Angular 2 and TypeScript Web Application Development

Course code: IJ -19
Course domain: Software Engineering
Number of modules: 1
Duration of the course: 40 study$^1$ hours

Sofia, 2016

Copyright © 2003-2016 IPT – Intellectual Products & Technologies Ltd. All rights reserved.

$^1$ Duration of a study hour is 45 minutes.
Angular 2 and TypeScript Web Application Development

STUDY PLAN

<table>
<thead>
<tr>
<th>Module name</th>
<th>1. Angular 2 and TypeScript Web Application Development</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lectures, study hours</td>
<td>10</td>
</tr>
<tr>
<td>Laboratory exercises, study hours</td>
<td>30</td>
</tr>
<tr>
<td>Total, study hours</td>
<td>40</td>
</tr>
</tbody>
</table>

Lecturer: Trayan Iliev
IPT – Intellectual Products & Technologies Ltd.
E-mail: ttiliev@iproduct.org

Target audience: Medium level JavaScript developers with practical experience in building web applications using HTML 5, CSS 3, JavaScript and Bootstrap.

Course duration: Duration of the course is 40 study hours total. Training will be conducted in 5 days – 8 study hours each day.

Course Description:

The course provides in-depth study of state-of-the-art JavaScript (ES 5 & 6), TypeScript and MV* Angular 2 JS/TS framework for rapid development of modern, mobile-first, responsive single-page applications that are easy to extend and maintain in long run. Angular 2 builds on success of AngularJS web application development framework, but is significantly more efficient (up to 10-15 times faster for big number of updated components), streamlined, elegant and easy to learn. The main topics that will be covered during the course include:

1. Object-oriented JavaScript – primitive types and objects, accessing objects by reference, properties, functions and methods, using this keyword, call, apply, & bind function methods, prototypal inheritance, polymorphism and method overriding, classes and constructors, classical inheritance and using instanceof, using this, EcmaScript 6 (Harmony, ES 2015) class and constructor syntax, let and var, function lambdas (=>), Promises. Overview of common JS design patterns: Constructor, Module, Revealing Module, Singleton, Observer, Mediator, Prototype, Command, Facade, Factory, Mixin, Decorator, Flyweight. (3 st. h.)
2. Introduction to Model-View-Controller (MVC), Model-View-Presenter (MVC), Model-View-ViewModel (MVVM) – MV* patterns for development of more complex, extensible and easy to maintain web applications using JavaScript/TypeScript and Angular 2 JS/TS framework. Different versions of JavaScript: EcmaScript 5, EcmaScript 6 and TypeScript – feature comparison. Preparing setup and installation with Visual Studio 2015 Community and Visual Studio Code. Creating TypeScript projects. TypeScript compilation options. (1 st. h.)


5. Displaying data using template interpolation and built-in directives: NgModel, NgFor, NgIf, NgSwitch, NgStyle, NgClass. Handling user input – binding event handlers, getting data from the $event object, using local template variables, event filtering, handling multiple events. Building forms – two-way data binding: [(ngModel)], change tracking, validation, and error handling:
ngControl, providing custom styles for different control states, resetting and submitting forms, edit/display forms. Using NgFormModel directive to bind a domain model to the form. Demos and exercises. (4 st. h.)

6. Angular 2 services. Building custom services. Understanding Dependency Injection (DI) – DI using constructors, ng2 DI framework, configuring the injector, registering providers in components, declarative and programmatic dependency injection, injecting service dependencies in a service using @Injectable() decorator, registering custom providers using provide function, dependency injection tokens. Using hierarchical injectors. Demos and exercises. (3 st. h.)


9. Routing and navigation in Angular 2 single page applications – loading router library, configuring a router, injecting router service, link parameters array that propels router navigation, navigating by user clicks - data-bound RouterLink,
programmatic navigation, using route parameters, using child routers, setting default routes, passing optional data in query params. Browser URL styles for SPA navigation: `history.pushState` vs. "#" ("hash"). Advanced topics: using router lifecycle hooks to customize navigation – `CanActivate`, `OnActivate`, `CanDeactivate`. (4 st. h.)

10. Advanced Angular 2 topics: writing custom attribute and structural directives, component lifecycle and using custom lifecycle hooks: `ngOnInit`, `ngOnChanges`, `ngDoCheck`, `ngOnDestroy`, `ngAfterContentInit`, `ngAfterContentChecked`, `ngAfterViewInit`, `ngAfterViewChecked`. Use examples. `Shadow DOM` and Angular 2 view encapsulation modes – `ViewEncapsulation.None`, `ViewEncapsulation.Emulated` and `ViewEncapsulation.Native`. Test Driven Development (TDD) - unit testing Angular 2 classes and components using `Jasmine`. Using Angular 2 project seeds. Review of sample Angular 2 GitHub projects (bigger ones). (5 st. h.)

The workshop contains 25% lecture materials and 75% lab exercises. Lectures and exercises will be conducted in parallel and will not be divided in separate sessions in order to achieve immediate reinforcement of theoretical discussions with practical examples and exercises.

During the workshop participants will get practical experience using `Angular 2 JS/TS framework` for building `single page applications` by solving problems and exercises. The learning is conducted in small groups – up to 11 participants using problem-based methodology. During the workshop there will be opportunity for discussion of additional questions the participants are interested in.