

SZKOLENIE ŚREDNIO ZAAWANSOWANE

PWA - Progresywne aplikacje webowe

PWA

Czas trwania: 3 dni (24h)

HTML5 i ServiceWorkers pozwalają na budowanie aplikacji offline oraz progresywnych aplikacji internetowych w przeglądarkach

Cele szkolenia

- Pokazanie możliwości budowania aplikacji progresywnych - wyglądających i działających jak natywne aplikacje mobilne, ale działające w przeglądarce z użyciem JavaScript i HTML5

Zalety

- Szkolenie skupia się na praktycznym wykorzystaniu możliwości nowoczesnych przeglądarek
- Uczestnik podczas praktycznych zajęć poznaje strategie szybszego ładowania aplikacji, cache, pracy offline oraz budowania progresywnych aplikacji, które dostosowują się do środowiska oraz korzystają z natywnych API, dzięki czemu strony HTML5 mogą wyglądać i zachowywać się jak zainstalowane aplikacje mobilne
- 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?

- Programiści chcący budować aplikacje progresywne i offline przy użyciu JavaScript i HTML5
- Architekci chcący poznać możliwości nowoczesnych przeglądarek pod kątem projektowania aplikacji progresywnych

Wymagania

- Dobra znajomość JavaScript i HTML5

Program

1. Wprowadzenie
 - a. Aplikacje webowe a natywne
 - b. Aplikacje natywne a hybrydowe



- c. Progresywne aplikacje webowe - PWA
- d. Koncepcje i zastosowanie
- e. Progressive Enhancement
- f. PWA w kontekście SPA - Single Page Applications
- 2. Przygotowanie PWA
 - a. Plik manifestu aplikacji
 - b. Wsparcie przeglądarek
 - c. Narzędzia deweloperskie
- 3. Testowanie i debugowanie
 - a. Testowanie w przeglądarce
 - b. Emulowanie urządzenia
 - c. Testowanie na fizycznym urządzeniu
 - d. Instalacja i aktualizacja
- 4. Service workers a PWA
 - a. Wprowadzenie do Service workers
 - b. Cykl życia SW
 - c. Rejestracja workera
 - d. Zdarzenia i ich obsługa
 - e. Aktualizacja i aktywacja workera
 - f. Pozostałe zdarzenia
 - g. Baner instalacyjny
- 5. Promise i Fetch w przeglądarkach
 - a. Fetch API
 - b. Wsparcie przeglądarek i Polyfill
 - c. Obsługa CORS
- 6. Service Worker jako Cache
 - a. Cache API
 - b. Wsparcie przeglądarek
 - c. Cache statyczny/precaching
 - d. Cache wielu plików
 - e. Obsługa błędów
 - f. Wersjonowanie cache
 - g. Dynamiczny cache
- 7. Zaawansowany cache
 - a. Cache na żądanie
 - b. Offline fallback
 - c. Cache only, Network only i inne
 - d. Routing i strategie cache
- 8. Przechowywanie danych aplikacji
 - a. LocalStorage i SessionStorage
 - b. IndexedDB
 - c. Przykład z użyciem Google Firebase
 - d. Synchronizacja w tle
- 9. Notyfikacje WebPush



- a. Omówienie, ważne koncepcje
 - b. Narzędzia deweloperskie
 - c. Uprawnienia
 - d. Wyświetlanie powiadomień
 - e. Akcje powiadomień
 - f. Subskrypcja powiadomień Push
 - g. Wysyłanie powiadomień z serwera
10. Natywne API
- a. Obsługa Plików
 - b. Audio/Video i grafika
 - c. Lokalizacja i śledzenie Użytkownika
 - d. API Baterii
 - e. API połączenia sieciowego
 - f. Accelerometr
11. Narzędzie WorkBox
- a. Omówienie możliwości
 - b. Konfiguracja i generowanie workera
 - c. Strategia precache
 - d. Routing
 - e. Dynamiczny cache
 - f. Offline HTML fallback
 - g. Proces budowania
12. Narzędzie Lighthouse
13. Integracja z popularnymi frameworkami
- a. Angular
 - b. React
 - c. Vue

