**DOCUMENTACIÓN TÉCNICA DE LAS VISTAS DEL GESTOR DE TAREAS ACADÉMICAS**

**DISEÑO DE SOFTWARE EDUCATIVO III**

**ESTUDIANTES: SARYEL NARVAEZ PEREZ**

**DODDIE ALAN SUAREZ BOLAÑOS**

**RAY DAVID MARTÍNEZ GUEVARA**

**LUIS ÁNGEL DÍAZ NISPERUZA**

**DOCENTE: RAUL EMIRO TOSCANO MIRANDA**

**UNIVERSIDAD DE CÓRDOBA**

**FACULTAD DE EDUCACIÓN Y CIENCIAS HUMANAS**

**PROGRAMA LICENCIATURA EN INFORMÁTICA**

**MONTERÍA – CÓRDOBA**

**24/06/2025**

Contenido

[Documentación Técnica de las Vistas del Gestor de Tareas Académicas 6](#_Toc200040849)

[Introducción 6](#_Toc200040850)

[Propósito del proyecto 6](#_Toc200040851)

[Fase 1: Diseño de la Aplicación y Análisis de Requisitos 6](#_Toc200040852)

[Fase 2: Persistencia de Datos con Backend – Servidor 7](#_Toc200040853)

[Fase 3: Consumo de Datos y Desarrollo Frontend – Cliente 7](#_Toc200040854)

[Alcance del proyecto 7](#_Toc200040855)

[Funcionalidades Futuras (Propuestas) 9](#_Toc200040856)

[Definiciones y acrónimos 9](#_Toc200040857)

[DESCRIPCIÓN GENERAL 10](#_Toc200040858)

[Objetivos del sistema 10](#_Toc200040859)

[Funcionalidad General 11](#_Toc200040860)

[ Gestión de Tareas Académicas: 11](#_Toc200040861)

[Usuarios del sistema 12](#_Toc200040862)

[Restricciones 13](#_Toc200040863)

[ Acceso limitado según el rol: 13](#_Toc200040864)

[ Registro previo: 13](#_Toc200040865)

[ Configuración de fechas: 13](#_Toc200040866)

[ Requisitos técnicos: 13](#_Toc200040867)

[ Seguridad y privacidad: 13](#_Toc200040868)

[ Control de versiones: 13](#_Toc200040869)

[REQUISITOS NO FUNCIONALES 14](#_Toc200040870)

[Requisitos de Desempeño 14](#_Toc200040871)

[ Tiempo de Respuesta Rápido: 14](#_Toc200040872)

[ Compatibilidad Multiplataforma: 14](#_Toc200040873)

[ Optimización de Recursos: 14](#_Toc200040874)

[ Gestión de Archivos: 14](#_Toc200040875)

[Requisitos de Seguridad 14](#_Toc200040876)

[Requisitos de Usabilidad 15](#_Toc200040877)

[Requisitos de Escalabilidad 15](#_Toc200040878)

[Requisitos funcionales 16](#_Toc200040879)

[ Iniciar Sesión: 16](#_Toc200040880)

[ Crear y Asignar Tareas: 16](#_Toc200040881)

[ Visualizar Tareas Asignadas: 16](#_Toc200040882)

[o Los estudiantes visualizan sus tareas pendientes y completadas, incluyendo detalles de la actividad y fechas límite. 16](#_Toc200040883)

[ Entrega de Evidencias: 16](#_Toc200040884)

[ Seguimiento y Retroalimentación: 17](#_Toc200040885)

[ Generar Reportes: 17](#_Toc200040886)

[ Historial de Actividades: 17](#_Toc200040887)

[Casos de uso 18](#_Toc200040888)

[Vista de Inicio de Sesión (Login) 19](#_Toc200040889)

[Descripción 19](#_Toc200040890)

[Componentes: 19](#_Toc200040891)

[ Título Principal: 19](#_Toc200040892)

[ Subtítulo: 19](#_Toc200040893)

[ Formulario de inicio de sesión: 19](#_Toc200040894)

[ Enlaces superiores: 19](#_Toc200040895)

[ Credenciales de prueba: texto informativo al pie para facilitar las pruebas del sistema. 20](#_Toc200040896)

[ Diseño Visual: 20](#_Toc200040897)

[ Fondo con degradado morado. 20](#_Toc200040898)

[ Contenedor central claro y con bordes redondeados. 20](#_Toc200040899)

[ Funcionalidad: 20](#_Toc200040900)

[Panel del Docente 21](#_Toc200040901)

[Descripción 21](#_Toc200040902)

[Componentes: 21](#_Toc200040903)

[ Barra lateral de navegación: 21](#_Toc200040904)

[ Encabezado: 21](#_Toc200040905)

[ Panel central para crear nuevas tareas: 22](#_Toc200040906)

[Panel del Estudiante 22](#_Toc200040907)

[Propósito 22](#_Toc200040908)

[Estructura General de la Vista 23](#_Toc200040909)

[ Encabezado de Bienvenida: 23](#_Toc200040910)

[ Menú Lateral (Sidebar): 23](#_Toc200040911)

[ Barra de Búsqueda y Filtro: 23](#_Toc200040912)

[ Sección de Tareas Pendientes: 23](#_Toc200040913)

[ Sección de Tareas Asignadas: 23](#_Toc200040914)

[Diseño Visual 24](#_Toc200040915)

[ Paleta de Colores: 24](#_Toc200040916)

[ Tipografía: 24](#_Toc200040917)

[ Botones y Acciones: 24](#_Toc200040918)

[ Comportamiento Interactivo 25](#_Toc200040919)

[ Accesibilidad 25](#_Toc200040920)

[ Buenas Prácticas en la Implementación 25](#_Toc200040921)

[ Elementos de Interfaz en la Vista 26](#_Toc200040922)

[Diseño de guía de metáforas 27](#_Toc200040923)

[Conexión del Frontend y Backend 28](#_Toc200040924)

[Frontend (Cliente) 29](#_Toc200040925)

[ Stack: 29](#_Toc200040926)

[ Estructura: 29](#_Toc200040927)

[ Plugins de Visual Editor: 30](#_Toc200040928)

[Backend (Servidor) 30](#_Toc200040929)

[Flujo de Conexión 30](#_Toc200040930)

[ Solicitud de Tareas (GET) 30](#_Toc200040931)

[ Creación de Tareas (POST) 30](#_Toc200040932)

[ Entrega de Evidencias (POST / PUT) 31](#_Toc200040933)

[ Actualización de Estados (PATCH / PUT) 31](#_Toc200040934)

[Seguridad y Validación 31](#_Toc200040935)

[Estilos Dinámicos y Personalización 31](#_Toc200040936)

[Herramientas y Dependencias 32](#_Toc200040937)

[ Vite: 32](#_Toc200040938)

[ Tailwind CSS: 32](#_Toc200040939)

[ postcss.config.js y tailwind.config.js: 32](#_Toc200040940)

[ package.json / package-lock.json: 32](#_Toc200040941)

[Buenas Prácticas Observadas 32](#_Toc200040942)

# Documentación Técnica de las Vistas del Gestor de Tareas Académicas

# Introducción

El Gestor de Tareas Académicas está diseñado para facilitar la gestión de actividades educativas, brindando herramientas intuitivas para que profesores y estudiantes interactúen de manera eficiente y segura. Las vistas se han construido teniendo en cuenta la accesibilidad, la usabilidad y la coherencia visual, cumpliendo con buenas prácticas de desarrollo web.

# Propósito del proyecto

El **Gestor de Tareas Académicas** es un proyecto de desarrollo de software educativo que tiene como propósito optimizar la planificación, asignación y seguimiento de actividades académicas en instituciones educativas. Surge como respuesta a la necesidad de contar con una herramienta digital eficiente que permita a los docentes crear tareas, asignarlas a grupos específicos de estudiantes y monitorear su progreso, así como facilitar a los estudiantes la visualización, entrega y seguimiento de sus actividades.

Este documento busca documentar todo el proceso de diseño, análisis e implementación del **Gestor de Tareas Académicas** en las plataformas educativas. El proyecto se organiza en tres fases principales, cada una con un enfoque claro y específico:

Fase 1: Diseño de la Aplicación y Análisis de Requisitos  
En esta etapa se realiza un levantamiento detallado de las necesidades del usuario final (docentes y estudiantes), así como el diseño de las funcionalidades clave de la aplicación. Se prioriza la experiencia de usuario (UX) y la interfaz de usuario (UI) para asegurar la accesibilidad y la facilidad de uso. Se documentan los casos de uso y se definen los flujos de trabajo necesarios para que la aplicación cumpla con sus objetivos.

Fase 2: Persistencia de Datos con Backend – Servidor  
Durante esta fase se desarrolla la lógica de negocio, la arquitectura de la base de datos y las operaciones necesarias para almacenar, recuperar y manipular la información de las tareas, usuarios y entregas. Se asegura la integridad de los datos y se implementan mecanismos de seguridad para proteger la privacidad y los registros académicos.

Fase 3: Consumo de Datos y Desarrollo Frontend – Cliente  
Aquí se materializa la capa visual o presentación de la aplicación, implementando los diseños definidos y conectando las vistas con los servicios del backend. Se garantiza que la interfaz sea atractiva, funcional y compatible con múltiples dispositivos, cumpliendo con los estándares de usabilidad y accesibilidad.

# Alcance del proyecto

El objetivo principal del **Gestor de Tareas Académicas** es crear un entorno digital que permita a los docentes y estudiantes gestionar de manera eficaz las actividades académicas en la plataforma. La herramienta está diseñada para agilizar los procesos de planificación y entrega de tareas, centralizando toda la información en un solo lugar.

El proyecto busca atender las necesidades de dos tipos principales de usuarios:

* **Docentes:** quienes podrán crear tareas personalizadas, asignarlas a cursos específicos, definir fechas de entrega y hacer seguimiento a las entregas de sus estudiantes.
* **Estudiantes:** quienes podrán visualizar las tareas asignadas, conocer las fechas límite y entregar sus evidencias de trabajo de manera sencilla y ordenada.

Para lograr estos objetivos, el proyecto contempla las siguientes funcionalidades clave:

* **Creación y Asignación de Tareas:** Los docentes pueden crear actividades detalladas, incluir descripciones, archivos adjuntos y definir fechas límite.
* **Visualización de Tareas:** Los estudiantes acceden a una lista clara de sus actividades pendientes y completadas, con información sobre la fecha de entrega y los criterios de evaluación.
* **Entrega de Evidencias:** Los estudiantes pueden adjuntar archivos, responder directamente en la plataforma o marcar las tareas como completadas.
* **Seguimiento del Progreso:** Los docentes cuentan con herramientas para verificar las entregas, calificar y dar retroalimentación.
* **Registro de Actividades:** La plataforma mantiene un historial detallado de las tareas creadas, asignadas y entregadas, con fechas y datos relevantes.
* **Seguridad y Privacidad:** Se implementan medidas para proteger la integridad y confidencialidad de los datos académicos de los usuarios.

# Funcionalidades Futuras (Propuestas)

Si bien la implementación inicial se centra en las funcionalidades antes mencionadas, se proyecta incorporar nuevas características en futuras versiones, tales como:

* **Notificaciones Personalizadas:** Recordatorios automáticos para docentes y estudiantes acerca de tareas próximas a vencer.
* **Integración con Herramientas de Comunicación:** Conexión con foros o chats para facilitar la interacción entre docentes y estudiantes.
* **Análisis de Rendimiento Académico:** Visualización de métricas sobre el cumplimiento y calidad de las entregas, con reportes detallados para docentes y estudiantes.

# Definiciones y acrónimos

* **Tarea Académica:** Actividad asignada por el docente a los estudiantes, con un objetivo de aprendizaje y una fecha límite.
* **Docente:** Usuario con permisos para crear, asignar y evaluar tareas dentro de la plataforma.
* **Estudiante:** Usuario que accede a las tareas, visualiza las instrucciones y entrega evidencias.
* **Interfaz de Usuario (UI):** Parte visual del sistema donde interactúan docentes y estudiantes.
* **Experiencia de Usuario (UX):** Percepción y satisfacción del usuario al utilizar la plataforma.
* **Entrega de Tareas:** Proceso mediante el cual los estudiantes presentan sus trabajos o productos académicos.
* **Seguimiento de Tareas:** Funcionalidad que permite a los docentes monitorear el progreso y cumplimiento de las actividades.
* **Base de Datos:** Sistema que almacena de manera estructurada la información de usuarios, tareas y entregas.
* **Plataforma Educativa:** Espacio digital donde se aloja el sistema, permitiendo la interacción entre docentes y estudiantes.

# DESCRIPCIÓN GENERAL

## Objetivos del sistema

El objetivo principal del sistema es facilitar la planificación, gestión y seguimiento de las tareas académicas en las instituciones educativas, mejorando la experiencia de estudiantes y docentes en el proceso de enseñanza-aprendizaje. Este sistema ha sido diseñado específicamente para estudiantes y docentes de las instituciones, con el propósito de optimizar la organización de actividades académicas, asegurar la entrega oportuna de las mismas y fortalecer la comunicación y la retroalimentación.

El **Gestor de Tareas Académicas** busca simplificar el proceso de asignación y control de actividades, garantizando la accesibilidad, eficiencia y seguridad de la información. Además, promueve la autonomía y responsabilidad de los estudiantes, facilitando la gestión de sus compromisos académicos y mejorando la interacción con las herramientas y recursos disponibles.

## Funcionalidad General

* Gestión de Tareas Académicas: La funcionalidad principal de este proyecto consiste en permitir a los docentes crear y asignar tareas de manera sencilla, detallando instrucciones, fechas de entrega y recursos complementarios.
* **Visualización de Actividades:** Los estudiantes podrán visualizar de forma clara y ordenada las tareas asignadas, sus fechas límite y el estado de entrega, lo que les permitirá organizar su tiempo y priorizar actividades.
* **Entrega de Evidencias:** El sistema permite a los estudiantes adjuntar archivos o realizar entregas directas a través de la plataforma, eliminando la necesidad de entregas físicas o por otros medios menos eficientes.
* **Seguimiento del Progreso:** Los docentes pueden monitorear el estado de las entregas, revisar evidencias, calificar y brindar retroalimentación oportuna a los estudiantes.
* **Historial de Actividades:** Se registrarán y documentarán todas las tareas asignadas y entregadas, generando un historial detallado que servirá como referencia tanto para docentes como para estudiantes.
* **Exportación de Reportes:** El sistema ofrecerá la capacidad de generar reportes en formatos como PDF o Excel, facilitando el seguimiento del desempeño académico y la toma de decisiones por parte de los docentes.
* **Integración con la Plataforma Educativa:** El **Gestor de Tareas Académicas** se integrará de manera fluida, permitiendo a los usuarios acceder a recursos, foros de discusión y demás herramientas académicas.

## Usuarios del sistema

Los siguientes usuarios pueden interactuar con el sistema del gestor de tareas, de acuerdo con las funcionalidades disponibles:

|  |  |  |
| --- | --- | --- |
| Funcionalidad | Docente | Estudiante |
| Iniciar sesión | ✅ | ✅ |
| Crear y asignar tareas | ✅ |  |
| Visualizar tareas asignadas | ✅ | ✅ |
| Entregar evidencias |  | ✅ |
| Calificar y dar retroalimentación | ✅ |  |
| Exportar reportes | ✅ |  |
| Visualizar historial de tareas | ✅ | ✅ |

## Restricciones

El sistema puede estar sujeto a restricciones para asegurar el cumplimiento de las políticas institucionales y la protección de los datos:

* Acceso limitado según el rol: Solo los usuarios autorizados (docentes y estudiantes) podrán acceder a determinadas funcionalidades.
* Registro previo: Los usuarios deben estar registrados y autorizados para utilizar el sistema.
* Configuración de fechas: Los docentes pueden definir fechas límite y horarios específicos para la entrega de tareas.
* Requisitos técnicos: Se requiere conexión a internet y dispositivos compatibles con la plataforma para un funcionamiento adecuado.
* Seguridad y privacidad: Se implementarán mecanismos de encriptación y autenticación para proteger la integridad de los datos y garantizar la privacidad de los usuarios.
* Control de versiones: Se podrán realizar actualizaciones periódicas para garantizar la mejora continua del sistema y la adaptación a nuevas necesidades académicas.

# REQUISITOS NO FUNCIONALES

## Requisitos de Desempeño

* Tiempo de Respuesta Rápido: El sistema debe ser capaz de procesar las solicitudes de los usuarios (visualización de tareas, creación de actividades, envío de entregas) en un tiempo máximo de 2 segundos, garantizando una experiencia fluida.
* Compatibilidad Multiplataforma: El sistema debe funcionar correctamente en dispositivos móviles y de escritorio, con independencia del sistema operativo utilizado.
* Optimización de Recursos: El sistema debe solicitar al servidor únicamente los recursos necesarios, reduciendo la carga en el hardware y optimizando el rendimiento de la aplicación.
* Gestión de Archivos: El sistema debe permitir la carga y descarga eficiente de evidencias o reportes académicos en diferentes formatos (PDF, DOCX, JPG, entre otros).

## Requisitos de Seguridad

* **Integridad de los Datos:** Toda la información de usuarios, tareas y entregas debe almacenarse y transmitirse de manera segura, evitando accesos no autorizados.
* **Autenticación Segura:** Se debe implementar un mecanismo de autenticación confiable (usuario y contraseña), además de encriptar las credenciales almacenadas en la base de datos.
* **Pruebas de Seguridad:** Se deberán realizar pruebas continuas para detectar vulnerabilidades y corregirlas oportunamente, garantizando la integridad y la privacidad de la información.
* **Soporte para Integraciones:** El sistema debe contar con la capacidad de integrarse con otros módulos o plataformas educativas de la universidad, manteniendo la coherencia y seguridad de los datos.

## Requisitos de Usabilidad

1. **Interfaz Clara y Accesible:** La plataforma debe ser fácilmente comprensible para usuarios de diferentes niveles de experiencia, incluyendo el uso de elementos visuales y textos descriptivos.
2. **Navegación Intuitiva:** El flujo de trabajo dentro de la aplicación debe ser lógico y sencillo, disminuyendo la necesidad de capacitación adicional.
3. **Retroalimentación del Sistema:** El sistema debe proporcionar mensajes claros y oportunos (éxito, error o advertencia) al realizar cualquier acción, como la entrega o la calificación de tareas.

## Requisitos de Escalabilidad

1. **Carga de Usuarios:** La plataforma debe poder manejar a un número significativo de estudiantes y docentes conectados simultáneamente, sin afectar la velocidad o estabilidad.
2. **Escalabilidad del Almacenamiento:** El sistema debe poder crecer en capacidad de almacenamiento a medida que aumentan los registros de tareas, entregas y usuarios.
3. **Alta Disponibilidad y Tolerancia a Fallos:** El sistema debe garantizar disponibilidad continua y ser capaz de recuperarse rápidamente ante fallas de hardware o software, manteniendo la operatividad y evitando la pérdida de datos.

## Requisitos funcionales

### Iniciar Sesión:

* Los usuarios (docentes y estudiantes) ingresan sus credenciales (usuario y contraseña).
* El servidor valida las credenciales y permite el acceso a la plataforma.

### Crear y Asignar Tareas:

* Los docentes crean nuevas actividades académicas, definiendo título, descripción, recursos adjuntos y fechas límite.
* Asignan las tareas a cursos específicos o grupos de estudiantes.

### Visualizar Tareas Asignadas:

### Los estudiantes visualizan sus tareas pendientes y completadas, incluyendo detalles de la actividad y fechas límite.

### Entrega de Evidencias:

* Los estudiantes pueden subir archivos o enviar textos como respuesta directa a la tarea, marcando la entrega como completada.

### Seguimiento y Retroalimentación:

* Los docentes pueden revisar las evidencias entregadas, calificarlas y brindar retroalimentación oportuna.

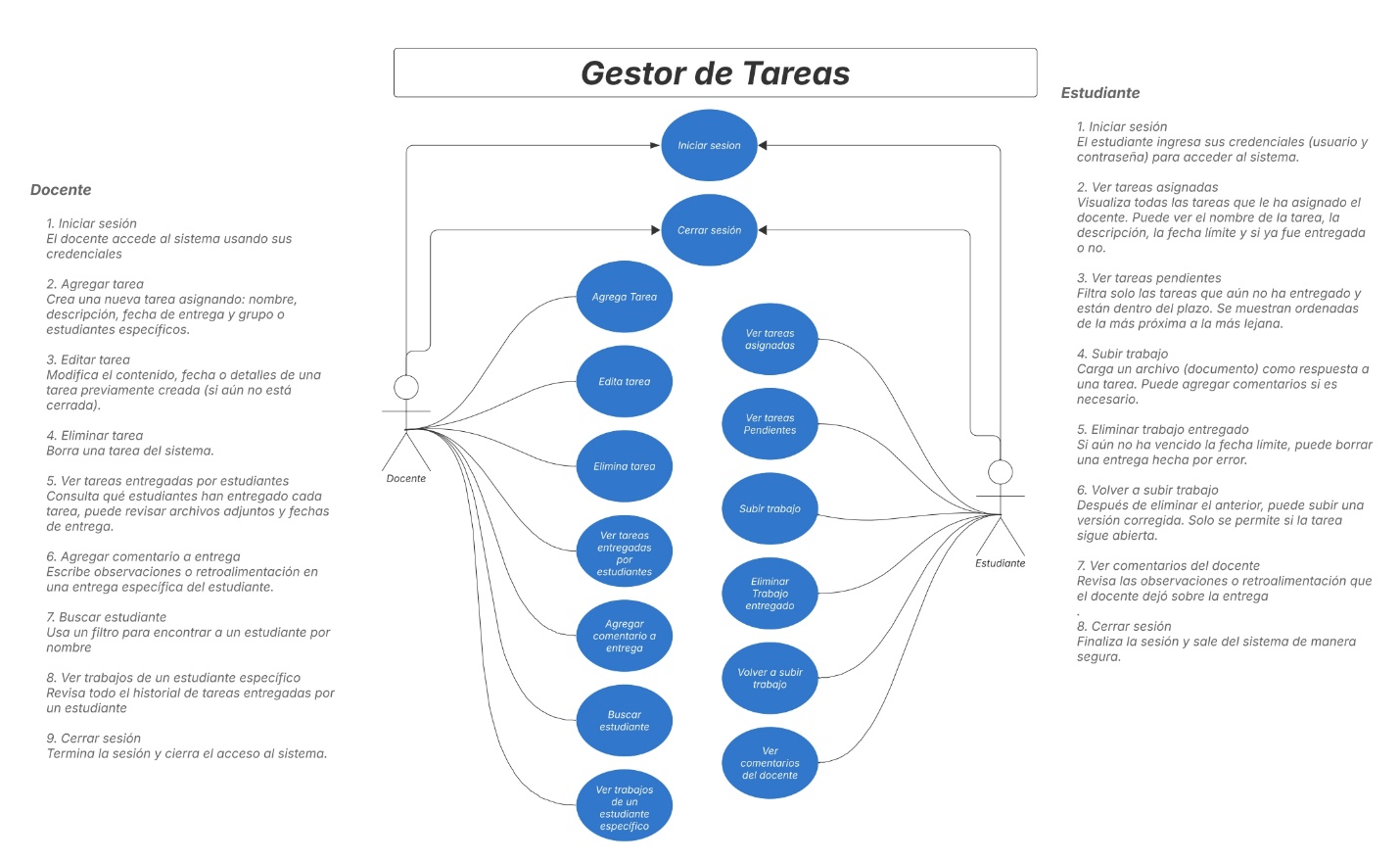
### Generar Reportes:

* Los docentes pueden generar reportes de tareas entregadas, pendientes y desempeño de los estudiantes, exportándolos en diferentes formatos (PDF, Excel).

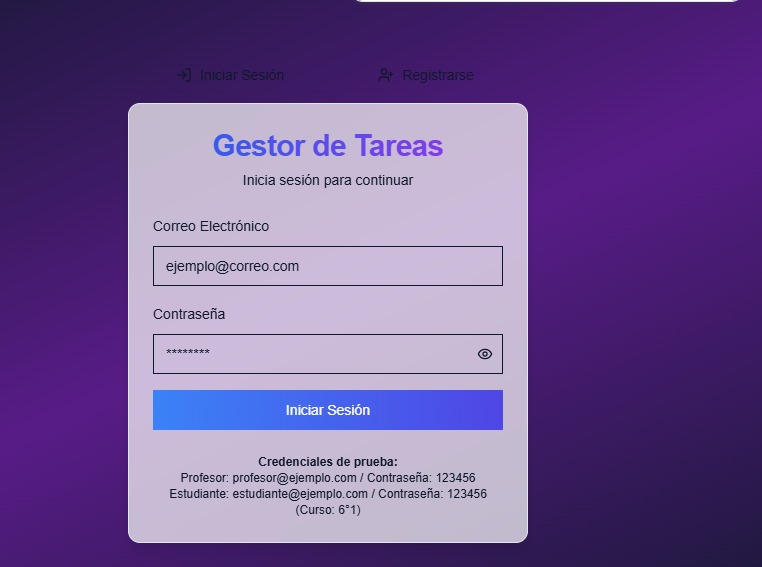
### Historial de Actividades:

* El sistema registra la hora y fecha de creación de las tareas, las entregas realizadas y la calificación obtenida, manteniendo un historial detallado de la gestión académica.

# Casos de uso

Diagrama de casos de uso

# Vista de Inicio de Sesión (Login)



Descripción  
Interfaz de inicio para que los usuarios accedan a la plataforma (profesores y estudiantes).

## Componentes:

* Título Principal: “Gestor de Tareas” con tipografía destacada.
* Subtítulo: “Inicia sesión para continuar”.

### Formulario de inicio de sesión:

* Campo de **Correo Electrónico** (input[type="email"]).
* Campo de **Contraseña** (input[type="password"]) con icono para mostrar/ocultar.
* **Botón de Iniciar Sesión:** botón principal con color llamativo (azul/morado).

### Enlaces superiores:

* Iniciar Sesión (activo, resaltado).
* Registrarse (opción alternativa).

### **Credenciales de prueba:** texto informativo al pie para facilitar las pruebas del sistema.

### Diseño Visual:

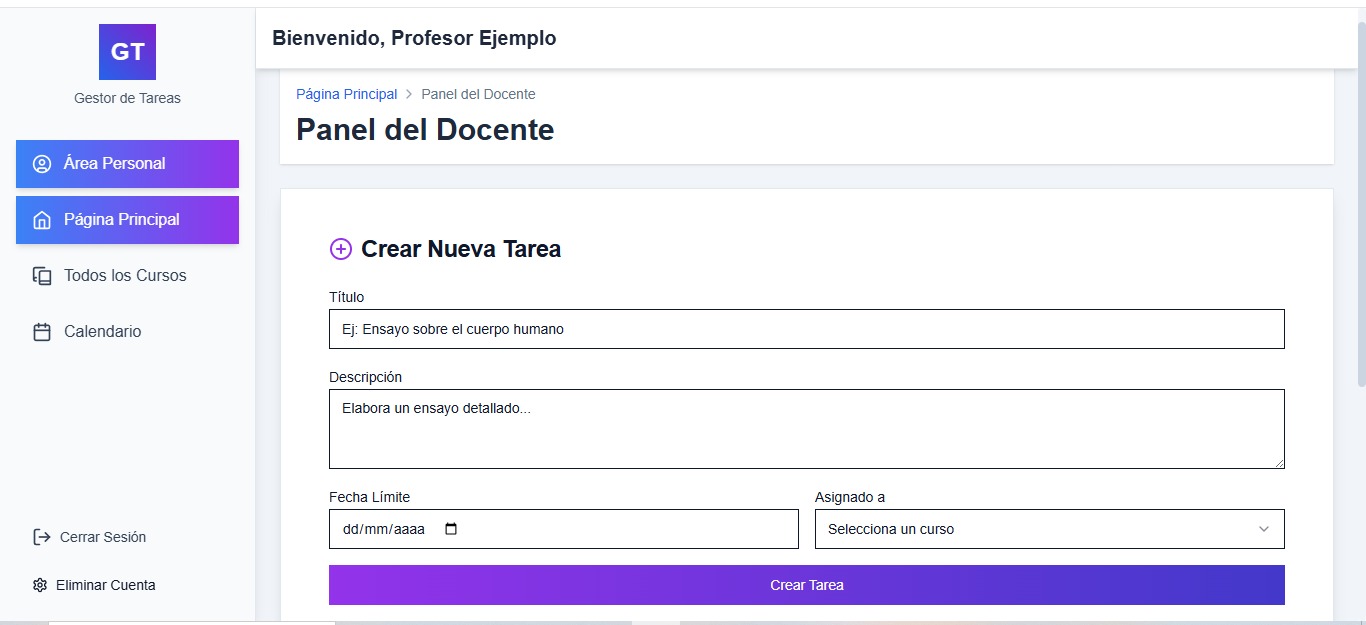
### Fondo con degradado morado.

### Contenedor central claro y con bordes redondeados.

### Funcionalidad:

* Validación de campos antes de enviar.
* Opción para mostrar/ocultar la contraseña.
* Mensajes de error claros en caso de credenciales inválidas.

# Panel del Docente



Descripción  
Interfaz principal para los profesores, donde pueden gestionar sus tareas y cursos.

## Componentes:

### Barra lateral de navegación:

* **Área Personal** (perfil del usuario).
* **Página Principal** (acceso al panel de creación de tareas).
* **Todos los Cursos**.
* **Calendario**.
* **Cerrar Sesión** y **Eliminar Cuenta** al final.

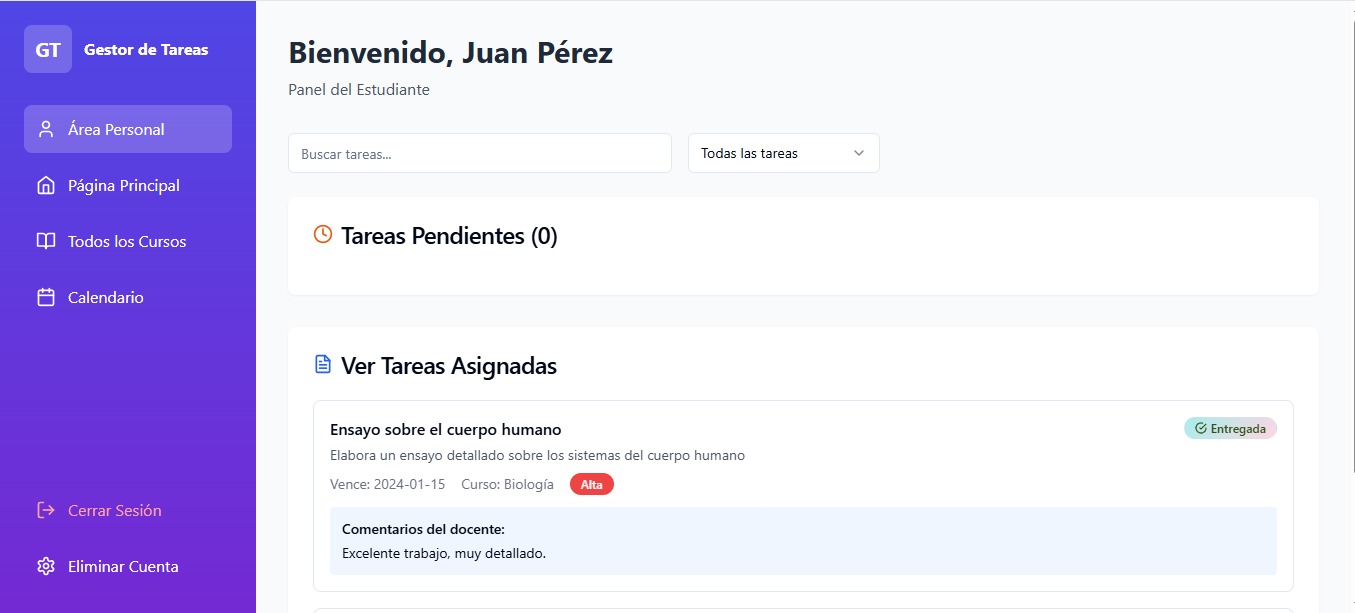
### Encabezado:

* Bienvenida personalizada (Bienvenido, Profesor Ejemplo).
* Migas de pan: Página Principal > Panel del Docente.

### Panel central para crear nuevas tareas:

* Campo **Título** con ejemplo (Ej: Ensayo sobre el cuerpo humano).
* Campo **Descripción** para detallar la tarea.
* **Fecha Límite:** selector de fecha con formato dd/mm/aaaa.
* **Asignado a:** menú desplegable para seleccionar un curso.
* **Botón “Crear Tarea”** destacado en morado degradado.

# Panel del Estudiante



## Propósito

La vista del **Panel del Estudiante** es el punto central de interacción de los estudiantes con la plataforma. Proporciona una experiencia intuitiva y accesible para consultar, entregar y gestionar las tareas asignadas, facilitando la organización académica y la comunicación con los docentes.

## Estructura General de la Vista

* Encabezado de Bienvenida: Texto de bienvenida con el nombre del usuario (ejemplo: “Bienvenido, Juan Pérez”), reforzando la personalización y la identificación dentro de la plataforma.
* Menú Lateral (Sidebar): Ubicado a la izquierda con fondo degradado púrpura, contiene:
* Área Personal
* Página Principal
* Todos los Cursos
* Calendario
* Acciones finales: Cerrar Sesión y Eliminar Cuenta

### Barra de Búsqueda y Filtro:

* Input para buscar tareas por nombre.
* Filtro desplegable para ver “Todas las tareas” o categorías específicas (pendientes, entregadas, vencidas).
* Sección de Tareas Pendientes: Contenedor con ícono de alerta, indicando el número de actividades pendientes de entrega.
* Sección de Tareas Asignadas:

Contenedor con un ícono de carpeta o documento, donde se listan todas las tareas asignadas con:

* Título y Descripción de la tarea.
* Fecha de Vencimiento y Curso al que pertenece.
* Nivel de Prioridad: con etiquetas de colores (Alta, Media, etc.).
* Estado de la Tarea: Entregada, Vencida, o pendiente.
* Comentarios del Docente: Sección destacada con retroalimentación individual.

## Diseño Visual

### Paleta de Colores:

* **Fondo principal:** blanco puro para claridad.
* **Sidebar:** degradada púrpura, reforzando la identidad visual de la plataforma.
* **Etiquetas de prioridad:** rojo (Alta), azul (Media), para diferenciarlas fácilmente.
* **Estados de entrega:**
* Verde claro para “Entregada”.
* Rojo claro para “Vencida”.
* Neutro para pendiente.

### Tipografía:

* Moderna y legible (sans-serif).
* Títulos destacados en negrita.
* Cuerpo de texto con tamaño adecuado para evitar fatiga visual.

### Botones y Acciones:

* Íconos visuales en cada entrada de tarea para indicar estado o acción.
* Feedback visual claro para cada acción (p. ej., cambio de color en el estado de entrega).

### Comportamiento Interactivo

La vista responde dinámicamente a la interacción del usuario:

* **Búsqueda de Tareas:** El campo de búsqueda filtra resultados en tiempo real.
* **Actualización de Estado:** Cada tarea refleja inmediatamente su estado (Entregada o Vencida).
* **Retroalimentación en Comentarios:** Los comentarios del docente aparecen destacados en un recuadro sutilmente sombreado.
* **Cambio de Prioridad o Curso:** Las etiquetas de prioridad y curso se muestran como chips visuales para fácil identificación.

### **Accesibilidad**

* Contraste adecuado entre texto y fondo.
* Tamaño de fuente legible para usuarios con visión reducida.
* Compatibilidad con lectores de pantalla para navegación mediante teclado o voz.

### Buenas Prácticas en la Implementación

* **Componentización:** Cada bloque (p. ej., “Tareas Pendientes”, “Tareas Asignadas”) está implementado como componente independiente, facilitando el mantenimiento y la escalabilidad.
* **Responsive Design:** La interfaz se adapta a dispositivos móviles y de escritorio, garantizando usabilidad en todas las plataformas.
* **Mensajes de Feedback:** Uso de colores y mensajes claros (p. ej., “¡Entregada!”, “¡Vencida!”) para reducir la incertidumbre del usuario.
* **Separación de Roles:** La interfaz de estudiante solo muestra acciones permitidas para este rol, garantizando seguridad y evitando confusiones.

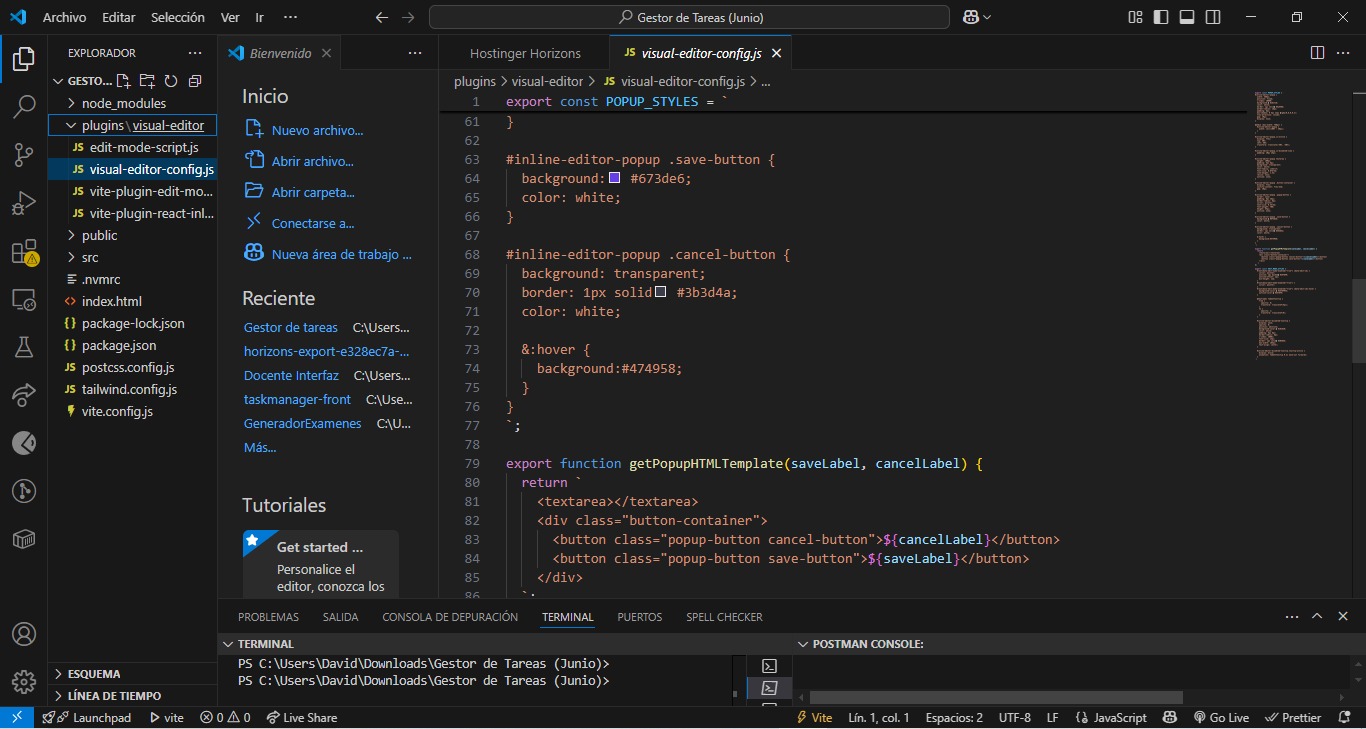
### Elementos de Interfaz en la Vista

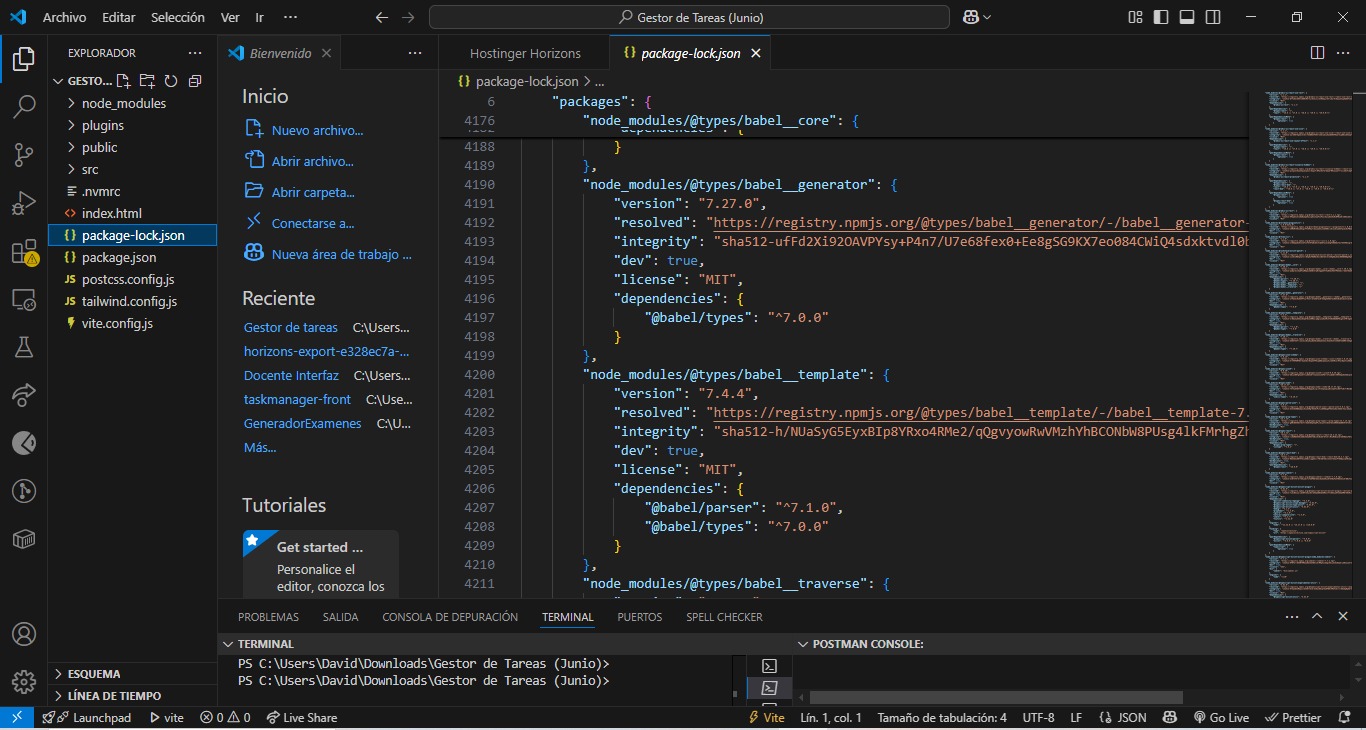
|  |  |
| --- | --- |
| Elemento | Función / Descripción |
| Sidebar | Navegación entre secciones principales y acciones de sesión. |
| Encabezado | Bienvenida y refuerzo de identidad. |
| Búsqueda y Filtros | Ayudan al estudiante a encontrar rápidamente tareas específicas. |
| Tareas Pendientes | Muestra tareas que requieren atención inmediata. |
| Tareas Asignadas | Vista completa de todas las actividades y sus detalles. |
| Etiquetas de Prioridad | Permiten distinguir tareas críticas o urgentes (alta, media, baja). |
| Estados de Entrega | Ayudan al estudiante a hacer seguimiento de su progreso. |
| Retroalimentación del Docente | Permite ver observaciones o recomendaciones sobre las entregas. |
| Opciones de Sesión | Cerrar sesión o eliminar la cuenta de manera segura. |

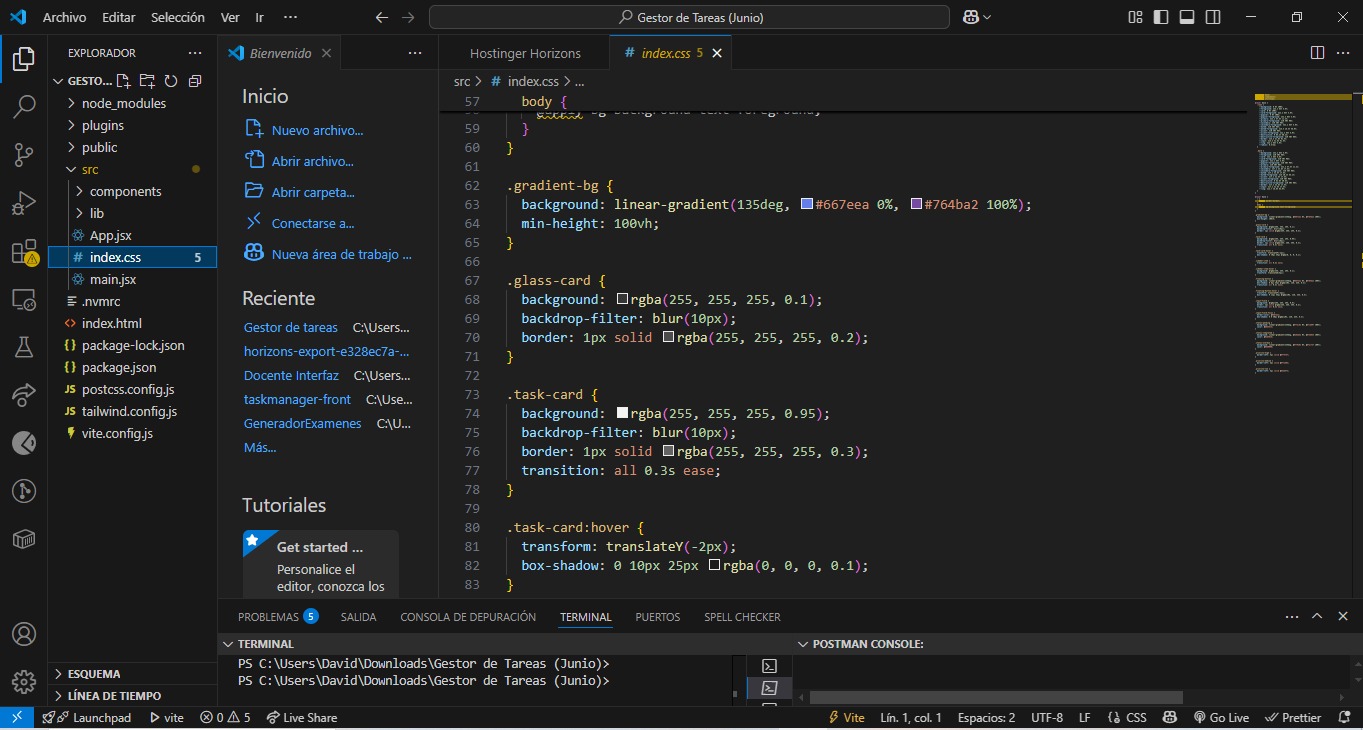
# Diseño de guía de metáforas

|  |  |  |
| --- | --- | --- |
| Nombre | Ícono | Descripción / Metáfora |
| Inicio de Sesión | 🔒 (candado abierto) | Representa el acceso seguro a la plataforma, como abrir la puerta de un aula virtual. |
| Página Principal | 🏠 (casa) | Espacio central de trabajo, como el escritorio donde se organizan las actividades del semestre. |
| Crear Nueva Tarea | ✏️ (lápiz o pluma) | Herramienta para diseñar nuevas actividades académicas, como redactar un plan de clase. |
| Visualizar Tareas | 📋 (hoja de tareas) | Lista de actividades por hacer, como una agenda diaria de compromisos académicos. |
| Entrega de Evidencias | 📤 (bandeja de salida) | Acción de subir los trabajos finales o entregables, como entregar un informe a un profesor. |
| Ver Retroalimentación | 💬 (bocadillo de chat) | Representa los comentarios y sugerencias del docente, como recibir observaciones durante una tutoría. |
| Exportar Reportes | 📄 (hoja con flecha) | Acción de guardar la información de tareas o desempeño en formatos descargables, como imprimir un boletín académico. |
| Historial de Tareas | 🗂️ (carpeta) | Registro organizado de todas las actividades académicas realizadas, como un archivo histórico del semestre. |
| Cerrar Sesión | 🔓 (candado cerrado) | Acción de salir de la plataforma de forma segura, como cerrar la puerta del aula virtual. |
| Navegación del Menú | 📑 (marcador) | Representa la guía lateral para moverse entre secciones, como los capítulos de un libro. |
| Notificaciones | 🔔 (campana) | Avisos importantes sobre entregas próximas o recordatorios, como un aviso de profesor en clase. |
| Calendario | 📆 (calendario) | Visualización de fechas clave de las tareas, como un horario de clases universitario. |
| Carga de Archivos | 📎 (clip) | Adjuntar evidencias o recursos, como enviar anexos por correo electrónico. |

# Conexión del Frontend y Backend







El sistema utiliza una arquitectura basada en **Vite.js** como herramienta de construcción (build tool) para el frontend y, presumiblemente, un servidor backend (API) para la persistencia y gestión de datos.

## Frontend (Cliente)

* Stack: React.js, CSS (Tailwind, clases personalizadas).

### Estructura:

* App.jsx / main.jsx: Archivos principales de React para el renderizado de la aplicación.
* Index.css: Archivo de estilos globales y personalización (ej.: .task-card, .gradient-bg, etc.).

### Plugins de Visual Editor:

* Visual-editor-config.js y edit-mode-script.js contienen componentes de edición visual, configurando estilos de ventanas emergentes (pop-ups) y botones de interacción.
* Uso de getPopupHTMLTemplate para renderizar HTML dinámico de componentes de edición (saveLabel, cancelLabel).

## Backend (Servidor)

La estructura sugiere un API RESTful que:

* Expone endpoints para autenticación de usuarios (iniciar sesión, cerrar sesión).
* Provee rutas para crear, actualizar y eliminar tareas.
* Gestiona la persistencia en base de datos (usuarios, tareas, entregas, calificaciones).

## Flujo de Conexión

* Solicitud de Tareas (GET): El **frontend** realiza solicitudes a un endpoint REST (/api/tasks o similar) usando fetch o axios. Las tareas se renderizan dinámicamente en el panel del estudiante o docente.
* Creación de Tareas (POST): Desde los formularios de "Crear Nueva Tarea" (panel docente), se envía un POST con los datos de la tarea: título, descripción, fecha límite y curso.
* Entrega de Evidencias (POST / PUT): Cuando el estudiante sube una evidencia, el cliente envía una solicitud con el archivo adjunto y los metadatos asociados (ID de la tarea, usuario que entrega).
* Actualización de Estados (PATCH / PUT): El backend actualiza el estado de las tareas (Entregada, Vencida, Pendiente), que se refleja en la interfaz (colores y etiquetas).
* **Autenticación y Sesión (POST / DELETE)**: Las credenciales del usuario se envían al servidor para validar el acceso. Uso de tokens o sesiones.

## Seguridad y Validación

* Uso de claves criptográficas y HTTPS para proteger la transmisión de datos sensibles.
* Los endpoints validan los roles (estudiante y docente) para permitir solo acciones permitidas.
* El archivo package-lock.json muestra dependencias modernas, reforzando la integridad del proyecto.

## Estilos Dinámicos y Personalización

**index.css** contiene clases personalizadas para mejorar la experiencia de usuario:

* gradient-bg: Fondo con gradiente para la estética general.
* glass-card: Efecto de vidrio para tarjetas informativas.
* task-card: Estilos específicos para las tarjetas de tareas, con hover dinámico (transform: translateY(-2px) y box-shadow para resaltar).

Estas clases mejoran la legibilidad, accesibilidad y atractivo visual.

## Herramientas y Dependencias

* Vite: Herramienta de bundling rápida para entornos de desarrollo y producción.
* Tailwind CSS: Framework para estilos utilitarios y responsivos.
* postcss.config.js y tailwind.config.js: Personalización y procesamiento de CSS.
* package.json / package-lock.json: Mantienen las dependencias actualizadas (babel, React, etc.).

## Buenas Prácticas Observadas

Separación de responsabilidades:

* Código de estilos (index.css).
* Código de renderizado de React (App.jsx, main.jsx).
* Plugins específicos (visual-editor-config.js).
* Código modular y reutilizable (componentes de popup, clases de tarjetas).  
  Uso de vite.config.js para optimización de rendimiento.