Informe N°4: Desarrollo Final del Sistema de Préstamo de Libros

**Asignatura:** Taller de Desarrollo de Aplicaciones.

**Nombre del docente:** Nilson Arturo

**Nombre de los integrantes del grupo:** Sebastian Ovando, Axel Villa, Juan Herrera

**Fecha de entrega: 20/04/2025**

1. Contenidos

* Introducción
* Toma de Requerimientos:
  + Contexto y Problemas Detectados
  + Requerimientos Funcionales y No Funcionales
* Casos de Uso
* Historias de Usuario
* Orden de Servicio (Anexo)
* Desarrollo y Avances Realizados
* Diagramas de la Base de Datos
* Desarrollo de aplicación
* Pruebas Realizadas
* Conclusiones y Próximos Pasos

1. Introducción

El presente informe detalla el proceso seguido para el desarrollo del Sistema de Préstamo de Libros. La finalidad del proyecto es transformar el proceso actual, basado en plantillas de Excel, en una solución automatizada, intuitiva y escalable que permita a la biblioteca gestionar eficientemente el préstamo, devolución y seguimiento de libros. Este sistema se desarrollará utilizando Python, Tkinter para la interfaz gráfica y SQLite para la gestión de datos, garantizando un acceso seguro y una experiencia de usuario optimizada.

La entrega final incluirá una aplicación de escritorio que abarque desde el registro y autenticación de usuarios, hasta la administración de libros y el control de préstamos y renovaciones, complementada con reportes y análisis de datos.

1. Toma de Requerimientos

**Contexto y Problemas Destacados**:

* El proceso actual se realiza mediante planillas Excel, lo que dificulta la búsqueda y gestión de libros.
* Se detectó que es complicado identificar si un libro está en préstamo, disponible o se encuentra en la librería.
* Los usuarios (estudiantes y docentes) enfrentan restricciones en cuanto a los tiempos de préstamo y en la posibilidad de renovar sus préstamos.

**Requerimientos Funcionales y No Funcionales**:

* **Autenticación**: Sistema de login y registro que garantice el acceso exclusivo a los trabajadores de la biblioteca.
* **Gestión de Libros**: Capacidad para agregar, modificar, eliminar y consultar libros, incluyendo información detallada (título, autor, editorial, código, descripción y estado).
* **Préstamos y Devoluciones**: Registro de préstamos con fecha de inicio y devolución, reglas diferenciadas para estudiantes y docentes, y posibilidad de renovación bajo ciertas condiciones.
* **Control de Multas**: Cálculo y registro de multas por retrasos, con sistema de abonos y notificaciones.
* **Reportes y Análisis**: Generación de informes diarios y gráficos que permitan visualizar el estado de los préstamos, devoluciones y situación de las multas.

1. Entrevista de Levantamiento de Requerimientos.

Para obtener una visión integral de las necesidades y expectativas del sistema, se realizó una entrevista de levantamiento de requerimientos. En esta entrevista se plantearon preguntas dirigidas a identificar tanto los problemas actuales como las oportunidades de mejora en el proceso de gestión de préstamos de libros. Las preguntas han sido diseñadas para cubrir aspectos generales, específicos del proceso de préstamo, usuarios, catálogo, reportes, seguridad y otras áreas críticas.

A continuación, se presenta la lista completa de preguntas formuladas:

* ¿Cuál es el objetivo principal del sistema de préstamo de libros?
* ¿Existen sistemas previos o se gestiona todo de forma manual?
* ¿Cuáles son los principales problemas que enfrenta la biblioteca actualmente?
* ¿Cómo se realiza actualmente el préstamo de libros a docentes y estudiantes?
* ¿Cuáles son los datos esenciales que deben registrarse en cada préstamo?
* ¿Hay diferentes reglas de préstamo según el tipo de usuario? (ejemplo: límite de días o cantidad de libros)
* ¿Existe algún mecanismo actual para la renovación de préstamos?
* ¿Cómo se manejan los retrasos en la devolución de libros y las multas?
* ¿Qué tipos de usuarios interactuarán con el sistema? (ejemplo: bibliotecarios, estudiantes, docentes)
* ¿Cuáles serán las funciones específicas de cada tipo de usuario dentro del sistema?
* ¿Habrá restricción de acceso a ciertas funcionalidades?
* ¿Cómo se registrarán los libros en el sistema? (manual o importación de datos)
* ¿Qué información debe contener la ficha de cada libro?
* ¿Se necesita una función para actualizar la disponibilidad y el estado de los libros?
* ¿Qué tipo de informes o reportes deben generarse en el sistema?
* ¿Quién tendrá acceso a estos reportes?
* ¿Deben incluirse filtros específicos para generar reportes más detallados?
* ¿Cómo se manejará la seguridad del sistema en cuanto al acceso y modificación de datos?
* ¿Se necesitará algún tipo de autenticación especial para ciertos usuarios?
* ¿Existe alguna otra funcionalidad que considere relevante para mejorar la eficiencia del sistema?
* ¿Hay procesos manuales que podrían automatizarse con el sistema?
* ¿Qué expectativas tiene sobre la interfaz y usabilidad del sistema?

1. Resultados de la entrevista.

Requerimientos Ordenados por temas

**1. Problema actual y necesidad del sistema**

* Necesitamos facilitar el tema de gestión de libros, ya que actualmente se utiliza una plantilla Excel y no es agradable utilizarlo. Los usuarios piden que se automatice.
* Se ejecuta todo en una plantilla Excel y no está automatizado.
* El principal problema es encontrar los libros: si lo tiene un usuario, si está ocupado o si está en la librería.

**2. Proceso de préstamo y devolución**

* Actualmente, se tiene una planilla por día donde se registra:
  + Nombre del solicitante
  + Su rol
  + Su certificado de estudiante
  + La fecha en que recibe y entrega el libro
  + El código del libro
* Datos esenciales que se deben registrar:
  + Fecha de préstamo
  + Tipo de usuario
  + Fecha de devolución
  + Código del libro
* Los alumnos tienen un máximo de 7 días para el préstamo.
* Los profesores tienen un mínimo de 7 días y un máximo de 20 días.

**3. Información de los libros y catálogo**

* Se necesita que los libros tengan más información aparte de su código, como:
  + - Editorial
    - Código
    - Título
    - Descripción del contenido
* Se necesita un estado que indique si el libro está en préstamo o disponible.
* Funcionalidades necesarias en la gestión de libros:
  + Solicitar préstamo
  + Registrar quién lo devuelve o lo pide prestado
  + Agregar y eliminar libros
  + Modificar cantidades de libros
  + Editar autor o tema del libro
  + Tener un listado de todos los libros con búsqueda o filtrado

**4. Reportes y consultas**

* Se necesita consultar sobre:
  + Libros en préstamo
  + Libros que están por vencer
  + Personas con multas de devolución
  + Resumen diario de entrada y salida de libros, idealmente en formato gráfico.
* Se podrán buscar libros por distintos parámetros comunes, como:
  + Etiquetas (tags)
  + Libros más solicitados
* Se debe tener acceso fácil a los solicitantes, a los libros y al listado de deudas.

**5. Usuarios del sistema y seguridad**

* Solo existirá un perfil de administrador del sistema, pero varios usuarios podrán tener ese perfil.
* Seguridad del sistema:
  + El usuario debe iniciar sesión con usuario y contraseña.
  + Debe quedar registro de qué usuario hace cada cambio.
* Solo los trabajadores de la biblioteca tendrán acceso al sistema.
* Información adicional de los solicitantes:
  + Nombre
  + RUT
  + Cantidad de libros que poseen
  + Fecha de devolución
  + Correo (para enviar notificaciones o recordatorios)
* No (falta contexto para saber a qué pregunta responde el "No").

**6. Registro y actualización de datos**

* Actualmente, los libros deben ingresarse uno por uno.
* Gestión de multas y pagos:
  + Actualmente, se tacha a los que pagan la deuda y se entregan boletas.
  + Se necesita un sistema de abono.
  + Si el solicitante paga todo, quedará como pagado (Booleano: Sí/No).
  + Se recomienda marcar con colores o íconos a los usuarios en deuda.

1. Casos de Uso

* **Sistema Biblioteca**

**Diagrama

El contenido generado por IA puede ser incorrecto.**

* **Registro de Préstamo de Libros**

Diagrama

El contenido generado por IA puede ser incorrecto.

* **Devolución de libros**

Diagrama

El contenido generado por IA puede ser incorrecto.

* **Renovación de préstamo**

**Diagrama

El contenido generado por IA puede ser incorrecto.**

* **Administración de multas**

**Diagrama

El contenido generado por IA puede ser incorrecto.**

* **Registrar nuevos libros**

Diagrama

El contenido generado por IA puede ser incorrecto.

* **Edición o eliminación de libros**

**Diagrama

El contenido generado por IA puede ser incorrecto.**

* **Consulta de Disponibilidad de Libros**

Diagrama

El contenido generado por IA puede ser incorrecto.

* **Generación de Reportes**

Diagrama

El contenido generado por IA puede ser incorrecto.

* **Gestión de Usuarios**

**Diagrama

El contenido generado por IA puede ser incorrecto.**

* **Autenticación en el Sistema**

**Diagrama

El contenido generado por IA puede ser incorrecto.**

1. Historias de Usuario

En esta sección se exponen las Historias de Usuario diseñadas para capturar los requerimientos funcionales del sistema de préstamo de libros. Es importante aclarar que, según lo solicitado por el cliente, el único usuario que tiene acceso directo al sistema es el bibliotecario. Tanto los estudiantes como los profesores realizan sus solicitudes a través del bibliotecario. Para reflejar este flujo, se han adaptado las historias de usuario utilizando la denominación "Bibliotecario solicitud de estudiantes" o "Bibliotecario solicitud de profesores" en la columna de rol, lo que permite mantener el mismo conjunto de historias diferenciando las condiciones específicas (por ejemplo, el tiempo límite de devolución).

A continuación, se muestra la tabla con las historias de usuario:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Rol** | **Característica / Funcionalidad** | **Razón / Resultado** | **Condiciones** | **Prioridad** |
| 01 | Bibliotecario solicitud de estudiantes | Necesito registrar la solicitud de préstamo de libros para estudiantes. | Con la finalidad de permitir que un estudiante disponga del material de estudio a domicilio para realizar tareas y estudios. | - El bibliotecario debe registrar la solicitud tras validar que el estudiante no tenga multas impagas. - Se puede solicitar hasta 4 libros. - Registrar fecha de préstamo y fecha de devolución esperada. | Alto |
| 02 | Bibliotecario solicitud de profesores | Necesito registrar la solicitud de préstamo de libros para profesores. | Con la finalidad de permitir que un profesor obtenga el material necesario para preparar sus clases o investigar, con condiciones de préstamo diferenciadas. | - El bibliotecario debe registrar la solicitud verificando que el profesor cumpla con el plazo de 7 a 20 días. - Solo se puede solicitar 1 ejemplar por título. - Registrar fechas de préstamo y devolución. | Alto |
| 03 | Bibliotecario solicitud de estudiantes / profesores | Quiero gestionar la renovación del préstamo de un libro. | Con la finalidad de extender el tiempo de uso del libro cuando el estudiante o profesor lo requiera. | - El sistema debe validar la disponibilidad del libro para renovación (sin solicitudes pendientes). - Estudiantes: 1 renovación de 3 días adicionales. - Profesores: hasta 3 renovaciones consecutivas. | Medio-Alto |
| 04 | Bibliotecario | Necesito registrar la devolución de libros. | Con la finalidad de liberar el ejemplar y ponerlo disponible para otros usuarios, actualizando el historial del préstamo. | - El bibliotecario debe buscar el libro por código o título. - Actualizar el estado a "Disponible". - Calcular si hay retrasos y aplicar multas si corresponde. | Alto |
| 05 | Bibliotecario | Necesito gestionar las multas generadas por retrasos en la devolución. | Con la finalidad de controlar las deudas y evitar nuevos préstamos si existen multas impagas. | - Registrar pagos parciales (abonos). - Marcar como "Sin Deuda" cuando se complete el pago total. - Resaltar usuarios con deuda mediante colores o íconos. | Alto |
| 06 | Bibliotecario | Necesito agregar libros al sistema (ingresando título, autor, editorial, código y descripción). | Con la finalidad de mantener actualizado el catálogo de la biblioteca, permitiendo que los usuarios realicen sus solicitudes de préstamo. | - Ingresar datos obligatorios: título, autor, código. - Estado inicial: "Disponible". - Permitir manejar múltiples ejemplares del mismo libro. | Medio-Alto |
| 07 | Bibliotecario | Quiero modificar y eliminar libros del sistema. | Con la finalidad de corregir datos erróneos o dar de baja libros dañados, manteniendo la información actualizada sin afectar el historial de préstamos. | - No se permite eliminar un libro que esté actualmente en préstamo. - Conservar el historial de préstamos incluso de los libros eliminados. | Medio |
| 08 | Bibliotecario | Necesito consultar la lista de libros en préstamo y próximos a vencer. | Con la finalidad de contactar a los usuarios y gestionar oportunamente las devoluciones para evitar retrasos y multas. | - Mostrar fecha estimada de devolución y el usuario que tiene el libro. - Resaltar aquellos libros cuyo plazo de devolución esté próximo a vencer (por ejemplo, faltan 2 días). | Alto |
| 09 | Bibliotecario | Deseo generar un reporte diario de préstamos y devoluciones. | Con la finalidad de tener un control de la actividad y tomar decisiones, como adquirir más ejemplares de libros muy solicitados. | - Incluir datos de fecha, cantidad de libros prestados, devoluciones y multas cobradas. - Mostrar gráficos o exportar a PDF/CSV si es posible. | Medio-Alto |
| 10 | Bibliotecario | Quiero buscar libros mediante parámetros (título, autor, editorial, código, tags). | Con la finalidad de ubicar rápidamente un libro o ejemplar específico para satisfacer las solicitudes de los usuarios. | - Mostrar coincidencias y el estado actual (Disponible / En préstamo / etc.). - Permitir filtrar por varios criterios simultáneamente (ej. título y autor). | Alto |
| 11 | Bibliotecario | Necesito un sistema de registro de solicitudes (login y registro) para los trabajadores. | Con la finalidad de proteger el acceso al sistema y registrar quién realiza cada acción (por ejemplo, quién registra una solicitud de préstamo o devolución). | - El sistema debe almacenar usuario y contraseña (idealmente encriptados). - Registrar cada acción para efectos de auditoría. | Alto |
| 12 | Bibliotecario solicitud de estudiantes / profesores | Quiero recibir notificaciones sobre la fecha de entrega de los libros. | Con la finalidad de evitar multas y asegurar la devolución oportuna de los libros solicitados. | - El sistema debe almacenar datos de contacto (correo, teléfono, etc.). - Enviar recordatorios cuando falte poco tiempo para la devolución. | Medio-Bajo |

1. Anexo: Orden de Servicio

**Datos del Cliente**

|  |  |
| --- | --- |
| **Nombre** | *Nilson Arturo Guerra Cabeza* |
| **Rut** | *15.285.399-1* |
| **Teléfono** | *+56 9 3887 5444* |
| **Email** | *Nilson.guerra@inacapmail.cl* |

**Datos de los Consultores técnicos**

|  |  |
| --- | --- |
| **Nombre** | Axel Villa Poblete |
| **Rut** | 21.705.120-7 |
| **Nombre** | Sebastián Ovando |
| **Rut** | 21.645.164-3 |

**Observaciones generales**

|  |
| --- |
| El proyecto consiste en desarrollar una aplicación de escritorio que modernice la gestión del préstamo de libros en una biblioteca. Su objetivo es automatizar el registro y control de préstamos, devoluciones, renovaciones y gestión de multas, centralizando la información y facilitando la labor del bibliotecario, quien actuara como único usuario del sistema. La aplicación está diseñada para funcionar de forma local, sin necesidad de conexión a internet, y se exportara a un archivo ejecutable (EXE) para su fácil implementación en el entorno de la biblioteca. |

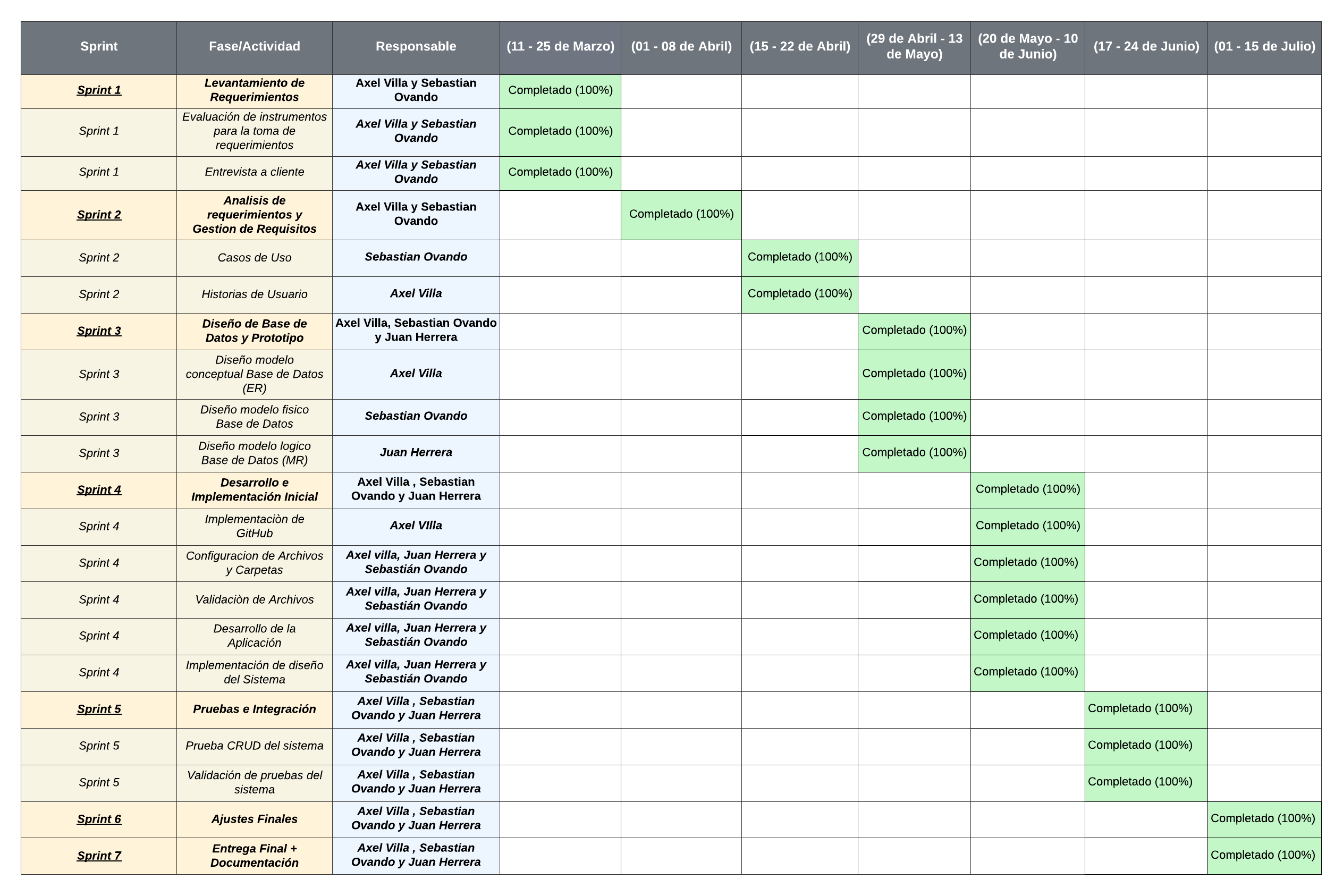
**Descripción del Servicio:**

|  |
| --- |
| **Para efectuar el desarrollo de la aplicación se requiere del siguiente entorno de trabajo:**  *1.- Infraestructura necesaria para la implementación de la aplicación*:  La implementación del sistema se realizará de forma local en la biblioteca, por lo que se requiere una infraestructura básica que incluya:  **Equipo de escritorio:** Uno o más computadores en los que se instalara y ejecutara el sistema  **Sin conexión a internet:** La aplicación está diseñada para funcionar completamente en un entorno sin conexión, basándose en SQLite para el almacenamiento de datos.  **-------------------------------------------------------------------------------------------------------------------------------**  *2.- Configuración necesaria del hardware para que funcione la aplicación*:  Para garantizar el correcto funcionamiento de la aplicación de escritorio se recomienda contar con las siguientes especificaciones mínimas:  Procesador: Intel i3 o equivalente a AMD  Memoria RAM: 8GB  Almacenamiento: 100GB disponibles para el sistema y la base de datos; se recomienda un SSD para mayor rapidez en lectura/escritura.  Pantalla: Resolución mínima de 1366x768  Conectividad: No se requiere conexión a Internet, ya que el sistema opera de manera local  **-------------------------------------------------------------------------------------------------------------------------------**  *3.- Software necesario para el desarrollo, y posterior implementación de la aplicación*:  **Para el desarrollo:**  Lenguaje de programación: Python 3 (Preferiblemente 3.8 o superior)  Framework de Interfaz Gráfica: Tkinter (Para la creación de la aplicación de escritorio)  Base de Datos: SQLite (Para desarrollo y pruebas iniciales)  Editor/Entorno de desarrollo: VSCode  Generación del Ejecutable: PyInstaller u otra herramienta similar para exportar la aplicación a un archivo EXE.  **Para la Implementación:**  Sistema Operativo: Windows 10 o superior  Base de Datos: SQLite (Integrada en la aplicación, sin necesitad de un servidor de bases de datos)  Entorno de ejecución: La aplicación se ejecutará de manera nativa en el equipo del usuario, sin requerir conectividad a internet) |

1. Carta Gantt del Proyecto: Sistema de Préstamos de Libros.

La presente Carta Gantt ofrece una representación visual del cronograma del proyecto "Sistema de Préstamo de Libros". En esta planificación, se detallan tanto las fases ya concluidas—como el levantamiento de requerimientos, análisis y la definición de casos de uso e historias de usuario—como las actividades pendientes que abarcan el diseño, desarrollo, pruebas e implementación final del sistema. Además, se agregó la sección de Sprint que consiste en la agrupación de tareas a realizar durante el proyecto, explicando así de mejor manera la planificación de este. El cronograma se extiende desde el inicio de clases el 11 de marzo de 2025 hasta la entrega final programada para el 15 de julio de 2025, abarcando 18 semanas de trabajo.

Esta herramienta de planificación permite asignar de manera clara las tareas a los integrantes del equipo (Axel Villa, Sebastián Ovando y Juan Herrera), facilitando el seguimiento del progreso y asegurando la coherencia entre las actividades y los objetivos del proyecto. Además, se han incluido indicadores visuales que distinguen las fases completadas de las pendientes, lo que mejora la visibilidad y el control sobre el avance global del proyecto.



1. Desarrollos y Avances Realizados

En esta fase se documentaron y organizaron los procesos fundamentales del proyecto a partir del levantamiento de requerimientos. Hasta el momento, los avances se han centrado en:

* **Análisis y Toma de Requerimientos:**

Se realizaron entrevistas, encuestas a estudiantes, reuniones y análisis del proceso actual (basado en plantillas Excel) para identificar los problemas y necesidades de la biblioteca. Estos instrumentos permitieron definir los requerimientos funcionales y no funcionales del sistema.

* **Definición de Casos de Uso e Historias de Usuario:**

Se diseñaron diagramas de casos de uso, siguiendo la nomenclatura estandarizada, para representar los procesos principales de la aplicación (registro de préstamos, devolución, renovación, gestión de multas, etc.) Además, se elaboraron historias de usuario que capturan las necesidades de los distintos actores, adaptándolas para reflejar que el único usuario del sistema es el bibliotecario, quien registra en nombre de estudiantes y profesores.

* **Propuesta de Diseño y Prototipado:**

Se definió la arquitectura conceptual del sistema, que se desarrollará con Python, Tkinter y SQLite, en una aplicación de escritorio exportable como archivo EXE.

* **Orden de Servicio (Anexo):**

Se elaboró un documento de “Orden de Servicio” que especifica los requerimientos de infraestructura, configuración de hardware y software necesarios para el desarrollo e implementación de la aplicación.

* **Diseño Carta Gantt:**

Se elaboró el diseño de la Carta Gantt basándonos en los avances registrados hasta la fecha; asimismo, se incorporaron las fases y tareas que se ejecutarán en el futuro, detallando de manera integral todas las etapas del proyecto.

* **Diseño de modelos de Base de Datos:**

Se creó el modelo de base de datos pensando en cómo organizar la información de forma clara y eficiente. Se definieron tablas principales como Usuarios, Libros, Préstamos y Multas, considerando los datos que necesita manejar el sistema. También se establecieron relaciones entre las tablas para que todo funcione de manera conectada y sin errores. Este diseño permite registrar bien los préstamos, controlar las devoluciones, aplicar multas y mantener un buen seguimiento del uso de los libros.

1. Diagramas de la Base de Datos

En esta sección se presentan los diagramas correspondientes al modelo de datos del sistema de préstamo de libros. Estos diagramas reflejan la estructura lógica de la base de datos, detallando las entidades principales, sus atributos y las relaciones entre ellas. La base de datos se ha diseñado en SQLite para asegurar una implementación liviana y eficiente, acorde al entorno local en el que se desplegará la aplicación.

* **Diseño Modelo Conceptual (MER):**

Diagrama, Esquemático

El contenido generado por IA puede ser incorrecto.

* **Diseño Modelo Lógico:**

**Diagrama

El contenido generado por IA puede ser incorrecto.**

* **Diseño Modelo Físico:**

**Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.**

1. Desarrollo de aplicación

En esta sección se presentan capturas reales del sistema funcionando. Se muestran las pantallas principales desarrolladas con Python y Tkinter, y la gestión de datos con SQLite. Las interfaces fueron diseñadas pensando en la simplicidad y eficiencia para el bibliotecario.

* Captura del inicio de sesión

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

* Captura del registro de usuario

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

* Captura de la pantalla principal del bibliotecario

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

* Captura de la pantalla de gestión de libros

Escala de tiempo

El contenido generado por IA puede ser incorrecto.

* Captura del registro de préstamo

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

* Captura del registro de devolución

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

* Captura de una multa generada

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

* Captura de la renovación del préstamo

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

1. Pruebas Realizadas

| **Módulo probado** | **Entrada simulada** | **Resultado esperado** | **Resultado obtenido** | **Estado** |
| --- | --- | --- | --- | --- |
| Registro de  usuario | Usuario nuevo,  contraseña | Usuario registrado | Correcto | ✅ |
| Iniciar sesión | Usuario válido | Acceso al sistema | Correcto | ✅ |
| Agregar libro | Datos válidos | Libro aparece en la tabla | Correcto | ✅ |
| Registrar préstamo | Solicitante + libro | Se guarda y cambia  disponibilidad | Correcto | ✅ |
| Registrar devolución | Devolver con atraso | Multa generada | Correcto | ✅ |
| Renovación | Préstamo activo | Se extiende fecha de  devolución | Correcto | ✅ |

1. Próximos pasos
2. **Implementación de la Aplicación:**

* Desarrollar los módulos definidos en el análisis (autenticación, gestión de libros, registro de préstamos, renovaciones, devoluciones y gestión de multas) utilizando Python, Tkinter y SQLite.
* Elaborar el prototipo funcional y generar un archivo ejecutable (EXE) para facilitar la implementación en el entorno de la biblioteca.

1. **Realización de Pruebas:**

* Ejecutar pruebas de integración y usabilidad para validar que cada módulo cumpla con los requerimientos establecidos.
* Corregir posibles incidencias y optimizar el flujo de navegación.

1. **Validación con el Cliente:**

* Presentar el prototipo funcional al cliente (docente) y, a través del bibliotecario, obtener retroalimentación directa para realizar los ajustes necesarios.

1. **Documentación y Capacitación:**

* Completar la documentación técnica y de usuario para facilitar el mantenimiento y la futura escalabilidad del sistema.
* Planificar y ejecutar sesiones de capacitación para el personal encargado de la biblioteca.

1. **Inicio del desarrollo de la aplicación:**

* Comenzar con la programación del sistema y construir los módulos definidos previamente
* Desarrollo de la aplicación en Python utilizando tkinter para la interfaz gráfica y sqlite3 como base de datos

1. **Fin del desarrollo de la aplicación:**

* El sistema se encuentra totalmente desarrollado, probado e implementado. El siguiente paso será entregar la documentación final al cliente, incluyendo el manual de usuario, la ficha técnica, y presentar el sistema en la exposición final del proyecto.

1. Conclusiones

**Conclusiones**:

El proceso integral de análisis y planificación llevado a cabo ha permitido la meticulosa identificación, documentación y organización de los requerimientos para el Sistema de Préstamo de Libros. Los procesos operativos fundamentales, que incluyen autenticación, gestión de volúmenes, registro de préstamos y devoluciones, control de multas y generación de informes, fueron definidos con precisión a través de un enfoque multifacético que abarcó entrevistas, encuestas, reuniones con stakeholders (Docente) y un riguroso análisis documental. El desarrollo estratégico de casos de uso detallados e historias de usuario, junto con la Orden de Servicio adjunta, establece una base robusta y clara para el subsiguiente desarrollo de la aplicación. Esta fase representa un esfuerzo colectivo significativo y una valiosa experiencia de aprendizaje para el equipo. Somos optimistas de que la aplicación resultante satisfará eficazmente las necesidades de los usuarios y anticipamos oportunidades para su mejora continua y expansión.