



**Universidad Metropolitana para  
la Educación y el Trabajo**

**Técnicas Avanzadas de la Programación**  
**“Spring Nro. 2”**

**Alumno:** Leandro Pasteur

**Profesora:** Antonieta Kuz

**Fecha:** 12/04/2025

**Índice:**

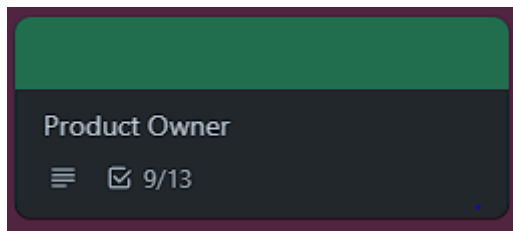
1) Tableros del proyecto (Scrum).....	PAG 3 - 11
2) Definir los casos de uso completos.....	PAG 12 - 17
3) Análisis funcional (requisitos funcionales y no funcionales).....	PAG 18
4) Diagrama de Historias de usuario.....	PAG 19
5) Diagrama burn down.....	PAG 19
6) Repositorio GitHub.....	PAG 19

**1) Tablero del proyecto “Jardín Maternal” Scrum:**

A continuación, detallo todas las características del tablero de tipo Scrum, hecho con la aplicación “TRELLO”.



Roles: En este apartado se encuentran cada uno de los integrantes que participan en el desarrollo de este proyecto:



**Ejemplo de como se visualizan los integrantes del proyecto en el Tablero de Trello**

**Product Owner (PO)**

Función Principal: Representa al cliente (jardín maternal), define el valor del producto y prioriza las tareas que generan más impacto para los usuarios.

**Analista Funcional (AF)**

Función Principal: Traducir las necesidades del usuario en especificaciones técnicas comprensibles para el equipo de desarrollo.

**Scrum Master (SM)**

Función Principal: Facilita el proceso SCRUM, elimina obstáculos y asegura que el equipo trabaje con fluidez bajo la metodología ágil.

**Documentador Técnico (D)**

Función Principal: Redactar y mantener actualizada toda la documentación del proyecto para su correcta entrega y evaluación.

**Diseñador UX/UI**

Función Principal: Diseñar la interfaz del sistema centrada en el usuario, asegurando una experiencia intuitiva, estética y accesible.

## Desarrollador Frontend (DF)

Función Principal: Implementar el diseño de interfaces en el navegador y conectar con el backend para mostrar los datos correctamente.

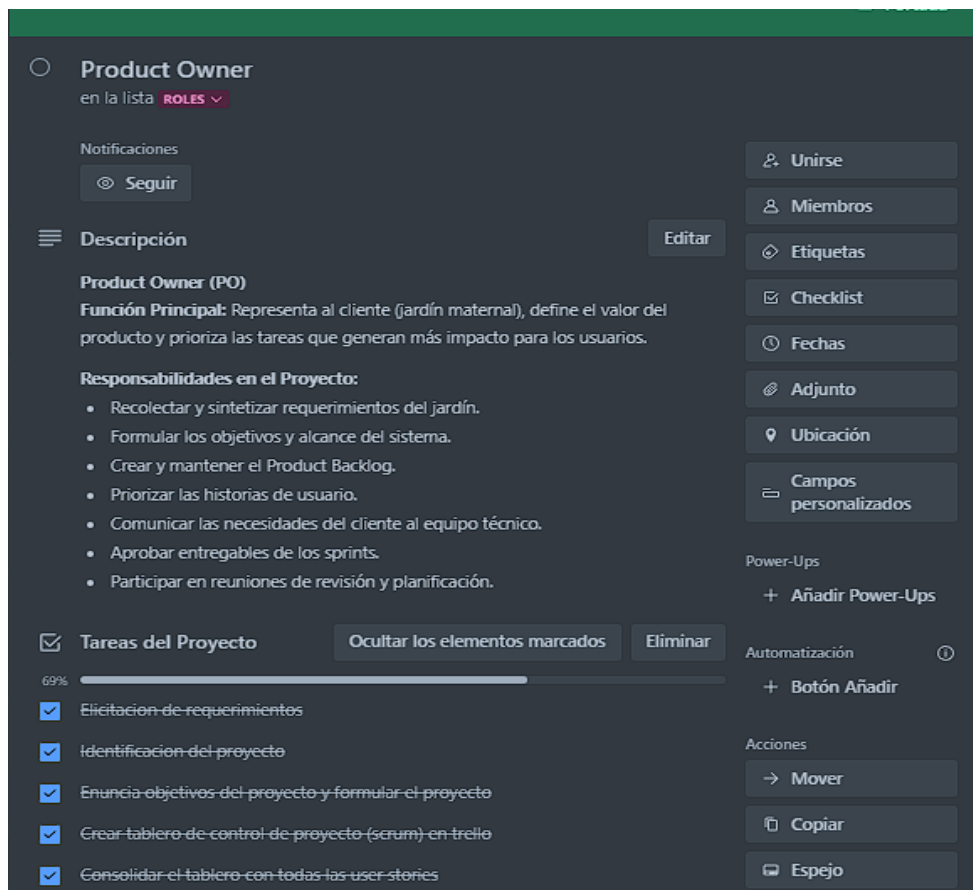
## Desarrollador Backend (DB)

Función Principal: Programar la lógica del sistema, la base de datos, API y las funcionalidades principales del sistema.

## Desarrollador Fullstack (FS)

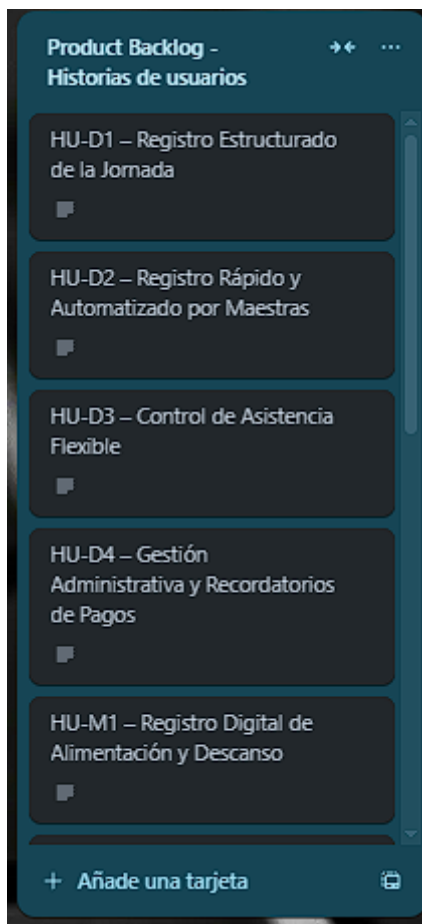
Función Principal: Es responsable de implementar tanto el frontend como el backend del sistema. Tiene una visión integral del desarrollo, lo que permite integrar y sincronizar correctamente la base de datos, la lógica del negocio y la interfaz de usuario.

**Dentro de cada tarjeta de los integrantes del proyecto se visualiza una descripción, y un apartado de Checklist con cada una de las tareas que debe realizar. Cuando se termina las tareas se debe marcar como realizada.**



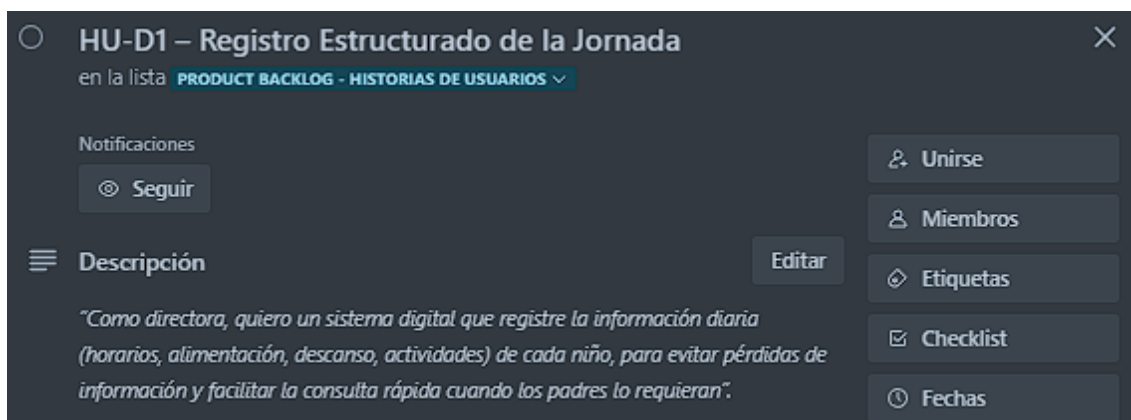
**Apartado llamado Product Backlog – Historias de usuario.**

Dentro de este apartado se encuentran tarjetas con cada una de las historias de usuarios:



**Ejemplo de cómo se visualizan las historias de uso en el Tablero de Trello**

Al ingresar se puede visualizar una descripción de cada una de las historias:



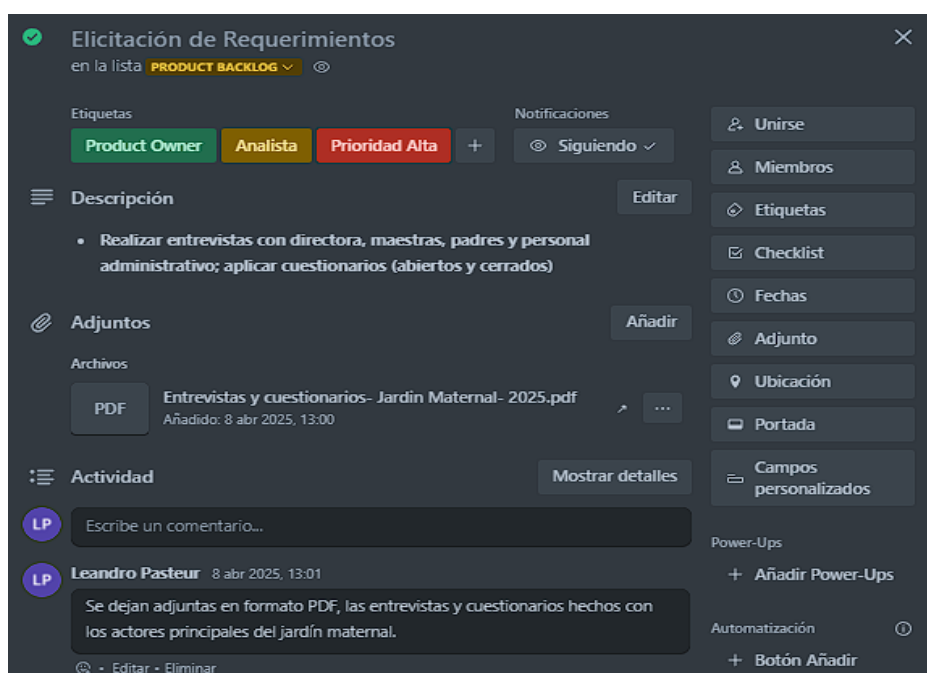
**Ejemplo de cómo se visualizan las tarjetas de las historias de uso en el Tablero de Trello**

**Apartado llamado Product Backlog.**

En este apartado se detallan todas las tareas que se van a desarrollar a lo largo del proyecto:

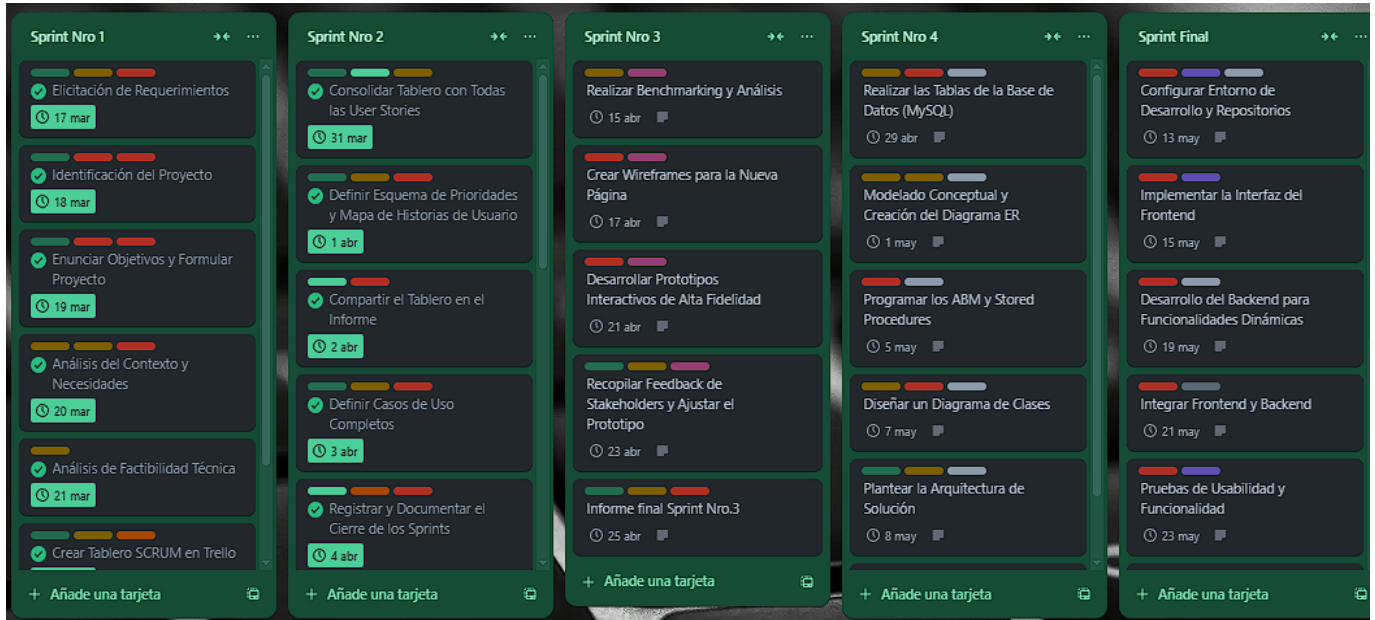


Dentro de las tarjetas se pueden visualizar etiquetas donde se detalla quien debe realizar la tarea y cuál es su prioridad (Alta, Media o Baja). Además, se realiza una descripción de la tarea a realizar.



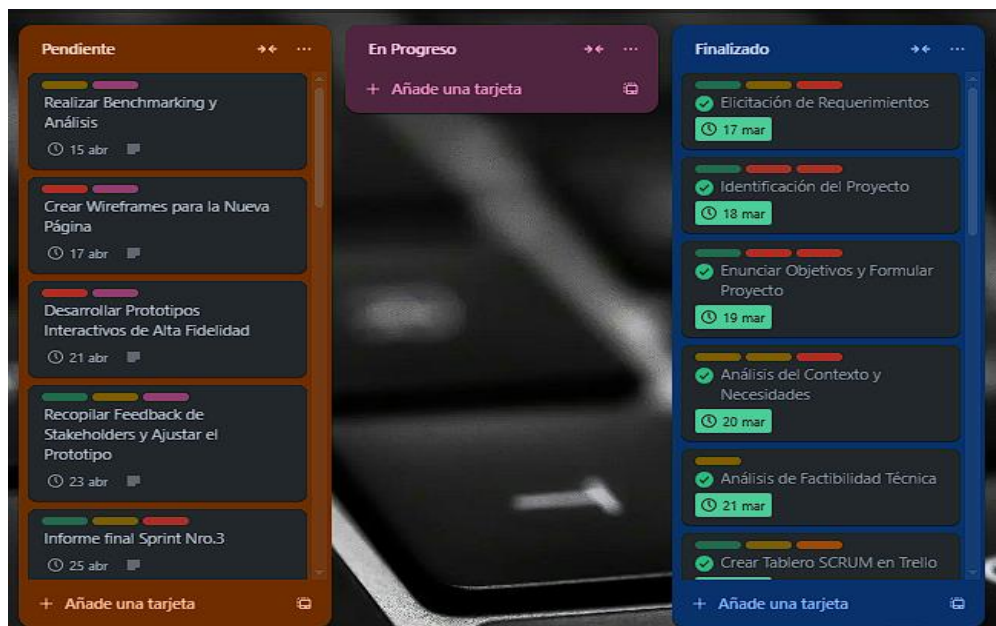
### Apartados de Sprints

Armamos este apartado para dividir las tareas por número de sprint, detallando quien debe realizar la tarea, cuál es su prioridad y las fechas de finalización de cada tarea (marcadas con una tilde cuando están realizadas).



### Apartados con estado de las tareas

En estos apartados se pueden ver los estados de las tareas (Pendiente, En proceso, Finalizado).



Link del Trello: <https://trello.com/b/tLO7Qah/proyecto-sistema-jardin-maternal-scrum>

### 1) Tablero de tareas hecho en Excel:

Se divide en las siguientes columnas:

- Realizamos un tablero donde detallamos todas las tareas del proyecto.
- Numero de Sprint
- Tarea
- Quien la realizó
- Estado
- Los diez días que dura cada Sprint (lunes a viernes por dos semanas)
- Las horas pendientes
- Fecha de duración de la tarea

### Sprint 1:

Nro. de Sprint	Tarea	Quién	Estado	Día 1/ Lunes	Día 2/ Martes	Día 3/ Miércoles	Día 4/ Jueves	Día 5/ Viernes	Día 6/ Lunes	Día 7/ Martes	Día 8/ Miércoles	Día 9/ Jueves	Día 10/ Viernes	Horas Pendientes	Fecha de duración
1	Elicitación de Requerimientos	Product owner/ Analista	Completada	8	0	0	0	0	0	0	0	0	0	0 hrs	17/3/2025
1	Identificación del Proyecto	Product Owner	Completada	0	8	0	0	0	0	0	0	0	0	0 hrs	18/3/2025
1	Enunciar Obj. y Formular el Proyecto	Product owner/ Documentador	Completada	0	0	8	0	0	0	0	0	0	0	0 hrs	19/3/2025
1	Análisis del Contexto y Necesidades	Analista/ Documentador	Completada	0	0	0	8	0	0	0	0	0	0	0 hrs	20/3/2025
1	Análisis de Factibilidad Técnica	Analista	Completada	0	0	0	0	8	0	0	0	0	0	0 hrs	21/3/2025
1	Crear Tablero de Control del Proyecto (SCRUM) en	Product owner/ Scrum Master	Completada	0	0	0	0	0	8	0	0	0	0	0 hrs	24/3/2025
1	Informe Final del Sprint Nro.1	Product owner/ Documentador	Completada	0	0	0	0	0	0	8	0	0	0	0 hrs	25/3/2025
Final del Sprint Nro 1 - Desde el día 17/03/2025 al 27/03/2025															



## Sprint 2:

2	Consolidar Tablero con Todas las User Stories	Product owner/ Analista	Completada	8	0	0	0	0	0	0	0	0	0	0 hrs	31/3/2025
2	Definir Esquema de Prioridades y Mapa de	Product owner/ Documentador	Completada	0	8	0	0	0	0	0	0	0	0	0 hrs	1/4/2025
2	Compartir el Tablero en el Informe	Documentador	Completada	0	0	8	0	0	0	0	0	0	0	0 hrs	2/4/2025
2	Definir Casos de Uso Completos	Product owner/ Analista	Completada	0	0	0	8	0	0	0	0	0	0	0 hrs	3/4/2025
2	Registrar y Documentar el Cierre de los Sprints	Serum Master/ Documentador	Completada	0	0	0	0	8	0	0	0	0	0	0 hrs	4/4/2025
2	Realizar Análisis Funcional (RF y RNF)	Analista	Completada	0	0	0	0	0	8	0	0	0	0	0 hrs	7/4/2025
2	Ajuntar Diagrama de Casos de Uso/Mapa	Analista/ Documentador	Completada	0	0	0	0	0	0	8	0	0	0	0 hrs	8/4/2025
2	Elaborar Gráfico Burn Down	Serum Master/ Documentador	Completada	0	0	0	0	0	0	0	8	0	0	0 hrs	9/4/2025
2	Crear Repositorio GitHub y Cargar Documentación	Documentador	Completada	0	0	0	0	0	0	0	0	8	0	0 hrs	10/4/2025
2	Informe final Sprint Nro.2	Product owner/ Documentador	Completada	0	0	0	0	0	0	0	0	0	8	0 hrs	11/4/2025
Final del Sprint Nro 2 - Desde el día 31/03/2025 al 11/04/2025															

### Sprint 3:

3	Realizar Benchmarking y Análisis	Diseñador/UI	Pendiente	8	8	0	0	0	0	0	0	0	0	16 hs	14/04/2025 - 15/04/2025
	Crear Wireframes para la nueva pagina	Diseñador/UI	Pendiente	0	0	8	8	0	0	0	0	0	0	16hs	16/04/2025 - 17/04/2025
3	Desarrollar Prototipos Interactivos de Alta	Diseñador/UI	Pendiente	0	0	0	0	8	8	0	0	0	0	16 hs	18/04/2025 - 21/04/2025
3	Recopilar Feedback Stakeholders / Ajustar	Product Owner/ Diseñador/UI	Pendiente	0	0	0	0	0	0	8	8	0	0	16 hs	22/04/2025 - 23/04/2025
3	Informe final Sprint Nro.3	Product Owner/ Documentador	Pendiente	0	0	0	0	0	0	0	0	8	8	16 hs	24/04/2025 - 25/04/2025
Final del Sprint Nro 3 - Desde el día 14/04/2025 al 25/04/2025															

4	Realizar las Tablas de la Base de Datos (MySQL)	Analista/ Desarrollador Backend	Pendiente	8	8	0	0	0	0	0	0	0	0	16 hs	28/04/2025 - 29/04/2025
4	Modelado Conceptual y Creación del Diagrama ER	Analista/ Desarrollador Backend	Pendiente	0	0	8	8	0	0	0	0	0	0	16 hs	30/04/2025 - 1/05/2025
4	Programar ABM y Stored Procedures	Desarrollador Backend	Pendiente	0	0	0	0	8	8	0	0	0	0	16 hs	2/05/2025 - 5/05/2025
4	Diseñar un Diagrama de Clases	Documentador/ Desarrollador Backend	Pendiente	0	0	0	0	0	0	8	8	0	0	16 hs	6/05/2025 - 7/05/2025
4	Plantear la Arquitectura de Solución	Product owner/ Desarrollador Backend	Pendiente	0	0	0	0	0	0	0	0	8	0	8 hs	8/5/2025
4	Informe final Sprint Nro.4	Product owner/ Documentador	Pendiente	0	0	0	0	0	0	0	0	0	8	8 hrs	9/5/2025
Final del Sprint Nro 4 - Desde el día 28/04/2025 al 09/05/2025															

Final	Configurar Entorno de Desarrollo y Repositorio	Desarrollador Backend/ Desarrollador Frontend	Pendiente	8	8	0	0	0	0	0	0	0	0	16 hs	12/05/2025 - 13/05/2025
Final	Implementar la Interfaz del Frontend	Desarrollador Frontend	Pendiente	0	0	8	8	0	0	0	0	0	0	16 hs	14/05/2025 - 15/05/2025
Final	Desarrollo del Backend para Funcionalidades	Desarrollador Backend	Pendiente	0	0	0	0	8	8	0	0	0	0	16 hs	16/05/2025 - 19/05/2025
Final	Integrar Frontend y Backend	Fullstack	Pendiente	0	0	0	0	0	0	8	8	0	0	16 hs	20/05/2025 - 21/05/2025
Final	Pruebas de Usabilidad y Funcionalidad	Desarrollador Frontend	Pendiente	0	0	0	0	0	0	0	0	8	8	16 hs	22/05/2025 - 23/05/2025
Final del Sprint Final Parte 1 - Desde el día 12/05/2025 al 23/05/2025															
Final	Optimización y Ajustes de Rendimiento	Desarrollador Frontend	Pendiente	8	8	0	0	0	0	0	0	0	0	16 hs	26/05/2025 - 27/05/2025
Final	Deployment y Configuración del	Desarrollador Backend	Pendiente	0	0	8	0	0	0	0	0	0	0	8 hrs	28/5/2025
Final	Revisión Final, Correcciones y	Product Owner	Pendiente	0	0	0	8	0	0	0	0	0	0	8 hrs	29/5/2025
Final	Documentación Técnica y Manual de Usuario	Documentador	Pendiente	0	0	0	0	8	0	0	0	0	0	8 hrs	30/5/2025
Final del Sprint Final Parte 2 - Desde el día 26/04/2025 al 30/05/2025															

**Al final de la tabla se puede visualizar las horas trabajadas y las horas restantes**

<b>Horas Realizadas</b>	<b>136 Horas</b>
<b>Horas Restantes</b>	<b>280 Horas</b>

- *Se adjuntará a la entrega el archivo Excel de la tabla. Además, se encuentra adjunto en el Trello (Product Backlog / “Crear Tablero de Control del Proyecto (SCRUM)”)*

**2) Definir casos de uso: A continuación, definiremos los casos de uso dentro del proyecto.**

### **CU01 Registrar Asistencia**

**Actores:**

Directora / Maestra y Niño (identificado mediante QR o huella)

**Precondiciones:**

- El usuario debe estar autenticado.
- El dispositivo (lector de QR o sensor biométrico) debe estar operativo.

**Flujo Principal:**

- ✓ La maestra inicia sesión en el sistema.
- ✓ Se activa el módulo de registro.
- ✓ El dispositivo capta la identificación del niño mediante QR o huella.
- ✓ El sistema registra la asistencia con la hora exacta.
- ✓ Se muestra una confirmación visual y/o sonora.

**Flujos Alternativos:**

- Si la identificación falla, se solicita reintentar el escaneo o se activa el registro manual temporal.

**Postcondiciones:**

- El registro queda almacenado con fecha y hora.

**Requerimientos Especiales:**

- Respuesta rápida y de alta precisión.

**CU02 Registrar Alimentación y Descanso**

**Actores:** Maestra

**Precondiciones:**

- La maestra está autenticada.
- Disponibilidad de formularios digitales preconfigurados.

**Flujo Principal:**

- ✓ La maestra accede al módulo correspondiente.
- ✓ Selecciona al niño y completa el formulario de alimentación (tipo, cantidad, hora) y descanso (inicio, fin, observaciones).
- ✓ El sistema guarda la información en la base de datos.
- ✓ Se muestra una confirmación del registro.

**Flujos Alternativos:**

- Si falta información, el sistema solicita completarla antes de guardar.

**Postcondiciones:**

- Los datos se asocian al historial del niño.

**Requerimientos Especiales:**

- Formularios adaptables para necesidades específicas (alergias, dietas especiales).

**CU03 Enviar Notificaciones en Tiempo Real**

**Actores:** Sistema (con intervención de maestra).

**Precondiciones:**

- El sistema debe estar configurado para notificaciones.
- Los padres tienen dispositivos con notificaciones activadas.

**Flujo Principal:**

- ✓ La maestra registra un evento significativo (incidente, cambio de horario, etc.).
- ✓ El sistema procesa la información y genera una notificación.
- ✓ La notificación se envía a los dispositivos de los padres.
- ✓ El padre recibe el aviso y puede confirmar o responder.

**Flujos Alternativos:**

- Si el envío falla, se reintenta y se notifica a la maestra.

**Postcondiciones:**

- El evento se queda registrado en el historial de notificaciones.

**Requerimientos Especiales:**

- Baja latencia y alta fiabilidad.

**CU04 Comunicación Directa (Chat)**

**Actores:** Maestra y Padres

**Precondiciones:**

- Los usuarios deben estar autenticados y tener acceso al módulo de chat.

**Flujo Principal:**

- ✓ La maestra inicia una conversación a través del chat.
- ✓ Se envían mensajes (texto, imágenes o archivos).
- ✓ El sistema guarda y muestra el historial del chat.
- ✓ El padre puede responder y la conversación queda registrada.

**Flujos Alternativos:**

- Si falla el envío, se notifica al usuario y se sugiere reintentar.

**Postcondiciones:**

- Se registra el historial completo de la conversación.

**Requerimientos Especiales:**

- Interfaz intuitiva y soporte para multimedia.

**CU05 Gestión Digital de Inscripciones y Pagos**

**Actores:** Administrativo, Padre y Directora.

**Precondiciones:**

- El administrativo tiene acceso al módulo.
- Los padres ya están registrados en el sistema.
- 

**Flujo Principal:**

- ✓ El administrativo ingresa nuevos registros o actualiza existentes.
- ✓ Se generan cuotas y se procesa el pago digitalmente.
- ✓ El sistema actualiza el estado del pago y notifica recordatorios en caso de mora.

**Flujos Alternativos:**

- En caso de error en la transacción, se notifica y se solicita reintentar.

**Postcondiciones:**

- Los registros de inscripciones y pagos se actualizan correctamente.

**Requerimientos Especiales:**

- Seguridad en la transacción y validación de datos.

**CU06 Generar Reportes**

**Actores:** Directora y Administrativo

**Precondiciones:**

- El sistema cuenta con datos completos (asistencia, alimentación, descansos, pagos).

**Flujo Principal:**

- ✓ El director solicita la generación del reporte.
- ✓ El sistema procesa la información y muestra gráficos y tablas con indicadores.
- ✓ El usuario puede filtrar y visualizar los datos según lo necesite.

**Flujos Alternativos:**

- Si faltan datos, el sistema notifica las entradas incompletas.

**Postcondiciones:**

- Se visualizan reportes claros y actualizados.

**Requerimientos Especiales:**

- Interfaz intuitiva y opciones de filtrado.

**CU07 Sistema de Alertas para Emergencias**

**Actores:** Sistema (con intervención del personal del jardín y padres)

**Precondiciones:**

- El sistema debe estar configurado para detectar eventos críticos.
- Los dispositivos de notificación deben estar activos.

**Flujo Principal:**

- ✓ Se detecta un evento de emergencia (por ejemplo, incidente médico).
- ✓ El sistema envía alertas inmediatas a padres y personal.
- ✓ La notificación contiene información básica y, de ser posible, ubicación o detalles del incidente.

**Flujos Alternativos:**

- Si no se envía, se reintenta y se notifica al personal.

**Postcondiciones:**

- La emergencia queda registrada y notificada en el historial.

**Requerimientos Especiales:**

- Baja latencia y alta robustez en el envío.

**CU08 Informe Final y Actualización del Tablero SCRUM**

**Actores:** Product Owner y Scrum Master

**Precondiciones:**

- Todos los sprints deben estar cerrados y documentados.

**Flujo Principal:**

- ✓ El Product Owner recoge la evidencia del tablero SCRUM y de cada sprint.
- ✓ Se elabora un informe final que consolida el progreso, diagramas y métricas.
- ✓ Se comparte el informe.

**Postcondiciones:**

- El informe final queda archivado y disponible para revisión.

**Requerimientos Especiales:**

- Documentación clara, con evidencia visual y datos actualizados.



**3) Requisitos funcionales y no funcionales**

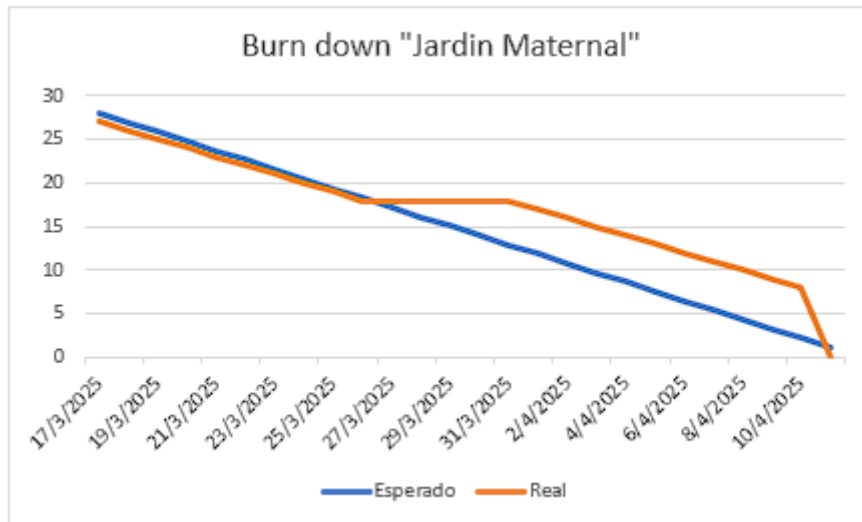
Requisitos Funcionales	Requisitos No Funcionales
<ul style="list-style-type: none"> <li>• El sistema debe permitir el registro automático de asistencia mediante QR o huella dactilar.</li> <li>• El sistema debe permitirles a las maestras registrar alimentación y descansos de cada niño.</li> <li>• El sistema debe enviar notificaciones en tiempo real a los padres sobre eventos importantes.</li> <li>• El sistema debe contar con un módulo de comunicación (chat) entre maestras y padres.</li> <li>• El sistema debe gestionar inscripciones de alumnos, incluyendo información personal y médica.</li> <li>• El sistema debe permitir realizar y gestionar pagos digitales de cuotas.</li> <li>• El sistema debe generar reportes de asistencia, alimentación y descanso por niño.</li> <li>• El sistema debe emitir alertas en caso de emergencias médicas o incidentes.</li> <li>• El sistema debe registrar el historial del niño (comidas, siestas, eventos, reportes).</li> <li>• El sistema debe registrar usuarios con diferentes roles: administrador, maestra o padre.</li> <li>• El sistema debe permitir a los docentes subir imágenes y observaciones del día.</li> </ul>	<ul style="list-style-type: none"> <li>• El sistema debe ser accesible desde dispositivos móviles y de escritorio.</li> <li>• El sistema debe tener estar disponible durante el horario escolar (08:00 a 18:00).</li> <li>• El tiempo de respuesta ante una acción del usuario debe ser instantáneo.</li> <li>• El sistema debe proteger los datos personales de acuerdo con la normativa de protección de datos.</li> <li>• La autenticación de usuarios debe contar con encriptación y niveles de acceso diferenciados.</li> <li>• Las notificaciones deben enviarse instantáneamente desde el momento del evento.</li> <li>• El sistema debe permitir escalar en cantidad de usuarios sin perder rendimiento.</li> <li>• Las actualizaciones del sistema no deben interrumpir el servicio y deben ser programadas.</li> <li>• La interfaz debe ser intuitiva y usable incluso por personas con conocimientos básicos de informática.</li> <li>• El sistema debe ser capaz de integrarse con plataformas de pago seguras (MercadoPago, etc.).</li> </ul>

#### **4) Historias de usuario**

El diagrama de historias de usuario fue realizado con la aplicación miro desde su sitio web. A continuación, dejo el link de acceso al Mapa:

<https://miro.com/app/board/uXjVIDygLlg=/>

#### **5) Diagrama Burn Down:**



En este grafico decidimos poner 28 puntos esperados, que se van restando a lo largo de los 26 días que duraron tanto el Sprint 1 sumando al Sprint 2. Los días no laborables que son los fines de semana, no se restaron puntos.

**Se adjuntará a la entrega el archivo Excel**

#### **6) Repositorio del proyecto en GITHUB**

<https://github.com/Leanpas98/Tecnicas-avanzadas-de-la-programacion---Proyecto-Jardin-Maternal->