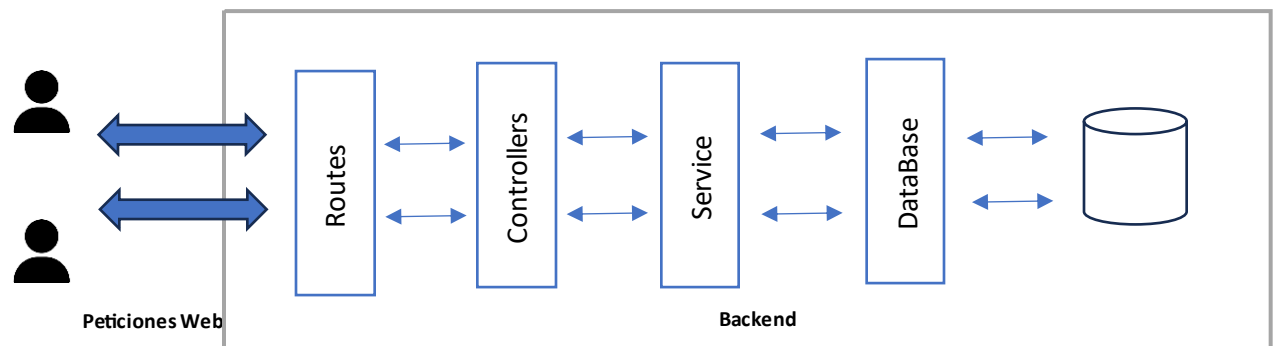


## Documento de Arquitectura

**Sistema:** Proyecto de gestión de un restaurante.

**Objetivos:** una mejor organización del código y una mayor escalabilidad.

**Arquitectura en capas:**



- Capa de rutas (Routes): Esta capa es responsable de definir las rutas y las acciones que se ejecutarán cuando se reciban solicitudes HTTP en cada ruta. Utiliza un enrutador (como Express.js) para gestionar las diferentes rutas y sus controladores.
- Capa de controladores (Controllers): Los controladores manejan las solicitudes entrantes desde las rutas. Su función es procesar la entrada del cliente, interactuar con la lógica de negocio (servicios) y enviar una respuesta adecuada al cliente.
- Capa de servicios (Services): La capa de servicios contiene la lógica de negocio principal del restaurante. Aquí es donde se implementan las funciones para gestionar las operaciones principales de la aplicación, como crear, leer, actualizar y eliminar datos. Los servicios interactúan con la capa de persistencia (base de datos) y encapsulan la lógica de negocio compleja.
- Capa de persistencia (Database): Esta capa es responsable de interactuar directamente con la base de datos. Puedes utilizar una biblioteca de acceso a la base de datos (como Sequelize para SQL o Mongoose para MongoDB) para facilitar la comunicación y las operaciones con la base de datos.
- Capa de modelos (Models): Los modelos representan la estructura de los datos almacenados en la base de datos. Cada modelo suele tener una correspondencia directa con una tabla en una base de datos relacional o un documento en una base de datos NoSQL.

**Otras Capas:**

Capa de utilidades (Utilities): Esta capa puede contener funciones de utilidad que se utilizan en varias partes de la aplicación. Por ejemplo, funciones para el manejo de errores, validación de datos, formateo de respuestas, etc.

Capa de autenticación y seguridad (Authentication & Security): Si el proyecto requiere autenticación de usuarios y medidas de seguridad, esta capa sería responsable de implementar esas funcionalidades. Puedes utilizar técnicas como JWT (JSON Web Tokens) para la autenticación y autorización.