



Optimización de Gestión de Proyectos

Daniel Santiago Palencia Sandoval

Michael Stiven Romero Soto

Juan Manuel Osorio López

Juan Camilo Salamanca Diaz

Nelson Andrés Ayala Álvarez

Facultad de Ingeniería, Ingeniería de Software III,

Universidad Libre

Docente: Castro Caicedo Rodrigo

10 de Marzo de 2025

OPTIMIZACION DE GESTION DE PROYECTOS

AUTORES

DANIEL SANTIAGO PALENCIA SANDOVAL

MICHAELL STIVEN ROMERO SOTO

JUAN MANUEL OSORIO LOPEZ

JUAN CAMILO SLAMANCA DIAZ

NELSON ANDRES AYALA ALVAREZ

DIRECTOR

CASTRO CAICEDO RODRIGO

UNIVERSIDAD LIBRE

FACULTAD DE INGENIERIA

CARRERA INGENIERIA DE SISTEMAS

Bogotá D.C

10 de Marzo de 2025

Tabla de contenido

Planteamiento del Proyecto	5
Planteamiento del Problema	5
Siemens	5
Problema	5
Fragmentación de herramientas.....	6
Dificultad para la colaboración.....	6
Limitaciones en el seguimiento y análisis	6
Complejidad en la gestión de recursos	6
Adaptación a Cambios.....	7
Impacto	7
Necesidad.....	7
Determinación de Requerimientos	8
Requerimientos Funcionales	9
Tabla 1	9
Tabla 2	10
Tabla 3	11
Tabla 4	12
Tabla 5	13
Tabla 6	13
Tabla 7	14
Tabla 8	15
Tabla 9	15
Tabla 10	16
Requerimientos no Funcionales	16
Tabla 11	16
Tabla 12	17
Tabla 13	17
Tabla 14	18
Tabla 15	18

Tabla 16	19
Tabla 17	20
Tabla 18	21
Tabla 19	22
Tabla 20	23
Tabla 21	23
Justificación del Planteamiento del Proyecto.....	24
Relevancia del problema	24
Necesidad de una solución integral.....	24
Beneficios potenciales.....	24
Optimización de recursos	24
Mejora de la colaboración	24
Seguimiento y análisis mejorados	25
Adaptación a cambios	25
Matriz de riegos y contingencias.....	26
Estudio de viabilidad.....	27
Económico	27
Tecnológica	27
Social u operativo	27
Ambiental o legal	27
Objetivos.....	28
Objetivo General.....	28
Objetivo Especifico	28
Delimitación y alcance.....	28
Delimitación.....	28
Ámbito organizacional	28
Sector de aplicación.....	29
Usuarios finales	29
Limitación tecnológica	29
Alcance	30
Análisis de requerimientos	30

Diseño de la plataforma	30
Desarrollo e implementación	30
Perfiles de usuario e Historia de usuario	31
Metodología de desarrollo SCRUM	34
Arquitectura.....	36
Tarjetas CRC	39
Diagramas de caso de uso	46
Diagrama 1.....	47
Diagrama 2.....	49
Diagrama 3.....	51
Diagrama 4.....	53
Diagrama 5.....	55
Diagrama 6.....	57
Diagrama 7.....	59
Diagrama 8.....	61
Diagrama 9.....	63
Diagrama 10.....	65
Diagrama 11.....	67
Diagrama 12.....	69
Diagrama 13.....	71
Diagrama 14.....	73
Diagrama 15.....	75
Diagrama 16.....	77
Diagrama 17.....	79
Diagrama 18.....	81
Diagrama 19.....	83
Diagrama 20.....	84
Diagrama de Distribución.....	84
Diagrama 21.....	84
Diagrama de Secuencia	86
Diagrama 22.....	86

Diagrama de Flujo	88
Diagrama 23.....	88
Diagrama de Actividades	89
Diagrama 24.....	89
Diagrama 25.....	89
Explicación del modelo de la BD, tablas y estructuras.....	90
Mapa de navegación	95

1. PLANTEAMIENTO DEL PROYECTO

1.1 PLANTEAMIENTO DEL PROBLEMA

Siemens:

Siemens, una empresa multinacional que opera en varios sectores, incluidos la electrificación, la automatización y la digitalización. Siemens es una de las empresas líderes a nivel mundial en tecnología e innovación, y tiene una presencia significativa en diferentes países, incluido Colombia.

En Colombia, Siemens ofrece una amplia gama de productos, soluciones y servicios en áreas como energía, industria, infraestructura y salud. Esto incluye sistemas de generación y distribución de energía, tecnologías de automatización industrial, soluciones de movilidad y transporte, sistemas de salud y equipamiento médico, entre otros.

Siemens contribuye al desarrollo tecnológico y económico del país al proporcionar soluciones innovadoras y sostenibles que ayudan a mejorar la eficiencia, la productividad y la calidad de vida de las personas. Además, la empresa está comprometida con la responsabilidad social corporativa y participa en iniciativas para el desarrollo comunitario y la protección del medioambiente.

Problema:

Los gerentes de proyectos se enfrentan a dificultades significativas en la ejecución eficiente y efectiva de proyectos debido a la falta de una plataforma digital integral y adecuada que satisfaga todas sus necesidades de gestión de proyectos. Esta falta de una herramienta adecuada resulta en:

Fragmentación de herramientas:

Los gerentes de proyectos se ven obligados a utilizar una variedad de herramientas y sistemas dispersos para diferentes aspectos de la gestión de proyectos, lo que conduce a la falta de cohesión y dificulta la integración de datos.

Dificultad para la colaboración:

La falta de una plataforma centralizada dificulta la colaboración entre equipos y partes interesadas, lo que puede dar lugar a retrasos, malentendidos y una comunicación ineficaz.

Limitaciones en el seguimiento y análisis:

La ausencia de capacidades integradas de seguimiento y análisis dificulta la evaluación del progreso del proyecto, la identificación de problemas y la toma de decisiones informadas.

Complejidad en la gestión de recursos:

La asignación y gestión eficiente de recursos, incluidos el personal, el tiempo y el presupuesto, se ve obstaculizada por la falta de herramientas que proporcionen visibilidad y control.

Adaptación a cambios:

La incapacidad para adaptarse rápidamente a cambios en los requisitos del proyecto, el alcance o las condiciones del mercado puede afectar negativamente la entrega oportuna y la calidad del proyecto.

Impacto:

Estos desafíos afectan directamente la eficiencia operativa de la organización, lo que resulta en retrasos en la entrega de proyectos, aumento de los costos, insatisfacción del cliente y pérdida de competitividad en el mercado.

Necesidad:

Existe una clara necesidad de una solución digital integral que aborde estos problemas y proporcione a los gerentes de proyectos las herramientas necesarias para planificar, ejecutar y supervisar proyectos de manera efectiva, fomentando la colaboración, la transparencia y la toma de decisiones basadas en datos.

1.1.1 Determinación de requerimientos:

Tabla 1.

REQUERIMIENTOS FUNCIONALES				
RF-1 Autenticación de usuario	RF-1	Prioridad Baja	Caso de uso	Inicio de sesión
Descripción	El sistema debe permitir a los usuarios autenticarse utilizando un nombre de usuario y contraseña válidos.			
Justificación	La autenticación de usuarios es un requisito fundamental para garantizar la seguridad y la integridad de un sistema.			
Origen	Usuario.			
Criterio de aceptación	El sistema verificará la autenticación utilizando una base de datos centralizada de usuarios.			
	Se mostrará un mensaje de error si las credenciales proporcionadas no son válidas.			
	Una vez autenticado, el usuario será redirigido a su página de inicio.			

Tabla 2.

REQUERIMIENTOS FUNCIONALES				
RF-2 Gestión de proyectos	RF-2	Prioridad Alta	Caso de uso	Edición y modificación
Descripción	El sistema debe permitir la creación, edición y eliminación de proyectos. Debe asignar roles y permisos a los miembros del equipo.			
Justificación	La gestión de proyectos es esencial para asegurar la organización y el éxito en la ejecución de tareas. Permitir la creación, edición y eliminación de proyectos, así como la asignación de roles y permisos, facilita la colaboración y la coordinación entre los miembros del equipo, mejorando la eficiencia y la productividad en el trabajo.			
Origen	Usuario.			
Criterio de aceptación	Los usuarios pueden crear nuevos proyectos y definir sus atributos como nombre, descripción y fechas de inicio y fin. Los roles de usuario (administrador, miembro del equipo) determinaran los permisos de acceso a funcionalidades específicas del proyecto.			

Tabla 3.

REQUERIMIENTOS FUNCIONALES				
RF-3 Seguimiento y control	RF-3	Prioridad Alta	Caso de uso	Seguimiento y control
Descripción	Monitorear el progreso del proyecto mediante paneles de control y tableros visuales			
Justificación	El seguimiento y control del progreso del proyecto son fundamentales para asegurar que este se desarrolle de acuerdo con los plazos y los objetivos establecidos.			
Origen	Equipo de desarrollo			
Criterio de aceptación	<p>Se registrará el tiempo dedicado a cada tarea por los miembros del equipo</p> <p>El sistema controlará el progreso de los proyectos</p> <p>Se generarán informes de avance, desempeño y cumplimiento de objetivos</p>			

Tabla 4.

REQUERIMIENTOS FUNCIONALES				
RF-4 Creación de proyectos	RF-4	Prioridad Alta	Caso de uso	Iniciación de proyectos
Descripción	El sistema debe permitir a los usuarios crear nuevos proyectos proporcionando detalles como nombre, descripción, fechas de inicio y finalización.			
Justificación	Facilita la gestión organizada de proyectos desde su inicio, asegurando que cada proyecto tenga información clara y completa.			
Origen	Usuario.			
Criterio de aceptación	El sistema permite la creación de proyectos con todos los campos requeridos completos.			

Tabla 5.

REQUERIMIENTOS FUNCIONALES				
RF-5 Edición de proyectos	RF-5	Prioridad Media	Caso de uso	Modificación de proyectos
Descripción	Los usuarios deben poder editar los detalles de un proyecto existente, como su nombre, descripción y fechas.			
Justificación	Permite flexibilidad en la gestión de proyectos, adaptándose a cambios en los requisitos o cronogramas.			
Origen	Usuario.			
Criterio de aceptación	El sistema permite la edición de proyectos existentes y guarda los cambios correctamente.			

Tabla 6.

REQUERIMIENTOS FUNCIONALES				
RF-6 Eliminación de proyectos	RF-6	Prioridad Media	Caso de uso	Borrado de proyectos
Descripción	El sistema debe permitir a los usuarios eliminar proyectos existentes.			
Justificación	Facilita la limpieza y gestión del sistema, asegurando que solo los proyectos relevantes permanezcan activos.			
Origen	Usuario.			
Criterio de aceptación	Los usuarios pueden eliminar proyectos, y el sistema confirma la eliminación con un mensaje de advertencia.			

Tabla 7.

REQUERIMIENTOS FUNCIONALES				
RF-7 Planificación de tareas	RF-7	Prioridad Alta	Caso de uso	Asignación de tareas
Descripción	El sistema debe permitir a los usuarios crear, asignar y priorizar tareas dentro de un proyecto.			
Justificación	Esencial para el seguimiento y la ejecución de tareas dentro de un proyecto, asegurando que los objetivos se cumplan de manera organizada.			
Origen	Usuario.			
Criterio de aceptación	Los usuarios pueden crear y asignar tareas con diferentes niveles de prioridad dentro de un proyecto.			

Tabla 8.

REQUERIMIENTOS FUNCIONALES				
RF-8 Seguimiento del progreso de tareas	RF-8	Prioridad Alta	Caso de uso	Monitoreo de tareas
Descripción	El sistema debe permitir a los usuarios actualizar el estado de las tareas (por ejemplo, "En progreso", "Completado").			
Justificación	Proporciona una visión clara del progreso de las tareas y ayuda a mantener el proyecto en el cronograma.			
Origen	Usuario.			
Criterio de aceptación	Los usuarios pueden cambiar el estado de las tareas y estos cambios se reflejan en el sistema en tiempo real.			

Tabla 9.

REQUERIMIENTOS FUNCIONALES				
RF-9 Notificaciones de tareas proyectos	RF-9	Prioridad Media	Caso de uso	Alertas de proyectos
Descripción	El sistema debe enviar notificaciones a los usuarios sobre tareas pendientes, plazos próximos y actualizaciones de proyectos.			
Justificación	Mantiene a los usuarios informados y ayuda a evitar retrasos en la entrega de tareas y proyectos.			
Origen	Usuario.			
Criterio de aceptación	Los usuarios reciben notificaciones automáticas según configuraciones predefinidas.			

Tabla 10.

REQUERIMIENTOS FUNCIONALES				
RF-10 Historial de actividades	RF-10	Prioridad Media	Caso de uso	Auditoría de Operaciones y Modificaciones
Descripción	El sistema debe mantener un registro del historial de actividades para cada proyecto, incluyendo cambios en tareas y recursos.			
Justificación	Proporciona un seguimiento de cambios y acciones, mejorando la transparencia y responsabilidad dentro del equipo.			
Origen	Usuario.			
Criterio de aceptación	El sistema guarda y muestra un historial detallado de todas las actividades relacionadas con cada proyecto.			

Tabla 11.

REQUERIMIENTOS FUNCIONALES				
RNF-11 Generación Automatizada de Informes	RF-11	Prioridad Alta	Caso de uso	Automatización de Informes Progresivos
Descripción	El sistema debe permitir la generación automática de informes detallados sobre el progreso de los proyectos, que pueden ser exportados en formatos como PDF o Excel.			
Justificación	Facilita la presentación de resultados y el análisis de datos de proyectos para la toma de decisiones.			
Origen	Requisito del cliente			
Criterio de aceptación	Los informes se generan correctamente y son exportables en los formatos especificados.			

Tabla 12.

REQUERIMIENTOS FUNCIONALES				
RF-12 Gestión de Permisos y Roles de Usuario	RF-12	Prioridad Alta	Caso de uso	Administración de Roles y Accesos
Descripción	El sistema debe permitir la asignación y gestión de permisos y roles a los usuarios, garantizando que solo accedan a la información y funcionalidades que les correspondan.			
Justificación	Asegura la seguridad y el control de acceso dentro del sistema.			
Origen	Requisito de seguridad			
Criterio de aceptación	Los usuarios solo pueden acceder a las secciones del sistema asignadas según su rol.			

Tabla 13.

REQUERIMIENTOS FUNCIONALES				
RF-13 Integración con Calendarios Externos	RF-13	Prioridad Media	Caso de uso	Sincronización con Calendarios Externos
Descripción	El sistema debe integrarse con calendarios externos como Google Calendar o Outlook, permitiendo la sincronización de fechas y plazos de proyectos.			
Justificación	Facilita la coordinación de actividades y la gestión del tiempo por parte de los usuarios.			
Origen	Requisito de interoperabilidad			
Criterio de aceptación	Los eventos del sistema se sincronizan correctamente con los calendarios externos.			

Tabla 14.

REQUERIMIENTOS FUNCIONALES				
RF-14 Seguimiento de Tareas en Múltiples Proyectos	RF-14	Prioridad Alta	Caso de uso	Panel Unificado de Tareas Múltiples
Descripción	El sistema debe permitir a los usuarios visualizar y hacer seguimiento de tareas distribuidas en varios proyectos desde un solo panel de control.			
Justificación	Mejora la eficiencia y organización al manejar múltiples proyectos simultáneamente.			
Origen	Requisito de gestión de proyectos			
Criterio de aceptación	Las tareas de diferentes proyectos se muestran correctamente en el panel de control unificado.			

Tabla 15.

REQUERIMIENTOS FUNCIONALES				
RF-15 Automatización de Procesos Recurrentes	RF-15	Prioridad Media	Caso de uso	Automatización de Tareas Repetitivas
Descripción	El sistema debe permitir la automatización de tareas recurrentes, como recordatorios o actualizaciones de estado, para reducir la carga manual sobre los usuarios.			
Justificación	Aumenta la eficiencia operativa al minimizar el trabajo manual repetitivo.			
Origen	Requisito de optimización operativa			
Criterio de aceptación	Las tareas recurrentes se ejecutan automáticamente según las configuraciones definidas por el usuario.			

Tabla 16.

REQUERIMIENTOS NO FUNCIONALES				
RNF-16 Gestión de recursos	RNF-16	Prioridad Alta	Caso de uso	Recursos
Descripción	La interfaz mostrara la asignación de recursos humanos, financieros y materiales a las tareas y proyectos			
Justificación	Las integraciones con herramientas externas permiten una mayor flexibilidad y funcionalidad en el sistema al conectarlo con otras aplicaciones ampliamente utilizadas. Sincronizar con calendarios como Google Calendar u Outlook facilita la gestión de plazos y el seguimiento de actividades, mejorando la eficiencia y la coordinación en la ejecución de proyectos.			
Origen	Usuario			
Criterio de aceptación	Administrar calendarios de disponibilidad de los miembros del equipo Optimizar la utilización de recursos y evitar sobrecargas			

Tabla 17.

REQUERIMIENTOS NO FUNCIONALES				
RNF-17 Usabilidad	RNF-17	Prioridad Media	Caso de uso	Usabilidad de interfaz
Descripción	La interfaz debe ser intuitiva y fácil de usar para usuarios de diferentes niveles de habilidad.			
Justificación	Una interfaz intuitiva y fácil de usar es esencial para garantizar la adopción y el éxito del sistema, especialmente para usuarios con diferentes niveles de habilidad y experiencia. La usabilidad mejora la eficiencia del usuario al permitir que realice tareas sin dificultades ni necesidad de formación adicional, lo que aumenta la productividad y la satisfacción del usuario.			
Origen	Usuario			
Criterio de aceptación	<p>Los usuarios deben poder realizar las tareas principales del sistema sin necesidad de capacitación adicional.</p> <p>La interfaz debe seguir principios de diseño centrados en el usuario.</p>			

Tabla 18.

REQUERIMIENTOS NO FUNCIONALES				
RNF-18 Seguridad	RNF-18	Prioridad Alta	Caso de uso	Privacidad de los usuarios
Descripción	El sistema debe implementar medidas de seguridad robustas para proteger los datos del proyecto y la privacidad de los usuarios.			
Justificación	La seguridad de los datos del proyecto y la privacidad de los usuarios son aspectos críticos que deben ser protegidos de posibles amenazas y ataques cibernéticos. Implementar medidas robustas de seguridad, como el cifrado de datos y los controles de acceso, garantiza la confidencialidad, integridad y disponibilidad de la información, construyendo la confianza del usuario y mitigando riesgos de exposición de datos sensibles.			
Origen	Usuario			
Criterio de aceptación	<p>Se deben aplicar controles de acceso adecuados para garantizar que solo usuarios autorizados puedan acceder a la información sensible.</p> <p>El sistema debe cifrar la información confidencial almacenada en la base de datos.</p>			

Tabla 19.

REQUERIMIENTOS NO FUNCIONALES				
RNF-19 Escalabilidad	RNF-19	Prioridad Alta	Caso de uso	Rendimiento
Descripción	El sistema debe ser capaz de manejar un gran número de proyectos y usuarios simultáneamente sin degradación del rendimiento.			
Justificación	La capacidad del sistema para crecer y adaptarse a medida que aumenta el número de proyectos y usuarios es fundamental para garantizar su eficacia a largo plazo. Una arquitectura escalable permite mantener el rendimiento del sistema incluso en situaciones de alta demanda, asegurando una experiencia consistente y sin interrupciones para los usuarios.			
Origen	Usuario			
Criterio de aceptación	<p>El sistema debe escalar de manera lineal con el aumento en el número de proyectos y usuarios.</p> <p>Se deben realizar pruebas de carga para verificar la capacidad del sistema para manejar situaciones de alta demanda</p>			

Tabla 20.

REQUERIMIENTOS NO FUNCIONALES				
RNF-20 Redundancia y Resiliencia del Sistema	RNF-21	Prioridad Alta	Caso de uso	Mecanismos de continuidad operativa
Descripción	El sistema debe contar con mecanismos de redundancia que aseguren su funcionamiento continuo incluso en caso de fallos de componentes.			
Justificación	Asegura la alta disponibilidad y la continuidad del servicio en situaciones adversas.			
Origen	Requisito de alta disponibilidad.			
Criterio de aceptación	El sistema sigue operativo durante simulaciones de fallos en componentes críticos.			

Tabla 21.

REQUERIMIENTOS NO FUNCIONALES				
RNF-21 Rendimiento	RNF-20	Prioridad Alta	Caso de uso	Respuesta instantánea
Descripción	Las operaciones del sistema, como la carga de proyectos y tareas, deben completarse en menos de 2 segundos en promedio.			
Justificación	Proporciona una experiencia de usuario fluida, evitando la frustración y aumentando la productividad.			
Origen	Requisito de rendimiento.			
Criterio de aceptación	El tiempo de respuesta para las operaciones clave es inferior a 2 segundos en pruebas de carga.			

Tabla 22.

REQUERIMIENTOS NO FUNCIONALES				
RNF-22 Compatibilidad Multiplataforma	RNF-22	Prioridad Alta	Caso de uso	Soporte para Múltiples Dispositivos
Descripción	La plataforma debe ser accesible desde cualquier dispositivo con conexión a internet, incluyendo computadoras, tabletas y teléfonos móviles.			
Justificación	Permite que los usuarios accedan a la herramienta en cualquier momento y desde cualquier lugar, aumentando la flexibilidad y productividad.			
Origen	Necesidad de los usuarios de trabajar desde diferentes dispositivos.			
Criterio de aceptación	Se realizarán pruebas en navegadores modernos (Chrome, Firefox, Edge, Safari) y dispositivos móviles para verificar la correcta visualización y funcionalidad.			

Tabla 23.

REQUERIMIENTOS NO FUNCIONALES				
RNF-23 Disponibilidad del Sistema	RNF-23	Prioridad Alta	Caso de uso	Alta Disponibilidad y Continuidad del Servicio
Descripción	Se debe garantizar un tiempo de actividad del 99.9% al año, reduciendo fallos y tiempos de inactividad.			
Justificación	Evita interrupciones en la gestión de proyectos y asegura acceso permanente a la plataforma.			
Origen	Requerimiento de estabilidad del sistema.			
Criterio de aceptación	Implementación de infraestructura redundante y monitoreo de uptime.			

Tabla 24.

REQUERIMIENTOS NO FUNCIONALES				
RNF-24 Mantenimiento y Actualizaciones Automatizadas	RNF-24	Prioridad Media	Caso de uso	Gestión de Actualizaciones y Soporte Técnico
Descripción	El sistema debe permitir actualizaciones y mantenimiento sin afectar la experiencia del usuario ni interrumpir el servicio.			
Justificación	Asegura la evolución del sistema sin comprometer su funcionamiento en producción.			
Origen	Buenas prácticas de desarrollo de software.			
Criterio de aceptación	Se validará que las actualizaciones se implementen sin afectar las operaciones activas.			

Tabla 25.

REQUERIMIENTOS NO FUNCIONALES				
RNF-25 Registro y Auditoría de Actividades	RNF-25	Prioridad Alta	Caso de uso	Trazabilidad y Registro de Eventos del Sistema
Descripción	Se deben registrar las acciones críticas de los usuarios, como la creación, modificación y eliminación de proyectos y tareas.			
Justificación	Proporciona un historial detallado para auditorías y control de cambios.			
Origen	Necesidad de seguimiento y transparencia.			
Criterio de aceptación	Implementación de un módulo de auditoría accesible para administradores.			

1.2. JUSTIFICACION DEL PLANTEAMIENTO DE PROYECTO

Relevancia del Problema:

La gestión eficaz de proyectos es fundamental para el éxito de las organizaciones en un entorno empresarial cada vez más competitivo y dinámico. Los desafíos identificados en el planteamiento del problema son comunes en muchas empresas y tienen un impacto directo en la eficiencia operativa y la capacidad de cumplir con los objetivos estratégicos.

Necesidad de una Solución Integral:

La fragmentación de herramientas, la dificultad para la colaboración y el seguimiento, así como la complejidad en la gestión de recursos, destacan la urgencia de una solución integral. Una plataforma digital unificada proporcionará a los gerentes de proyectos las herramientas necesarias para superar estos desafíos y mejorar la eficacia de sus operaciones.

Beneficios Potenciales:

Implementar una plataforma digital integral para la gestión de proyectos ofrecerá una serie de beneficios tangibles:

Optimización de recursos: Facilitará la asignación eficiente de recursos y la gestión del tiempo y el presupuesto, lo que ayudará a reducir los costos y mejorar la rentabilidad.

Mejora de la colaboración: Permitirá una comunicación más efectiva y una colaboración fluida entre equipos y partes interesadas, lo que impulsará la productividad y la cohesión del equipo.

Seguimiento y análisis mejorados: Ofrecerá capacidades avanzadas de seguimiento y análisis, lo que permitirá a los gerentes de proyectos evaluar el progreso, identificar áreas de mejora y tomar decisiones informadas de manera oportuna.

Adaptación a cambios: Facilitará la adaptación rápida y eficaz a cambios en los requisitos del proyecto, el alcance o las condiciones del mercado, lo que garantizará una mayor flexibilidad y agilidad en la ejecución de proyectos.

Al abordar estos desafíos, la solución propuesta contribuirá directamente a la consecución de los objetivos estratégicos de la organización, incluida la mejora de la competitividad, la satisfacción del cliente y la maximización de la eficiencia operativa.

MATRIZ DE RIESGOS Y CONTINGENCIAS

RIESGO	DESCRIPCION	CONTINGENCIA
Riesgo tecnológico	Problemas de compatibilidad con sistemas existentes de Siemens Colombia.	Plan de migración gradual para implementar la plataforma de forma incremental.
Riesgo de Cambio Organizacional	Resistencia al cambio por parte de los usuarios	Desarrollar un plan de comunicación y capacitación efectiva para promover la aceptación del cambio.
Riesgo de Seguridad de Datos	Vulnerabilidades en la plataforma que comprometan la seguridad de los datos del proyecto.	Implementar medidas de seguridad adicionales, como actualizaciones regulares y auditorías de seguridad.
Riesgo de Tiempo	Retrasos en el desarrollo e implementación debido a problemas técnicos o de recursos.	Asignar recursos adicionales y revisar periódicamente los hitos del proyecto para mantenerlo en curso.
Riesgo de Costo	Aumento de costos debido a cambios en requisitos o estimaciones incorrectas.	Establecer una reserva de contingencia financiera para cubrir posibles aumentos de costos.

ESTUDIO DE VIABILIDAD

Económico:

Impacto Positivo: Reducción de costos operativos al mejorar la eficiencia en la gestión de proyectos.

Impacto Negativo: Costos iniciales de desarrollo e implementación de la plataforma.

Tecnológica:

Impacto Positivo: Mejora en la tecnología de gestión de proyectos, aumentando la productividad y la competitividad.

Impacto Negativo: Posibles problemas de compatibilidad con sistemas existentes, que podrían requerir inversiones adicionales en integración.

Social u Operativo:

Impacto Positivo: Mejora en la colaboración y comunicación entre equipos, lo que puede conducir a un aumento en la satisfacción y el compromiso de los empleados.

Impacto Negativo: Resistencia al cambio por parte de los usuarios finales, lo que podría afectar la adopción y el éxito de la plataforma.

Ambiental o Legal:

Impacto Positivo: Reducción del uso de papel y recursos físicos mediante la transición a una plataforma digital.

Impacto negativo: Necesidad de cumplir con regulaciones de protección de datos y seguridad cibernética, con posibles consecuencias legales en caso de incumplimiento.

1.3. OBJETIVOS

1.3.1 Objetivo General

Implementar un plan de mejoramiento en la gestión de proyectos mediante la optimización de procesos, la adopción de herramientas tecnológicas y metodologías ágiles, con el fin de mejorar la planificación, ejecución y control de proyectos, garantizando una mayor eficiencia, reducción de riesgos y cumplimiento de objetivos estratégicos.

1.3.2 Objetivos Específicos

- Diagnosticar las deficiencias actuales en la gestión de proyectos para identificar oportunidades de mejora y establecer estrategias que optimicen la planificación y ejecución de tareas.
- Estandarizar metodologías y procesos aplicando buenas prácticas en la administración de proyectos, como Scrum, PRINCE2 o PMBOK, asegurando mayor control y seguimiento de las actividades.
- Fortalecer la comunicación y colaboración entre equipos de trabajo mediante la implementación de plataformas digitales que centralicen la información y permitan el monitoreo en tiempo real.
- Automatizar el seguimiento y control de proyectos a través de herramientas que faciliten la asignación de recursos, la identificación de riesgos y la generación de reportes estratégicos para la toma de decisiones.
- Capacitar a los involucrados en la gestión de proyectos mediante formación en metodologías ágiles, herramientas de planificación y mejores prácticas, promoviendo un cambio organizacional efectivo.

1.4. PLAN DE PRUEBAS

1.4.1 Documentos relacionados

NOMBRE	DESCRIPCION	URL
Guía del PMBOK (Project Management Body of Knowledge)	Este documento es una referencia fundamental en la gestión de proyectos. Proporciona buenas prácticas, estándares y metodologías aplicadas en diversas industrias, incluyendo el desarrollo de software.	https://www.pmi.org/pmbok-guide-standards
ISO/IEC 12207: Procesos del Ciclo de Vida del Software	Es un estándar internacional que define los procesos que deben seguirse en la gestión y desarrollo de software, garantizando calidad y cumplimiento normativo.	https://www.iso.org/standard/63712.html
IEEE Std 830-1998 – Requisitos de Software	Este estándar IEEE establece guías para la documentación de requisitos de software, ayudando a definir especificaciones claras y completas en proyectos.	https://standards.ieee.org/standard/830-1998.html

1.4.2 Alcance de las pruebas

MODULOS DEL SISTEMA A SER PROBADOS	
Objetivos	<ul style="list-style-type: none"> • Verificar que el sistema permita la creación, edición y eliminación de proyectos sin errores. • Garantizar que los usuarios puedan asignar roles y permisos correctamente. • Evaluar la integración del sistema con herramientas externas como calendarios y notificaciones.
Detalle del orden de ejecución de los módulos	<ul style="list-style-type: none"> • Autenticación de usuario – Validación del acceso según permisos. • Creación de proyectos – Prueba de ingreso de datos y validaciones. • Asignación de roles y permisos – Verificación del acceso adecuado para cada usuario. • Seguimiento y control – Evaluación de paneles, reportes y alertas. • Eliminación de proyectos – Comprobación de restricciones y confirmaciones.
Responsabilidad de la prueba	<p>Equipo de QA: Realizar pruebas funcionales y de usabilidad.</p> <p>Desarrolladores: Corregir errores detectados y optimizar el rendimiento del módulo.</p> <p>Usuarios finales (prueba piloto): Validar la facilidad de uso y detectar posibles mejoras.</p>

1.4.3 Requerimiento de prueba Excluidos

NOMBRE	DESCRIPCION	TIPO	NIVEL DE CRITICIDAD
Integración con sistemas heredados	No se realizarán pruebas de compatibilidad con sistemas antiguos o que no formen parte del ecosistema actual de la empresa.	Integración	Bajo
Pruebas de rendimiento extremo	No se contemplarán pruebas de carga con escenarios extremos que excedan la capacidad máxima esperada del sistema.	Rendimiento	Medio

1.4.4 Casos de prueba incluidos

# CASOS DISPONIBLES	# ESTIMADO DE CASOS NUEVOS	TIPO	MODULO	TOTAL, DE CASOS

1.5. DELIMITACION Y ALCANCE**Delimitación:**

Ámbito Organizacional: El proyecto se enfocará exclusivamente en el diseño de la plataforma de gestión de proyectos para Siemens Colombia, excluyendo otras filiales o divisiones de la empresa.

Sector de Aplicación: La plataforma estará diseñada específicamente para cubrir las necesidades de gestión de proyectos en los sectores de energía, industria, infraestructura y salud en los que opera Siemens Colombia.

Usuarios Finales: La plataforma estará dirigida principalmente a gerentes de proyectos, líderes de equipos, miembros del equipo y partes interesadas involucradas en la ejecución de proyectos dentro de Siemens Colombia.

Limitación Tecnológica: El proyecto se basará en tecnologías y recursos disponibles actualmente, evitando el desarrollo de funcionalidades o características que requieran nuevas inversiones significativas o tecnologías no probadas.

Alcance:

Análisis de Requerimientos: Este proceso comprenderá la identificación y documentación de las necesidades específicas de gestión de proyectos de Siemens, incluyendo entrevistas con los stakeholders y la recopilación de información relevante.

Diseño de la Plataforma: Se desarrollará un diseño detallado de la plataforma, definiendo su arquitectura, estructura de datos, flujos de trabajo y experiencia de usuario, en base a los requerimientos identificados en la etapa anterior.

Desarrollo e Implementación: El diseño de la plataforma se llevará a cabo, utilizando metodologías ágiles para garantizar la entrega iterativa y la adaptación a los cambios en los requisitos durante el proceso de desarrollo. La implementación se realizará en colaboración con nuestro contacto, asegurando la integración adecuada con los sistemas existentes.

1.6 PERFILES DE USUARIO E HISTORIAS DE USUARIO

1.6.1 Administrador del Sistema:

Descripción: El Administrador del Sistema es responsable de la configuración, mantenimiento y administración general de la plataforma. Tiene acceso completo a todas las funcionalidades del sistema y es el encargado de gestionar los permisos y roles de los demás usuarios.

Historia de Usuario: “Como Administrador del Sistema, quiero poder crear y gestionar cuentas de usuario para que cada persona tenga el acceso adecuado a las funcionalidades según su rol.”

Funciones Principales:

- Gestión de usuarios: Crear, modificar y eliminar cuentas de usuario.
- Asignación de roles y permisos: Definir qué funcionalidades están disponibles para cada tipo de usuario.
- Supervisión del sistema: Monitorear la actividad del sistema y realizar auditorías de seguridad.
- Configuración del sistema: Realizar ajustes en la configuración del sistema según sea necesario.

1.6.2 Gerente de proyecto:

Descripción: El Gerente de Proyecto es responsable de la planificación, ejecución y cierre de proyectos. Este perfil tiene acceso a todas las funcionalidades relacionadas con la gestión de proyectos, como la creación, edición y seguimiento de proyectos, tareas y recursos.

Historia de Usuario: “Como Gerente de Proyecto, quiero poder crear y asignar tareas dentro de un proyecto para asegurarme de que todas las actividades necesarias sean completadas dentro de los plazos establecidos.”

Funciones Principales:

- 1.6.2.1 Creación y gestión de proyectos: Definir los objetivos del proyecto, asignar recursos, y establecer plazos.
- 1.6.2.2 Seguimiento de tareas: Monitorizar el progreso de las tareas y realizar ajustes según sea necesario.
- 1.6.2.3 Generación de informes: Crear informes de progreso y rendimiento del proyecto para compartir con stakeholders. auditorías de seguridad.
- 1.6.2.4 Gestión de riesgos: Identificar y mitigar riesgos potenciales que puedan afectar el éxito del proyecto.

1.6.3 Miembro del Equipo:

Descripción: Los Miembros del Equipo son los encargados de ejecutar las tareas asignadas dentro de los proyectos. Este perfil tiene acceso limitado, principalmente enfocado en la visualización y actualización de sus tareas, así como en la colaboración con otros miembros del equipo.

Historia de Usuario: “Como Miembro del Equipo, quiero poder ver las tareas que me han sido asignadas y actualizar su estado para que el gerente pueda monitorear el progreso.”

Funciones Principales:

- 1.6.3.1 Ejecución de tareas: Completar las tareas asignadas según los plazos establecidos.
- 1.6.3.2 Actualización de estado: Reportar el progreso de sus tareas y actualizar el estado en la plataforma.
- 1.6.3.3 Colaboración: Comunicarse y colaborar con otros miembros del equipo a través de la plataforma.
- 1.6.3.4 Participación en reuniones: Asistir a reuniones diarias y de revisión para discutir el progreso y posibles impedimentos.

1.6.4 Cliente / Stakeholder:

Descripción: Los Clientes o Stakeholders tienen acceso a la plataforma para revisar el progreso de los proyectos en los que están involucrados. Su acceso está limitado a la visualización de reportes y a la retroalimentación sobre los entregables.

Historia de Usuario: “Como Cliente, quiero poder revisar el progreso del proyecto a través de informes detallados para asegurarme de que todo va según lo planeado.”

Funciones Principales:

- 1.6.4.1 Revisión de progreso: Visualizar informes y estados de los proyectos.
- 1.6.4.2 Retroalimentación: Proporcionar feedback sobre los entregables y el progreso del proyecto.
- 1.6.4.3 Participación en reuniones: Asistir a reuniones de revisión para discutir el avance del proyecto y cualquier cambio necesario.

1.7 METODOLOGIA DE DESARROLLO SCRUM

El equipo de Scrum estará compuesto por cinco miembros, cada uno con roles y responsabilidades específicas dentro del proyecto:

1.7.2 Nelson Ayala - jefe de Proyecto / Scrum Master:

1.7.2.1 Responsabilidades:

1. Facilitar ceremonias de Scrum.
2. Eliminar impedimentos que afectan al equipo.
3. Asegurar el cumplimiento de las prácticas ágiles de Scrum.
4. Fomentar la mejora continua del proceso.
5. Aceptar o rechazar los incrementos del producto.

1.7.2.2 Requerimiento Asignado:

ID: RF-2, RF-12, RNF-16, RNF-17.

1.7.3 Daniel Palencia – Subjefe de proyecto / Product Owner

1.7.3.1 Responsabilidades:

1. Gestionar y priorizar el Product Backlog.
2. Definir la visión del producto
3. Asegurar la entrega de valor máximo al negocio.

1.7.3.2 Requerimiento Asignado:

ID: RF-3, RF-4, RF-7, RF-9, RF-10, RNF-18, RNF-19.

1.7.4 Juan Osorio - Desarrollador Principal

1.7.4.1 Responsabilidades:

1. Desarrollar y programar las funcionalidades clave del proyecto
2. Colaborar en la planificación y estimación de tareas.
3. Asegurar que el código cumple con los estándares de calidad.

1.7.4.2 Requerimiento Asignado:

ID: RF-1, RF-11 RF-13, RF-14, RF-15, RNF-20.

1.7.5 Michael Romero – Analista de Requerimientos

1.7.5.1 Responsabilidades:

1. Soporte en el desarrollo y pruebas de las funcionalidades.

2. Implementar funcionalidades de menor complejidad.
3. Participar activamente en las revisiones de código y pruebas.

1.7.5.2 Requerimiento Asignado:

ID: RF-6, RF-8, RNF-16, RNF-17.

1.7.6 Camilo Salamanca – Desarrollador de Pruebas

1.7.6.1 Responsabilidades:

1. Validar la calidad del software a través de pruebas exhaustivas.
2. Crear y ejecutar casos de prueba.
3. Reportar y hacer seguimiento de defectos.

1.7.6.2 Requerimiento Asignado:

ID: RF-8, RF-9, RF-10, RNF-18, RNF-19.

1.8 ARQUITECTURA

Capa 1: Presentación (Clientes Web y Desktop)

Descripción: Esta capa interactúa directamente con los usuarios a través de interfaces visuales. Para cada interfaz, tenemos dos divisiones:

- Web: Se accede desde navegadores (Google Chrome, Firefox, etc.), donde los gerentes de proyectos y usuarios interactúan con la plataforma.
- Desktop (App Escritorio): Una aplicación de escritorio que replica las funcionalidades del frontend web, pero con acceso nativo a recursos del sistema operativo (archivos, notificaciones locales, etc.).

Tecnologías Sugeridas:

- Web: React.js, Angular, o Vue.js para el frontend web, con comunicación a la API mediante Fetch o Axios.
- App Escritorio: Electron.js o Tauri para crear aplicaciones de escritorio que también se conecten a la API.
- Comunicación: Ambas interfaces (web y desktop) interactúan con el backend mediante peticiones RESTful API.

Requisitos implementados en esta capa:

- REQ001: Autenticación de usuario
- REQ002: Gestión de proyectos
- REQ008: Planificación de tareas

- REQ009: Notificaciones de tareas y proyectos
- REQ012: Gestión de permisos y roles de usuario
- REQ014: Seguimiento de tareas en múltiples proyectos

Capa 2: Lógica de Negocio (Servidor de Aplicaciones)

Descripción: Es el núcleo donde se implementan las reglas de negocio y se toman decisiones. Aquí se gestiona la lógica relacionada con la creación y modificación de proyectos, notificaciones, seguimiento de tareas y generación de informes.

- El servidor puede estar basado en Node.js/Express, Java/Spring, o Python/Django. Este servidor también expone una API REST para que las capas de presentación (web y escritorio) interactúen con él.

Divisiones:

- Backend Web API: A través de una API REST, se proporcionan los servicios para las aplicaciones cliente (web y escritorio).
- Servicios de Escritorio: Adicionalmente, para la app de escritorio, se pueden añadir funciones nativas como el acceso a sistemas de archivos locales.

Requisitos implementados en esta capa:

- REQ002: Gestión de proyectos
- REQ003: Seguimiento y control
- REQ011: Generación automatizada de informes

- REQ015: Automatización de procesos recurrentes

Capa 3: Acceso a Datos (Base de Datos)

Descripción: Se encarga de almacenar los datos, como los proyectos, las tareas, los usuarios y los roles. Es común tanto para la aplicación web como para la app de escritorio, que interactúan con la base de datos a través del backend.

- Se puede implementar con MySQL, PostgreSQL, o MongoDB, dependiendo de las necesidades del proyecto.
- ORM como Sequelize (para Node.js) o Hibernate (para Java) puede facilitar la interacción con la base de datos.

Requisitos implementados en esta capa:

- REQ003: Creación de proyectos
- REQ004: Edición de proyectos
- REQ005: Eliminación de proyectos
- REQ010: Historial de actividades

Capa 4: Infraestructura

Descripción: Se ocupa de la seguridad, la escalabilidad, el rendimiento y la disponibilidad del sistema. Aquí se garantiza que las conexiones entre la app web, la app de escritorio y el backend se mantengan seguras y rápidas.

Servidores: Implementación en la nube mediante AWS, Azure, o Google Cloud para

escalabilidad.

Contenedores: Uso de Docker y Kubernetes para orquestar los servicios y asegurar la alta disponibilidad.

Seguridad: Implementación de certificados SSL/TLS para cifrar las comunicaciones y garantizar autenticación segura (mediante JWT o OAuth2).

Requisitos implementados en esta capa:

- REQ018: Seguridad
- REQ019: Escalabilidad
- REQ020: Rendimiento
- REQ021: Redundancia y resiliencia del sistema

Arquitectura Cliente-Servidor con Web y App de Escritorio

1. Clientes:

- Aplicación Web: Los usuarios acceden al sistema desde navegadores web, enviando solicitudes al servidor mediante HTTP/HTTPS.
- App Escritorio: Los usuarios también podrán utilizar una aplicación de escritorio, que tendrá acceso offline y capacidades adicionales como notificaciones nativas del sistema operativo.

2. Servidor:

- Este componente maneja todas las solicitudes de los clientes. Responde con datos y lógica de negocio procesada. También interactúa con la base de datos para obtener y almacenar información.

3. Comunicación Cliente-Servidor:

- HTTP/HTTPS: Las aplicaciones (web y desktop) enviarán solicitudes a los endpoints de la API para acceder o modificar datos.
- WebSockets: Podrían implementarse para notificaciones en tiempo real o actualizaciones sin necesidad de recargar la página o la aplicación.

Implementación de la Arquitectura de Capas con Hory MVS

Hory MVS es útil para organizar múltiples vistas que interactúan con una misma lógica de negocio. Aquí, las diferentes vistas serían:

- Web Frontend (Vista 1): Interfaz web para gerentes y empleados.
- Desktop Frontend (Vista 2): Aplicación de escritorio con capacidades adicionales.

Ambas vistas se conectan a la misma capa lógica de negocio y base de datos.

Ejemplo de Flujo de Implementación:

1. Autenticación de usuarios: El cliente (web o desktop) envía credenciales al servidor.
2. Backend verifica las credenciales: A través de JWT, el servidor valida la autenticidad y genera un token para futuros accesos.
3. Gestión de proyectos: Una vez autenticado, el cliente solicita al servidor una lista de proyectos o detalles específicos.
4. Servidor responde con los datos: Los datos se obtienen de la base de datos y se envían al cliente para visualización.
5. Modificaciones en proyectos: Las solicitudes de modificación son enviadas por el cliente al servidor, que las valida y actualiza en la base de datos.

1.9 TARJETAS CRCRF 1

Autenticación de Usuario	
Responsabilidad: <ul style="list-style-type: none"> • Validar nombre de usuario y contraseña. • Redirigir al usuario autenticado a su página de inicio. • Mostrar mensaje de error si las credenciales no son válidas. 	Colaboradores: <ul style="list-style-type: none"> • Base Datos Usuarios, Gestor Mensajes, Página Inicio Usuario.

RF 2

Gestión Proyectos	
Responsabilidad: <ul style="list-style-type: none"> • Crear, editar y eliminar proyectos. • Asignar roles y permisos a los miembros del equipo. 	Colaboradores: <ul style="list-style-type: none"> • Base De Datos Proyectos: Almacenar y gestionar los datos de los proyectos. • Asignación Roles: Controlar roles y permisos asignados a los usuarios. • Interfaz Usuario: Permitir a los usuarios interactuar con el sistema.

RF 3

Seguimiento Proyectos	
Responsabilidad: <ul style="list-style-type: none"> • Monitorear el progreso del proyecto mediante paneles de control. • Registrar el tiempo dedicado a cada tarea. • Generar informes de avance, desempeño y cumplimiento de objetivos. 	Colaboradores: <ul style="list-style-type: none"> • Base Datos Tareas: Almacenar el progreso de cada tarea. • Panel Control: Mostrar visualmente el estado del proyecto. • Generador Informes: Crear informes detallados del progreso.

RF 4

Creación Proyectos	
<p>Responsabilidad:</p> <ul style="list-style-type: none"> Permitir a los usuarios crear nuevos proyectos con nombre, descripción y fechas. 	<p>Colaboradores:</p> <ul style="list-style-type: none"> Base Datos Proyectos: Almacenar la información del nuevo proyecto. Interfaz Usuario: Permitir la entrada de los datos necesarios para el proyecto. Validación Datos: Verificar que todos los campos requeridos estén completos.

RF 5

Edición Proyectos	
<p>Responsabilidad:</p> <ul style="list-style-type: none"> Permitir a los usuarios editar detalles del proyecto como nombre, descripción y fechas. 	<p>Colaboradores:</p> <ul style="list-style-type: none"> Base Datos Proyectos: Guardar las actualizaciones. Interfaz Usuario: Mostrar la interfaz para editar los datos del proyecto. Validación Datos: Verificar que los datos editados sean correctos.

RF 6

Eliminación Proyectos	
<p>Responsabilidad:</p> <ul style="list-style-type: none"> Permitir a los usuarios eliminar proyectos existentes. 	<p>Colaboradores:</p> <ul style="list-style-type: none"> Base Datos Proyectos: Eliminar la información del proyecto de la base de datos. Interfaz Usuario: Confirmar la eliminación con el usuario. Gestor Mensajes: Mostrar un mensaje de advertencia antes de eliminar el proyecto.

RF 7

Planificación Tareas	
<p>Responsabilidad:</p> <ul style="list-style-type: none"> • Crear, asignar y priorizar tareas dentro de un proyecto. 	<p>Colaboradores:</p> <ul style="list-style-type: none"> • Base Datos Tareas: Almacenar las tareas creadas. • Asignación Tareas: Permitir la asignación de tareas a los usuarios. • Interfaz Usuario: Proveer un mecanismo para la creación y priorización de tareas.

RF 8

Seguimiento Tareas	
<p>Responsabilidad:</p> <ul style="list-style-type: none"> • Permitir a los usuarios actualizar el estado de las tareas. 	<p>Colaboradores:</p> <ul style="list-style-type: none"> • Base Datos Tareas: Registrar y actualizar el estado de las tareas. • Interfaz Usuario: Proveer la interfaz para actualizar el estado de las tareas. • Panel Control: Mostrar el estado actualizado de las tareas en tiempo real.

RF 9

Notificaciones Proyectos	
<p>Responsabilidad:</p> <ul style="list-style-type: none"> • Enviar notificaciones a los usuarios sobre tareas pendientes, plazos y actualizaciones. 	<p>Colaboradores:</p> <ul style="list-style-type: none"> • Sistema Notificaciones: Gestionar las alertas y recordatorios. • Notificaciones: Gestionar las alertas y recordatorios. • Notificaciones: Gestionar las alertas y recordatorios.

RF 10

Historial Actividades	
<p>Responsabilidad:</p> <ul style="list-style-type: none"> Mantener un registro del historial de actividades en cada proyecto. 	<p>Colaboradores:</p> <ul style="list-style-type: none"> Base Datos Actividades: Almacenar todas las modificaciones y cambios realizados. Interfaz Usuario: Mostrar el historial de actividades al usuario. Auditoría: Garantizar la integridad del historial y prevenir modificaciones no autorizadas.

RF 11

Generación Automatizada de Informes	
<p>Responsabilidad:</p> <ul style="list-style-type: none"> Generar informes sobre el progreso del proyecto, exportables en PDF o Excel. 	<p>Colaboradores:</p> <ul style="list-style-type: none"> Base Datos Tareas: Proveer la información necesaria para los informes. Exportador PDF/Excel: Transformar la información en formatos exportables. Interfaz Usuario: Permitir la solicitud y descarga de informes por el usuario.

RF 12

Administración de Roles y Accesos	
<p>Responsabilidad:</p> <ul style="list-style-type: none"> Asignar roles a los usuarios. Definir los permisos asociados a cada rol. Administrar los de los usuarios Verificar que los usuarios accedan únicamente a las funciones correspondientes a su rol. 	<p>Colaboradores:</p> <ul style="list-style-type: none"> Usuario: Recibe el rol y accesos según su perfil Administrador: Gestiona los roles y permisos Módulo de Seguridad: Verifica el acceso Base de Datos: Almacena la información de roles y permisos

RF 13

Generación Automatizada de Informes	
Responsabilidad: <ul style="list-style-type: none"> • Integrar calendarios externos • Sincronizar eventos y plazos con los calendarios externos • Actualizar los eventos en tiempo real 	Colaboradores: <ul style="list-style-type: none"> • Calendario externo: Outlook Google calendar entre otros • Usuario: Recibe notificaciones y actualizaciones de eventos • Administrador de proyecto: Gestiona los eventos de los calendarios externos • Base de Datos: Almacena la información de eventos y autenticación

RF 14

Panel Unificado de Tareas Múltiples	
Responsabilidad: <ul style="list-style-type: none"> • Mostrar distintas tareas en un solo panel • Permitir la visualización de prioridades de tareas • Filtrar y ordenar tareas por prioridad • Sincronizar cambios en tiempo real 	Colaboradores: <ul style="list-style-type: none"> • Usuario: Organiza y actualiza las tareas • Administrador de proyecto: Supervisa las distintas tareas • Base de Datos: Almacena la información de tareas y detalles relevantes

RF 15

Automatización de Tareas Repetitivas	
Responsabilidad: <ul style="list-style-type: none"> • Gestionar tareas repetitivas • Ejecutar automáticamente las tareas programadas • Notificar al usuario la ejecución de tareas repetitivas 	Colaboradores: <ul style="list-style-type: none"> • Usuario: Recibe los beneficios de la automatización • Administrador de sistema: Supervisa la ejecución de tareas repetitivas • Base de Datos: Almacena los estados de las tareas repetitivas

RNF 16

Gestión de recursos	
Responsabilidad: <ul style="list-style-type: none"> • Asignar, monitorear y optimizar los recursos. • Generar informes. • Gestionar los recursos de la empresa. 	Colaboradores: <ul style="list-style-type: none"> • Proyecto, gestor de empleados, Inventario, finanzas.

RNF 17

Usabilidad	
Responsabilidad: <ul style="list-style-type: none"> • Proporcionar un acceso claro y sencillo a todas las secciones del sitio. • Asegurarse de que los usuarios puedan interactuar fácilmente con el contenido y funciones. • Organizar los elementos visuales y textuales de forma lógica, con una jerarquía clara. 	Colaboradores: <ul style="list-style-type: none"> • Diseñador: Responsable de la apariencia y experiencia del usuario, asegurando que el diseño sea atractivo y funcional. • Desarrollador web: Implementa el código que hace que la página sea interactiva y accesible. • Tester de usabilidad: Realiza pruebas con usuarios para detectar posibles problemas de interacción y navegación.

RNF 18

Seguridad	
Responsabilidad: <ul style="list-style-type: none"> • Protección de la información. • Implementación de políticas de seguridad. • Gestión de acceso. • Encriptación de datos. 	Colaboradores: <ul style="list-style-type: none"> • Equipo de ciberseguridad: Monitorea las amenazas, vulnerabilidades y realiza análisis de riesgo. • Desarrollador web: Implementa y mantiene las herramientas y medidas de seguridad, como antivirus y sistemas de detección de intrusos.

RNF 19

Escalabilidad	
Responsabilidad: <ul style="list-style-type: none"> • Asegurar que la plataforma soporte un número creciente de usuarios concurrentes sin afectar el rendimiento. • Mantener tiempos de carga rápidos y una experiencia fluida a medida que crece la base de usuarios y datos. • Tener una base de datos que permita manejar un gran volumen de datos, consultas y transacciones sin reducir la eficiencia. 	Colaboradores: <ul style="list-style-type: none"> • Equipo de desarrollo backend: Diseña la arquitectura del sistema y optimiza el manejo de bases de datos, servidores y servicios web. • Desarrolladores frontend: Aseguran que la interfaz sea eficiente, ligera y escalable a medida que aumenta la interacción de los usuarios.

RNF 20

Rendimiento	
<p>Responsabilidad:</p> <ul style="list-style-type: none"> • Reducir el tiempo de carga de páginas y recursos, asegurando una experiencia de usuario rápida y fluida. 	<p>Colaboradores:</p> <ul style="list-style-type: none"> • Administradores de bases de datos: Optimizan las bases de datos para asegurar que las consultas sean rápidas y se manejen eficientemente grandes volúmenes de datos. • Desarrolladores: Optimizan la lógica del servidor, las consultas a la base de datos y la eficiencia del manejo de datos.

RNF 21

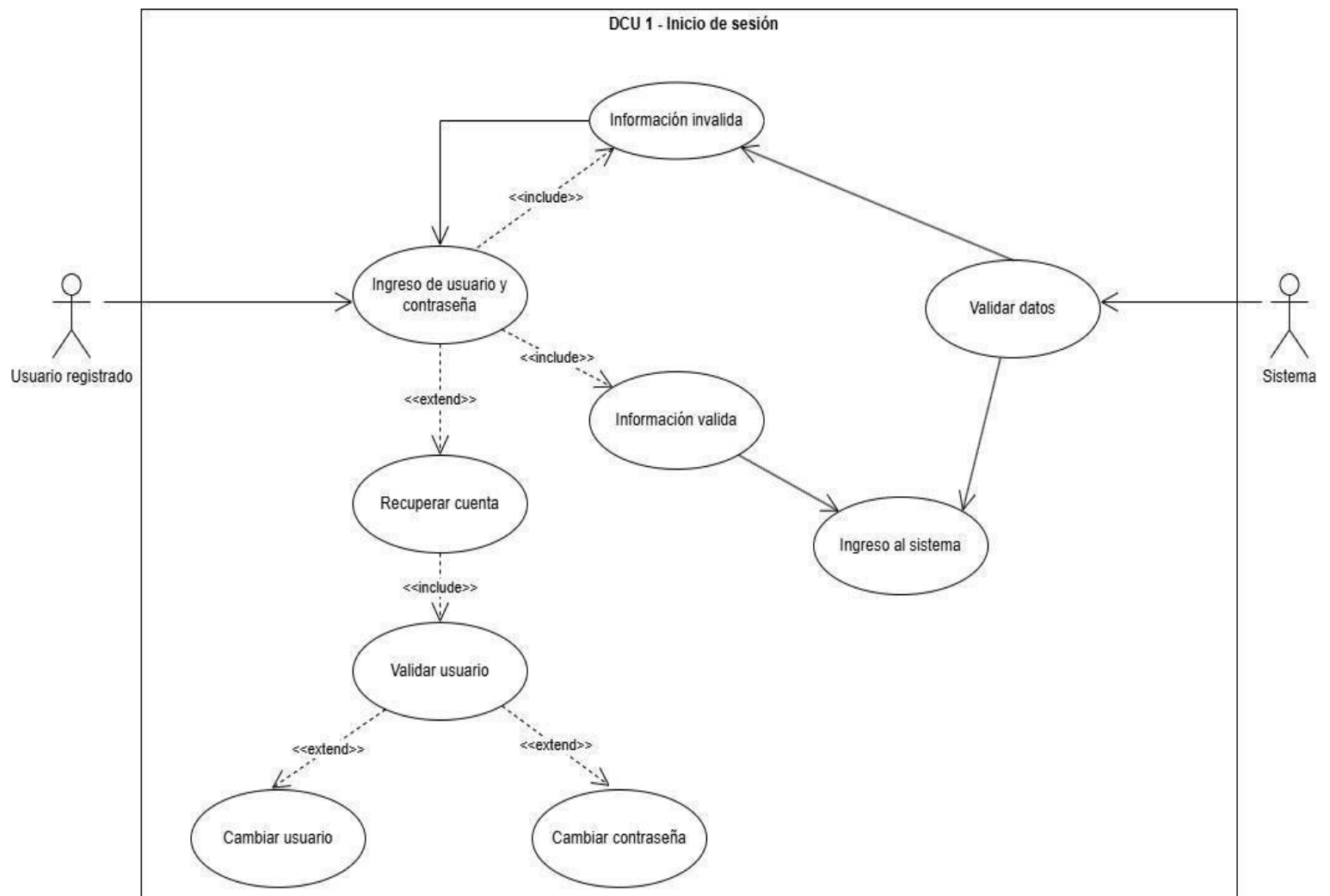
Redundancia y Resiliencia del Sistema	
<p>Responsabilidad:</p> <ul style="list-style-type: none"> • Asegurar que haya componentes duplicados (servidores, bases de datos) para tomar el relevo en caso de fallos. • Configurar el sistema para que esté disponible la mayor parte del tiempo, incluso en caso de fallos o mantenimiento programado. 	<p>Colaboradores:</p> <ul style="list-style-type: none"> • Administradores de sistemas: Configuran y gestionan la infraestructura redundante, como servidores y redes. • Equipo de desarrollo: Diseña la aplicación para soportar la redundancia y la resiliencia, implementando prácticas como la replicación de datos y el manejo de fallos.

1.10 DIAGRAMAS DE CASO DE USO

Inicio de sesión:

Nombre:	Iniciar Sesión
Campo de prioridad:	Baja
Descripción: El sistema debe permitir a los usuarios autenticarse utilizando un nombre de usuario y contraseña válidos.	
Actores: Usuario, sistema.	
Precondiciones: Existencia de una cuenta de usuario: El usuario debe de tener un usuario y una contraseña en la base de datos para poder ingresar. Sin estos datos previamente establecidos no se dará acceso. Usuario y contraseña validos: El usuario proporciona un usuario y una contraseña asociados los cuales deben de coincidir con la base de datos.	
Flujo Normal: El usuario accede a la pantalla de inicio de sesión El usuario proporciona el usuario y la contraseña El sistema valida los datos ingresados por el usuario Si los datos ingresados coinciden con la base de datos se otorga el acceso El usuario es redirigido a la página principal para seleccionar un proyecto para comenzar a editar sus propiedades	
Flujo alternativo: El usuario accede a la pantalla de inicio de sesión El usuario proporciona el usuario y la contraseña El sistema valida los datos ingresados por el usuario Los datos ingresados no coinciden con la base de datos, se muestra un mensaje de error indicando que los datos no son validos Al usuario se le da la opción de volver a intentarlo con las credenciales correctas o puede tener la opción de recuperar cuenta donde valida primero el usuario y después si se da la posibilidad de cambiar usuario o contraseña Una vez restablecidos los datos se dará la posibilidad de volver a intentar iniciar sesión	
Postcondiciones: Autenticación del usuario: Garantizar que el usuario haya sido autenticado correctamente, lo que asegura que solo los usuarios autorizados tengan acceso al sistema. Inicio de sesión exitoso: Hay que confirmar que el usuario ha podido iniciar sesión con éxito, lo que implica que el sistema ha validado las credenciales proporcionadas y ha permitido el acceso a las funcionalidades de la aplicación. Estado de la sesión: Establecer una sesión válida para el usuario, lo que le permite realizar acciones autorizadas dentro del sistema durante un tiempo determinado.	

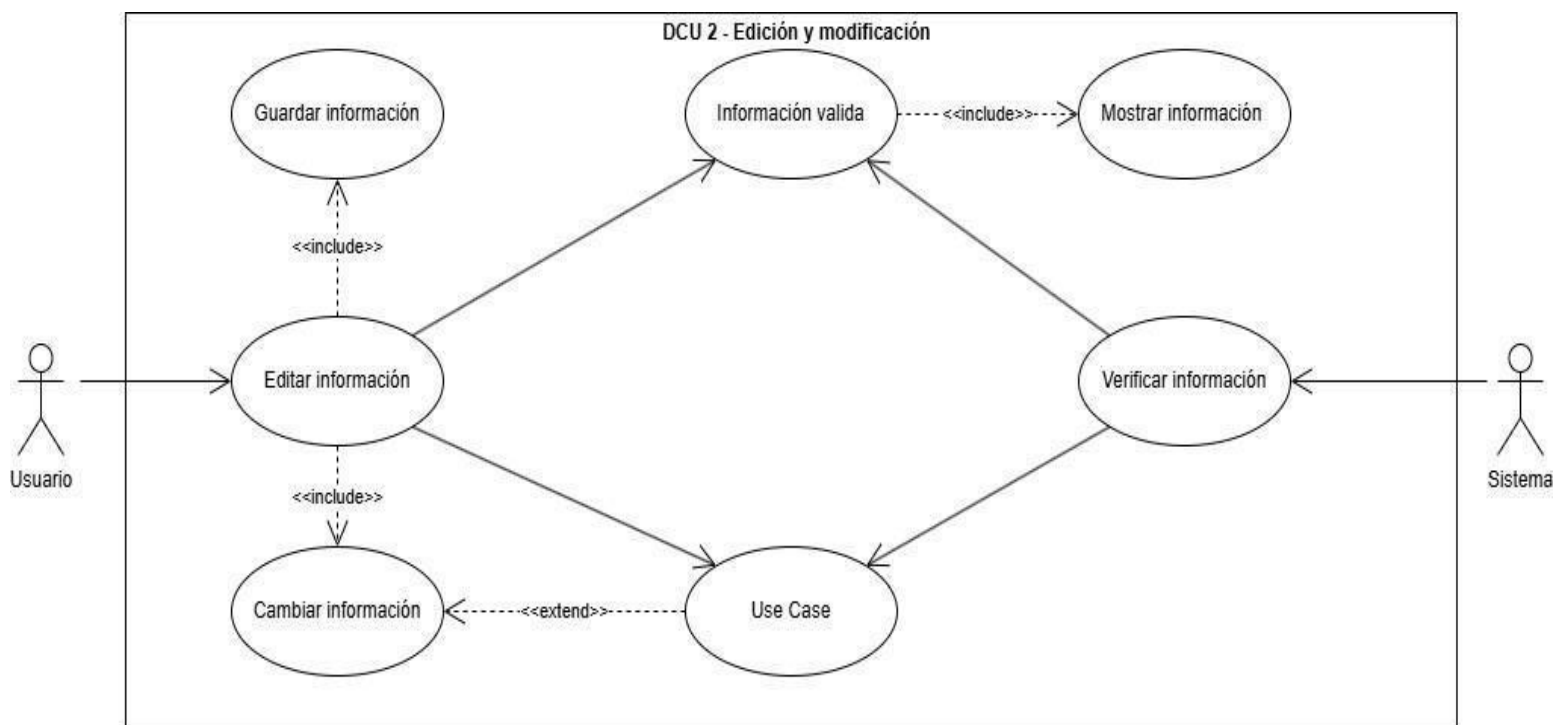
Diagrama 1.



Edición y modificación:

Nombre:	Edición y modificación
Campo de prioridad:	Alta
Descripción: El sistema debe permitir la creación, edición y eliminación de proyectos. Debe asignar roles y permisos a los miembros del equipo.	
Actores: Usuario.	
Precondiciones: El usuario debe estar autenticado en el sistema. El proyecto debe existir en la base de datos. El usuario debe tener los permisos necesarios para editar o modificar proyectos.	
Flujo Normal: El usuario inicia sesión en el sistema. Navega a la sección de proyectos. Selecciona el proyecto que desea editar o modificar. Realiza las modificaciones necesarias (por ejemplo, actualiza la descripción, agrega archivos adjuntos, etc.). Guarda los cambios.	
Flujo alternativo: Si el proyecto no existe, mostrar un mensaje de error al usuario. Si el usuario no tiene permisos suficientes, mostrar un mensaje de acceso denegado. Si se produce un error al guardar los cambios, mostrar un mensaje de error y permitir al usuario intentarlo nuevamente.	
Postcondiciones: El proyecto se ha actualizado correctamente en la base de datos. El usuario recibe una confirmación de que los cambios se han guardado.	

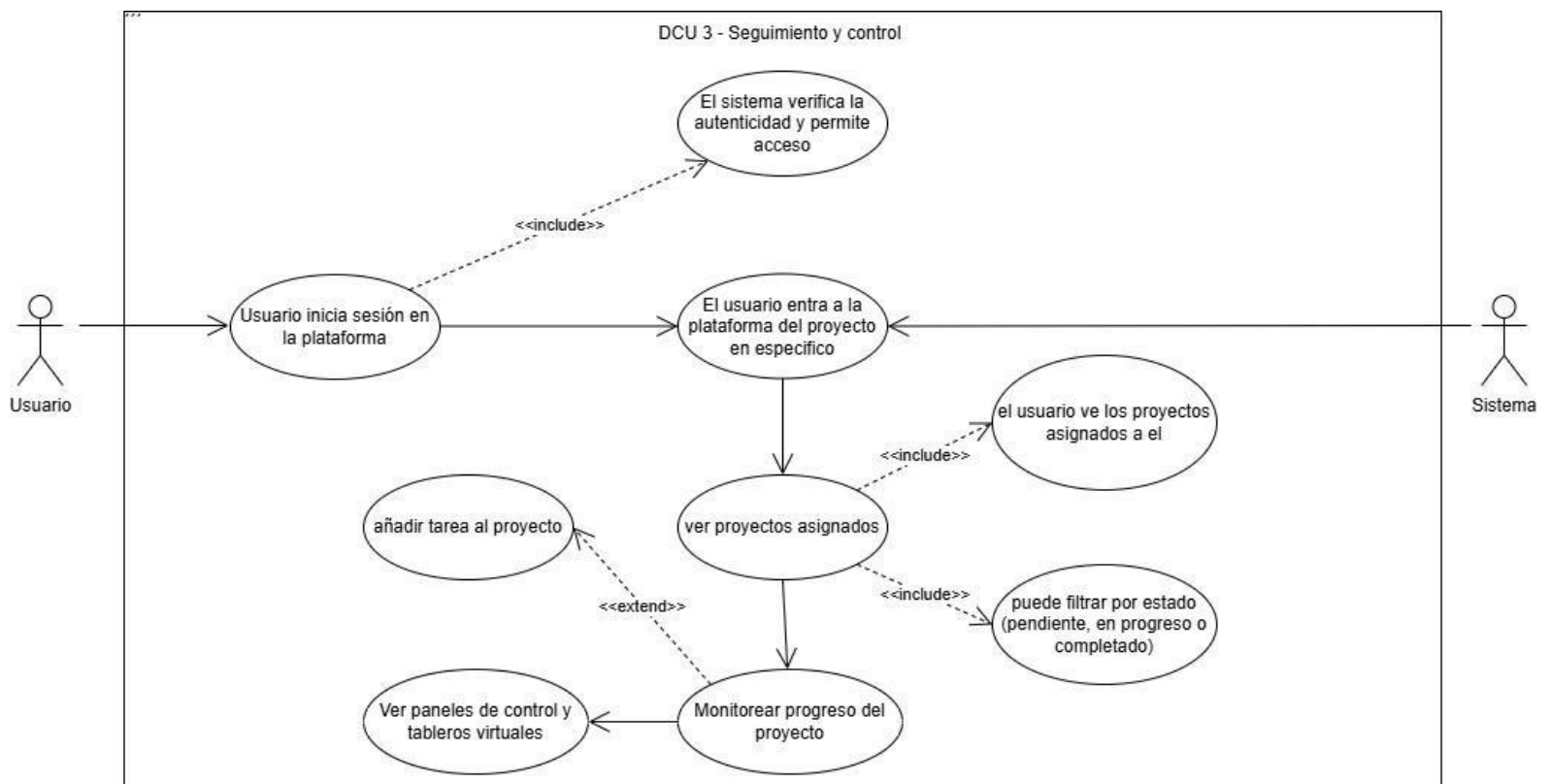
Diagrama 2.



Seguimiento y control:

Nombre:	Seguimiento y control
Campo de prioridad:	Alta
Descripción:	Monitorear el progreso del proyecto mediante paneles de control y tableros visuales.
Actores:	Usuario.
Precondiciones:	<p>El usuario debe estar autenticado en el sistema.</p> <p>El proyecto debe estar registrado en la base de datos.</p> <p>Los permisos del usuario deben permitir el acceso al módulo de seguimiento y control.</p> <p>Donde el usuario podrá verificar el estado de un proyecto (pendiente, en progreso o completado)</p>
Flujo Normal:	<p>El usuario inicia sesión en el sistema.</p> <p>Selecciona un proyecto específico.</p> <p>Accede a la funcionalidad de seguimiento y control.</p> <p>Realiza acciones como actualizar el estado del proyecto, etc.</p>
Flujo alternativo:	<p>Si el proyecto no existe en la base de datos, mostrar un mensaje de error.</p> <p>Si el usuario no tiene permisos para acceder al módulo de seguimiento, redirigirlo a una página de acceso denegado.</p>
Postcondiciones:	<p>Después de agregar una tarea, la base de datos se actualiza con la nueva información.</p> <p>Después de cambiar el estado del proyecto, se envía una notificación al equipo.</p>

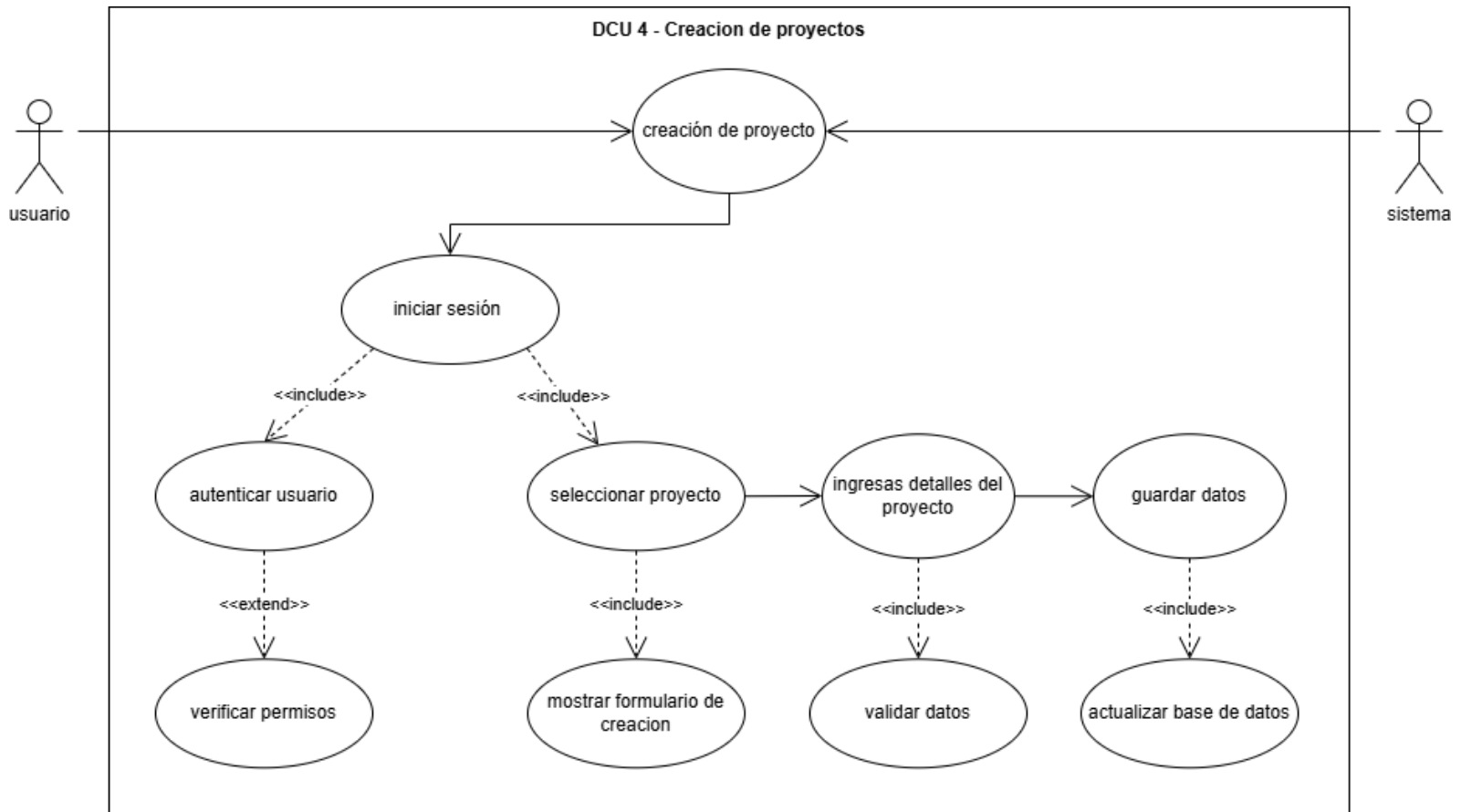
Diagrama 3.



Creación de proyectos:

Nombre:	Creación de proyectos
Campo de prioridad:	Alta
Descripción: El sistema debe permitir a los usuarios crear nuevos proyectos proporcionando detalles como nombre, descripción, fechas de inicio y finalización.	
Actores: Usuario.	
Precondiciones: El usuario debe estar autenticado en el sistema. El usuario debe tener permisos para crear proyectos	
Flujo Normal: El usuario inicia sesión en el sistema. Selecciona la opción para crear un nuevo proyecto. El usuario ingresa los detalles del proyecto, como nombre, descripción, fecha de inicio y fecha de finalización. El sistema guarda el nuevo proyecto en la base de datos. El sistema confirma la creación exitosa del proyecto al usuario.	
Flujo alternativo: Si el usuario no tiene permisos para crear proyectos, se redirige a una página de acceso denegado Si los datos ingresados están incompletos o no son válidos, se muestra un mensaje de error solicitando corrección.	
Postcondiciones: El proyecto se registra en la base de datos. Se notifica al usuario y a los miembros del equipo (si corresponde) sobre la creación del nuevo proyecto.	

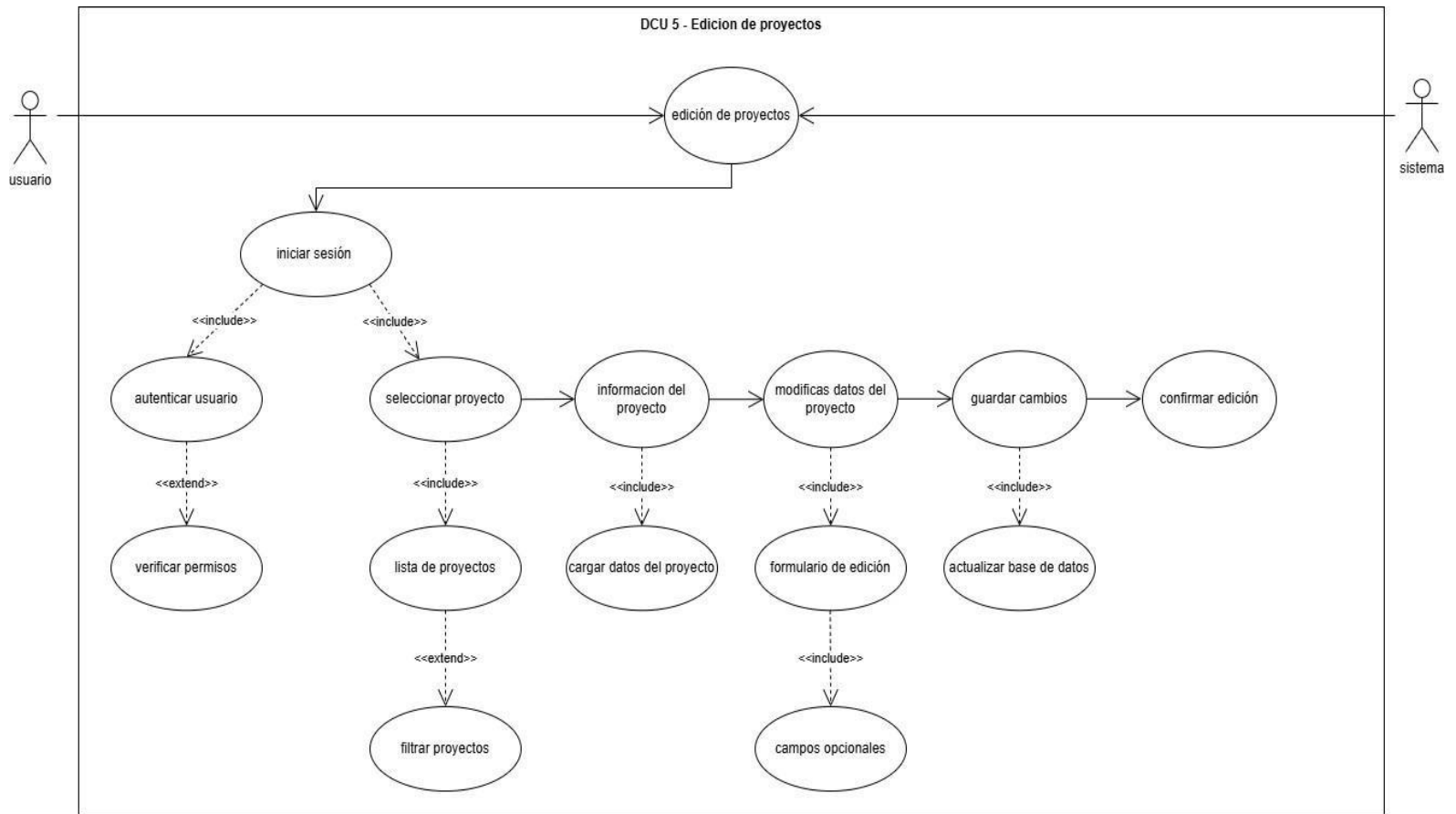
Diagrama 4.



Edición de proyectos:

Nombre:	Edición de proyectos
Campo de prioridad:	Media
Descripción: Los usuarios deben poder editar los detalles de un proyecto existente, como su nombre, descripción y fechas.	
Actores: Usuario.	
Precondiciones: El usuario debe estar autenticado en el sistema. El usuario debe tener permisos para editar proyectos. El proyecto debe existir en la base de datos.	
Flujo Normal: El usuario inicia sesión en el sistema. Selecciona el proyecto que desea editar. El usuario modifica los detalles del proyecto, como el nombre, descripción o fechas. El sistema guarda los cambios realizados en la base de datos. El sistema confirma la edición exitosa del proyecto al usuario.	
Flujo alternativo: Si el usuario no tiene permisos para editar proyectos, se muestra un mensaje de error Si los datos ingresados son incompletos o no válidos, se muestra un mensaje de error solicitando corrección.	
Postcondiciones: Se notifica al usuario y a los miembros del equipo (si corresponde) sobre los cambios realizados en el proyecto.	

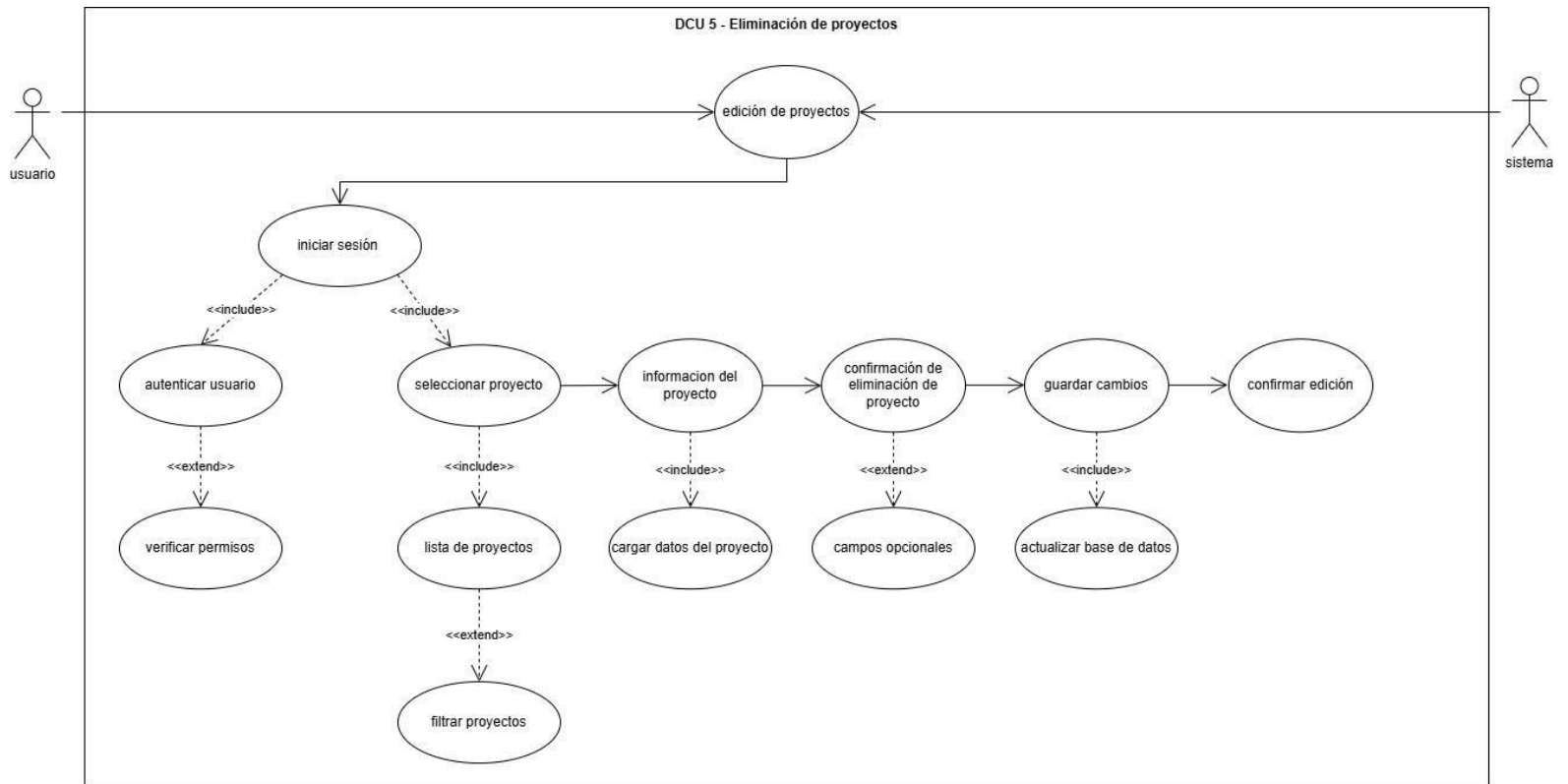
Diagrama 5.



Eliminación de proyectos:

Nombre:	Eliminación de proyectos
Campo de prioridad:	Media
Descripción: El sistema debe permitir a los usuarios eliminar proyectos existentes.	
Actores: Usuario.	
Precondiciones: El usuario debe estar autenticado en el sistema. El usuario debe tener permisos adecuados para eliminar proyectos. El proyecto que se desea eliminar debe existir en el sistema.	
Flujo Normal: El usuario selecciona el proyecto que desea eliminar desde la lista de proyectos. El sistema muestra una confirmación de eliminación. El usuario confirma la eliminación del proyecto. El sistema elimina el proyecto seleccionado de la base de datos. El sistema muestra un mensaje de éxito confirmando que el proyecto ha sido eliminado.	
Flujo alternativo: El usuario cancela la eliminación El proyecto no puede ser eliminado debido a restricciones	
Postcondiciones: El proyecto eliminado ya no debe aparecer en la lista de proyectos ni en ningún otro lugar del sistema.	

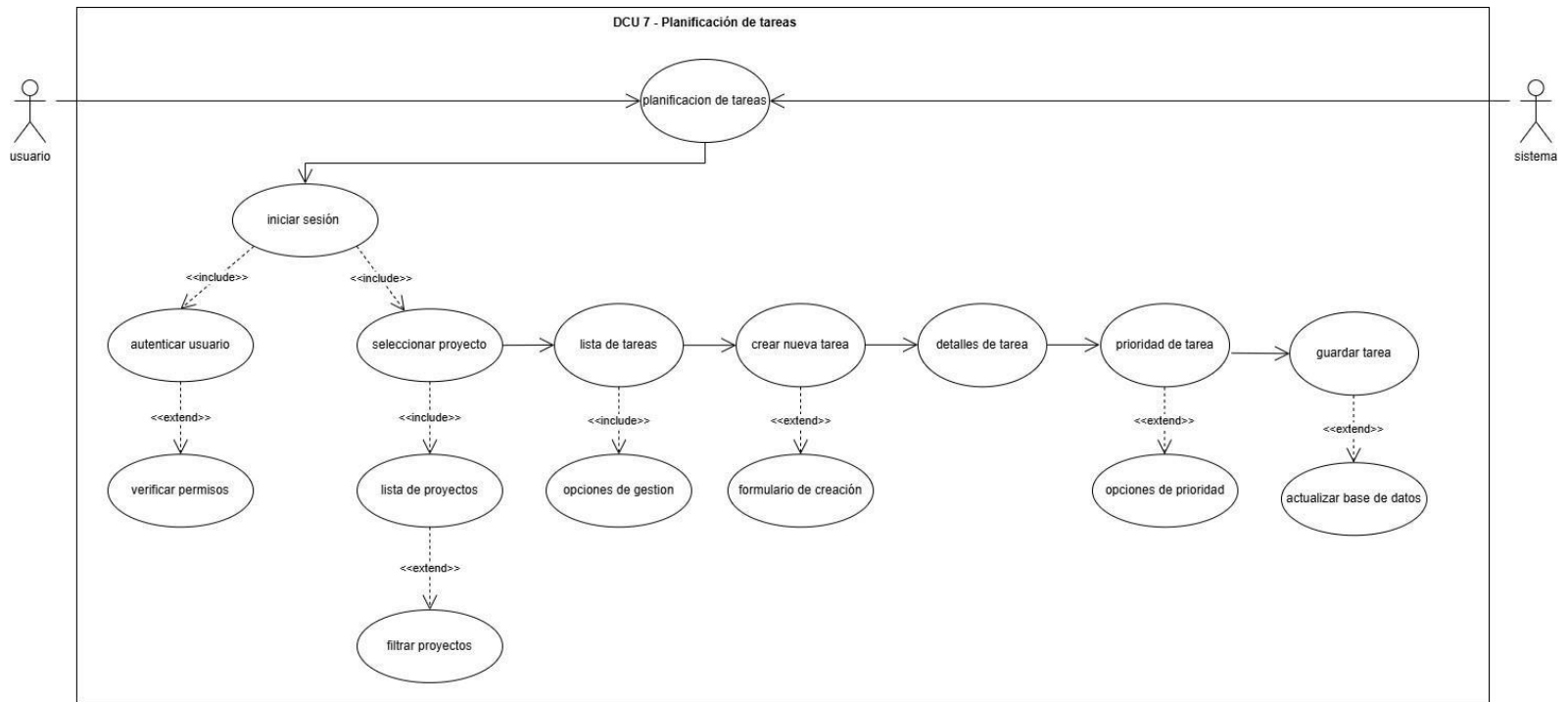
Diagrama 6.



Planificación de tareas:

Nombre:	Planificación de tareas
Campo de prioridad:	Alta
Descripción: El sistema debe permitir a los usuarios crear, asignar y priorizar tareas dentro de un proyecto.	
Actores: Usuario.	
Precondiciones: El usuario debe estar autenticado en el sistema. El usuario debe tener permisos adecuados para gestionar tareas dentro del proyecto. El proyecto al que se desea añadir tareas debe existir	
Flujo Normal: El usuario selecciona el proyecto en el que desea planificar tareas. El sistema muestra una vista con las opciones para crear y gestionar tareas. El usuario crea una nueva tarea, especificando detalles como el título, descripción, fecha límite, y prioridad. El usuario establece la prioridad de la tarea (por ejemplo, alta, media, baja). El sistema guarda la tarea y la muestra en la lista de tareas del proyecto. El sistema muestra un mensaje de éxito confirmando que la tarea ha sido creada y asignada.	
Flujo alternativo: El usuario omite detalles necesarios	
Postcondiciones: La tarea debe aparecer en la lista de tareas del proyecto con los detalles especificados. La tarea debe estar asignada al miembro del equipo seleccionado con la prioridad establecida. El usuario debe recibir una confirmación de que la tarea ha sido creada y asignada correctamente.	

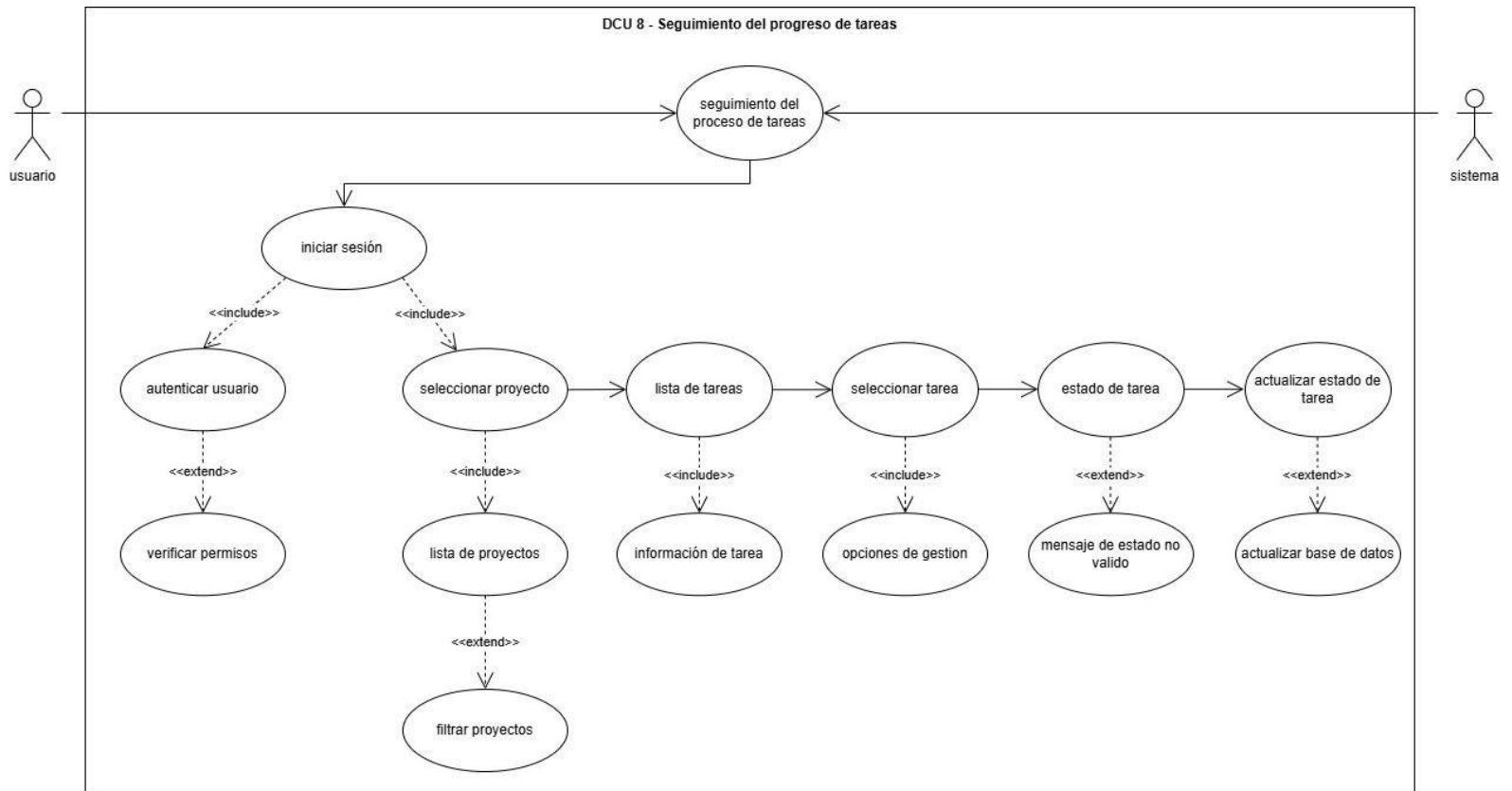
Diagrama 7.



Seguimiento del progreso de tareas:

Nombre:	Seguimiento del progreso de tareas
Campo de prioridad:	Alta
Descripción: El sistema debe permitir a los usuarios actualizar el estado de las tareas (por ejemplo, "En progreso" "Completado").	
Actores: Usuario.	
Precondiciones: El usuario debe estar autenticado en el sistema. El usuario debe tener permisos para actualizar el estado de las tareas. La tarea que se desea actualizar debe existir en el sistema.	
Flujo Normal: El usuario accede a la lista de tareas asignadas o a las tareas del proyecto. El usuario selecciona la tarea cuyo estado desea actualizar. El sistema muestra la información actual de la tarea junto con opciones para cambiar su estado. El usuario elige el nuevo estado de la tarea (por ejemplo, "En progreso", "Completado"). El sistema actualiza el estado de la tarea en la base de datos. El sistema muestra un mensaje de éxito confirmando que el estado de la tarea ha sido actualizado.	
Flujo alternativo: La tarea no se puede encontrar o no está disponible El usuario selecciona un estado no válido	
Postcondiciones: El estado de la tarea debe reflejar el nuevo estado seleccionado por el usuario. La actualización del estado debe ser visible en la lista de tareas y en cualquier informe relacionado El usuario debe recibir una confirmación de que el estado de la tarea ha sido actualizado correctamente	

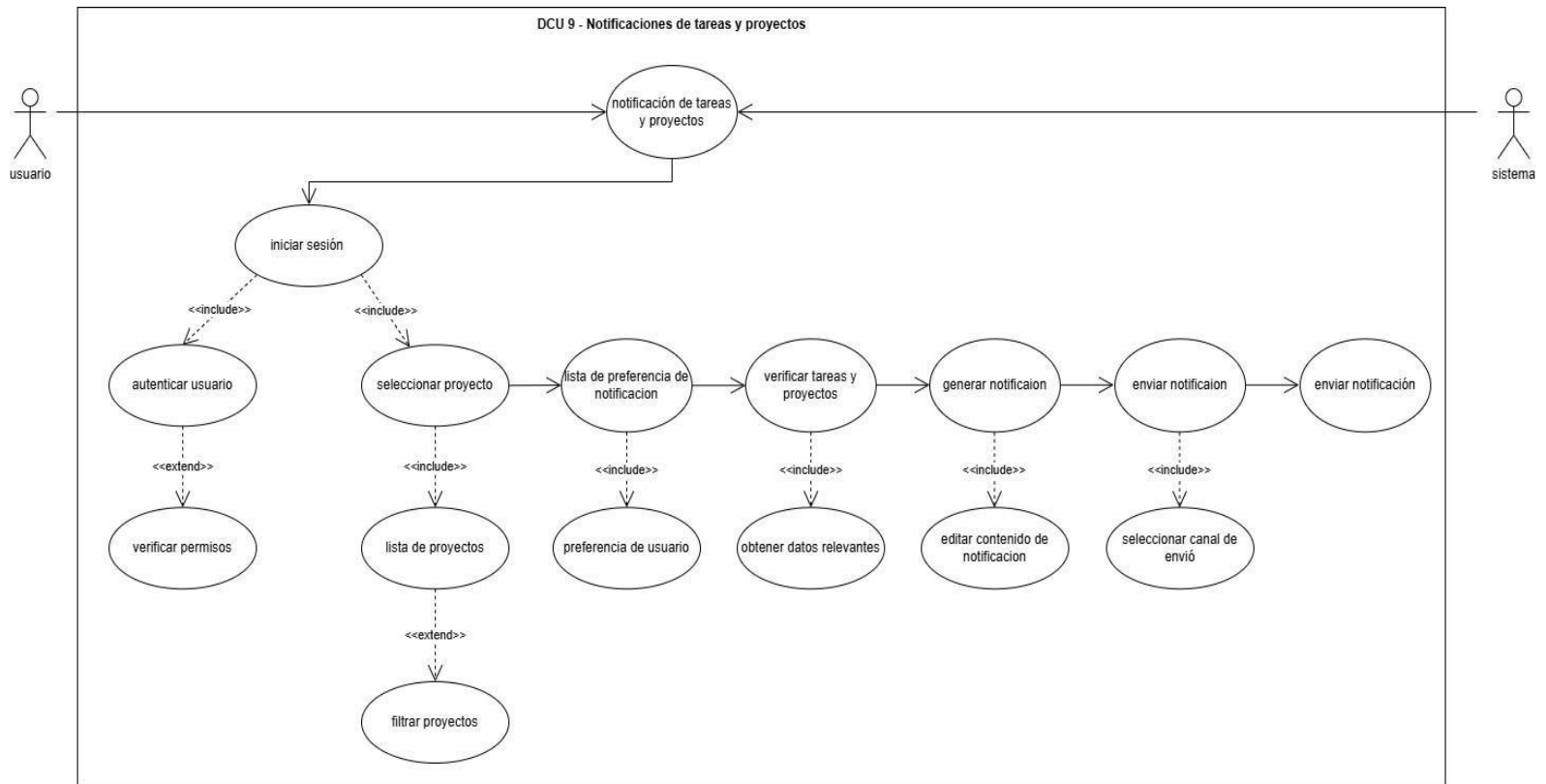
Diagrama 8.



Notificaciones de tareas y proyectos:

Nombre:	Notificaciones de tareas y proyectos
Campo de prioridad:	Media
Descripción: El sistema debe enviar notificaciones a los usuarios sobre tareas pendientes, plazos próximos y actualizaciones de proyectos.	
Actores: Usuario.	
Precondiciones: El usuario debe estar autenticado en el sistema. El usuario debe haber configurado sus preferencias de notificación (por ejemplo, tipo de notificaciones, frecuencia). Las tareas y proyectos deben tener plazos y actualizaciones relevantes para notificar.	
Flujo Normal: El sistema verifica las tareas y proyectos con plazos próximos y actualizaciones desde la última notificación. El sistema determina los usuarios que deben recibir notificaciones basadas en sus preferencias y en las tareas y proyectos relevantes. El sistema genera notificaciones para los usuarios sobre tareas pendientes, plazos próximos y actualizaciones de proyectos. El sistema envía las notificaciones a los usuarios por el canal seleccionado (por ejemplo, correo electrónico, notificaciones en la aplicación). El usuario recibe las notificaciones y revisa la información actualizada.	
Flujo alternativo: No hay tareas para completar	
Postcondiciones: Las notificaciones deben reflejar la información más reciente sobre las tareas y proyectos.	

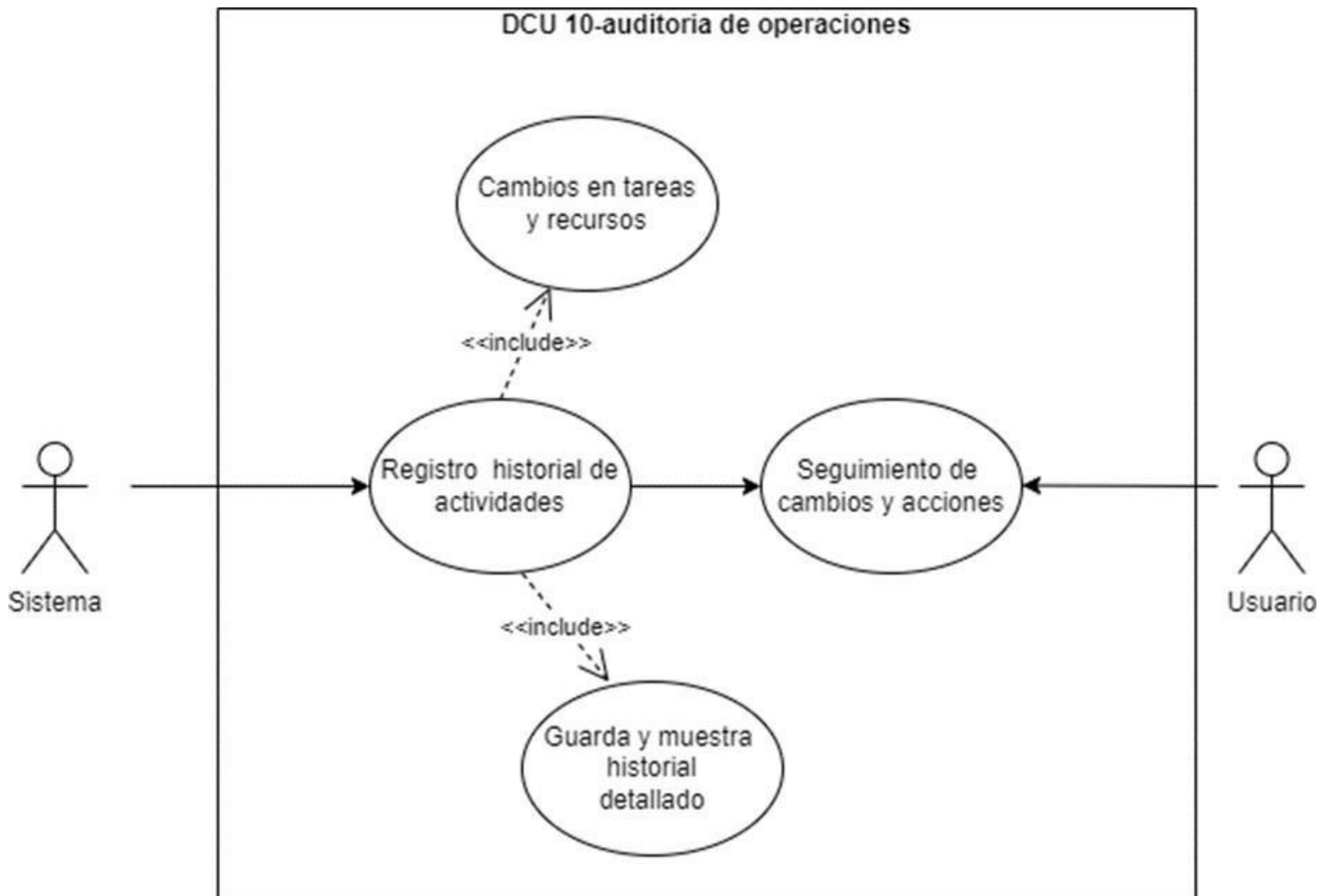
Diagrama 9.



Auditoria de operaciones y modificaciones:

Nombre:	Auditoria de operaciones y modificaciones
Campo de prioridad:	Media
Descripción: El sistema debe mantener un registro del historial de actividades para cada proyecto, incluyendo cambios en tareas y recursos.	
Actores: Usuario.	
Precondiciones: El usuario está autenticado en el sistema y tiene los permisos necesarios para acceder al proyecto. El proyecto seleccionado existe en la base de datos.	
Flujo Normal: El usuario selecciona un proyecto de la lista de proyectos disponibles. El sistema recupera el historial completo de actividades asociado al proyecto seleccionado. El sistema presenta al usuario una interfaz clara y organizada que muestra el historial, incluyendo: Descripción detallada de la actividad (Usuario que realizó la actividad) Otros detalles relevantes (cambios en el estado de la tarea, recursos asignados)	
Flujo alternativo: Usuario selecciona un proyecto inexistente: El sistema muestra un mensaje de error indicando que el proyecto no se encuentra. El usuario no tiene permisos para acceder al proyecto: El sistema deniega el acceso y muestra un mensaje de error. Ocurre un error al recuperar el historial: El sistema muestra un mensaje de error general y registra el incidente para su posterior análisis	
Postcondiciones: El usuario ha visualizado el historial de actividades del proyecto seleccionado. El sistema mantiene la integridad de los datos del historial.	

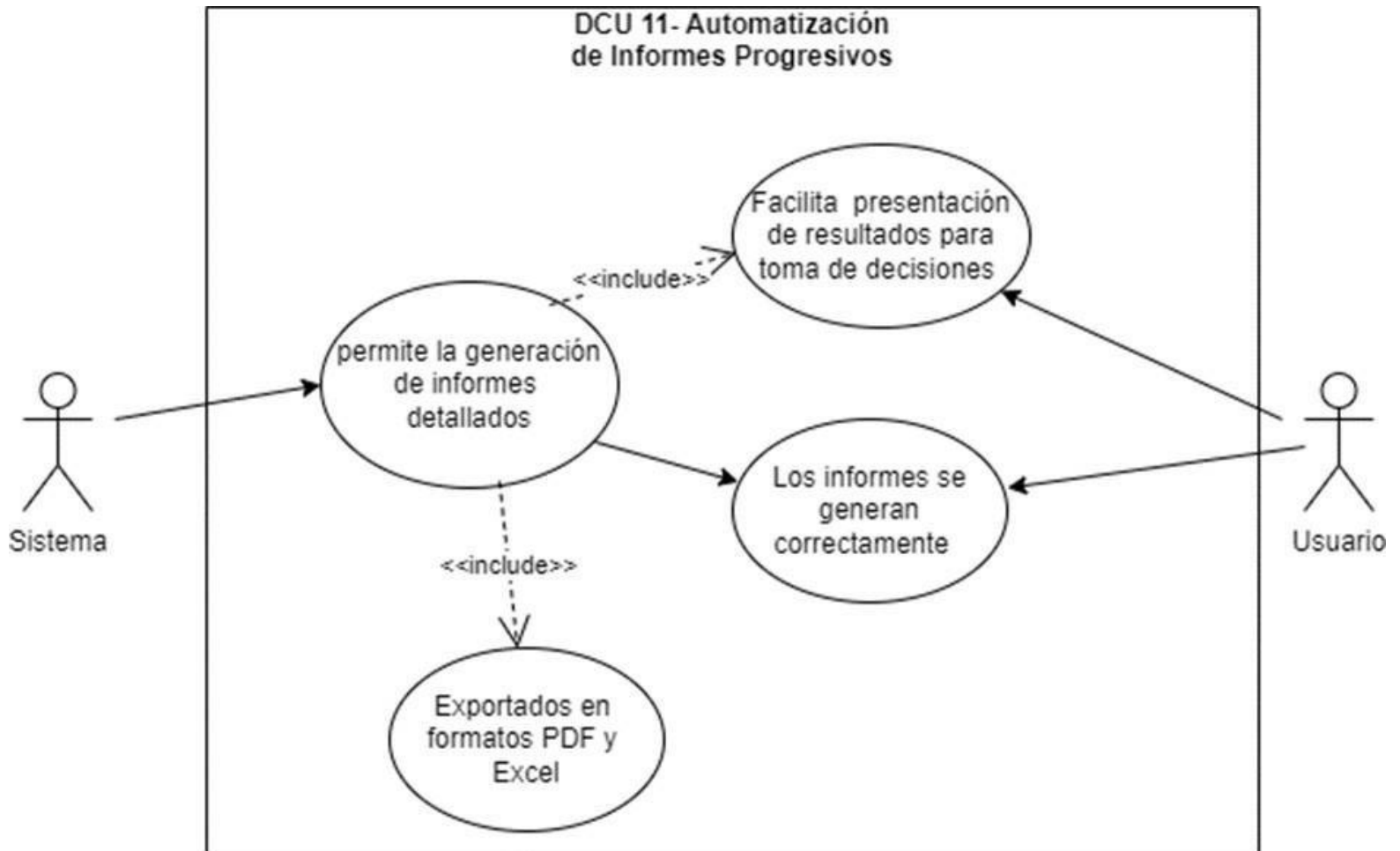
Diagrama 10.



Automatización de Informes Progresivos:

Nombre:	Automatización de Informes Progresivos
Campo de prioridad:	Alta
Descripción: El sistema debe permitir la generación automática de informes detallados sobre el progreso de los proyectos, que pueden ser exportados en formatos como PDF o Excel.	
Actores: Usuario.	
Precondiciones: El sistema cuenta con los datos necesarios para generar los informes (tareas, recursos, fechas de inicio y fin, etc.). El usuario ha configurado los parámetros del informe (tipo de informe, frecuencia, formato de salida, etc.).	
Flujo Normal: El sistema verifica si se ha alcanzado el intervalo de tiempo programado para la generación del informe o si el usuario ha solicitado un informe manualmente. El sistema recopila los datos relevantes de los proyectos según los parámetros configurados. El sistema procesa los datos y genera el informe en el formato especificado (PDF, Excel). El sistema notifica al usuario sobre la generación del informe (por correo electrónico, notificación en la plataforma, etc.).	
Flujo alternativo: No hay datos suficientes para generar el informe: El sistema muestra un mensaje de error indicando que faltan datos. Ocurre un error durante la generación del informe: El sistema registra el error y notifica al usuario. El formato de salida especificado no es válido: El sistema genera el informe en un formato por defecto o muestra un mensaje de error.	
Postcondiciones: El informe se genera correctamente y está disponible para el usuario en el formato especificado. Los datos del informe reflejan el estado actual de los proyectos.	

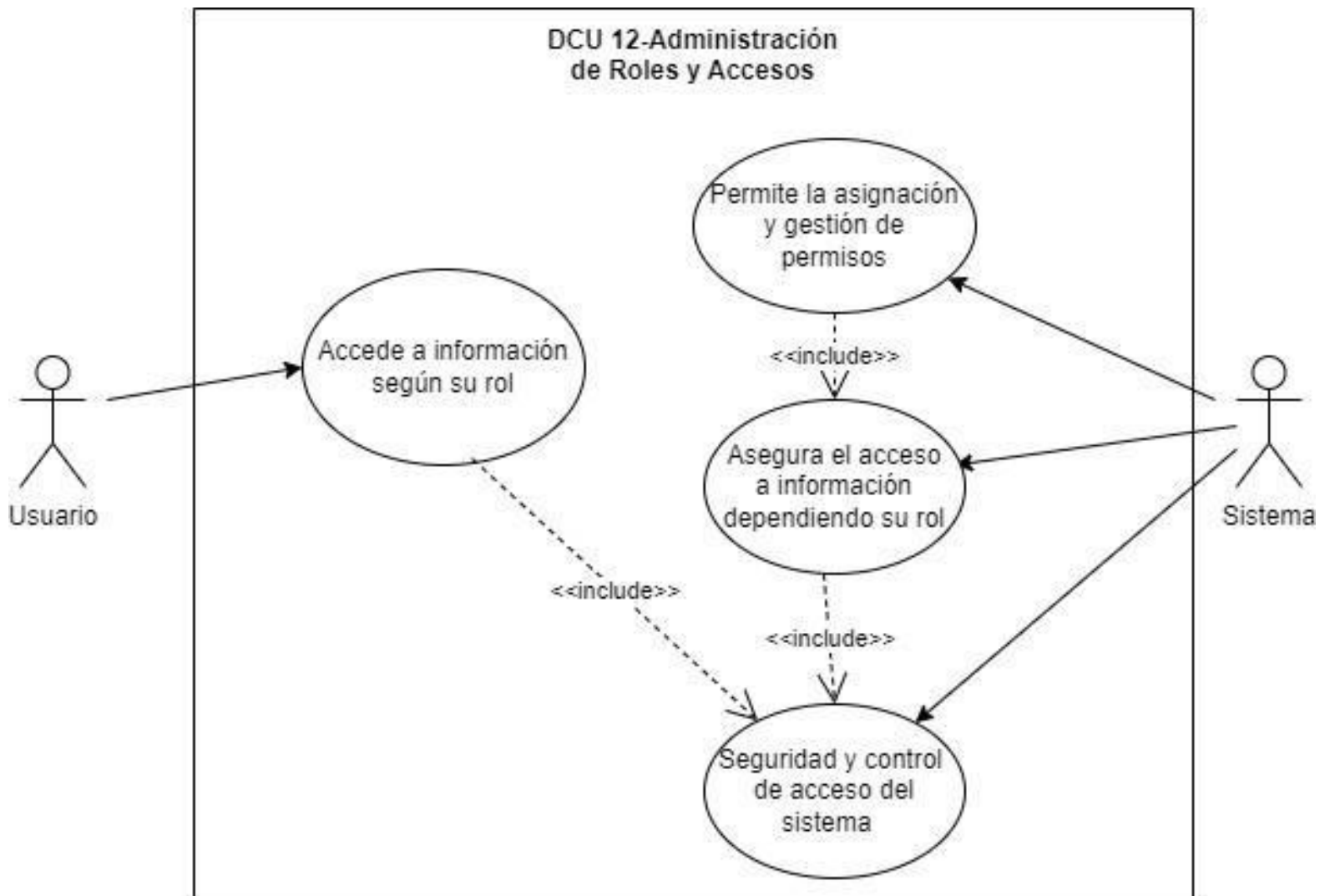
Diagrama 11.



Administración de Roles y Accesos:

Nombre:	Administración de Roles y Accesos
Campo de prioridad:	Alta
Descripción: El sistema debe permitir la asignación y gestión de permisos y roles a los usuarios, garantizando que solo accedan a la información y funcionalidades que les correspondan.	
Actores: Usuario.	
Precondiciones: El administrador ha iniciado sesión en el sistema con privilegios administrativos. Existen roles y permisos definidos en el sistema.	
Flujo Normal: El administrador selecciona un usuario de la lista de usuarios. El sistema muestra los roles y permisos disponibles para el usuario seleccionado. El administrador selecciona los roles y permisos que desea asignar al usuario. El sistema actualiza la información del usuario, asignándole los roles y permisos seleccionados	
Flujo alternativo: El administrador intenta asignar un rol que no existe: El sistema muestra un mensaje de error indicando que el rol no es válido. El administrador intenta asignar un permiso que no está asociado al rol: El sistema muestra un mensaje de error indicando que el permiso no está disponible para el rol seleccionado. Ocurre un error al actualizar la información del usuario: El sistema registra el error y notifica al administrador	
Postcondiciones: El informe se genera correctamente y está disponible para el usuario en el formato especificado. Los datos del informe reflejan el estado actual de los proyectos. Los roles y permisos del usuario seleccionado se han actualizado correctamente en el sistema. El usuario solo podrá acceder a las funcionalidades y datos permitidos por los roles asignados.	

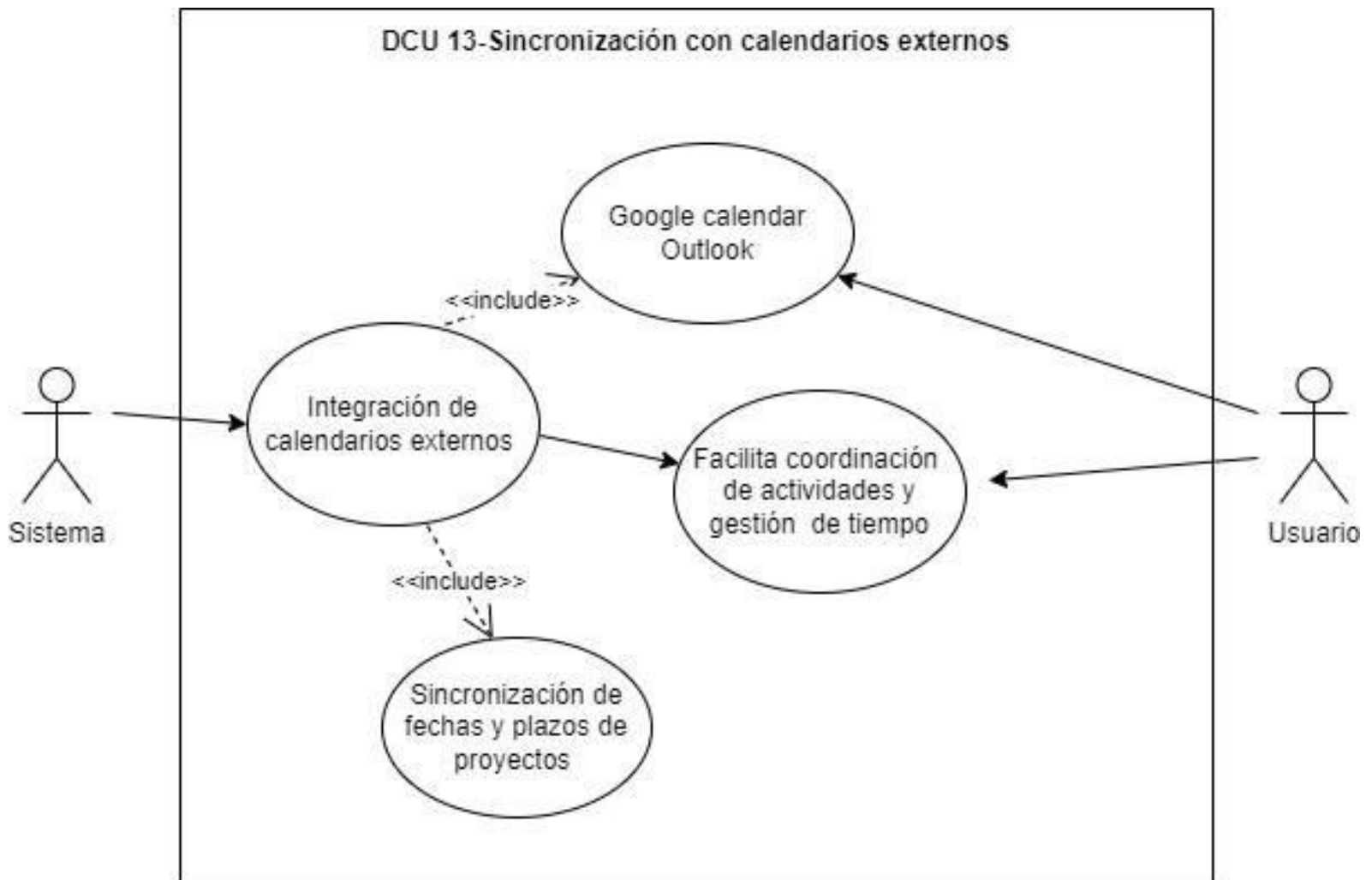
Diagrama 12.



Sincronización con Calendarios Externos:

Nombre:	Sincronización con Calendarios Externos
Campo de prioridad:	Media
Descripción: El sistema debe integrarse con calendarios externos como Google Calendar o Outlook, permitiendo la sincronización de fechas y plazos de proyectos.	
Actores: Usuario.	
Precondiciones: El usuario tiene una cuenta en un servicio de calendario externo (Google Calendar, Outlook). El usuario ha otorgado los permisos necesarios al sistema para acceder a su calendario externo.	
Flujo Normal: El usuario selecciona la opción para conectar un calendario externo. El sistema redirige al usuario al servicio de calendario externo para autenticarse. Una vez autenticado, el usuario autoriza al sistema a acceder a su calendario. El sistema establece una conexión entre el calendario interno y el calendario externo. Los eventos del proyecto se sincronizan automáticamente con el calendario externo.	
Flujo alternativo: El usuario cancela la autorización: El proceso de sincronización se detiene y no se establece conexión. Ocurre un error de autenticación: El sistema muestra un mensaje de error indicando que la autenticación ha fallado. Hay un conflicto de datos entre los calendarios: El sistema prioriza los datos del calendario interno o muestra un mensaje de alerta al usuario.	
Postcondiciones: Los eventos del proyecto se reflejan en el calendario externo del usuario y viceversa. El usuario puede ver y gestionar sus eventos desde ambas plataformas.	

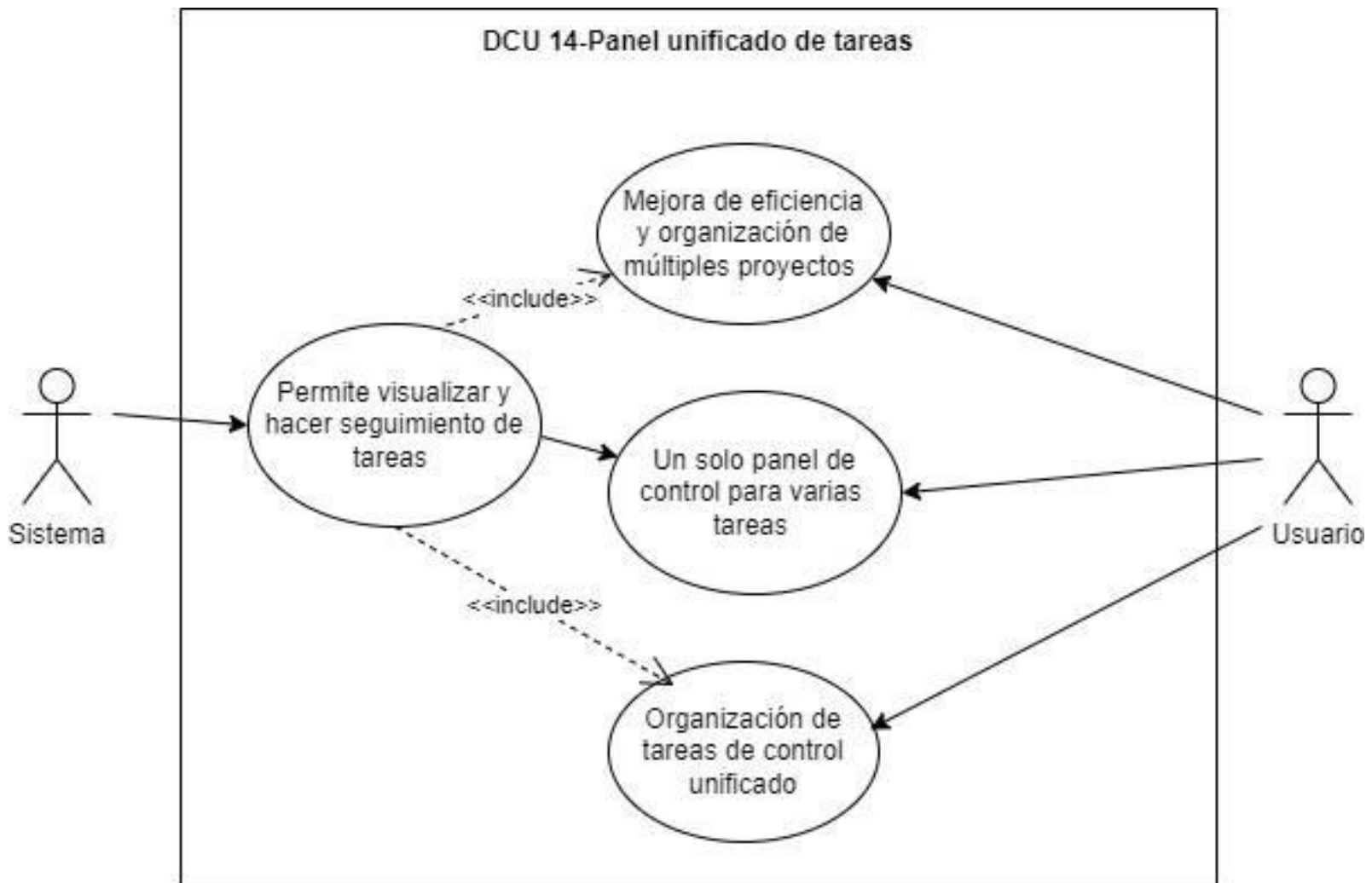
Diagrama 13.



Panel unificado de tareas múltiples:

Nombre:	Panel unificado de tareas múltiples
Campo de prioridad:	Alta
Descripción: El sistema debe permitir a los usuarios visualizar y hacer seguimiento de tareas distribuidas en varios proyectos desde un solo panel de control.	
Actores: Usuario.	
Precondiciones: El usuario tiene al menos dos proyectos asignados. Las tareas de los proyectos están debidamente registradas en el sistema.	
Flujo Normal: El usuario accede al panel de control unificado. El sistema consulta la base de datos y recupera todas las tareas asignadas al usuario, independientemente del proyecto. El sistema presenta las tareas en una vista organizada, mostrando información relevante como: Título de la tarea, Proyecto al que pertenece, Estado de la tarea, Fecha de vencimiento y Prioridad El usuario puede filtrar y ordenar las tareas según diferentes criterios (proyecto, estado, fecha de vencimiento)	
Flujo alternativo: El usuario no tiene tareas asignadas: El sistema muestra un mensaje indicando que no hay tareas disponibles. Ocurre un error al recuperar las tareas: El sistema muestra un mensaje de error y registra el incidente.	
Postcondiciones: El usuario puede visualizar y gestionar sus tareas de manera eficiente desde un único punto de acceso	

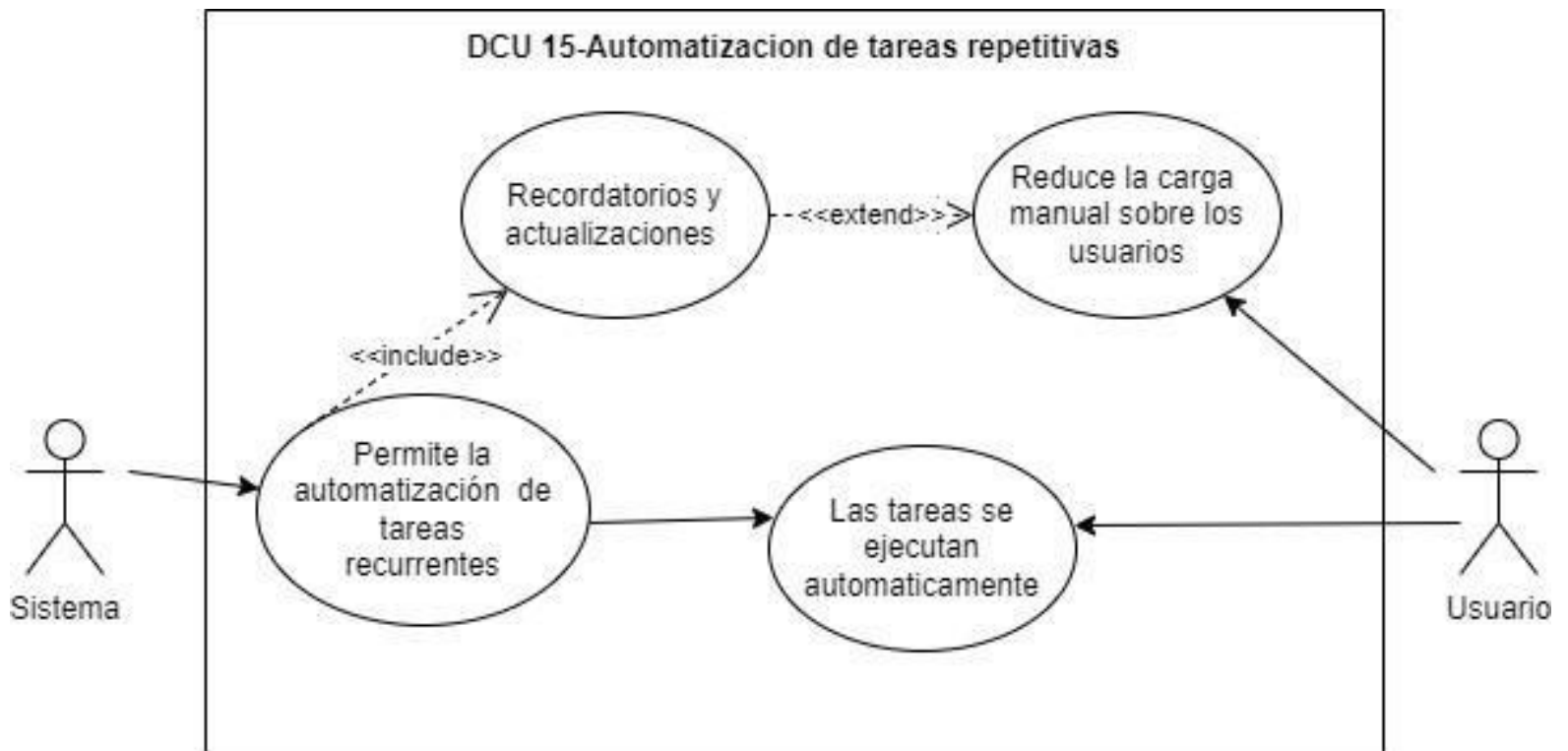
Diagrama 14.



Automatización de Tareas Repetitivas:

Nombre:	Automatización de Tareas Repetitivas
Campo de prioridad:	Media
Descripción:	El sistema debe permitir la automatización de tareas recurrentes, como recordatorios o actualizaciones de estado, para reducir la carga manual sobre los usuarios.
Actores:	Usuario.
Precondiciones:	El usuario ha definido una tarea como recurrente y ha establecido los parámetros de recurrencia (frecuencia, día de la semana)
Flujo Normal:	El sistema verifica constantemente si hay tareas programadas para ejecutarse. Si encuentra una tarea cuya condición de recurrencia se cumple, la ejecuta. El sistema registra el resultado de la ejecución de la tarea.
Flujo alternativo:	<p>Ocurre un error durante la ejecución de la tarea: El sistema registra el error y notifica al usuario.</p> <p>La tarea depende de datos externos que no están disponibles: El sistema intenta volver a ejecutar la tarea más tarde.</p>
Postcondiciones:	La tarea se ejecuta automáticamente según la programación establecida. El sistema registra el historial de ejecución de las tareas.

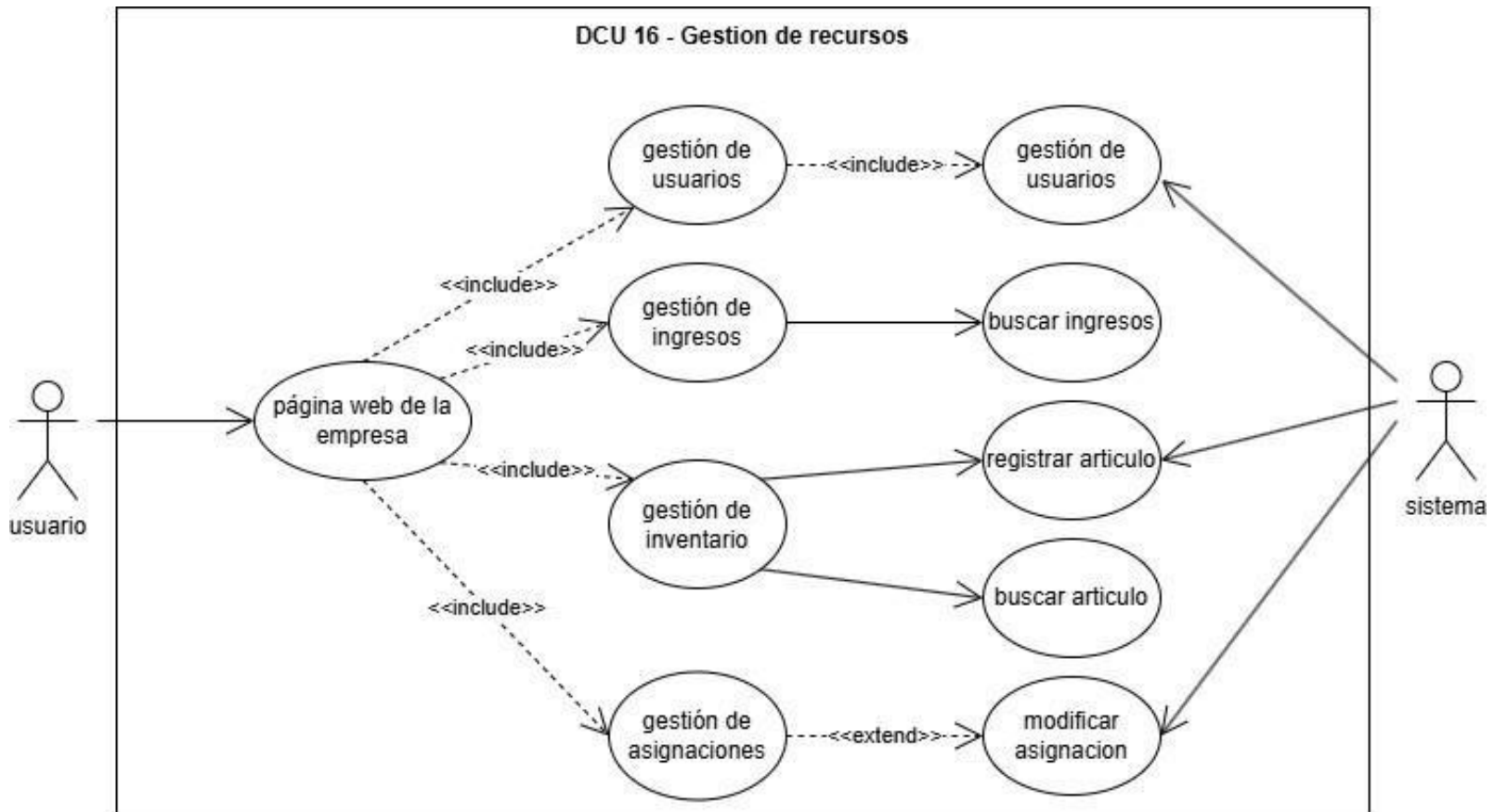
Diagrama 15.



Gestión de recursos:

Nombre:	Gestión de recursos
Campo de prioridad:	Alta
Descripción:	La interfaz mostrara la asignación de recursos humanos, financieros y materiales a las tareas y proyectos
Actores:	Usuario, sistema.
Precondiciones:	El usuario debe de tener los permisos necesarios para ingresar a la base de datos y a todos los recursos de la empresa
Flujo Normal:	<p>El usuario accede a la pantalla de inicio de sesión</p> <p>El usuario ingresa al sistema</p> <p>El usuario decide que recurso quiere administrar (ingresos, inventario, asignaciones)</p> <p>El usuario si lo desea modifica o añade más recursos a la base de datos</p>
Flujo alternativo:	<p>El usuario accede a la pantalla de inicio de sesión El usuario ingresa al sistema</p> <p>Si el usuario no puede ingresar al apartado de gestión de recursos porque no cuenta con dicho cargo se le indicara esto con una notificación y se le enviara de nuevo al inicio de la pagina</p>
Postcondiciones:	<p>Garantizar que el usuario haya sido autenticado correctamente, lo que asegura que solo los usuarios autorizados tengan acceso al sistema.</p> <p>Si se realiza algún cambio se registrará en la base de datos y se notificará a los usuarios pertinentes</p>

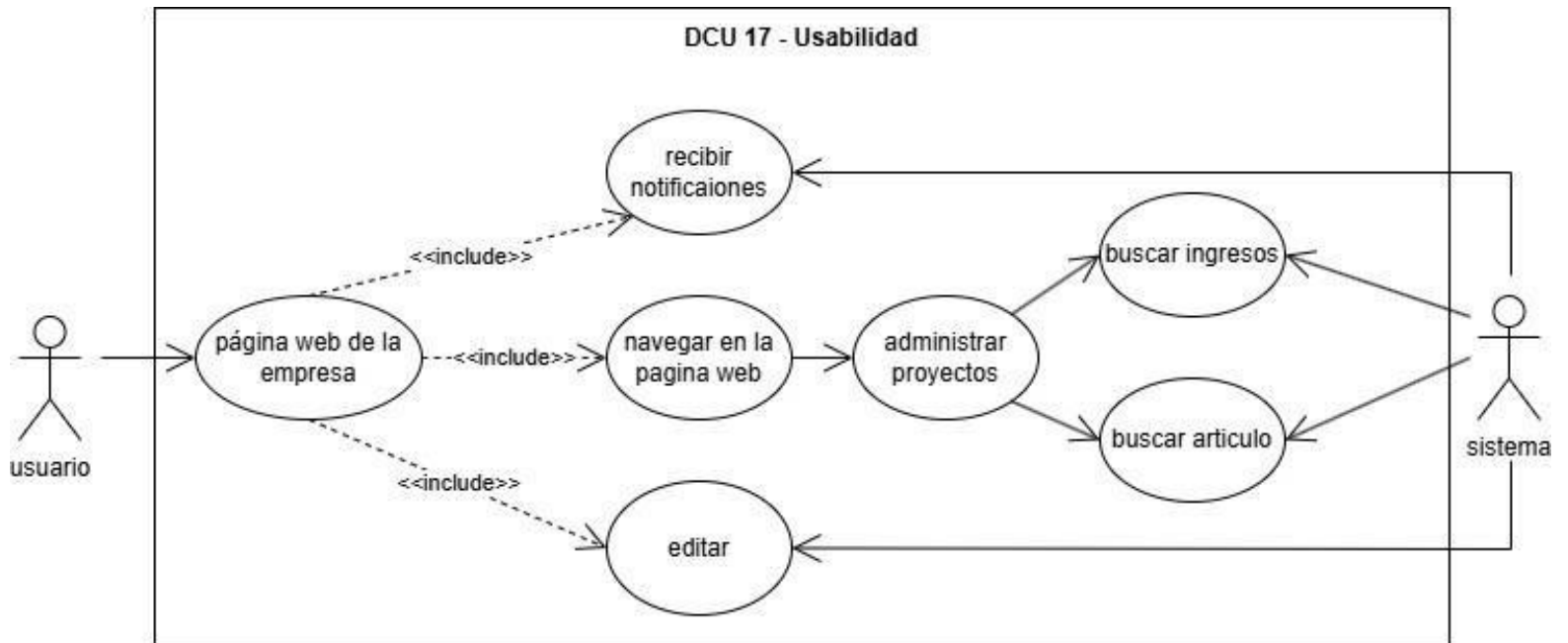
Diagrama 16.



Usabilidad:

Nombre:	Usabilidad
Campo de prioridad:	Media
Descripción: La interfaz debe ser intuitiva y fácil de usar para usuarios de diferentes niveles de habilidad.	
Actores: Usuario, sistema.	
Precondiciones: El usuario debe de tener un usuario y una contraseña en la base de datos para poder ingresar. Sin estos datos previamente establecidos no se dará acceso.	
Flujo Normal: La página carga rápidamente y el usuario puede comenzar a interactuar sin demoras El usuario encuentra un menú claro y bien organizado que le permite localizar fácilmente la sección que necesita Después de realizar una acción, el usuario recibe una confirmación clara de que su solicitud ha sido procesada correctamente	
Flujo alternativo: El usuario accede a la pantalla de inicio de sesión El usuario ingresa al sistema Si el usuario tiene algún problema con el sistema hay opciones de soporte fácilmente accesibles, como chat en vivo, FAQ o contacto por correo electrónico	
Postcondiciones: El usuario debe de tener una buena y fácil experiencia al utilizar el sistema Comunicar a soporte cualquier problema que tenga en cuanto a la usabilidad del sistema	

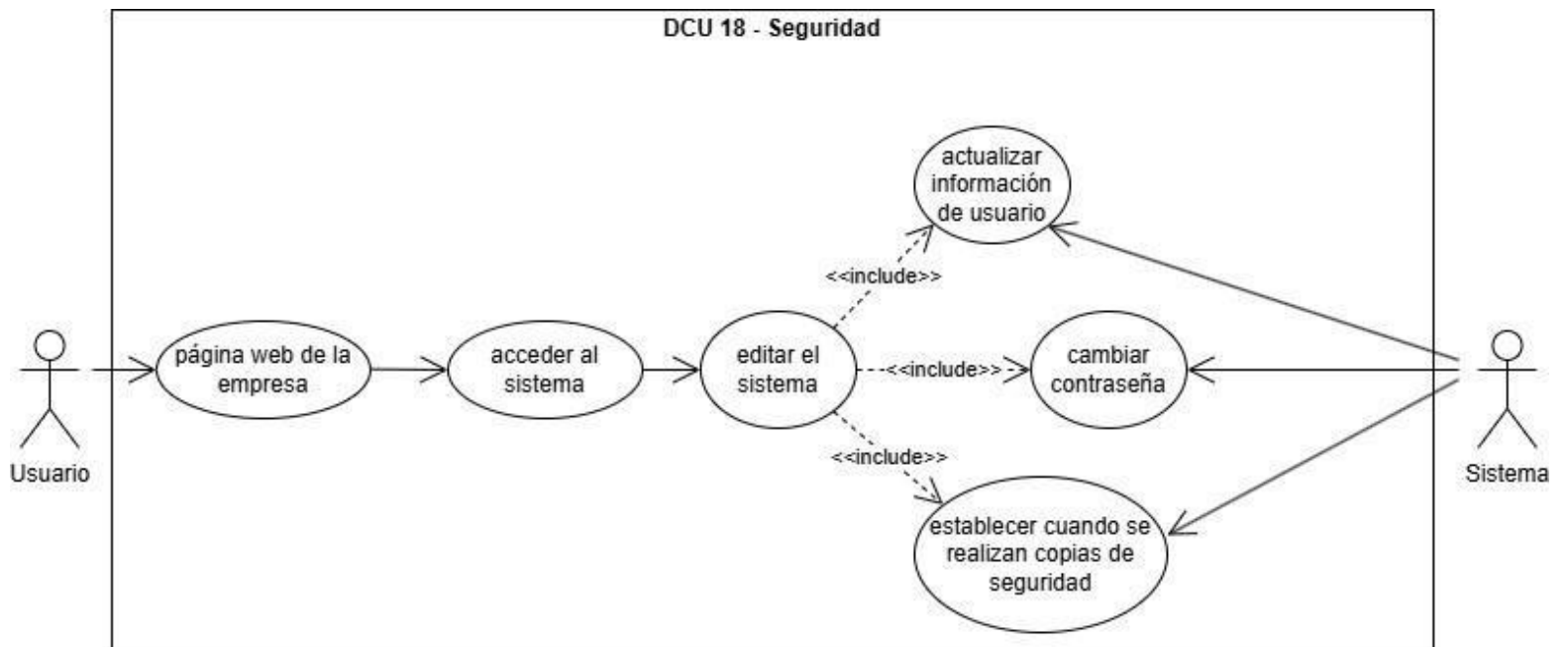
Diagrama 17.



Seguridad:

Nombre:	Seguridad
Campo de prioridad:	Alta
Descripción: El sistema debe implementar medidas de seguridad robustas para proteger los datos del proyecto y la privacidad de los usuarios.	
Actores: Usuario, sistema.	
Precondiciones: El usuario debe de tener un usuario y una contraseña en la base de datos para poder ingresar. Si es la primera vez que el usuario accede desde un dispositivo nuevo, se le puede pedir que verifique su identidad a través de un código enviado a su correo electrónico o teléfono	
Flujo Normal: El usuario ingresa sus credenciales en una página de inicio de sesión protegida El usuario accede a los recursos o servicios que necesita, con permisos y roles claramente definidos para limitar el acceso a información sensible El usuario cierra sesión de manera segura, asegurándose de que su sesión se termine correctamente para evitar accesos no autorizados El sistema monitorea continuamente la actividad para detectar y responder a posibles amenazas en tiempo real	
Flujo alternativo: Si el usuario no tiene la cuenta ni las credenciales necesarias no podrá ingresar al sistema ni tampoco a los recursos que necesita debido a la falta de permisos	
Postcondiciones: Si el usuario necesita actualizar información personal, el sistema puede requerir una verificación adicional para confirmar la identidad del usuario	

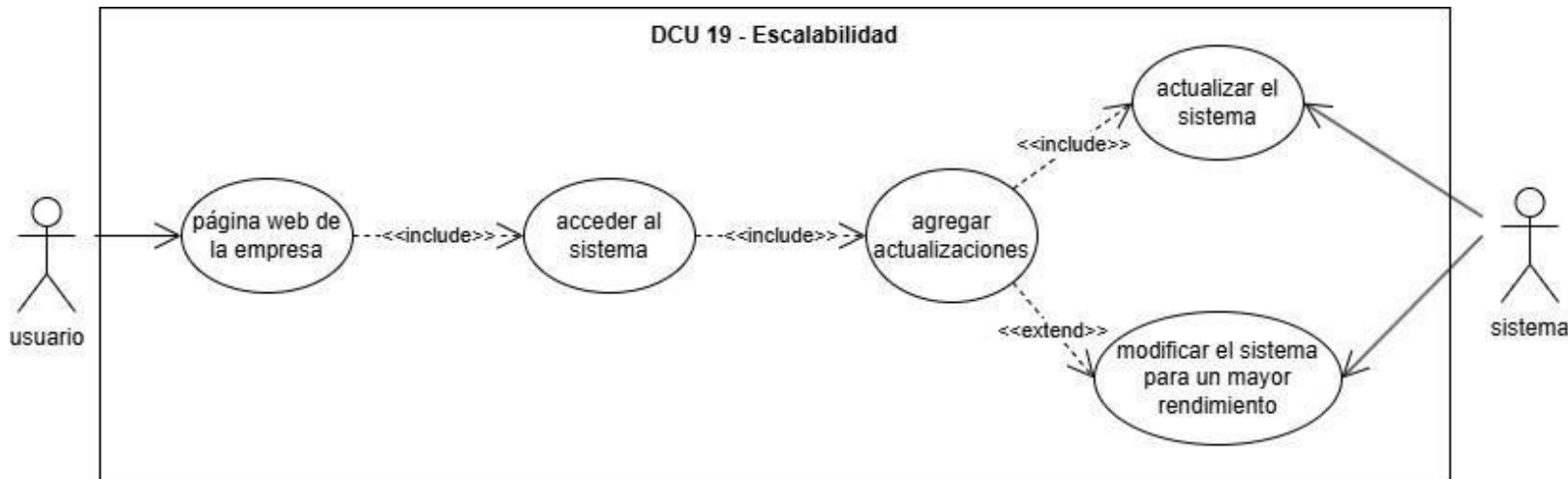
Diagrama 18.



Escalabilidad:

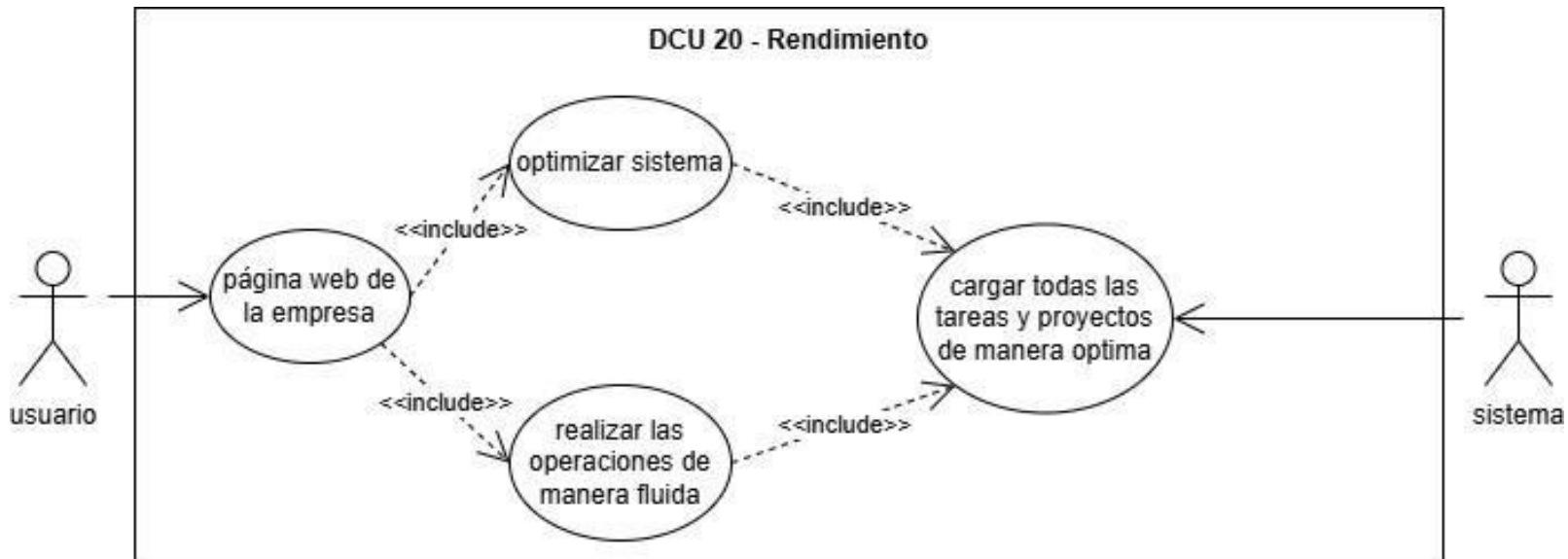
Nombre:	Escalabilidad
Campo de prioridad:	Alta
Descripción: El sistema debe ser capaz de manejar un gran número de proyectos y usuarios simultáneamente sin degradación del rendimiento.	
Actores: Usuario, sistema.	
Precondiciones: El sistema debe tener sus debidas actualizaciones y realizarlas de la mejor manera para el uso de los usuarios	
Flujo Normal: El usuario ingresa sus credenciales para acceder a la página El usuario puede acceder a nuevas funciones o actualizaciones sin interrupciones, gracias a un sistema de despliegue continuo Las actualizaciones del sistema se realizan en segundo plano, sin afectar la experiencia del usuario El usuario recibe notificaciones claras sobre nuevas funciones o mejoras El usuario cierra sesión de manera segura, con la opción de recordar su información para futuros inicios de sesión	
Flujo alternativo: El usuario ingresa sus credenciales para acceder a la página El usuario inicia la actualización del sistema y espera a que dicha actualización o cambios finalicen por completo antes de iniciar con su trabajo	
Postcondiciones: El usuario debe de asegurarse de que el sistema este actualizado y optimizado antes de volver a ingresar a la plataforma	

Diagrama 19.

**Rendimiento:**

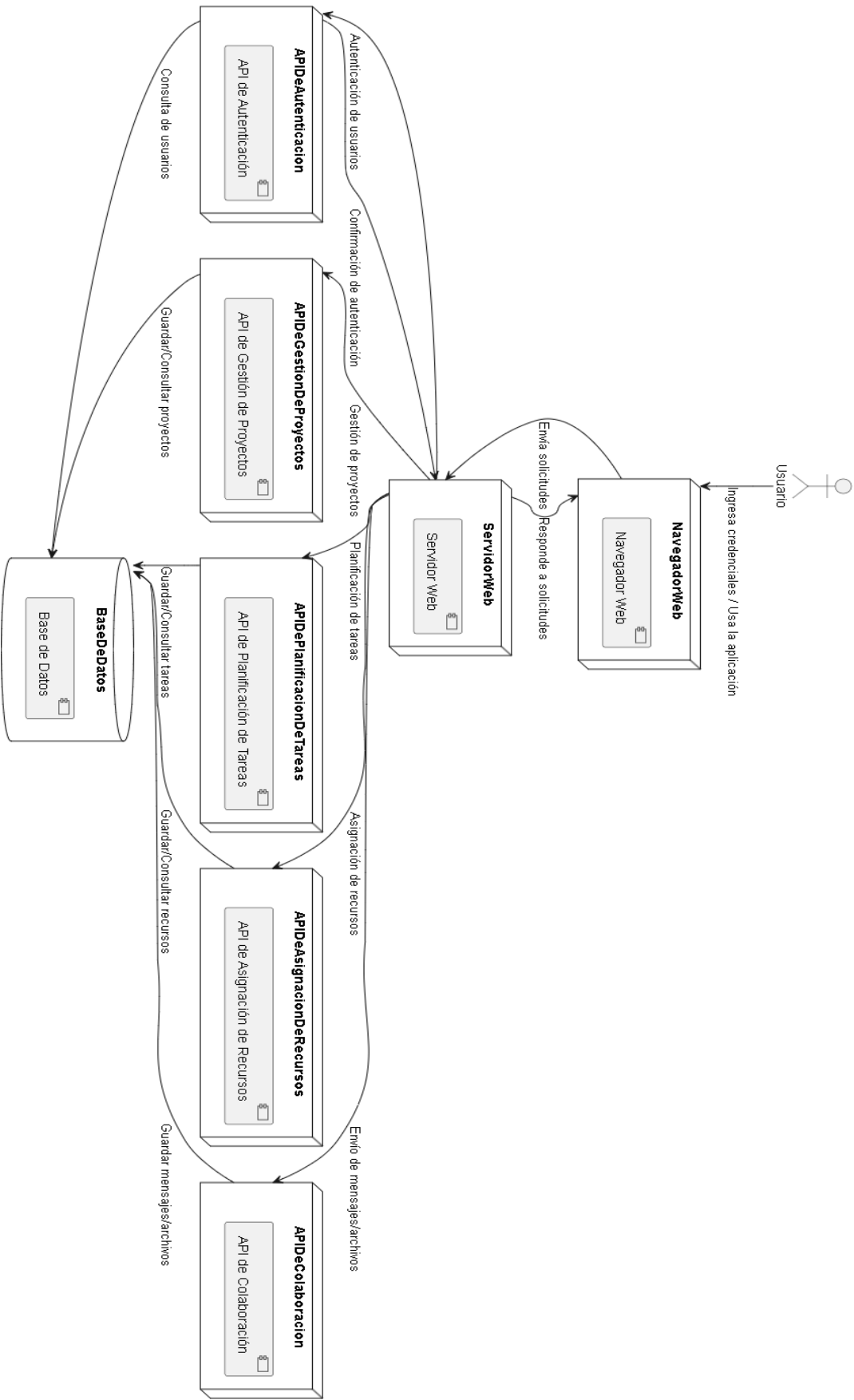
Nombre:	Rendimiento
Campo de prioridad:	Alta
Descripción:	Las operaciones del sistema, como la carga de proyectos y tareas, deben completarse en menos de 2 segundos en promedio
Actores:	Usuario, sistema.
Precondiciones:	El sistema debe estar bien optimizado para que funcione de manera eficiente y con fluidez para la buena experiencia de los usuarios
Flujo Normal:	<p>El usuario ingresa sus credenciales para acceder a la página</p> <p>La página debe cargar en pocos segundos para contribuir con la eficiencia en el trabajo del usuario</p> <p>El sistema funciona sin complicaciones y con respuestas rápidas a las acciones del usuario.</p>
Flujo alternativo:	<p>La página carga en pocos segundos</p> <p>En lugar de usar el menú principal, el usuario puede utilizar un buscador interno para encontrar rápidamente lo que necesita</p> <p>El usuario puede interactuar con banners o recomendaciones personalizadas que lo llevan directamente a la sección de interés</p>
Postcondiciones:	El usuario debe asegurarse de que el sistema tenga un buen rendimiento y optimización

Diagrama 20.



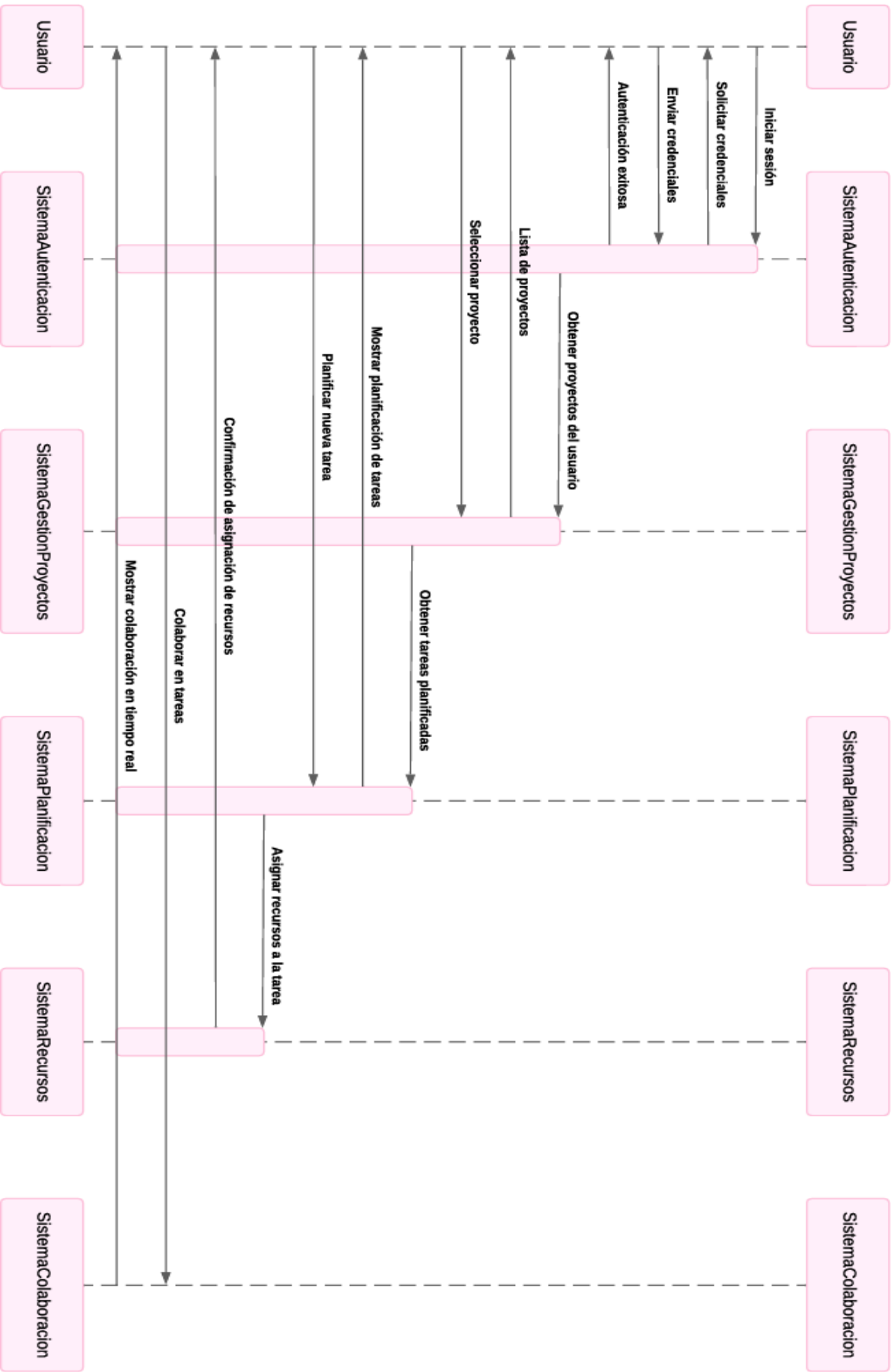
1.11 DIAGRAMA DE DISTRIBUCION

Diagrama 21.



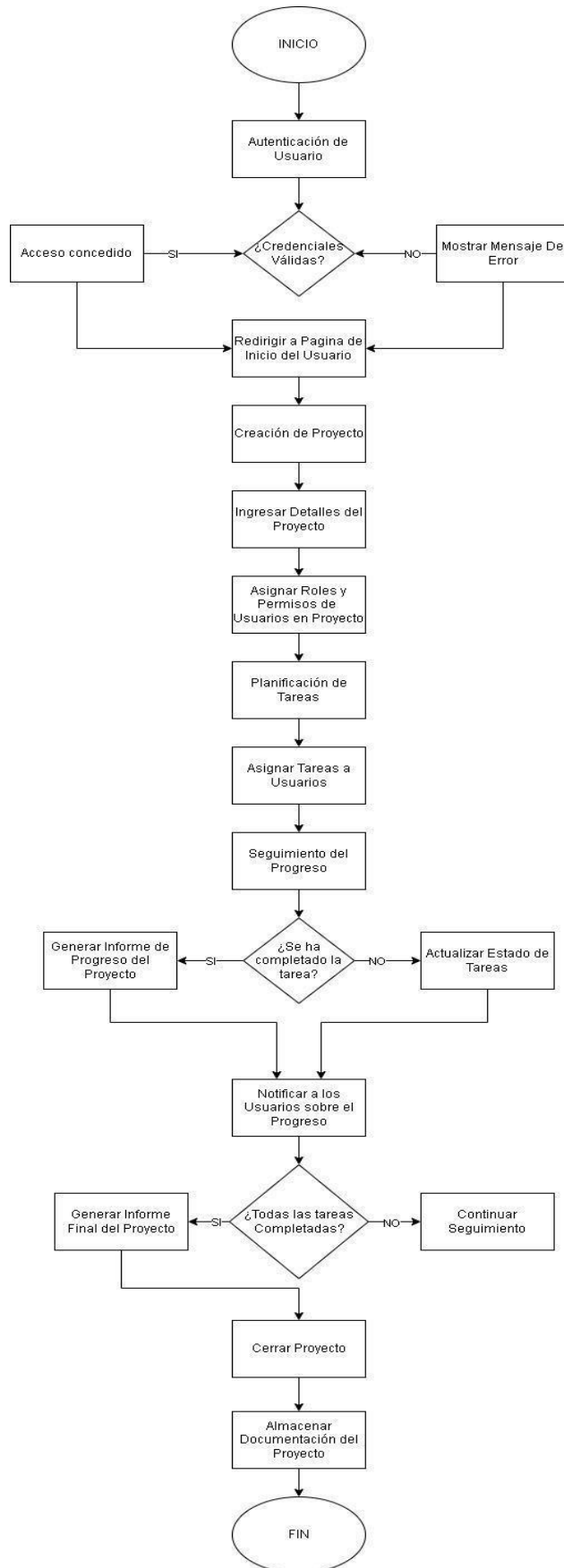
1.12 DIAGRAMA DE SECUENCIA

Diagrama 22.



1.13 DIAGRAMA DE FLUJO

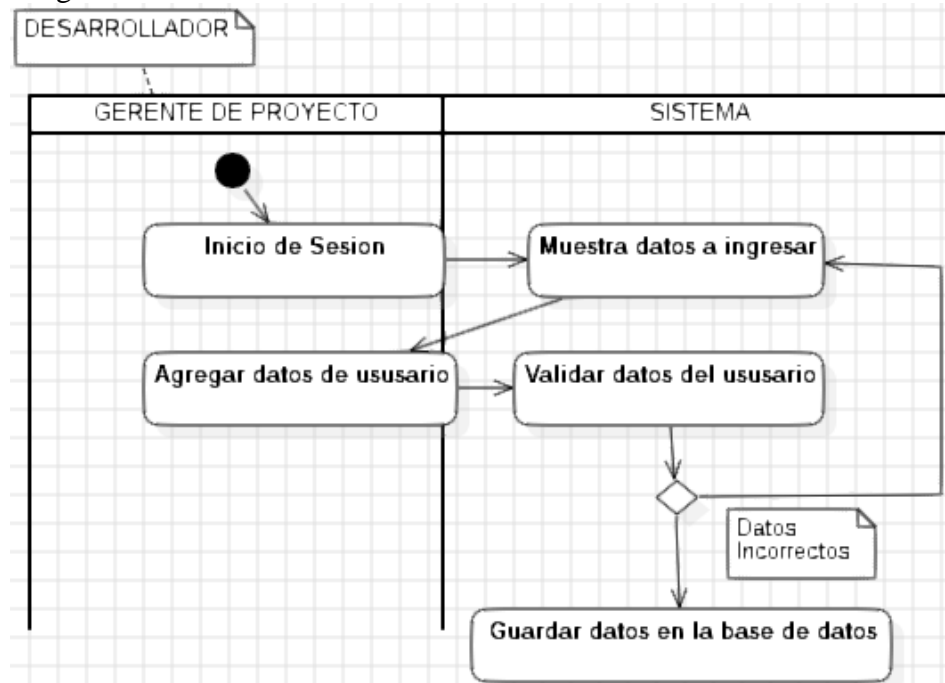
Diagrama 23.



1.14 DIAGRAMA DE ACTIVIDADES

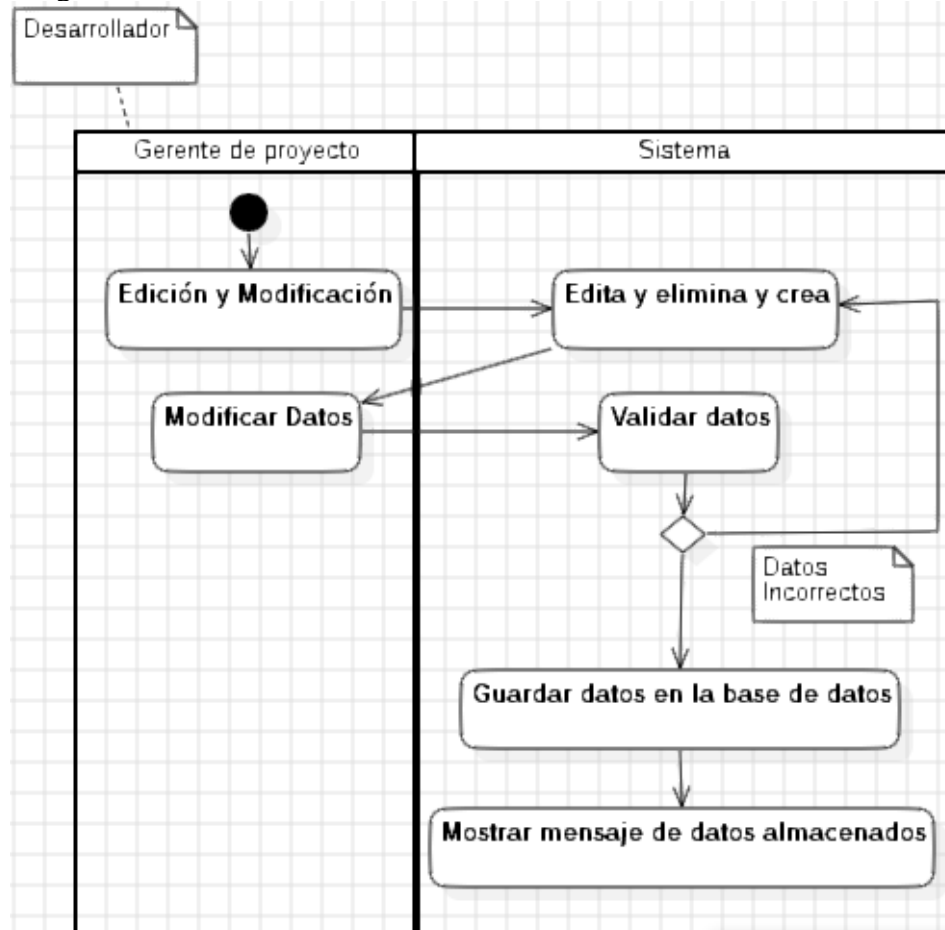
Inicio Sesión:

Diagrama 25.



Edición y modificación:

Diagrama 26.



1.15 EXPLICACION DEL MODELO DE LA BD, TABLAS Y ESTTRUCTURA

El modelo de base de datos (BD) para este proyecto se basa en una arquitectura relacional que permita gestionar de manera eficiente los proyectos, las tareas y la relación entre usuarios, roles, y permisos. Las tablas clave incluyen:

Usuarios: Guarda información de los usuarios del sistema, como gerentes y desarrolladores.

Proyectos: Contiene los detalles de los proyectos, como el nombre, descripción y fechas.

Tareas: Almacena las tareas de los proyectos, asignadas a los usuarios.

Roles: Define los roles dentro del sistema, como gerente de proyecto o desarrollador.

Permisos: Gestiona los permisos que se otorgan a los usuarios de acuerdo con sus roles.

Historial de Actividades: Registra todas las acciones que los usuarios realizan en los proyectos para tener un seguimiento completo.

Cada tabla tiene relaciones de cardinalidad específicas para modelar correctamente las interacciones y el flujo de información entre usuarios, proyectos y tareas.

Estructura y campos clave

Usuarios (ID Usuario, Nombre, Correo, Contraseña, ID Rol)

Proyectos (ID Proyecto, Nombre Proyecto, Descripción, Fecha Inicio, Fecha Fin, ID Gerente)

Tareas (ID Tarea, Nombre Tarea, Descripción Tarea, Estado, ID Usuario, ID Proyecto, Fecha Asignación, Fecha Vencimiento)

Roles (ID Rol, Nombre Rol)

Permisos (ID Permiso, Nombre Permiso, ID Rol)

Historial de Actividades (ID Historial, ID Usuario, ID Tarea, Acción, Fecha Acción)

Deficiencias de información del proyecto para generar la estructura y la BD

Para completar el diseño de la base de datos, faltan los siguientes detalles:

Requisitos adicionales: No se han definido campos específicos para informes, notificaciones y automatización de procesos, lo cual influye en la estructura de la BD.

Relaciones detalladas: Se necesita mayor claridad sobre cómo se relacionan las tareas entre proyectos y si se pueden compartir tareas entre proyectos.

Gestión de recursos: No está claro si el sistema debe administrar recursos financieros o materiales, lo cual podría requerir tablas adicionales.

Cardinalidad

Usuarios a Roles: Muchos a uno. Un usuario tiene un solo rol, pero un rol puede estar asociado a muchos usuarios.

Proyectos a Usuarios (Gerentes): Muchos a uno. Cada proyecto tiene un solo gerente, pero un gerente puede gestionar varios proyectos.

Proyectos a Tareas: Uno a muchos. Cada proyecto puede tener varias tareas.

Tareas a Usuarios (Asignados): Muchos a uno. Cada tarea es asignada a un solo usuario, pero un usuario puede tener varias tareas.

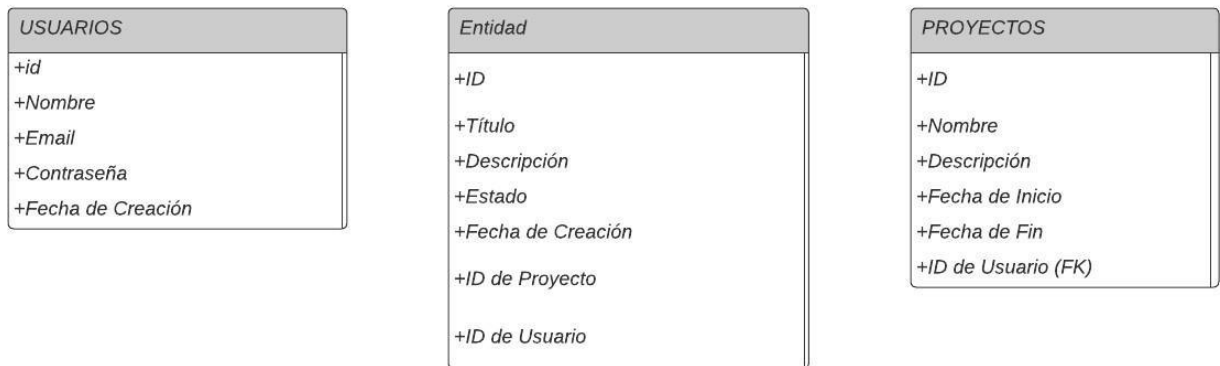
Permisos a Roles: Muchos a uno. Un rol puede tener múltiples permisos, pero un permiso puede aplicarse a varios roles.

Diagramas de BD

1.15.1 Modelo Conceptual:

Incluye las entidades principales (Usuarios, Proyectos, Tareas, Roles, Permisos, Historial) y sus relaciones.

MODELO CONCEPTUAL

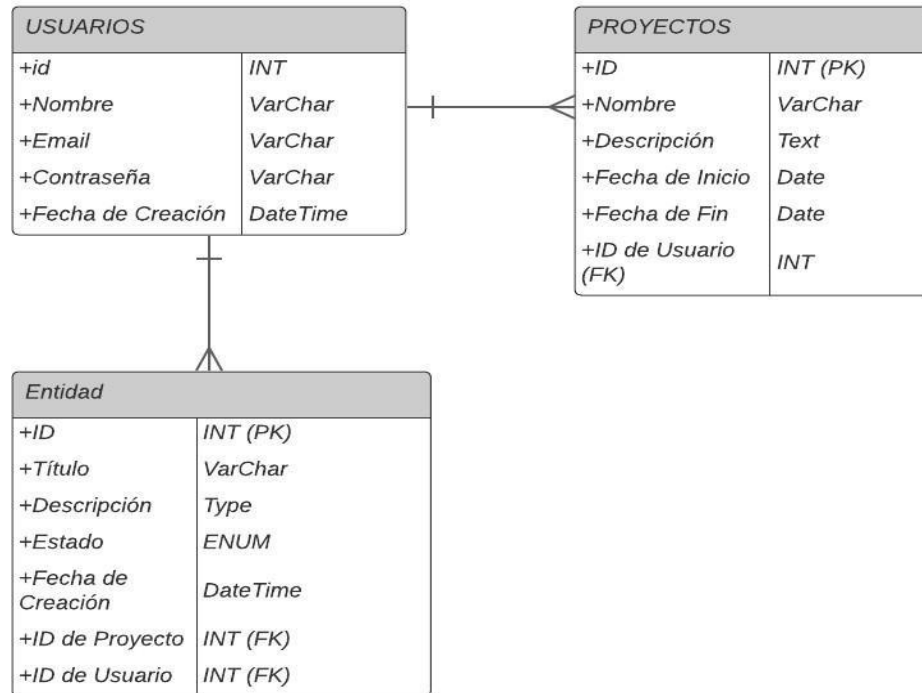


1.15.2 Modelo Lógico:

Representa las relaciones y los campos principales en cada entidad.

Relaciones entre las tablas, tipos de datos, claves primarias y claves foráneas.

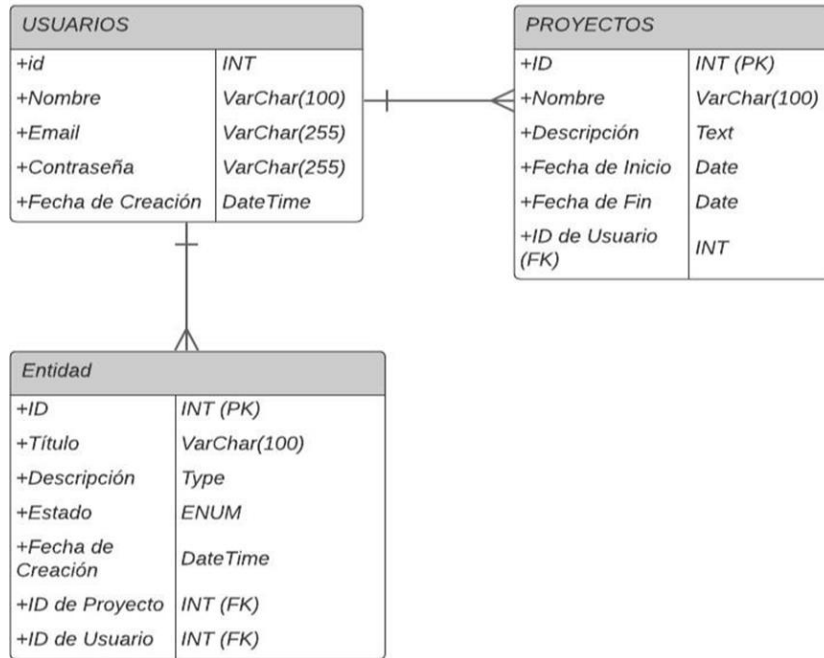
MODELO LÓGICO



1.15.3 Modelo Físico:

Define las tablas con todos los detalles técnicos: tipos de datos específicos (VARCHAR, DATE, etc.), restricciones de integridad, índices, y relaciones entre las tablas a nivel físico.

MODELO FÍSICO



Script SQL de la base de datos del proyecto:

-- Creación de la base de datos

```
CREATE DATABASE IF NOT EXISTS ProyectoGestion;USE
ProyectoGestion;
```

-- Tabla: Usuarios

```
CREATE TABLE IF NOT EXISTS Usuarios ( ID INT AUTO_INCREMENT
PRIMARY KEY,Nombre VARCHAR(100) NOT NULL,
Email VARCHAR(255) NOT NULL UNIQUE,Contraseña VARCHAR(255)
NOT NULL,
FechaCreacion DATETIME DEFAULT CURRENT_TIMESTAMP
);
```

-- Tabla: Proyectos

```
CREATE TABLE IF NOT EXISTS Proyectos ( ID INT AUTO_INCREMENT
PRIMARY KEY,Nombre VARCHAR(100) NOT NULL,
Descripcion TEXT,
FechaInicio DATE NOT NULL,FechaFin DATE NOT NULL, IDUsuario INT,
FOREIGN KEY (IDUsuario) REFERENCES Usuarios(ID) ON
DELETE CASCADE
);
```

-- Tabla: Tareas

```
CREATE TABLE IF NOT EXISTS Tareas (
ID INT AUTO_INCREMENT PRIMARY KEY,Titulo VARCHAR(100) NOT
NULL,
Descripcion TEXT,
Estado ENUM('Pendiente', 'En Progreso', 'Completo') NOT NULL,
FechaCreacion DATETIME DEFAULT CURRENT_TIMESTAMP,
IDProyecto INT,IDUsuario INT,
FOREIGN KEY (IDProyecto) REFERENCES Proyectos(ID)
ONDELETE CASCADE,
FOREIGN KEY (IDUsuario) REFERENCES Usuarios(ID) ON
DELETE CASCADE
);
```

Explicación del Script:

Creación de la base de datos: El script inicia creando la base de datos

ProyectoGestionsi no existe y luego selecciona esa base de datos

para realizar las siguientes operaciones.

Tabla Usuarios:

ID: Identificador único de cada usuario, auto-incremental.

Nombre, Email, Contraseña: Campos para almacenar la información del usuario.

FechaCreacion: Se establece automáticamente en la fecha y hora actual es al momento de la creación del registro.

Tabla Proyectos:

ID: Identificador único del proyecto.

Nombre, Descripción, FechaInicio, FechaFin: Campos que almacenan los detalles del proyecto.

IDUsuario: Clave foránea que hace referencia al ID de la tabla Usuarios.

Tabla Tareas:

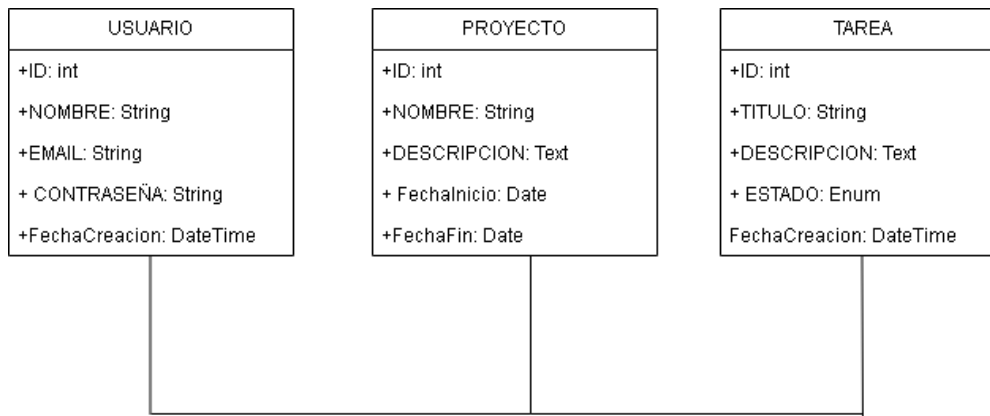
ID: Identificador único de la tarea.

Título, Descripción, Estado: Campos que almacenan los detalles de la tarea.

FechaCreacion: Se establece automáticamente en la fecha y hora actuales.

IDProyecto, IDUsuario: Claves foráneas que hacen referencia a las tablas Proyectos y Usuarios, respectivamente.

1.16 MAPA DE NAVEGACION



Lenguajes Utilizados y Justificación

HTML (HyperText Markup Language):

Descripción: HTML es el lenguaje base del web utilizado para estructurar el contenido de las páginas web.

Justificación: HTML permite estructurar de manera clara el contenido, con secciones definidas para elementos como el inicio de sesión, el registro, la lista de tareas, y otras características del proyecto. Su simplicidad, compatibilidad y estandarización en todos los navegadores lo hacen esencial.

CSS (Cascading Style Sheets):

Descripción: CSS permite definir el diseño y la apariencia visual de los elementos de HTML.

Justificación: CSS fue seleccionado para brindar una interfaz visualmente atractiva y profesional que se ajuste a una experiencia de usuario intuitiva. Además, CSS permite la adaptación de la página en diferentes dispositivos, proporcionando una experiencia de diseño responsivo.

JavaScript

Descripción: JavaScript es un lenguaje de programación que permite crear interactividad y dinamismo en la web.

Justificación: Es un lenguaje clave para el desarrollo de aplicaciones web dinámicas y se utiliza ampliamente para gestionar interacciones en la interfaz de usuario. En este proyecto, JavaScript será fundamental para validar formularios, manejar el flujo de la interfaz y permitir acciones como la creación y modificación de tareas sin necesidad de recargar la página.

SQL (Structured Query Language):

Descripción: SQL es el lenguaje estándar para el manejo de bases de datos relacionales.

Justificación: SQL permite gestionar de manera eficiente la base de datos del sistema, que contiene la información de los usuarios, proyectos y tareas. Su capacidad de realizar operaciones de consulta complejas, junto con su fiabilidad y velocidad en entornos de producción, lo hace la elección más adecuada para esta plataforma.

Estructura del Proyecto

Frontend (Interfaz de usuario): Construido con HTML y CSS, permite a los usuarios interactuar con el sistema y realizar acciones de forma intuitiva. JavaScript se usa aquí para controlar la lógica en el cliente, manejando eventos como el inicio de sesión y la creación de nuevas tareas.

Backend y Base de Datos:

El backend emplea SQL para las operaciones de base de datos. Las interacciones entre la interfaz y la base de datos se manejan de manera segura para garantizar que los datos de los usuarios y las tareas se procesen y almacenen de forma eficiente.