Angular Advanced & Architecture mit Nx Workshop

Eine wachsende Angular-Anwendung so strukturieren, dass zehn Entwickler daran arbeiten können, ohne sich in die Quere zu kommen.

2 TAGE | EXPERTEN | REMOTE ODER INHOUSE | DEUTSCH & ENGLISCH | BIS ZU 12 TEILNEHMENDE (MEHR AUF ANFRAGE)

FÜR WEN

- Erfahrene Angular-Entwickler und Tech Leads
- Teams, deren Anwendung schneller gewachsen ist als ihre Struktur
- Teams, die ein Monorepo oder eine Migration zu Nx planen

DANACH KANN EUER TEAM

- Eine Anwendung in Libraries mit klaren Grenzen und Abhängigkeitsregeln schneiden
- Entscheiden, wo State hingehört, und ihn mit NgRx oder dem Signal Store umsetzen
- Typisierte, validierte Formulare für komplexe Geschäftsfälle bauen
- Einen Nx-Workspace mit Affected Builds, Linting und Tests aufsetzen

VORAUSSETZUNGEN

- Mindestens ein Jahr praktische Angular-Erfahrung
- Vertrautheit mit RxJS-Grundlagen und der Angular CLI
- Idealerweise: Zugriff auf eure eigene Codebasis, um echte Beispiele zu besprechen

Die meisten Angular-Anwendungen scheitern nicht an Angular, sondern an der Struktur: alles in einem Ordner, State über Komponenten verteilt, Formulare, die sich niemand mehr anzufassen traut. Dieser Workshop gibt eurem Team die Patterns, um das zu ändern: Komponenten-APIs mit Signals, eine Standalone-Architektur, fortgeschrittene Formulare, Testing, State Management mit NgRx und dem Signal Store, Libraries und Nx Workspaces.

Der letzte Block ist für euer Projekt reserviert: Bringt eure Fragen und euren Code mit, und wir arbeiten sie gemeinsam durch. Teams gehen meist mit einem konkreten Plan für die nächsten Refactoring-Schritte nach Hause.

AGENDA

Reihenfolge und Tiefe der Blöcke passen wir an euer Team an – was ihr schon beherrscht, lassen wir weg, eure Fragen bekommen die Zeit.

01 Komponenten-APIs mit Signals

- Signal Inputs, Outputs und Models für klare Komponenten-APIs.
- Smart und Presentational Components – und wo Signals und wo RxJS passen.

02 Standalone-Architektur und Lazy Loading

- Features ohne NgModules organisieren: Provider, Routen und gemeinsame Abhängigkeiten.
- Lazy Loading mit loadComponent und loadChildren, und @defer für schwere Teile einer Seite.

03 Fortgeschrittene Formulare mit feldübergreifender Validierung

- Typisierte Reactive Forms für komplexe fachliche Fälle.
- Eigene, feldübergreifende und asynchrone Validatoren.
- Wiederverwendbare Formular-Komponenten mit ControlValueAccessor.

04 Testing

- Unit- und Komponententests mit Vitest.
- End-to-End-Tests mit Playwright.

05 State Management mit NgRx und dem Signal Store

- Wohin State gehört: Komponente, Service oder Store.
- NgRx Store mit Actions, Reducern, Effects und Selektoren.
- Der NgRx Signal Store und resource() für Signal-basierten State und asynchrone Daten.

06 Arbeiten mit Libraries und Nx

- Die Anwendung in Feature-, Domain-, UI- und Util-Libraries schneiden.
- Wiederverwendbare Libraries bei Bedarf im Angular Package Format (APF) veröffentlichen.

07 Nx Workspaces

- Modulgrenzen und Abhängigkeitsregeln, per Lint durchgesetzt.
- Affected Builds, Tests und Caching in der CI.
- Generatoren für einheitliche neue Features.

08 Fragen aus eurem Projekt

- Euer Code, eure Fragen: Wir arbeiten gemeinsam echte Beispiele durch.
- Ein konkreter Plan für die nächsten Refactoring-Schritte.

EUER TRAINER



Fabian Gosebrink

MICROSOFT MVP, GOOGLE DEVELOPER EXPERT

Baut jede Woche mit Teams Angular-Anwendungen und hat 120+ Konferenzvorträge zu Angular, State Management und Architektur gehalten.

TRAINING ANFRAGEN

- Nur für euer Team – Remote oder Inhouse
- bis zu 12 Teilnehmende (mehr auf Anfrage)
- ab CHF 2’400 pro Tag für das ganze Team
- Vor Ort: zuzüglich Reisekosten
- Fixes Angebot innerhalb eines Werktags
- Das Erstgespräch ist kostenlos und unverbindlich
- Deutsch oder Englisch
- Sitz bei Zürich – vor Ort in CH, DE, AT; remote überall

Training anfragen: offering.solutions/de/angular-advanced-enterprise-architecture/ · mail@offering.solutions