

Service-Architecture

Das folgende Dokument dient als Grundlage für die Entwicklung der Conference-App. Es definiert Standards für die zu verwendeten Technologien, der Projektstruktur, Namenskonventionen und Coding Standards.

Technologien

Backend

Es kommen überwiegend Technologien aus dem JavaEE 7 Technologie-Stack zum Einsatz. Wo ergänzend andere Technologien verwendet werden, sind diese gesondert aufgeführt.

- JavaEE 7
 - JPA 2.1
 - EJB 3.1
 - CDI 1.1
 - JAXRS 1.1
 - JMS 2.0
- Sonstige
 - JodaTime : einfache DateTime-API, performanter als JAVA-Bordmittel, etabliert

Frontend: HTML, CSS & JavaScript

Es kommen ausschließlich HTML, CSS und JavaScript zum Einsatz. Zur Vereinfachung der Entwicklung, kommen die folgenden Frameworks und Libraries zum Einsatz:

- AngularJS 1.2.0.RC2
- AngularUI 0.7 / Bootstrap 2.3

Tools

Die folgenden Technologien, sind für die Entwicklung und den Betrieb relevant:

- Glassfish 4.0
- IntelliJ IDEA, NetBeans, Eclipse (freie Wahl)
- Maven
- Git

Projektstruktur & Namenskonventionen

Im folgenden sind Projektstruktur und Namenskonventionen, getrennt für Backend und Frontend, beschrieben:

Backend

- Projektname:
 - conference
- Root-Package:
 - com.prodyna.pac.conference
- Komponenten-Packages:

- com.prodyna.pac.conference.<komponentenname>
- Technische Komponenten
 - com.prodyna.pac.conference.core
 - com.prodyna.pac.conference.monitoring
- Fachliche Komponenten
 - com.prodyna.pac.conference.conference
 - com.prodyna.pac.conference.room
 - com.prodyna.pac.conference.speaker
 - com.prodyna.pac.conference.talk
- Klassen- und Interfacenamen:
 - “Natürliche” Interfaces:
Interface-Namen dürfen weder Prefix noch Suffix beinhalten
 - REST-Ressourcen sind durch das Suffix `Resource` als solche kenntlich zu machen
 - Entitäten heißen wie die Fachlichkeit, welche sie repräsentieren (ohne Prefix/Suffix)
 - Klassen und Interfaces sollten immer den Sinn der Klasse erkennen lassen; nichtssagende Prefixe und Suffixe sind zu vermeiden (z.B. `*Util`, `*Handler`, `Abstract*`, `*Base`)
- Keine harte Trennung von Komponenten, stattdessen Checkstyle-Regeln zur Abgrenzung über Package Imports

Frontend

- Projektname: conference-angular
- Projektstruktur
 - libs/
Third-Party-Libraries
 - modules/
AngularJS-Module
 - index.html
Startseite
- Aufbau von AngularJS-Modulen
 - <module>/<template1>.html
 - <module>/<template2>.html
 - ...
 - <module>/<module>.js
- Ausnahme: app-Module
 - Beinhaltet sämtliches CSS gesammelt
app.css

Coding Standards

Die "Code Conventions for the Java Programming Language" sind einzuhalten:

<http://www.oracle.com/technetwork/java/codeconv-138413.html>