Code Style & Conventions

TypeScript Style Guide

Follow Angular's TypeScript style guide:

- Use 2 spaces for indentation
- Use single quotes for strings
- Use camelCase for variables and functions
- Use PascalCase for classes and interfaces
- Use UPPER_CASE for constants

Component Structure

```
@Component({
  selector: 'app-example',
  templateUrl: './example.component.html',
  styleUrls: ['./example.component.css']
})
export class ExampleComponent implements OnInit {
  // 1. Public properties
  public title: string;

  // 2. Private properties
  private data: any[];

  // 3. Constructor with DI
  constructor(
    private apiService: ApiService,
    private router: Router
  ) {}

  // 4. Lifecycle hooks
  ngOnInit(): void {
    this.loadData();
```

```
  }

  // 5. Public methods
  loadData(): void {
    // Implementation
  }

  // 6. Private methods
  private processData(data: any): void {
    // Implementation
  }
}
```

Naming Conventions

- **Components**: `feature-name.component.ts`
- **Services**: `feature-name.service.ts`
- **Pipes**: `pipe-name.pipe.ts`
- **Guards**: `guard-name.guard.ts`
- **Interfaces**: `interface-name.ts`
- **Modules**: `module-name.module.ts`

Template Guidelines

```
<!-- Use semantic HTML -->
<header>
  <nav>
    <!-- Navigation -->
  </nav>
</header>

<!-- Use Angular directives -->
<div *ngIf="isLoading">Loading...</div>
<ul>
  <li *ngFor="let item of items">{{ item.name }}</li>
</ul>

<!-- Use event binding -->
```

```
<button (click)="handleClick()">Click Me</button>

<!-- Use property binding -->
<img [src]="imageUrl" [alt]="imageAlt">

<!-- Use two-way binding -->
<input [(ngModel)]="searchTerm">
```

SCSS/CSS Guidelines

- Use component-scoped styles
- Follow BEM naming convention for complex components
- Use Angular Material theming
- Avoid `!important` unless absolutely necessary

Revision #1
Created 24 February 2026 07:27:14 by ondeliveloper
Updated 24 February 2026 07:27:46 by ondeliveloper