

Sistema de Gestión de Tiempo y Productividad dentro de los estudiantes y docentes de la universidad de Cundinamarca

IEEE Std. 830-1998

11 de Octubre de 2024

Resumen

Este documento presenta el desarrollo de una aplicación destinada a mejorar la gestión del tiempo y la productividad de los estudiantes de la Universidad de Cundinamarca. La problemática identificada radica en las dificultades que muchos estudiantes enfrentan para organizar su tiempo, lo que afecta su rendimiento académico. Como solución, se propone una plataforma que combina el método Pomodoro para la gestión del tiempo y la metodología Kanban para la priorización de tareas. La aplicación se inspira en herramientas como Jira, Trello y Notion, y ofrece un valor agregado con un temporizador Pomodoro y plantillas de proyectos específicos.

El objetivo general del proyecto es crear un sistema que ayude a los estudiantes a mejorar su organización y productividad mediante la integración de técnicas de administración del tiempo y priorización de tareas. Los objetivos específicos incluyen la implementación de un temporizador Pomodoro, el desarrollo de un tablero Kanban intuitivo y la generación de reportes de progreso. Estos elementos permitirán a los estudiantes visualizar su carga de trabajo, organizar sus tareas según su importancia y analizar su rendimiento, facilitando la mejora continua en su gestión del tiempo.

Índice

Resumen.....	1
1. Introducción.....	3
1.1. Propósito.....	3
1.2. Ámbito del Sistema.....	3
1.3. Definiciones, Acrónimos y Abreviaturas.....	3
1.4. Referencias.....	3
1.5. Visión General del Documento.....	4
2. Descripción General.....	4
2.1. Perspectiva del Producto.....	4
2.2. Funciones del Producto.....	4
2.3. Características de los Usuarios.....	4
2.4. Restricciones.....	4
2.5. Suposiciones y Dependencias.....	4
2.6. Requisitos Futuros.....	4
3. Metodología Prototipado y desarrollo Iterativo.....	5
3.1. Diseño del Prototipo Inicial.....	5
3.2. Desarrollo de Prototipos Funcionales.....	5
3.3. Recopilación de Retroalimentación.....	5
3.4. Revisión y Refinamiento de Requisitos.....	5
3.5. Implementación de Funciones Específicas.....	5
4. Requisitos Específicos.....	5
3.1. Interfaces Externas.....	5
3.2. Funciones.....	6
3.2.1. Requisitos no funcionales.....	9
3.3. Requisitos de Rendimiento.....	9
3.4. Restricciones de Diseño.....	9
3.5. Atributos del Sistema.....	9
3.6. Otros Requisitos.....	9
3.6.1. Requisitos de almacenamiento.....	9
3.6.1.1. Estimación de almacenamiento por usuario.....	10
3.6.2. Requisitos de red.....	10
5. Apéndices.....	11
5.1. Diagrama de casos de uso.....	11
5.2. Modelo entidad-relación(MER).....	12
5.3. Diagrama de secuencia.....	13
5.4. Mockups.....	15
5.5. Cronograma de actividades.....	15
4.6. Descripción del equipo de desarrollo.....	15

1. Introducción

1.1. Propósito

El propósito de este documento es definir de manera detallada los requerimientos funcionales y no funcionales del Sistema de Gestión de Productividad. Este sistema permitirá a los usuarios gestionar de manera eficiente sus proyectos, tareas y tiempos de trabajo mediante funcionalidades como tableros Kanban, notas, un temporizador Pomodoro y un calendario con recordatorios. Este documento está dirigido a los desarrolladores, clientes y partes interesadas involucradas en el desarrollo del sistema.

1.2. Ámbito del Sistema

El Sistema permitirá a los usuarios crear y gestionar proyectos, organizar tareas en diferentes estados (Por hacer, En proceso, Terminado), tomar notas, controlar su tiempo de concentración utilizando el temporizador Pomodoro, y planificar eventos en un calendario. Los usuarios tendrán funcionalidades CRUD para cada módulo, y el sistema será accesible a través de una interfaz web adaptable a dispositivos móviles. Además, contará con un sistema de autenticación para usuarios registrados y administradores, que podrán monitorear la actividad del sistema.

1.3. Definiciones, Acrónimos y Abreviaturas

Nombre	Definición
CRUD	Create, Read, Update, Delete (Crear, Leer, Actualizar, Eliminar).
Pomodoro	Técnica de gestión de tiempo que divide el trabajo en intervalos de concentración y descanso.
Kanban	Sistema visual de gestión de tareas que organiza el trabajo en columnas según su estado.
UX/UI	User Experience / User Interface (Experiencia de Usuario / Interfaz de Usuario).
QA	Quality Assurance (Aseguramiento de Calidad)

1.4. Referencias

Título del Documento	Referencia
Standard IEEE 830 - 2008	IEEE

1.5. Visión General del Documento

Este documento está organizado en cuatro secciones principales. La sección 1 proporciona una introducción general al sistema. La sección 2 describe la visión global del producto, sus funciones, usuarios y restricciones. La sección 3 detalla los requisitos específicos del sistema, tanto funcionales como no funcionales. Finalmente, la sección 4 presenta apéndices con información adicional relevante.

2. Descripción General

2.1. Perspectiva del Producto

El Sistema es una aplicación web diseñada para individuos y equipos que deseen organizar y gestionar sus proyectos de manera eficiente. Ofrece herramientas de organización de tareas a través de tableros Kanban, un temporizador basado en la técnica Pomodoro y un calendario para eventos y recordatorios. Está dirigido a estudiantes y profesionales que busquen mejorar su productividad.

2.2. Funciones del Producto

- **Gestión de Proyectos:** Los usuarios pueden crear proyectos que contengan listas de tareas y notas.
- **Tableros Kanban:** Los usuarios pueden organizar sus tareas en diferentes estados.
- **Temporizador Pomodoro:** Funcionalidad para gestionar intervalos de trabajo y descanso.
- **Calendario:** Visualización de eventos y recordatorios.
- **Autenticación:** Sistema de registro y login para garantizar la seguridad de los datos de los usuarios.

2.3. Características de los Usuarios

El sistema está diseñado para dos tipos de usuarios principales:

- **Usuarios:** Aquellos que crean y gestionan sus proyectos y tareas.
- **Administradores:** Tienen permisos adicionales para monitorear la actividad del sistema y realizar tareas de mantenimiento.

2.4. Restricciones

- El sistema debe ser accesible únicamente para usuarios registrados.
- La aplicación debe ser compatible con los principales navegadores web.
- Se debe garantizar un rendimiento óptimo para al menos 500 usuarios concurrentes.

2.5. Suposiciones y Dependencias

- Se pretende que todos los empleados tengan conocimientos básicos de operación de sistemas.
- Se asume que los usuarios tendrán acceso a Internet y dispositivos con navegadores modernos.

2.6. Requisitos Futuros

- Posibilidad de integrar más herramientas de gestión de proyectos, como la asignación de tareas a múltiples usuarios.
- Incorporación de análisis de productividad y reportes visuales.

- Expansión de roles y permisos de usuario.

3. Metodología Prototipado y desarrollo Iterativo

3.1. Diseño del Prototipo Inicial

Se desarrollará un prototipo inicial que incluirá las funciones básicas del sistema, como la gestión de tareas, el temporizador Pomodoro y los tableros Kanban. Este prototipo se utilizará para demostrar las capacidades fundamentales del sistema y recoger retroalimentación inicial de los usuarios.

3.2. Desarrollo de Prototipos Funcionales

A medida que se avance en el desarrollo del sistema, se crearán prototipos funcionales que incluirán características adicionales y mejoras basadas en la retroalimentación recibida. Cada iteración del prototipo se someterá a pruebas de usabilidad y rendimiento para asegurar que se cumplan los requisitos de los usuarios.

3.3. Recopilación de Retroalimentación

Después de cada fase de desarrollo, se llevará a cabo una sesión de recopilación de retroalimentación con los usuarios. Esta retroalimentación se utilizará para identificar áreas de mejora y ajustar los requisitos del sistema antes de avanzar a la siguiente iteración.

3.4. Revisión y Refinamiento de Requisitos

Los requisitos se revisarán y refinarán continuamente a medida que se recopile retroalimentación de los usuarios. Este enfoque permitirá que el sistema evolucione de acuerdo con las necesidades cambiantes de los usuarios, asegurando su relevancia y utilidad.

3.5. Implementación de Funciones Específicas

Las funciones específicas del sistema se implementarán en iteraciones, comenzando por las más críticas y avanzando hacia características adicionales basadas en la priorización de las necesidades del usuario. Esto asegurará un desarrollo ágil y eficiente.

4. Requisitos Específicos

3.1. Interfaces Externas

- **Interfaz Web:** El sistema debe ser accesible desde cualquier navegador compatible con HTML5, CSS3 y JavaScript.

3.2. Funciones

Actividad # RF1	
Nombre de la actividad	Descripción
Inicio de sesión	<p>El sistema debe permitir que los usuarios registrados inicien sesión de manera segura mediante la autenticación de sus credenciales (usuario y contraseña). Dependiendo del tipo de usuario, el sistema debe otorgar los permisos y accesos correspondientes.</p> <p>Criterios de aceptación:</p> <ul style="list-style-type: none">● El usuario puede acceder al sistema introduciendo un correo electrónico válido y una contraseña.● Si las credenciales son incorrectas, se mostrará un mensaje de error claro.● El sistema debe autenticar al usuario en menos de 3 segundos.
Actividad # RF2	
Nombre de la actividad	Descripción
Registro de Usuario	<p>Proporcionará a los nuevos usuarios la posibilidad de crear una cuenta en el sistema. Este proceso incluirá la solicitud de información básica como el nombre, correo electrónico y contraseña, los cuales serán almacenados de manera segura para permitir la autenticación futura.</p> <p>Criterios de aceptación:</p> <ul style="list-style-type: none">● El usuario puede crear una nueva cuenta proporcionando un correo electrónico válido y una contraseña.● El sistema validará que el correo electrónico no esté registrado previamente.● Si el correo ya está registrado, se mostrará un mensaje de error al usuario.● El sistema debe confirmar la creación de la cuenta mediante un correo electrónico de validación.
Actividad # RF3	
Nombre de la actividad	Descripción
CRUD proyectos	<p>Permitirá a los usuarios gestionar sus proyectos de manera flexible. Los usuarios podrán crear múltiples proyectos, cada uno de los cuales podrá contener listas de tareas y notas asociadas. Esta funcionalidad también ofrecerá la opción de editar y eliminar proyectos según las necesidades de cada usuario, permitiendo así una gestión completa de las actividades.</p>

	Criterios de aceptación: <ul style="list-style-type: none"> ● El usuario puede crear un proyecto con un único nombre. ● El sistema permite asociar notas y listas de tareas al proyecto creado. ● El usuario puede editar los detalles del proyecto y ver los cambios reflejados en tiempo real. ● El usuario puede eliminar un proyecto y se eliminarán también las notas y tareas asociadas.
Actividad # RF4	
Nombre de la actividad	Descripción
CRUD Notas	<p>Ofrecerá la posibilidad de crear, editar y eliminar notas dentro del sistema. Las notas podrán asignarse a tableros específicos dentro de los proyectos, lo que facilitará su organización y asociación con tareas o ideas relevantes a un proyecto en particular.</p> <p>Criterios de aceptación:</p> <ul style="list-style-type: none"> ● El usuario puede crear, editar y eliminar una nota. ● Las notas pueden ser asociadas a tableros específicos dentro de los proyectos. ● Las notas deben poder visualizarse en los tableros correspondientes inmediatamente después de su creación o edición.
Actividad # RF5	
Nombre de la actividad	Descripción
CRUD Tareas	<p>Permitirá a los usuarios crear, modificar y eliminar tareas. Estas tareas podrán clasificarse en tres categorías: "Por hacer" (lista de tareas pendientes), "En proceso" (tareas que están siendo trabajadas actualmente) y "Terminadas" (tareas completadas). Esta categorización facilitará el seguimiento del progreso de las tareas dentro de los proyectos.</p> <p>Criterios de aceptación:</p> <ul style="list-style-type: none"> ● El usuario puede crear, editar y eliminar tareas. ● Las tareas pueden clasificarse en tres estados: "Por hacer", "En proceso", "Terminadas". ● El cambio de estado de una tarea debe reflejarse en los tableros de tareas correspondientes de forma instantánea. ● Al eliminar una tarea, debe desaparecer de los tableros sin causar errores.
Actividad # RF6	
Nombre de la actividad	Descripción

CRUD Tableros	<p>Proporcionará una estructura visual y organizativa a las tareas. Los usuarios podrán crear tableros donde organizar sus tareas según su estado. Los tableros incluirán tres listas: "Por hacer", donde se colocarán las tareas pendientes; "En proceso", donde se agruparán las tareas que estén en curso; y "Terminadas", que listará las tareas ya completadas. Esto permitirá una gestión visual eficiente y clara del trabajo.</p> <p>Criterios de aceptación:</p> <ul style="list-style-type: none"> • El usuario puede organizar sus tareas en tableros según los tres estados: "Por hacer", "En proceso" y "Terminadas". • Las tareas deben moverse entre los estados mediante una funcionalidad de arrastrar y soltar (drag and drop). • Los cambios en los estados de las tareas deben guardarse automáticamente. • Los tableros deben reflejar con precisión el estado actual de todas las tareas de un proyecto.
Actividad # RF7	
Nombre de la actividad	Descripción
Temporizador pomodoro	<p>Este temporizador les ayudará a trabajar en intervalos de tiempo predefinidos, alternando entre periodos de trabajo y descanso. Los usuarios tendrán la opción de configurar los intervalos según sus necesidades, estableciendo tiempos específicos tanto para las fases de trabajo como para las pausas.</p> <p>Criterios de aceptación:</p> <ul style="list-style-type: none"> • El usuario puede iniciar, pausar y reiniciar el temporizador Pomodoro desde la interfaz de usuario. • El usuario puede configurar los intervalos de trabajo y descanso, con opciones predefinidas o personalizadas. • El sistema debe notificar al usuario cuando finalice el periodo de trabajo o descanso.
Actividad # RF8	
Nombre de la actividad	Descripción
Calendario	<p>Permitirá a los usuarios visualizar sus eventos y recordatorios a lo largo de un mes completo. Esta funcionalidad les permitirá agregar, editar y eliminar eventos y recordatorios, asegurando que puedan planificar y visualizar sus compromisos y actividades.</p> <p>Criterios de aceptación:</p> <ul style="list-style-type: none"> • El usuario puede visualizar un calendario mensual con eventos y recordatorios.

- | | |
|--|--|
| | <ul style="list-style-type: none"> ● El usuario puede agregar, editar y eliminar eventos o recordatorios. ● Los eventos modificados o eliminados deben reflejarse en el calendario inmediatamente. |
|--|--|

3.2.1. Requisitos no funcionales

- **Rendimiento:** Las operaciones CRUD deben ejecutarse con una latencia mínima.
- **Seguridad:** Autenticación mediante correo electrónico y protección de datos sensibles con cifrado.
- **Escalabilidad:** El sistema debe poder manejar un crecimiento en la base de usuarios sin impactar negativamente en el rendimiento. La capacidad inicial del sistema puede estar diseñada para soportar entre 100 a 500 usuarios simultáneos durante la fase de lanzamiento. Posteriormente, el sistema debe escalar de manera eficiente a medida que más usuarios se registren.
 - **Usuarios simultáneos:** En la fase inicial, el sistema debería ser capaz de manejar al menos 200 usuarios interactuando simultáneamente con una baja latencia.
 - **Usuarios registrados:** Se estima que la capacidad inicial debe soportar hasta 1,000 usuarios registrados con la posibilidad de expansión hasta 5,000 usuarios en los próximos dos años.

3.3. Requisitos de Rendimiento

El sistema debe poder manejar hasta 500 usuarios simultáneos sin que se vea afectada la velocidad de respuesta de las operaciones CRUD y el temporizador Pomodoro debe actualizarse en tiempo real sin retrasos visibles.

3.4. Restricciones de Diseño

El sistema debe estar diseñado para ser responsive, adaptándose a dispositivos móviles y de escritorio. Además, debe seguir los principios de diseño modular para facilitar futuras expansiones.

3.5. Atributos del Sistema

- **Fiabilidad:** El sistema debe tener una disponibilidad del 99.9%.
- **Mantenibilidad:** El código debe estar bien documentado y ser fácilmente extensible para nuevas funcionalidades.
- **Usabilidad:** El sistema debe ser fácil de usar, con una curva de aprendizaje mínima.

3.6. Otros Requisitos

3.6.1. Requisitos de almacenamiento

- **Estimación inicial:** Para 1,000 usuarios con un promedio de 5 proyectos, 50 tareas y 20 notas por usuario, se estima que el almacenamiento inicial será de aproximadamente 500 MB a 1 GB.
- **Escalabilidad:** Si el sistema crece a 5,000 usuarios, el almacenamiento requerido podría aumentar a 5 GB. El sistema debe estar preparado para escalar utilizando soluciones de almacenamiento en la nube que permitan un crecimiento sin interrupciones.

3.6.1.1. Estimación de almacenamiento por usuario

- **Datos de usuario:** ~1-2 KB por usuario.

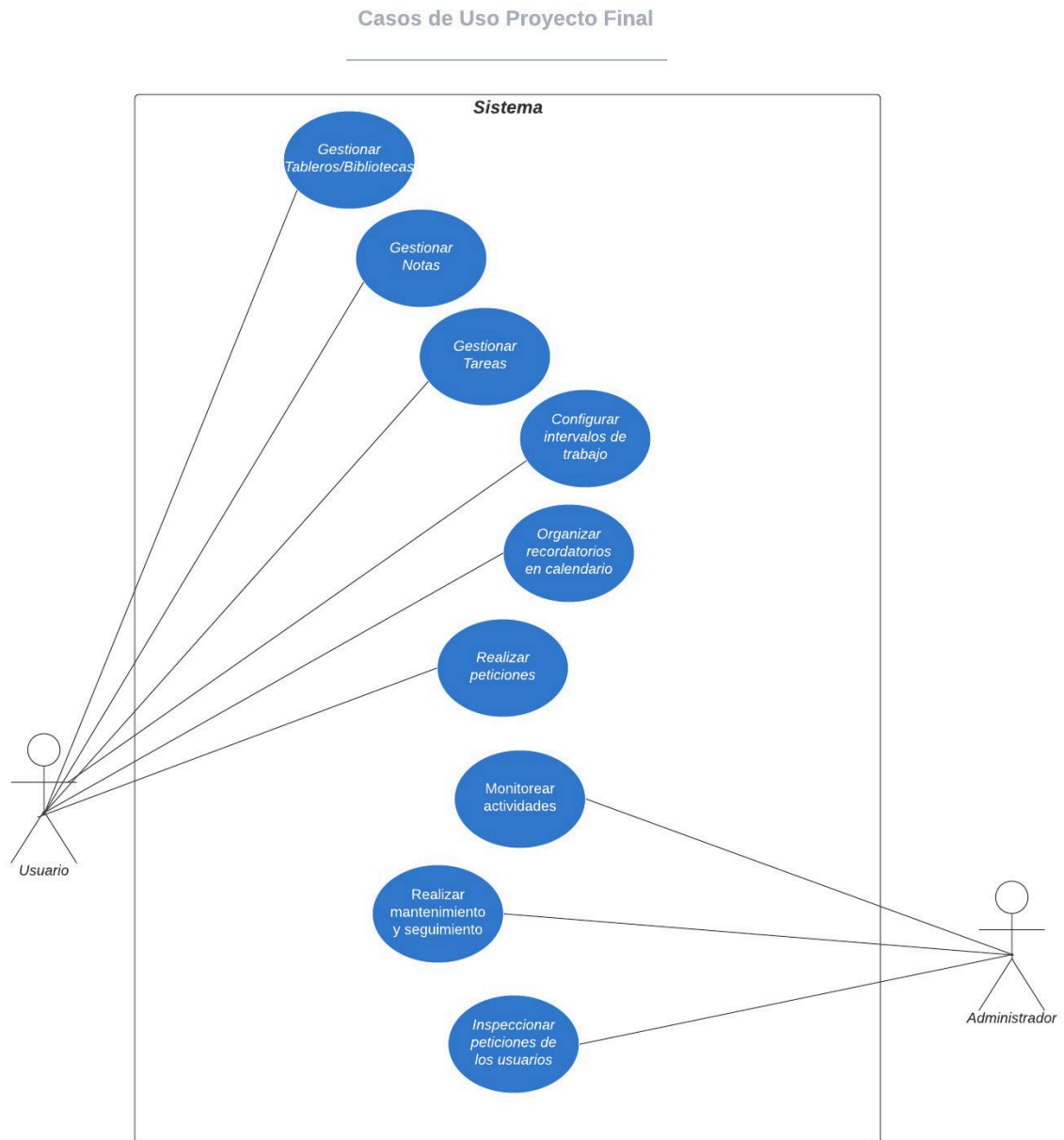
- **Proyectos:** ~10-50 KB por proyecto (dependiendo de la cantidad de tareas y notas).
- **Notas:** ~1-5 KB por nota.
- **Tareas:** ~0.5-2 KB por tarea.
- **Eventos del calendario:** ~0.5-2 KB por evento.

3.6.2. Requisitos de red

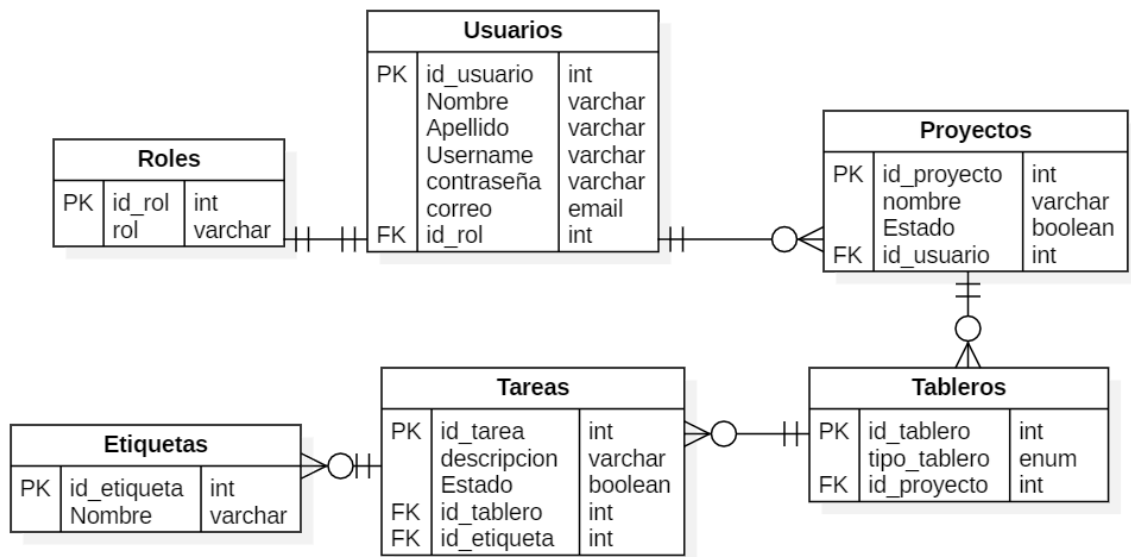
1. **Conectividad:** El sistema debe estar disponible a través de Internet, permitiendo el acceso remoto desde cualquier ubicación. Se requiere soporte para conexiones simultáneas de al menos 500 usuarios para asegurar la disponibilidad y el rendimiento en tiempos de alta demanda.
2. **Ancho de Banda:** Se recomienda un ancho de banda mínimo de 1 Mbps por usuario concurrente para garantizar una experiencia fluida durante la interacción con el sistema, especialmente al cargar o descargar datos (como archivos adjuntos en notas o tareas). El sistema debe ser capaz de manejar picos de tráfico donde el ancho de banda podría aumentar hasta 5 Mbps por usuario durante periodos de alta actividad.
3. **Latencia:** La latencia en la comunicación entre el cliente y el servidor no debe exceder los 100 ms para operaciones críticas (como inicio de sesión, creación de tareas y acceso a notas). El sistema debe estar optimizado para minimizar la latencia a través de la implementación de técnicas como la caché y la optimización de consultas a la base de datos.
4. **Protocolos de Red:** El sistema debe utilizar HTTPS para asegurar la transferencia de datos entre el cliente y el servidor, garantizando la privacidad y la integridad de la información.

5. Apéndices

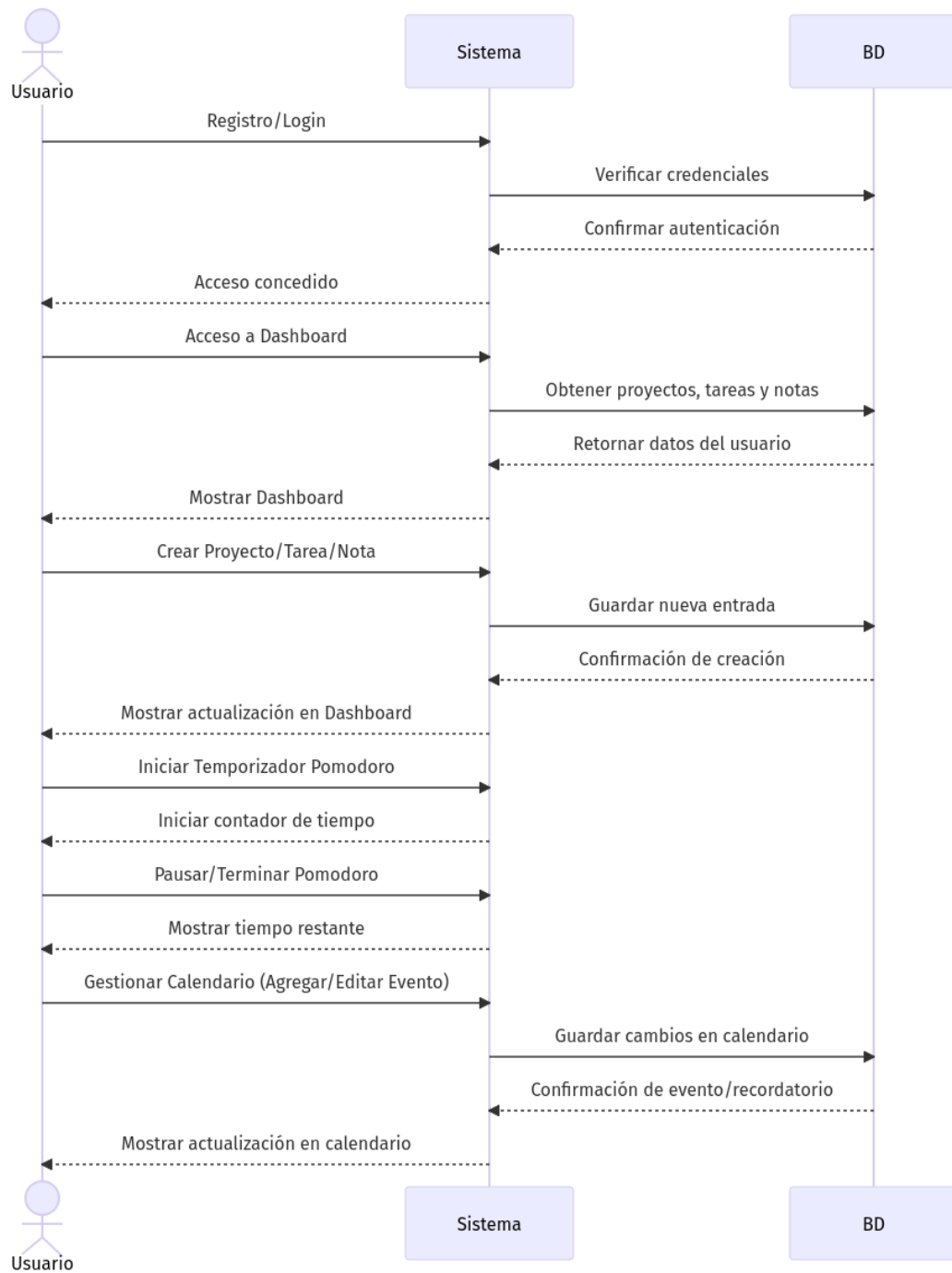
5.1. Diagrama de casos de uso

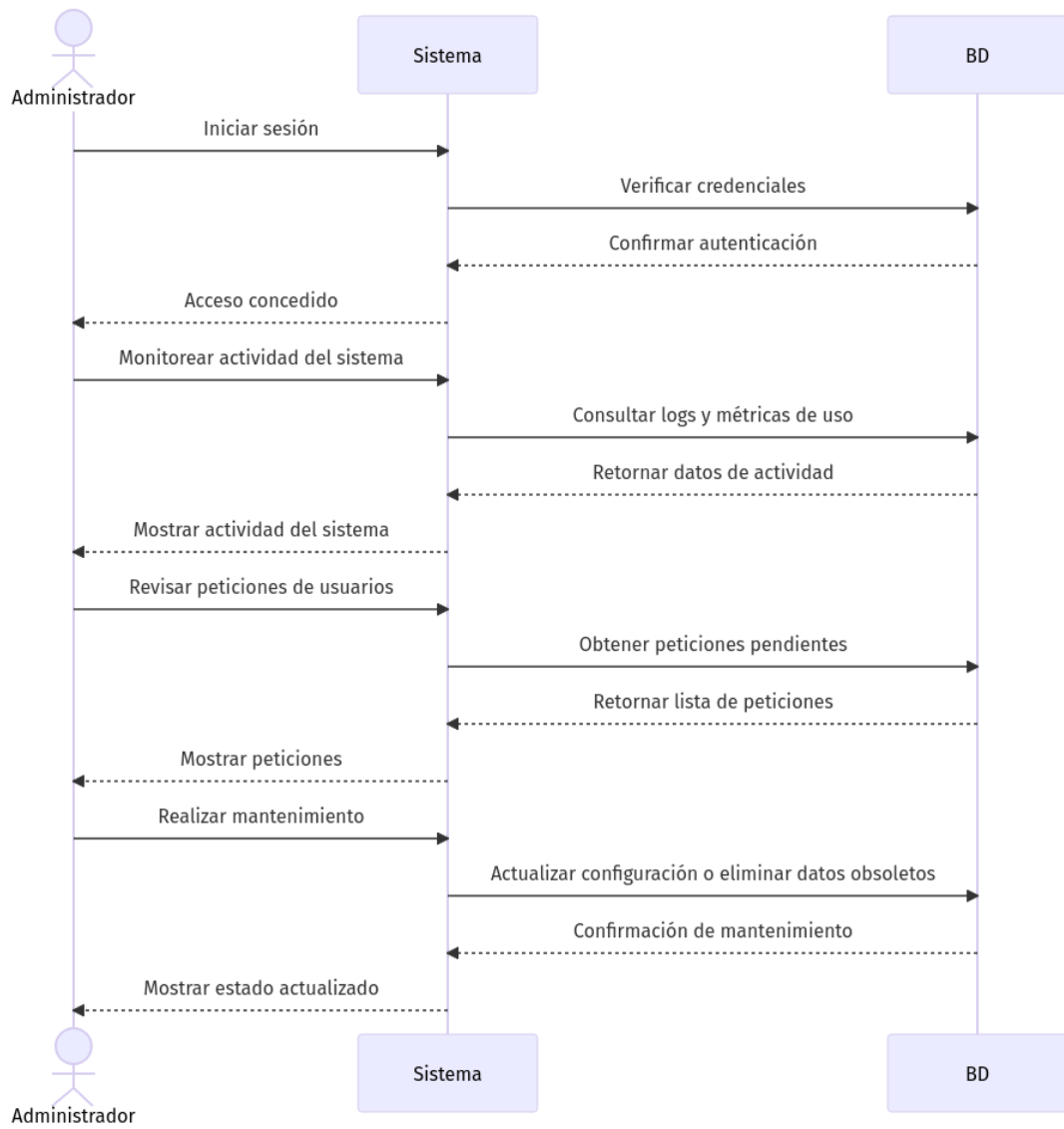


5.2. Modelo entidad-relación(MER)



5.3. Diagrama de secuencia





5.4. Mockups

5.5. Cronograma de actividades

N°	ACTIVIDADES	2024														
		JULIO	AGOSTO					SEPTIEMBRE				OCTUBRE				NOVIEMBRE
		SEMANA 1	SEMANA 2	SEMANA 3	SEMANA 4	SEMANA 5	SEMANA 6	SEMANA 7	SEMANA 8	SEMANA 9	SEMANA 10	SEMANA 11	SEMANA 12	SEMANA 13	SEMANA 14	
Planificación y Requerimientos																
1	Análisis de la problemática.	1														
2	Definición de requisitos funcionales y no funcionales.	1														
3	Identificación de usuarios clave y casos de uso.	2	1													
4	Creación del backlog del producto (tareas y funcionalidades).	2	1													
5	Asignación de roles y responsabilidades del equipo.	2	1													
Diseño																
6	Diseño de la arquitectura del sistema (base de datos, frontend, backend).	3	1													
7	Diseño del flujo de interacción de usuario.	3	1													
8	Modelado de diagramas UML y MER.	3	1													
9	Prototipado del tablero Kanban (visualización de tareas).	3	1													
10	Prototipado del temporizador Pomodoro (interfaz y funcionalidad).	3	1													
11	Diseño de la estructura de reportes de productividad.	3	1													
Desarrollo Backend																
12	Configuración del entorno de desarrollo.	4		1												
13	Integración de la base de datos con funcionalidades de backend.	4		1												
14	Implementación del temporizador Pomodoro (gestión de tiempo y descansos).	4		1												
15	Implementación de la lógica de gestión de tareas para el tablero Kanban.	4		1												
Desarrollo Frontend																
16	Configuración del entorno de desarrollo frontend (frameworks, herramientas).	4		1												
17	Desarrollo de la interfaz del temporizador Pomodoro (UI interactiva).	4		1												
18	Desarrollo del tablero Kanban (gestión visual de tareas).	4		1												
19	Diseño de la interfaz de usuario para la vista de reportes de productividad.	4		1												
Integración																
20	Conexión entre frontend y backend (API, flujos de datos).	5							1							
Pruebas de calidad																
21	Pruebas unitarias de cada funcionalidad (Pomodoro, Kanban, reportes).	5									1					
22	Pruebas de integración (validar la interacción entre módulos).	5									1					
23	Pruebas de usabilidad y rendimiento del sistema.	5									1					
24	Corrección de errores detectados durante las pruebas.	5									1					
Ajustes y mejoras																
25	Refinamiento de funcionalidades (Pomodoro y Kanban) basados en pruebas.	6											1			
26	Ajustes en la interfaz gráfica según feedback de pruebas.	6											1			
27	Optimización de código (rendimiento y eficiencia).	6											1			
Entrega final																
28	Entrega del producto final y cierre del proyecto.	6													1	

4.6. Descripción del equipo de desarrollo

- **Cristian Camilo Posada García (Líder de equipo y desarrollador Backend):**
Responsable de la planificación general, coordinación del equipo y comunicación con los interesados. Además, encargado de implementar la lógica del sistema, bases de datos y la funcionalidad del temporizador Pomodoro y tableros Kanban.
- **Michael Steven Naranjo Bautista (Beta Tester):** Probar la aplicación antes de su lanzamiento oficial para identificar problemas, errores y áreas de mejora y proporciona retroalimentación al equipo de desarrollo
- **Liseth Johanna Gomez Ariza (Diseñadora UX/UI y Desarrolladora Frontend):**
Diseña y construye la interfaz de usuario, incluyendo el tablero Kanban y la visualización de reportes. Se encarga de la experiencia del usuario y el diseño visual de la interfaz, asegurando que la aplicación sea intuitiva y atractiva.
- **Oscar Julian Montaña Alfonso (Analista QA):** Desarrollar y ejecutar pruebas funcionales, de rendimiento, de regresión y de integración.