

# Installationsdokumentation

Dieses Dokument soll eine kurze Anleitung zur Installation der im System verwendeten Softwarekomponenten bieten.

## Inhaltsverzeichnis

|                                   |          |
|-----------------------------------|----------|
| <b>INSTALLATIONSdokumentation</b> | <b>1</b> |
| <b>INSTALLATIONSANLEITUNG</b>     | <b>1</b> |
| NODE.JS                           | 1        |
| MONGODB                           | 1        |
| OPENSOURCEROUTINGMACHINE(OSRM)    | 1        |
| RABBITMQ                          | 3        |
| <b>TESTSETUP</b>                  | <b>3</b> |
| ANDROID STUDIO                    | 3        |
| GENYMOTION EMULATOR & SHELL       | 3        |
| <b>AUSFÜHRUNGSANLEITUNG</b>       | <b>3</b> |
| <b>DOWNLOADLNKS</b>               | <b>6</b> |

## Installationsanleitung

Das System wurde auf Mac OS X Version 10.11.2 entwickelt und getestet.

Folgende Komponenten müssen vor Ausführung installiert werden:

### Node.js

Das System wurde unter Node.js Version 0.12.2 entwickelt. Kompatibilität auch unter dem latest stable Build 5.4.1 wurde noch nicht vollständig getestet.

### MongoDB

Das System verwendet MongoDB Version 3.2.0. Ein Downloadlink ist unter [2] angegeben. Ist das dort heruntergeladene File entpackt können im Verzeichnis /bin können die Kommandos zum Start einer MongoDB Instanz (kommando mongod) sowie zum Start der Mongo shell (Kommando mongo) ausgeführt werden.

### OpenSourceRoutingMachine(OSRM)

Das System verwendet OSM zur Berechnung von Routen. Detaillierte Installationsanleitungen finden sich unter [3] und [4] , werden aber folgenden Abschnitt noch einmal erläutert.

### OSRM Setup

Um OSRM unter Mac OSX zu kompilieren müssen folgende Dependencies über homebrew installiert werden:

## Mac OS X 10.7.1, 10.8.2

To compile the source on a Mac, please install the [homebrew](#) package system. It will provide all necessary dependencies:

```
brew install boost git cmake libzip libstxxl libxml2 lua51 luajit luabind tbb
```

Erst dann kann Mittels :

## General build instructions from source

To download OSRM using git and build it with CMake run the following commands.

```
git clone https://github.com/Project-OSRM/osrm-backend.git
cd osrm-backend
mkdir -p build
cd build
cmake .. -DCMAKE_BUILD_TYPE=Release
cmake --build .
sudo cmake --build . --target install
```

ein build erzeugt werden.

### *Kartenmaterial Setup*

Testdaten des Dienstgebers wurden für den Regierungsbezirk Köln erstellt. Kartenmaterial für den Regierungsbezirk Köln können unter [5] heruntergeladen werden. Die heruntergeladene .pbp Datei muss nun in das Verzeichnis osrm-backend verschoben werden. Dann können folgende Schritte ausgeführt werden:

1. Navigiere in das Verzeichnis osrm-backend/build
2. Führe das Kommando `ln -s ../profiles/lib/` aus um einen symbolischen Link zu erzeugen
3. Führe `ln -s ../profiles/car.lua profile.lua` aus .Die Profilkonfiguration von OSRM wird damit auf das Fortbewegungsmittel „Auto“ gesetzt. Routenberechnungen dieser Instanz nehmen also das Auto als Fortbewegungsmittel an und beachten Einbahnstraßen und andere für Autofahrer geltende Verkehrsregeln.
4. Navigiere zurück in das Verzeichnis „osrm-backend“ und führe folgende Befehle aus

- 4.1. `osrm-extract koeln-regbez-latest.osm.pbf` , dies entpackt das .pbf Verzeichnis ,welches OpenStreetMap Daten für Regierungsbezirk Köln enthält. Diese Operation kann einige Zeit in Anspruch nehmen.
- 4.2. Das entpackte Archiv enthält eine Menge vorberechnete Daten, mit deren Hilfe OSRM sehr kurze Antwortzeiten erreicht. Diese müssen für Verarbeitung in der Engine mit dem Befehl `osrm-prepare koeln-regbez-latest.osm.pbf` vorbereitet werden.
- 4.3. Zu Testzwecken kann der OSRM-Server nun mit `osrm-routed koeln-regbez-latest.osrm` gestartet werden. Repository des Projektes können im Verzeichnis `MS 6/Implementation` können Testrequests für den `CocoaRestClient`[8] gefunden werden, mit denen die Funktionen von OSRM ausprobiert werden können.

## RabbitMQ

Das System verwendet RabbitMQ für Benachrichtigungen über Zustandsänderungen der REST-Ressourcen. RabbitMQ kann über homebrew mit dem Kommando :  
`brew install rabbitmq`  
installiert werden [6]. Getestet wurde für Version 3.6.0.

## Testsetup

### Android Studio

Der Android-Dienstnutzer wurde für API Level 22 und Android Version 5.1 entwickelt. Entwickelt wurde in der Android Studio Version 1.5.1 [9]. Zur Ausführung wird ein JRE in der Version 1.8 benötigt.

### Genymotion Emulator & Shell

Zu Testzwecken sollte der Genymotion Emulator in Version 2.6 sowie die Genymotion Shell verwendet werden [7] . Zur Demonstration der Anwendungslogik der Android App wird ein Skript für die Genymotion Shell mitgeliefert, welches die GPS-Koordinaten des Emulators verändert und so eine Bewegung simuliert, die von der Situationsanalyse erfasst werden kann. Um den Emulator zu erhalten ist eine Registrierung bei Genymotion notwendig. Genymotion verwendet Oracles virtual Box VMM, diese muss ebenfalls installiert werden und kann unter [10] heruntergeladen werden.

## Ausführungsanleitung

### 1. Starte OSRM

- 1.1. Navigiere in das Verzeichnis ,in das das Kartenmaterial aus [5] während der Installation entpackt wurde

- 1.2. Führe in diesem Verzeichnis den Befehl „ osrm-routed koeln-regbez-latest.osrm „ aus. Dies startet eine Instanz der der OpenSourceRoutingMachine, die fortan auf localhost:5000 auf Anfragen lauscht.
- 1.3. Hinweis: Zur Verwendung unterschiedlicher Fortbewegungsmittel müssen weitere Instanzen gestartet werden. Während der Installation[4] wurde dieses Profil auf „Car“ gesetzt, die Maschine nutzt also standartmäßig Kartenmaterial für mit dem Auto befahrbare Routen.

## **2. Starte den RabbitMQ – Server**

- 2.1. Führe das Kommando „rabbitmq-server“ aus, der Server ist fortan auf localhost:15672 erreichbar. Defaultmäßig wird dieses Kommando in usr/local/sbin installiert.
- 2.2. Für bequemes Monitoring kann der Server unter localhost:15672 über den Browser erreicht werden. Defaultmäßig können hier Username : guest und password : guest angegeben werden.

## **3. Starte eine mongoDB Instanz**

- 3.1. Navigiere in das unter [2] heruntergeladene und entpackte Verzeichnis
- 3.2. wurde mongoDB global installiert kann dies mit dem Kommando „mongod“ durchgeführt werden, ansonsten funktioniert dies nur im Unterverzeichnis /bin.
- 3.3. Hinweis: Dienstgeber ist so konfiguriert, dass er auf dem default-Setup von MongoDB (MongoURL : mongodb://127.0.0.1:27017/test ) arbeitet. So kann das mongod-Kommando ohne Angabe von Parametern ausgeführt werden

## **4. Starte den Dienstgeber**

- 4.1. Kclone das Repository  
(<https://github.com/TobiGe/EISWS1516MichelsGerstenberg>)
- 4.2. Navigiere in das Verzeichnis MS 6/Implementation/dienstgeber/
- 4.3. Die verwendeten Node Module wurden mittels .gitignore nicht in git indiziert, führe npm install aus um sie mithilfe der package.json zu installieren
- 4.4. Starte den dienstgeber mit node dienstgeber.js –testdata , nun werden Testdatensätze in MongoDB eingepflegt. Die Testdaten enthalten einige Lebensmittelspenden in der Kölner Innenstadt (PLZ 50668). Die Abholtermine dieser Angebote werden dynamisch auf die aktuelle Uhrzeit beim Start des Dienstgebers angepasst Der Dienstgeber horcht nun auf localhost , Port 3000 auf Anfragen.
  - 4.4.1. Hinweis : Erneutes Ausführen des Dienstgebers mit der Option „-testdata“ generiert doppelte Datensätze ,die ggf. zu inkosistenten Ergebnissen führen

können, im Zweifelsfall können alle Datensätze aus MongoDB auf der Mongo-Shell (zu öffnen mit dem Kommando „mongo“) mit dem Kommando : `db.getCollectionNames().forEach(function(n){db[n].remove({})});` gelöscht werden.

## 5. Öffne Genymotion und starte die Android App

- 5.1. Erstelle ein virtuelles Device (>ADD)
- 5.2. Wähle Google Nexus 5 – 5.10 – API 22 – 1080x1920
- 5.3. Navigiere in das Verzeichnis  
MS6/Implementation/Android\_Client/ausführbare\_app
- 5.4. Drag & Drop das app-debug.apk in den Emulator , starte die App im Emulator , navigiere zum Tab „Sammelaktionen“

## 6. Führe ein Testscript für die Genymotion-Shell aus

- 6.1. Zur Demonstration der Anwendungslogik ,die die Fortbewegungssituation des Benutzers analysiert wird ein Genymotion-Shell Script mitgeliefert, welches die GPS-Koordinaten des Android Emulators im Bereich der passenden Testdaten des Dienstgebers (Köln 50668) verändert.
  - 6.1.1. Navigiere in das Verzeichnis MS 6/Konfigurationsdateien/
  - 6.1.2. Öffne die Datei Bewegungssimulationsskript in einem regulären Texteditor
  - 6.1.3. Im markierten Bereich (Zeile 39-41) muss der Pfad zur Genymotion-Shell-executable eingetragen werden.  
*Alternativ* können auch die beiden betroffenen Dateien („gpx\_to\_genymotion.sh“ und sample1.gpx) in das gleiche Verzeichnis wie die Genymotion-Shell executable bewegt werden.
  - 6.1.4. Das skript kann nun mit dem Befehl `sh gpx_to_genymotion.sh sample1.gpx` ausgeführt werden. Dies sollte nur ausgeführt werden, wenn der Genymotion Emulator vorher erfolgreich gestartet wurde ( siehe Schritt 5.1-5.4)
  - 6.1.5. Hinweis: Bei jeder erneuten Ausführung dieses Skripts wird die Android-App eine neue „Situation“ analysieren. Ob unter dem Viewtab „Sammelaktionen“ eine Aktion gefunden werden kann hängt davon ab ob es passende Abholtermine in den Testdaten gibt.

## 7. Starte den Tafel-Dienstnutzer

- 7.1. Navigiere in Implementation / TafelClient und führe node Dienstnutzer\_Tafel aus. Diese Instanz wurde nur Implementiert um zu demonstrieren wie der Nachrichtenaustausch zwischen Dienstgeber und dieser Instanz funktionieren soll.

## DownloadInks

- [1] <https://nodejs.org/download/release/v0.12.2/>
- [2] <https://www.mongodb.org>
- [3] <https://github.com/Project-OSRM/osrm-backend/wiki/Building-OSRM>
- [4] <https://github.com/Project-OSRM/osrm-backend/wiki/Running-OSRM>
- [5] <http://download.geofabrik.de/europe/germany/nordrhein-westfalen/koeln-regbez.html> - Dort die Datei : koeln-regbez-latest.osm.pbf
- [6] <https://www.rabbitmq.com/install-homebrew.html>
- [7] <https://www.genymotion.com/#!/download>
- [8] <https://github.com/mmattozzi/cocoa-rest-client>
- [9] <http://tools.android.com/download/studio/builds/1-5-1> (Android Studio 1.5.1)
- [10] <https://www.virtualbox.org/wiki/Downloads>