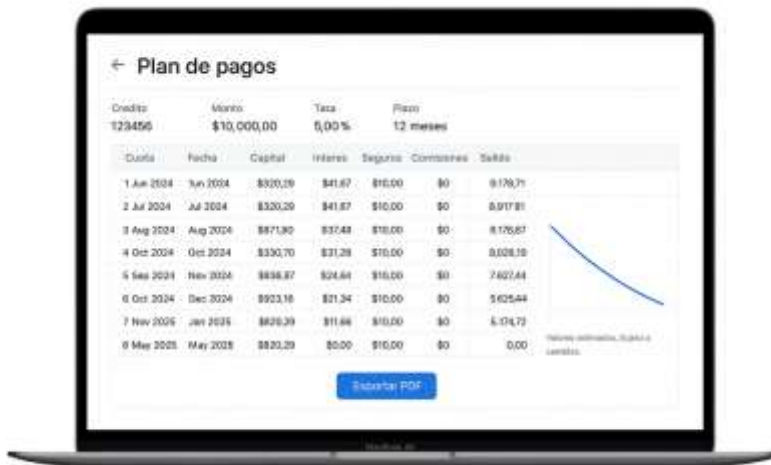
CU12 - Registrar solicitud de crédito (Web)

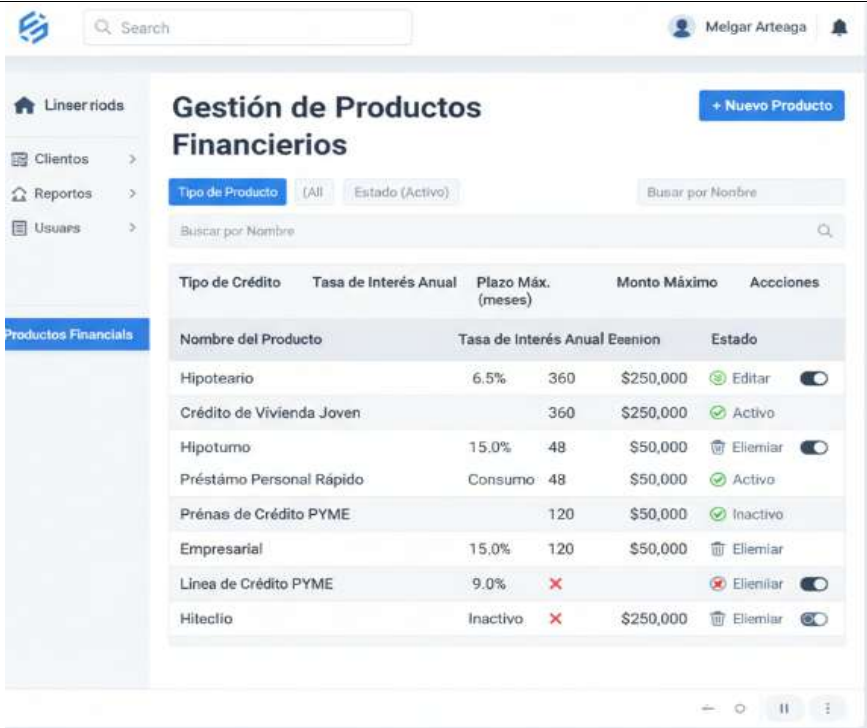
Nombre de HU	Registrar solicitud de crédito (Web)
Prioridad	Alta
HU Estado	En desarrollo
Criterios de aceptación	<ul style="list-style-type: none"> • El Oficial accede al módulo “Registrar solicitud de crédito”. • Se muestran formularios con campos obligatorios y opcionales para datos personales, laborales/ingresos, financieros y producto/condiciones (monto, plazo, tasa, modalidad). • El sistema valida formato y rangos (p. ej., ingresos vs. cuota estimada) antes de permitir avanzar. • Se listan y se pueden adjuntar documentos requeridos según producto (CI, comprobantes, extractos, etc.). • Los archivos cumplen tipo/tamaño/legibilidad; si no, se muestran errores y se impide continuar hasta corregir. • Al confirmar, el sistema genera número de solicitud y la deja con estado inicial “Registrada”. • La solicitud y los vínculos a documentos quedan persistidos en base de datos con trazabilidad (fecha, oficial, cliente). • Se notifica al Cliente (email/SMS) con el número de solicitud; si falla la notificación, se permite reenvío manual sin afectar el registro. • Se verifica: cliente existente (<<include>> CU06) y consentimiento vigente (<<include>> CU08); si falta alguno, se bloquea y se indica cómo resolver. • Mensajes de error claros y accionables para datos inconsistentes, documentos inválidos, validación fallida o errores de persistencia. • Si hay documentos inválidos tras el registro, el sistema permite reemplazarlos (<<extend>> CU19). • El Cliente puede consultar el estado y recibir notificaciones de avance (<<extend>> CU16).
Desarrollado por	Maikol Molina
Prototipo	

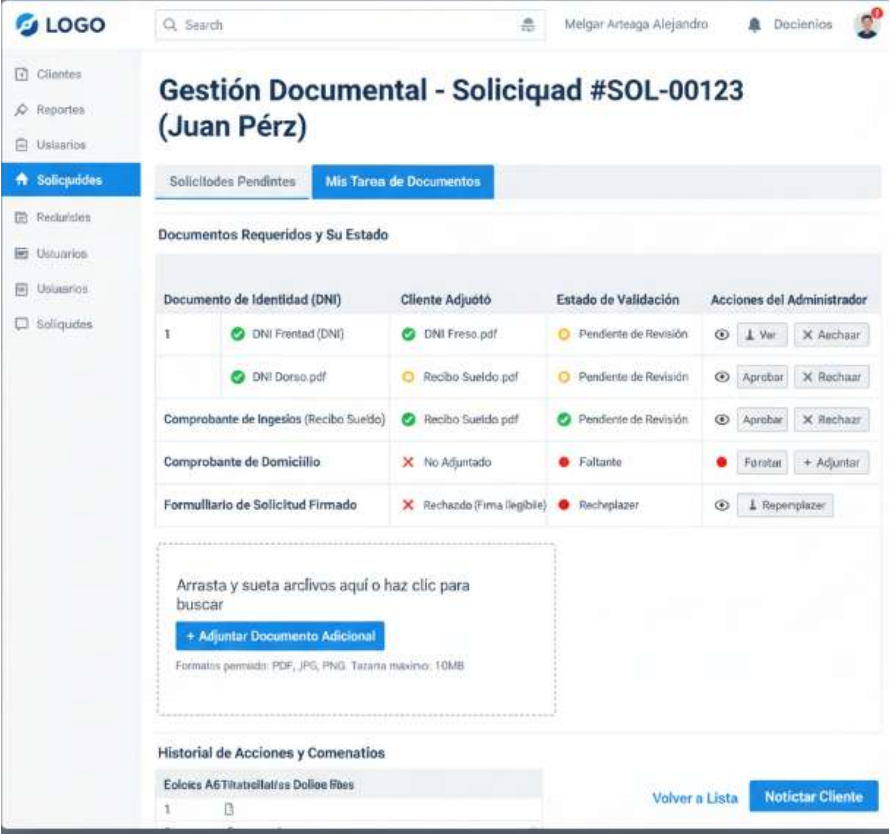
CU13 - Validar información (Web)

Nombre de HU	Validar datos y documentos de la solicitud (Automática/Manual)
Prioridad	Alta
HU Estado	En desarrollo
Criterios de aceptación	<ul style="list-style-type: none"> • Al pasar la solicitud a “En validación”, el sistema inicia validaciones automáticas: verificación de identidad (RENIEC/SPEN u organismo local), consulta a buró de crédito (score, deudas activas) y validación de documentos (autenticidad, vigencia, legibilidad). • Si hay alertas, el Oficial puede revisarlas manualmente, marcar documentos sospechosos, adjuntar observaciones y confirmar o rechazar su validez. • Resultado de la validación actualiza el estado de la solicitud: “Validada”, “Rechazada” (con motivo) o “Observada” (solicita aclaración al cliente; <<extend>> CU26). • Las integraciones externas tienen reintentos configurados (3) y manejo de indisponibilidad con mensaje claro al Oficial y registro en bitácora (<<include>> CU09). • Todos los pasos (automáticos y manuales) se registran en bitácora con fecha, usuario/sistema, acción, fuente externa consultada y resultado (éxito/fallo). • El sistema notifica a Cliente y Oficial el resultado (éxito/fallo de envío con opción de reenvío). • Si el resultado es “Validada”, queda habilitado iniciar evaluación/scoring (<<extend>> CU20). • Mensajes de error claros y accionables para: sistema externo no disponible, documentos ilegibles/falsos, información contradictoria, error de notificación.
Desarrollado por	Maikol Molina
Prototipo	

HU15 – Generar plan de pagos (Web) – CU15


Nombre de HU	Generar plan de pagos (Web)
Prioridad	Alta
HU Estado	En desarrollo
Criterios de aceptación	<ul style="list-style-type: none"> • El cliente puede ingresar al módulo “Plan de pagos” desde la plataforma web. • El sistema permite seleccionar un crédito aprobado o una simulación existente. • Se obtienen automáticamente los parámetros del crédito (monto, tasa, plazo, frecuencia, fecha de inicio). • El sistema calcula el cronograma aplicando la fórmula financiera correspondiente (francesa, alemana u otra). • Se muestra el plan con detalle de cada cuota: número, fecha, capital, interés, seguros, comisiones y saldo restante. • El cliente puede visualizar el cronograma completo en tabla o gráfico, y exportarlo en PDF o Excel. • Se incluyen advertencias (“valores estimados”, “sujeto a cambios”). • Ante errores de cálculo o datos inválidos, el sistema informa con mensajes claros y permite reintentar. • El plan generado puede guardarse y vincularse al crédito correspondiente. • El sistema muestra una constancia con fecha y número de crédito al finalizar.
Desarrollado por	Fernando Banegas
Prototipo	

HU-18 :Gestionar Productos Financieros (Web)			
Caso de Uso:	CU18	Encargado:	Melgar Arteaga Alejandro
Prioridad:	Alta	PHU:	20
Descripción:	Gestionar los productos financieros para configurar los tipos de crédito, tasas y condiciones.		
Características:	<ul style="list-style-type: none"> • Crear, editar y eliminar productos financieros • Configurar tasas de interés y plazos • Establecer montos mínimos y máximos • Activar/desactivar productos 		
Prototipo:			

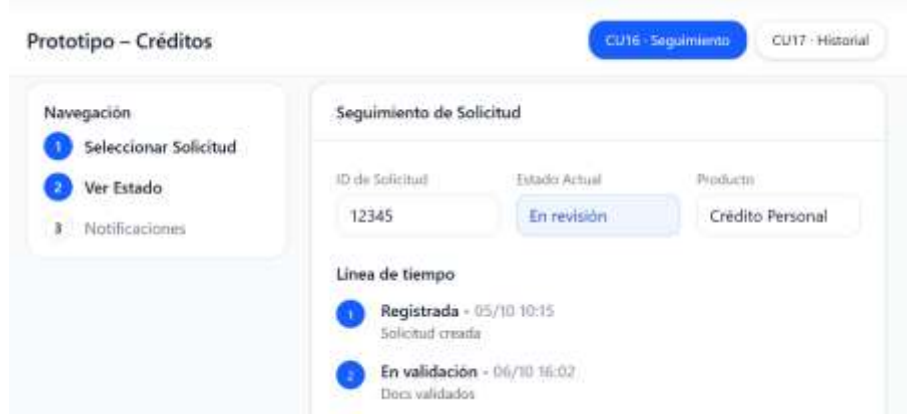
HU-19:Cargar y Gestionar Documentos (Web)			
Caso de Uso:	CU19	Encargado:	Melgar Arteaga Alejandro
Prioridad:	Alta	PHU:	16
Descripción:	Es la parte donde se puede cargar los documentos requeridos y también poder gestionarlos con el rol de administrador, para las solicitudes de los clientes.		
Características:	<ul style="list-style-type: none"> • Adjuntar archivos desde dispositivo • Validación de formatos permitidos • Verificación de vigencia documental • Visualización de documentos cargados • Reemplazar documentos rechazados 		
Prototipo:			

HU-20: Cargar y Gestionar Documentos (movil)			
Caso de Uso:	CU19	Encargado:	Melgar Arteaga Alejandro
Prioridad:	Alta	PHU:	16
Descripción:	Es la parte donde se puede cargar los documentos requeridos y también poder gestionarlos con el rol de administrador, para las solicitudes de los clientes.		
Características:	<ul style="list-style-type: none"> • Adjuntar archivos desde dispositivo • Validación de formatos permitidos • Verificación de vigencia documental • Visualización de documentos cargados • Reemplazar documentos rechazados 		
Prototipo:			


HU15 – Generar plan de pagos (Móvil) – CU15

Nombre de HU	Generar plan de pagos (Móvil)
Prioridad	Alta
HU Estado	En desarrollo
Criterios de aceptación	<ul style="list-style-type: none"> • El cliente puede acceder al módulo “Plan de pagos” desde la aplicación móvil. • El sistema muestra una interfaz optimizada para dispositivos móviles, manteniendo las mismas funcionalidades del entorno web. • Permite seleccionar un crédito aprobado o simulación para calcular el cronograma. • El sistema obtiene automáticamente los parámetros del crédito (monto, tasa, plazo, frecuencia, fecha de inicio). • Calcula y muestra las cuotas con su desglose (capital, interés, seguros, comisiones). • El cliente puede desplazarse fácilmente entre las cuotas y ver gráficos simplificados del saldo. • Puede descargar o compartir el plan de pagos en PDF directamente desde el móvil. • El sistema valida datos y muestra mensajes claros ante errores de cálculo o falta de conexión. • Permite guardar el plan generado y consultarlo posteriormente desde el historial de créditos. • Muestra constancia final con fecha y número de crédito generado.
Desarrollado por	Fernando Banegas
Prototipo	

HU16 – Seguir estado de solicitud (Web) – CU16

Nombre de HU	Seguir estado de solicitud (Web)
Prioridad	Alta
HU Estado	En desarrollo
Criterios de aceptación	<ul style="list-style-type: none"> • Cliente autenticado accede al módulo “Seguimiento” en la web. • El front invoca GET /seguimiento/{solicitudId} con JWT y recibe un DTO Seguimiento: estadoActual, timelineEstados[] {fecha, estado, observaciones, responsable}, notificaciones[] {fecha_envio, titulo, mensaje, leído}. • La UI muestra estado vigente, línea de tiempo de cambios y avisos recientes (push/email/in-app). • Se aplican reglas de negocio: catálogo de estados (registrada, en_validación, en_revisión, aprobada, rechazada, cancelada); toda transición genera registro en historial y dispara notificación. • Seguridad: solo el dueño puede ver su solicitud (RBAC + verificación de pertenencia). • Paginación del historial/notificaciones (20–50 ítems) y filtros por rango de fechas/estado. • Manejo de alternos/excepciones: 403 si no pertenece; 404 si no existe; error en servicio de notificaciones no bloquea la consulta; se registra en bitácora. • Rendimiento: respuesta ≤ 800 ms (P95); auditoría de accesos en bitácora; soporte i18n de estados/mensajes.
Desarrollado por	Daniela Perez
Prototipo	

HU16 – Seguir estado de solicitud (Móvil) – CU16

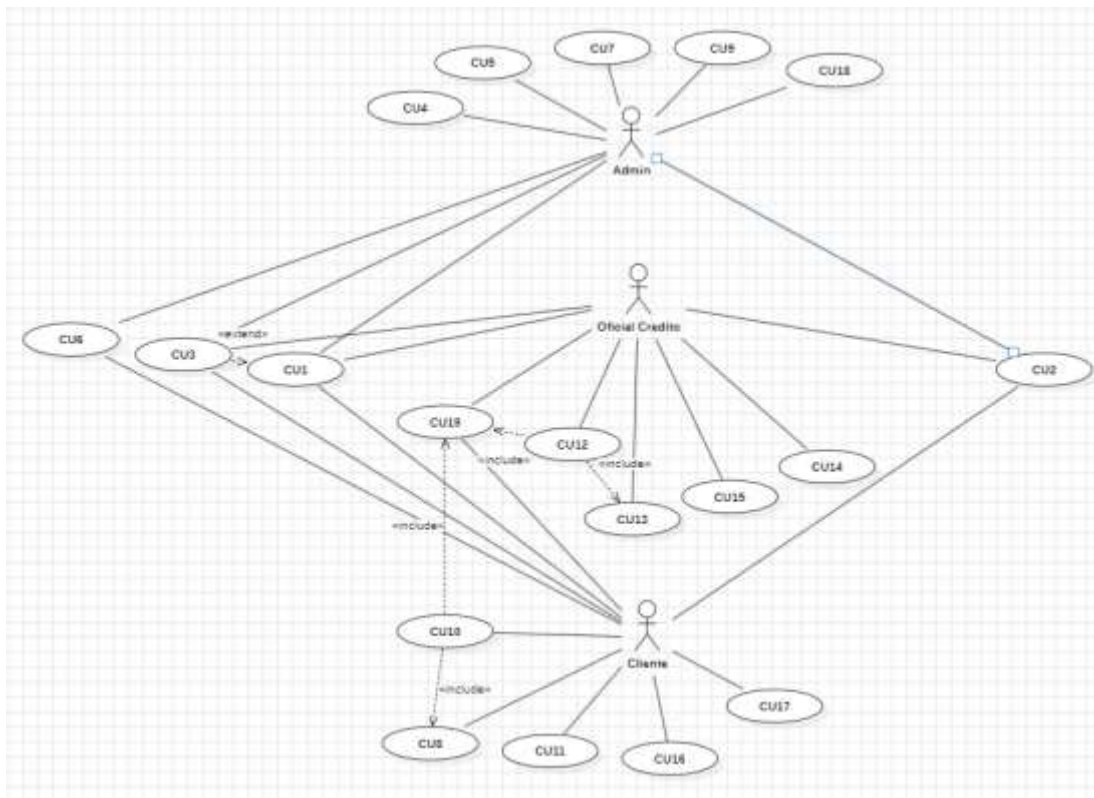
Nombre de HU	Seguir estado de solicitud (Móvil)
Prioridad	Alta
HU Estado	En desarrollo
Criterios de aceptación	<ul style="list-style-type: none"> • Cliente autenticado accede a “Seguimiento” en la app móvil (UI adaptada). • La app consume GET /seguimiento/{solicitudId} con JWT y renderiza estadoActual, timeline scrollable y notificaciones con badges de leído/no leído. • Recibe push notifications cuando cambia el estado; al tocar la notificación, navega a la pantalla de seguimiento de esa solicitud. • Seguridad y RBAC idénticos a web; bloqueo de acceso si la solicitud no pertenece al cliente (403). • Paginación infinita (lazy load) para historial/notificaciones; control de offline (cache de última vista) y reintento si falla la red. • Manejo de alternos/excepciones: 404 si no existe; fallas del servicio de notificaciones se registran pero no bloquean la vista; mensajes claros y accionables. • Rendimiento: primera carga ≤ 800 ms (P95) con optimizaciones móviles (memoización y compresión); soporte i18n e iconografía por estado.
Desarrollado por	Daniela Perez
Prototipo	

1.3 Contexto del sistema

El sistema de gestión de solicitudes de crédito con evaluación de solvencia mediante IA continúa evolucionando en este Sprint 2 hacia una **plataforma operativa completa**, donde se combinan la automatización, el análisis inteligente y la interacción en tiempo real entre clientes y oficiales de crédito.

En este contexto, el sistema amplía sus capacidades para cubrir todo el ciclo de vida de una solicitud crediticia, desde su creación hasta su resolución. Los actores mantienen su rol principal, pero se incorporan nuevas interacciones:

- **Clientes:** pueden **simular, registrar y dar seguimiento** a sus solicitudes, así como **adjuntar documentos** y consultar su historial de créditos desde la aplicación web o móvil.
- **Oficiales de crédito:** analizan la información recibida, aplican criterios de validación, y realizan la **aprobación o rechazo** de solicitudes. Además, pueden gestionar productos financieros y generar planes de pago adaptados a cada caso.
- **Administradores:** supervisan el correcto funcionamiento del sistema, gestionan productos financieros disponibles y garantizan la integridad de los procesos mediante el control de usuarios y roles.
- **Módulo de IA inicial:** ejecuta validaciones automáticas y genera indicadores preliminares de solvencia, basados en la información y documentos proporcionados por el cliente.
- **Sistema de notificaciones:** comunica a los clientes los cambios de estado de su solicitud en tiempo real (push, email o in-app).



1.4 Sprint backlog

Sprint Backlog	
Número de Sprint : 2	Tiempo de programación estimado: 120h
Objetivo: Implementar funcionalidades mas detalladas y especificas, para arrancar el a desarrollar el núcleo principal de la parte de programación del workflow.	
Fecha inicio: 19/09/2025	Fecha final: 12/10/2025

ID	Tarea	Tipo	Duración	Responsable	Estado
TP01	Diseñar interfaz de usuario (UI)	diseño	4h	Madeline Flores Quispe	en progreso
TP02	Crear modelos de usuarios	desarrollo	3h	Alejandro Melgar Arteaga	terminado
TP03	Configurar autenticación de usuarios	configuración	2h	Luis Pablo Mendoza	terminado
TP04	Implementar login y logout	desarrollo	3h	Daniela Pérez Gomez	en progreso
TP05	Diseñar formulario de registro	diseño	2h	Madeline Flores Quispe	pendiente
TP06	Crear vistas para gestión de perfiles	desarrollo	4h	Maikol Molina Cortez	terminado
TP07	Configurar envío de emails	configuración	3h	Luis Fernando Banegas	pendiente
TP08	Implementar recuperación de contraseña	desarrollo	3h	Alejandro Melgar Arteaga	en progreso
TP09	Crear panel de administración	desarrollo	5h	Daniela Pérez Gomez	pendiente
TP10	Diseñar base de datos para posts	diseño	3h	Madeline Flores Quispe	terminado
TP11	Implementar CRUD de posts	desarrollo	6h	Luis Pablo Mendoza	en progreso

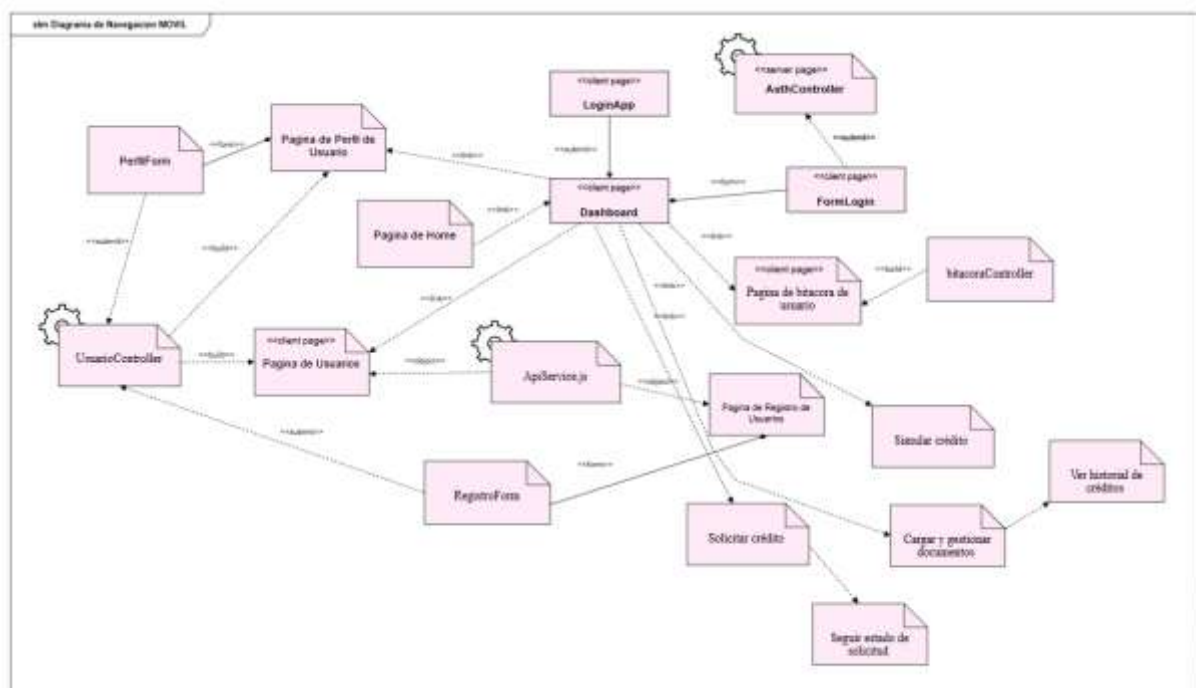
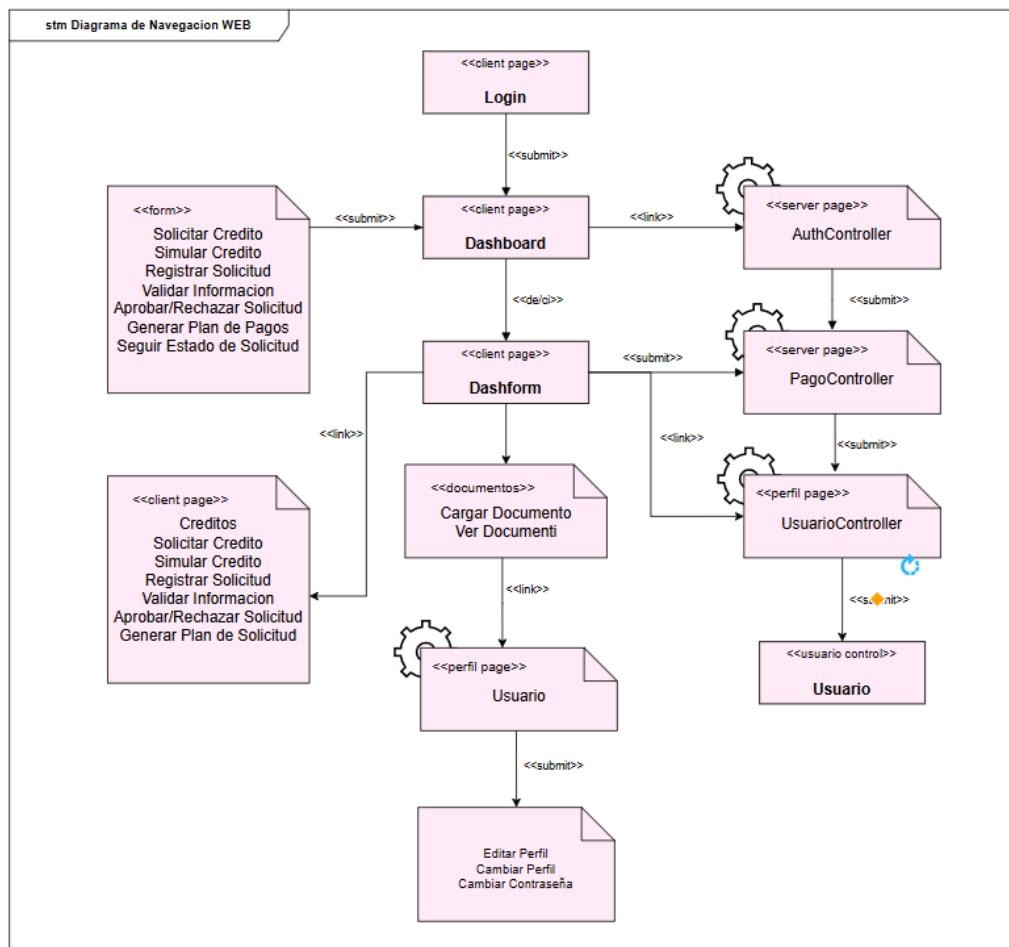
TP12	Crear templates para listar posts	desarrollo	4h	Maikol Molina Cortez	terminado
TP13	Integrar Bootstrap para estilos	diseño	3h	Madeline Flores Quispe	en progreso
TP14	Configurar rutas y URLs	desarrollo	2h	Alejandro Melgar Arteaga	terminado
TP15	Realizar pruebas unitarias	testing	4h	Daniela Pérez Gomez	pendiente
TP16	Revisión de código (Code Review)	gestión	2h	Luis Fernando Banegas	pendiente
TP17	Despliegue en entorno de pruebas	configuración	3h	Luis Pablo Mendoza	en progreso
TP18	Documentación técnica	documentación	3h	Maikol Molina Cortez	pendiente
TP19	Optimización de consultas BD	desarrollo	3h	Alejandro Melgar Arteaga	terminado
TP20	Implementar paginación en listas	desarrollo	2h	Daniela Pérez Gomez	en progreso
TP21	Diseñar iconos y elementos gráficos	diseño	4h	Madeline Flores Quispe	pendiente
TP22	Reunión de revisión de sprint	gestión	1h	Luis Fernando Banegas	terminado

1.5 Equipo SCRUM

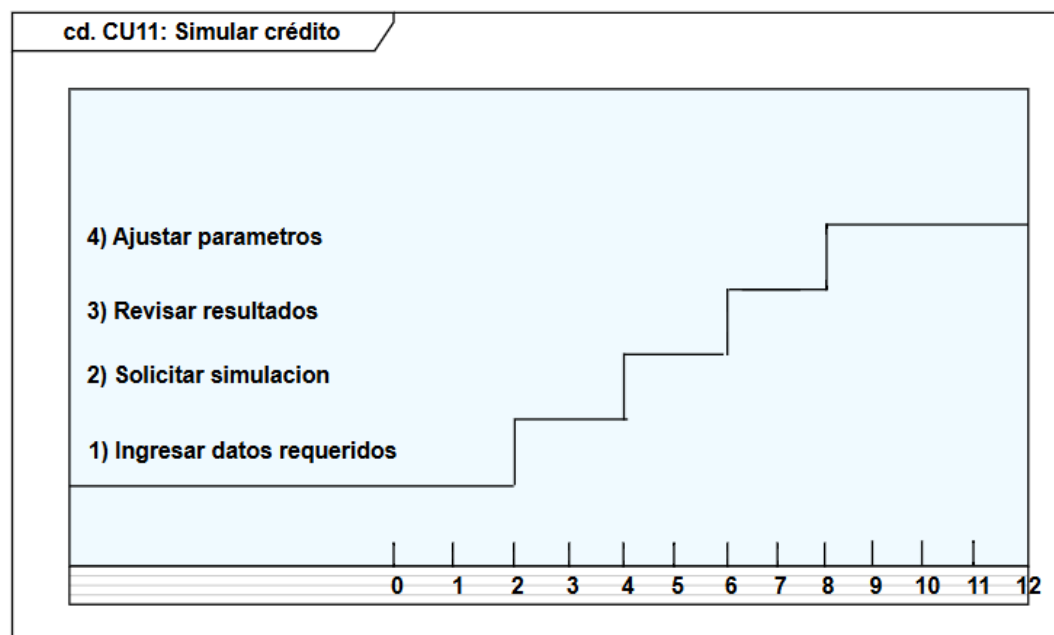
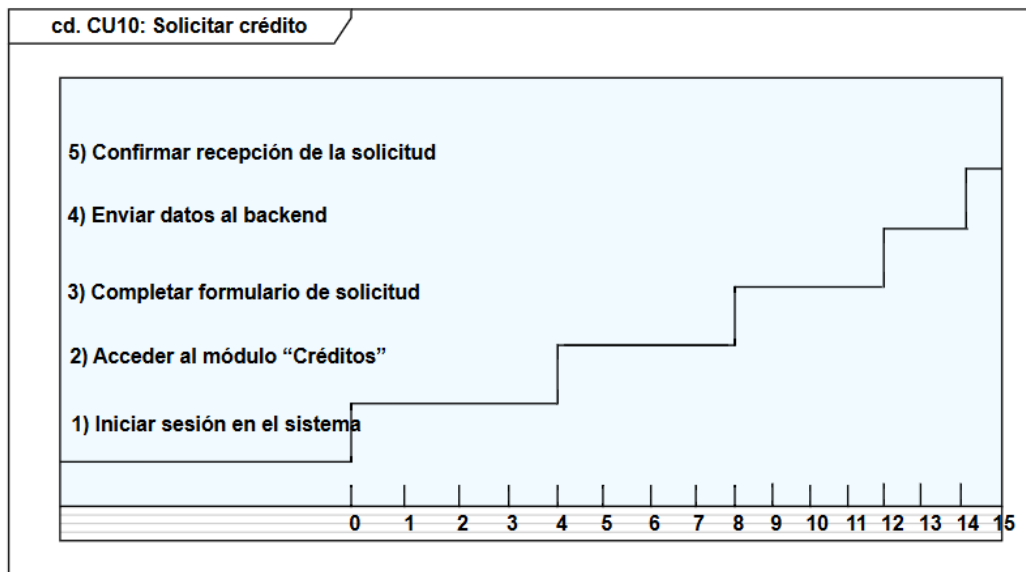
Sprint 2	
Product Owner	Molina Cortez Maikol Anthony
Scrum Master	Perez Gomez Daniela
Team Developer	<ul style="list-style-type: none"> ❖ Mendoza Rodríguez Luis Pablo ❖ Flores Quispe Nathaly Madeline ❖ Banegas Roca Luis Fernando ❖ Melgar Arteaga Alejandro

1.6 UML 2.5+

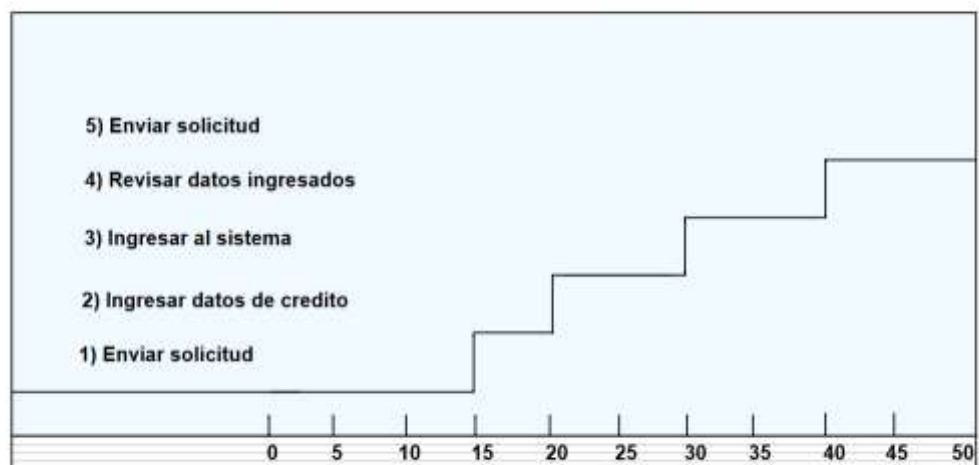
1.6.1 Diagrama de Navegación



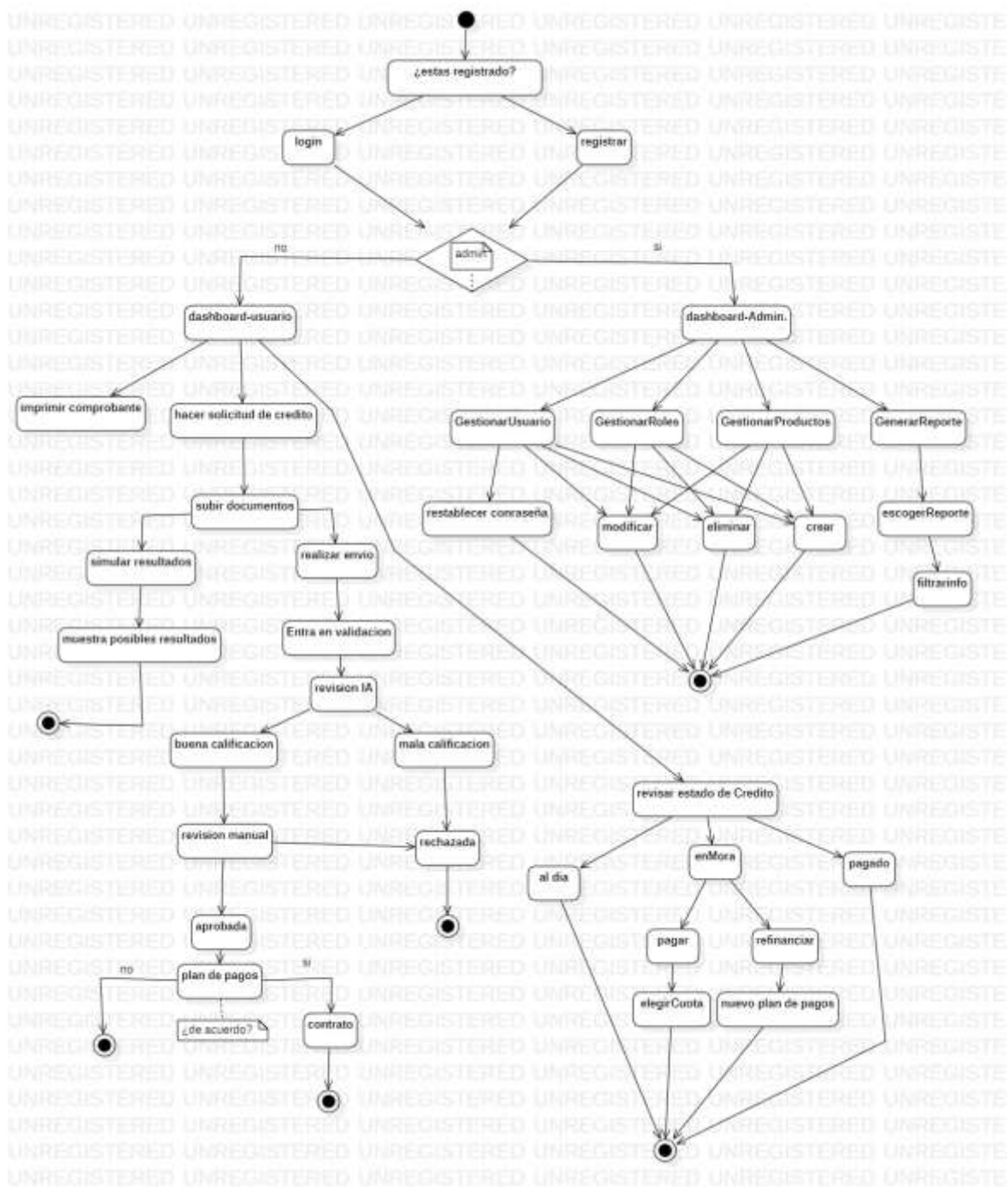
1.6.2 Diagrama de tiempo

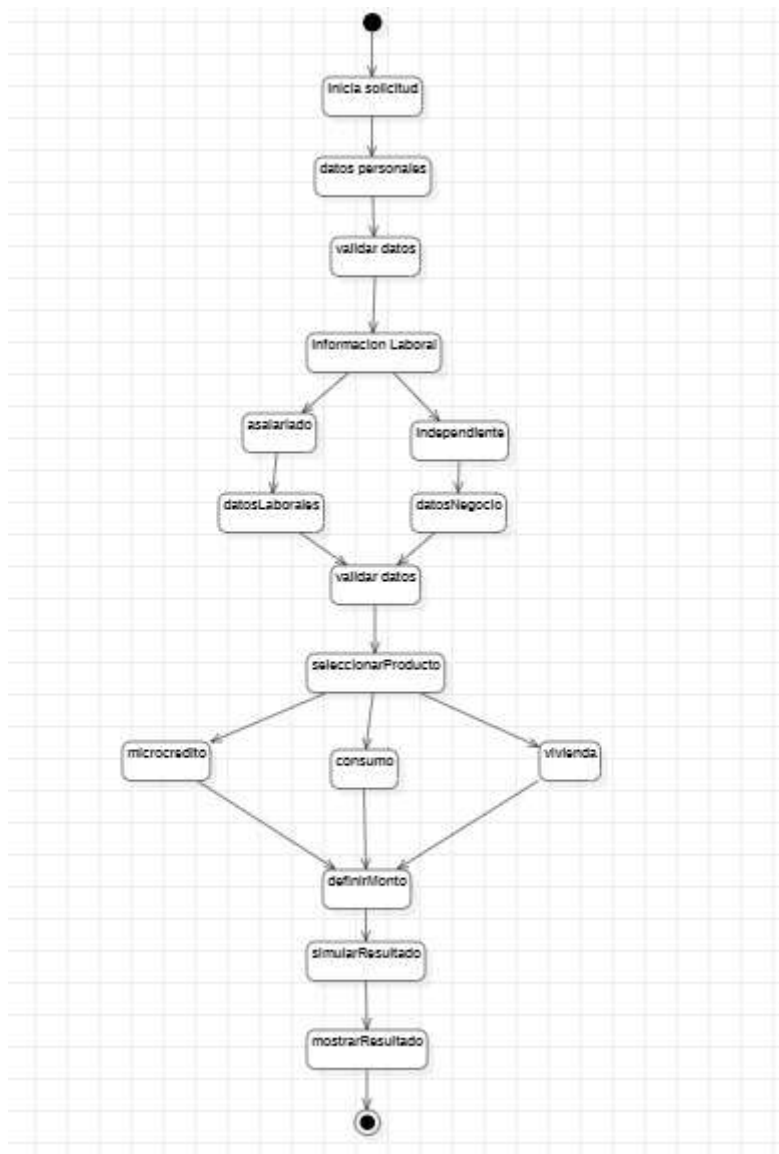


cd. CU12: Registrar solicitud de crédito



1.6.3 Diagrama de estado





1.6.4 Características generales

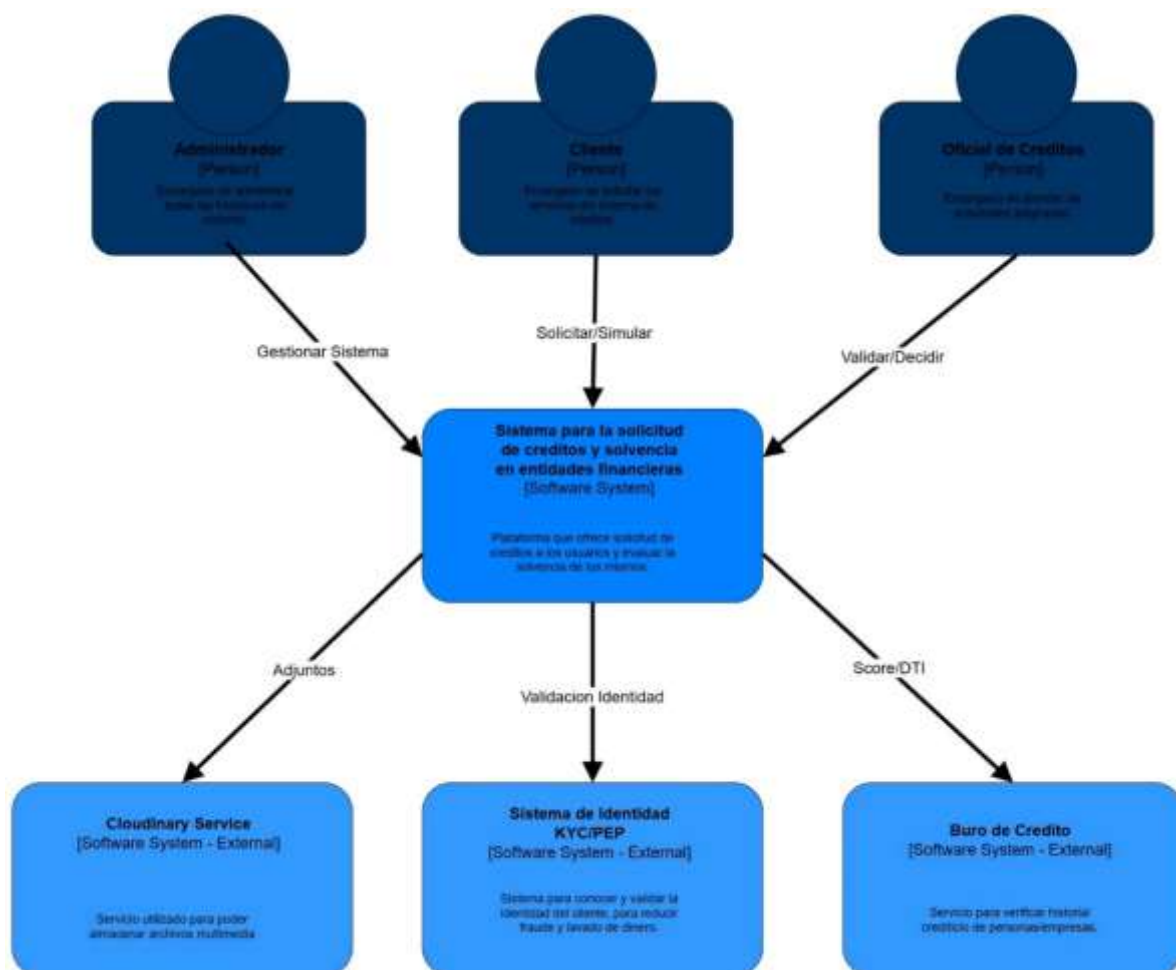
2 Proceso/patrón de desarrollo por Historia de Usuario

Sprint 2	
Product Owner	Molina Cortez Maikol Anthony
Scrum Master	Perez Gomez Daniela
Team Developer	<ul style="list-style-type: none"> ❖ Mendoza Rodríguez Luis Pablo ❖ Flores Quispe Nathaly Madeline ❖ Banegas Roca Luis Fernando ❖ Melgar Arteaga Alejandro

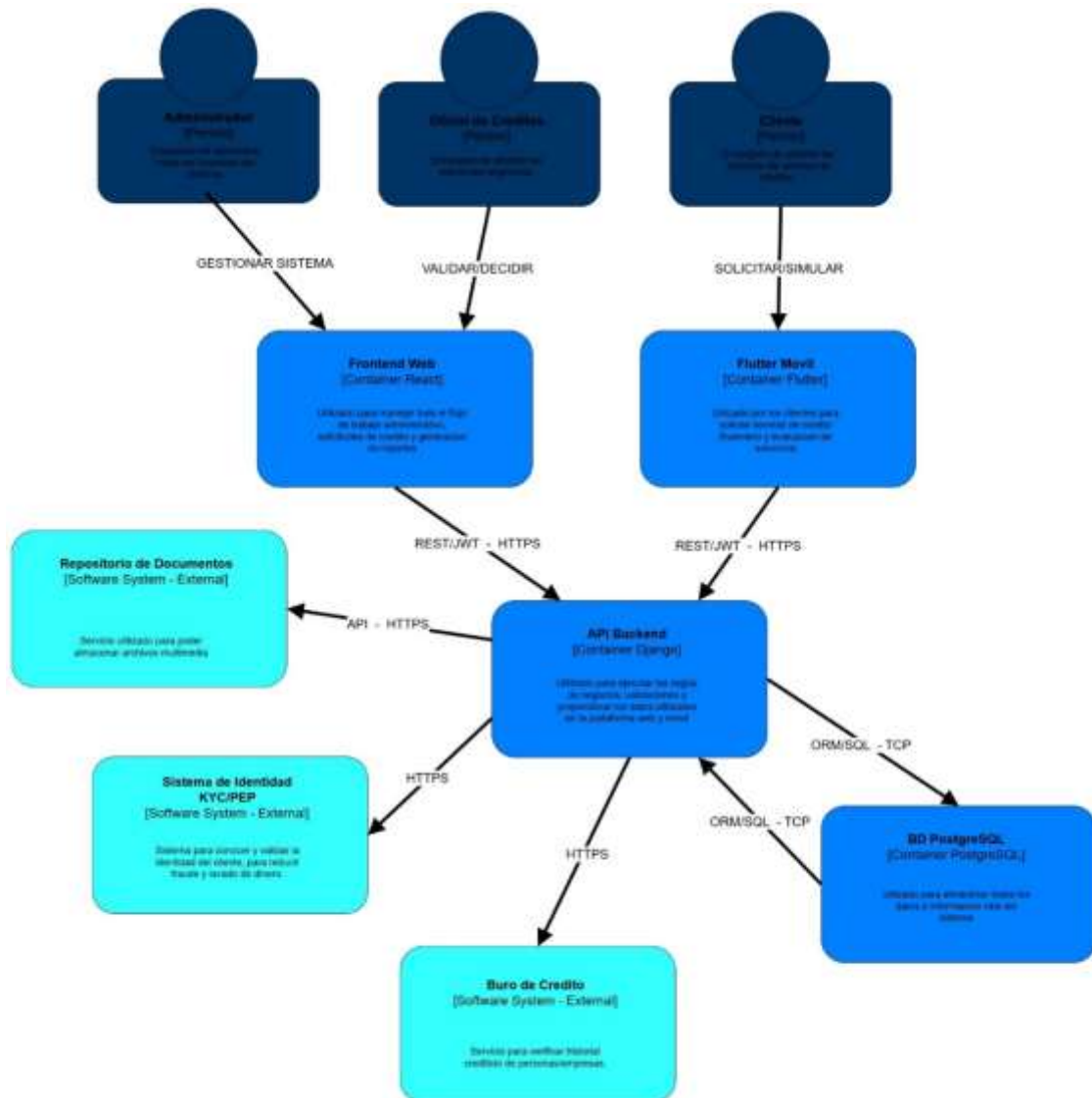
2.1 Diseño

2.1.1 Diseñar de la arquitectura C4

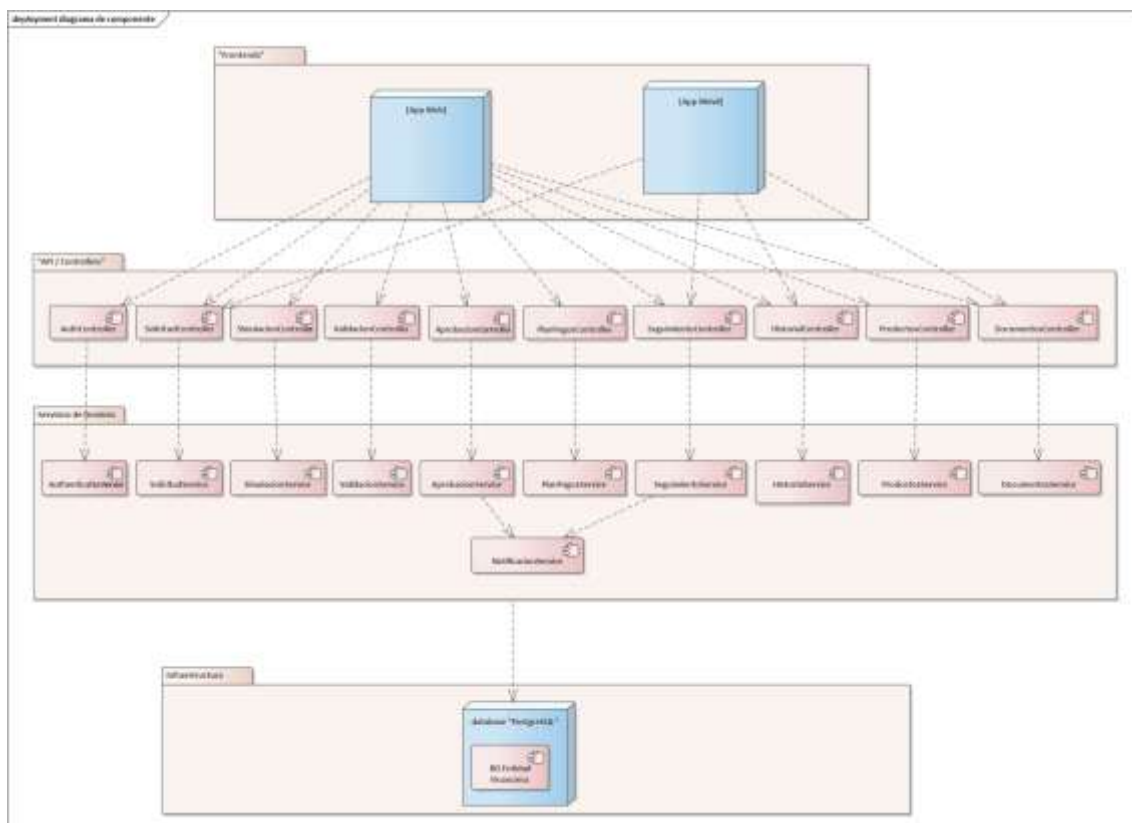
Nivel 1: Context



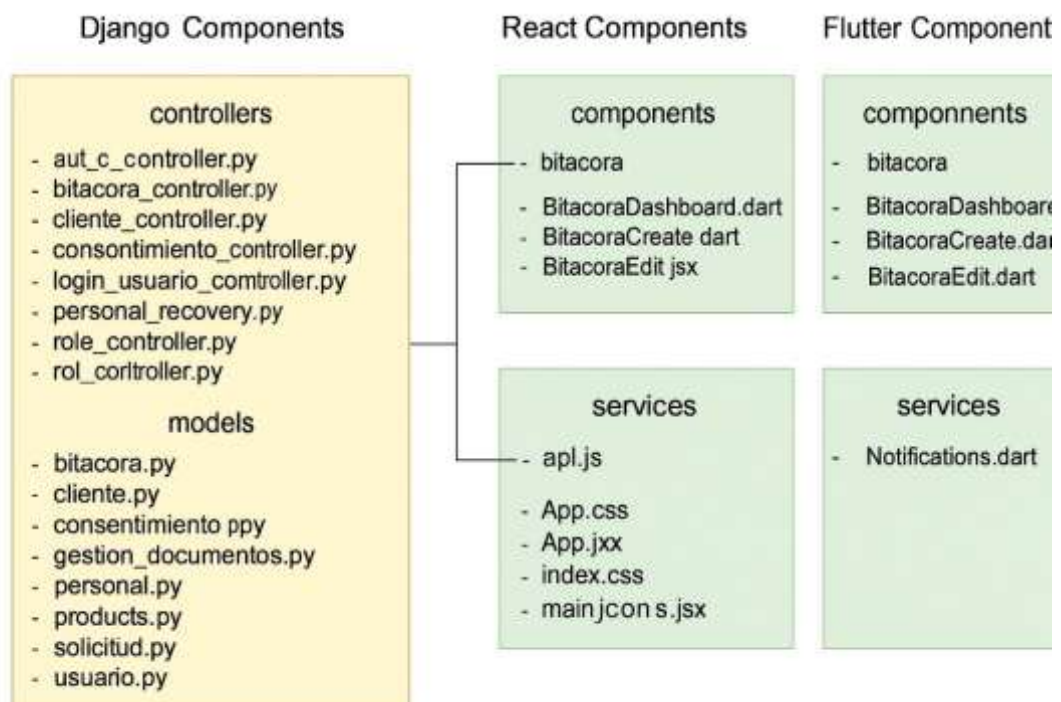
Nivel 2: Contenedor



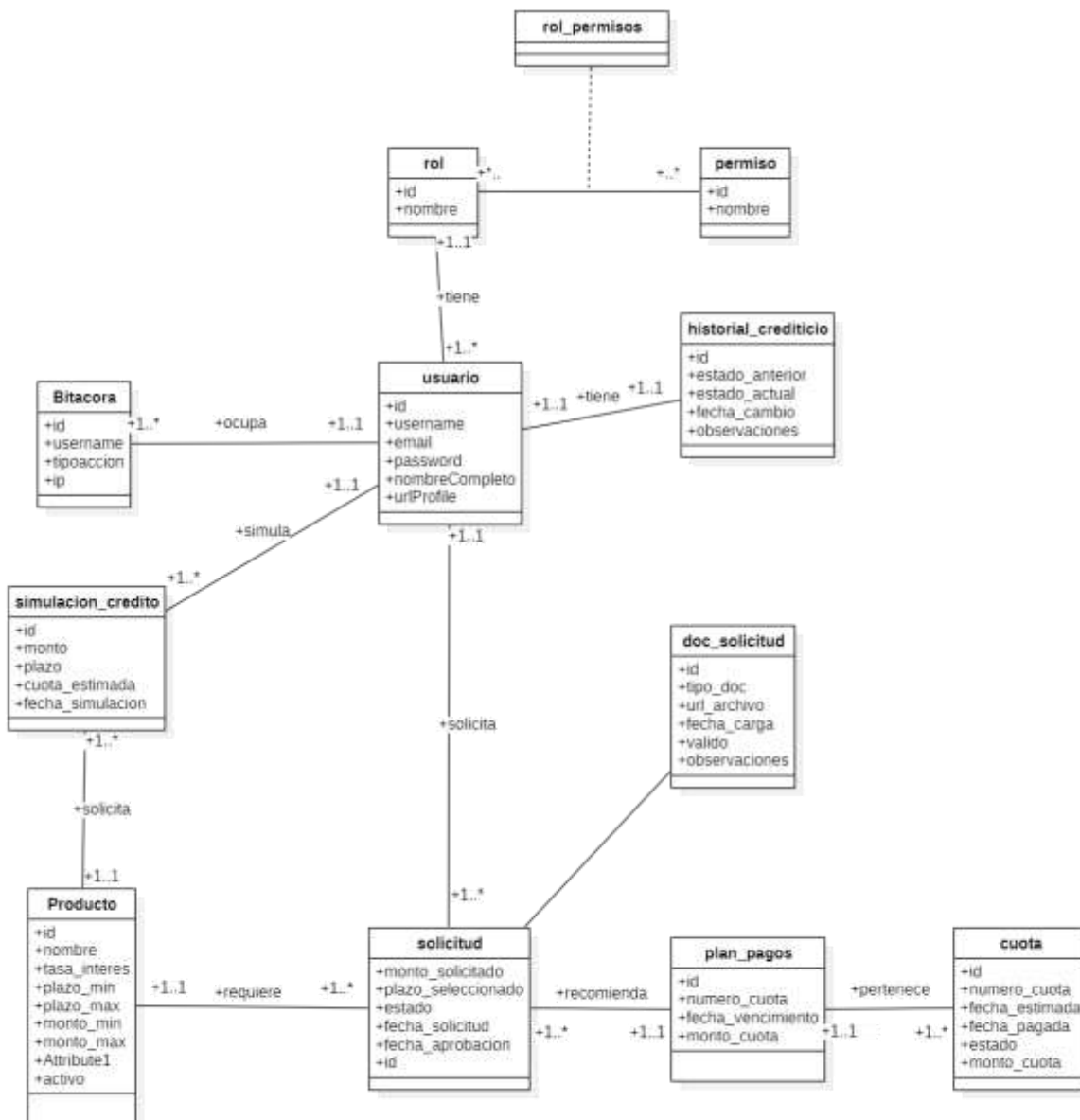
Nivel 3: Diagrama de Componente



Nivel 4: Código



2.2 Diseño de datos



Mapeo

TABLA USUARIO							
PK				FK			
id	username	email	password	rol_id	created_id	updated_id	is_deleted

TABLA ROLES				
PK				
id	nombre	created_id	updated_id	is_deleted

TABLA PERMISOS				
PK			FK	
id	nombre	created_id	updated_id	is_deleted

TABLA BITACORA_USUARIOS				
PK				
id	ip	tipo_accion	created_at	update_at

TABLA ROL_PERMISOS			
PK/FK	PK/FK		
rol_id	permiso_id	created_id	updated_id

TABLA BITACORA					
PK			FK		
id	username	tipoaccion	ip	fecha_registro	id_usuario

TABLA SIMULACION_CREDITO						
PK				FK		
id	monto	plazo	cuota_estimada	fecha_simulacion	id_usuario	id_producto

TABLA PRODUCTO								
PK								
id	nombre	tasa_interes	plazo_min	plazo_max	monto_min	monto_max	atributo1	activo

TABLA SOLICITUD								
PK					FK			
id	monto_solicitado	plazo_seleccionado	estado	fecha_solicitud	fecha_aprobacion	id_producto	id_usuario	id_planpagos

TABLA DOC_SOLICITUD						
PK					FK	
id	tipo_doc	url_archivo	fecha_carga	valido	observaciones	id_solicitud

TABLA PLAN_PAGOS				
PK				
id	numero_cuota	fecha_vencimiento	monto_cuota	

TABLA CUOTA						
PK					FK	
id	numero_cuota	fecha_estimada	fecha_pagada	estado	monto_cuota	id_planpagos

TABLA HISTORIAL_CREDITICIO				
PK				
id	estado_anterior	estado_actual	fecha_cambio	observaciones

TABLA NOTIFICACIONES							
PK						FK	
id	titulo	mensaje	canal	tipo	fecha_envio	leido	id_usuario
							id_solicitud

Script

```
CREATE TABLE roles (  
  id          SERIAL NOT NULL,  
  nombre      VARCHAR(80) NOT NULL,  
  created_at  TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),  
  updated_at  TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),  
  is_deleted  BOOLEAN NOT NULL DEFAULT FALSE,  
  CONSTRAINT pk_roles PRIMARY KEY (id),  
  CONSTRAINT uq_roles_nombre UNIQUE (nombre)  
);  
  
CREATE TABLE permisos (  
  id          SERIAL NOT NULL,  
  nombre      VARCHAR(80) NOT NULL,  
  created_at  TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),  
  updated_at  TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),  
  is_deleted  BOOLEAN NOT NULL DEFAULT FALSE,  
  CONSTRAINT pk_permisos PRIMARY KEY (id),  
  CONSTRAINT uq_permisos_nombre UNIQUE (nombre)  
);  
  
CREATE TABLE usuarios (  
  id          SERIAL NOT NULL,  
  username    VARCHAR(80) NOT NULL,  
  nombre      VARCHAR(80) NOT NULL,  
  email       VARCHAR(120) NOT NULL,  
  password    VARCHAR(200) NOT NULL,  
  rol_id      INTEGER NOT NULL,  
  created_at  TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),  
  updated_at  TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),  
  is_deleted  BOOLEAN NOT NULL DEFAULT FALSE,  
  CONSTRAINT pk_usuarios PRIMARY KEY (id),  
  CONSTRAINT uq_usuarios_username UNIQUE (username),  
  CONSTRAINT uq_usuarios_email UNIQUE (email),  
  CONSTRAINT fk_usuarios_roles  
    FOREIGN KEY (rol_id) REFERENCES roles (id)  
    ON UPDATE CASCADE ON DELETE RESTRICT  
);  
  
CREATE TABLE rol_permisos (  
  rol_id      INTEGER NOT NULL,  
  permiso_id  INTEGER NOT NULL,  
  created_at  TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),  
  updated_at  TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),  
  is_deleted  BOOLEAN NOT NULL DEFAULT FALSE,  
  CONSTRAINT pk_rol_permisos PRIMARY KEY (rol_id, permiso_id),  
  CONSTRAINT fk_rol_permisos_rol  
    FOREIGN KEY (rol_id) REFERENCES roles(id)  
    ON UPDATE CASCADE ON DELETE CASCADE,  
  CONSTRAINT fk_rol_permisos_permiso  
    FOREIGN KEY (permiso_id) REFERENCES permisos(id)  
    ON UPDATE CASCADE ON DELETE CASCADE  
);  
  
CREATE TABLE bitacora (  
  id          SERIAL NOT NULL,  
  usuario_id  INTEGER NOT NULL,  
  tipoaccion  VARCHAR(50) NOT NULL,  
  ip          VARCHAR(45),  
  created_at  TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),  
  updated_at  TIMESTAMP WITHOUT TIME ZONE NOT NULL DEFAULT now(),  
  is_deleted  BOOLEAN NOT NULL DEFAULT FALSE,  
  CONSTRAINT pk_bitacora PRIMARY KEY (id),  
  CONSTRAINT fk_bitacora_usuario
```



```

        FOREIGN KEY (usuario_id) REFERENCES usuarios(id)
        ON UPDATE CASCADE ON DELETE RESTRICT
    );

CREATE TABLE producto (
    id SERIAL PRIMARY KEY,
    nombre VARCHAR(100) NOT NULL,
    tasa_interes DECIMAL(5,2) NOT NULL,
    plazo_min INT,
    plazo_max INT,
    monto_min DECIMAL(12,2),
    monto_max DECIMAL(12,2),
    atributo1 VARCHAR(100),
    activo BOOLEAN DEFAULT TRUE,
    created_id INT,
    updated_id INT,
    is_deleted BOOLEAN DEFAULT FALSE
);

CREATE TABLE simulacion_credito (
    id SERIAL PRIMARY KEY,
    monto DECIMAL(12,2),
    plazo INT,
    cuota_estimada DECIMAL(12,2),
    fecha_simulacion TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    usuario_id INT NOT NULL,
    producto_id INT NOT NULL,
    created_id INT,
    updated_id INT,
    is_deleted BOOLEAN DEFAULT FALSE,
    FOREIGN KEY (usuario_id) REFERENCES usuario(id),
    FOREIGN KEY (producto_id) REFERENCES producto(id)
);

CREATE TABLE solicitud (
    id SERIAL PRIMARY KEY,
    monto_solicitado DECIMAL(12,2),
    plazo_seleccionado INT,
    estado VARCHAR(50) DEFAULT 'pendiente',
    fecha_solicitud TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    fecha_aprobacion TIMESTAMP,
    usuario_id INT NOT NULL,
    producto_id INT NOT NULL,
    created_id INT,
    updated_id INT,
    is_deleted BOOLEAN DEFAULT FALSE,
    FOREIGN KEY (usuario_id) REFERENCES usuario(id),
    FOREIGN KEY (producto_id) REFERENCES producto(id)
);

CREATE TABLE doc_solicitud (
    id SERIAL PRIMARY KEY,
    tipo_doc VARCHAR(100),
    url_archivo TEXT,
    fecha_carga TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    valido BOOLEAN DEFAULT TRUE,
    observaciones TEXT,
    solicitud_id INT NOT NULL,
    created_id INT,
    updated_id INT,

```

```

        is_deleted BOOLEAN DEFAULT FALSE,
        FOREIGN KEY (solicitud_id) REFERENCES solicitud(id)
    );

CREATE TABLE plan_pagos (
    id SERIAL PRIMARY KEY,
    numero_cuota INT,
    fecha_vencimiento DATE,
    monto_cuota DECIMAL(12,2),
    solicitud_id INT NOT NULL,
    created_id INT,
    updated_id INT,
    is_deleted BOOLEAN DEFAULT FALSE,
    FOREIGN KEY (solicitud_id) REFERENCES solicitud(id)
);

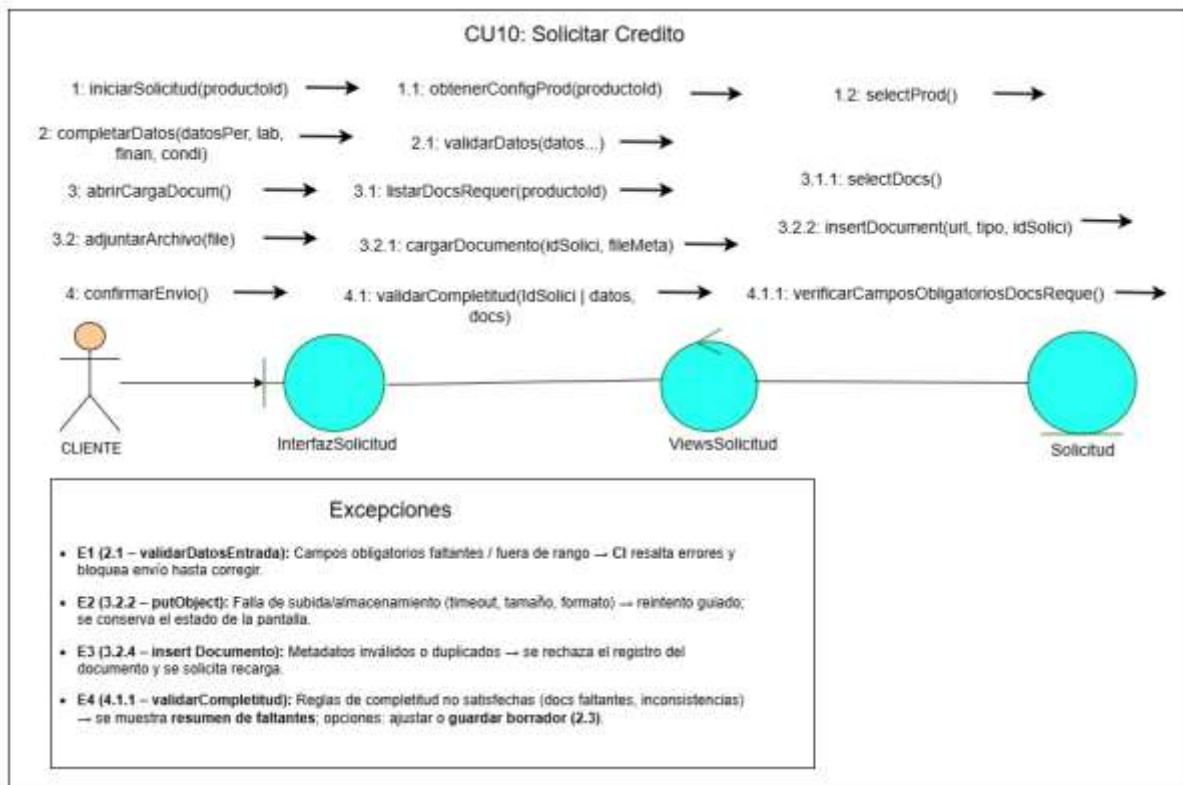
CREATE TABLE cuota (
    id SERIAL PRIMARY KEY,
    numero_cuota INT,
    fecha_estimada DATE,
    fecha_pagada DATE,
    estado VARCHAR(50),
    monto_cuota DECIMAL(12,2),
    plan_pagos_id INT NOT NULL,
    created_id INT,
    updated_id INT,
    is_deleted BOOLEAN DEFAULT FALSE,
    FOREIGN KEY (plan_pagos_id) REFERENCES plan_pagos(id)
);

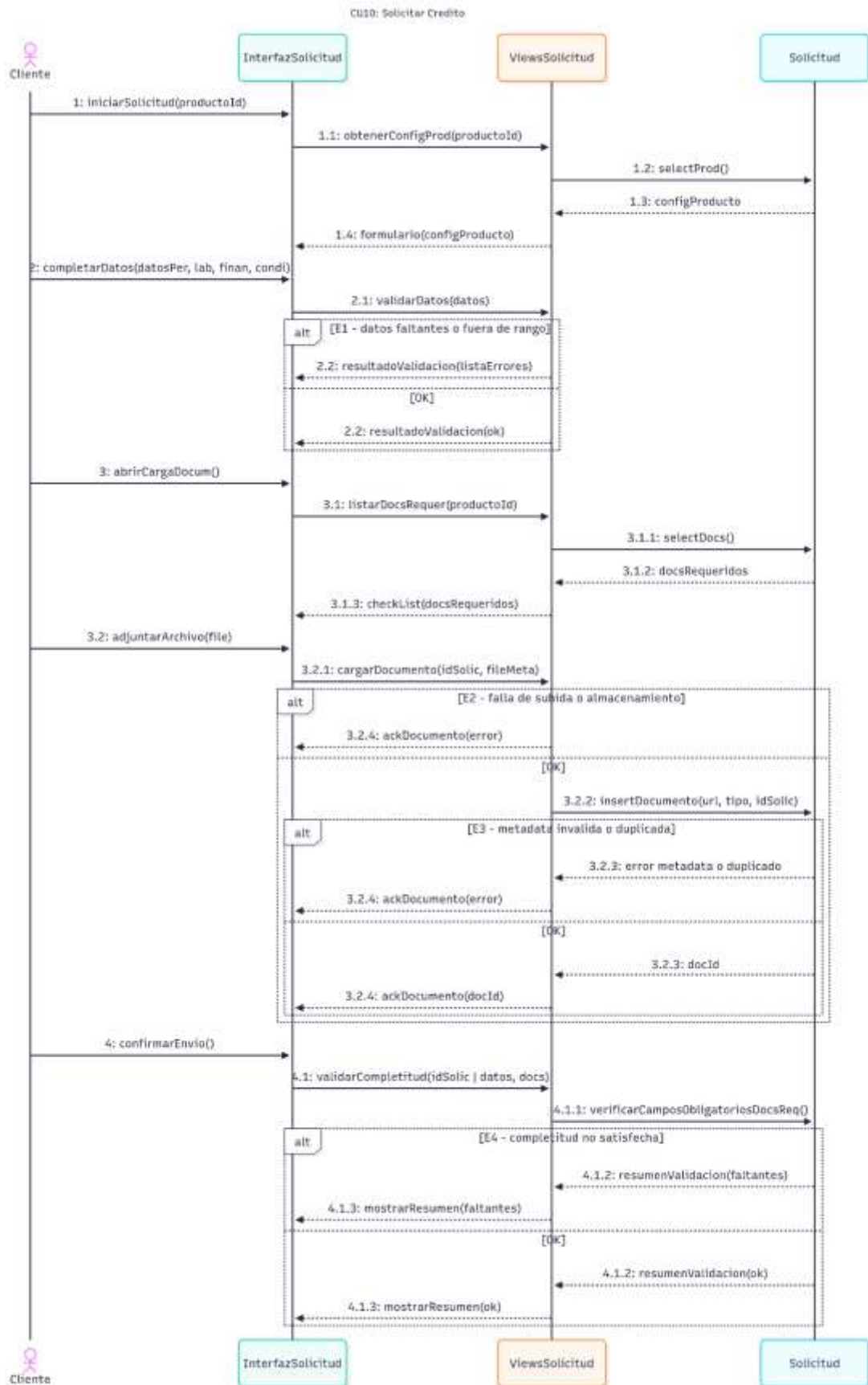
CREATE TABLE historial_credificio (
    id SERIAL PRIMARY KEY,
    estado_anterior VARCHAR(50),
    estado_actual VARCHAR(50),
    fecha_cambio TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    observaciones TEXT,
    solicitud_id INT NOT NULL,
    usuario_id INT NOT NULL,
    created_id INT,
    updated_id INT,
    is_deleted BOOLEAN DEFAULT FALSE,
    FOREIGN KEY (solicitud_id) REFERENCES solicitud(id),
    FOREIGN KEY (usuario_id) REFERENCES usuario(id)
);

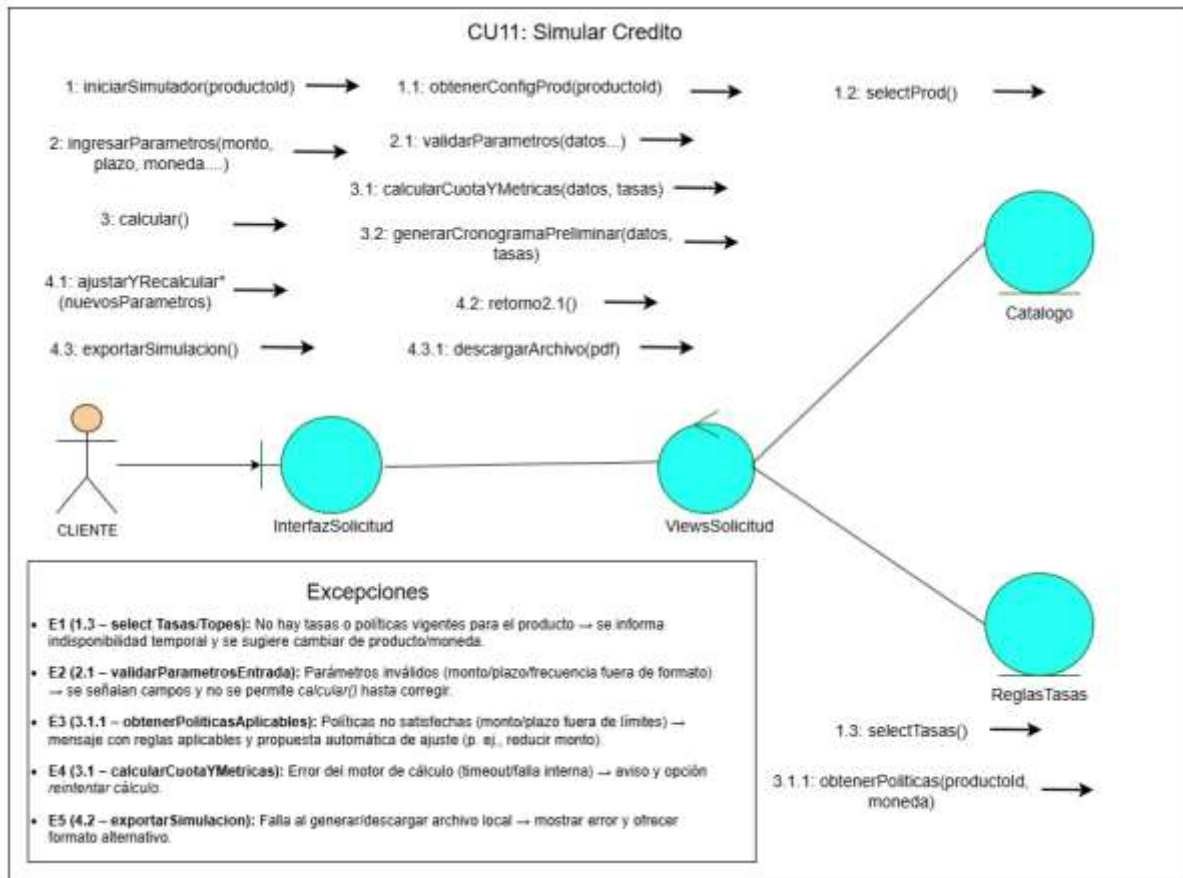
CREATE TABLE notificacion (
    id SERIAL PRIMARY KEY,
    titulo VARCHAR(200),
    mensaje TEXT,
    canal VARCHAR(50),
    tipo VARCHAR(50),
    fecha_envio TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    leído BOOLEAN DEFAULT FALSE,
    usuario_id INT NOT NULL,
    solicitud_id INT NOT NULL,
    created_id INT,
    updated_id INT,
    is_deleted BOOLEAN DEFAULT FALSE,
    FOREIGN KEY (usuario_id) REFERENCES usuario(id),
    FOREIGN KEY (solicitud_id) REFERENCES solicitud(id)
);

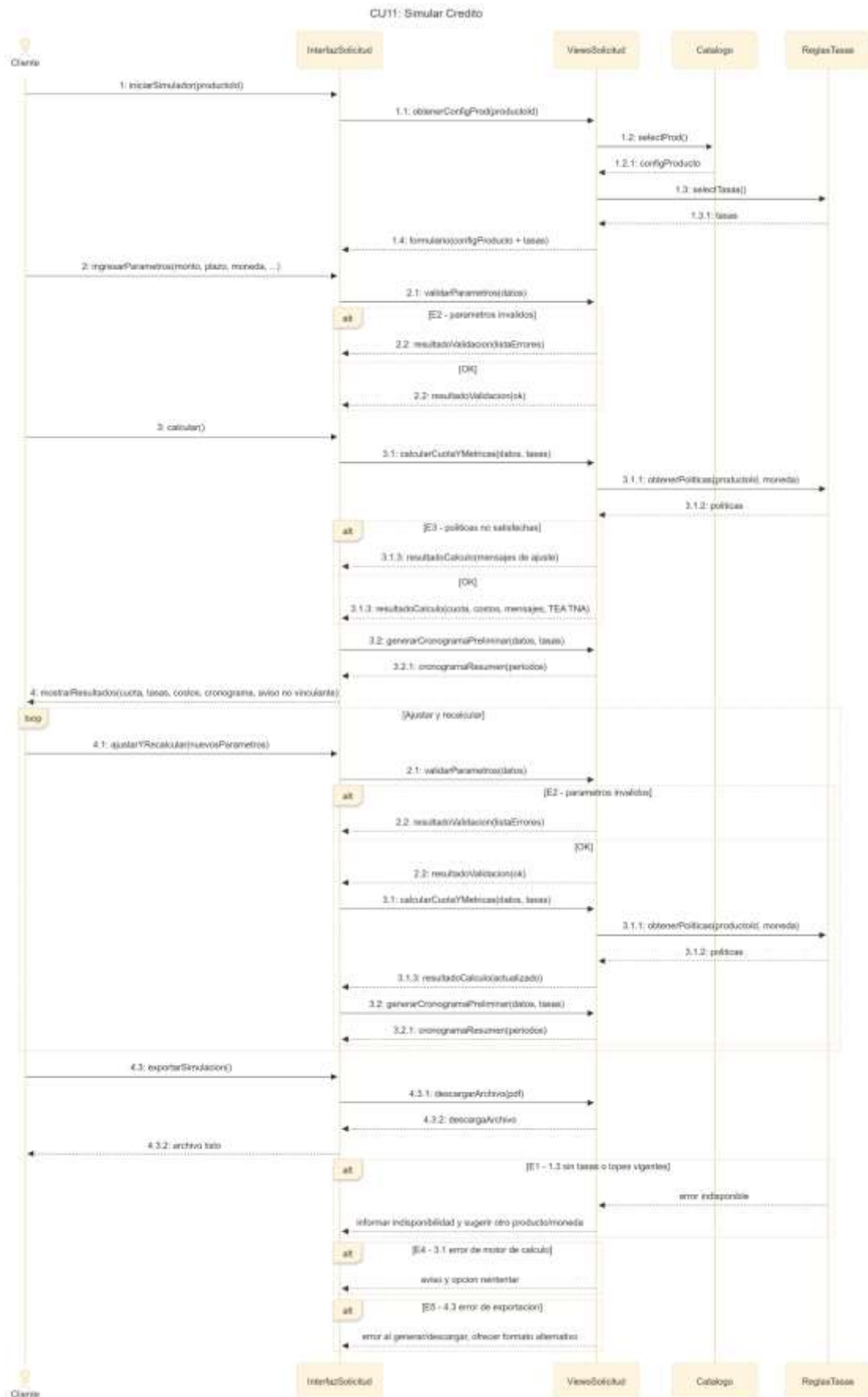
```

2.2.1 Diseño de la lógica de negocio

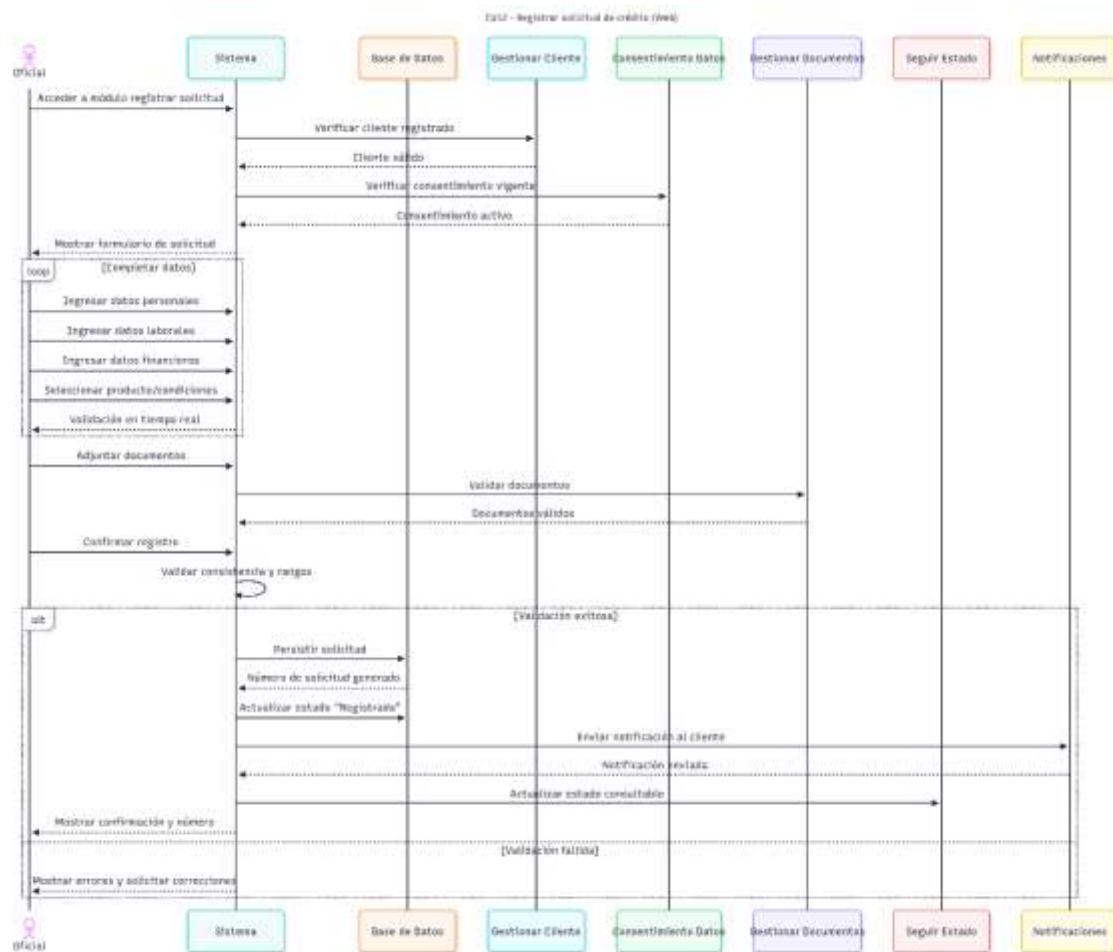
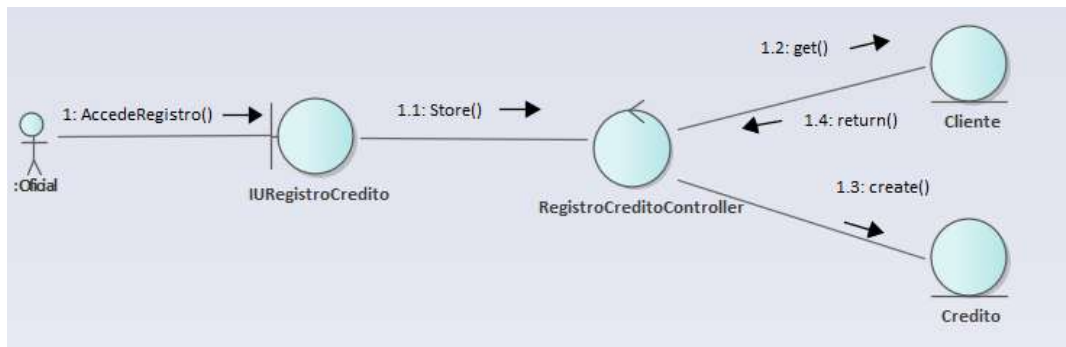




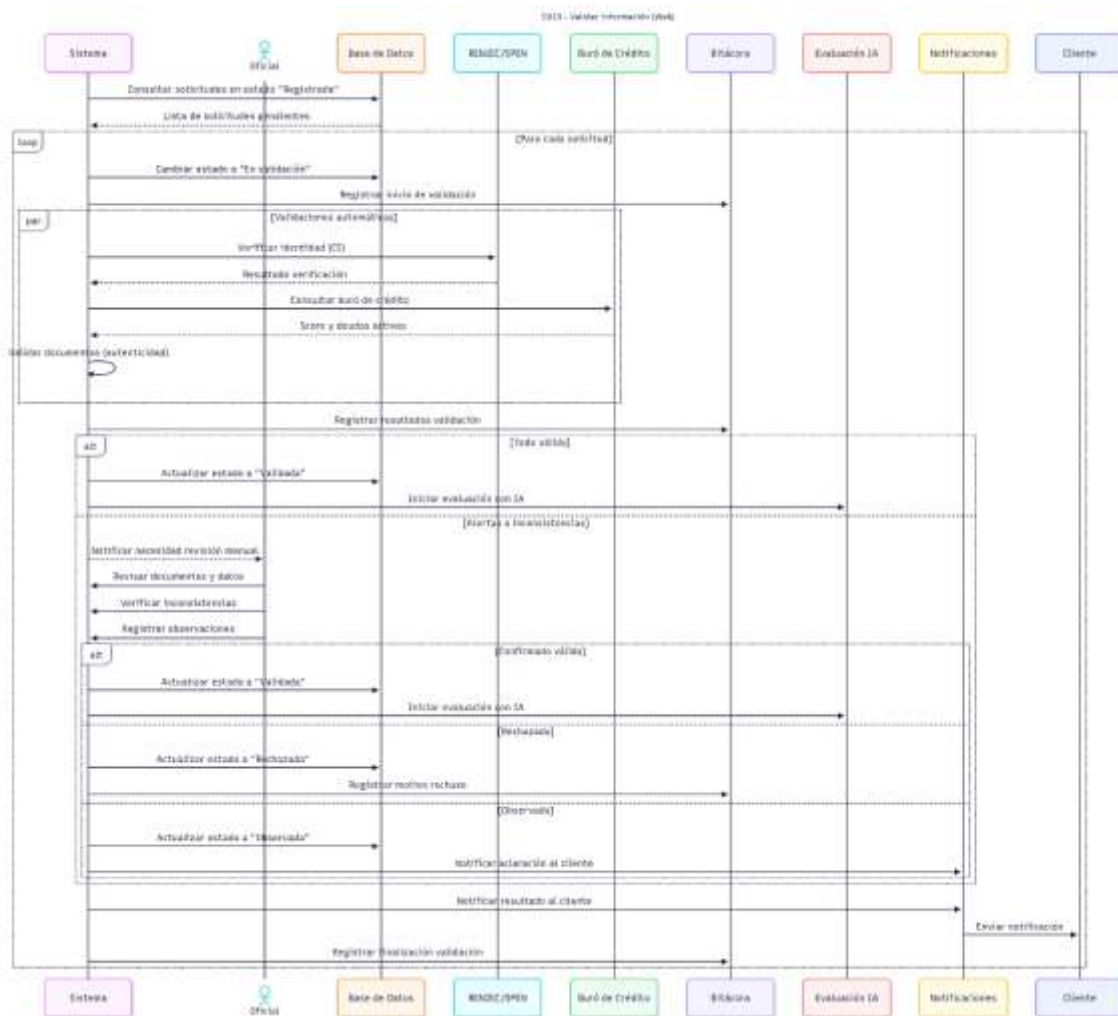
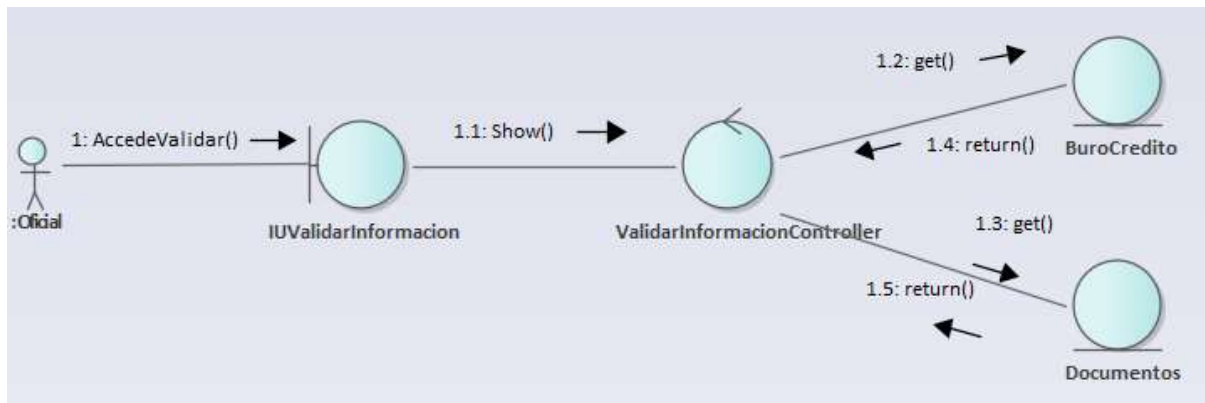


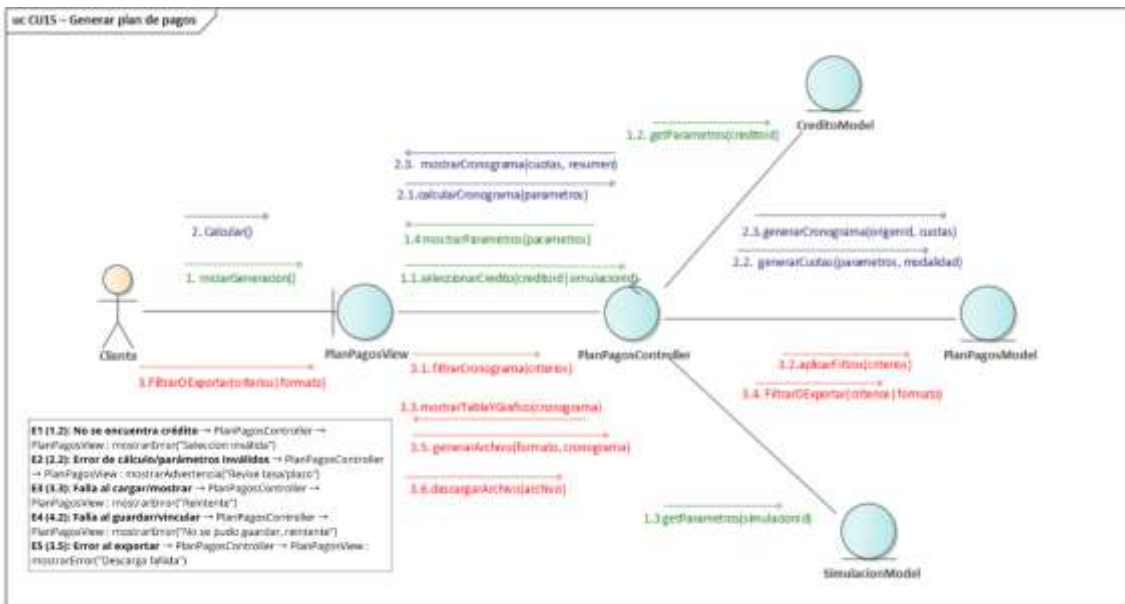


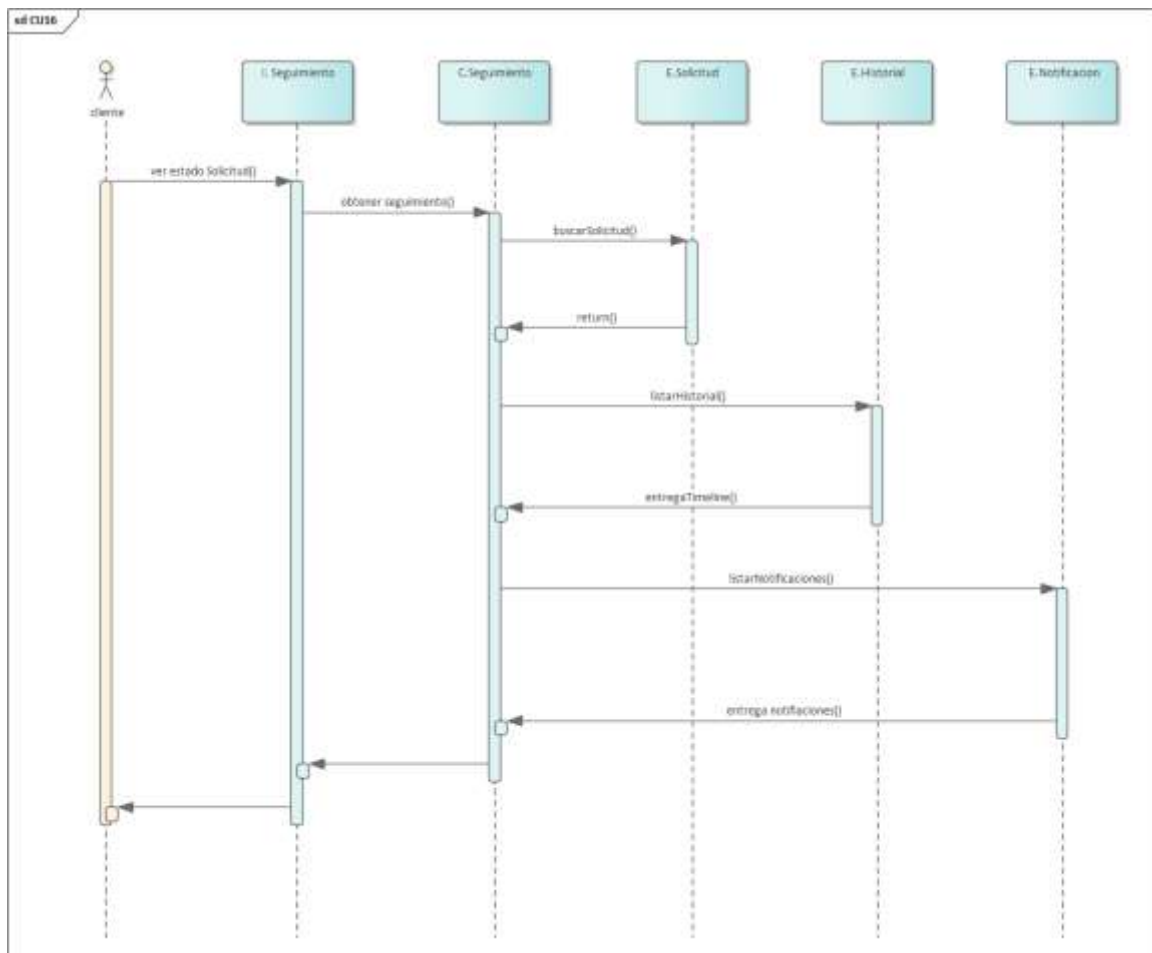
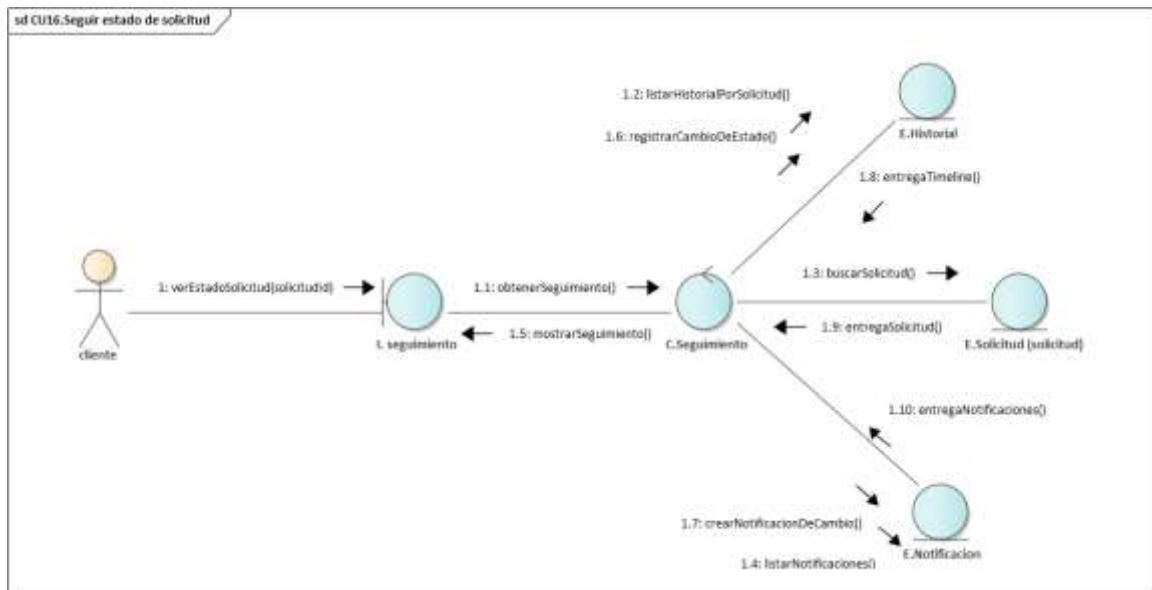
CU12:



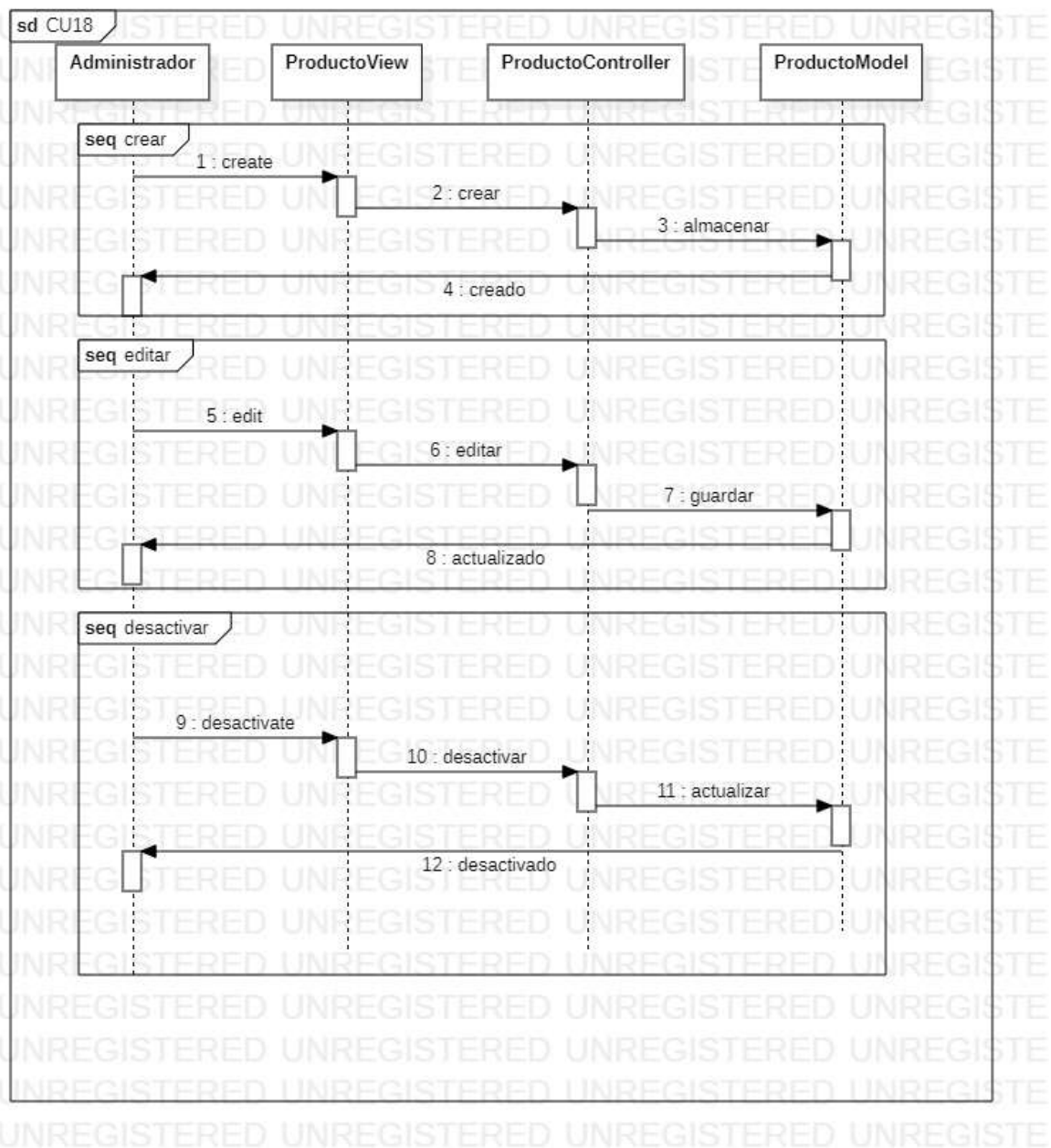
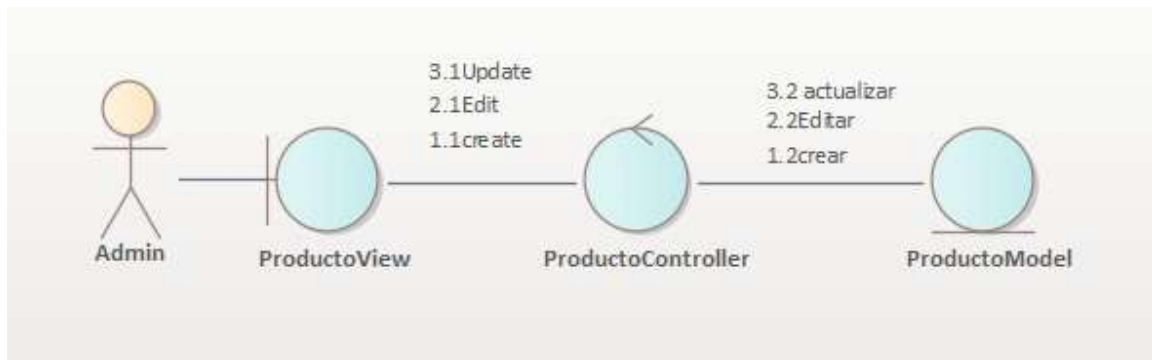
CU13:



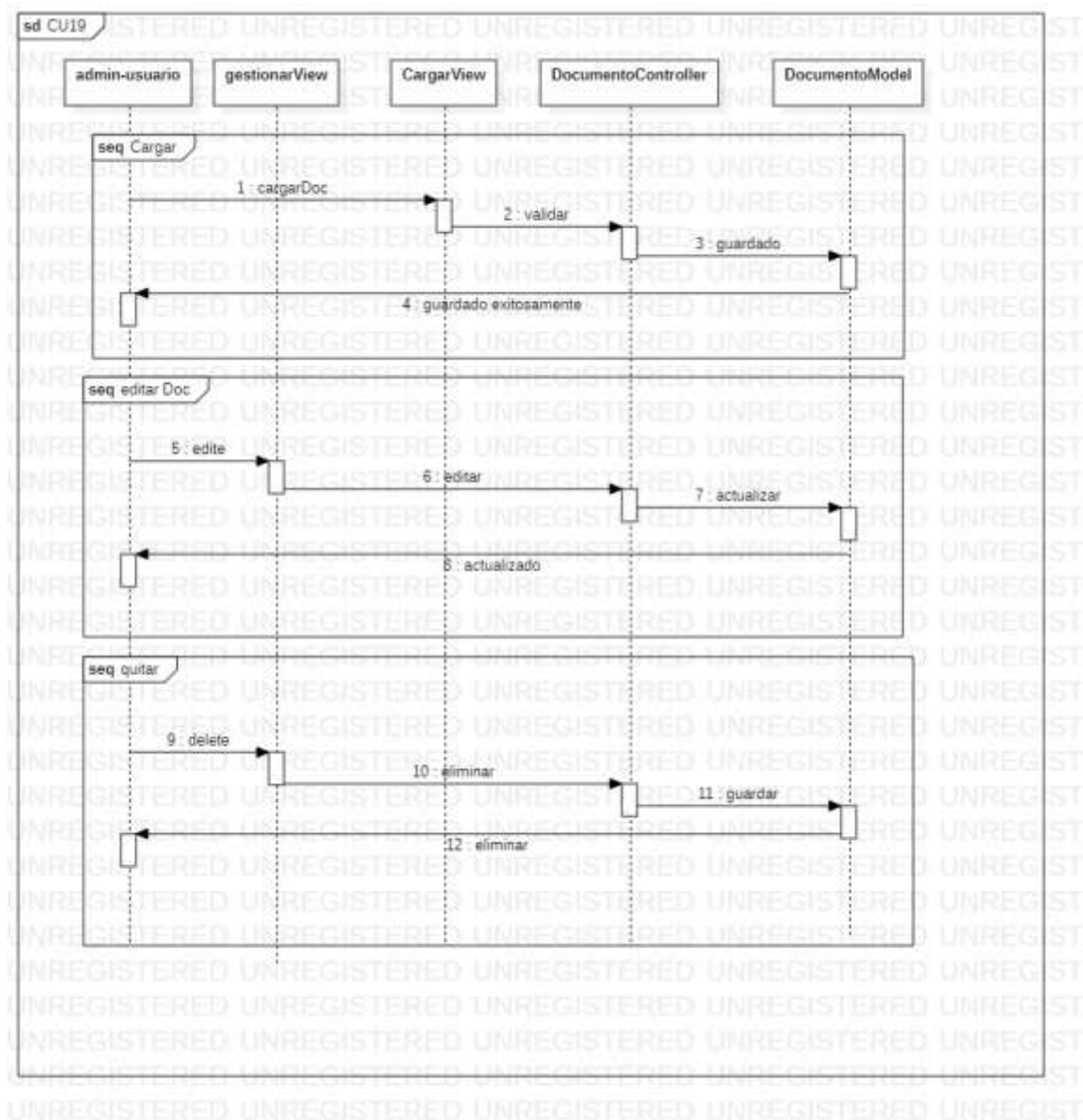
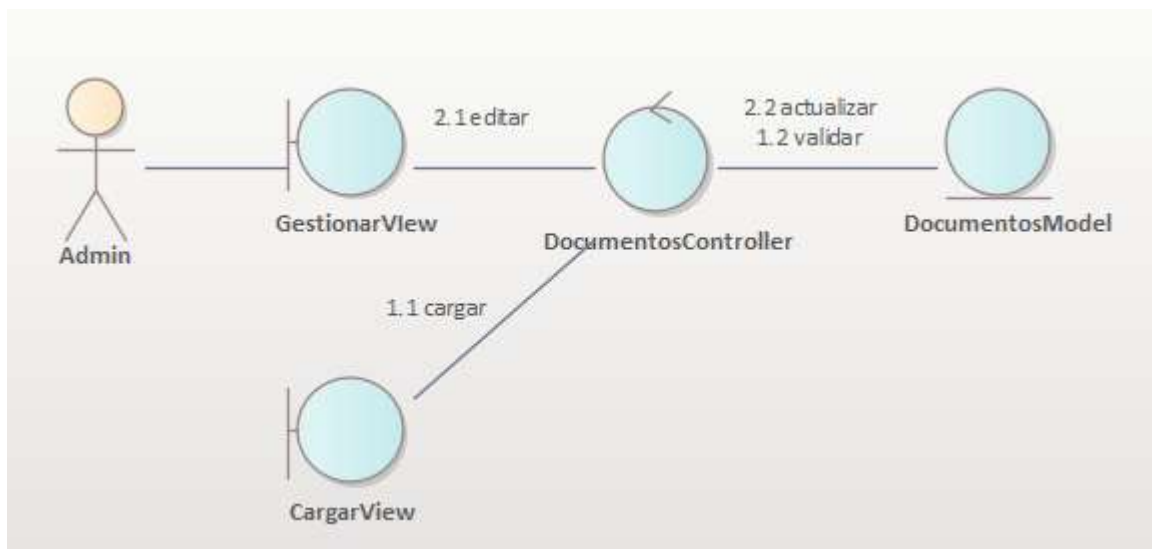




CU18: Gestionar Productos Financieros



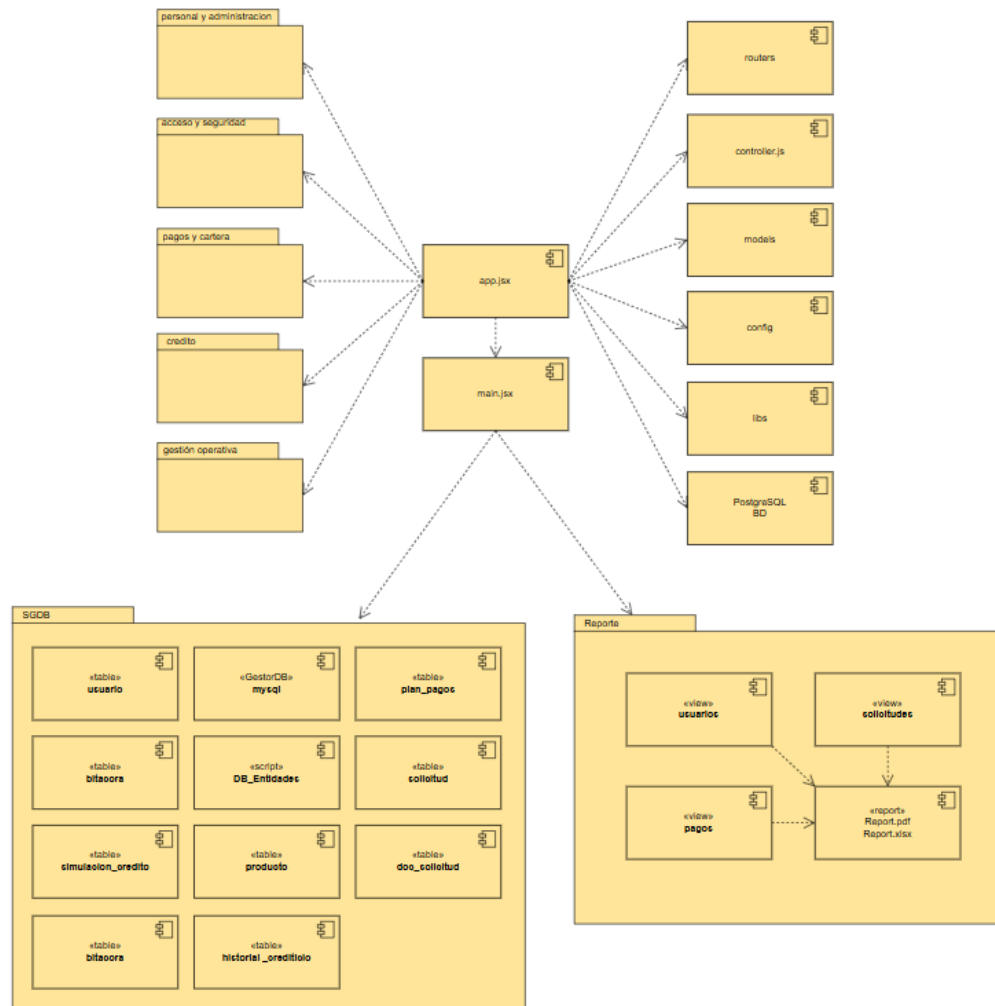
CU19: Cargar y Gestionar Documentos



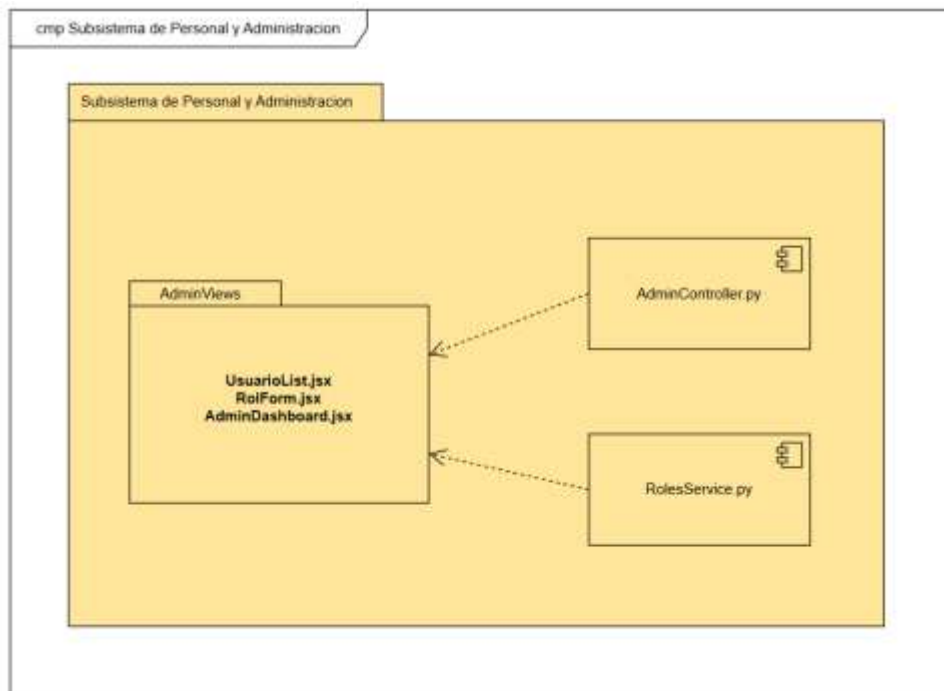
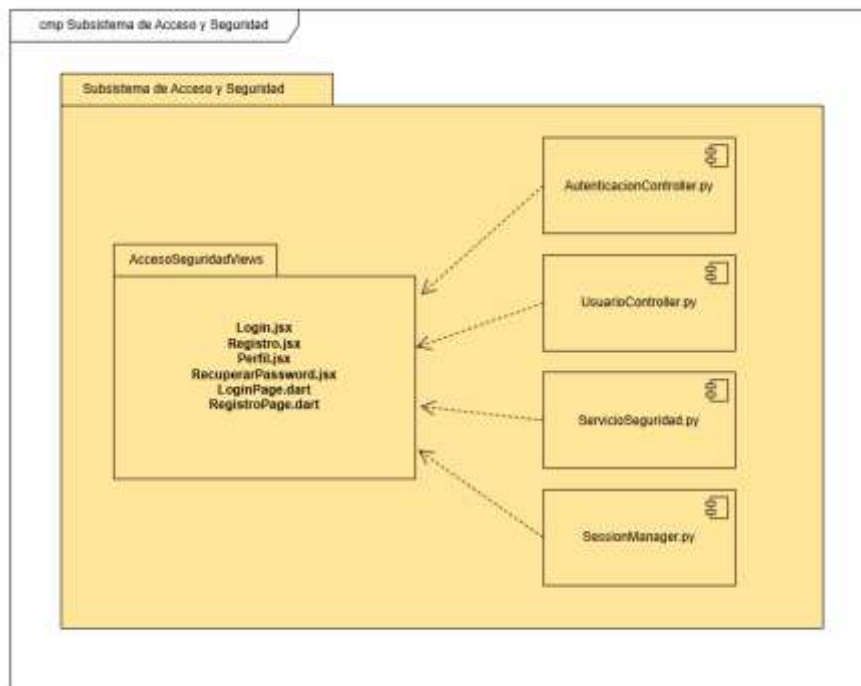
2.3 Implementación

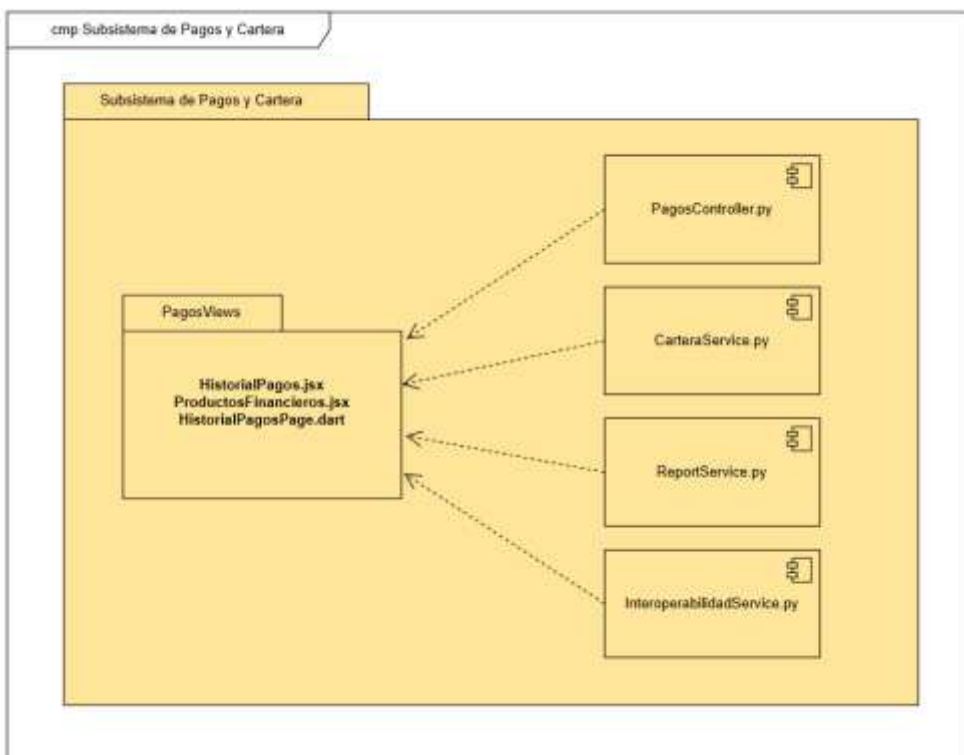
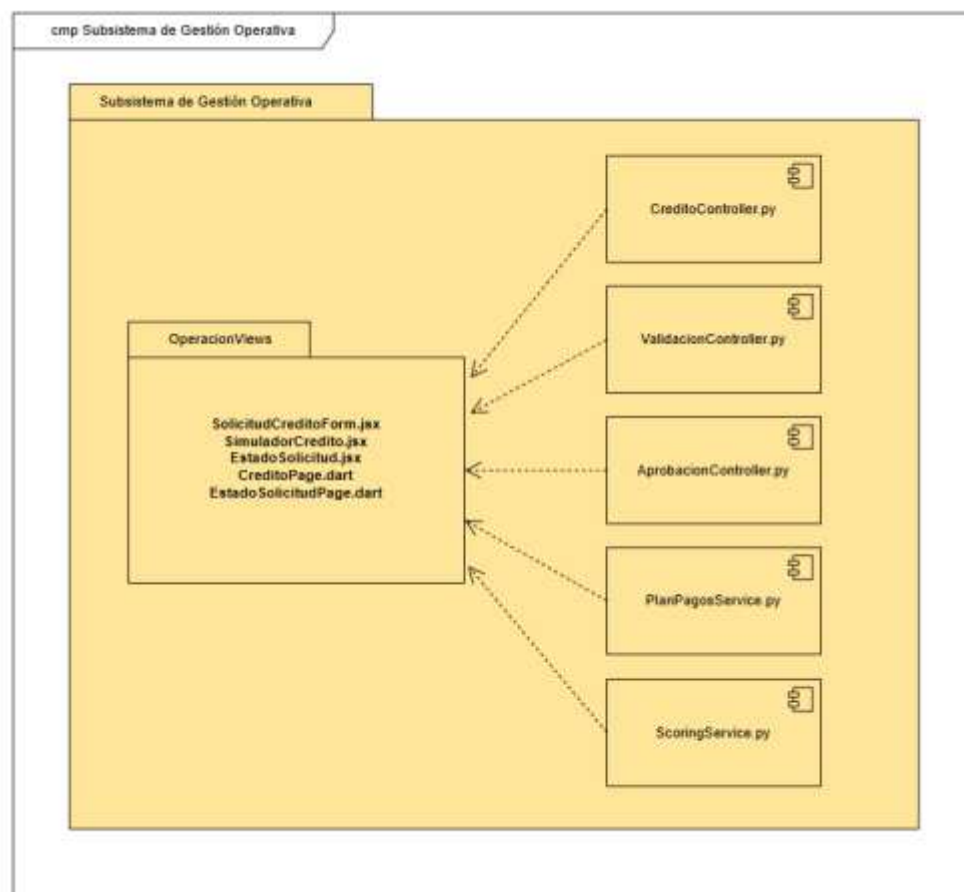
2.3.1 Componentes y artefactos generados

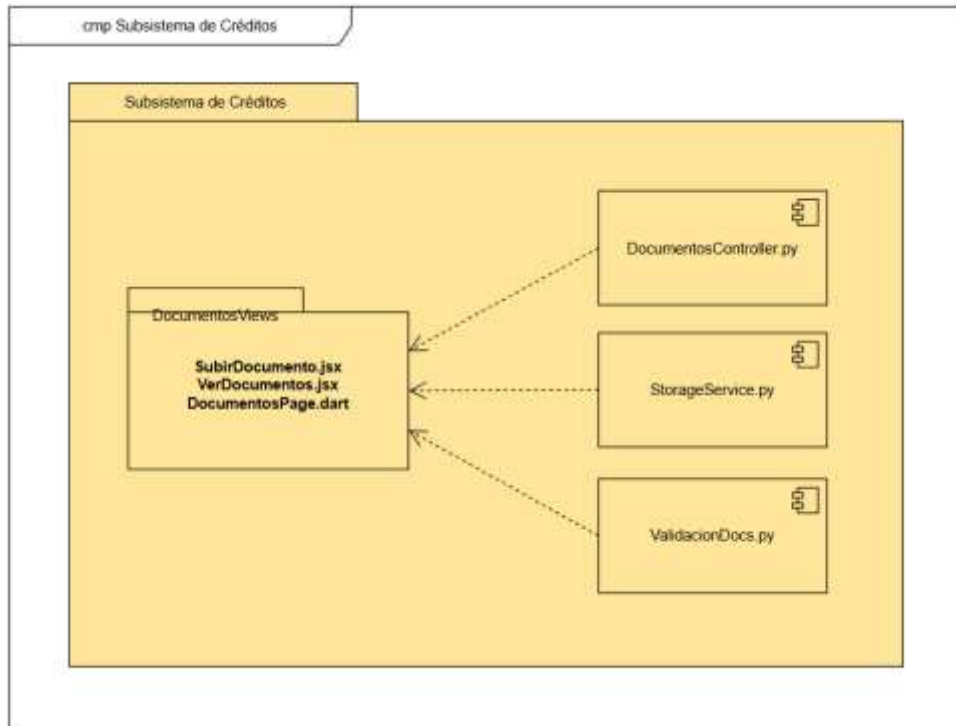
2.3.2 Implementación de la arquitectura del sistema



2.3.3 Implementación de la arquitectura del subsistema







2.4 Pruebas

2.4.1 Plan de pruebas (criterios de aceptación)

CU10 – Solicitar Crédito (Web / Móvil)			
Nº	Criterio de aceptación	Tipo	Evidencia sugerida
1	Se puede crear una solicitud desde Móvil y Web con producto, monto, plazo y moneda.	Funcional	Capturas flujo Móvil/Web
2	Validaciones de obligatorios y rangos (monto/plazo) al enviar.	Funcional	Mensajes de validación
3	Solo usuario autenticado (cliente) puede solicitar; no autenticado es bloqueado.	Seguridad	Prueba con sesión expirada / no login
4	La solicitud creada aparece en el listado con estado inicial (“Registrada/Enviada”).	Funcional	Captura del listado
5	Auditoría registra creación (quién/cuándo/IP mínima).	Auditoría	Registro histórico / bitácora

CU11 – Simular Crédito (Web / Móvil)			
Nº	Criterio	Tipo	Evidencia
1	Se puede simular ingresando producto, monto, plazo y moneda en Móvil y Web .	Funcional	Capturas flujo Móvil/Web
2	Validaciones de obligatorios y rangos (monto/plazo) muestran mensajes al calcular.	Funcional	Mensajes de validación
3	La cuota se calcula correctamente (sistema francés) y muestra tasa aplicada.	Funcional	Captura de resultado con fórmula/cuota
4	Se muestra resumen : total a pagar, intereses totales y primera fila del plan.	Funcional	Captura del resumen
5	Se pueden comparar 2 escenarios (p. ej., distintos plazos) en la misma vista.	Funcional	Captura de comparativa
6	Manejo de errores (timeout/500 de catálogo de productos): mensaje amable y reintento.	Robustez	Captura del mensaje / retry

CU12 – Registrar Solicitud de Crédito (Web)			
Nº	Criterio	Tipo	Evidencia
1	Se puede crear una solicitud con producto, monto, plazo, moneda y cliente.	Funcional	Captura formulario y confirmación
2	Validaciones de obligatorios y de rangos (monto/plazo) al enviar.	Funcional	Mensajes de validación
3	Duplicados: bloquea si el cliente ya tiene una solicitud activa del mismo producto.	Regla de negocio	Mensaje de bloqueo / consulta
4	Permisos: solo cliente autenticado/rol permitido puede registrar.	Seguridad	Prueba con sesión expirada / rol no autorizado
5	Tras crear, aparece en el listado con estado inicial (p. ej., “Registrada”).	Funcional	Captura del listado
6	Auditoría: registra quién, cuándo e IP mínima de creación.	Auditoría	Registro histórico / bitácora

CU13 – Validar Información (Web)			
Nº	Criterio	Tipo	Evidencia
1	La vista muestra los datos de la solicitud (CU12) y panel de checks.	Funcional	Captura de la vista
2	Documentos requeridos presentes según producto; faltantes marcan “Observado”.	Funcional	Captura de adjuntos / marca “Observado”
3	KYC/PEP/listas: si hay match o bloqueo KYC, no permite cerrar validación.	Seguridad	Mensaje de bloqueo
4	Reglas de negocio: relación cuota/ingreso y DTI dentro de umbral; si excede, “No cumple”.	Regla	Captura del cálculo/resultado
5	Transición de estado: al finalizar → “Validada”; con observaciones → “Observada”.	Funcional	Captura del estado
6	Auditoría: registra usuario, fecha y criterios evaluados.	Auditoría	Registro histórico / bitácora

CU14 – Aprobar/Rechazar Solicitud (Web)

Nº	Criterio	Tipo	Evidencia
1	Solo se puede decidir (aprobar/rechazar) solicitudes en estado Validada/Observada .	Regla	Mensaje de bloqueo si estado no válido
2	Aprobar cambia estado a Aprobada y registra usuario/fecha (y condiciones finales si aplica).	Funcional	Captura de detalle con nuevo estado
3	Rechazar cambia estado a Rechazada y exige motivo obligatorio.	Funcional	Captura del motivo y estado
4	Permisos : únicamente roles autorizados pueden decidir; otros reciben 403 .	Seguridad	Prueba con usuario sin rol
5	Auditoría : registra decisión, usuario, timestamp y resumen (sin datos sensibles).	Auditoría	Registro histórico / bitácora
6	Consistencia : el listado y el detalle reflejan la decisión inmediatamente.	Funcional	Captura del listado actualizado

CU15 – Generar Plan de Pago (Web)

Nº	Criterio	Tipo	Evidencia
1	Se puede generar plan a partir de una solicitud Aprobada (o ingresando monto, plazo, tasa, moneda).	Funcional	Captura formulario/generación
2	Cálculo de cuota (sistema francés) correcto; suma de intereses y capital cuadra con el monto.	Funcional	Captura tabla + verificación
3	Fechas : primera cuota según regla (p. ej., próximo mes mismo día); calendario consecutivo sin saltos.	Regla	Captura cabeceras de fechas
4	Redondeo : montos por cuota con 2 decimales; última cuota ajusta diferencia (centavos).	Regla	Captura última fila
5	Exportación del plan a PDF/Excel disponible y descargable.	Funcional	Archivo descargado
6	Auditoría : registra quién generó, cuándo y parámetros usados.	Auditoría	Registro histórico / bitácora

CU16 – Seguir Estado de Solicitud (Web / Móvil)

Nº	Criterio	Tipo	Evidencia
1	La vista muestra estado actual (p.ej., Registrada/En evaluación/Observada/Validada/Aprobada/Rechazada).	Funcional	Captura de la vista
2	Se visualiza línea de tiempo / historial con hitos y fechas.	Funcional	Captura timeline
3	Permisos: solo el propietario de la solicitud puede verla; terceros reciben 403.	Seguridad	Prueba con otro usuario
4	Actualización del estado en tiempo real o al refrescar (polling/WebSocket).	Funcional	Captura antes/después
5	Filtros/búsqueda por número de solicitud o estado .	Funcional	Captura filtros aplicados
6	Acceso a detalle (resumen, producto, monto, plazo) y a adjuntos visibles si el estado lo permite.	Funcional	Captura del detalle

CU17 – Ver Historial de Créditos (Web / Móvil)

Nº	Criterio	Tipo	Evidencia
1	La vista muestra el listado de créditos del usuario (nro, producto, monto, estado, fechas).	Funcional	Captura del listado
2	Permisos: el cliente solo ve sus créditos; personal autorizado puede filtrar por cliente.	Seguridad	Prueba con cliente y con rol staff
3	Filtros/Búsqueda por estado, rango de fechas y número de crédito.	Funcional	Captura de filtros aplicados
4	Detalle: al abrir un crédito se muestran datos clave y estado vigente.	Funcional	Captura del detalle
5	Paginación / carga incremental funciona y mantiene filtros activos.	Funcional	Captura antes/después
6	Auditoría: registra consulta (usuario, fecha, criterio de búsqueda).	Auditoría	Registro histórico / bitácora

U18 – Gestionar Productos Financieros (Web)

Nº	Criterio	Tipo	Evidencia
1	CRUD de productos (crear, editar, habilitar/deshabilitar, eliminar) disponible para roles autorizados.	Funcional	Capturas CRUD
2	Validaciones: nombre único, moneda permitida (BOB/USD), rangos de monto y plazo coherentes.	Funcional	Mensajes de validación
3	Tasas (TEA/tem) dentro de límites configurados y consistentes con el tipo de producto.	Regla	Captura de error/aceptación
4	Permisos: solo rol de administración puede gestionar; otros reciben 403 .	Seguridad	Prueba con rol no autorizado
5	Estado del producto: al deshabilitar deja de aparecer en catálogos de solicitud/simulación.	Funcional	Listado catálogos antes/después
6	Auditoría: registra quién, qué cambios y cuándo (antes/después).	Auditoría	Registro histórico / bitácora

CU19 – Cargar y Gestionar Documentos (Web/Móvil)

Nº	Criterio	Tipo	Evidencia
1	Subir documento válido (PDF/JPG, ≤ 5 MB) y asociarlo a la solicitud.	Funcional	Captura de carga correcta
2	Validaciones: tipo no permitido o tamaño excedido → mensaje y bloqueo.	Funcional	Mensajes de validación
3	Listar/Visualizar documentos de la solicitud (nombre, tipo, fecha).	Funcional	Captura del listado/preview
4	Reemplazar un documento (misma categoría) conservando el historial mínimo.	Funcional	Captura antes/después
5	Eliminar documento (si política lo permite) y actualizar listado.	Funcional	Captura listado actualizado
6	Permisos: solo propietario (cliente) o rol autorizado pueden gestionar; otros → 403.	Seguridad	Prueba con rol no autorizado
7	Auditoría: registra alta, reemplazo y baja (quién/cuándo).	Auditoría	Registro histórico / bitácora

2.4.2 Reporte de prueba

CU10 – Solicitar Crédito (Web / Móvil)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-10-01	Crear solicitud desde Web con datos válidos	Solicitud creada (201) y visible en listado		Pendiente	Captura CRUD
TC-10-02	Crear solicitud desde Móvil con datos válidos	Igual resultado que Web		Pendiente	Captura móvil
TC-10-03	Enviar formulario vacío / con monto fuera de rango	Se muestran mensajes de validación por campo		Pendiente	Mensajes de validación
TC-10-04	Sin login (o sesión expirada) intenta solicitar	Acceso denegado / redirección a login		Pendiente	Captura error
TC-10-05	Ver listado luego de crear	Nueva solicitud con estado inicial		Pendiente	Captura listado
TC-10-06	Ver auditoría de creación	Entrada con usuario, fecha y acción		Pendiente	Registro histórico

CU11 – Simular Crédito (Web / Móvil)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-11-01	Simular en Web con datos válidos (Consumo, 30.000 BOB, 24 m)	Muestra cuota, tasa y resumen		Pendiente	Captura resultado
TC-11-02	Simular en Móvil con datos válidos (USD, 12 m)	Igual comportamiento que Web		Pendiente	Captura móvil
TC-11-03	Campos vacíos / monto fuera de rango	Mensajes de validación por campo		Pendiente	Captura validación
TC-11-04	Verificación de cálculo de cuota (francés) vs cálculo manual	Coincidencia (± 1 centavo)		Pendiente	Captura fórmula
TC-11-05	Comparar dos plazos (12 vs 24 m)	Tabla/box comparativo con cuotas y totales		Pendiente	Captura comparativa
TC-11-06	Error al cargar catálogo de productos (simulado)	Mensaje amable + botón Reintentar		Pendiente	Captura retry

CU12 – Registrar Solicitud de Crédito (Web)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-12-01	Crear solicitud con datos válidos	Respuesta 201 y aparece en listado con estado inicial		Pendiente	Captura CRUD
TC-12-02	Enviar vacío / monto fuera de rango	Mensajes de validación por campo		Pendiente	Captura validación
TC-12-03	Cliente con solicitud activa del mismo producto	Bloqueo por duplicado		Pendiente	Mensaje/consulta
TC-12-04	Sin login o rol no permitido	401/403 y no registra		Pendiente	Captura error
TC-12-05	Ver listado luego de crear	Nueva fila con datos y estado inicial		Pendiente	Captura listado
TC-12-06	Ver auditoría de creación	Entrada con usuario, fecha y acción		Pendiente	Registro histórico

CU13 – Validar Información (Web)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-13-01	Abrir validación con Solicitud existente	Se muestran datos + checks		Pendiente	Captura vista
TC-13-02	Falta de documento requerido	Marca “Observado” y pide comentario		Pendiente	Captura adjuntos
TC-13-03	KYC/PEP con match	Bloquea cierre y muestra alerta		Pendiente	Mensaje bloqueo
TC-13-04	Cuota/Ingreso o DTI supera umbral	Resultado “No cumple”		Pendiente	Captura cálculo
TC-13-05	Cerrar validación sin observaciones	Estado → Validada		Pendiente	Captura estado
TC-13-06	Ver auditoría de validación	Entrada con usuario, fecha y reglas		Pendiente	Registro histórico

CU14 – Aprobar/Rechazar Solicitud (Web)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-14-01	Aprobar una solicitud en estado Validada	Estado → Aprobada , registra usuario/fecha		Pendiente	Captura roles
TC-14-02	Rechazar una solicitud (con motivo)	Estado → Rechazada + motivo obligatorio		Pendiente	Captura validación
TC-14-03	Intentar decidir con rol no autorizado	403 y sin cambios en la solicitud		Pendiente	Captura prueba
TC-14-04	Intentar decidir una solicitud no Validada/Observada	Bloqueo por estado no válido		Pendiente	
TC-14-05	Aprobar ajustando condiciones (p. ej., plazo/monto permitido por política)	Condiciones finales guardadas		Pendiente	
TC-14-06	Ver listado y auditoría tras la decisión	Listado muestra nuevo estado; auditoría con evento		Pendiente	

CU15 – Generar Plan de Pago (Web)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-15-01	Generar plan desde solicitud Aprobada	Muestra tabla con cuotas, fechas, capital, interés, saldo		Pendiente	Captura plan
TC-15-02	Verificar cálculo (monto 30.000, 24m, tasa X)	Cuota coincide con fórmula ($\pm 0,01$)		Pendiente	Captura verificación
TC-15-03	Regla de fechas (primera cuota próximo mes)	Fechas consecutivas correctas		Pendiente	Captura fechas
TC-15-04	Redondeo y ajuste en última cuota	Diferencia de centavos absorbida en la última		Pendiente	Captura última cuota
TC-15-05	Exportar a PDF y Excel	Archivos descargados y legibles		Pendiente	Archivos export
TC-15-06	Auditoría de la generación	Evento con usuario, fecha y parámetros		Pendiente	Bitácora

CU16 – Seguir Estado de Solicitud (Web / Móvil)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-16-01	Abrir seguimiento de una solicitud propia	Se muestra estado actual y resumen		Pendiente	Captura vista
TC-16-02	Revisar línea de tiempo	Hitos con fecha y orden cronológico		Pendiente	Captura timeline
TC-16-03	Intentar ver solicitud de otro usuario	403 / acceso denegado		Pendiente	Captura error
TC-16-04	Cambiar estado en backend (p.ej., a Validada) y refrescar	UI refleja el nuevo estado		Pendiente	Captura antes/después
TC-16-05	Filtrar por estado “Aprobada”	Lista solo con coincidencias		Pendiente	Captura filtros
TC-16-06	Abrir detalle y revisar adjuntos permitidos	Datos y adjuntos visibles según política		Pendiente	Captura detalle

CU17 – Ver Historial de Créditos (Web / Móvil)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-17-01	Cliente abre Historial en Móvil/Web	Lista solo sus créditos con columnas básicas		Pendiente	Captura listado
TC-17-02	Staff busca por cliente y estado=Vigente	Resultados filtrados según criterios		Pendiente	Captura filtros
TC-17-03	Cliente intenta ver historial de otro usuario (forzando ID)	403 / acceso denegado		Pendiente	Captura error
TC-17-04	Abrir detalle de un crédito	Se muestran datos clave y estado actual		Pendiente	Captura detalle
TC-17-05	Paginación a la segunda página con filtros activos	Mantiene filtros y muestra más resultados		Pendiente	Captura paginación
TC-17-06	Ver auditoría de consultas	Evento con usuario, fecha y criterio		Pendiente	Bitácora

CU18 – Gestionar Productos Financieros (Web)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-18-01	Crear producto “Consumo BOB” con rangos y tasas válidas	Producto creado y visible en listado		Pendiente	Captura creación
TC-18-02	Editar tasas y plazos de un producto existente	Cambios guardados y visibles		Pendiente	Captura edición
TC-18-03	Intentar crear con nombre duplicado o rango inválido	Mensajes de validación y bloqueo		Pendiente	Captura validación
TC-18-04	Deshabilitar producto y revisar catálogo en Solicitar/Similar	Producto no aparece en catálogos		Pendiente	Captura antes/después
TC-18-05	Intentar gestionar con rol no autorizado	403 y sin cambios		Pendiente	Captura error
TC-18-06	Ver auditoría de cambios (antes/después)	Evento con usuario, fecha		Pendiente	Registro histórico

CU19 – Cargar y Gestionar Documentos (Web/Móvil)					
Caso	Escenario	Resultado esperado	Resultado obtenido	Estado	Evidencia
TC-19-01	Subir PDF 2 MB desde Web/Móvil	Archivo asociado a la solicitud		Pendiente	Captura carga
TC-19-02	Subir EXE o >5 MB	Mensaje de validación y sin guardar		Pendiente	Captura validación
TC-19-03	Listar/Visualizar documentos	Aparecen con nombre, tipo y fecha; preview disponible		Pendiente	Captura listado
TC-19-04	Reemplazar documento “CI anverso”	Nuevo archivo visible; historial mínimo registrado		Pendiente	Captura antes/después
TC-19-05	Eliminar documento permitido por política	Se quita del listado sin errores		Pendiente	Captura listado
TC-19-06	Intentar gestionar con rol no autorizado	403 y sin cambios		Pendiente	Captura error
TC-19-07	Ver auditoría de operaciones	Eventos de alta/reemplazo/baja con usuario y fecha		Pendiente	Bitácora

3 Daily Scrum (o Scrum diario)

Luis Fernando Banegas

Fecha	¿Qué hiciste ayer?	¿Qué harás hoy?	¿Qué obstáculo encontraste en el camino?
19-sep	Revisión inicial del Sprint 2	Configurar envío de emails (SP2-07)	Ninguna
20-sep	Avance en configuración de emails	Probar envío de correos de prueba	Ninguna

21-sep	Resolver problema de spam	Revisión de código de login (SP2-16)	Ninguna
22-sep	Revisión de código de Daniela	Revisión de código de Alejandro	Ninguna
23-sep	Reunión de ajuste con el equipo	Documentar flujo de autenticación	Ninguna
24-sep	Revisión de CRUD de posts (SP2-11)	Preparar reunión de revisión de Sprint	Ninguna
25-sep	Revisión final de módulo de usuarios	Revisar integración con Bootstrap	Ninguna
26-sep	Reunión de revisión de Sprint (SP2-22)	Actualizar repositorio en GitHub	Ninguna
27-sep	Revisar y merge de pull requests	Soporte a Madeline en diseño de iconos	Ninguna
28-sep	Ayuda en configuración de despliegue	Revisar documentación técnica (SP2-18)	Ninguna
29-sep	Revisar pruebas unitarias de Daniela	Revisar optimización de BD (SP2-19)	Ninguna
30-sep	Revisar paginación (SP2-20)	Revisar panel de administración (SP2-09)	Ninguna

01-oct	Revisar recuperación de contraseña (SP2-08)	Revisar formulario de registro (SP2-05)	Ninguna
02-oct	Revisión general del frontend	Revisar despliegue en pruebas (SP2-17)	Ninguna
03-oct	Revisar y ajustar estilos con Bootstrap	Preparar demo final del Sprint	Ninguna
04-oct	Revisión final de funcionalidades	Actualizar documentación en GitHub	Ninguna
05-oct	Preparar presentación del Sprint 2	Soporte general al equipo	Ninguna

aikol Molina Cortez

Fecha	¿Qué hiciste ayer?	¿Qué harás hoy?	¿Qué obstáculo encontraste en el camino?
19-sep	Revisar modelos de usuarios (SP2-02)	Crear vistas para gestión de perfiles (SP2-06)	Ninguna
20-sep	Avance en vistas de perfil	Integrar templates con Bootstrap (SP2-13)	Ninguna

21-sep	Ajustar estilos de perfil	Crear templates para listar posts (SP2-12)	Ninguna
22-sep	Avance en listado de posts	Coordinar con Madeline para UI	Ninguna
23-sep	Revisar paginación con Alejandro	Implementar paginación en listas (SP2-20)	Ninguna
24-sep	Corregir estilos de paginación	Revisar documentación de la API	Ninguna
25-sep	Ayuda en pruebas unitarias	Revisar panel de admin (SP2-09)	Ninguna
26-sep	Revisar y ajustar CRUD de posts	Revisar recuperación de contraseña	Ninguna
27-sep	Revisar estilos de login	Revisar formulario de registro	Ninguna
28-sep	Revisar y probar envío de emails	Revisar configuración de CORS	Ninguna
29-sep	Revisar optimización de consultas	Revisar despliegue en pruebas	Ninguna
30-sep	Revisar y ajustar responsive	Soporte a Luis Pablo en BD	Ninguna
01-oct	Revisar y probar toda la app en local	Preparar versión para demo	Ninguna

02-oct	Revisar y ajustar detalles finales	Revisar y probar funcionalidades clave	Ninguna
03-oct	Revisar y actualizar README	Revisar presentación del Sprint	Ninguna
04-oct	Revisar y probar en diferentes navegadores	Revisar y ajustar permisos	Ninguna
05-oct	Revisión final antes de la entrega	Soporte en presentación final	Ninguna

Alejandro Melgar Arteaga

Fecha	¿Qué hiciste ayer?	¿Qué harás hoy?	¿Qué obstáculo encontraste en el camino?
19-sep	Revisar modelos de usuarios (SP2-02)	Implementar login y logout (SP2-04)	Ninguna
20-sep	Avance en lógica de login	Implementar recuperación de contraseña (SP2-08)	Ninguna
21-sep	Corregir generación de token	Revisar rutas y URLs (SP2-14)	Ninguna
22-sep	Revisar y ajustar todas las URLs	Revisar CRUD de posts (SP2-11)	Ninguna

23-sep	Revisar y corregir eliminación en cascada	Revisar paginación con Maikol	Ninguna
24-sep	Revisar y ajustar lógica de negocio	Revisar panel de administración (SP2-09)	Ninguna
25-sep	Revisar y agregar filtros en admin	Revisar optimización de BD (SP2-19)	Ninguna
26-sep	Revisar y agregar índices a modelos	Revisar configuración de CORS	Ninguna
27-sep	Revisar y ajustar settings de CORS	Revisar y probar autenticación en Postman	Ninguna
28-sep	Revisar y ajustar tiempo de expiración	Revisar y probar flujo completo de auth	Ninguna
29-sep	Revisar y corregir logout en segundo plano	Revisar y ajustar validaciones de modelos	Ninguna
30-sep	Revisar y ajustar modelos	Revisar y probar API en entorno de pruebas	Ninguna
01-oct	Revisar y corregir subida de archivos	Revisar y ajustar permisos de usuarios	Ninguna

02-oct	Revisar y ajustar permisos	Revisar y probar funcionalidades en móvil	Ninguna
03-oct	Revisar y ajustar responsive	Revisar y actualizar documentación de API	Ninguna
04-oct	Revisar y completar documentación	Revisar y probar en diferentes dispositivos	Ninguna
05-oct	Revisión final de lógica de backend	Soporte en pruebas finales	Ninguna

Madeline Flores Quispe

Fecha	¿Qué hiciste ayer?	¿Qué harás hoy?	¿Qué obstáculo encontraste en el camino?
19-sep	Revisar diseño de BD para posts (SP2-10)	Diseñar interfaz de usuario (UI) (SP2-01)	Ninguna
20-sep	Avance en diseño de UI	Diseñar formulario de registro (SP2-05)	Ninguna
21-sep	Ajustar diseño responsive del formulario	Integrar Bootstrap para estilos (SP2-13)	Ninguna

22-sep	Revisar y unificar estilos	Diseñar iconos y elementos gráficos (SP2-21)	Ninguna
23-sep	Avance en diseño de iconos	Coordinar con Maikol para templates	Ninguna
24-sep	Revisar y mejorar diseño de listas	Diseñar panel de administración	Ninguna
25-sep	Revisar y ajustar paleta de colores	Revisar y ajustar responsive en tablets	Ninguna
26-sep	Revisar y rediseñar tablas de admin	Revisar y ajustar diseño de emails	Ninguna
27-sep	Revisar y ajustar plantilla de email	Revisar y ajustar iconos de la app	Ninguna
28-sep	Revisar y estandarizar iconos	Revisar y ajustar diseño general	Ninguna
29-sep	Revisar y agregar estados hover a botones	Revisar y ajustar diseño de paginación	Ninguna
30-sep	Revisar y ajustar tamaño de botones	Revisar y ajustar diseño en móviles	Ninguna
01-oct	Revisar y corregir menú móvil	Revisar y ajustar tipografía	Ninguna

02-oct	Revisar y cambiar fuente web-safe	Revisar y ajustar contraste de colores	Ninguna
03-oct	Revisar y ajustar colores para accesibilidad	Revisar y preparar assets para demo	Ninguna
04-oct	Revisar y rehacer imágenes en alta calidad	Revisar y ajustar últimos detalles	Ninguna
05-oct	Revisión final de diseño en todos los dispositivos	Preparar guía de estilos	Ninguna

Luis Pablo Mendoza

Fecha	¿Qué hiciste ayer?	¿Qué harás hoy?	¿Qué obstáculo encontraste en el camino?
19-sep	Revisar configuración de autenticación (SP2-03)	Configurar conexión con la BD (SP2-05)	Ninguna
20-sep	Resolver error de conexión a BD	Revisar y optimizar consultas (SP2-19)	Ninguna
21-sep	Revisar y agregar índices a la BD	Revisar y ajustar modelos	Ninguna

22-sep	Revisar y recrear migraciones	Revisar y ajustar configuración de CORS	Ninguna
23-sep	Revisar y ajustar settings de CORS	Revisar y probar API en Postman	Ninguna
24-sep	Revisar y ajustar permisos de API	Revisar y ajustar autenticación JWT	Ninguna
25-sep	Revisar e implementar refresh token	Revisar y ajustar despliegue (SP2-17)	Ninguna
26-sep	Revisar y corregir variables de entorno	Revisar y ajustar configuración de email	Ninguna
27-sep	Revisar y ajustar servidor SMTP	Revisar y probar subida de archivos	Ninguna
28-sep	Revisar y ajustar límite de subida	Revisar y optimizar rendimiento	Ninguna
29-sep	Revisar e implementar compresión de imágenes	Revisar y ajustar backups de BD	Ninguna
30-sep	Revisar y corregir script de backup	Revisar y ajustar permisos de archivos	Ninguna
01-oct	Revisar y ajustar permisos en servidor	Revisar y probar recuperación de contraseña	Ninguna

02-oct	Revisar y ajustar tiempo de expiración	Revisar y probar funcionalidades en servidor	Ninguna
03-oct	Revisar y corregir error en admin	Revisar y ajustar logs de aplicación	Ninguna
04-oct	Revisar y ajustar sistema de logs	Revisar y preparar servidor para demo	Ninguna
05-oct	Revisión final de configuración de servidor	Soporte en puesta en producción	Ninguna

Daniela Perez Gomez

Fecha	¿Qué hiciste ayer?	¿Qué harás hoy?	¿Qué obstáculo encontraste en el camino?
19-sep	Revisar migraciones de BD (SP2-06)	Implementar login y logout (SP2-04)	Ninguna
20-sep	Corregir redirección de logout	Implementar CRUD de posts (SP2-11)	Ninguna
21-sep	Revisar y corregir actualización de posts	Crear panel de administración (SP2-09)	Ninguna
22-sep	Revisar y agregar búsqueda en admin	Revisar y ajustar templates de posts	Ninguna

23-sep	Revisar y mejorar formulario de posts	Revisar y ajustar validaciones de frontend	Ninguna
24-sep	Revisar y ajustar validaciones	Realizar pruebas unitarias (SP2-15)	Ninguna
25-sep	Revisar y corregir tests flaky	Revisar y ajustar paginación (SP2-20)	Ninguna
26-sep	Revisar y corregir paginación con búsqueda	Revisar y ajustar permisos de usuarios	Ninguna
27-sep	Revisar y corregir filtros por usuario	Revisar y ajustar manejo de errores	Ninguna
28-sep	Revisar y mejorar manejo de errores	Revisar y ajustar responsive general	Ninguna
29-sep	Revisar y ajustar formularios responsive	Revisar y probar flujo completo de la app	Ninguna
30-sep	Revisar y corregir publicación sin imagen	Revisar y ajustar tiempos de carga	Ninguna
01-oct	Revisar y optimizar carga de imágenes	Revisar y ajustar notificaciones al usuario	Ninguna
02-oct	Revisar y agregar mensajes de feedback	Revisar y probar en diferentes navegadores	Ninguna

03-oct	Revisar y corregir datepicker en Firefox	Revisar y ajustar accesibilidad	Ninguna
04-oct	Revisar y agregar etiquetas ARIA	Revisar y preparar versión para demo	Ninguna
05-oct	Revisión final de funcionalidades de frontend	Soporte en presentación final	Ninguna

4 Sprint Review

REVISION DEL SPRING 1	
Objetivos del Sprint: <ul style="list-style-type: none"> Habilitación de Clientes: Implementar la Simulación de Crédito y el Registro Formal de Solicitudes tanto en la aplicación móvil como en la web. Captura Documental: Desarrollar el módulo completo para la Carga, Almacenamiento y Gestión Digital de todos los documentos requeridos por el cliente. Gestión del Flujo: Crear la interfaz web para que el Oficial de Crédito pueda Gestionar y Revisar el estado de todas las solicitudes ingresadas. Pre-Evaluación: Integrar las funciones de Validación de la Información, Consulta al Buró de Crédito y la Generación del Plan de Pagos preliminar. Mantenimiento: Implementar la interfaz de Administración de Productos Financieros para configurar tasas, montos y tipos de crédito. 	
PARTICIPANTES	
NOMBRE	ROL
Perez Gomez Daniela	Scrum Master
Molina Cortez Maikol Anthony	Product owner
Banegas Roca Luis Fernando	Team Development
Flores Quispe Nathaly Madeline	Team Development
Melgar Arteaga Alejandro	Team Development
Mendoza Rodriguez Luis Pablo	Team Development
PRESENTACION DEL INCREMENTO	
FUNCIONES PRESENTADAS (Elemento de trabajo a presentar)	RETROALIMENTACION
CU10: Solicitar crédito	Notificación de Registro: Solicitud creada con éxito y pasa a estado Pendiente de Documentación.
CU11: Simular crédito	Muestra el Plan de Pagos Estimado (monto de cuota, interés total, plazo).
CU12: Registrar solicitud de crédito	Confirmación de Alta: Se genera el ID de solicitud y el estado inicial es Pendiente de Revisión.
CU13: Validar información	Alerta al Oficial: Documentación Completa y Válida O Incompleta/Requiere Corrección (lo que podría generar un email al cliente).
CU14: Aprobar / rechazar solicitud	Transición de Estado: Estado de Solicitud cambia a Pre-Aprobado (pasa a fase IA) O Rechazo Preliminar (si la validación fue negativa).
CU15: Generar plan de pagos	Generación Exitosa del Plan de Pagos (el expediente queda marcado como "Técnicamente Apto" para la fase IA).

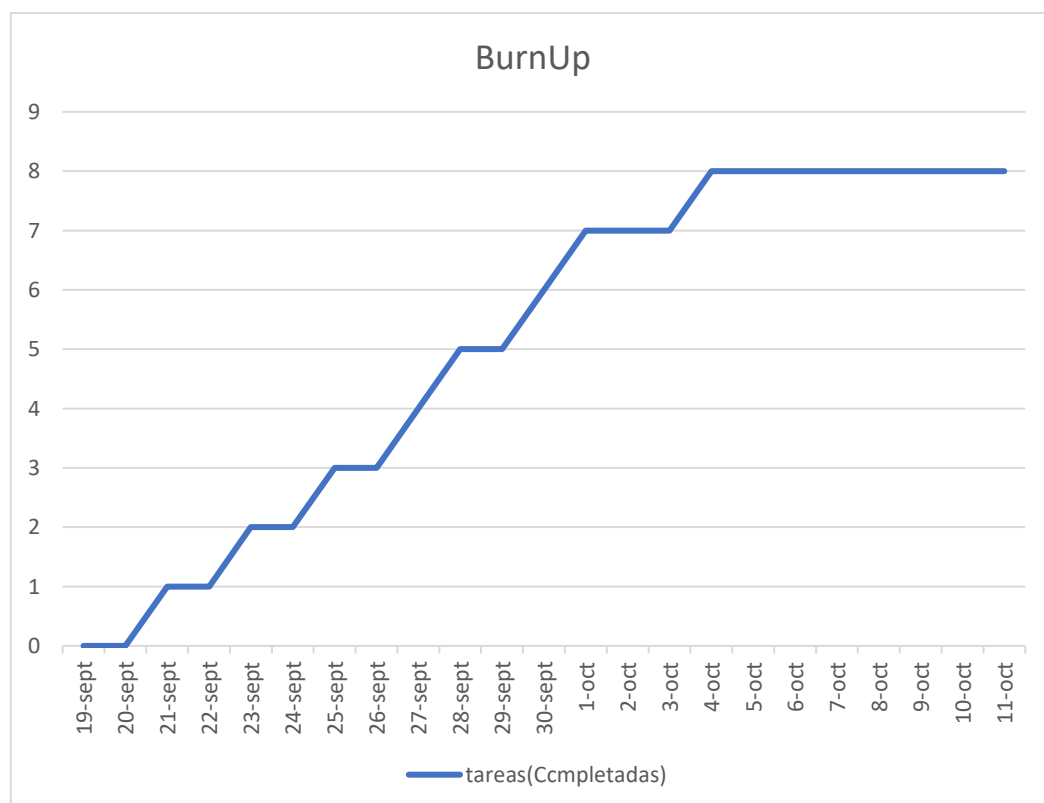
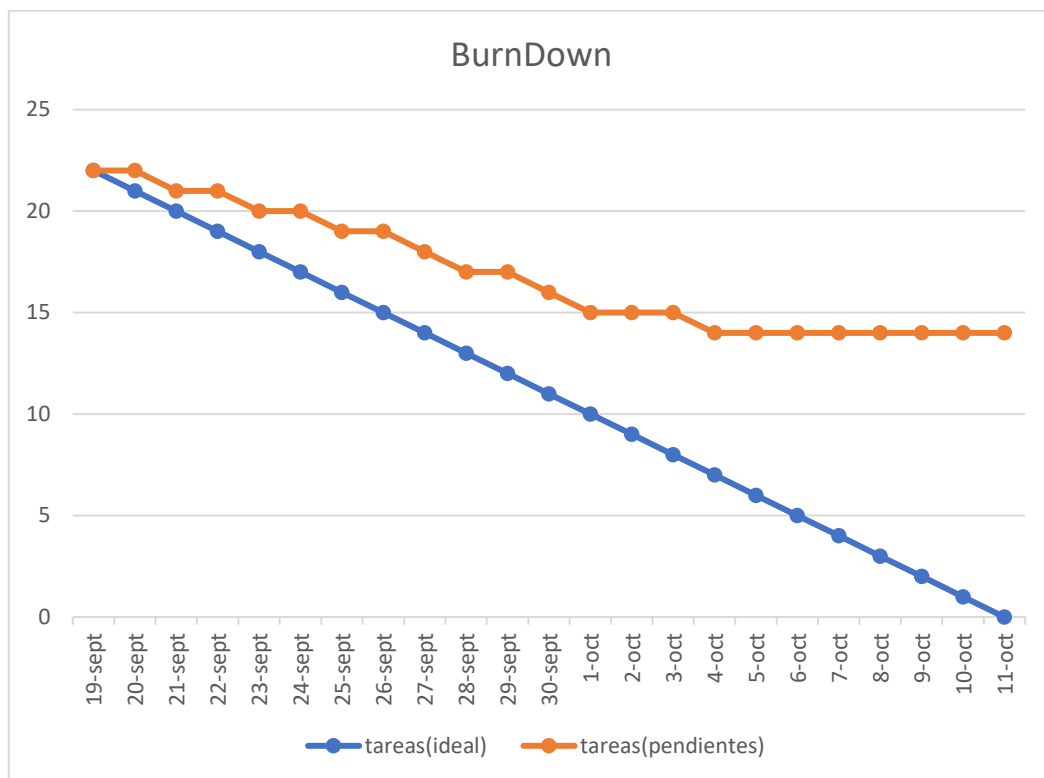
CU16: Seguir estado de solicitud	Actualización de Estado en Tiempo Real: ("En Revisión Documental", "Lista para Evaluación IA", "Rechazo Preliminar").
CU17: Ver historial de créditos	Muestra el historial completo de solicitudes anteriores del cliente dentro del sistema.
CU18: Gestionar productos financieros	Producto Financiero Creado/Actualizado con Éxito y disponible para nuevas simulaciones y solicitudes.
CU19: Cargar y gestionar documentos	Confirmación de Carga: Archivos subidos y validados contra formato O Error de Formato/Tamaño.

5 Sprint Retrospective

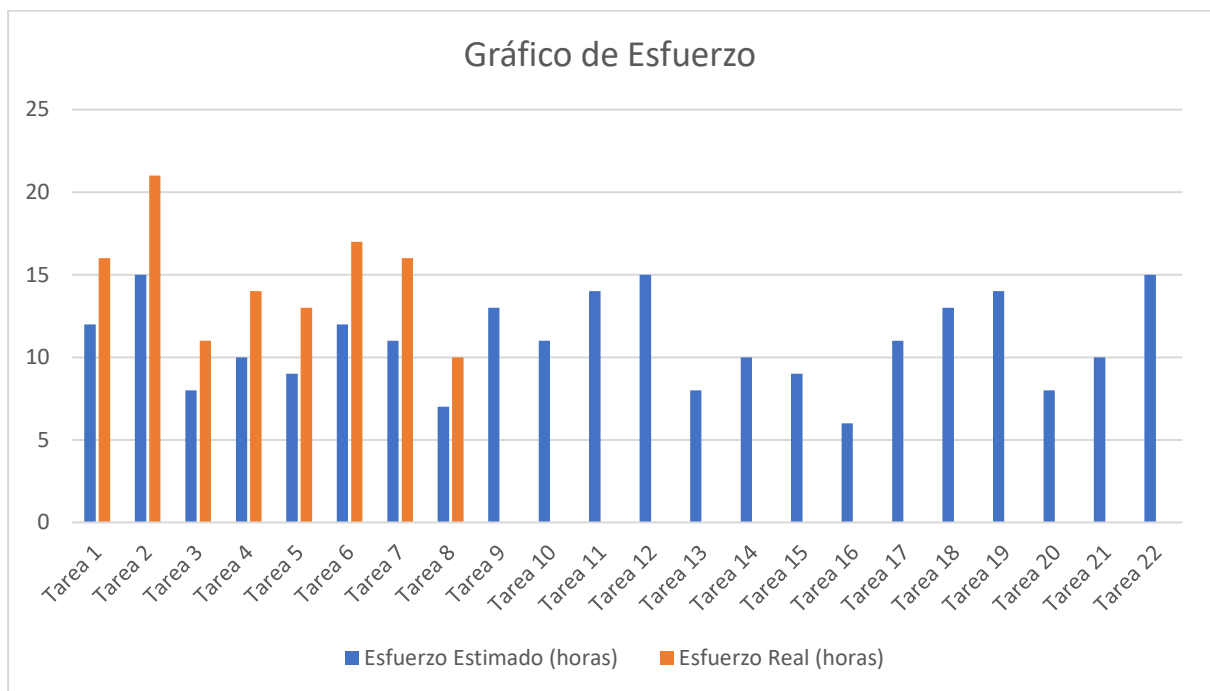
Retrospectiva del Sprint 2			
Fecha:		05/10/2025	
Facilitador:		Molina Cortez Maikol Anthony	
Objetivo		Evaluar el proceso del desarrollo del Sprint 2	
Nombre de Asistentes:		<ul style="list-style-type: none">Banegas Roca Luis FernandoFlores Quispe Nathaly MadelineMelgar Arteaga AlejandroMendoza Rodríguez Luis PabloPérez Gómez Daniela	
<ul style="list-style-type: none">Temas a Tratar:		<ul style="list-style-type: none">Flujo extremo a extremo de crédito (CU10→CU19): handoffs, estados y tiempos de paso entre módulos.Consistencia de estados (CU12, CU13, CU14, CU16): definiciones, transiciones válidas y sincronización en listado/timeline.Reglas de negocio clave (CU11, CU13, CU14, CU15): cálculo de cuota/redondeo y motivos de rechazo.Catálogo de productos (CU18): impacto de cambios (tasas/rangos/monedas) en Solicitar y Simular (CU10–CU11–CU12).Experiencia Móvil vs Web (CU10, CU11, CU16, CU19): diferencias de UX, tiempos de carga y validaciones.Gestión de documentos (CU19): tipos/tamaños, reemplazo, borrado y trazabilidad del historial.Permisos y seguridad (todos, foco CU18–CU19): roles efectivos, 401/403 y exposición de datos del cliente.Auditoría/bitácora (CU12–CU19): qué eventos faltan, granularidad y consulta rápida para soporte.Integraciones externas (CU13): estabilidad de KYC/buró, tiempos de respuesta y estrategias de reintento.Exportaciones y evidencias (CU15): PDF/Excel del plan de pagos y formato requerido por stakeholders.	
Discusión			
Nro.	¿Qué salió bien?	¿Qué no salió bien?	¿Qué haremos de manera diferente?
1	CU10–CU12 operativos en Web; flujo de registro de solicitud estable.	Inconsistencias de estados entre detallan/listado al cambiar de Validada→Aprobada.	Definir máquina de estados única y usarla en API y front; agregar pruebas unitarias de transición.

2	Simulador (CU11) reutiliza componente de cálculo en Móvil/Web.	Diferencias de redondeo en cuotas y última cuota.	Centralizar la fórmula en un helper compartido y prueba de snapshot de plan (CU15).
3	Catálogo de productos (CU18) con creación/edición funcionando.	Cambios de tasas/rangos no se reflejaron de inmediato en Simular/Solicitar.	Implementar cache con invalidación por evento “producto actualizado” y refresco en front.
4	Gestión de documentos (CU19): subir/visualizar/reemplazar.	Lento al subir >3 archivos y algunos formatos rechazados sin mensaje claro.	Activar carga asíncrona con barra de progreso y validar tipo/tamaño con mensajes breves.
5	Seguimiento de estado (CU16) con timeline visible.	WebSocket/polling no siempre actualiza; usuarios ven estado atrasado.	Unificar en polling cada 15s con etag/last-modified ; alertar si hay reconexión pendiente.
6	Permisos básicos aplicados (roles cliente/staff).	Casos con 401/403 mezclados y rutas sin proteger.	Revisar matriz RBAC y aplicar guard único en router + tests de permisos por CU.
7	Plan de pruebas corto aplicado a CU10–CU13.	Faltó evidencia estándar (nombres de archivos y ruta) y algunos TC sin ID.	Estandarizar nomenclatura (SS/NET/LOG/VID) y plantilla de reporte con numeración TC-xx.
8	Daily y coordinación mejoraron bloqueos.	Aún hay tareas que entran sin datos de esfuerzo/criterios.	En Planning, exigir Definition of Ready (criterios, esfuerzo, dependencias) antes de mover a <i>Doing</i> .

6 Burndown y BurnUp (Gráfica de tareas y Datos de tareas)



7 Gráfica de esfuerzo y Datos de esfuerzo



8 Scrum TaskBoard (Backlog, to do, doing, done)



Sprint 3

Sprint 4

Bibliografía

Libros/literatura papel/digital

1. **Schwaber, K. & Sutherland, J. (2020).** *The Scrum Guide: The Definitive Guide to Scrum*. Scrum.org.
→ Fuente oficial de la metodología ágil Scrum, base para el desarrollo del proyecto mediante sprints.
2. **Rumbaugh, J., Jacobson, I. & Booch, G. (2005).** *Lenguaje Unificado de Modelado: Manual de Referencia* (2da ed.). Addison-Wesley.
→ Manual técnico para el modelado UML, aplicado a casos de uso, diagramas de clases y modelado del sistema.
3. **O'Brien, J. A. & Marakas, G. M. (2011).** *Fundamentos de los Sistemas de Información Basados en Computadoras* (15a ed.). McGraw-Hill.
→ Soporte teórico sobre los componentes de un SIBC (hardware, software, datos, procesos y personas).
4. **Pressman, R. S. & Maxim, B. R. (2021).** *Ingeniería del Software: Un Enfoque Práctico* (9na ed.). McGraw-Hill.
→ Referencia sobre procesos de desarrollo, gestión de requisitos y buenas prácticas de ingeniería de software.
5. **Sommerville, I. (2016).** *Ingeniería del Software* (10ma ed.). Pearson.
→ Recurso fundamental para la conceptualización de sistemas de información y metodologías de desarrollo.
6. **Bertalanffy, L. von (1986).** *Teoría General de los Sistemas* (edición revisada). Fondo de Cultura Económica.
→ Referencia para comprender el proyecto como un sistema organizado con entradas, procesos y salidas, aplicable al enfoque de sistemas financieros.

7. **Date, C. J. (2019).** *Introducción a los Sistemas de Bases de Datos* (8va ed.). Pearson.
→ Fuente clave para el diseño y gestión de bases de datos relacionales (aplicado al uso de PostgreSQL en el proyecto).
8. **Goodfellow, I., Bengio, Y. & Courville, A. (2016).** *Deep Learning*. MIT Press.
→ Sustento académico para el uso de inteligencia artificial en la evaluación de solvencia crediticia.

Sitios web especializados

1. **Scrum.org. (2024).** *The Scrum Guide*. Disponible en: <https://www.scrum.org/resources/scrum-guide>
→ Fuente oficial de Scrum, referencia principal para la metodología ágil que aplicamos en el proyecto.
2. **Django Software Foundation. (2024).** *Django Documentation*. Disponible en: <https://docs.djangoproject.com/>
→ Documentación oficial del framework Django, utilizado como base del backend en el sistema.
3. **PostgreSQL Global Development Group. (2024).** *PostgreSQL Documentation*. Disponible en: <https://www.postgresql.org/docs/>
→ Documentación oficial del gestor de base de datos PostgreSQL, empleado para el almacenamiento de información financiera.
4. **Celery Project. (2024).** *Celery Documentation*. Disponible en: <https://docs.celeryq.dev/>
→ Guía oficial de Celery, herramienta usada para tareas asíncronas como notificaciones y procesamiento en segundo plano.
5. **Redis Labs. (2024).** *Redis Documentation*. Disponible en: <https://redis.io/docs/>
→ Referencia oficial de Redis, sistema de colas utilizado junto con Celery en el proyecto.
6. **Amazon Web Services (AWS). (2024).** *AWS Documentation*. Disponible en: <https://docs.aws.amazon.com/>
→ Documentación oficial de AWS, referente para el despliegue del sistema en la nube (IaaS y SaaS).
7. **ASFI – Autoridad de Supervisión del Sistema Financiero. (2024).** *Normativa Vigente*. Disponible en: <https://www.asfi.gob.bo/>
→ Fuente oficial de regulación boliviana sobre créditos y requisitos financieros, utilizada como base para reglas de negocio.
8. **INFOCRED Bolivia. (2024).** *Central de Riesgo Crediticio*. Disponible en: <https://www.infocred.bo/>

→ Plataforma oficial de historial crediticio, tomada como referencia para el análisis de solvencia (simulada en el proyecto).

9. **OWASP Foundation. (2024).** *OWASP Top Ten*. Disponible en:

<https://owasp.org/www-project-top-ten/>

→ Referencia en seguridad de aplicaciones web, útil para reforzar la sección de seguridad en el sistema financiero.

Anexos

Caso de Estudio 1 – Banco Prodem de Bolivia

Título

Plataforma web y móvil para la gestión de microfinanzas y servicios financieros en Banco Prodem de Bolivia

Objetivo

Diseñar e implementar una plataforma digital integral para el Banco Prodem que permita a los clientes —principalmente microempresarios, emprendedores y población en áreas rurales— acceder a productos financieros (cuentas, microcréditos, ahorros), realizar transacciones en línea, gestionar pagos y recibir atención personalizada. El objetivo contempla mejorar la inclusión financiera, fortalecer la cercanía del banco con sus clientes y garantizar seguridad, trazabilidad y cumplimiento de la normativa boliviana (ASFI).

Alcance

1. Servicios para clientes:

- Apertura y consulta de cuentas de ahorro y depósitos.
- Solicitud y seguimiento de microcréditos y créditos productivos.
- Transferencias nacionales y pagos de servicios.
- Acceso a plataformas de educación financiera.

2. Servicios administrativos:

- Gestión de clientes, roles y permisos.
- Reportes financieros e indicadores de inclusión financiera.
- Auditoría y monitoreo de transacciones.

3. Integraciones externas:

- Pagos con QR y billeteras móviles.

- Integración con plataformas de pagos de servicios básicos.
- Conexión con sistemas de remesas y transferencias internacionales.

4. Atención al cliente:

- Chat en línea y formulario de contacto.
- Sistema de reclamos y tickets.
- Mapa de sucursales y agentes corresponsales.

Funcionalidades y Prototipos

1. Cuentas y productos financieros: Consulta de saldos y movimientos, solicitud de nuevos productos.

2. Créditos y microfinanzas: Simulador de microcréditos, solicitud en línea y seguimiento de cuotas.

3. Pagos y transferencias: Transferencias interbancarias, pagos de servicios básicos y facturación electrónica.

4. Atención al cliente: Chat en vivo, ticket de reclamos, preguntas frecuentes.

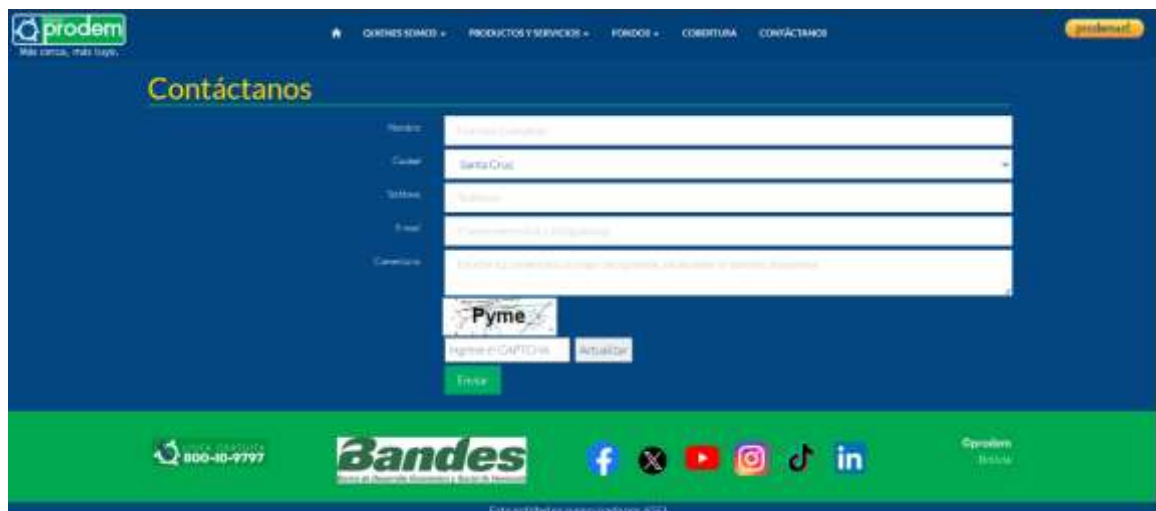
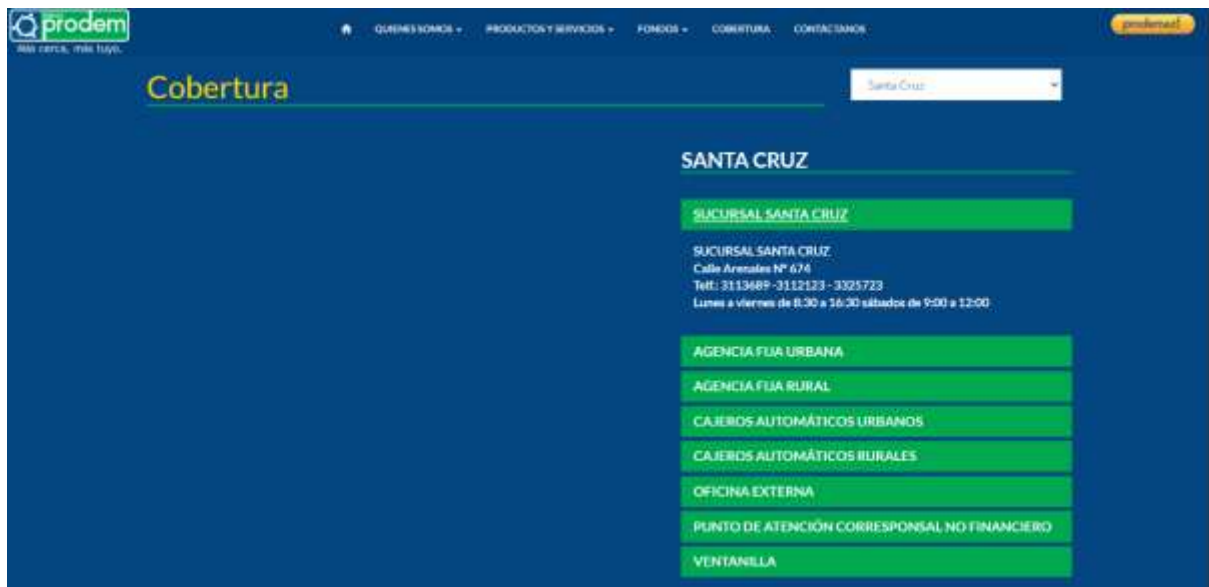
5. Administración del sistema: Gestión de usuarios internos, reportes de inclusión financiera, auditoría de seguridad.

6. Integraciones externas: Pagos con QR y billeteras móviles, interoperabilidad con fintechs y corresponsales no bancarios.

Prototipos (Referencias)

Los prototipos visuales incluyen capturas de la página oficial del Banco Prodem, tales como:

- Pantalla de inicio con acceso a ProdemNet.
- Sección de productos y servicios.
- Vista de cobertura y agencias.
- Formulario de contacto.
- Promociones y campañas vigentes.
- Sección de ahorro y cuentas ProCuenta.



prodem
Más cerca, más tuyo.

QUIÉNES SOMOS • PRODUCTOS Y SERVICIOS • FONDOS • COBERTURA • CONTACTAMOS

CUANDO LLUEVE EN PRODEM GANAN TODOS

¿Cómo puedes GANAR?

La promoción empresarial "CUANDO LLUEVE EN PRODEM GANAN TODOS V2" está dirigida a todos nuestros clientes de caja de ahorro bajo el producto PRO-CUENTA.

Las cajas de ahorro que están en parte de la promoción empresarial, cuentan con las siguientes características:

- Tipo de persona: NATURAL
- Moneda: BOLIVIANOS
- Tipo de manejo: INDIVIDUAL, BIENESTAR Y CONJUNTO
- Tipo de uso: COMPROMISANTE / TARJETA INTELIGENTE
- Estado: ACTIVO / INACTIVO

Periodo de duración

- Inicio: 01 de mayo de 2025.
- Fin: 31 de mayo de 2026.

Puedes acumular tus tickets hasta el 30 de abril de 2026.

Modalidad de premiación

La promoción empresarial "CUANDO LLUEVE EN PRODEM GANAN TODOS V2", cuenta con la modalidad de premiación SORTEO.

Modalidad de premiación

prodem
Más cerca, más tuyo.

QUIÉNES SOMOS • PRODUCTOS Y SERVICIOS • FONDOS • COBERTURA • CONTACTAMOS

Directorio y Fundadores
Historia
Misión y Visión

Historia

Sobre Nosotros

El día 18 de agosto de 1998, se reunió la Junta, organizó y constituyó el Fondo Financiero Privado Prodem S.A., mediante acto delos, el mismo que tiene por objeto principal la realización de operaciones destinadas al financiamiento de las actividades de los miembros, pequeños y microempresarios, a cuyo fin la sociedad podrá efectuar todas las operaciones pasivas, contingentes y de servicios financieros autorizadas por ley, en el territorio del Estado Plurinacional de Bolivia, con prioridad en el área rural.

La Superintendencia de Bancos y Entidades Financieras (actualmente ASFI) en el uso de sus facultades, el 19 de abril de 1999, mediante Resolución 58197-AS/99, confiere al permiso de Constitución a los Fundadores del Fondo Financiero Privado Prodem S.A., como entidad financiera privada nacional no bancaria, para realizar operaciones previstas en sus estatutos, sujetas a los límites establecidos por la Ley de Bancos y Entidades Financieras de Bolivia.

Adicionalmente, el 28 de diciembre de 1999, la Superintendencia de Bancos y Entidades Financieras (actualmente ASFI), de conformidad con el artículo 36 de la Ley de Bancos y Entidades Financieras otorga la Licencia de Funcionamiento al Fondo Financiero Privado Prodem S.A., para que inicie sus operaciones en forma simultánea en sus oficinas regionales y agencias a partir de 3 de enero de 2000.

El Fondo Financiero Privado Prodem S.A., opera en Bolivia a través de su oficina principal y oficinas en La Paz, Santa Cruz, Cochabamba, Oruro, Chuquisaca, Tarija, Potosí, Beni y Pando, distribuida en una red de 34 agencias urbanas y 69 agencias rurales.

El promedio de capital del Fondo Financiero Privado Prodem S.A., ha sido de 2,604, 241 y 2,340 al 31 de diciembre de 2013, 2014 y 2015, respectivamente.

prodem
Más cerca, más tuyo.

QUIÉNES SOMOS • PRODUCTOS Y SERVICIOS • FONDOS • COBERTURA • CONTACTAMOS

Procesos Empresariales
Créditos
Ahorro
Servicios
Prodemont
Boletín de Gestión

Ahorro

CAJA DE AHORRO
hasta

6%

Hace más de 5 años,
la tradición más
interesante
del mercado.

prodem

DPF hasta

9%

Hace más de 5 años,
la tradición más
interesante
del mercado.

prodem

Caja de Ahorro PRO CUENTA

Cuenta de Caja de Ahorro en bolivianos o dólares americanos

Una cuenta accesible, segura y sencilla, que te permite ahorrar y valorizar tu dinero de forma eficiente.

Ventajas:

• **Libre disponibilidad:** Ahorra y gana intereses con la libertad de retirar o depositar tu dinero cuando lo necesitas, sin ningún mínimo de depósito y con transacciones

<https://www.prodem.bo>

Caso de Estudio 2 – Banco Economico

Workflow de Aprobación de Créditos Bancarios mediante IA en Banco

a) Título

“Implementación de un Workflow Inteligente para la Automatización de Aprobación de Créditos en Baneco”

b) Objetivo

Diseñar e implementar un sistema de información basado en IA que permita a Banco automatizar el proceso de aprobación de créditos bancarios, reduciendo tiempos de espera, aumentando la precisión en la evaluación de la solvencia de clientes y mejorando la experiencia digital del usuario.

c) Alcance

El proyecto abarca:

- Clientes de Banco que soliciten créditos de consumo, microcréditos o créditos personales.
- Digitalización del proceso de aprobación de créditos mediante un flujo (workflow)

automatizado que evalúe los requisitos básicos:

- Identidad (CI válida).
 - Historial crediticio (ASFI/INFOCRED).
 - Antigüedad laboral o de negocio.
 - Ingresos mínimos y capacidad de endeudamiento.
- Uso de IA para:
- Predecir solvencia y riesgo crediticio en base a patrones históricos.
 - Recomendar montos adecuados según el perfil del solicitante.
 - Detectar inconsistencias o posibles fraudes.
- Cobertura geográfica inicial: Santa Cruz de la Sierra, con posibilidad de escalar a nivel nacional.

d) Prototipo (conceptual)

1. Ingreso de solicitud:

- Cliente accede a la web/app de Baneco o escanea un QR Crédito.
- Completa formulario digital (datos personales, ingresos, empleo/negocio, monto solicitado).

2. Validación automática:

- El sistema cruza datos con bases externas (ASFI, INFOCRED, impuestos, AFP).
- Se aplica un motor de IA de scoring crediticio entrenado con datos históricos de Baneco.

3. Decisión preliminar automática:

- Aprobado inmediato: si cumple todos los criterios y el riesgo es bajo.
- Rechazado automático: si incumple criterios básicos (ej. antigüedad insuficiente).
- En revisión: si hay inconsistencias, derivado a un oficial de crédito.

4. Respuesta al cliente:

- El sistema genera un resultado en minutos: monto aprobado, condiciones, o recomendaciones alternativas.

5. Notificación:

- Cliente recibe la respuesta vía app, correo o SMS.

Banco ECONOMICO PERSONAS MYPE EMPRESAS

Créditos Tarjetas Ahorro Seguros Servicios CLUB DE BENEFICIOS BANCA POR INTERNET

ZA\$ COBROS EN SEGUNDOS

¡PRUEBA HOY

LA APLICACIÓN MÁS RÁPIDA Y COMPLETA!

CONOCE MÁS **AQUÍ**



Cuenta Premium

Soy Premium

Por solo largo beneficios exclusivos

VER MÁS

Tarjeta Personalizada

Personaliza tu Tarjeta de Débito con tu foto favorita.

VER MÁS

Descuentos

Ahora los descuentos de tu Tarjeta de Débito.

VER MÁS

Tarjeta Digital

Tu Tarjeta siempre a la mano en tu celular.

VER MÁS

¡IMPULSA TU FUTURO CON EL CRÉDITO ECOLÓGICO!



QR Crédito

¿Sin efectivo?
¡Con QR CRÉDITO tienes la solución!
La primera línea de crédito de libre disponibilidad para comprar con QR lo que quieras cuando quieras.

Solicítalo aquí según tu situación laboral:

SOY ASALARIADO

¿Trabajas como dependiente de una empresa?

SOLICÍTALO AQUÍ

SOY PROFESIONAL INDEPENDIENTE

¿Eres un profesional independiente? Ejemplo: Abogado, Arquitecto, Psicólogo, Consultor, Médico, etc.

SOLICÍTALO AQUÍ

TENGO UN NEGOCIO MYPE

¿Tienes tu negocio propio?

SOLICÍTALO AQUÍ

¿Te ha pasado que se te presenta una oportunidad o emergencia y requieres efectivo?

A veces una necesidad efectiva para distintas situaciones, como cuando tu TV se daña y necesitas comprar una nueva, o una laptop nueva, o zapatos nuevos, u otro tipo de gastos, o mantenimiento mecánico de vehículo, entre otros.

¡Con QR CRÉDITO tienes la solución! Es una línea de crédito rotativa de libre disponibilidad que permite cubrir todo tipo de consumos, en todo lugar donde acepten pagos con QR.



Banco ECONOMICO Créditos Tarjetas Ahorro Seguros Servicios CLUB DE BENEFICIOS BANCA POR INTERNET

Paso 1 Paso 2 Paso 3 Paso 4

SOLICITA TU CRÉDITO

¡Calcula hasta cuánto puedes acceder en el préstamo que necesitas y solicítalo ahora! Aprovecha nuestros créditos en bolívianos para dependientes a partir de un año de antigüedad.

¿Cuáles son tus ingresos mensuales (tu sueldo más los ingresos de tu cónyuge) en Bs.? *

(Coloca el monto que recibes menos todos los descuentos de ley)

Ingresos

Aporta a la Gestora *

Aporta a la Gestora

¿Tienes otros créditos en entidades financieras? *

Otros créditos

¿Tienes otros créditos en Casa Comerciales? *

Otros créditos

¿Cuál es el monto específico que deseas solicitar en Bs.? *

Monto Requerido

CLUB

CALCULAR CRÉDITO **LIMPIAR**

e) Funcionalidades

- Portal de solicitud en línea: acceso desde web, app o código QR.
- Autenticación y seguridad: validación biométrica o token SMS.
- Workflow automatizado: pasos definidos para verificar documentos, validar requisitos y aplicar reglas de negocio.
- Módulo de IA de scoring: análisis de riesgo crediticio en tiempo real.
- Gestión de expedientes digitales: cada cliente tiene un expediente electrónico con todos los datos cargados.
- Notificaciones automáticas: aprobación, rechazo o solicitud de documentación adicional.
- Reportes y analítica: métricas de aprobación, tiempos de respuesta, perfil de clientes aprobados/rechazados.

d) Url:

<https://www.baneco.com.bo/>

Caso de estudio 3 - Banco BCB

Título

Sistema para la Gestión de Solicitudes de Crédito y digitalización de tramites financieros del BCB

Objetivo

Implementar un sistema digital eficiente, seguro y transparente para la recepción, evaluación y seguimiento de solicitudes de crédito en el Banco Central de Bolivia, optimizando procesos administrativos, reduciendo tiempos de atención y facilitando el acceso a financiamiento institucional.

Alcance

funcionalidades principales:

Workflow automatizado: Gestión integral del ciclo de solicitud de crédito (registro, evaluación, aprobación/rechazo).

Registro de solicitudes: Vía aplicación móvil para clientes o plataforma web para asesores.

Carga de documentos digitales:

- Identificación personal
- Estados financieros
- Datos de AFP
- Comprobantes de ingresos

Análisis con IA:

- Verificación de historial crediticio
- Cruce de datos con sistemas de AFP e impuestos
- Predicción de capacidad de pago

Enrutamiento inteligente: Asignación automática al oficial de crédito correspondiente

Sistema de notificaciones: Comunicación en tiempo real al cliente sobre el estado de su solicitud

Paneles de seguimiento:

- Para clientes: consulta del estado de trámites
- Para administradores: indicadores de riesgo y tiempos de aprobación

Seguridad y validación:

- Autenticación digital mediante credenciales únicas
- Firma electrónica con certificados digitales autorizados

- Integración con sistemas de verificación de identidad

Prototipo

Plataforma web que incluye:

- Módulo de registro de usuarios
- Interfaz para carga de documentación requerida
- Sistema de firma electrónica integrado
- Panel de seguimiento de estado de trámites
- Módulo administrativo para gestión de solicitudes

URL: <https://www.bcb.gob.bo/>

QR del Repositorio (GitHub)

