Ducin IT Consulting

Architektura JavaScript

Program szkolenia, Ducin IT Consuling

Czas trwania: 3-5 dni
Formuła: 33% wykłady, 33% ćwiczenia, 33% projektowanie

Szkolenie ma na celu ugruntowanie oraz poszerzenie wiedzy z zakresu designu oraz architektury oprogramowania - ze szczególnym naciskiem na platformę JavaScript. Przegląd narzędzi oraz technik odbywa się poprzez omawianie zarówno powodów, dla których dane rozwiązanie należy zastosować, jak i kluczowych różnic, jakie między nimi występują. To z kolei kładzie nacisk na rozumienie konsekwencji dokonywanych przez architektów decyzji. Przykłady oraz implementacja koncentrują się na języku JavaScript, jednak omawiane zagadnienia (wzorce projektowe, architektoniczne, pryncypia) są uniwersalne; niektóre dotyczą także backendu.

Uczestnicy szkolenia poznają szerszy kontekst wokół narzędzi, których na co dzień używają. Dzięki ćwiczeniom z projektowania w grupach zwiększają rozumienie problemów, które należy rozwiązać. Szkolenie nie uczy kodowania w konkretnym frameworku/ narzędziu. Przygotowuje do roli architekta - osoby odpowiedzialnej za zrozumienie klas problemów i wybór klas rozwiązań a także rozumienia konsekwencji podejmowania decyzji (ich zakresu oraz czasu).

Zakres

Poniższy program jest ramowy, tj. uwzględnia szeroki zakres, niekoniecznie możliwych do realizacji w ciągu 3 dni. Uczestnicy szkolenia w dużej mierze na bieżąco mają wpływ na wybór zagadnień, na których realizacji najbardziej im zależy. W praktyce, docelowy program formułuje się sam w trakcie szkolenia.

Zalety Szkolenia:

- Przegląd klasycznych problemów z jakimi mierzą się aplikacje frontendowe wraz z możliwymi rozwiązaniami
- Style architektoniczne, ich zastosowanie, zalety i wady
- Wzorce projektowe

Ducin IT Consulting

Program szkolenia:

1. Architektura - od strony miękkiej

1.1. Cele architektury

1.2. Architekt - rola w projekcie, obowiaązki i wyzwania

1.3. Projektowanie architektury

1.4. Poszukiwanie cech rozwiązań - oraz samych rozwiązań

1.5. Zarzaądzanie decyzjami architektonicznymi

1.6. Identyfikowanie potencjalnych problemoów

1.7. Planowanie i realizacja refaktoringów

2. Concepts

2.1. Coupling

2.2. Cohesion

2.3. Reusability

2.4. Inversion of Control

2.5. Caching

3. Principles

3.1. SOLID

3.2. DRY

3.3. KISS

3.4. YAGNI

3.5. Law of Demeter

3.6. The Holywood Principle

4. Design Patterns

4.1. Singleton