

Documento de Propuesta de Diseño de Software I, II y II

Creación de módulo de gestión de tareas (Task Manager)

Alexander Enrique Toscano Ricardo



Twitter: @kikret

Github: @atoscano

Edwin Bertel Negrette

Guillermo del Valle Vitola

German Rivera Rosario

Descripción del software

El módulo de gestión de tareas está diseñado para simplificar la organización y el seguimiento de las actividades diarias en plataformas de contenido. La solución ofrece una interfaz intuitiva y múltiples funcionalidades que permiten la autogestión y distribución eficiente de tareas, con monitoreo en tiempo real del estado de cada proyecto. Se enfoca en mejorar la comunicación y coordinación dentro de los equipos, optimizando el rendimiento y asegurando una mayor calidad en la gestión de contenidos a través de diferentes plataformas.

ETAPA 1 DISEÑO DE LA APLICACIÓN Y ANÁLISIS DE REQUISITOS	6
1. INTRODUCCIÓN	6
PROPÓSITO DEL DOCUMENTO	6
ALCANCE DEL PROYECTO MÓDULO DE PIZARRA COMPARTIDA	8
DEFINICIONES Y ACRÓNIMOS	¡ERROR! MARCADOR NO DEFINIDO.
2. DESCRIPCIÓN GENERAL	9
OBJETIVOS DEL SISTEMA	9
FUNCIONALIDAD GENERAL	9
USUARIOS DEL SISTEMA	10
RESTRICCIONES	¡ERROR! MARCADOR NO DEFINIDO.
3. REQUISITOS FUNCIONALES	¡ERROR! MARCADOR NO DEFINIDO.
CASOS DE USO	¡ERROR! MARCADOR NO DEFINIDO.
DIAGRAMAS DE FLUJO DE CASOS DE USO	¡ERROR! MARCADOR NO DEFINIDO.
DESCRIPCIÓN DETALLADA DE CADA CASO DE USO	¡ERROR! MARCADOR NO DEFINIDO.
PRIORIDAD DE REQUERIMIENTOS	¡ERROR! MARCADOR NO DEFINIDO.
4. REQUISITOS NO FUNCIONALES	¡ERROR! MARCADOR NO DEFINIDO.
REQUISITOS DE DESEMPEÑO	¡ERROR! MARCADOR NO DEFINIDO.
REQUISITOS DE SEGURIDAD	¡ERROR! MARCADOR NO DEFINIDO.
REQUISITOS DE USABILIDAD	¡ERROR! MARCADOR NO DEFINIDO.
REQUISITOS DE ESCALABILIDAD	¡ERROR! MARCADOR NO DEFINIDO.
5. MODELADO E/R	¡ERROR! MARCADOR NO DEFINIDO.
DIAGRAMA DE ENTIDAD-RELACIÓN	¡ERROR! MARCADOR NO DEFINIDO.
DIAGRAMA RELACIONAL	¡ERROR! MARCADOR NO DEFINIDO.
SCRIPT DE MODELO RELACIONAL	¡ERROR! MARCADOR NO DEFINIDO.
DESCRIPCIÓN DE ENTIDADES Y RELACIONES	¡ERROR! MARCADOR NO DEFINIDO.
REGLAS DE INTEGRIDAD REFERENCIAL	¡ERROR! MARCADOR NO DEFINIDO.
COLECCIONES (NoSQL)	¡ERROR! MARCADOR NO DEFINIDO.
6. ANEXOS	¡ERROR! MARCADOR NO DEFINIDO.
DIAGRAMAS ADICIONALES	¡ERROR! MARCADOR NO DEFINIDO.
REFERENCIAS	¡ERROR! MARCADOR NO DEFINIDO.
ETAPA 2: PERSISTENCIA DE DATOS CON BACKEND	¡ERROR! MARCADOR NO DEFINIDO.
7. INTRODUCCIÓN	¡ERROR! MARCADOR NO DEFINIDO.
PROPÓSITO DE LA ETAPA	¡ERROR! MARCADOR NO DEFINIDO.
ALCANCE DE LA ETAPA	¡ERROR! MARCADOR NO DEFINIDO.
DEFINICIONES Y ACRÓNIMOS	¡ERROR! MARCADOR NO DEFINIDO.
8. DISEÑO DE LA ARQUITECTURA DE BACKEND	¡ERROR! MARCADOR NO DEFINIDO.
DESCRIPCIÓN DE LA ARQUITECTURA PROPUESTA	¡ERROR! MARCADOR NO DEFINIDO.
COMPONENTES DEL BACKEND	¡ERROR! MARCADOR NO DEFINIDO.
DIAGRAMAS DE ARQUITECTURA	¡ERROR! MARCADOR NO DEFINIDO.
9. ELECCIÓN DE LA BASE DE DATOS	¡ERROR! MARCADOR NO DEFINIDO.

	EVALUACIÓN DE OPCIONES (SQL o NoSQL).....	
	JUSTIFICACIÓN DE LA ELECCIÓN	
	DISEÑO DE ESQUEMA DE BASE DE DATOS.....	
10.	IMPLEMENTACIÓN DEL BACKEND	¡ERROR! MARCADOR NO DEFINIDO.
	ELECCIÓN DEL LENGUAJE DE PROGRAMACIÓN.....	¡ERROR! MARCADOR NO DEFINIDO.
	CREACIÓN DE LA LÓGICA DE NEGOCIO.....	¡ERROR! MARCADOR NO DEFINIDO.
	DESARROLLO DE ENDPOINTS Y APIS.....	¡ERROR! MARCADOR NO DEFINIDO.
	AUTENTICACIÓN Y AUTORIZACIÓN	¡ERROR! MARCADOR NO DEFINIDO.
11.	CONEXIÓN A LA BASE DE DATOS	¡ERROR! MARCADOR NO DEFINIDO.
	CONFIGURACIÓN DE LA CONEXIÓN.....	¡ERROR! MARCADOR NO DEFINIDO.
	DESARROLLO DE OPERACIONES CRUD.....	¡ERROR! MARCADOR NO DEFINIDO.
	MANEJO DE TRANSACCIONES.....	¡ERROR! MARCADOR NO DEFINIDO.
12.	PRUEBAS DEL BACKEND.....	¡ERROR! MARCADOR NO DEFINIDO.
	DISEÑO DE CASOS DE PRUEBA.....	¡ERROR! MARCADOR NO DEFINIDO.
	EJECUCIÓN DE PRUEBAS UNITARIAS Y DE INTEGRACIÓN	¡ERROR! MARCADOR NO DEFINIDO.
	MANEJO DE ERRORES Y EXCEPCIONES	¡ERROR! MARCADOR NO DEFINIDO.
	ETAPA 3: CONSUMO DE DATOS Y DESARROLLO FRONTEND	¡ERROR! MARCADOR NO DEFINIDO.
13.	INTRODUCCIÓN	¡ERROR! MARCADOR NO DEFINIDO.
	PROPÓSITO DE LA ETAPA.....	¡ERROR! MARCADOR NO DEFINIDO.
	ALCANCE DE LA ETAPA	¡ERROR! MARCADOR NO DEFINIDO.
	DEFINICIONES Y ACRÓNIMOS.....	¡ERROR! MARCADOR NO DEFINIDO.
14.	CREACIÓN DE LA INTERFAZ DE USUARIO (UI)	¡ERROR! MARCADOR NO DEFINIDO.
	DISEÑO DE LA INTERFAZ DE USUARIO (UI) CON HTML Y CSS	¡ERROR! MARCADOR NO DEFINIDO.
	CONSIDERACIONES DE USABILIDAD	¡ERROR! MARCADOR NO DEFINIDO.
	MAQUETACIÓN RESPONSIVA	¡ERROR! MARCADOR NO DEFINIDO.
15.	PROGRAMACIÓN FRONTEND CON JAVASCRIPT (JS).....	¡ERROR! MARCADOR NO DEFINIDO.
	DESARROLLO DE LA LÓGICA DEL FRONTEND.....	¡ERROR! MARCADOR NO DEFINIDO.
	MANEJO DE EVENTOS Y COMPORTAMIENTOS DINÁMICOS.....	¡ERROR! MARCADOR NO DEFINIDO.
	USO DE BIBLIOTECAS Y FRAMEWORKS (SI APLICABLE)	¡ERROR! MARCADOR NO DEFINIDO.
16.	CONSUMO DE DATOS DESDE EL BACKEND	¡ERROR! MARCADOR NO DEFINIDO.
	CONFIGURACIÓN DE CONEXIONES AL BACKEND.....	¡ERROR! MARCADOR NO DEFINIDO.
	OBTENCIÓN Y PRESENTACIÓN DE DATOS	¡ERROR! MARCADOR NO DEFINIDO.
	ACTUALIZACIÓN EN TIEMPO REAL (SI APLICABLE).....	¡ERROR! MARCADOR NO DEFINIDO.
17.	INTERACCIÓN USUARIO-INTERFAZ	¡ERROR! MARCADOR NO DEFINIDO.
	MANEJO DE FORMULARIOS Y VALIDACIÓN DE DATOS.....	¡ERROR! MARCADOR NO DEFINIDO.
	IMPLEMENTACIÓN DE FUNCIONALIDADES INTERACTIVAS.....	¡ERROR! MARCADOR NO DEFINIDO.
	MEJORAS EN LA EXPERIENCIA DEL USUARIO	¡ERROR! MARCADOR NO DEFINIDO.
18.	PRUEBAS Y DEPURACIÓN DEL FRONTEND	¡ERROR! MARCADOR NO DEFINIDO.

DISEÑO DE CASOS DE PRUEBA DE FRONTEND	¡ERROR! MARCADOR NO DEFINIDO.
PRUEBAS DE USABILIDAD.....	¡ERROR! MARCADOR NO DEFINIDO.
DEPURACIÓN DE ERRORES Y OPTIMIZACIÓN DEL CÓDIGO	¡ERROR! MARCADOR NO DEFINIDO.
19. IMPLEMENTACIÓN DE LA LÓGICA DE NEGOCIO EN EL FRONTEND.....	¡ERROR! MARCADOR NO DEFINIDO.
MIGRACIÓN DE LA LÓGICA DE NEGOCIO DESDE EL BACKEND (SI NECESARIO).....	¡ERROR! MARCADOR NO DEFINIDO.
VALIDACIÓN DE DATOS Y REGLAS DE NEGOCIO EN EL FRONTEND.....	¡ERROR! MARCADOR NO DEFINIDO.
20. INTEGRACIÓN CON EL BACKEND.....	¡ERROR! MARCADOR NO DEFINIDO.
VERIFICACIÓN DE LA COMUNICACIÓN EFECTIVA CON EL BACKEND.....	¡ERROR! MARCADOR NO DEFINIDO.
PRUEBAS DE INTEGRACIÓN FRONTEND-BACKEND	¡ERROR! MARCADOR NO DEFINIDO.
ANEXOS	¡ERROR! MARCADOR NO DEFINIDO.

Etapa 1 Diseño de la Aplicación y Análisis de Requisitos

1.Introducción

Propósito del Documento

El propósito del módulo de gestión de tareas descrito es mejorar la organización, seguimiento y eficiencia en la ejecución de actividades diarias en plataformas de contenido. A través de una interfaz intuitiva y herramientas colaborativas, el sistema facilita la gestión eficiente de tareas, distribuye responsabilidades de manera automática y optimiza la coordinación dentro de los equipos. Además, se enfoca en mejorar el bienestar de los usuarios mediante el monitoreo de la carga de trabajo y sugerencias de autocuidado, todo con el objetivo de mejorar el rendimiento y la calidad de la gestión de contenidos

Etapa 1.

Definición de requisitos:

Identifica al usuario: ¿Quién usará el Task Manager? ¿Son individuos, equipos, o departamentos?

Define los objetivos: ¿Para qué se usará el Task Manager? ¿Organización, colaboración, control de plazos, etc.?

Recopila las funcionalidades: ¿Qué funciones específicas necesita el Task Manager? Ej: creación de tareas, asignación, seguimiento del progreso, recordatorios, colaboración, priorización, etc.

Define los flujos de trabajo: ¿Cómo se utilizará el Task Manager en el día a día?

Determina restricciones: ¿Hay limitaciones de hardware, software, tiempo o presupuesto?

Etapa 2

Diseño conceptual:

Crea una estructura de datos: Define las tablas y campos necesarios para almacenar la información de las tareas, usuarios, proyectos, etc.

Diseña la interfaz de usuario: Crea prototipos de la interfaz, considerando la experiencia de usuario (UX), la accesibilidad y la usabilidad.

Escribe la lógica del sistema: Define los procesos que se ejecutarán en el backend para gestionar las tareas, las notificaciones, los permisos, etc.

Planifica la arquitectura del sistema: Decide la tecnología a usar (lenguajes de programación, bases de datos, framework), el modelo de desarrollo (ágil, cascada), etc.

Etapa 3.

Diseño detallado:

Define la arquitectura de software: Elige las tecnologías y componentes específicos a usar.

Diseña la base de datos: Define las tablas, campos, relaciones y tipos de datos.

Crea diagramas de flujo: Define la lógica de los procesos principales del sistema.

Diseña las API: Define las interfaces para la comunicación entre el frontend y el backend.

Diseña las pruebas: Planifica los casos de prueba para asegurar la calidad del software.

Etapa 4.

Implementación:

Desarrolla el frontend: Implementa la interfaz de usuario utilizando HTML, CSS, JavaScript y framework (React, Vue, Angular).

Desarrolla el backend: Implementa la lógica del sistema utilizando un lenguaje de programación (Python, Java, Node.js) y una base de datos (MySQL, PostgreSQL, MongoDB).

Integra las API: Conecta el frontend y el backend mediante las API definidas.

Realiza las pruebas: Ejecuta los casos de prueba para asegurar la calidad del software.

Etapa 5.

Pruebas y despliegue:

Pruebas de usuario: Obtiene feedback de usuarios reales para mejorar la usabilidad y la experiencia de usuario.

Despliegue: Publica el Task Manager en producción, utilizando una infraestructura de hosting adecuada.

Mantenimiento: Actualiza y mejora el sistema con nuevas funcionalidades, correcciones de errores y adaptaciones a los cambios en los requisitos.

Alcance del Proyecto Módulo de Pizarra Compartida

Alcance:

Funcionalidades:

- Gestión de tareas:
- Creación, edición y eliminación de tareas.
- Asignación de tareas a usuarios específicos.
- Establecimiento de prioridades y plazos.
- Seguimiento del progreso de las tareas.
- Adición de descripciones y archivos adjuntos.
- Integración con calendarios.
- Notificaciones por correo electrónico o mensajes en la aplicación.

Colaboración:

- Comentar tareas y discutir las con otros usuarios.
- Asignación de roles y permisos a los usuarios.
- Creación de equipos y listas de tareas compartidas.
- Integración con plataformas de comunicación (Slack, Microsoft Teams, etc.).

Reportes:

- Visualización de estadísticas sobre el progreso de las tareas.
- Filtrado y ordenamiento de tareas por distintos criterios.
- Generación de informes personalizados.

Integración:

- Integración con otras aplicaciones (CRM, ERP, etc.).
- Importación y exportación de datos.
- API para permitir la integración con otros sistemas.

Seguridad:

- Gestión de usuarios y permisos.
- Control de acceso a las tareas y datos.
- Encriptación de datos sensibles.

2.Descripción General

Objetivos del Sistema

El objetivo principal del sistema de gestión de tareas es optimizar la organización, eficiencia y colaboración en la ejecución de proyectos y tareas. Busca facilitar la creación, gestión y seguimiento de tareas, automatizar recordatorios y notificaciones, permitir la comunicación y colaboración entre miembros de un equipo, y brindar una visión general del progreso de los proyectos. En resumen, el sistema aspira a mejorar la productividad y el éxito en la realización de proyectos al centralizar la información, agilizar el flujo de trabajo y fomentar el trabajo en equipo

Funcionalidad General

1. **Creación de tareas:** Permite a los usuarios generar nuevas tareas, proporcionando detalles como nombre, descripción, fecha límite y responsable.
2. **Edición de tareas:** Facilita la modificación de las tareas existentes, permitiendo cambiar cualquier detalle, como la descripción o fecha límite.
3. **Eliminación de tareas:** Da la opción de eliminar tareas completadas o irrelevantes, garantizando que el tablero de tareas esté siempre actualizado.
4. **Asignación de tareas:** Asigna automáticamente o manualmente las tareas a los usuarios, equilibrando la carga de trabajo y las habilidades.
5. **Priorización de tareas:** Ofrece la posibilidad de establecer el nivel de prioridad de cada tarea para facilitar la organización según la urgencia o importancia.

6. Monitoreo en tiempo real: Proporciona un tablero en el que se puede observar el estado de cada tarea en tiempo real, con indicadores de progreso y alertas.

7. Asignación automática de tareas: El sistema distribuye tareas de manera automática según la carga de trabajo y la disponibilidad de los usuarios.

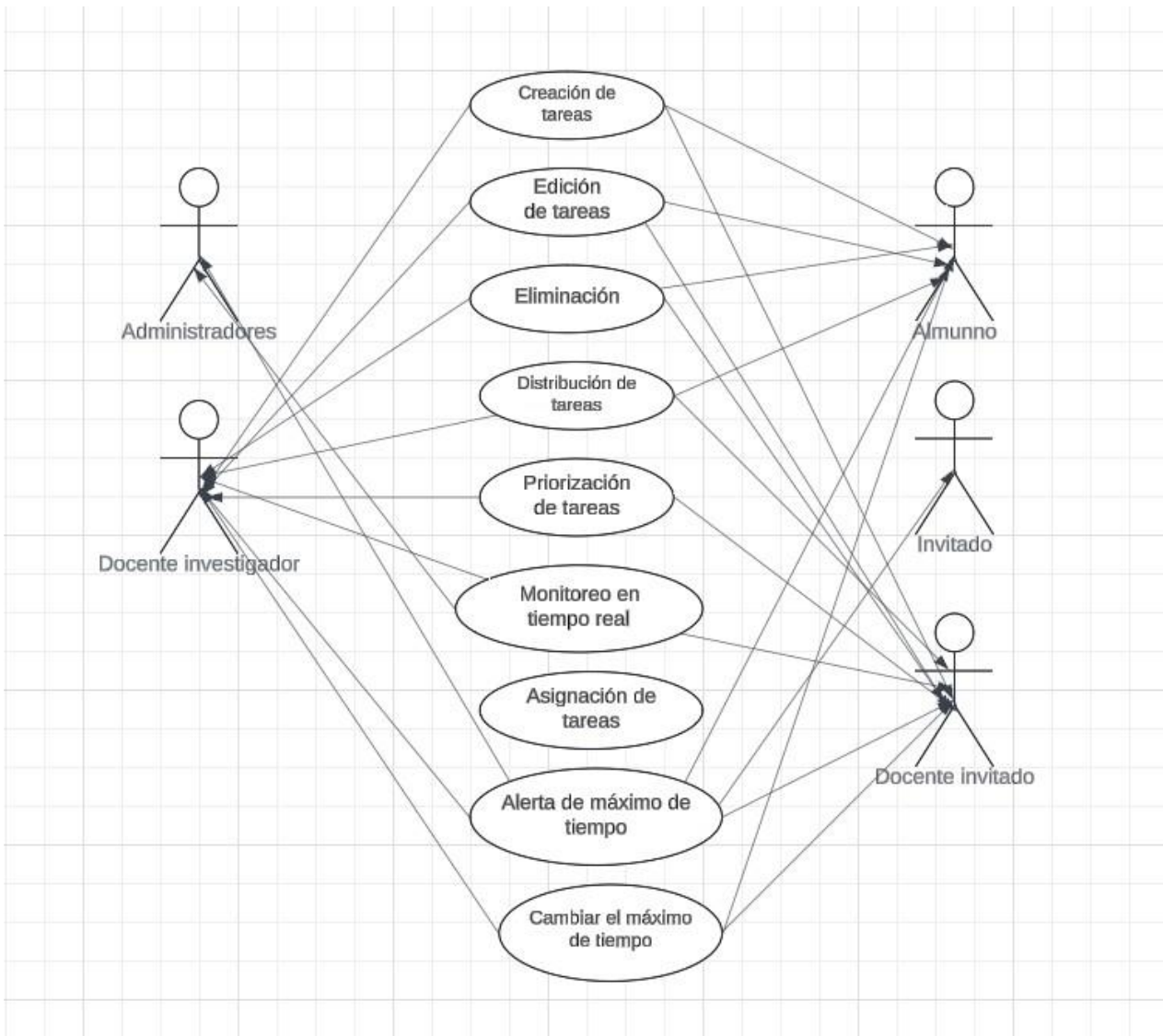
8. Alerta de máximo de tiempo: Emite notificaciones cuando una tarea excede el tiempo máximo asignado, ayudando a evitar retrasos.

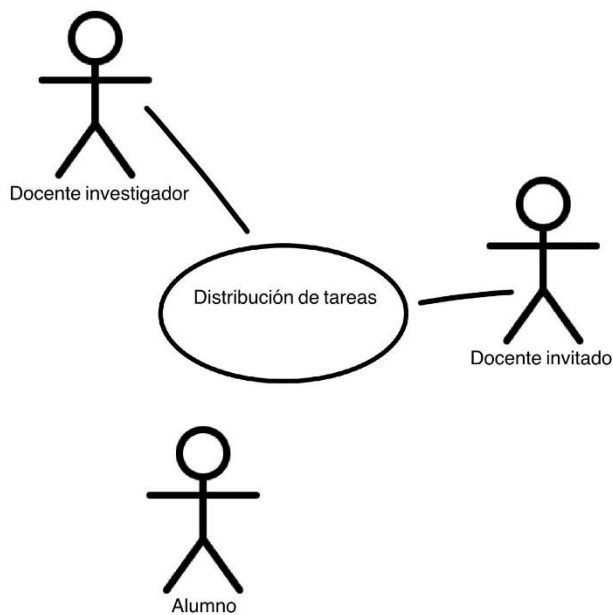
9. Cambiar el máximo de tiempo: Permite modificar el límite de tiempo máximo asignado a una tarea, ajustándolo según las necesidades del proyecto.

Usuarios del Sistema

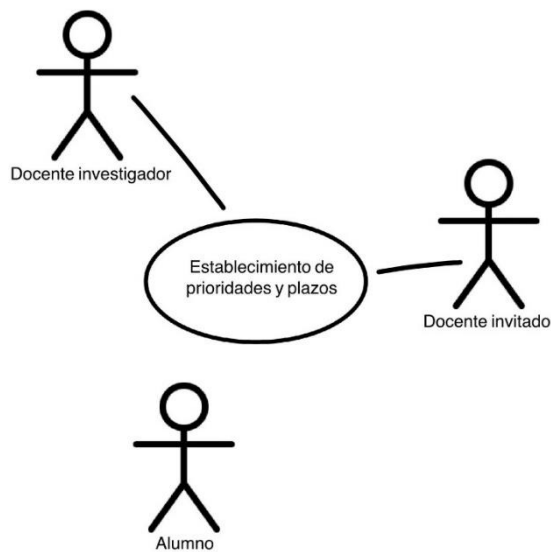
Los siguientes usuarios pueden interactuar con la pizarra dependiendo de las funcionalidades.

Funcionalidad	Administradores	Docente Investigador	Docente Invitado	Alumno	Invitado
Creación de tareas		✓	✓	✓	
Edición de tareas		✓	✓	✓	
Eliminación		✓	✓	✓	
Distribución de tareas		✓	✓	✓	
Priorización de tareas		✓	✓		
Monitoreo en tiempo real	✓	✓	✓		
Asignación de tareas	✓	✓	✓		
Alerta de máximo de tiempo	✓	✓	✓	✓	✓
		✓	✓	✓	
Integración con el CMS	✓	✓	✓		
Personalización y Temas	✓	✓	✓		
Acceso Seguro	✓	✓	✓	✓	
Notificaciones en Tiempo Real	✓	✓	✓	✓	
Acceso Móvil	✓	✓	✓	✓	✓
Búsqueda y Filtros	✓	✓	✓	✓	✓
Gestión de Usuarios y Permisos	✓				
Informes y Analíticas	✓	✓	✓		

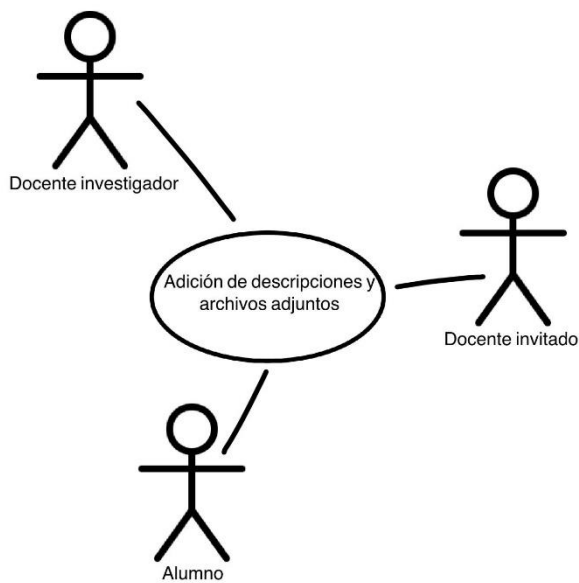




ID	FLUJO 001
Nombre	Distribución de tareas
Actores	<ul style="list-style-type: none"> • Docente Investigador • Docente Invitado
Objetivo	Organizar y asignar tareas específicas entre los diferentes actores involucrados en un proyecto o actividad académica para optimizar el tiempo y los recursos.
Urgencia	Nivel de urgencia que depende del contexto. Puedes clasificarlo como "Alta" si las tareas deben completarse en un corto plazo o "Media" si es para una planificación a mediano plazo.
Esfuerzo	Determinado en función de la carga de trabajo de cada actor. Puede ser una escala de "Bajo", "Medio" o "Alto" según las tareas asignadas.
Precondicionales	Reunión entre el Docente Investigador, el Docente Invitado y el Alumno para identificar las tareas y roles.
Flujo Normal	Si el Docente Invitado no puede asistir a la reunión, el Docente Investigador asigna las tareas basándose en la disponibilidad y comunicación posterior con el Docente Invitado.
Flujo alternativo 1	todas las tareas han sido distribuidas y cada actor tiene claridad sobre sus responsabilidades.
Post- Condicionales	Se establecen fechas de revisión y seguimiento.

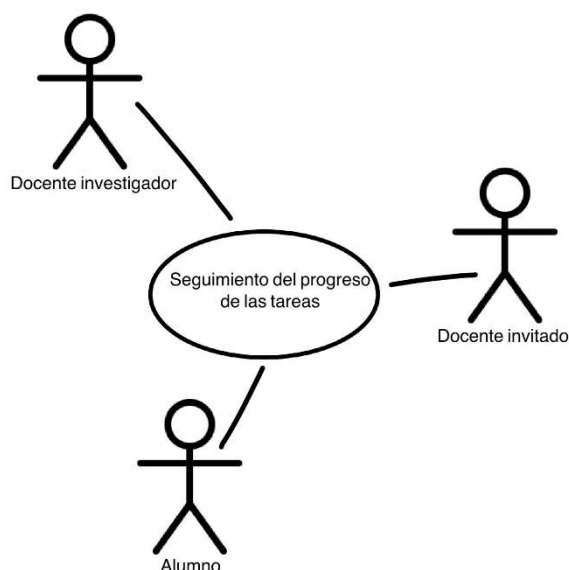


ID	FLUJO 002
Nombre	Establecimiento de prioridades y plazos
Actores	<ul style="list-style-type: none"> • Docente Investigador • Docente Invitado
Objetivo	Garantizar que las tareas sean priorizadas y planificadas de acuerdo con su impacto y urgencia para optimizar el flujo de trabajo del proyecto
Urgencia	Alta: Tareas críticas para el cumplimiento del proyecto o próximas a su fecha límite. Media: Tareas necesarias pero no críticas para el plazo inmediato. Baja: Actividades sin impacto inmediato en el proyecto.
Esfuerzo	Bajo: Menos de 2 horas de trabajo. Medio: De 2 a 8 horas de trabajo. Alto: Más de 8 horas o que requieren varios días de trabajo colaborativo.
Precondicionales	Análisis previo de las dependencias entre tareas. Definición de los recursos disponibles. Identificación de responsables para cada tarea.
Flujo Normal	Registrar todas las tareas en el software. Asignar prioridades basadas en la urgencia y el esfuerzo requerido. Establecer plazos y responsables para cada tarea. Supervisar el avance y actualizar el estado en tiempo real.
Flujo alternativo 1	Si no se puede determinar la urgencia de una tarea: Consultar al equipo o al cliente. Reasignar prioridades temporalmente según recursos disponibles.
Post- Condicionales	Tareas completadas marcadas como "Hechas" en el software. Generación de un informe de progreso. Evaluación de la efectividad del proceso para ajustar prioridades futuras.

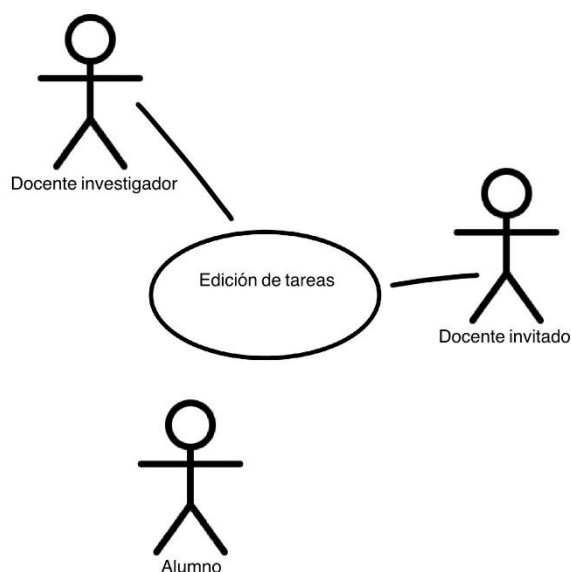


ID	FLUJO 003
Nombre	Adición de descripciones y archivos adjuntos
Actores	Usuario: Persona que añade descripciones y adjuntos. Administrador del sistema: Persona encargada de gestionar permisos y supervisar el correcto uso. Equipo colaborador: Personas que acceden a las descripciones y archivos para avanzar en sus tareas.
Objetivo	Facilitar la comprensión y el contexto de las tareas o proyectos mediante descripciones claras y la incorporación de archivos relevantes.
Urgencia	Alta: Cuando la descripción o archivo es crítico para iniciar o completar una tarea. Media: Necesario para evitar confusiones o retrasos futuros. Baja: Información complementaria que puede añadirse posteriormente
Esfuerzo	Bajo: Adición de un texto breve o un archivo ligero. Medio: Redacción extensa o carga de múltiples archivos. Alto: Requiere consultas adicionales para redactar o recopilar información detallada.
Precondicionales	Definición clara de la tarea o proyecto. Acceso al sistema con permisos necesarios. Disponibilidad de los archivos a adjuntar en el formato requerido.
Flujo Normal	Seleccionar la tarea o proyecto en el software. Redactar la descripción de manera clara y concisa. Adjuntar los archivos relevantes (documentos, imágenes, etc.). Guardar y verificar que la información es accesible para los demás usuarios
Flujo alternativo 1	Si el archivo excede el tamaño permitido: Comprimir el archivo o dividirlo en partes. Usar un enlace externo (por ejemplo, a un servicio en la

	nube). Adjuntar o vincular el archivo nuevamente
Post- Condicionales	La descripción y los archivos son visibles para todos los usuarios autorizados. Actualización automática en el registro del proyecto o tarea. Notificación enviada a los responsables o colaboradores (si está configurada).



ID	FLUJO 004
Nombre	Seguimiento del progreso de tareas
Actores	<ul style="list-style-type: none"> • Docente Investigador • Docente Invitado
Objetivo	Asegurar el cumplimiento efectivo de las tareas mediante el monitoreo constante de su progreso, identificando posibles obstáculos y garantizando que se cumplan los plazos establecidos.
Urgencia	Nivel de urgencia que depende del contexto. Puedes clasificarlo como “Alta” si las tareas deben completarse en un corto plazo o “Media” si es para una planificación a mediano plazo.
Esfuerzo	Determinado en función de la carga de trabajo de cada actor. Puede ser una escala de “Bajo”, “Medio” o “Alto” según las tareas asignadas.
Precondicionales	Reunión entre el Docente Investigador, el Docente Invitado y el Alumno para identificar las tareas y roles.
Flujo Normal	Si el Docente Invitado no puede asistir a la reunión, el Docente Investigador asigna las tareas basándose en la disponibilidad y comunicación posterior con el Docente Invitado.
Flujo alternativo 1	todas las tareas han sido distribuidas y cada actor tiene claridad sobre sus responsabilidades.
Post- Condicionales	Se establecen fechas de revisión y seguimiento.



ID	FLUJO 005
Nombre	Edición de tareas
Actores	<p>Usuario: Persona autorizada para realizar cambios en las tareas.</p> <p>Administrador del sistema: Responsable de gestionar permisos y acceso a las tareas.</p> <p>Colaboradores: Miembros del equipo afectados por los cambios.</p>
Objetivo	Modificar la información de una tarea existente para garantizar que los detalles reflejen los cambios en prioridades, responsables, plazos o especificaciones.
Urgencia	<p>Alta: Cambios críticos para evitar retrasos en el proyecto.</p> <p>Media: Ajustes necesarios para reflejar la evolución de las tareas.</p> <p>Baja: Cambios menores sin impacto inmediato en el flujo de trabajo.</p>
Esfuerzo	<p>Bajo: Ajuste rápido (e.g., corregir una fecha o agregar detalles menores).</p> <p>Medio: Modificación de múltiples campos o comunicación con el equipo.</p> <p>Alto: Reestructuración completa de la tarea o reasignación de recursos.</p>
Precondicionales	<p>Acceder al sistema y localizar la tarea a editar.</p> <p>Abrir el formulario de edición de la tarea.</p> <p>Realizar los cambios requeridos (e.g., título, descripción, fecha límite).</p> <p>Confirmar y guardar los cambios realizados.</p> <p>Notificar automáticamente a los involucrados sobre los cambios.</p>
Flujo Normal	<p>Si no se tienen los permisos necesarios:</p> <p>Solicitar acceso al administrador del sistema.</p> <p>Una vez otorgados los permisos, repetir el flujo normal.</p>
Flujo alternativo 1	todas las tareas han sido distribuidas y cada actor tiene claridad sobre sus responsabilidades.

Post- Condicionales	Tarea actualizada en el sistema con los cambios reflejados. Notificaciones enviadas a los responsables o interesados. Historial de cambios actualizado (si aplica en el sistema)..

Requisitos No Funcionales

Requisitos de Desempeño

Requisitos de Seguridad

Requisitos de Usabilidad

Requisitos de Escalabilidad

Modelado E/R

Diagrama de Entidad-Relación

Diagrama relacional

Descripción de Entidades y Relaciones

Reglas de Integridad

Anexos (si es necesario)

Diagramas Adicionales

Referencias

Etapa 2: Persistencia de Datos con Backend

Introducción

Propósito de la Etapa

Alcance de la Etapa

Definiciones y Acrónimos

Diseño de la Arquitectura de Backend

Descripción de la Arquitectura Propuesta

Componentes del Backend

Diagramas de Arquitectura

Elección de la Base de Datos

Evaluación de Opciones (SQL o NoSQL)

Justificación de la Elección

Diseño de Esquema de Base de Datos

Implementación del Backend

Elección del Lenguaje de Programación

Creación de la Lógica de Negocio

Desarrollo de Endpoints y APIs

Autenticación y Autorización

Conexión a la Base de Datos

Configuración de la Conexión

Desarrollo de Operaciones CRUD

Manejo de Transacciones

Pruebas del Backend

Diseño de Casos de Prueba

Ejecución de Pruebas Unitarias y de Integración

Manejo de Errores y Excepciones

