

Arbeitspakete LBS SS 18

Web-Client / Contentmanagement

(2 Personen) Ionic 2, Leaflet

Weiterentwicklung der MoCalInfo-Web-App in den Bereichen:

- Darstellung von Indoorkarten und ortsabhängigen Informationen verbessern
- Darstellung und Suche von Informationen (POIs, Personen, Events) verbessern
- Weiterentwicklung des CMS
- Konzept / Umsetzung einer Routen-Darstellung in Zusammenarbeit mit Team Routing-Lib

Routing-Lib, Campus-Geo-Backend

(2 Person) Java, Dropwizard

- Wegenetz aus OSM-Daten extrahieren
- Bestehende Routing-Lib mit extrahiertem Wegenetz testen und evtl. anpassen
- Service-API für das Wegenetz entwickeln
- Service-API für das Routing entwickeln
- Konzept / Umsetzung einer Routen-Darstellung in Zusammenarbeit mit Team Web-Client

3D-Kartenansicht

(2 Personen) Web-Technologien

Technische Möglichkeiten einer 3D-Kartenansicht (evtl. auch Indoor) für Web / Hybrid-Client evaluieren und prototypisch umsetzen.

Data-Backend - ortsbezogene Informationen

(2 Personen) Java, Dropwizard

- Aktualisierte Version im "Master" Testen und in "Production" überführen
- Authentifizierung auf Token-based umstellen
- Importer Personen & Stundenplan entwickeln
- THM-CAS-Anbindung
- Optimierung der REST-API und Anpassung an REST-Guidelines

MoCalInfo Indoor 360°

(2 Personen)

Mit den Erkenntnissen des Indoor 360° Prototypen eine Service-API umsetzen und die Ansicht in den Web-Client integrieren.

- Service-API konzipieren und umsetzen
- Integration in Web-Client
- Ortsgebundene Informationen mit einblenden

Mobiler Client (Android)

(2 Personen) Java, CARTO Mobile SDK (früher Nutiteq)

Weiterentwickeln der bestehenden Android-App.

- Positionierung: GNSS, Fingerprinting, QR-Code integrieren
- Routing-Konzept/Umsetzung in Zusammenarbeit mit Team Routing-Lib
- POI-Anzeige / Indoor-Gebäudeansicht optimieren

Mobiler Client (hybrid)

(2 Personen) Ionic 2

Weiterentwicklung der Hybrid-App nach dem Vorbild der Android-App.

- Positionierung: Fingerprinting integrieren
- GNSS, QR-Code weiterentwickeln / optimieren
- Routing-Konzept / Umsetzung in Zusammenarbeit mit Team Routing-Lib
- POI-Anzeige / Indoor-Gebäudeansicht optimieren

Ideen:

- Motion Tracking mit **ARCore** und absoluten Positionen untersuchen

Dead Reckoning Android-Komponente (Abschlussarbeit?)

Java, Android

- Kompass-Implementierung und -Kalibrierung prüfen
- Step-Detection überarbeiten
- PDR mit Kompass oder Accelerometer?