Arquitectura Front

ARQUITECTURA CLIENTE-SERVIDOR DISTRIBUIDO CERTIFICADO PARA MICROFRONTENDS

Acercad de:

1. **Estructura del Proyecto Angular**:

    1. En la raíz del proyecto, tendrás la configuración de Angular CLI y otros archivos de configuración necesarios.

    1. La carpeta `src` contendrá todos los archivos fuente de tu aplicación Angular.

2. **Módulos de Funcionalidad**:

    1. Organiza tu aplicación en módulos de Angular que representen las principales funcionalidades o características de la aplicación. Cada módulo puede corresponder a un microfrontend potencial en el futuro.

    1. Utiliza el enrutador de Angular para gestionar la navegación entre los diferentes módulos y componentes, sin embargo en la aplicación se implemente SPA para la navegación por pestaña levantando un componente cada vez que se requiera.

3. **Componentes Reutilizables**:

    1. Crea componentes reutilizables que puedan ser compartidos entre los diferentes módulos de la aplicación. Estos componentes pueden incluir elementos de interfaz de usuario comunes, como barras de navegación, barras laterales, formularios, etc.

    1. Estos componentes son en su mayoría independientes y desacoplados, lo que facilitará su reutilización en diferentes partes de la aplicación y su posible migración a microfrontends en el futuro.

4. **Servicios para la Comunicación con el Servidor**:

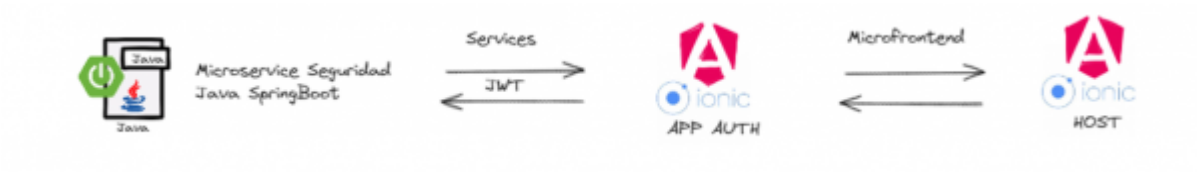
    1. Implementa servicios en Angular para la comunicación con el servidor y la gestión de datos. Estos servicios pueden encapsular las llamadas a APIs proporcionadas por los microservicios en el backend.

    1. Si decides introducir un BFF en el futuro, puedes adaptar estos servicios para comunicarse con el BFF en lugar de acceder directamente a los microservicios.

5. **Capa de Presentación y Lógica de Negocio**:

    1. Utiliza los componentes de Angular para implementar la interfaz de usuario y la lógica de presentación de tu aplicación.

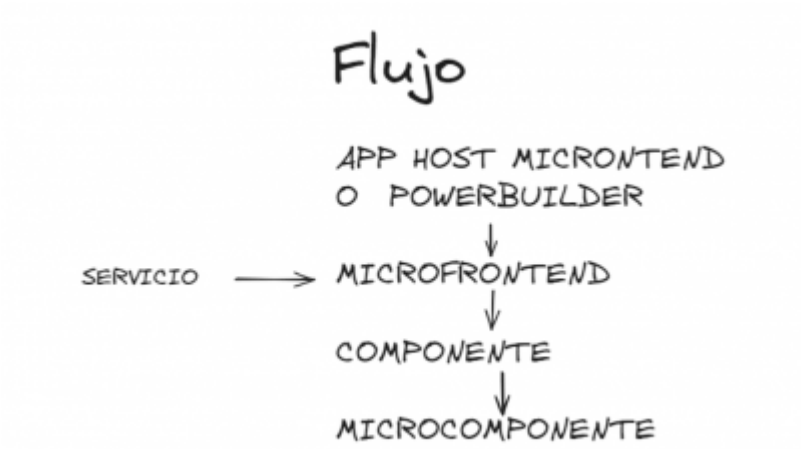
    1. Mantén la lógica de negocio separada de los componentes de presentación, siguiendo los principios de separación de preocupaciones (SoC). Esto facilitará la escalabilidad y la evolución de tu aplicación.



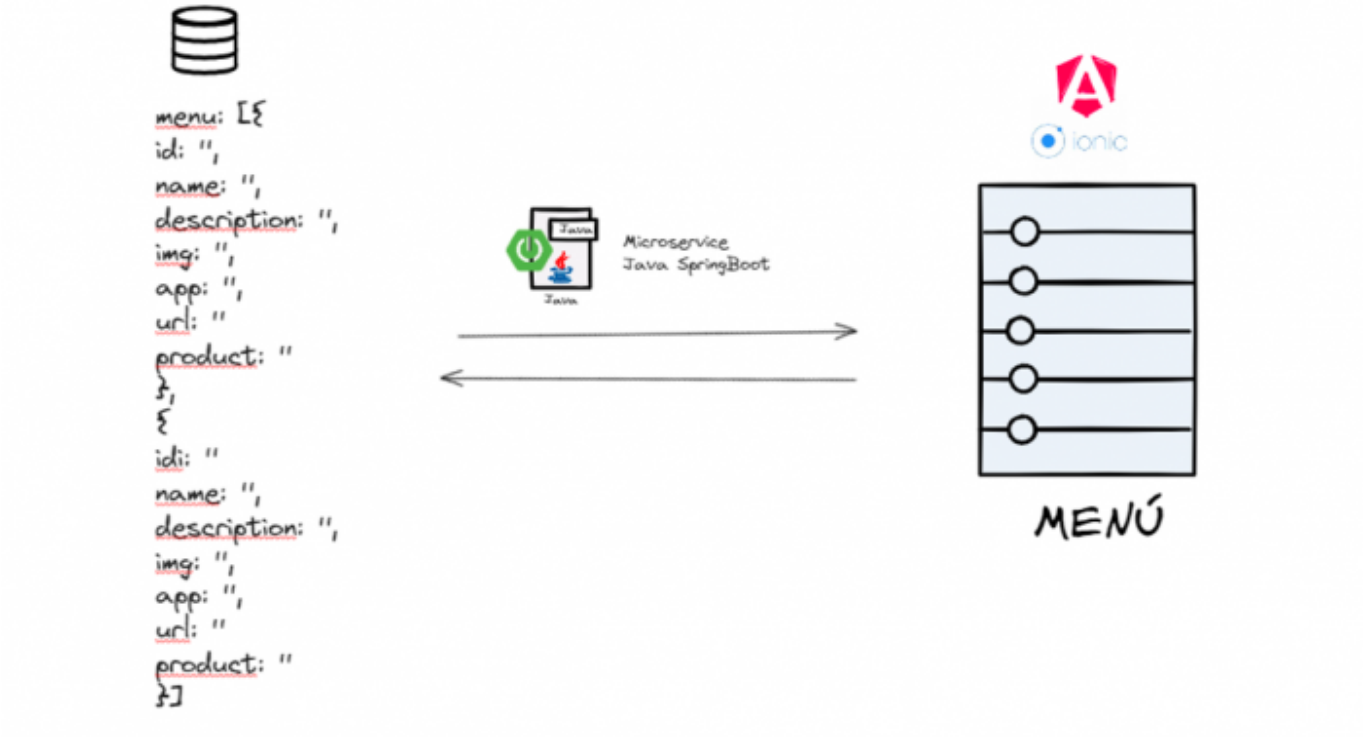
ARQUETIPO MFSICOF

```
- src/
|- app/
   |- layout/
   |- api/
   |- config/
   |- service/
|- shared/
   |- api/
   |- components/
   |- service/
   |- directives/
   |- models/
   |- onecomponents/
   |- footer/
   |- nav-bar/
|- app-component.css
|- app-component.html
|- app-component.ts
|- app-config.ts
|- app-routes.ts
```

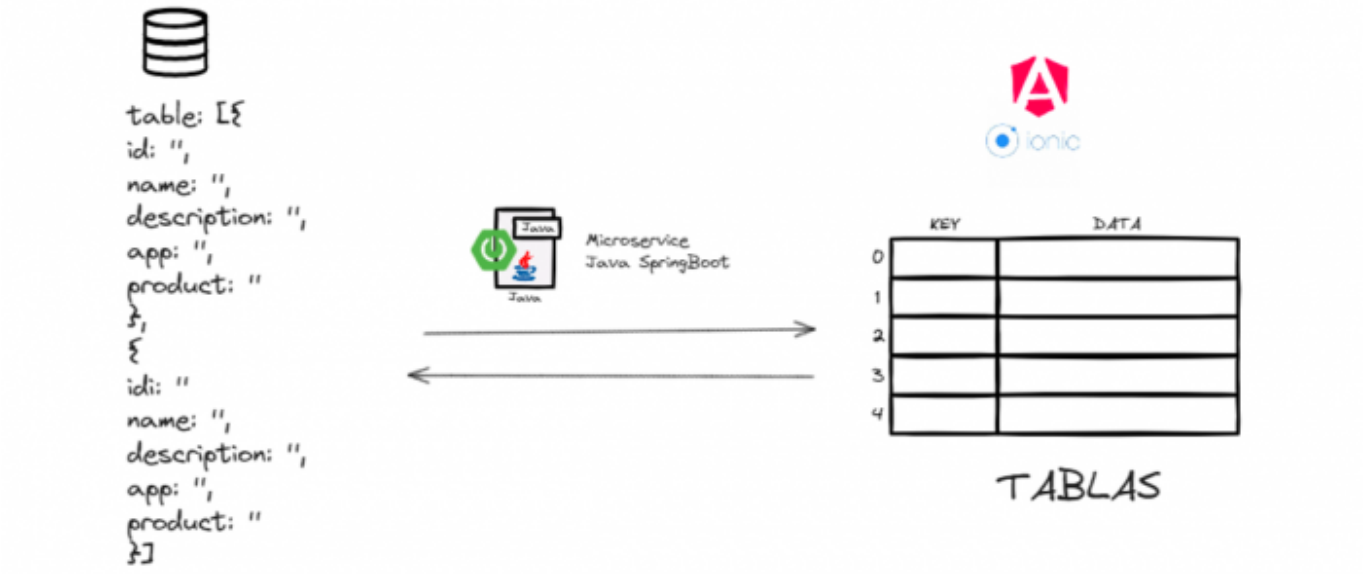
```
- environments/
|- environment.dev.ts
|- environment.prod.ts
|- environment.qa.ts
|- environment.ts
```



Ejemplo de flujo entre api-componentes



Ejemplo formularios



Ejemplo tablas

From:
https://wiki.adacsc.co/ - **Wiki**

Permanent link:
https://wiki.adacsc.co/doku.php?id=ada:howto:sicoferp:factory:new-migracion-sicoferp:front:arquitectura&rev=1718984740

Last update: **2024/06/21 15:45**