

Redux

zarządzanie stanem w aplikacjach

Ducin IT Consulting - Program szkolenia

Czas trwania: 3 dni

Formuła: 25% wykłady, 75% ćwiczenia

Szkolenie przeznaczone jest dla wszystkich, którzy chcą poznać lub pogłębić wiedzę dotyczącą zarządzania stanem w nowoczesnych aplikacjach frontendowych. Szczegóły omawiane są na podstawie Reduxa, natomiast mnóstwo podobieństw i analogii można odnaleźć w narzędziach takich jak NGRX czy VUEX. Szkolenie jest pogłębionym studium nad zastosowaniem, możliwościami oraz dobrymi praktykami używania Reduxa (i/lub jego alternatyw w odpowiednich frameworkach).

Początkowa część szkolenia uwzględni dogłębne zrozumienie paradygmatu funkcyjnego, będącego fundamentem Reduxa. Zawiera ona również dokładne wejście w filozofię i mechanikę biblioteki.

Pozostała, większa część szkolenia, przeznaczona jest na kodowanie rozwiązań - oraz ich testowanie. Przewidziane są etapy pracy indywidualnej, grupowej oraz warsztaty designowe. Szczególny nacisk kładziemy na zaawansowane techniki zarządzania stanem dostępne dzięki redux- saga oraz redux- observables - ewentualne braki dotyczące podejścia CSP/coroutines oraz FRP/reactive streams zostaną uzupełnione na bieżąco.

Kluczowe Aspekty:

- architektura redux
- bilans zysków i strat: jakie problemy redux rozwiązuje, jakich nie rozwiązuje, oraz jakie niesie ze sobą koszty
- dobre praktyki, częste błędy

Program szkolenia:

1. Functional Programming recap

- 1.1. FP vs OOP
- 1.2. Immutability
- 1.3. Pure Functions, Side Effects
- 1.4. Higher Order Functions
- 1.5. Reducers
- 1.6. Closures

2. Redux - building blocks

- 2.1. actions
- 2.2. reducers
- 2.3. state
- 2.4. store

3. Redux - philosophy & architecture

- 3.1. Inversion of Control
- 3.2. Pub-Sub

4. Async Flow

- 4.1. redux-thunk
- 4.2. redux-saga & coroutines
- 4.3. redux-observables & reactivity

5. Custom Middlewares

- 5.1. async/await
- 5.2. logging
- 5.3. storage

5.4. network communication

6. Tooling

6.1. redux-devtools

6.2. immer

7. Środowiska (opcjonalnie)

7.1. Vanilla JS / jQuery

7.2. React

7.3. Angular / NGRX

8. Alternatives

8.1. Private State

8.2. Services (angular)

8.3. Context API (react)

8.4. Mobx

9. Usage with TypeScript

9.1. Static typing actions, reducers and state

10. Redux Design exercises

10.1. design big application actions, state, async flows