*OIT Frame*

Utilizzate per le variabili in ogni punto possibile i signal anche quelli di “input” e “output”.

Per recuperare la versione dal package.json e recuperarla in “appComponent”, nel file “tsconfig.json” devo aggiungere le seguenti opzioni:

{

"compilerOptions": {

// ... altre opzioni ...

"resolveJsonModule": true,

"esModuleInterop": true,

"allowSyntheticDefaultImports": true

}

}

**🎨 Caratteristiche implementate:**

**✅ Navbar (Top)**

* Mat-toolbar con colore primary
* Pulsante menu per aprire/chiudere la sidebar
* Icona utente a destra
* Fixed in alto

**✅ Sidebar a scomparsa**

* Mode "over" (si sovrappone al contenuto)
* **Menu diviso in due sezioni:**
  + **Back-office**: Dashboard, Utenti, Impostazioni, Report
  + **Front-office**: Home, Prodotti, Contatti, Chi Siamo
* Icone Material per ogni voce
* Larghezza fissa di 256px (w-64)

**✅ Footer (Bottom)**

* Mat-toolbar fixed al bottom
* Copyright e versione dell'app
* Testo responsive

**✅ Contenuto principale**

* Area scorrevole tra navbar e footer
* Layout responsive con grid Tailwind
* Padding adattivo (p-4 su mobile, p-6 su tablet, p-8 su desktop)

**📱 Responsive**

* Grid che si adatta: 1 colonna su mobile, 2 su tablet, 3 su desktop
* Testi con dimensioni responsive
* Padding e margini adattivi

**🔧 Per utilizzare il codice:**

1. **Installa le dipendenze:**

shell

ng add @angular/material

npm install tailwindcss @tailwindcss/postcss postcss --force

1. **Configura tailwind**:

Create a .postcssrc.json file in the root of your project and add the @tailwindcss/postcss plugin to your PostCSS configuration.

{

"plugins": {

"@tailwindcss/postcss": {}

}

}

se volessi gestire navbar, sidebar e footer in componenti come riscriveresti AppComponent?

**🏗️ Architettura a componenti**

**📦 Componenti creati:**

1. **NavbarComponent**
   * Gestisce solo la barra superiore
   * Emette un evento menuToggle quando si clicca sul menu
   * Standalone component
2. **SidebarComponent**
   * Riceve i menu items tramite @Input()
   * Gestisce la visualizzazione del menu diviso in sezioni
   * Aggiunto routerLinkActive per evidenziare la pagina corrente
3. **FooterComponent**
   * Riceve appName e version come parametri configurabili
   * Anno corrente calcolato automaticamente
4. **AppComponent** (semplificato)
   * Coordina i componenti figli
   * Gestisce solo la logica del toggle della sidebar con @ViewChild
   * Template molto più pulito e manutenibile

**✨ Vantaggi di questa struttura:**

* **Riusabilità**: Ogni componente può essere riutilizzato
* **Manutenibilità**: Più facile modificare un singolo componente
* **Testabilità**: Ogni componente può essere testato separatamente
* **Separazione delle responsabilità**: Ogni componente ha un compito specifico
* **Type-safe**: Interface MenuItem per tipizzazione forte

**🎨 Miglioramenti aggiunti:**

* Icona notifiche nella navbar
* Transizioni smooth al hover
* routerLinkActive per evidenziare il menu attivo
* Supporto completo per router-outlet

Vuoi che aggiunga anche un service per gestire lo stato della sidebar o altre funzionalità?