

# **Enunciado del Ejercicio: Sistema de Gestión de Préstamos de Libros para una Biblioteca Pequeña (SGPL)**

Una pequeña biblioteca comunitaria desea automatizar la gestión de sus préstamos de libros. Actualmente, todo se realiza de forma manual, lo que genera errores y pérdida de tiempo. Necesitan un sistema sencillo que les permita registrar sus libros, gestionar a los usuarios (miembros de la biblioteca) y controlar los préstamos y devoluciones. El objetivo inicial es tener un prototipo funcional que cubra las funcionalidades básicas de préstamo y devolución de libros en un plazo de un mes.

## **Requisitos Generales Iniciales (EPICAS):**

- **Gestión de Libros:** La biblioteca necesita registrar los detalles de cada libro (título, autor, ISBN, editorial, año de publicación, cantidad disponible).
- **Gestión de Usuarios:** Los usuarios deben poder registrarse como miembros de la biblioteca con sus datos básicos (nombre, apellido, cédula de identidad, dirección, teléfono, correo electrónico).
- **Préstamo de Libros:** Los bibliotecarios deben poder registrar cuándo un usuario toma prestado un libro, con la fecha de préstamo y la fecha de devolución esperada. Se debe verificar la disponibilidad del libro.
- **Devolución de Libros:** Los bibliotecarios deben poder registrar la devolución de un libro, actualizando su disponibilidad.
- **Notificaciones (Simplificado):** El sistema debería indicar si un libro está vencido al momento de la devolución.

## **Otros aspectos que se deben considerar del sistema:**

- La interfaz debe ser intuitiva para los bibliotecarios, que no son expertos en tecnología.
- Las operaciones de préstamo y devolución deben ser casi instantáneas.
- La información de los usuarios debe ser tratada con confidencialidad.

---

## **Solución del Ejercicio: Ejemplo de Llenado de la Plantilla para el SGPL (Versión Given-When-Then y Planificación Semanal)**

A continuación, se muestra cómo se llenaría la plantilla propuesta para el Sistema de Gestión de Préstamos de Libros (SGPL), enfocándose en un subconjunto de Historias de Usuario con criterios Given-When-Then y una planificación semanal detallada del Sprint.

---

# **Plantilla de Documento de Entrega 3 - XP: Historias de Usuario y Planificación del Sprint 1**

---

## **1. Portada**

Universidad Central de Venezuela

Facultad de Ciencias

Escuela de Computación

Semestre 1-2025

Asignatura: Ingeniería de Software

Equipo: [Nombres integrantes "Equipo de Desarrollo"]

Fecha de Entrega: [...]

---

## **2. Requisitos del Sistema (Plan de lanzamiento Inicial)**

Esta sección detalla los requisitos funcionales y no funcionales identificados para el Sistema de Gestión de Préstamos de Libros (SGPL). Se utilizarán **historias de usuario** para describir la funcionalidad desde la perspectiva del usuario, y cada historia incluirá sus **criterios de aceptación** para definir cuándo se considera completa, utilizando el patrón Given-When-Then.

### **2.1. Historias de Usuario**

Se recomienda organizar las historias de usuario por módulos o funcionalidades principales, utilizando el siguiente formato para cada una:

---

**Historia de Usuario:** HU-001 - Registrar Nuevo Libro en el Catálogo

Como: Bibliotecario

Quiero: Poder añadir los datos de un nuevo libro al catálogo de la biblioteca

Para: Mantener un registro actualizado de los libros disponibles para préstamo.

Prioridad: Alta

Estimación (Puntos de Historia): 3

#### Criterios de Aceptación:

- **Escenario: Registro exitoso de un nuevo libro.**
  - **Dado que** estoy en la pantalla de registro de libros,
  - **Cuando** ingreso un Título, Autor, ISBN (único), Editorial, Año de Publicación y Cantidad Disponible válidos,
  - **Entonces** el sistema guarda el libro en el catálogo y muestra un mensaje de confirmación "Libro registrado exitosamente".
- **Escenario: Intentar registrar un libro con ISBN duplicado.**
  - **Dado que** un libro con el ISBN "978-0123456789" ya existe en el catálogo,
  - **Cuando** intento registrar un nuevo libro usando el mismo ISBN "978-0123456789",
  - **Entonces** el sistema impide el registro y muestra un mensaje de error "El ISBN ingresado ya existe".
- **Escenario: Intentar registrar un libro con campos obligatorios vacíos.**
  - **Dado que** estoy en la pantalla de registro de libros,
  - **Cuando** intento guardar un libro dejando el campo Título o ISBN vacío,
  - **Entonces** el sistema no guarda el libro y muestra un mensaje de error indicando que los campos son obligatorios.

#### Notas Adicionales (Opcional):

- Ninguna.

---

#### Historia de Usuario: HU-002 - Registrar Nuevo Usuario

Como: Bibliotecario

Quiero: Poder registrar a una nueva persona como miembro de la biblioteca

Para: Permitirles tomar libros prestados y mantener un registro de los usuarios.

Prioridad: Alta

Estimación (Puntos de Historia): 5

#### Criterios de Aceptación:

- **Escenario: Registro exitoso de un nuevo usuario.**
  - **Dado que** estoy en la pantalla de registro de usuarios,
  - **Cuando** ingreso un Nombre, Apellido, Cédula de Identidad (única), Dirección, Teléfono y Correo Electrónico válidos,
  - **Entonces** el sistema guarda el usuario, asigna un ID de miembro único y muestra un mensaje de confirmación "Usuario registrado exitosamente con ID: [ID\_Generado]".

- **Escenario: Intentar registrar un usuario con Cédula de Identidad duplicada.**
  - **Dado que** un usuario con la Cédula de Identidad "V-12345678" ya existe en el sistema,
  - **Cuando** intento registrar un nuevo usuario usando la misma Cédula de Identidad "V-12345678",
  - **Entonces** el sistema impide el registro y muestra un mensaje de error "La Cédula de Identidad ingresada ya está registrada".
- **Escenario: Intentar registrar un usuario sin Cédula de Identidad.**
  - **Dado que** estoy en la pantalla de registro de usuarios,
  - **Cuando** intento guardar un usuario dejando el campo Cédula de Identidad vacío,
  - **Entonces** el sistema no guarda el usuario y muestra un mensaje de error "La Cédula de Identidad es obligatoria".

#### Notas Adicionales (Opcional):

- Ninguna.

---

#### **Historia de Usuario:** HU-003 - Realizar Préstamo de Libro

Como: Bibliotecario

Quiero: Registrar cuando un usuario toma prestado un libro

Para: Controlar la salida de los libros y su fecha de retorno.

Prioridad: Alta

Estimación (Puntos de Historia): 8

#### Criterios de Aceptación:

- **Escenario: Préstamo exitoso de un libro disponible.**
  - **Dado que** existe un libro con copias disponibles y un usuario registrado,
  - **Cuando** busco y selecciono el libro por su Título/ISBN y el usuario por su Cédula/ID, y confirmo el préstamo,
  - **Entonces** el sistema registra la fecha actual como fecha de préstamo, calcula y registra la fecha de devolución esperada (15 días después), disminuye la cantidad disponible del libro en el catálogo y muestra un mensaje de confirmación "Préstamo registrado exitosamente".
- **Escenario: Intentar prestar un libro sin disponibilidad.**
  - **Dado que** un libro está en el catálogo pero su "Cantidad Disponible" es cero,
  - **Cuando** intento realizar un préstamo de ese libro,
  - **Entonces** el sistema impide el préstamo y muestra un mensaje de error "Libro no disponible para préstamo".

- **Escenario: Intentar prestar a un usuario no registrado.**
  - **Dado que** existe un libro disponible,
  - **Cuando** intento realizar un préstamo para una Cédula de Identidad o ID de Miembro que no está registrada,
  - **Entonces** el sistema impide el préstamo y muestra un mensaje de error "Usuario no encontrado".

#### Notas Adicionales (Opcional):

- Asumir una política de préstamo de 15 días calendario.

---

#### Historias de Usuario Identificadas:

A continuación, se listan las historias de usuario que conforman el *Plan de Lanzamiento* o *Product Backlog* inicial del SGPL:

- **HU-001** - Registrar Nuevo Libro en el Catálogo
- **HU-002** - Registrar Nuevo Usuario
- **HU-003** - Realizar Préstamo de Libro
- **HU-004** - Realizar Devolución de Libro
- **HU-005** - Buscar Libros en el Catálogo
- **HU-006** - Ver Historial de Préstamos de un Usuario

## 2.2. Requisitos No Funcionales

Además de las historias de usuario, se deben identificar y listar los requisitos no funcionales relevantes para el Sistema de Gestión de Préstamos de Libros (SGPL).

- **Usabilidad:** La interfaz para el bibliotecario debe ser sencilla y fácil de usar, requiriendo mínima capacitación.
- **Rendimiento:** Las operaciones de búsqueda, registro, préstamo y devolución deben ejecutarse en menos de 1 segundo.
- **Seguridad:** La información personal de los usuarios (nombre, cédula, dirección, etc.) debe ser almacenada de forma segura y accesible solo por personal autorizado.
- **Fiabilidad:** El sistema debe garantizar que los datos de préstamos y devoluciones no se pierdan en caso de fallas.

---

## 3. Planificación del Sprint 1 (Semanas: 01/07/2025 - 14/07/2025)

El primer Sprint tendrá una duración de dos semanas, del 01/07/2025 al 14/07/2025. Durante este Sprint, el equipo se enfocará en las historias de usuario seleccionadas del *Product Backlog* inicial que son críticas para el núcleo del sistema.

## 3.2. Objetivos del Sprint 1

Los objetivos específicos para este Sprint son:

- Desarrollar la funcionalidad completa para registrar nuevos libros en el catálogo.
- Implementar la capacidad de registrar nuevos usuarios en el sistema.
- Construir la funcionalidad básica para registrar un préstamo de libro, incluyendo la verificación de disponibilidad.

## 3.3. Tareas del Sprint 1

Para cada historia de usuario seleccionada para el Sprint, se han desglosado las tareas individuales necesarias para su implementación, con su respectivo responsable.

### Historia de Usuario: HU-001 - Registrar Nuevo Libro en el Catálogo

- **Tarea 1:** Diseñar y prototipar la interfaz de usuario para el formulario de registro de libros.
  - **Responsable:** [Nombre del Estudiante A]
  - **Estimación:** 4 horas
- **Tarea 2:** Implementar la interfaz gráfica (Swing) y sus controladores básicos.
  - **Responsable:** [Nombre del Estudiante B]
  - **Estimación:** 8 horas
- **Tarea 3:** Desarrollar la lógica de validación y persistencia para el registro de libros.
  - **Responsable:** [Nombre del Estudiante C]
  - **Estimación:** 12 horas
- **Tarea 4:** Escribir y ejecutar pruebas unitarias para la lógica de registro y validaciones.
  - **Responsable:** [Nombre del Estudiante A]
  - **Estimación:** 6 horas

### Historia de Usuario: HU-002 - Registrar Nuevo Usuario

- **Tarea 1:** Diseñar y prototipar la interfaz de usuario para el formulario de registro de usuarios.
  - **Responsable:** [Nombre del Estudiante D]
  - **Estimación:** 4 horas
- **Tarea 2:** Implementar la interfaz gráfica (Swing) y sus controladores básicos.
  - **Responsable:** [Nombre del Estudiante A]
  - **Estimación:** 8 horas
- **Tarea 3:** Desarrollar la lógica de validación (ej. cédula única) y persistencia para el registro de usuarios.
  - **Responsable:** [Nombre del Estudiante E]

- **Estimación:** 10 horas
- **Tarea 4:** Escribir y ejecutar pruebas unitarias para la lógica de registro de usuarios.
  - **Responsable:** [Nombre del Estudiante D]
  - **Estimación:** 5 horas

### **Historia de Usuario: HU-003 - Realizar Préstamo de Libro**

- **Tarea 1:** Diseñar y prototipar la interfaz de usuario para la pantalla de préstamo (búsqueda, selección).
  - **Responsable:** [Nombre del Estudiante B]
  - **Estimación:** 6 horas
- **Tarea 2:** Implementar la interfaz gráfica (Swing) y sus controladores para el préstamo.
  - **Responsable:** [Nombre del Estudiante C]
  - **Estimación:** 12 horas
- **Tarea 3:** Desarrollar la lógica de búsqueda de libros y usuarios.
  - **Responsable:** [Nombre del Estudiante E]
  - **Estimación:** 15 horas
- **Tarea 4:** Implementar la lógica de verificación de disponibilidad, registro de préstamo y actualización de stock.
  - **Responsable:** [Nombre del Estudiante D]
  - **Estimación:** 10 horas
- **Tarea 5:** Escribir y ejecutar pruebas unitarias para la funcionalidad de préstamo.
  - **Responsable:** [Nombre del Estudiante E]
  - **Estimación:** 7 horas

### **3.4. Cronograma Detallado del Sprint 1 (01/07/2025 - 14/07/2025)**

Este cronograma proporciona una guía de cómo distribuir las actividades de desarrollo a lo largo de las dos semanas del Sprint.

#### **Semana 1: (01/07/2025 - 07/07/2025)**

- **Día 1 (Martes 01/07):**
  - Reunión de Planificación del Sprint (confirmación de Historias de Usuario y Tareas).
  - Inicio de Diseño de Interfaces (Wireframes/Prototipos de HU-001, HU-002, HU-003).
  - Configuración inicial del entorno de desarrollo y repositorios Git/GitHub.
- **Día 2 (Miércoles 02/07):**
  - Implementación de Interfaces Gráficas (Swing) para HU-001 y HU-002.
  - Inicio del desarrollo de la lógica de negocio para validaciones básicas (HU-001, HU-002).
- **Día 3 (Jueves 03/07):**
  - Continuación de la implementación de interfaces (HU-001, HU-002).

- Desarrollo de la lógica de persistencia para HU-001 y HU-002.
  - Inicio de Pair Programming en tareas críticas.
- **Día 4 (Viernes 04/07):**
  - Finalización de la implementación de HU-001.
  - Inicio de pruebas unitarias para HU-001.
  - Integración inicial del código de HU-001 en la rama de desarrollo (GitHub Flow).
- **Día 5 (Sábado 05/07):**
  - Continuación de la implementación de HU-002.
  - Continuación de pruebas unitarias para HU-002.
  - Reunión de Sincronización del equipo (revisión de progreso).
- **Día 6 y 7 (Domingo 06/07 - Lunes 07/07):**
  - Finalización de la implementación de HU-002.
  - Finalización de pruebas unitarias para HU-002.
  - Integración del código de HU-002.

## **Semana 2: (08/07/2025 - 14/07/2025)**

- **Día 8 (Martes 08/07):**
  - Reunión diaria de Stand-up.
  - Inicio del diseño e implementación de la interfaz para HU-003 (Préstamo).
  - Inicio de la lógica de búsqueda de libros y usuarios para HU-003.
- **Día 9 (Miércoles 09/07):**
  - Continuación de la implementación de la interfaz y lógica de búsqueda para HU-003.
  - Inicio de la lógica de verificación de disponibilidad del libro para préstamo.
- **Día 10 (Jueves 10/07):**
  - Desarrollo de la lógica para registrar el préstamo y actualizar el stock del libro.
  - Inicio de pruebas unitarias para HU-003.
- **Día 11 (Viernes 11/07):**
  - Finalización de la implementación de HU-003.
  - Finalización de pruebas unitarias para HU-003.
  - Integración del código de HU-003.
- **Día 12 (Sábado 12/07):**
  - Pruebas de Integración y Pruebas del Sistema de las funcionalidades desarrolladas en el Sprint.
  - Refactorización y limpieza de código.
- **Día 13 (Domingo 13/07):**
  - Preparación de la Demostración del Incremento (presentación al Grupo Docente).
  - Revisión final de la "Definición de Hecho" para todas las HU.
- **Día 14 (Lunes 14/07):**
  - **Presentación del Incremento 1 a los Stakeholders (Grupo Docente).**
  - Reunión de Retrospectiva del Sprint.

### 3.5. Definición de Hecho (Definition of Done)

Para este Sprint, una historia de usuario se considerará "hecha" cuando se cumplan todos los siguientes criterios:

- El código de la funcionalidad ha sido implementado y cumple con **todos los criterios de aceptación (Given-When-Then)** de la historia de usuario.
  - Todas las pruebas unitarias asociadas a la funcionalidad han sido escritas y pasan el 100%.
  - El código ha sido revisado por al menos otro miembro del equipo (Pair Programming o Code Review) y aprobado.
  - El código ha sido integrado exitosamente en la rama principal del repositorio (GitHub Flow), sin conflictos y las *builds* son exitosas.
  - La funcionalidad es demostrable al Grupo Docente y se ha preparado para la presentación.
  - La documentación relevante (si aplica, como comentarios en el código) está actualizada.
-