

SZKOLENIE ŚREDNIO ZAAWANSOWANE

RxJS - Reaktywne programowanie w JavaScript

RXJS

Czas trwania: 2 dni

RxJS to coraz popularniejsza biblioteka do reaktywnego programowania w JavaScript

Cele szkolenia

- Przekazanie praktycznej wiedzy z zakresu programowania reaktywnego w JavaScript z wykorzystaniem RxJS
- Nauka myślenia reaktywnego
- Asynchroniczność w JavaScript (callback, promise, observable)
- Wykorzystanie RxJS po stronie przeglądarki oraz serwera
- Podstawowe elementy RxJS, takie jak Observable, Observer, operator, subscription
- Wbudowane operatory, te podstawowe oraz bardziej zaawansowane
- Komponowanie strumieni
- Tworzenie własnych operatorów
- Zarządzanie reaktywnym stanem aplikacji webowej
- Zarządzanie subskrypcją, czyli jak unikać wycieków pamięci
- Testowanie strumieni

Zalety

- Dzięki przygotowanym ćwiczeniom, na szkoleniu RXJS poznamy korzyści jakie płyną z pisania kodu reaktywnego
- Pozbędziemy się imperatywnych nawyków pisania kodu, zamieniając architekturę stateful na architekturę reaktywną opartą o strumienie
- Zdobytą wiedzę przećwiczymy budując aplikację do pracy zespołowej w czasie rzeczywistym, dzięki czemu lepiej zapamiętasz zastosowanie poszczególnych elementów biblioteki
- Praktyka przed teorią - wszystkie szkolenia technologiczne prowadzone są w formie warsztatowej. Konieczna teoria jest wyjaśniana na przykładzie praktycznych zadań
- Konkretność umiejętności - w ramach każdego szkolenia rozwijamy praktyczne umiejętności związane z daną technologią i tematyką
- Nauka z praktykami - wszyscy trenerzy na co dzień pracują w projektach, gwarantuje to dostęp do eksperckiej wiedzy i praktycznego know-how



Dla kogo?

- Szkolenie przeznaczone dla programistów ze znajomością JavaScript, którzy chcą rozszerzyć swój warsztat o reaktywne programowanie po stronie klienta oraz serwera

Wymagania

- Od uczestników szkolenia RXJS wymagana jest podstawowa znajomość JavaScript
- Zagadnienia, które należy znać: var, const, let, pętle, operacje warunkowe, funkcja, arrow function, klasa, ES Modules, Promise, async/await



Program

1. Wprowadzenie
 - a. ECMAScript 2015
 - b. TypeScript
 - c. Angular
2. Programowanie funkcyjne
 - a. Funkcja
 - b. Scope
 - c. Closures
 - d. Funkcja wyższego rzędu
 - e. Funkcje czyste a efekty uboczne
3. Asynchroniczność
 - a. Callback
 - b. Promise
 - c. Observable
4. Podstawy RxJS
 - a. Podstawy działania strumieni
 - Observable
 - Observer
 - Subscription
 - b. Tworzenie strumieni
 - Własny strumień: Observable.create()
 - Wbudowane kreatory strumieni: of, from, fromEvent, timer
 - c. Jak działają Operatory
 - Użycie metody pipe()
 - Podstawowe operatory: map, filter, reduce, switchMap
 - Tworzenie własnych operatorów
 - d. Observable i Observer w jednym
 - Subject
 - BehaviorSubject
 - ReplaySubject
 - e. Multicasting
 - Cold i Hot Observable
 - Operator: publish, share, multicast
 - f. Scheduler
 - g. Testowanie przy użyciu Marble Diagrams
5. Budowa przykładowej aplikacji do pracy zespołowej w czasie rzeczywistym (real time)
 - a. Zarządzanie danymi i stanem aplikacji webowej (Angular)
 - b. Budowa serwera WebSockets (Nest)
 - c. Komunikacja przez WebSockets
 - d. Zarządzanie stanem offline/online
 - e. Praktyczne zastosowanie zaawansowanych operatorów



f. Sposoby zarządzania subskrypcją w celu uniknięcia wycieków pamięci

