

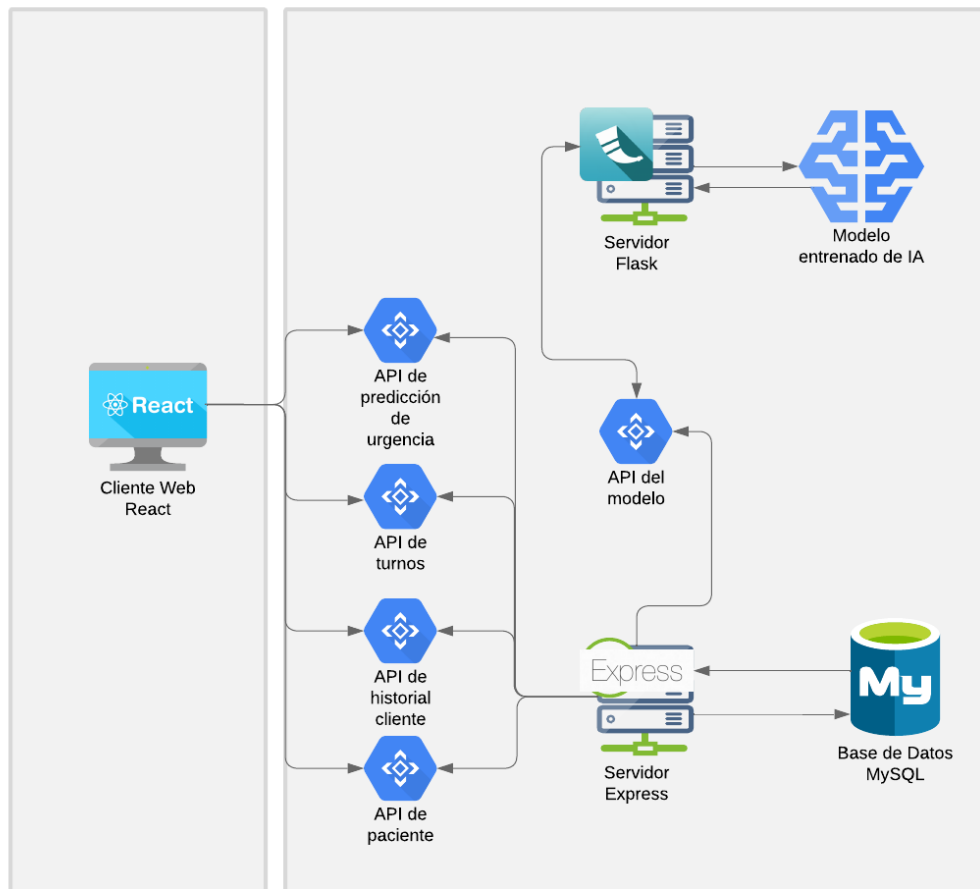
Universidad de las Fuerzas Armadas "ESPE"

# Diseño del Proyecto

Diagrama de Arquitectura

Dylan Tipán, Camila Rivera  
25-11-2024

# 1. Diagrama de Arquitectura



El sistema está compuesto por una arquitectura cliente-servidor distribuida en varios servicios, donde cada componente cumple una función específica. La interacción entre las diferentes capas permite gestionar pacientes, turnos, historias clínicas y realizar predicciones basadas en un modelo de inteligencia artificial (IA).

## Componentes del Diagrama

### 1. Cliente Web React:

- **Función:** Representa la interfaz gráfica del sistema, desde donde los usuarios (como médicos, enfermeros o administradores) interactúan con el sistema.
- **Interacciones:** Se conecta con las APIs del backend para acceder y gestionar los datos de pacientes, historias clínicas, turnos, y realizar predicciones de urgencia.

2. **APIs Backend (Express):** Estas APIs intermedian las solicitudes del cliente web y los servicios de base de datos o de inteligencia artificial.

- **API de predicción de urgencia:**

- **Función:** Determina la urgencia de atención de un paciente en base a parámetros proporcionados.
- **Interacciones:** Llama al servidor Flask, que aloja el modelo de IA entrenado.

- **API de turnos:**

- **Función:** Permite gestionar las citas médicas (creación, actualización, eliminación o consulta).
- **Relación:** Asociada a la clase TURNO.

- **API de historial cliente:**

- **Función:** Proporciona acceso al historial clínico de un paciente.
- **Relación:** Asociada a la clase HISTORIA.

- **API de paciente:**

- **Función:** Permite gestionar la información de los pacientes.
- **Relación:** Asociada a la clase PACIENTE.

3. **Servidor Flask:**

- **Función:** Aloja y ejecuta el modelo de inteligencia artificial entrenado que se utiliza para realizar predicciones de urgencia.
- **Interacciones:** Recibe datos procesados desde el servidor Express, ejecuta predicciones y retorna los resultados.

4. **Base de Datos MySQL:**

- **Función:** Almacena los datos estructurados de pacientes, historias clínicas y turnos.
- **Relaciones:**
  - La tabla de PACIENTE se relaciona con:
    - La tabla de HISTORIA mediante el atributo pac\_id (1 a N).
    - La tabla de TURNO mediante el atributo pac\_id (1 a N).

- La tabla de HISTORIA contiene detalles como diagnósticos y observaciones relacionados con un paciente específico.
- La tabla de TURNO gestiona las citas programadas y su estado.

### **Flujo General del Sistema**

1. Un usuario accede al sistema desde la **interfaz React**.
2. Las solicitudes realizadas desde la interfaz son procesadas por las **APIs del servidor Express**, que actúan como intermediarios.
3. Si la solicitud requiere predicciones de IA, la API de predicción de urgencia interactúa con el **servidor Flask** para obtener resultados del modelo entrenado.
4. Para la gestión de datos, las APIs acceden directamente a la **base de datos MySQL** para almacenar o consultar información.