

**ENTWICKLUNGSPROJEKT  
INTERAKTIVE SYSTEME  
“BikeFriends”**



**WS2016/2017**

**Dozenten**

Prof. Dr. Kristian Fischer  
Prof. Dr. Gerhard Hartmann

**Betreut von**

Ngoc-Anh Dang  
Daniela Reschke

Patrick Reimringer  
Vanakh Chea

<b>1 INSTALLATIONSdokUMENTATION</b>	<b>3</b>
1.1 Systemanforderungen	3
1.1.1 Server	3
1.1.2 Client	3
1.2 Installationsschritte	4
1.2.1 Server	5
1.2.2 Client	5
1.3 Abweichungen von MS2	6
<b>2 FAZIT</b>	<b>6</b>
<b>3 PROZESSASSESSMENTS</b>	<b>7</b>

# 1 INSTALLATIONSdokUMENTATION

## 1.1 Systemanforderungen

Die Hardware, auf dem das System ordnungsgemäß laufen soll, muss diverse Anforderungen erfüllen. Im Folgenden werden Anforderungen an den Server- und Clienthardware getrennt aufgelistet.

### 1.1.1 Server

- Betriebssystem: Windows oder MacOS, für andere Betriebssysteme kann die Funktionstüchtigkeit nicht garantiert werden.
- Installierte Java-Version: mindestens 1.7 (MacOS)
- node.js und npm
- Android Studio 2.2 oder neuer
- mongoDB: 3.0.2

### 1.1.2 Client

- Smartphone mit Android 6.0 Marshmallow, API Level 23.
- Eingebauter GPS-Empfänger
- WLAN

Da das System eine RESTful API bereitstellt, können HTTP Methoden auch über einen aktuellen Webbrowser verwendet werden. Für die Evaluation des Systems spielt dieser Hinweis keine wichtige Rolle.

## Datenbank starten

1. Terminal öffnen
2. "mongod" ausführen

Hinweis: mongoDB schreibt in die Datenbank mydb in die Collections „profiles“ und „events“

## Beispieldaten in die Datenbank importieren

Diese Daten umfassen Benutzerprofile und die Veranstaltungsdaten. Die Benutzerprofile enthalten auch die Login-Daten, die auf dem Client eingegeben werden.

1. In das Verzeichnis mit den Dateien profiles.json und events.json wechseln.
2. *mongoimport -db mydb -collection profiles -file profiles.json* bzw. *mongoimport -db mydb -collection events -file events.json* ausführen.

## Express.js-Server starten

1. Terminal öffnen
2. in das Projektverzeichnis (/Implementation/Bikefriends\_server/server/), in der sich die Datei „app.js“ befindet, wechseln.
3. "nodemon app" ausführen

## Netzwerkeinstellungen

Sowohl der Server als auch die Clients müssen sich im selben Netzwerk befinden.

## 1.2 Installationsschritte

Sind alle Anforderungen erfüllt, kann mit der Installation fortgefahren werden.

Implementation aus dem Github-Repository herunterladen:

<https://github.com/Preim/EISWS1617CheaReimringer/tree/master/M3/Implementation>

### 1.2.1 Server

1. Node.js kann über die folgende URL heruntergeladen und installiert werden. Dort ist eine offizielle Installationsanleitung zu finden: <http://nodejs.org/>
2. Über <https://www.mongodb.com/download-center#previous> kann das DBMS MongoDB heruntergeladen und installiert werden.

Empfohlen wird die **Version 3.0.14**.

- a. Nach der MongoDB-Installation muss im Root-Verzeichnis des Systems der Ordner `/data/db` angelegt werden.
  - b. Die Datenbank kann mit dem Befehl `mongod` gestartet werden.
  - c. Beispieldaten (`profiles.json` und `events.json`) können über den Befehl `mongoimport --db mydb --collection profiles --in profiles.json` bzw. `mongoimport --db mydb --collection events --in events.json`
3. Die Installation der Node.js-Module geschieht über den Befehl: `npm install`. Dieser Befehl muss im Projektordner `Implementation/Bikefriends_Server/server/` ausgeführt werden.
  4. Die Node.js-Serverinstanz kann über den Befehl `node ./bin/www`

### 1.2.2 Client

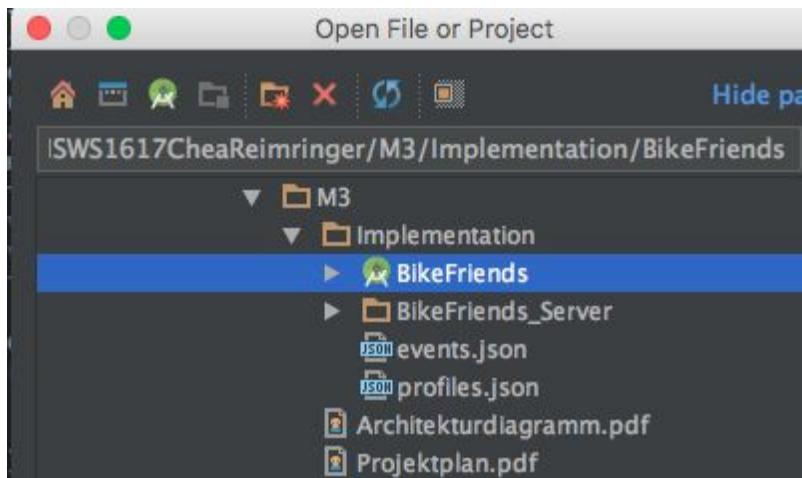
Die Android-App kann über Android Studio auf den Client installiert werden.

1. Android Studio, sofern noch nicht vorhanden, installieren:

<https://developer.android.com/studio/index.html>

Zusätzlich wird Android 6.0, API level 23 benötigt. Diese SDK kann über den Menüpunkt `Tools>Android>SDK Manager` installiert werden.

2. Projektimport über File > Open...



3. *Run > Run "App"* erlaubt das Ausführen der App auf einem angeschlossenen Android-Gerät oder in einem Emulator.

### 1.3 Abweichungen von MS2

Im Zuge der Implementation wurden fortlaufend Änderungen am Datenmodell vorgenommen. Fahrprofile sind nun keine separaten Datenobjekte mehr, sondern sind ein Teil des Benutzerprofils.

Ein "Flüchtigkeitsfehler" trat bei der Modellierung der Veranstaltungsdaten auf. Die (Start-)Uhrzeit einer Veranstaltung wurde erst bei der Implementation hinzugefügt.

## 2 FAZIT

Im Rahmen der Veranstaltung konnte das Projekt leider nicht vollständig implementiert werden, was sich aufgrund von Zeitmangel und fehlender Erfahrung in App-Programmierung zurückführen lassen. Dennoch konnten wir das Hauptalleinstellungsmerkmal "Matching von Fahrradfahrern" implementieren, welches aber stark reduziert werden musste. Dies kam dadurch zustande, da wir wenig bis keine Erfahrung hinsichtlich der Verarbeitung von GPS Daten mit Android hatten. Allgemein wurden zu viele Ziele in Meilenstein 1 definiert anstatt sich auf wenige wichtige Ziele festzulegen, was uns mitte Meilenstein 2 bewusst wurde.

### **3 PROZESSASSESSMENTS**

Die führung des Projektplans wurde während des Projektes nicht sehr gepflegt, was zu Zeitlichen Problemen führte hinsichtlich der fertigstellung und der zusammenführung der einzelnen Artefakte. Oft wurden Abgabe erst im letzten Moment fertiggestellt oder sogar mit fehlenden Artefakten abgegeben.

Während der gesamten Projektdauer kam es dadurch oft zu einer schlechten Arbeitsaufteilung. Dennoch wurde sich meist täglich getroffen oder am Projekt gearbeitet, um das Projekt stetig nach vorne zu bringen. Lediglich zwischen Weihnachten und Neujahr kam es aufgrund der Feiertage zu einer kurzen unproduktiven Phase.