Module Title : Angular Intermediate–Advanced Training
Duration : 4 days

Overview

This 4-day instructor-led course is designed for developers who have a working knowledge of Angular and want to master intermediate to advanced techniques, including Signals, Effects, Change Detection optimization, and new features from Angular 17–19.

The course covers performance tuning, server-side rendering, and enterprise integration patterns including state management with NgRx Signals Store and Web Component interoperability.

Course Objectives

By the end of the course, participants will be able to:

- Apply best practices for scalable component architecture
- Master Angular's change detection and reactivity using Signals and Effects
- Implement advanced forms and routing techniques
- Optimize Angular apps for performance and SSR
- Build and test reactive state using NgRx Signals Store
- Export Angular components as Web Components

Prerequisites

Participants should have:

- 6+ months experience with Angular (v12+)
- Solid understanding of:
    - TypeScript fundamentals (interfaces, types, async/await)
    - Angular components, templates, services
    - Angular modules and dependency injection
    - Basic RxJS operators (map, switchMap, combineLatest)
    - Routing and form basics (template/reactive)

Not Required: Prior experience with Signals, SSR, or NgRx (covered in training)

Expected Outcomes

Participants will:

- Understand and apply Signal-based reactivity in Angular 17–19
- Build fully modular apps with SSR and Web Components
- Write testable, optimized, scalable Angular applications
- Confidently migrate or integrate with legacy systems using Signals

Course Outline

Day 1: Intermediate Angular Foundations & Component Architecture

Topics

- Component communication (Input/Output, ViewChild, shared services)
- Change Detection: Default vs OnPush
- Structural directives & control flow
- Smart vs Dumb components
- Feature, Shared, and Core Modules
- Build a smart/dumb component structure with OnPush change detection.
- Demonstrate efficient change detection using trackBy.

Day 2: Routing, Forms & Scalable State

Topics

- Advanced Reactive Forms
  - FormBuilder, FormArray, Async Validators
- Routing
  - Lazy loading, nested routes, guards, resolvers
- Global and local state management
  - RxJS state vs Service state
  - Introduction to Signals as state
- Create a form with nested FormGroups and custom validators
- Lazy-load a feature module with guards and preloading strategy

Day 3: Signals, Effects & Angular 17–19 Features

Topics

Angular Signals (Angular 16–19)

- signal(), computed(), effect()

- untracked() and cleanup() for lifecycle control
- Using signals in templates and services
- Signal Inputs (Angular 17+)
- Best practices and pitfalls

Angular 17–19 Features

- Standalone components & routing
- Control flow syntax (@if, @for)
- Zoneless Angular
- Signal reactivity enhancements in Angular 19
- Build system upgrades
- Convert existing service/observable state to Signals + Computed + Effects
- Use @input({ signal: true }) in a component to consume signals natively

Day 4: Performance, SSR, Testing & Advanced Integration

Topics

Performance & Testing

- Standalone optimization
- SSR (Angular Universal): hydration + pre-rendering
- Signals and zoneless SSR compatibility
- Testing Signals: unit tests and reactive flows

Advanced Topics

1. Signals with SSR (Server Side Rendering)

- How Signals behave in SSR
- Best practices for hydration
- Using toSignal() in server-rendered templates
- Challenges with Effects and lifecycle in SSR

2. NgRx Signals Store

- Overview of NgRx Signals Store
- Signal-based selectors and stores
- Migration from traditional NgRx to Signals-based approach
- When to choose Signals Store vs full NgRx

3. Web Components Integration

- Angular Elements with Standalone Components
- Using Signals in Angular Elements

- Embedding Signal-based Angular components in non-Angular apps
- Communication between Web Components and Angular components

Lab

- Implement SSR hydration with signal-based data
- Build a mini NgRx Signals Store
- Export a Signal-powered Angular component as a Web Component

Additional topics (Based on time)

- Migration checklist: from observables to signals
- Signals vs MobX/Reactivity in Vue/React
- Developer tooling updates (Angular DevTools, Signal Debugging)