

Course code: **PHGAP**

Course title: **Development of inter-platform mobile applications using PhoneGap and Angular JS**

Days: 3

Description:

Course intended for:

The training is intended for programmers, who would like to get familiar with technologies that enable development of inter-platform mobile applications. In particular, it is intended for programming teams, which are responsible for development of the same applications for several mobile platforms under a strong pressure of time.

Course objective:

The users will find out how to develop inter-platform mobile applications using the PhoneGap framework, HTML5 and AngularJS library. They will get familiar with the technology, which allows for elimination of the basic problem, associated with mobile device software: the necessity to develop separate versions of the application for each platform - Android or iOS. The participants will find out how to use the most recent version of the HTML and CSS specification, as well as the AngularJS library, to develop an application with a rich user interface, taking advantage of the hardware capabilities of individual platforms thanks to PhoneGap.

Requirements:

The training participants are required to have the basic programming skills and familiarity with HTML.

Course parameters:

3*8 hours (3*7 net hours) of lectures and workshops (with a visible emphasis on workshops).

Group size: no more than 8-10 participants.

Course curriculum:

1. Mobile applications- discussion



- I. Strengths and weaknesses of native applications and Web-based mobile applications
 - II. Differences in user interfaces
 - i. Requirements of Apple with regard to mobile applications
 - ii. User interface standards in the Android system
2. PhoneGap (Cordova) – a mobile Web application
- I. WebKit as a rendering engine – differences in comparison with Google Chrome
 - II. The system requirements
 - III. The programming environment components
 - i. Eclipse
 - ii. Cordova Application Generator
 - iii. Android system emulator
 - IV. Application setting
 - i. Supporting various devices
 - ii. Resolution differences
 - iii. Screen resolution management
 - V. Attaching of external libraries to PhoneGap/Cordova
 - i. Library installation
 - ii. compilation
 - iii. JavaScript library interface
 - VI. Use of API
 - i. Photo and movie camera
 - ii. Sound recording



- iii. Web connection checking
- iv. Geolocation (GPS)
- v. Other components of API PhoneGap

VII. Application packaging – apk files

- i. Setting of the target device
- ii. Developer mode in the Android system and system rooting
- iii. Installation of the application from the developer environment

3. CSS3 and mobile devices

I. Use of text styles

II. Application of styles to block components

III. CSS3 selectors

IV. Enhancing of graphic effects with CSS3

V. Exercises: header styling

VI. Fade-in effect in CSS3

VII. Device detection and selection of proper representation. CSS 3 Media Queries

VIII. Known problems with RETINA type displays

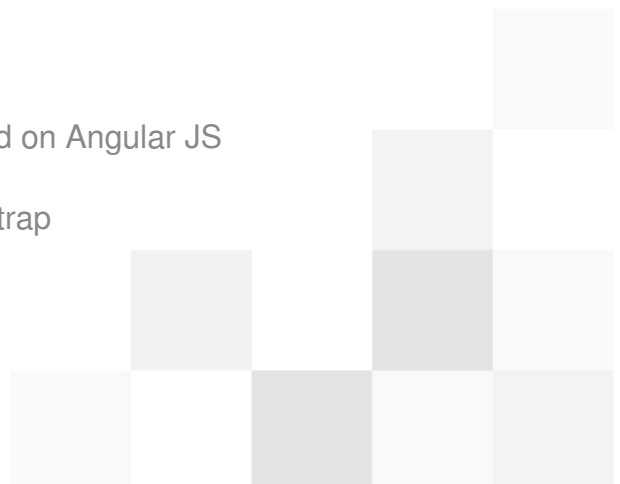
- i. Use of other images via Media Queries
- ii. Use of SVG in mobile applications as an alternative

4. Angular JS: MVVW in JavaScript

I. What is a MVVW and MVW pattern

- i. Building of applications based on Angular JS
- ii. Angular Seed / Twitter Bootstrap

II. Models in Angular JS



- i. Explanation of functioning of models in Angular JS
- ii. Bidirectional and unidirectional model linking
- iii. Isolated \$scope and its application

III. Views and templates

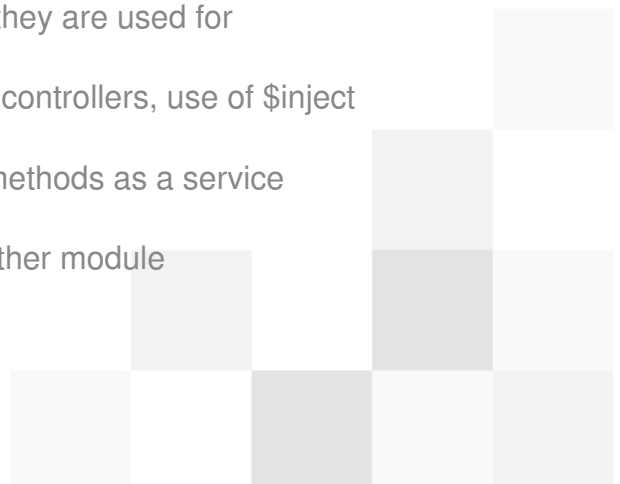
- i. discussing of the view function in Angular JS
- ii. repeaters
- iii. CSS class switches, logic expressions
- iv. Attaching of HTML code of various templates
- v. Form validators:
 - A. Use of in-built validators
 - B. Attachment of non-standard validators
 - C. Form and component error styles
- vi. Event handling
- vii. Attaching of external libraries of Angular UI

IV. Directives

- i. What is a directive and what it is used for
- ii. Attachment of external directives
- iii. Practical use – directives with isolated controllers

V. Services

- i. What are services and what they are used for
- ii. Injection of services to many controllers, use of \$inject
- iii. discussing of Angular \$http methods as a service
- iv. use of service module in another module



v. REST interface as an Angular Web Service

VI. Routers

- i. Template loading after URL change
- ii. Using many views for one and many controllers

VII. Angular UI

- i. Use of external UI libraries
- ii. Development of lists using ready libraries (ng-grid)
- iii. Management of touch events
 - A. Libraries needed (QuoJS)
 - B. Events and dependencies handled

VIII. Unit tests in Angular JS

IX. Exercises: RIA type application development with Angular JS

