

Course code: **JEE7/ANG**

Course title: **Modern web applications built with JEE7, AngularJS and HTML5**

Days: 5

Description:

Course intended for:

The training is aimed at web application developers, willing to supplement their skills on the latest technologies, as well as for people who know Java and who want to start an adventure with web applications using the latest technologies. As a presentation layer, the AngularJS is presented, as it is becoming more widely appreciated and used for its Model-View-Controller architecture. The training is carried out using the latest Java Enterprise Edition - Version 7. For those wishing to work with older and established technologies we recommend WEB2 training.

Course objective:

Participants will learn how to design and implement applications using Java EE platform technologies. The goal of the training is to gain the ability to create scalable applications, in accordance with the arcane art: from business components to highly ergonomic and functional interfaces suited to the WEB 2.0 trends.

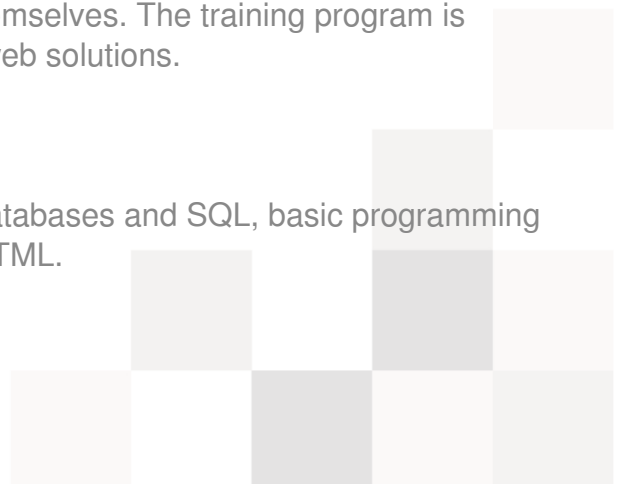
In particular: Participants will learn AngularJS, HTML5 and JavaScript. They will learn how to design and implement the business logic layer using EJB 3.2 combined with the JPA2.1 mechanisms of object-relational mapping. They will also learn the CDI specification.

Course strengths:

The training program covers issues related to a web application development. In contrast to other trainings, this course is not focused on a single technology, but it covers most important issues of selected technologies. As a result, after completing the training, participants will be able to implement complete web applications by themselves. The training program is constantly updated due to a rapid development of web solutions.

Requirements:

The participants are required basic knowledge of databases and SQL, basic programming skills in Java, basic knowledge of JavaScript and HTML.



Course parameters:

5*7 hours of lectures and workshops at a ratio of 1:3. During workshops, in addition to simple exercises, participants will design and implement a web application using JEE7, JS, HTML. Group size: max. 8-10 people.

Course curriculum:

1. Introduction

- I. Java Enterprise Edition
- II. JEE most commonly used components: EJB, CDI, JPA, web services, websockets, servlets.
- III. Design and implementation with JEE
- IV. AngularJS as a MVC framework, thin versus thick client

2. Business logic layer - EJB 3.2

- I. EJB 3.x component types
- II. How does it work?
- III. Session components
 - i. Stateless
 - ii. Statefull
 - iii. Singleton
 - iv. Lifecycle, lifecycle callbacks
 - v. Accessing EJB
 - A. Locally
 - B. Locally without an interface
 - C. Remotely
 - D. Web service



- vi. Asynchronous EJBs
- vii. Concurrency
 - A. Container role
 - B. SFSB and concurrency
- IV. Working with resources
 - i. JNDI lookups
 - ii. Inversion of Control
- V. Interceptors
- VI. Exceptions
- VII. Message Driven Beans
 - i. JMS2.0
 - ii. How does it work?
 - iii. MEPs
- VIII. Security
- IX. Transactions
 - i. Container managed transactions (CMT)
 - ii. Bean managed transactions (BMT)
 - iii. Client managed transactions
- X. EJB and web services
 - i. Web services basics
 - ii. JAX-WS/JAX-RS and EJB/CDI
- XI. Deployment
 - i. EJB module (jar)



- ii. Enterprise Archive (ear)
- iii. Web Archive (war)
- iv. Full vs light (web) server profile
- v. EJB embedded container

XII. Testing EJB

XIII. EJB Lite and web applications

3. Persistence layer and Java Persistence API 2.1

I. ORM conception

II. Entities

- i. simple
- ii. embedded
- iii. accessing fields
- iv. collections

III. Associations

IV. Keys

- i. Simple
- ii. Complex
- iii. Automatic generation of keys

V. PersistentContext

- i. Container managed and application managed
- ii. Transaction and Extended

VI. Entity lifecycle

VII. DML



VIII. Transactions

IX. Queries

i. JPQL

A. Syntax

B. Query and TypedQuery

C. Named queries

D. Native queries

ii. JPA2.1 Criteria API

A. Criteria API

iii. serach-within and serach-by-example

iv. Native queries mapping

X. Lifecycle listeners

XI. Testing JPA

XII. Deployment

i. Java SE/dektop

ii. Web module

iii. EJB module

4. CDI (WebBeans)

I. Introduction

i. Bean

ii. Context

iii. Scope

iv. Injection



II. Beans

- i. Managed beans
- ii. Session means
- iii. Specialization
- iv. Active and inactive beans

III. Scopes

- i. Built-in scopes
- ii. Conversation scope
- iii. Dependent scope
- iv. Nonstandard scopes
- v. Default scope
- vi. Lifecycle of scope objects

IV. Injection

- i. Where to inject
- ii. Qualifiers, Alternatives, Stereotypes
- iii. Named injections

V. Producers

VI. Events and listeners

VII. CDI and EJB

5. WebSockets

56. Endpoints 57. Sending and receiving messages 58. HTTP Long Polling and WebSockets

6. Web application using AJAX

I. AJAX communication models



II. JSON

- i. Introduction
- ii. Java API for JSON Processing
- iii. JSON and Restful web service

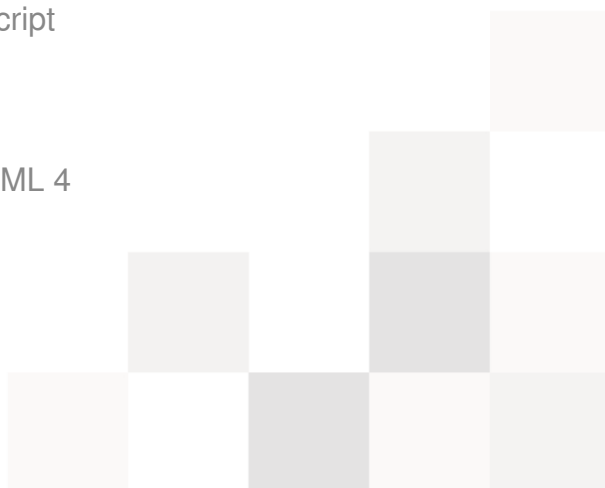
III. Prototype and JQuery libraries

7. Java Servlet 3.1

- I. Introduction
- II. Lifecycle of a servlet
- III. Sharing data
- IV. Servlet initialization
- V. Process HTTP requests
- VI. Filtering
- VII. File upload, media types
- VIII. WebContext
- IX. Asynchronous processing
- X. Nonblocking I/O
- XI. HTTP 1.1
- XII. Security

8. HTML 5

- I. Introduction to HTML 5, CSS, JavaScript
 - i. Simple website
 - ii. Advantages HTML 5 over HTML 4
 - iii. Web browsers and HTML 5



- iv. New tags in HTML 5
- v. JavaScript
- vi. JavaScript and HTML DOM
- vii. Events
- viii. Web forms validation using JS

II. Programming JavaScript

- i. JavaScript code structure
- ii. Creating and extending objects
- iii. Best practices

III. Programming HTML 5

- i. File operations
- ii. Drag & Drop
- iii. Embedding multimedia
- iv. Web Audio API
- v. Geolocation API
- vi. Debugging and profiling

IV. Altering application presentation

- i. Creating portable website
- ii. Detecting web browser, CSS 3 Media Queries

V. CSS 3 and HTML 5

- i. Text styles
- ii. Block element styles
- iii. CSS3 Selectors



iv. Graphics with CSS3

VI. Animation

- i. Transitions
- ii. Transformations
- iii. CSS Key-Frame

VII. AJAX

- i. XMLHttpRequest
- ii. Web Sockets

VIII. Web Messaging API

- i. Message passing
- ii. Authorizations
- iii. Channel and ports

IX. Web Worker Process

- i. Asynchronous processing
- ii. Types of Web Workers
- iii. Browser support
- iv. Busy message

9. CSS 3: selected issues

- I. Fonts
- II. Text effects
- III. Document layers
- IV. Border effects
- V. Color transparency and gradients



VI. 2D/3D animations

VII. Box Layout

VIII. Document background

10. Java Script

I. Selected issues

i. Tables and structured

ii. Functions as object

iii. Varargs

iv. Prototyping

v. Built-in prototypes

vi. Closed

vii. Regex

II. Selected issues and problems

i. SameOriginPolicy, CORS

ii. Mouse position

iii. "Forward" and "Back"

iv. Alerts

v. Redirecting

vi. New window

vii. Date and time

viii. Delayed execution

ix. Loading resources before usage

x. Onmouseover listener



- xi. Random number generator
- xii. Working with characters
- xiii. Number formating
- xiv. Memory leaks in JavaScript

11. AngularJS: MVC in JavaScript

I. What MVC and MVVM are?

- i. AngularJS
- ii. Angular Seed / Twitter Bootstrap

II. AngularJS models

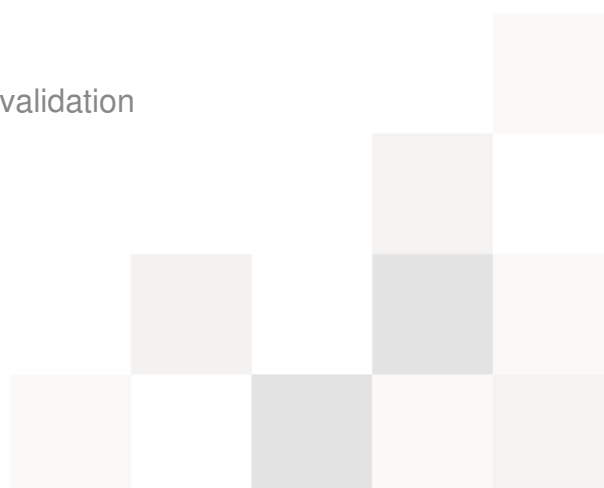
- i. Model binding
- ii. \$scope

III. Views and templates

- i. Views in AngularJS
- ii. repeaters
- iii. CSS switching
- iv. Form validators
 - A. Built-in validators
 - B. Custom validators
 - C. Validators and CSS
- v. Events
- vi. External JS libraries user for validation

IV. Directives

- i. Introduction



- ii. External directives
- iii. Isolated controller directives

V. Services

- i. Introduction
- ii. \$inject
- iii. Angular \$http as a service
- iv. Angular Web Service

VI. Routers

- i. Template reloading
- ii. Multiple views for multiple controllers

VII. Angular UI

- i. External libraries with Angular
- ii. ng-grid

VIII. Testing web ui with Jasmine/Karma/Selenium

IX. JavaScript debugging with web browser

X. Security

