

Documento de Arquitectura: Patrón de Diseño

1. Introducción

En este documento se describe el patrón de arquitectura utilizado en el desarrollo de la aplicación Angular. El objetivo de este documento es comunicar la estructura general del sistema y sus componentes principales a las partes interesadas del proyecto.

2. Patrón de Arquitectura

El patrón de arquitectura utilizado en esta aplicación es **MVC (Modelo-Vista-Controlador)** combinado con la arquitectura en capas de Angular.

2.1 Arquitectura MVC

El patrón MVC, como se mencionó anteriormente, separa la aplicación en tres componentes principales:

- **Modelo:** El modelo representa los datos de la aplicación y la lógica empresarial.
- **Vista:** La vista es la interfaz de usuario de la aplicación.
- **Controlador:** El controlador es el intermediario entre el modelo y la vista.

2.2 Arquitectura en Capas de Angular

Angular utiliza una arquitectura en capas para organizar el código de la aplicación. Las capas principales son:

- **Capa de componentes:** Los componentes son los bloques de construcción de la interfaz de usuario de la aplicación. Cada componente tiene su propia plantilla HTML, código TypeScript y lógica de negocio.
- **Capa de servicios:** Los servicios proporcionan una forma de compartir lógica y datos entre componentes. Los servicios se inyectan en los componentes utilizando el mecanismo de inyección de dependencias de Angular.
- **Capa de almacenamiento:** La capa de almacenamiento se encarga de almacenar y recuperar datos. Angular proporciona varios mecanismos de almacenamiento, como el almacenamiento local y el almacenamiento web.

3. Integración de MVC y Arquitectura en Capas

En la aplicación Angular, el patrón MVC se implementa utilizando las capas de componentes y servicios.

- **Componentes:** Los componentes representan la vista del patrón MVC. Los componentes contienen la plantilla HTML que se muestra al usuario y la lógica de negocio para manejar los eventos del usuario.
- **Servicios:** Los servicios representan el modelo y el controlador del patrón MVC. Los servicios contienen la lógica de negocio para acceder a los datos, procesar datos y actualizar la vista.

4. Beneficios del Patrón de Arquitectura

El patrón de arquitectura MVC combinado con la arquitectura en capas de Angular ofrece varios beneficios, entre los que se incluyen:

- **Modularidad:** El patrón MVC y la arquitectura en capas dividen la aplicación en módulos independientes, lo que facilita su desarrollo y mantenimiento.
- **Reutilización de código:** Los componentes y servicios se pueden reutilizar en diferentes partes de la aplicación, lo que reduce la cantidad de código que se debe escribir.
- **Testability:** El código es más fácil de probar porque los componentes y servicios están bien definidos y aislados.
- **Escalabilidad:** La aplicación es más fácil de escalar a medida que crece la base de usuarios o los datos.

5. Diagramas de Arquitectura

Para ilustrar mejor la arquitectura del sistema, se pueden incluir diagramas como:

- **Diagrama UML de componentes:** Muestra las clases del sistema y sus relaciones.
- **Diagrama de flujo de datos:** Muestra el flujo de datos entre los componentes del sistema.
- **Diagrama de secuencia:** Muestra la secuencia de interacciones entre los componentes del sistema.

6. Conclusión

El patrón de arquitectura MVC combinado con la arquitectura en capas de Angular es un patrón de diseño sólido y probado que puede utilizarse para desarrollar aplicaciones web escalables y mantenibles. Al documentar el patrón de arquitectura utilizado en el proyecto, se puede mejorar la comunicación y la colaboración entre las partes interesadas del proyecto, y se puede facilitar el mantenimiento y la evolución del sistema en el futuro.