

# Introducción

AI Diary es una aplicación web diseñada como un diario personal asistido por inteligencia artificial. Su objetivo es permitir a los usuarios registrar y recuperar entradas diarias, interactuar con un chatbot y gestionar su perfil de usuario. Este proyecto forma parte del examen final de la asignatura Bases de Datos I y busca demostrar la aplicación de PostgreSQL junto con un backend basado en Flask.

# Descripción del Proyecto

AI Diary es una aplicación web desarrollada con Flask que integra una base de datos PostgreSQL para almacenar datos de usuarios y entradas de diario. Los usuarios pueden interactuar con un chatbot para agregar nuevas entradas o recuperar información de fechas específicas. Además, la aplicación incluye un sistema de gestión de perfiles donde los usuarios pueden actualizar su información personal.

# Importancia del Proyecto

Este proyecto aborda varios problemas comunes relacionados con la organización personal y el registro de experiencias diarias:

* **Falta de un registro estructurado**: AI Diary permite a los usuarios guardar y recuperar sus pensamientos y experiencias diarias de forma organizada.
* **Dificultad para recordar eventos pasados**: Mediante el chatbot, los usuarios pueden consultar sus entradas previas sin necesidad de buscarlas manualmente.
* **Gestión de información personal**: La aplicación ofrece un sistema seguro de autenticación y gestión de perfiles, garantizando la protección de datos.
* **Accesibilidad y facilidad de uso**: Gracias a su interfaz intuitiva, cualquier usuario puede interactuar sin necesidad de conocimientos técnicos avanzados.

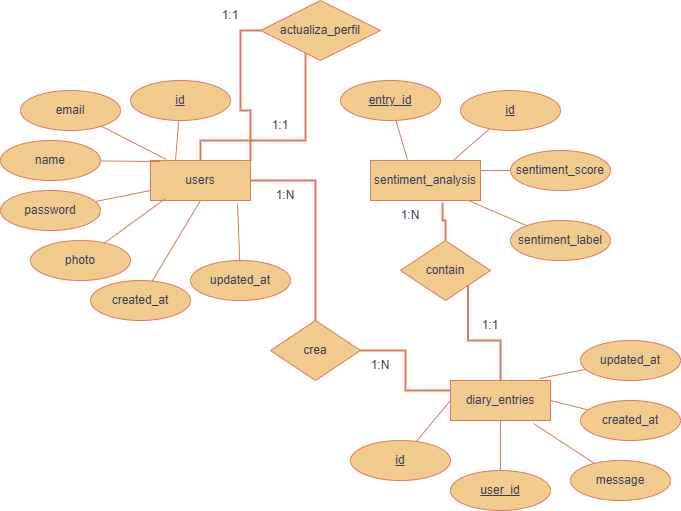
# Funcionalidades

1. **Registro y Autenticación de Usuarios:**
   1. Permite a los usuarios registrarse e iniciar sesión de manera segura.
   2. Las contraseñas se almacenan de forma segura mediante hashing y se agrega un gensalt para mayor seguridad.
2. **Entradas de Diario:**
   1. Los usuarios pueden escribir y guardar entradas diarias.
   2. Se pueden recuperar entradas específicas mediante comandos como "quiero saber mis entradas del día yyyy-mm-dd".
3. **Interacción con Chatbot:**
   1. Facilita la recuperación de entradas y la adición de nuevas mediante un chatbot interactivo.
4. **Gestión de Perfil:**
   1. Los usuarios pueden actualizar su nombre, correo electrónico, contraseña y foto de perfil.
   2. La contraseña solo se actualiza si el usuario proporciona una nueva.
   3. La foto de perfil se carga a través de una URL y se muestra en la barra de navegación.

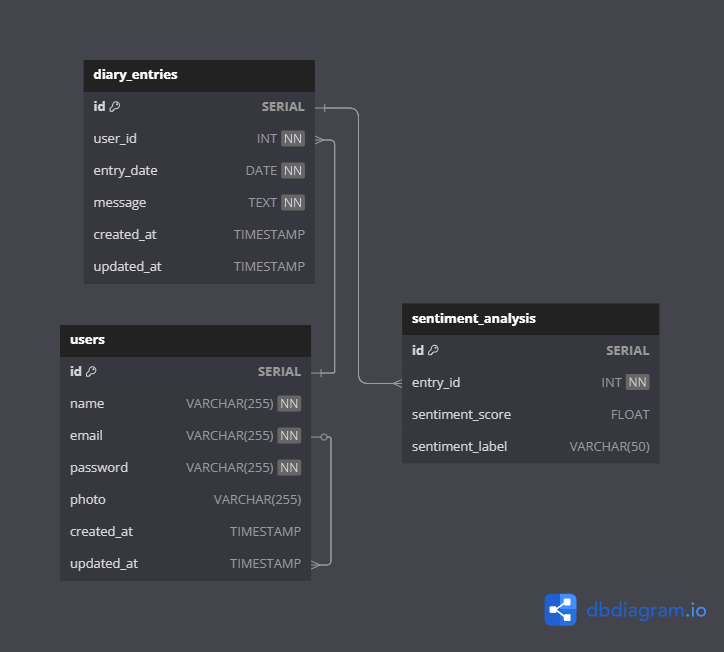
# Modelo Relacional y Entidad-Relación

La base de datos de AI Diary está diseñada con PostgreSQL y sigue un modelo relacional optimizado para la gestión de usuarios y entradas de diario.

* **Modelo Entidad-Relación:** Se representa mediante diagramas que incluyen entidades como "Usuario" y "Entrada", con relaciones entre ellas.



**Modelo Relacional:** Incluye las tablas "usuarios" y "entradas", con claves primarias y foráneas que garantizan la integridad de los datos.



# Tecnologías Utilizadas

* **Frontend:** HTML, CSS, JavaScript.
* **Backend:** Flask (Python).
* **Base de Datos:** PostgreSQL.
* **Autenticación:** Flask Sessions y hashing de contraseñas.
* **Chatbot:** Implementado con lógica personalizada en Flask.

Arquitectura Utilizada y Justificación

Se utilizó una arquitectura cliente-servidor con Flask como backend ligero y PostgreSQL como base de datos relacional. La elección de Flask se debe a su facilidad de implementación y flexibilidad, mientras que PostgreSQL fue seleccionado por su robustez y escalabilidad para el manejo de datos estructurados.

## Uso de POST para Creación y Actualización

**POST para Creación:** Se utiliza para enviar datos al servidor con el objetivo de agregar nuevos registros, como entradas de diario o usuarios. Este método es apropiado porque POST se emplea para crear nuevos recursos en el servidor de manera eficiente.

**POST para Actualización (en lugar de PUT):** Aunque PUT es la opción estándar para actualizar recursos completos, en este proyecto se usa POST también para actualizaciones. Esto se debe a que la actualización de algunos recursos podría implicar modificaciones parciales o la incorporación de nuevos datos a un recurso existente, lo cual se maneja adecuadamente mediante POST. Esto proporciona flexibilidad para manejar actualizaciones sin la necesidad de sobrescribir completamente el recurso.

**DELETE para Eliminación:** Se utiliza para eliminar recursos, como entradas de diario o cuentas de usuario, garantizando que los registros se borren de manera eficiente desde la base de datos.

Conclusión

AI Diary es una aplicación web que combina la inteligencia artificial con la gestión de bases de datos para proporcionar una experiencia de usuario enriquecida. Su diseño seguro y su arquitectura escalable garantizan un sistema eficiente para la gestión de diarios personales. Este proyecto demuestra la aplicación práctica de PostgreSQL y Flask en un entorno de desarrollo real, proporcionando una base sólida para futuras mejoras y expansiones.