### **Modelo de Arquitectura**

#### **1. Frontend (Interfaz de Usuario)**

* **Lenguaje/Frameworks**: HTML, CSS, JavaScript.
* **Funcionalidad**:
  + Permite a los usuarios (Médicos, Pacientes, Farmacéuticos, Administrador) acceder a la página web.
  + Interactúa con el backend para registrar medicamentos, alergias, y otra información médica.
* **Conexión**: Envía solicitudes HTTP al backend y recibe datos a través de la API.

#### **2. Backend (Lógica del Negocio)**

* **Lenguaje/Frameworks**: Node.js, Python u otro lenguaje como Java.
* **Funcionalidad**:
  + Controla la lógica del negocio, como autenticación de usuarios, manejo de sesiones y consultas a la API de medicamentos.
  + Gestiona la interacción entre la base de datos y el frontend.
* **Conexión**: Consume la API externa de medicamentos y procesa la información.

#### **3. API de Medicamentos**

* **Funcionalidad**:
  + Proporciona la información actualizada de medicamentos.
  + El backend envía solicitudes a esta API para obtener los datos necesarios.

#### **4. Base de Datos (AWS RDS)**

* **Funcionalidad**:
  + Almacena la información de los usuarios (Médicos, Pacientes ,Farmacéuticos y Administrador), los medicamentos registrados, alergias, etc.
  + Garantiza la seguridad de la información (cifrado y control de acceso).

#### **5. AWS (Amazon Web Services)**

* **Servicios utilizados**:
  + **AWS S3**: Para almacenar recursos estáticos como imágenes o archivos.
  + **AWS RDS**: Para la base de datos.
  + **AWS API Gateway**: Para gestionar las solicitudes a la API de medicamentos.
  + **AWS Lambda (opcional)**: Para funciones sin servidor si deseas un backend serverless.
  + **AWS EC2**: Para desplegar el backend si no utilizas Lambda.
  + **AWS Route 53**: Para gestionar el dominio y la URL.
  + **AWS IAM**: Para gestionar la seguridad de acceso.

#### **6. Seguridad**

* **Autenticación**: Puede implementarse con **AWS Cognito** para autenticar a los usuarios.
* **Cifrado**: Para garantizar que los datos en tránsito (HTTPS) y en reposo (cifrado de base de datos) estén seguros.
* **Control de acceso**: Roles y permisos diferenciados para Médicos, Pacientes y Farmacéuticos.

### **Flujo General:**

1. El usuario accede a la página web.
2. El frontend envía solicitudes al backend para recuperar y actualizar la información médica.
3. El backend se comunica con la API de medicamentos y la base de datos para obtener o almacenar información.
4. Los datos se procesan y se devuelven al frontend.
5. El sistema está alojado en AWS, utilizando servicios como EC2 o Lambda para el backend y S3 para los recursos estáticos.
6. Los usuarios interactúan con la plataforma a través de una URL gestionada por AWS Route 53.