Advanced Angular with Micro Front End Development

Course Duration: 80 Hours **Course code: AAMFED**

1. Course Overview

This advanced course is designed for developers who already have a working knowledge of Angular and want to deepen their expertise in building large-scale, enterprise-ready applications. You'll learn advanced Angular techniques, performance optimizations, state management strategies, and then progress into Micro Frontend (MFE) architecture using Module Federation, Webpack 5, and modern deployment strategies. By the end of the course, you will be able to design, implement, and scale Micro Frontend applications using Angular and seamlessly integrate them within enterprise ecosystems.

2. What you'll learn?

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

- Master advanced Angular concepts (Change Detection, RxJS, Dynamic Components, DI).
- Build scalable and maintainable Angular applications.
- Implement state management using NgRx and Akita.
- Optimize Angular apps for performance and lazy loading.
- Understand the Micro Frontend architecture and its use cases.
- Use Webpack 5 Module Federation to create and integrate Angular MFEs.
- Develop independently deployable feature modules in Angular.
- Handle inter-application communication and shared state.
- Manage authentication and authorization across MFEs.
- Deploy and scale Micro Frontend apps on cloud and containerized environments.

3. Target Audience

- Angular developers aiming to upskill with enterprise-level architecture.

- Full stack developers building modular and scalable frontend systems.
- Architects and technical leads responsible for frontend modernization.
- Professionals preparing for Angular + Micro Frontend enterprise projects.

4. Pre-Requisites

Familiarity with:

- Angular fundamentals (Components, Services, Routing, Modules).
- TypeScript and ES6+ concepts.
- Basic knowledge of RxJS and Observables.
- Experience with REST APIs.

5. Course content

Day 1: Advanced Angular Architecture & Change Detection

Topics:

- Angular Application Architecture Review
- Deep Dive into Change Detection (Default vs OnPush, NgZone)
- Immutability and its effect on change detection

Lab:

- Create two components: one with default and one with OnPush
- Use NgZone.runOutsideAngular() to optimize a background task
- Log change detection using console.log() in ngDoCheck

Day 2: Reactive Forms and Custom Validation

Topics:

- Reactive Forms Setup and Data Flow
- FormGroup, FormControl, FormArray
- Custom Validators (sync & async)
- Dynamic Forms and Validation Messaging

Lab:

- Build a dynamic employee registration form
- Add field-level and form-level custom validators
- Implement async validator (e.g., check username availability)

Day 3: Advanced Routing and Navigation

Topics:

- RouterModule: forRoot, forChild
- Lazy Loading Feature Modules
- Route Guards: CanActivate, CanDeactivate, Resolve
- Custom Preloading Strategies
- Route Reuse Strategies

Lab:

- Lazy-load a feature module
- Add auth guard and resolver to routes
- Implement custom preloading logic to load modules based on metadata

Day 4: RxJS Mastery and State Management Patterns

Topics:

- Observable vs Subject vs BehaviorSubject
- Hot vs Cold Observables
- Common Operators: map, switchMap, mergeMap, concatMap, withLatestFrom
- Centralized vs Local Component State

Lab:

- Build a real-time filter/search UI using RxJS operators
- Create a shared service using BehaviorSubject for cross-component communication

Day 5: State Management with NgRx

Topics:

- NgRx Concepts: Store, Actions, Reducers, Selectors, Effects
- Handling Side Effects with Effects
- Entity Adapter and Entity State
- Debugging with NgRx DevTools

Lab:

- Create an Orders feature using NgRx (store setup + reducer logic)
- Trigger API calls via effects and display data using selectors
- Use NgRx DevTools to inspect state changes

Day 6: Angular Performance Optimization Techniques

Topics:

- Lazy Loading, Code Splitting, Tree Shaking
- Using trackBy in *ngFor
- Pure vs Impure Pipes
- OnPush Strategy Recap
- Angular DevTools and Chrome Performance Profiler

Lab:

- Analyze bundle sizes using ng build --stats-json + webpack-bundle-analyzer
- Apply trackBy to improve *ngFor rendering
- Convert impure pipes to pure and compare performance

Day 7: Unit Testing and E2E Testing

Topics:

- Jasmine/Karma Basics
- Testing Services and Components with Mocks

- Testing Forms and Directives
- E2E Testing with Cypress or Playwright
- Test Strategies in Micro Frontends

Lab:

- Write unit tests for service with mocked HTTP responses
- Test a reactive form's validation
- Create a Cypress test for user login flow

Day 8: Micro Frontends Introduction

Topics:

- What are Micro Frontends (MFEs)?Monolith vs Micro Frontend Architecture
- MFE Approaches: iFrame, Web Components, Module Federation
- Use Cases, Pros, and Cons
- Communication Between MFEs

Lab:

- Create two standalone Angular apps
- Load one app inside another via iFrame
- Implement basic communication using postMessage

Day 9: Angular Micro Frontends using Module Federation

Topics:

- Webpack 5 Module Federation Basics
- Setting up Angular CLI projects with Module Federation
- Host vs Remote Architecture
- Shared Libraries and Dependency Management
- Navigation and Routing Between MFEs

Lab:

- Create host and two remote Angular apps
- Expose components and routes from remotes
- Load components remotely and manage shared dependencies

Day 10: Build & Deploy Micro Frontends

Topics:

- Independent Builds and Deployment Pipelines
- Runtime Integration of Remotes
- Version Compatibility and Contracts
- Deployment Strategies
- Production Readiness Considerations

Lab:

- Containerize host and remotes using Docker
- Deploy to local environment using Docker Compose