Planificación de Sprint N°1

Sistema de Biblioteca para la escuela Tecnico Aeronautica

| Duración | |
| --- | --- |
| Inicio | Fin |
| 29/09/10 | 13/10/2025 |

| Rol | Nombre |
| --- | --- |
| Scrum Master | Pablo Hernandez |
| Product Owner | Tomas Flores |
| Development Team | Pablo Hernandez y Tomas Flores |

30-09-2025

### Objetivo del Sprint

El objetivo principal de este primer sprint consiste en la implementación de las funcionalidades fundamentales del módulo de gestión de usuarios y seguridad dentro del Sistema de Gestión de Biblioteca.

Durante este sprint se busca desarrollar las capacidades esenciales para:

* La creación, modificación y eliminación de usuarios.
* La asignación de roles diferenciados que permitan establecer jerarquías de acceso y permisos en el sistema.
* La habilitación de un acceso confiable tanto en entornos web como móviles.
* La incorporación de medidas de seguridad orientadas a la protección de la información sensible.

El valor esperado de este sprint es garantizar que la institución cuente con una base sólida de administración de usuarios y políticas de seguridad, las cuales son imprescindibles para la correcta operación del sistema en fases posteriores.

### Sprint Backlog

A continuación se detallan las historias de usuario seleccionadas desde el Product Backlog, priorizadas y estimadas en puntos de historia de acuerdo con su complejidad:

| **N°** | **ID Backlog** | **Historia de Usuario** | **Prioridad** | **Estimación (pts)** | **Responsable** | **Estado inicial** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | PB-04.1 | Como administrador, quiero crear usuarios para gestionar el acceso al sistema. | Media | 5 | Development Team | Pendiente |
| 2 | PB-04.2 | Como administrador, quiero modificar usuarios existentes para mantener sus datos actualizados. | Media | 4 | Development Team | Pendiente |
| 3 | PB-04.3 | Como administrador, quiero eliminar usuarios del sistema para revocar accesos no autorizados. | Media | 3 | Development Team | Pendiente |
| 4 | PB-04.6 | Como administrador, quiero asignar roles diferenciados para controlar los permisos de acceso. | Media | 4 | Development Team | Pendiente |
| 5 | PB-03.2 | Como usuario lector, quiero acceder desde el celular para consultar libros en cualquier momento. | Alta | 4 | Development Team | Pendiente |
| 6 | PB-05.4 | Como equipo de desarrollo, queremos implementar mecanismos de seguridad para proteger la información del sistema. | Baja | 2 | Development Team | Pendiente |

### Criterios de Aceptación

| **ID Backlog** | **Criterio de Aceptación** |
| --- | --- |
| PB-04.1 | Crear usuarios: El sistema debe registrar un nuevo usuario, asociar credenciales únicas y reflejar en la lista de usuarios activos. |
| PB-04.2 | Modificar usuarios: Los administradores pueden actualizar datos de usuarios existentes, manteniendo un historial de cambios. |
| PB-04.3 | Eliminar usuarios: Solo usuarios con rol de administrador podrán eliminar registros; el acceso debe bloquearse inmediatamente. |
| PB-04.6 | Asignar roles: Se debe permitir asignar roles como administrador, bibliotecario o lector, y el sistema controlará los permisos de acceso de acuerdo con dichos roles. |
| PB-03.2 | Acceso móvil: El sistema debe ser completamente funcional y usable desde dispositivos móviles mediante diseño responsive. |
| PB-05.4 | Seguridad: Todas las contraseñas deben ser almacenadas con algoritmos de encriptación y las sesiones deben expirar automáticamente tras un periodo de inactividad. |

### Riesgos y Dependencias

Riesgos Identificados:

* Inconsistencias en la asignación de roles: La gestión de múltiples niveles de acceso puede generar conflictos si no se definen claramente los permisos. Esto podría derivar en errores de seguridad o mal funcionamiento de funcionalidades específicas.
* Complejidad técnica en protocolos de seguridad: La implementación de mecanismos de autenticación y autorización robustos (como JWT, OAuth2 o encriptación de datos) podría requerir más tiempo del estimado, afectando el cronograma del proyecto.
* Retrasos en la integración web-móvil: La sincronización entre la versión web y la aplicación móvil puede generar problemas de compatibilidad y dependencias entre módulos, provocando retrasos en la entrega de funcionalidades completas.
* Sobrecarga del equipo QA: Las múltiples pruebas de seguridad y funcionalidad aumentan la carga de trabajo del equipo de aseguramiento de calidad, lo que puede generar cuellos de botella si no se distribuyen adecuadamente las tareas.

Dependencias:

* Política de roles definida por el Product Owner: La correcta asignación de permisos depende de la definición clara de roles y responsabilidades dentro del sistema.
* Estándares de seguridad institucionales: La implementación de protocolos debe alinearse con las normas y buenas prácticas establecidas por la organización para garantizar la seguridad y cumplimiento regulatorio.
* Disponibilidad de ambientes de prueba: Para realizar pruebas funcionales y de integración, se requiere acceso a entornos de desarrollo y prueba tanto para web como para móvil, evitando interferencias con el entorno productivo.

### Plan de Trabajo

El plan de trabajo contempla la organización de las tareas en dos semanas, con reuniones diarias (Daily Scrum) para seguimiento y control de avances.

Reuniones:

* Daily Scrum: Reunión diaria a las 14:00 PM vía Discord, enfocada en revisar avances, identificar bloqueos y ajustar tareas pendientes.
* Sprint Review: Programada para el 14/10/2025, donde se presentarán los resultados y entregables al Product Owner, recopilando retroalimentación para ajustes posteriores.
* Sprint Retrospective: Posterior al Sprint Review, reunión dedicada a la reflexión sobre el proceso de trabajo, identificando mejoras en la organización, comunicación y eficiencia del equipo.

Definición de Done:

Para que una tarea se considere finalizada, debe cumplir con los siguientes criterios:

* Código versionado correctamente en GitHub, con commits descriptivos y estructurados.
* Funcionalidades desarrolladas con pruebas unitarias e integración completadas y aprobadas.
* Documentación técnica actualizada en el repositorio compartido, incluyendo diagramas, instrucciones de configuración y guías de uso.
* Validación de los criterios de aceptación por parte del Product Owner, asegurando que la funcionalidad cumpla con los requerimientos especificados.

Herramientas de apoyo:

* Visual Studio Code: Entorno de desarrollo integrado (IDE) utilizado para la escritura, depuración y gestión de código fuente de manera eficiente.
* MySQL: Sistema de gestión de bases de datos relacional, utilizado para el almacenamiento, consulta y gestión de los datos del proyecto.
* Angular: Framework de desarrollo front-end para construir interfaces de usuario dinámicas y responsivas en aplicaciones web.
* Django: Framework de desarrollo back-end en Python, que facilita la creación de aplicaciones web robustas, seguras y escalables.
* GitHub: Repositorio de control de versiones para gestionar el código fuente, seguimiento de cambios y colaboración entre el equipo de desarrollo.

### Compromiso del Equipo

El equipo de desarrollo asume el compromiso de entregar al término de este sprint un módulo funcional de gestión de usuarios y seguridad, que cumpla con los estándares de calidad y usabilidad esperados, garantizando la continuidad del proyecto y sirviendo como base para futuros incrementos en el sistema. Se asegura, además, que todas las funcionalidades desarrolladas estarán documentadas, probadas y alineadas con los requerimientos definidos por el Product Owner.