**PONTIFICIA UNIVERSIDAD JAVERIANA**

**INGENIERÍA DE SISTEMAS**

**FUNDAMENTOS DE INGENIERÍA DE SOFTWARE**

**GESTIÓN DE BANCO: PMP**

Imagen que contiene Patrón de fondo

El contenido generado por IA puede ser incorrecto.

**INTEGRANTES:**

**Santiago Andrés Mesa N.**

**Luis Enrique Santos Marulanda**

**Damián Rey Salcedo**

**Juan Diego Barreto Castañeda**

**PROFESOR:**

**Jaime Andrés Pavlich Mariscal.**

# Historial de Cambios

|  |  |  |
| --- | --- | --- |
| Fecha del cambio | Descripción del cambio | Responsable |
| 1/03/2025 | Creación básica del documento | Santiago Mesa |
| 1/03/2025 | Propuesta del proyecto | Santiago Mesa |
| 1/03/2025 | Organigrama | Santiago Mesa |
| 1/03/2025 | Modelo de Ciclo de Vida | Santiago Mesa |
| 1/03/2025 | Ambiente de trabajo | Santiago Mesa |
| 3/03/2025 | Tabla de Contenidos | Damian Rey |
| 3/03/2025 | Inicio introducción | Damian Rey |
| 3/03/2025 | Interfaces Externas | Damian Rey |
| 6/03/2025 | Monitoreo y Control de Progreso | Damian Rey |
| 9/03/2025 | Lenguajes y Herramientas | Luis Santos |
|  |  |  |

# Tabla de Contenidos

1. Introduccion  
2. Propuesta de Proyecto  
3. Administración del Proyecto  
 3.1 Modelo de Ciclo de Vida  
 3.2 Lenguajes y Herramientas  
 3.3 Interfaces Externas  
 3.4 Organigrama y Descripción de Roles  
 3.5 Calendarización y Entregables  
4. Procesos de Soporte  
 4.1 Ambiente de Trabajo  
 4.2 Monitoreo y Control de Progreso  
 4.3 Administración de Configuración y Documentacion  
 4.4 Control de Calidad  
5. Referencias

# Introducción

En este documento se detallará el desarrollo de un sistema de gestión de banco realizado por un grupo conformado de 4 integrantes.

Primero se introducirá la propuesta general del proyecto, luego se profundizará más en la administración de este, mostrando diversos detalles como el modelo de ciclo de vida, los lenguajes y herramientas que se usaran al transcurso del desarrollo del sistema, las interfaces externas que se utilizaran, un organigrama y descripción de roles de los miembros del grupo de desarrollo, y calendarización y entregables esperados.

Luego se hablará de los procesos de soporte tales como el ambiente del trabajo, monitoreo y control de progreso, administración de configuración y documentación y control de calidad del proyecto en general.  
~Expandir más~

# Propuesta de proyecto

**Propósito del Proyecto**

Este proyecto tiene como objetivo desarrollar un **sistema bancario local** que permita a los usuarios gestionar sus cuentas, realizar transferencias, solicitar préstamos y otras operaciones básicas.

El sistema será diseñado para ejecutarse en el equipo del usuario sin necesidad de conexión a internet, lo que simplifica su implementación y reduce dependencias externas.

El propósito principal del proyecto es proporcionar una experiencia de desarrollo cercana a un entorno real, donde se apliquen buenas prácticas de ingeniería de software, metodologías ágiles (Scrum) y gestión de proyectos con un cliente simulado.

**Alcance del Proyecto**

El sistema incluirá las siguientes funcionalidades principales:

* **Registro e inicio de sesión** con autenticación local.
* **Gestión de cuentas bancarias**, permitiendo apertura y consulta de cuentas de ahorro y corriente.
* **Transferencias entre cuentas**, tanto propias como de terceros.
* **Gestión de tarjetas de débito y crédito**, incluyendo solicitud y bloqueo.
* **Pagos programados** y automatización de pagos recurrentes.
* **Préstamos y créditos**, con simulación de pagos y proceso de aprobación.
* **Inversiones y ahorros**, permitiendo la simulación y apertura de cuentas de inversión.
* **Seguridad básica**, con detección de transacciones sospechosas y configuración de alertas.
* **Reportes financieros**, proporcionando resúmenes de ingresos y gastos.
* **Servicio al cliente**, a través de un módulo de soporte con preguntas frecuentes.

Para evitar complejidad innecesaria, el sistema **no** incluirá:

* Integración con bancos externos o pasarelas de pago.
* Funcionalidad en la nube o acceso remoto.
* Sucursales físicas o soporte para múltiples administradores.
* Soporte para criptomonedas o inversiones avanzadas.

**Ejecución y Plataforma**

El sistema será una aplicación de escritorio que deberá instalarse o ejecutarse localmente en el equipo del usuario. Se explorará la posibilidad de utilizar **mensajes a Telegram** para la verificación de identidad si su implementación resulta sencilla.

**Enfoque Metodológico**

Se utilizará **Scrum** como metodología de desarrollo, con un enfoque en iteraciones cortas, planificación de sprints y entregables parciales para facilitar el seguimiento del proyecto y asegurar el cumplimiento de los objetivos establecidos.

# Administración del Proyecto

Propósito: Explicar los aspectos más importantes del funcionamiento interno del proyecto, así como su comunicación con entidades externas.

Contenido: ver cada subsección

## Modelo de Ciclo de Vida

Este proyecto sigue el **modelo ágil Scrum**, con un enfoque en la entrega continua de valor a través de iteraciones llamadas **sprints**.

El proyecto se organiza en **sprints de 1 o 2 semanas**, en los cuales se definen las tareas a realizar y se establece un objetivo claro para cada ciclo. A lo largo de cada sprint, el equipo trabajará en las funcionalidades del sistema bancario, con revisiones periódicas al final de cada sprint para evaluar el progreso.

**Prácticas Específicas Utilizadas en el Proyecto**

* **Gestión de Tareas con Trello**: Todas las tareas serán registradas y gestionadas en Trello, donde se asignarán a los programadores y se establecerán fechas límite. Trello permitirá un seguimiento claro del progreso de cada tarea y su estado (por hacer, en progreso, o completada).
* **Colaboración mediante GitHub**: El desarrollo se realizará de manera colaborativa utilizando GitHub. Los miembros del equipo compartirán y revisarán el código de forma conjunta, asegurando que todas las funcionalidades sean desarrolladas de manera coherente y bajo un control de versiones adecuado.
* **Pruebas Manuales**: Aunque no se implementarán pruebas automatizadas, se realizarán pruebas manuales al final de cada sprint para verificar que las funcionalidades implementadas cumplan con los requisitos y no afecten el funcionamiento de otras partes del sistema.
* **Comunicación**: La comunicación se gestionará principalmente a través de reuniones en clase y mensajes por WhatsApp para resolver dudas o coordinar tareas fuera del horario de las reuniones formales.

**Seguimiento del Progreso**

El avance del proyecto será monitorizado utilizando herramientas como Trello para ver el estado de las tareas y el cumplimiento de los plazos. Además, se realizarán reuniones diarias de **standup** (breves reuniones) para discutir el progreso y cualquier bloqueo que el equipo pueda estar enfrentando.

## Lenguajes y Herramientas

Propósito: Que el lector sepa qué **lenguajes de modelado y programación**, así como las **herramientas** que se utilizarán en el proyecto (desde ofimática hasta herramientas para pruebas automáticas y de control y administración de versiones).

Contenido: Enumerar y describir brevemente cada herramienta y lenguaje. Importante clasificar las herramientas y tener presente su uso en los demás planes presentados.

Tamaño sugerido: 1 página

Para la realización del proyecto se incentiva el uso de lenguajes y herramientas que puedan proveer de un modelado integral del proyecto, que sirvan para la ejecución del mismo, que ayuden al diseño del proyecto y finalmente que documenten cada detalle de las operaciones que se harán de manera interna. Es por esto que a continuación se identificarán cuales son los principales aspectos y herramientas que se usarán para el proyecto.

La herramienta principal, el núcleo de la ejecución del proyecto es la programación en un lenguaje en concreto. El lenguaje de programación establecido para el desarrollo del proyecto es Java. Esto se debe a dos motivos principales; es un lenguaje de programación extremadamente robusto y confiable, y todos los integrantes que participan en el proyecto cuentan con conocimientos de este. Adicionalmente, Java permite la programación orientada a objetos, paradigma clave para el desarrollo, y la integración de herramientas que se verán mas adelante.

Posteriormente se utilizará el IDE Intellij Idea. Con la ventaja de tener un correo universitario, Jet Brains, empresa dueña de Intellij Idea, provee de una licencia profesional para tener acceso a todas las características dentro del IDE. De esta manera se tiene una solución profesional para comenzar con el desarrollo del proyecto, así como un IDE confiable para la ejecución del mismo, creando un entorno de desarrollo fácil y seguro de utilizar.

Java FX es una herramienta que permite crear y programar interfaces gráficas basadas en el lenguaje de programación JAVA. Su facilidad de uso es clave para el desarrollo del proyecto, por lo que este será la herramienta asignada para la creación de interfaces gráficas. Java FX a su vez, se puede integrar con el IDE anteriormente mencionado para lograr una construcción integral de desarrollo asociado a su lenguaje y los elementos gráficos.

El control de versiones distribuido va a ser clave para el desarrollo completo del proyecto. Tener una trazabilidad de todos los cambios que se van cometiendo a través de una línea de tiempo ayuda a evitar problemas y conflictos en el desarrollo de la solución. Es por esto que se va a utilizar Git como herramienta para control de versiones del proyecto, así como GitHub para un control de versiones remoto en donde todos los integrantes puedan extraer y colocar cambios a medida que el proyecto vaya progresando.

Para la documentación, la suite de office que provee la universidad es suficiente. Herramientas como Microsoft Word, o Word Online serán utilizadas para la redacción y publicación de los documentos de alta importancia que vayan surgiendo durante el desarrollo del proyecto.

## Interfaces Externas

Propósito: Que un integrante del proyecto sepa quiénes son las principales entidades involucradas externas al equipo de desarrollo, su relación con el proyecto y cómo comunicarse con ellas.

Contenido:

Descripción de entidades: Una tabla que indique el nombre de la entidad, una descripción y sus responsabilidades asociadas al proyecto y datos de contacto o medios de comunicación.

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre Entidad | Descripción | Responsabilidades asociadas al proyecto | Datos de contacto/ Medios de comunicación |
| Jaime Andrés Pavlich | Profesor de la materia | Supervisa y califica el proyecto | Horas de clase comunes/ Correo Electrónico: jpavlich@javeriana.edu.co |

## Organigrama y Descripción de Roles

Diagrama

El contenido generado por IA puede ser incorrecto.

|  |  |  |
| --- | --- | --- |
| Rol | Descripción | Responsabilidades |
| Cliente (Profesor) | Actúa como el cliente del proyecto, proporcionando requisitos y validando entregables. | * Evaluar avances y entregables. * Proporcionar retroalimentación al equipo. |
| Scrum Master (Facilitador) | Coordina la aplicación de la metodología Scrum, asegurando la correcta gestión del proyecto. | * Gestionar sprints y facilitar reuniones. * Apoyar la organización del equipo. * Resolver bloqueos y coordinar con el cliente. * Supervisar el uso de Trello y GitHub. |
| Programadores | Encargados del desarrollo, pruebas y documentación del sistema bancario. | * Implementar funcionalidades del sistema. * Realizar pruebas manuales de las características desarrolladas. * Gestionar código en GitHub y tareas en Trello. * Asistir a reuniones de equipo y sprints. |
| Consultores Externos (Profesores) | Brindan orientación técnica y resuelven dudas específicas. | * Responder preguntas técnicas. * Orientar sobre metodologías o herramientas. |

## Calendarización y Entregables

|  |  |  |
| --- | --- | --- |
| Fechas | Descripción | Responsabilidades |
| Cliente (Profesor) | Actúa como el cliente del proyecto, proporcionando requisitos y validando entregables. | * Evaluar avances y entregables. * Proporcionar retroalimentación al equipo. |
| Scrum Master (Facilitador) | Coordina la aplicación de la metodología Scrum, asegurando la correcta gestión del proyecto. | * Gestionar sprints y facilitar reuniones. * Apoyar la organización del equipo. * Resolver bloqueos y coordinar con el cliente. * Supervisar el uso de Trello y GitHub. |
| Programadores | Encargados del desarrollo, pruebas y documentación del sistema bancario. | * Implementar funcionalidades del sistema. * Realizar pruebas manuales de las características desarrolladas. * Gestionar código en GitHub y tareas en Trello. * Asistir a reuniones de equipo y sprints. |
| Consultores Externos (Profesores) | Brindan orientación técnica y resuelven dudas específicas. | * Responder preguntas técnicas. * Orientar sobre metodologías o herramientas. |

Propósito: Describir las principales tareas, sus fechas de inicio y término, así como los principales hitos y entregables asociados.

Si el método a usar no es ágil, se puede utilizar una carta Gantt detallada. De lo contrario, con una tabla puede ser suficiente.

Tamaño sugerido: 1 página

# Procesos de Soporte

Propósito: Que los integrantes del proceso sepan cuáles los procesos transversales al proyecto que permiten soportar su ejecución y cuáles son sus responsabilidades en dichos procesos.

Contenido: ver sub-secciones

## Ambiente de Trabajo

**Reglas del Trabajo en Equipo**

* **Asistencia a reuniones**: Todos los integrantes deben asistir a las reuniones programadas, salvo en caso de fuerza mayor.
* **Comunicación**: La comunicación principal del equipo se realizará a través de **WhatsApp**, tanto para coordinación como para resolver dudas rápidas.
* **Resolución de conflictos**: Si surge un desacuerdo, se intentará resolver dentro del equipo. Si no se llega a un acuerdo, se consultará con el profesor para una decisión final.

**Mecanismos para Asegurar el Cumplimiento**

* **Seguimiento semanal**: Se revisará el avance de cada integrante semanalmente para asegurar que todas las tareas asignadas estén en progreso o completadas.
* **Monitoreo en Trello**: Se utilizará **Trello** para asignar y seguir el estado de las tareas.
* **Registro de código en GitHub**: El desarrollo colaborativo se gestionará en **GitHub**, donde se validará el trabajo realizado.
* **Manejo de incumplimientos**: No habrá penalizaciones iniciales, pero si un integrante incumple repetidamente sus responsabilidades sin justificación, podría ser excluido del equipo.

**Responsables y Artefactos de Control**

* **Responsable del cumplimiento**: El **Scrum Master** (santiago) se encargará de verificar que las reglas se cumplan y consultará con el profesor en caso de problemas recurrentes.
* **Herramientas de control**: Se utilizarán **Trello y GitHub** como evidencia del trabajo realizado y para documentar avances.

## Monitoreo y Control de Progreso

**MÉTRICA** utilizada para medir progreso: Para propósitos de medir el progreso del desarrollo del proyecto, se va a usar una métrica basada en las horas faltantes para completar una tarea de acuerdo con las fechas acordadas por el grupo, así se podrá medir cuanto progreso se ha dado en determinados intervalos de tiempo y ayudar a formular posibles acciones correctivas.

Actividades a realizar para reportar progreso: El responsable para los reportes de progreso es Damian Rey, estos reportes se realizarán aproximadamente 24 horas antes de las fechas establecidas, estos consistirán en un reporte en las mismas palabras de los integrantes del grupo para testificar sus progresos, asimismo se usará la tabla de historial de cambios para corraborar y manetener un registro de los cambios y modificaciones que se hayan hecho.

Acciones correctivas: En caso de que el progreso del proyecto no vaya según lo esperado, ya sea que haya un decremento de progreso en comparación con el intervalo pasado, o se presente una dificultad notable que impida el progreso de varios miembros del grupo, se realizara una reunion entre todos los miembros del grupo en el que se discutirán las causas de las dificultades en el progreso del proyecto y se decidirá entre todos que acciones en concreto se deberían tomar, estas reuniones sucederían en conjunto con los reportes de progreso cotidiano para aclarar todas las posibles dudas y prepararse para la siguiente fecha.

## Administración de Configuración y Documentación

Propósito: identificar los principales ítems de configuración y cómo evolucionarán durante el tiempo, particularmente los artefactos de documentación y el código.

Contenido:

Enumerar y describir brevemente los ítems de configuración identificados

Qué tipos de ramas se utilizarán en el repositorio GIT del código fuente, cuál será su propósito y cuáles serán los procesos para crear nuevas ramas o combinarlas (merge).

Tabla con el “Caso de Desarrollo”

Tamaño sugerido: 1 a 2 páginas

## Control de Calidad

Propósito: Que los integrantes conozcan las actividades relacionadas a todos los procesos de control de calidad del proyecto.

Contenido

Explicación de alto nivel de todos los procesos asociados a control de calidad, lo cual incluye verificación, validación, revisiones y auditorías al proyecto. Por ejemplo: revisiones de calidad de documentos, pruebas de software, etc. Para cada proceso indicar: nombre del proceso, momentos del proyecto en que ocurren y quiénes son los responsables. Se recomienda resumir esta información en una tabla.

Tamaño sugerido: 1 a 2 páginas

# Referencias

Indique aquí todas las referencias bibliográficas utilizadas en el documento. Utilice formato IEEE o APA para definirlas. Para administrar automáticamente las referencias, se recomienda el uso de la herramienta Zotero (www.zotero.org).

# Anexos

Agregue aquí cualquier información adicional relevante para el proyecto que no quepa en las secciones previas.

Es importante tener en cuenta que, hasta antes de esta sección, el documento debe ser **autocontenido**, es decir, el lector no debe necesitar leer ningún documento adicional o archivo externo para entender el proyecto. La información que se coloque aquí es simplemente un complemento, en caso de que el lector requiriera más detalles sobre algún tema.

**Cualquier figura que sea referenciada en el documento debe ir en la sección respectiva. No deben colocarse como anexos.**