



# **Konzeption und Realisierung einer mo- bilien Anwendung zur Unterstützung von gestressten Patienten mithilfe des "Mind- ful Walking Gedankens"**

Bachelorarbeit an der Universität Ulm

**Vorgelegt von:**

Dennis Kozlowski  
dennis.kozlowski@uni-ulm.de

**Gutachter:**

Prof. Dr. Manfred Reichert

**Betreuer:**

Dr. Rüdiger Pryss

2018

Fassung 22. Mai 2018

© 2018 Dennis Kozlowski

This work is licensed under the Creative Commons Attribution 4.0 International (CC BY 4.0) License. To view a copy of this license, visit <https://creativecommons.org/licenses/by/4.0/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.

Satz: PDF-L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>

---

## Kurzfassung

Heutzutage bewegen wir Menschen uns immer von einem Ort zum anderen. Auf uns warten Termine, die nicht verschoben werden können. Die Arbeit, die erledigt werden muss. Kinder die abgeholt werden müssen. All diese Dinge können einen überlasten und stressen, da uns kaum Zeit bleibt für Entspannung und Erholung, um den Kopf freizukriegen und in sich zu gehen. Haben wir jedoch mal ein paar Minuten Zeit, so können diese dabei helfen, mithilfe von Übungen. Den Körper und Geist zu beruhigen. Eine davon ist Mindful Walking. Mindful Walking ist eine Art von Gehmeditation, bei der sich während des Gehens, die Umwelt und der eigene Körper bewusst gemacht werden soll. Diese Arbeit beschäftigt sich mit der Konzeption und Realisierung einer mobilen Anwendung zum Mindful Walking. Nutzer können die Anwendung auf dem Smartphone nutzen, um angeleitete Mindful Walking Übungen durchzuführen. Dabei werden auch verschiedene Daten erfasst, welche im Nachhinein analysiert und ausgewertet werden. Die Ergebnisse, wie z.B. die Geschwindigkeit des Nutzers, sind in der App einsehbar. Mithilfe von Mindful Walking kann die Gesundheit und Konzentration des Nutzers positiv beeinflusst werden.

---

## **Danksagung**

Ich möchte an dieser Stelle den Personen danken, die mich bei dieser Bachelorarbeit unterstützt und zum Gelingen beigetragen haben. Ein besonderer Dank gebührt hierbei Dr. Rüdiger Pryss, welcher mich auf dieses Thema aufmerksam gemacht hat und mich bei dieser Arbeit betreut hat. Auch möchte ich mich beim Prof. Dr. Manfred Reichert bedanken, der diese Arbeit genehmigt und begutachtet hat. Zuletzt möchte ich noch meiner Familie und meinen Freunden danken, die mich während des Studiums immer unterstützt haben.

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Problemstellung . . . . .	2
1.2	Zielsetzung . . . . .	2
1.3	Aufbau der Arbeit . . . . .	2
<b>2</b>	<b>Grundlagen</b>	<b>3</b>
2.1	Volley . . . . .	3
2.2	JSON . . . . .	4
2.3	Google Location Service API . . . . .	5
<b>3</b>	<b>Anforderungsanalyse</b>	<b>6</b>
3.1	Funktionale Anforderungen . . . . .	6
3.2	Nicht-funktionale Anforderungen . . . . .	7
<b>4</b>	<b>Architektur</b>	<b>9</b>
4.1	Architekturübersicht . . . . .	9
4.2	Genereller Ablauf . . . . .	9
4.3	Architektur der App . . . . .	11
<b>5</b>	<b>Vorstellung des Mindful Walking Rahmenwerks</b>	<b>16</b>
5.1	Vorstellung der App . . . . .	16
5.1.1	Starten der App . . . . .	16
5.1.2	Registrierung in der App . . . . .	16
5.1.3	Anmeldung mit Benutzername und Passwort . . . . .	19
5.1.4	Passwort vergessen . . . . .	20
5.1.5	Aktivierungsmail erneut versenden . . . . .	20
5.1.6	Statische Fragebögen . . . . .	21
5.1.7	Menü . . . . .	23
5.1.8	Startseite . . . . .	23
5.1.9	Übung durchführen . . . . .	25
5.1.10	Studien . . . . .	27

5.1.11 Fragebögen . . . . .	28
5.1.12 Ergebnisse . . . . .	29
5.1.13 Benutzerkonto . . . . .	29
5.1.14 Benachrichtigungen . . . . .	30
5.1.15 Einstellungen . . . . .	30
<b>6 Implementierung</b>	<b>32</b>
6.1 Volley Request . . . . .	32
6.2 Google Play Services Location API und die Geschwindigkeitsmessung	34
<b>7 Anforderungsabgleich</b>	<b>36</b>
7.1 Funktionale Anforderungen . . . . .	36
7.2 Nicht-Funktionale Anforderungen . . . . .	37
<b>8 Zusammenfassung und Ausblick</b>	<b>38</b>
8.1 Zusammenfassung . . . . .	38
8.2 Ausblick . . . . .	38
8.2.1 Offline-Modus . . . . .	39
8.2.2 Verbesserung am User-Interface . . . . .	39
8.2.3 GPS-Alternativen . . . . .	39
8.2.4 Netzwerkkommunikation . . . . .	39
<b>Literaturverzeichnis</b>	<b>43</b>

# 1 Einleitung

„Mindfulness“, zu deutsch Achtsamkeit ist eine bestimmte Form der Aufmerksamkeit, die absichtsvoll und unvoreingenommen ist und sich auf den gegenwärtigen Moment bezieht [1]. Mindfulness hat ihren Ursprung in der traditionellen, asiatischen/östlichen Meditation und ist in vielen Bereichen sehr relevant geworden, darunter auch in der Gesundheitsforschung. Bei Mindfulness steht der Körper im Fokus vieler Achtsamkeitsübungen (mindfulness exercises). Beispiele für solche Übungen sind die Mindfulness-based Stress Reduction [2] und die Mindfulness-based Cognitive Therapy [3]. Der Anwender solcher Übungen wird darauf trainiert Körperfunktion, wie z.B. das Gehen und Atmen, absichtsvoll und unvoreingenommen durchzuführen. Ziel dieser Übungen ist es den Anwender weg vom ständigen Gedankenfluss des Alltags und der darauffolgenden Reaktionen zubringen und diesen in einen Zustand des „hier und jetzt Sein“ zu führen, indem der Moment wahrgenommen wird. Diese Übungen können die Konzentration und den Fokus stärken, den Stress verringern und den Anwender emotional stärken. Sogar Depression kann durch Achtsamkeit entgegengewirkt werden. Eine solche Mindfulness-Übung ist Mindful Walking. Während einer Mindful Walking Übung soll ganz besonders auf die Bewegungsabläufe geachtet werden.

Ein typischer Übungsablauf:

- a) Vor dem Gehen auf die Haltung achten. Sich der einzelnen Bewegungen bewusst sein, die dabei helfen die Balance zu halten.
- b) Auf die Atmung achten.
- c) Fuß langsam anheben.
- d) Fuß langsam nach vorne bewegen.
- e) Fuß langsam auf den Boden absetzen. Spüren wie die Erde/Steine unter dem Fuß etwas nachgeben.
- f) Das Gewicht langsam auf den vorderen Fuß verlagern und den hinteren Fuß anheben, wobei die Zehen immer noch den Boden berühren.

## 1.1 Problemstellung

Um Mindful Walking Übungen und die daraus resultierenden Erfahrungen verbessern zu können, müssen dem Nutzer und den Forschern Daten vorliegen. Diese Daten können mithilfe von „mobile crowd sensing“ [4] gesammelt werden. Die Daten verschiedener Anwender werden gesammelt, analysiert und evaluiert um sowohl die individuelle, als auch die allgemeine Erfahrung von Mindful Walking zu verbessern.

## 1.2 Zielsetzung

Das Ziel dieser Arbeit ist es, eine einfach bedienbare und leicht verständliche Applikation zu erstellen, die den Nutzer tagtäglich bei seinen Mindful Walking Übungen begleiten und unterstützen soll. Der Nutzer soll mithilfe der App regelmäßig zu diesen Übungen aufgefordert werden. Die Daten, die vor und während der einzelnen Übungen erfasst werden, sollen die Erfahrung und die Übung des Nutzers verbessern. Diese Übungen können auch gesundheitliche Vorteile mit sich bringen. Jedoch soll nicht nur der Nutzer von diesen Übungen profitieren, sondern auch die Allgemeinheit. Die Ergebnisse können zu Forschungszwecken analysiert und evaluiert werden.

## 1.3 Aufbau der Arbeit

Die Arbeit ist in sieben Kapitel unterteilt. Nach der eben erfolgten Einleitung, folgen die technischen Grundlagen, die für dieses Projekt wichtig sind. In Kapitel 3 werden die Anforderungen der Anwendung Mindful Walking erläutert. Das 4. Kapitel beinhaltet die Darstellung der Architektur des Projekts. Zuerst wird eine Übersicht der Architektur gegeben, danach wird ein genereller Ablauf der Anwendung aufgezeigt und anschließend wird die Datenstruktur der mobilen Anwendung aufgezeigt. Im 5. Kapitel wird die App vorgestellt und ihre Funktionen beschrieben. Kapitel 6 stellt einen Teil der Implementierung vor. In Kapitel 7 werden die Anforderungen aus Kapitel 3 mit dem Stand der Entwicklungen abgeglichen. Im letzten Kapitel wird sowohl die Arbeit kurz zusammengefasst, als auch ein Ausblick auf mögliche Weiterentwicklungen gegeben.



## 2 Grundlagen

### 2.1 Volley

Für die Netzwerkanfragen an die API wird in diesem Projekt nur Volley verwendet. Die API wird von der Universität gestellt. Bei dieser handelt es sich um Volley ist eine HTTP Bibliothek, welche die Netzwerkkommunikation für Android Apps erleichtert und schnell bewältigt. Volley wurde von Google entwickelt und 2013 vorgestellt [5]. Volley funktioniert asynchron zur eigentlich App, was dazu führt, dass der Nutzer nichts von diesen Anfragen mitbekommt während er die App nutzt. Die Anfragen werden in einen extra Thread ausgelagert, womit der Main-Thread nicht blockiert wird. Sollte bei einer sich bei einer Netzwerkanfrage die Antwort bereits im Cache befinden, so wird die Antwort aus dem Cache gelesen, ansonsten wird einen neue Anfrage gestellt. Falls es nun zu Fehlern in solchen Anfragen kommt, führen diese nicht zum Absturz der App.

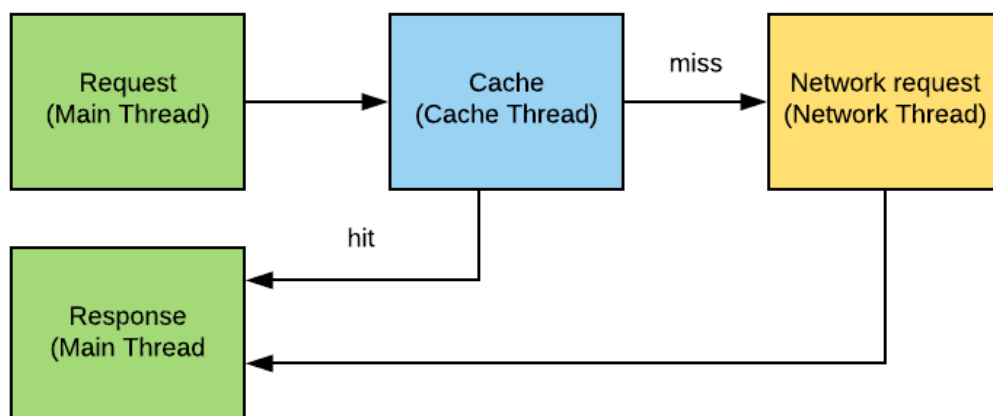


Abbildung 2.1: Volley Request

## 2.2 JSON

Da die API mit JSON arbeitet, wird JSON hier kurz erläutert. Bei JSON, kurz für JavaScript Object Notation handelt es sich um ein Datenformat, das zur Speicherung und zum Austausch von Daten verwendet wird [6]. Ein Beispielcode eines JSON-Objekts ist als Nächstes zu sehen.

```
1 {
2   "data":{
3     "type":"elements/elements",
4     "attributes":{
5       "elements":[
6         {
7           "name":"Headline01",
8           "elementtype":"elements/headlines",
9           "position":1,
10          "is_active":true
11        }
12      ]
13    }
14  }
15 }
```

Listing 2.1: JSON-Objekt

JSON-Objekte sind immer gleich aufgebaut. Sie beginnen mit einer offenen, geschweiften Klammer. Danach folgt eine Kennzeichnung, in Fall der Abbildung mit „data“ gefolgt von einem Doppelpunkt und den Werten. JSON kennt verschiedene Datentypen, dazu gehören Boolesche Werte, Zahlen, Zeichenketten und Nullwerte. Außerdem kennt JSON Arrays und Objekte. Arrays beginnen und enden mit eckigen Klammern. Sie besitzen geordnete Listen von Werten (unterschiedliche Typen möglich), die durch Kommata getrennt geteilt. Ein Objekt beginnt und endet mit geschweiften Klammern. Dieses enthält eine ungeordnete Liste von Eigenschaften, die durch Kommata getrennt werden. Eine JSON-Objekt endet mit einer geschlossenen, geschweiften Klammer. Die Abbildung 2.2 zeigt einen generellen Aufbau eines JSON-Objekts und Arrays. Die wichtigste Eigenschaft von JSON ist die Tatsache, das JSON sowohl vom Menschen, als auch vom Computer leicht les- und interpretierbar ist.

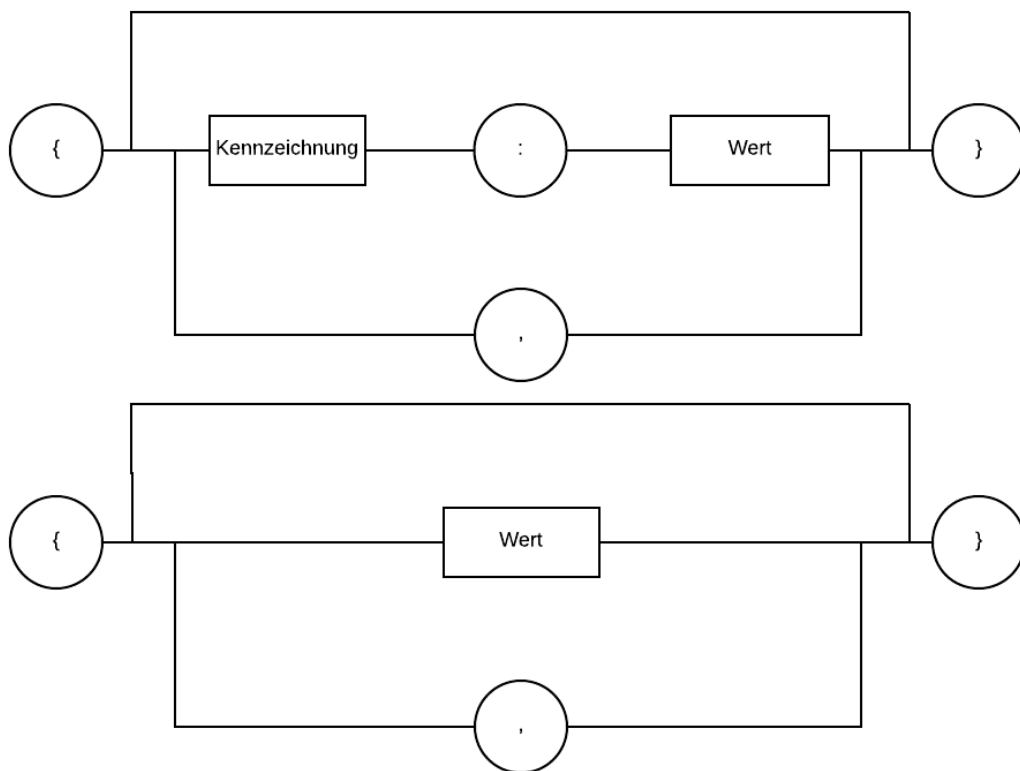


Abbildung 2.2: Aufbau JSON-Objekt (oben) und JSON-Array (unten)

## 2.3 Google Location Service API

Um die Geschwindigkeit per GPS zu messen standen zwei Möglichkeiten zur Verfügung. Zum einen die Google Location Service API und zum anderen Androids Location API. Für die App wurde die Google Location Service API genutzt [7]. Im Vergleich zur Androids Location API verfügt die Google Location Service API, die ein Teil der Google Play Services ist über ein hoch-gradiges Programmiergerüst (high-level API). Diese kümmert sich automatisch um die Positionsanbieter, die Bewegung des Nutzers und die Genauigkeit. Außerdem kümmert sich die API um einen besseren Akkuverbrauch durch ein besseres Planen der Positionsupdates, anhand von gegebenen Parametern. Die Abläufe werden in Kapitel 6.2 etwas genauer beschrieben.

# 3 Anforderungsanalyse

Dieses Kapitel definiert die Anforderungen an das Projekt. Diese werden in funktionale und nicht-funktionale Anforderungen unterteilt.

## 3.1 Funktionale Anforderungen

Nun werden die funktionalen Anforderungen aufzeigt. Die wichtigsten Funktionen, die dem Benutzer zur Verfügungen stehen werden kurz erläutert.

<b>Nr.</b>	<b>Beschreibung</b>	<b>Problembeschreibung</b>
1.	Registrierung in der App	Ein Benutzerkonto ist Voraussetzung für die Benutzung der App, daher soll es möglich sein, direkt auf dem Gerät ein solches Benutzerkonto zu erstellen.
2.	An Studien teilnehmen	Der Nutzer soll die Möglichkeit haben, sich zu verschiedenen Studien einzutragen.
3.	Studieneinladungen annehmen	Der Nutzer soll persönliche Einladungen zur Studien erhalten können. Diese Einladungen können in der App eingesehen und akzeptiert werden.
4.	An einer privaten Studie teilnehmen	Der Nutzer soll sich mithilfe der App in eine private Studie einschreiben können, sofern er das Passwort kennt.
5.	Fragebögen einer Studie ausfüllen	Der Nutzer soll mithilfe der App Fragebögen ausfüllen können. Diese Fragebögen werden von den einzelnen Studien zur Verfügung gestellt. Ist der Nutzer nicht in einer Studie eingeschrieben, so kann dieser auch nicht auf die Fragebögen der Studie zugreifen

6.	Ergebnisse anzeigen	Die Ergebnisse aus der Übung sollen für den Nutzer angezeigt werden.
7.	Ergebnisse an den Server schicken	Die Ergebnisse der Übungen sollen an den Server geschickt werden. So können diese zu Forschungszwecken analysiert und evaluiert werden.
8.	Meldung von Benachrichtigungen	Der Nutzer soll mithilfe der Benachrichtigungen auf seine Übung hingewiesen werden. Die Häufigkeit der Benachrichtigungen und den Zeitraum der Benachrichtigungen sollen einstellbar werden können.
9.	Beginn der Übung	Zu Beginn der Übung soll der Nutzer Fragebögen zur aktuellen Stimmung ausfüllen. Danach wird die aktuelle Gehgeschwindigkeit gemessen. Ein Ton wird ausgegeben sobald die Geschwindigkeit gemessen wurde. Danach soll die Zielgeschwindigkeit ausgewählt werden können.
10.	Übung durchführen	Während der Nutzer die Übung durchführt, soll immer ein Ton ausgegeben werden, sobald der Nutzer die Zielgeschwindigkeit überschreitet. Durchschnittsgeschwindigkeit, Anzahl der Tonausgaben und die Zeiten sollen gespeichert werden.
11.	Vibration	Anstatt der Tonausgabe während der Übung, sollte der Nutzer auch Vibration auswählen können.

Tabelle 3.1: Funktionale Anforderungen

### 3.2 Nicht-funktionale Anforderungen

Die nicht-funktionalen Anforderungen beschreiben, wie die App mit Aspekten wie Aussehen, Handhabung und Datenschutz umgeht.

Nr.	Beschreibung	Problembeschreibung
-----	--------------	---------------------

### 3 Anforderungsanalyse

---

1.	Technische Anforderungen	Die App soll unter Android lauffähig sein.
2.	Stabilität und Robustheit	Die App soll gegenüber Falscheingaben korrekt reagieren. Nach einem Neustart des Systems soll die Anwendung fortgeführt werden können

Tabelle 3.2: Nicht-funktionale Anforderungen

# 4 Architektur

Dieses Kapitel beschreibt die Architektur des Mindful Walking Projekts. Zuerst wird eine kurze Übersicht über die Architektur gegeben (Kapitel 4.1). Danach geht es weiter mit einem typischen Ablauf in Kapitel 4.2. Zuletzt wird die Architektur der App in Kapitel 4.3 beschrieben.

## 4.1 Architekturübersicht

Das Mindful Walking Projekt besteht aus zwei Komponenten. Die erste Komponente ist der Server bzw. das Backend (API inkludiert). Dieser Server wurde vom Institut für Datenbanken und Informationssysteme bereitgestellt. Die zweite Komponente ist die mobile Anwendungen. Diese wurde für Android Geräte entwickelt und basiert deshalb auf Java. Es existiert zu keiner Zeit eine direkte Kommunikation zwischen Server und App, weshalb die App nicht direkt auf die Daten zugreifen kann. Die Kommunikation läuft jedoch über eine JSON-API. Ein grober Aufbau ist in der Abbildung 4.1. zu sehen.

## 4.2 Genereller Ablauf

Nachdem die App installiert und geöffnet wurde erscheint das Begrüßungsfenster. Weiter kommt man entweder zum Loginbereich oder zur Registrierung. Ein Benutzerkonto ist Voraussetzung für die Nutzung dieser App, weshalb eine Registrierung Pflicht ist. Nachdem die Daten des Nutzers eingegeben wurden erhält dieser eine Rückmeldung über die erfolgreiche Registrierung. Eine E-Mail wird an den Benutzer verschickt, welche einen Aktivierungslink enthält. Durch diesen wird das Benutzerkonto aktiviert und die Registrierung ist abgeschlossen. Der Nutzer kann nun im Loginbereich seine Daten eingeben und sich einloggen. Falls statische Fragebögen vorliegen, wird der User beim erstmaligen Einloggen dazu aufgefordert, diese

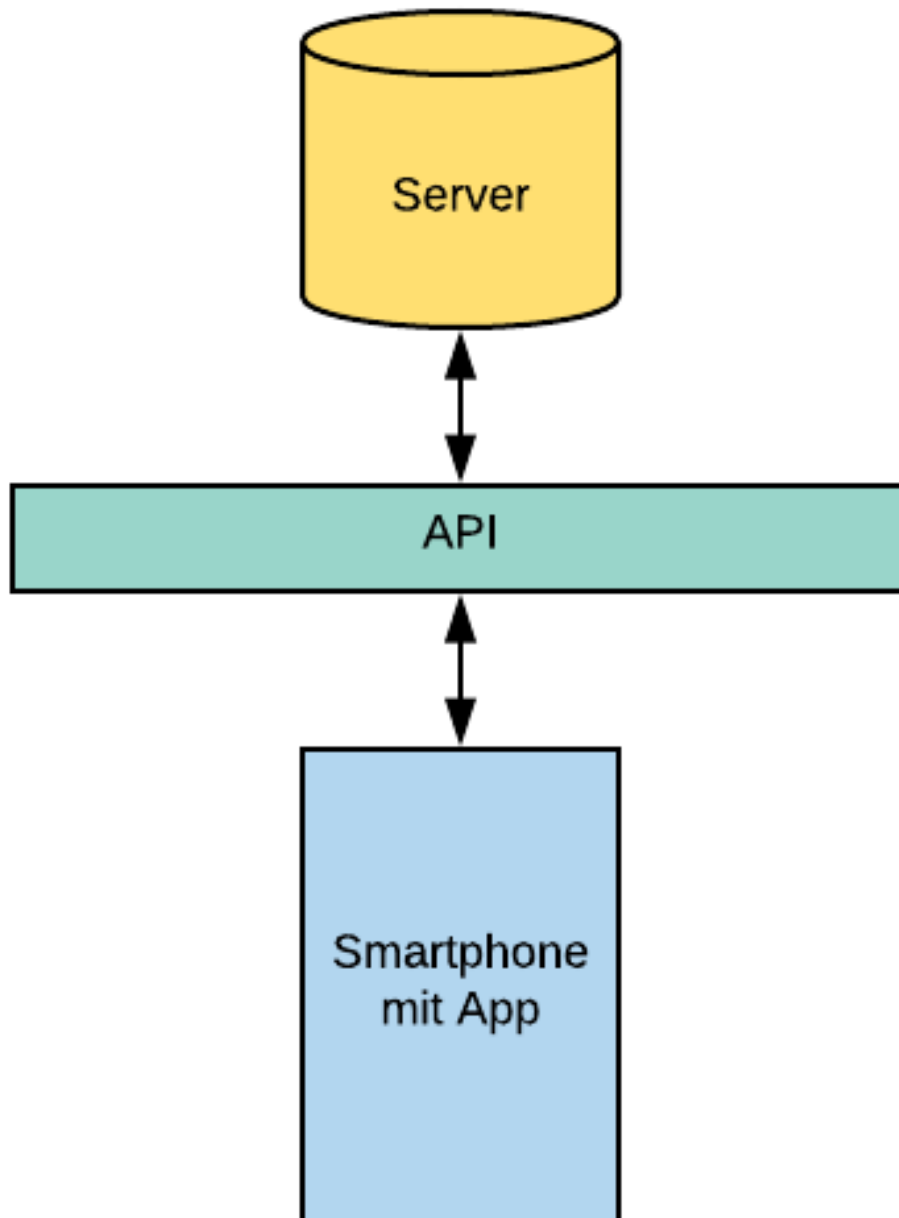


Abbildung 4.1: Übersicht der Architektur



auszufüllen. Anschließend wird der Nutzer zum Hauptbereich weitergeleitet. Durch das Menü im Hauptbereich kommt der Nutzer zu den verschiedenen Bereichen. Im Bereich Startseite befindet sich ein Button. Dieser führt den Nutzer zur eigentlichen Mindful Walking Übung. Bevor es mit der Übung losgeht, wird der Nutzer darum gebeten, Fragebögen auszufüllen. Anschließend wird die Übung gestartet. Im Bereich Studien, werden verschiedene Studien aufgeführt. Zu diesen Studien kann sich der Nutzer gegebenenfalls eintragen oder falls möglich austragen. Der Bereich Fragebögen zeigt alle Fragebögen auf, die der Nutzer ausfüllen kann. Die Fragebögen werden von den Studien, zu denen sich der Nutzer eingetragen hat, zur Verfügung gestellt. Im Bereich Ergebnisse, werden die vorherigen Ergebnisse der einzelnen Mindful Walking Übungen aufgeführt. Der Nutzer kann im Bereich Benachrichtigungen, Zeiten angeben, zu denen er auf Mindful Walking Übungen aufmerksam gemacht werden soll. Der letzte Bereich sind die Einstellungen. Hier kann die Sprache geändert, sowie auf Sound / Vibration umgestellt werden. Die Abläufe sind in den Abbildungen 4.2 4.3 4.4 zu sehen.

### **4.3 Architektur der App**

Der Login/Registrierungsprozess ist mit Activities aufgebaut worden. Sobald der Nutzer einmal eingeloggt ist wird die MainActivity genutzt. Diese beinhaltet die meisten Methoden. Die MainActivity ist also hauptsächlich für die Funktionen der App zuständig. Außerdem gibt es einen Sessionmanager, der sich um kleinen Datenmengen, wie z.B. den Token und die Anmeldedaten kümmert. Es wurden Pager verwendet um die Daten auf Listen und andere Widgets verteilen zu können. Die Model-View befindet sich in Abbildung 4.5.

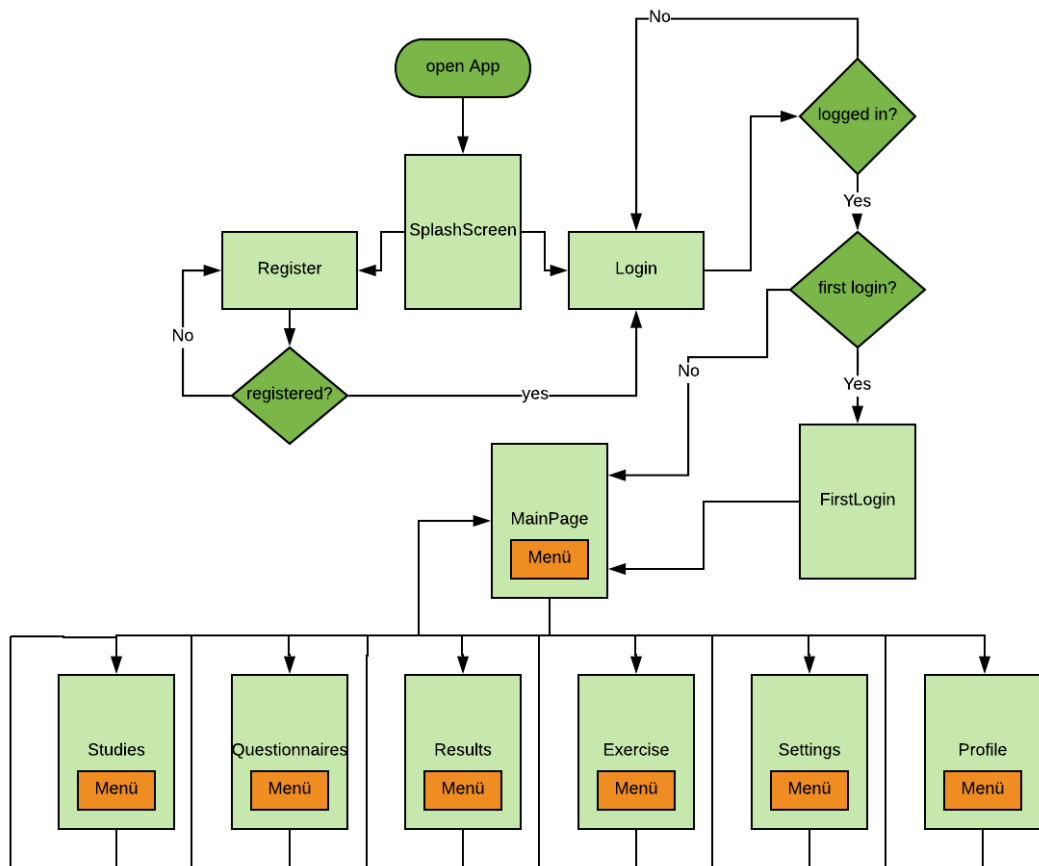


Abbildung 4.2: Genereller Ablauf

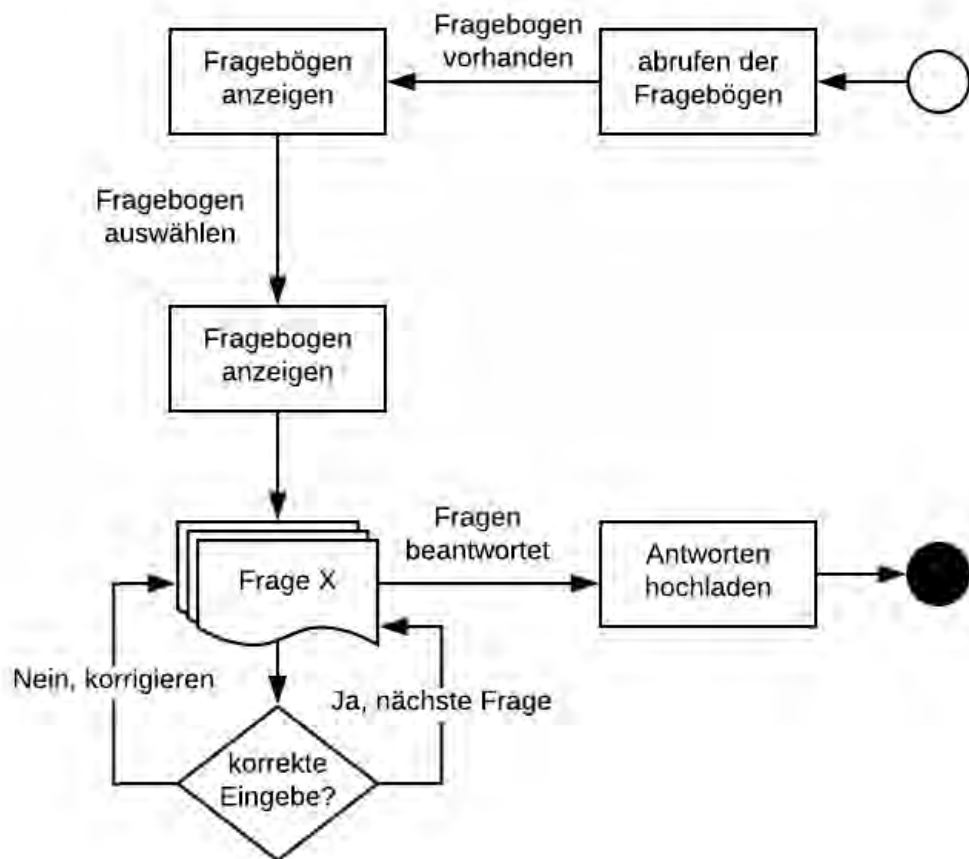


Abbildung 4.3: Fragebögen anzeigen und beantworten

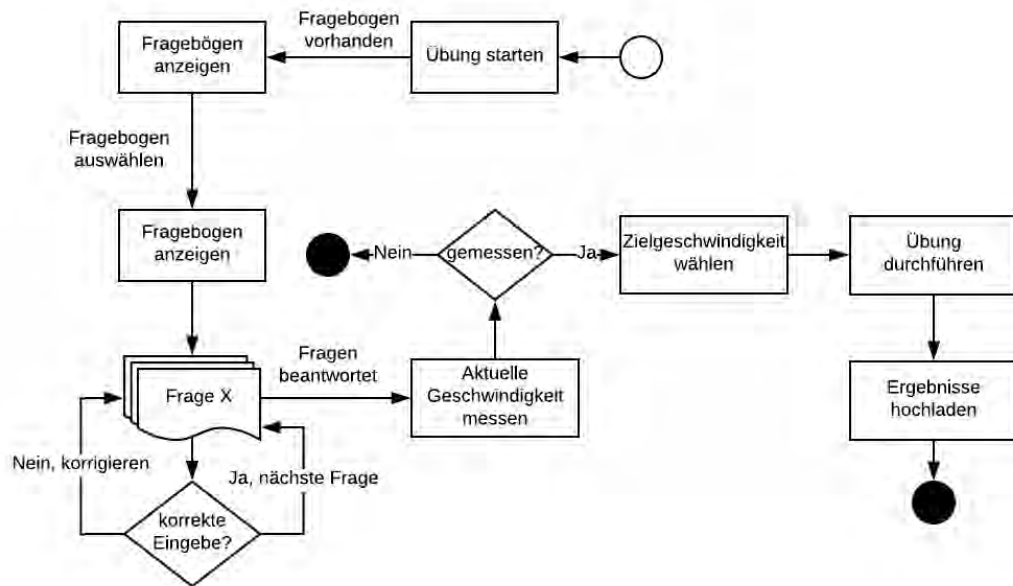


Abbildung 4.4: Übung durchführen

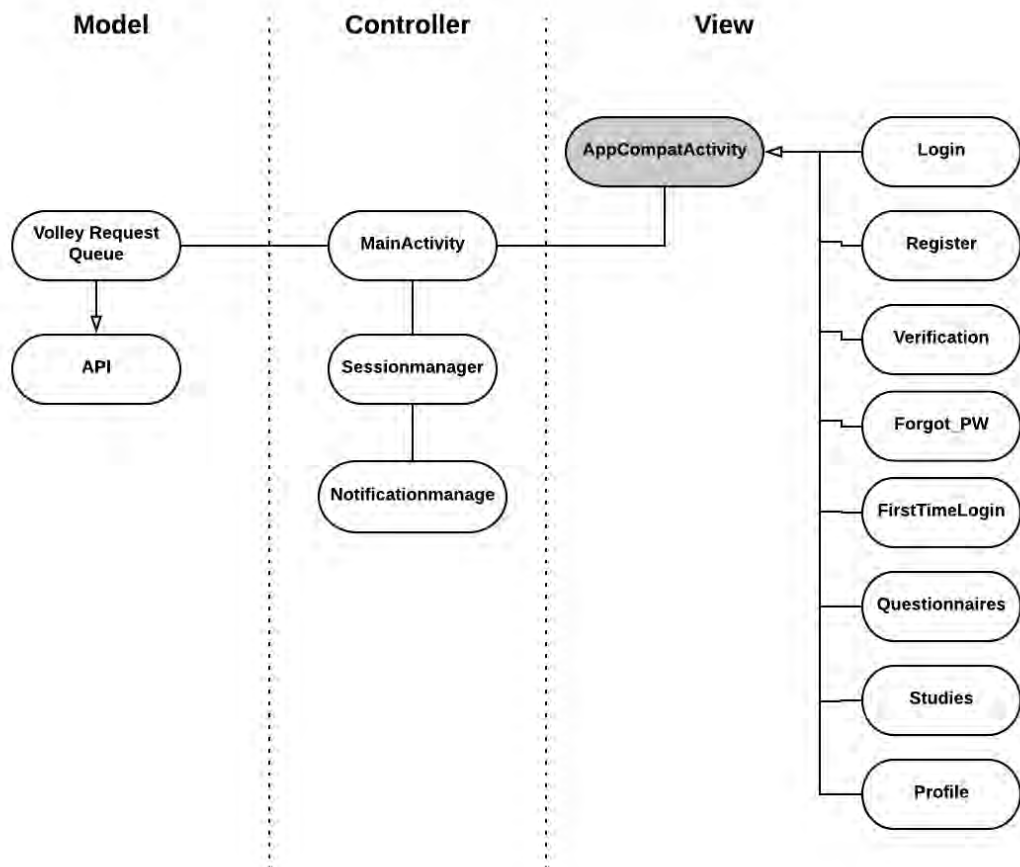


Abbildung 4.5: Model-View-Controller

# **5 Vorstellung des Mindful Walking Rahmenwerks**

In diesem Kapitel wird das Projekt Mindful Walking aus der Sicht der Benutzers dargestellt. Die einzelnen Funktionen der Anwendung und ihre zugehörigen grafischen Oberflächen werden erläutert.

## **5.1 Vorstellung der App**

Das Ziel der App ist es, dem Nutzer eine Plattform zu geben, die ihm dabei hilft, seine Mindful Walking Erfahrung und Übung zu verbessern. Dies wird gewährleistet, indem die Ergebnisse der Übung festgehalten und analysiert werden. Die App ist auf Android Geräten ab der Version 4.0.3 lauffähig sein. Optimiert wurde die App für neue Android Versionen.

### **5.1.1 Starten der App**

Durch das Öffnen der App erscheint der Splashscreen. Dieser verschwindet nach ca. einer Sekunde und der Nutzer befindet sich in der ersten Ansicht. Da viele Menschen keine Vorstellung von Mindful Walking haben, befindet sich in der ersten Ansicht eine Link, der den Begriff Mindful Walking erläutert. Außerdem befinden sich im unteren Bereich zwei Buttons. Diese führen zum einen zur Registrierung und zum anderen zur Anmeldung.

### **5.1.2 Registrierung in der App**

Da ein Benutzerkonto für die Nutzung der App notwendig ist, kommt der Nutzer durch das Betätigen des Buttons „Registrierung“ zur Registrierung. Hier kann der



Abbildung 5.1: Splashscreen



Abbildung 5.2: Erste Ansicht

Nutzer nun eine E-Mail Adresse, einen Benutzernamen und ein Passwort eingeben. Das Passwort wird doppelt eingegeben, um einen Fehler möglichst auszuschließen. Durch drücken des Buttons „registrieren“ und bei korrekter Eingabe der Daten wird ein Benutzerkonto erstellt. Eine E-Mail wird durch den Server an die E-Mail Adresse des Nutzers gesendet, welche Informationen zur Aktivierung des Benutzerkontos enthält. Wird der Aktivierungslink in der E-Mail betätigt, wird das Benutzerkonto aktiviert. Falls es die E-Mail oder der Benutzername bereits genutzt werden, oder falls die Passwörter nicht miteinander übereinstimmen, liefert der Server einen Fehler, welcher auf dem Smartphone dargestellt wird. Mithilfe des „Zurück“-Buttons, welcher bei Android Standard ist, wird der Nutzer zurück zum Splashscreen geleitet und die Registrierung wird abgebrochen. In der Abbildung 5.3 ist die Ansicht zu sehen.

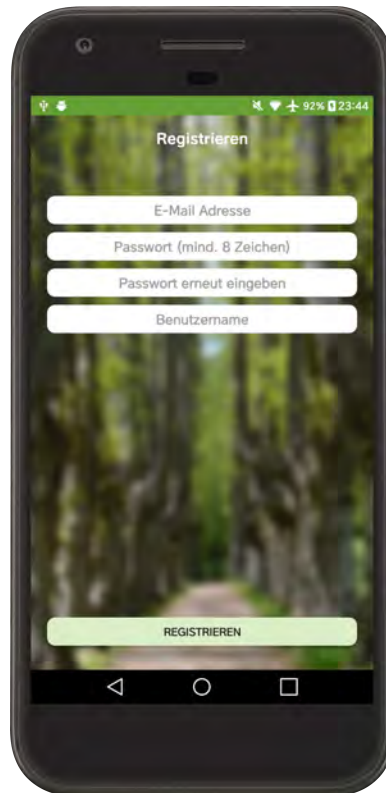


Abbildung 5.3: Registrierung in der App



### 5.1.3 Anmeldung mit Benutzername und Passwort

Nachdem nun das Benutzerkonto erstellt und aktiviert wurde, kann der Nutzer sich nun mithilfe seiner E-Mail und seines Passworts anmelden. Hat sich der Nutzer erfolgreich angemeldet und handelt es sich bei dieser Anmeldung um die erste, so wird der Nutzer dazu aufgefordert, seinen Vor- und Nachnamen anzugeben. Dem Nutzer werden auch statische Fragebögen (falls verfügbar) angezeigt, welche ausgefüllt werden müssen. Diese werden vom Server geladen. Nachdem die Fragebögen erfolgreich ausgefüllt worden sind, wird der Nutzer zum Hauptbereich der App geleitet. Nachdem der Nutzer sich angemeldet hat, werden die Daten des Nutzers in der App für die Dauer einer Session lokal gespeichert. Dies ermöglicht eine einfachere Anmeldung ohne Eingabe von E-Mail und Passwort, nachdem die App geschlossen und wieder geöffnet wurde. Der Nutzer wird, außer bei der Erstanmeldung, immer zum Hauptbereich weitergeleitet. Die Abbildung 5.4 zeigt den Login-Screen.

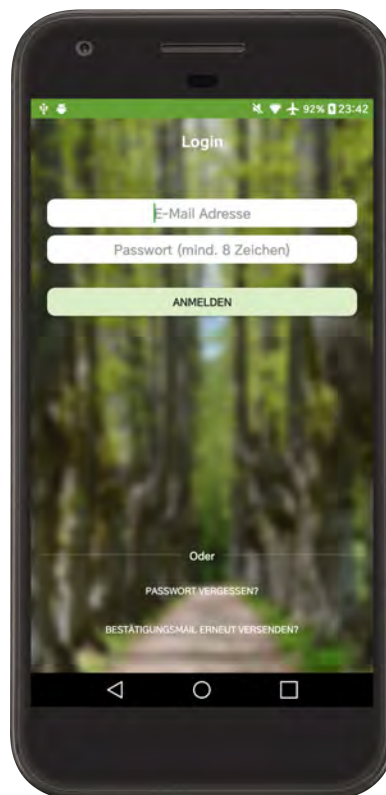


Abbildung 5.4: Login in der App

#### 5.1.4 Passwort vergessen

Hat der Nutzer einmal sein Passwort vergessen, kann das Passwort mithilfe des Buttons „Passwort vergessen“ zurückgesetzt werden. Wenn nun eine E-Mail Adresse angegeben und der „Versenden“-Button gedrückt, wird eine Anfrage mit dieser E-Mail an den Server geschickt. Stimmt nun eine der E-Mail Adressen in der Datenbank mit der eingegebene E-Mail Adresse überein, wird an diese Adresse eine E-Mail geschickt, die einen Link zum Zurücksetzen des Passworts enthält. Bei erfolgreicher Eingabe der Mail oder bei einem Fehler, wird jeweils eine Meldung ausgegeben. Diese Ansicht ist zu sehen in Abbildung 5.5.

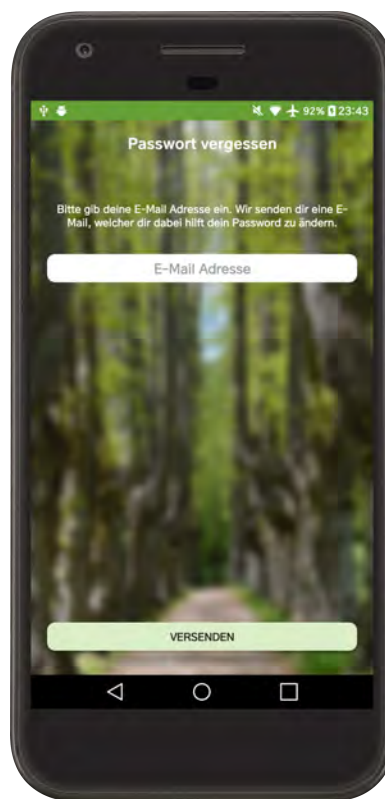


Abbildung 5.5: Passwort vergessen

#### 5.1.5 Aktivierungsmail erneut versenden

Hat der Nutzer keine Aktivierungsmail erhalten, kann eine neue Aktivierungsmail angefordert werden. Durch das Betätigen des Buttons „Bestätigungsmail erneut

senden“ und der Eingabe einer korrekten E-Mail Adresse, wird eine neue Aktivierungsmail an diese Adresse versandt. Dies ist in der Abbildung 5.6.

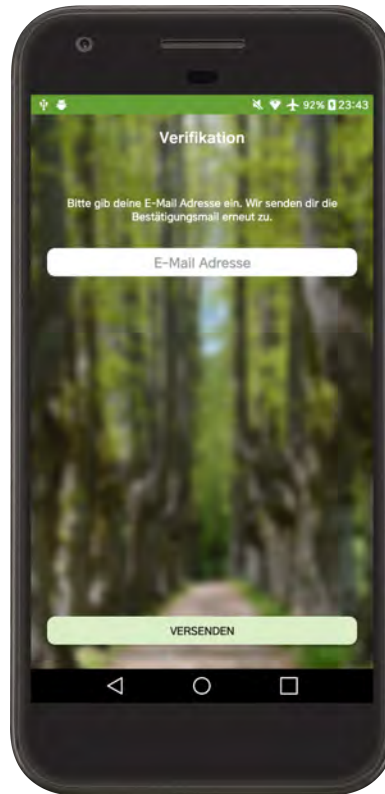


Abbildung 5.6: Verifikation

### 5.1.6 Statische Fragebögen

Handelt es sich bei der Anmeldung des Nutzers um die Erstanmeldung, wird er zur Eingabe von Vor- und Nachnamen aufgefordert. Außerdem werden diesem falls vorhanden statische Fragebögen angezeigt, die ausgefüllt werden sollen. Nachdem diese ausgefüllt wurden, wird der Nutzer mithilfe des „Weiter“-Buttons zum Hauptbereich weitergeleitet. In der Abbildung 5.7 sind mehrere statische Fragebögen in einer Liste zu sehen.



Abbildung 5.7: Statische Fragebögen

### 5.1.7 Menü

Nach der Anmeldung befindet der Nutzer sich im Hauptbereich. Initial wird immer die Startseite angezeigt. Über das Menü, welches sich im oberen linken Rand befindet, kann der Nutzer zu den einzelnen Bereichen gelangen. Ausloggen kann sich der Nutzer über das im Menü befindende Menüpunkt „Logout“. Abbildung 5.8 zeigt das Menü.

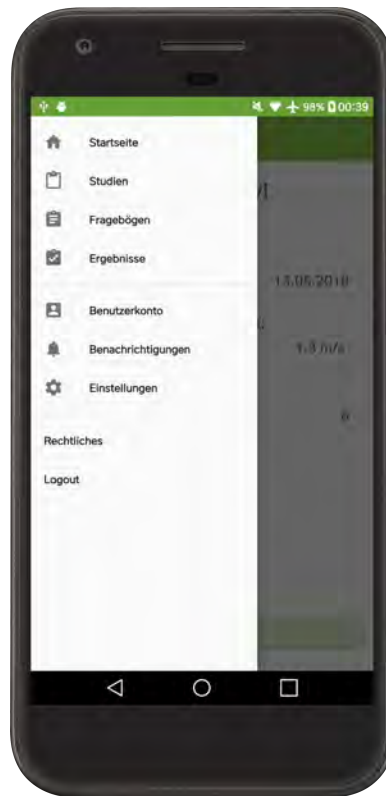


Abbildung 5.8: Drawer

### 5.1.8 Startseite

Die Startseite wird initial angezeigt. Diese enthält Informationen zur letzten Mindful Walking Übung. Außerdem kann durch das Betätigen des Buttons „Beginne Übung“ eine Mindful Walking Übung gestartet werden. Die Ansicht der Startseite ist in Abbildung 5.8 zu sehen.



Abbildung 5.9: Startseite

### 5.1.9 Übung durchführen

Nachdem der Button betätigt wurde. Wird der Nutzer dazu aufgefordert, einen Fragebogen auszufüllen. Wurde der Fragebogen ausgefüllt, wird der Nutzer über den Button „Weiter“ zur nächsten Ansicht weitergeleitet werden. In dieser Ansicht wird die aktuelle Gehgeschwindigkeit der Nutzers innerhalb einer Minute gemessen (Abbildung 5.10). Im nächsten Schritt kann der Nutzer seine Zielgeschwindigkeit auswählen (Abbildung 5.11). Nachdem die gewünschte Option ausgewählt wurde, kann die eigentlich Übung beginnen (Abbildung 5.12). Tippt man die Instruktionen an, erhält der Nutzer eine kurze Instruktion zur korrekten Durchführung einer Mindful Walking Übung. Wird die Übung gestartet, so wird erneut die Geschwindigkeit des Nutzers gemessen. Jedes mal, wenn die Gehgeschwindigkeit größer ist als die Zielgeschwindigkeit, erhält der Nutzer eine akustische Rückmeldung (Vibration, wenn diese in den Einstellungen gewählt wurde). Die Anzahl der Rückmeldungen wird gezählt und später ausgegeben. Nach der Übung werden die Daten an den Server geschickt.



Abbildung 5.10: Aktuelle Geschwindigkeit    Abbildung 5.11: Zielgeschwindigkeit



Abbildung 5.12: Eigentliche Übung



### 5.1.10 Studien

Der Bereich Studien ist in drei Reiter unterteilt. Im ersten Reiter „Meine Studien“ befinden sich die Studien, zu denen der Nutzer sich eingetragen hat, im zweiten Reiter „Alle Studien“ sind alle verfügbaren Studien zu sehen (siehe Abbildung 5.13) und im letzten Reiter „Eingeladene Studien“ befinden sich die Studien, zu denen der Nutzer eingeladen wurde. Über einen Klick auf den Titel der Studie, wird dieser Bereich erweitert und eine Beschreibung dieser Studie wird aufgezeigt. Um dieser Studie dann beizutreten, muss der Nutzer auf den Pfeil drücken und ein Dialogfenster wird geöffnet. Hier kann der Nutzer auswählen ob er sich zu der Studie eintragen möchte oder nicht. Dafür muss sich der Nutzer jedoch im Reiter „Alle Studien“ oder im reiter „Eingeladene Studien „befinden. Das Austreten einer Studie funktioniert analog und nur im Reiter „Meine Studien“. Bei erfolgreichem oder nicht erfolgreichem Ein- und Austragen einer Studie, wird dem Nutzer eine Meldung angezeigt.

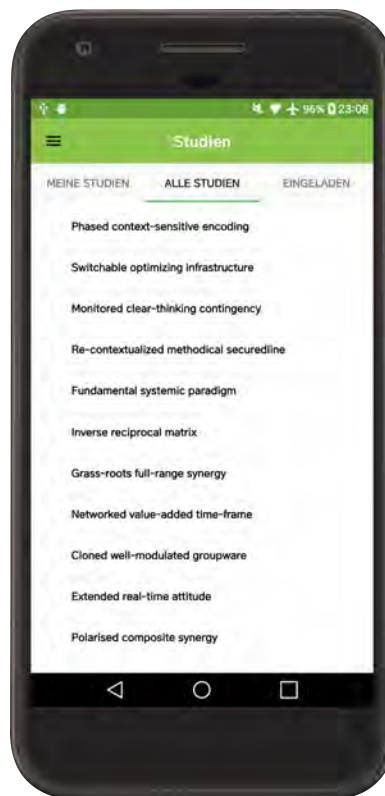


Abbildung 5.13: Alle Studien Tab

### 5.1.11 Fragebögen

Sollte die aktuelle Studie „Mindful Walking“ über Fragebögen verfügen, werden diese im Bereich Fragebögen angezeigt. Unterteilt werden diese auf zwei Reiter (Abbildung 5.14). Zum einen die normalen Fragebögen, die auch öfters beantwortet werden können und zum anderen die Einmal-Fragebögen. Diese können nur ein einziges Mal ausgefüllt werden. Sollte der Nutzer sich zu weiteren Studien eintragen, werden auch die Fragebögen dieser Studie im Bereich Fragebögen angezeigt. Durch das Antippen eines Fragebogens, wird der Nutzer zu diesem weitergeleitet. Es befindet sich immer nur eine Frage und die zugehörigen Antwortmöglichkeiten zur selben Zeit auf dem Bildschirm (siehe Abbildung 5.15). Durch das Drücken des „Weiter“-Buttons gelangt der Nutzer zur nächsten Frage. Nachdem ein Fragebogen ausgefüllt wurde, werden die Antworten an den Server geschickt und der Nutzer wird zurück in den Bereich Fragebögen geleitet.

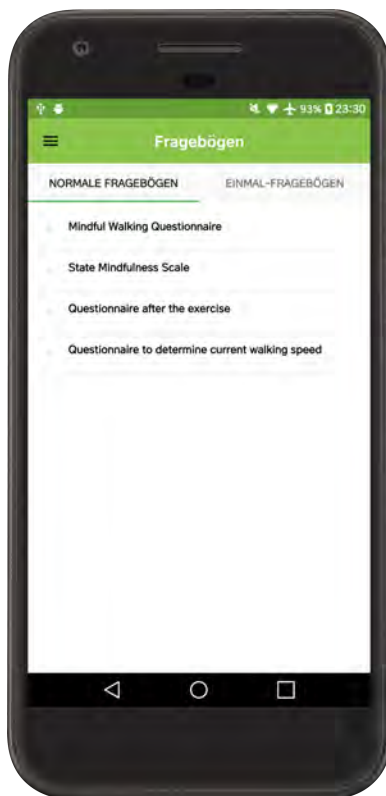


Abbildung 5.14: Alle Frageboegen

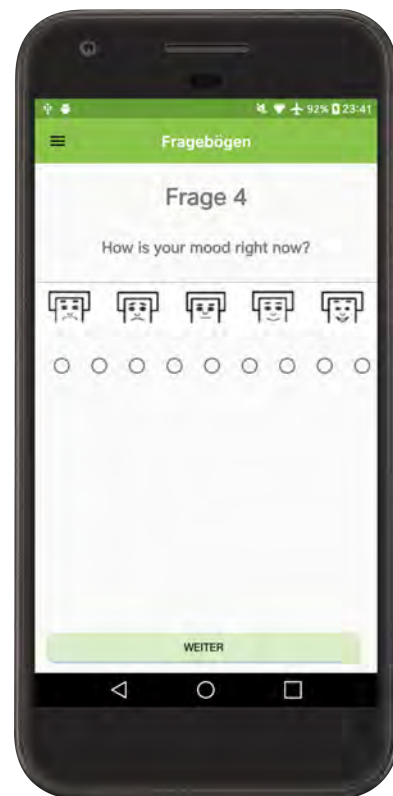


Abbildung 5.15: Frage

### 5.1.12 Ergebnisse

Dem Nutzer werden im Bereich Ergebnisse, die Ergebnisse seiner Mindful Walking Übungen angezeigt. Die Übungen sind anhand ihres Datums gelistet. Wird nun auf eine der Übungen gedrückt, so kommt der Nutzer zu einer Detailansicht (Abbildung 5.16). In dieser werden dem Nutzer die Durchschnittsgeschwindigkeit, die Anzahl der Geschwindigkeitsüberschreitungen und das Start- und Enddatum angezeigt.



Abbildung 5.16: Ergebnisse

### 5.1.13 Benutzerkonto

Im Bereich Benutzerkonto wird die E-Mail Adresse, der Vorname, der Nachname und der Benutzername des Nutzers angezeigt (Abbildung 5.17). Durch einen Klick auf den „Passwort ändern“ Button, kann der Nutzer sein Passwort ändern (Abbildung 5.18). Möchte der Nutzer sein Konto löschen, so kann er dies über den Button



Abbildung 5.17: Benutzerkonto

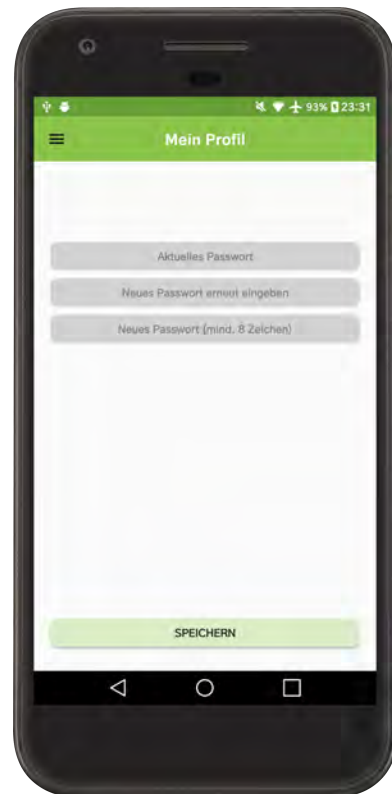


Abbildung 5.18: Passwort ändern

„Account löschen“ erreichen. Bevor das Konto gelöscht wird, wird der Nutzer gefragt ob er dies auch wirklich möchte.

### 5.1.14 Benachrichtigungen

In diesem Bereich kann der Nutzer einstellen, wann und wie oft er Benachrichtigungen zum Durchführen von seiner Mindful Walking Übung erhalten soll (Abbildung 5.19).

### 5.1.15 Einstellungen

In den Einstellungen kann der Nutzer die Sprache ändern. Die verfügbaren Sprachen sind zum Zeitpunkt dieser Arbeit Englisch und Deutsch. Außerdem kann der Nutzer die Soundeinstellungen ändern. Soll während der Mindful Walking Übung

## 5 Vorstellung des Mindful Walking Rahmenwerks

---



Abbildung 5.19: Benachrichtigungen



Abbildung 5.20: Einstellungen

ein Ton ausgegeben werden oder doch nur Vibrationen, so kann dies hier geändert werden (Abbildung 5.20).

# 6 Implementierung

## 6.1 Volley Request

Da zu Beginn der Arbeit erwähnt wurde, dass Volley für die Anfragen an die API genutzt wurde, wird hier nun ein solcher Request anhand eines Codes beschrieben. Bevor jedoch ein solche Request gesendet werden kann, muss zuerst eine Requestqueue erstellt werden. Beim folgenden Volley-Request handelt es sich um die Registrierung. Ein JSONObject mit den Daten des Nutzers wird erstellt (Zeilen 6-23). Dieses JSON-Objekt wird in einen JSONObjectRequest gepackt (Zeilen 29/30) und anschließend der Requestqueue hinzugefügt (Zeilen 70/71). Dieser Request wird durch den Cache-Thread auf gegriffen und vorselektiert. Kann der Request durch den Cache erfüllt werden, so wird die im Cache gespeicherte Antwort zerlegt und umgewandelt (parsed) und dem Main-Thread geliefert. Ansonsten wird der Request der NetworkQueue hinzugefügt. Der erste verfügbare Network-Thread übernimmt diesen Request und führt die Transaktion aus, wandelt die Response um, schreibt diese in den Cache und liefert diese dann dem Main-Thread. Ist eine Response angekommen, so kann diese verarbeitet werden (Zeilen 33-39). Bei einem Fehler, werden dem Nutzer Fehlermeldungen angezeigt (Zeilen 40-70).

```
1
2 public void register() {
3
4     String url = "https://www.api1234.com/api/register";
5
6     JSONObject jsonreg = new JSONObject();
7     JSONObject jsonreg1 = new JSONObject();
8     JSONObject jsonreg2 = new JSONObject();
9     JSONObject jsonreg3 = new JSONObject();
10
11
12     try {
```

```
13
14     jsonreg.put("data", jsonreg1);
15     jsonreg1.put("type", "users");
16     jsonreg1.put("attributes", jsonreg2);
17     jsonreg2.put("email", email);
18     jsonreg2.put("password", password1);
19     jsonreg2.put("password_confirmation", password2);
20     jsonreg2.put("name", username);
21     jsonreg2.put("settings", jsonreg3);
22     jsonreg3.put("firsttime", "true");
23
24 } catch (JSONException e) {
25     e.printStackTrace();
26 }
27
28 JsonObjectRequestWithNull registerreq;
29 registerreq = new JsonObjectRequestWithNull(Request.Method.POST,
30     url, jsonreg, new Response.Listener<JSONObject>() {
31
32     @Override
33     public void onResponse(JSONObject response) {
34         Toast.makeText(Register.this,
35             getString(R.string.account_created),
36             Toast.LENGTH_LONG).show();
37         startActivity(new Intent(Register.this, Login.class));
38         finish();
39     }
40 }, new Response.ErrorListener() {
41     @Override
42     public void onErrorResponse(VolleyError error) {
43
44         NetworkResponse networkResponse = error.networkResponse;
45
46         if (networkResponse != null) {
47
48             switch (Integer.valueOf(networkResponse.statusCode)) {
49
50                 case 409:
```

```
51     Toast.makeText(Register.this,
52                   getString(R.string.email_used),
53                   Toast.LENGTH_LONG).show();
54         break;
55
56     case 422:
57         Toast.makeText(Register.this,
58                   getString(R.string.password_notequal),
59                   Toast.LENGTH_LONG).show();
60         break;
61
62     default:
63         Toast.makeText(Register.this,
64                   getString(R.string.something_wrong),
65                   Toast.LENGTH_LONG).show();
66         }
67     }
68 }
69 });
70 GlobalRequestQueue.getInstance(getApplicationContext())
71     .addToRequestQueue(registerrequest);
72 }
```

Listing 6.1: JSONObjectRequest

## 6.2 Google Play Services Location API und die Geschwindigkeitsmessung

Für die Geschwindigkeitsbestimmung wurde die Google Play Services Location API genutzt. Damit der GPS-Chip des Smartphones genutzt werden kann, muss die App durch das System eine Genehmigung (permissions) erhalten. Diese permissions befinden sich in der Android-Manifest.xml-Datei. Danach muss ein Location Service Client erstellt werden. Wurde dieser erstellt, kann die letzte, bekannte Position des Geräts ermittelt werden. Bevor die Positionsupdates erhalten werden können, muss ein LocationRequest erstellt und dessen Parameter gesetzt werden. Um nun die Updates zu erhalten, ruft unser Client eine Methode auf. Dieser werden



dann eine Instanz des LocationRequest Objekts und ein LocationCallback übergeben. Dadurch wird eine Methode des LocationCallbacks aufgerufen, die somit eine Liste von Position zurückgibt. Nun kann mithilfe dieser Positionen die Geschwindigkeit berechnet werden.

```
1
2 mLocationCallback = new LocationCallback() {
3     @Override
4     public void onLocationResult(LocationResult locationResult) {
5         super.onLocationResult(locationResult);
6
7         for (Location location :
8             locationResult.getLocations()) {
9             if (location != null) {
10                if (location.hasSpeed()) {
11                    txtCurrentSpeed.
12                        setText(String.valueOf(location.getSpeed())
13                            + "m/s");
14                    speed = location.getSpeed();
15                    averagespeed = (averagespeed + speed) / i;
16                    txtAverageSpeed.setText(averagespeed + "m/s");
17
18                    if(speed>maxspeed){
19                        maxspeed = speed;
20                    }
21                    i++;
22                } else {
23                    txtCurrentSpeed.
24                        setText(getString(R.string.no_speed));
25                    txtAverageSpeed.
26                        setText(getString(R.string.no_speed));
27                }
28            }
29        }
30    }
31};
```

Listing 6.2: LocationCallback mit UI-Update.

# 7 Anforderungsabgleich

Nun werden die Anforderung an die App, die in Kapitel 3 aufgeführt wurden, mit der eigentlich App abgeglichen. Wie schon in den Anforderung aus Kapitel 3, werden hier zunächst die funktionalen und anschließend die nicht-funktionalen Anforderungen betrachtet.

## 7.1 Funktionale Anforderungen

In der folgenden Tabelle werden die funktionalen Anforderung an die App überprüft. Hierbei wird überprüft ob die Implementierung alle funktionalen Anforderung erfüllt hat.

<b>Nr.</b>	<b>Beschreibung</b>	<b>Problembeschreibung</b>
1.	Registrierung in der App	Anforderung erfüllt (siehe Kapitel 5.1.2)
2.	An Studien teilnehmen	Anforderung erfüllt (siehe Kapitel 5.1.10)
3.	Studieneinladungen annehmen	Anforderung erfüllt (siehe Kapitel 5.1.10)
4.	An einer privaten Studie teilnehmen	Anforderung erfüllt (siehe Kapitel 5.1.10)
5.	Fragenbögen einer Studie ausfüllen	Anforderung erfüllt (siehe Kapitel 5.1.11)
6.	Ergebnisse anzeigen	Anforderung erfüllt (siehe Kapitel 5.1.12)
7.	Ergebnisse an den Server schicken	Anforderung erfüllt (siehe Kapitel 5.1.9)
8.	Meldung von Benachrichtigungen	Anforderung erfüllt (siehe Kapitel 5.1.14)
9.	Beginn der Übung	Anforderung erfüllt (siehe Kapitel 5.1.9)

10.	Übung durchführen	Anforderung erfüllt (siehe Kapitel 5.1.9)
11.	Vibration	Anforderung erfüllt (siehe Kapitel 5.1.15)

Tabelle 7.2: Abgleich funktionaler Anforderungen

## 7.2 Nicht-Funktionale Anforderungen

Anschließend werden die nicht-funktionalen Anforderungen betrachtet und abgeglichen.

<b>Nr.</b>	<b>Beschreibung</b>	<b>Problembeschreibung</b>
1.	Technische Anforderungen	Anforderung erfüllt. App ist optimiert für Smartphone mit neueren Android Versionen.
2.	Stabilität und Robustheit	Anforderung erfüllt.

Tabelle 7.3: Abgleich nicht-funktionaler Anforderungen

# 8 Zusammenfassung und Ausblick

## 8.1 Zusammenfassung

Mit dieser Arbeit ist nun die erste Phase der Entwicklung des Mindful Walking Projekts beendet. Im Rahmen dieser Bachelorarbeit ist eine mobile Anwendung zur Unterstützung von Mindful Walking Übungen entstanden. Die App wurde für Smartphones entwickelt, die Android als Betriebssystem verwenden. Die Anforderungen, welche in Kapitel 3 zu finden sind, wurden an mehreren Treffen definiert. Die Anforderungen wurden bestmöglich berücksichtigt und realisiert. Neue Anforderungen und Ideen wurden ebenfalls analysiert und gegebenenfalls umgesetzt. Die erfüllten Anforderungen sind im Kapitel 7 zu sehen. Zu den wichtigsten Anforderungen gehören, die Registrierung und die Anmeldung auf einem Server, wodurch sich die Daten des Nutzers besser verwalten lassen, sowie die Durchführung einer Mindful Walking Übung. Zu Beginn einer Mindful Walking Übung werden Fragebögen auf dem Smartphone dargestellt, diese sollen ausgefüllt werden. Die Geschwindigkeit wird durch die Position mithilfe des GPS erfasst. Die Ergebnisse der Fragebögen und der Übung werden an den Server geschickt. Dadurch können die Ergebnisse im Nachhinein eingesehen und evaluiert werden.

## 8.2 Ausblick

Dieser Abschnitt befasst sich mit den den möglichen Verbesserungen und Ideen, welche bei der Realisierung der App aufgekommen sind. Dazu gehört ein Offline-Modus, welcher die App auch ohne Internetverbindungen nutzbar macht. Verbesserungen am User-Interface, die dem User eine intuitivere Bedienung ermöglichen könnten. Auch könnten Alternativen zu GPS implementiert werden, die das Messen der Geschwindigkeit verbessern könnten. Bei der Netzwerkkommunikation könnten Alternativen zu Volley zu gesucht werden.

### **8.2.1 Offline-Modus**

Die Applikation besitzt zum Zeitpunkt der Arbeit keinen Offline-Modus und benötigt zwingend eine Internetverbindung. Dies könnte in Gebieten, die eine schwache Netzabdeckung besitzen, seien es LTE oder UMTS, zu Problemen in der Anwendungen führen. Um dem entgegenzuwirken könnte ein Offline-Modus implementiert werden. Durch diesen könnten die Fragebögen heruntergeladen und lokal abgespeichert werden. Auch die Ergebnisse der Übungen könnten abgespeichert werden. Bei verfügbarem Internetzugang, könnten diese dann automatische hochgeladen werden.

### **8.2.2 Verbesserung am User-Interface**

Im Rahmen dieser Bachelorarbeit wurden die rein funktionalen Anforderungen in den Vordergrund gestellt. Das User-Interface ist somit nicht ganz ausgereift und könnte überarbeitet werden. Das Wesentliche wäre hierbei eine Verbesserung der Bedienung und Interaktion des Nutzers mit der App.

### **8.2.3 GPS-Alternativen**

Zur Geschwindigkeitsmessung während einer Übung wird zum Zeitpunkt der Arbeit GPS verwendet. Sollte der Nutzer aber keine Sicht zum Satelliten haben, sei es in Gebäuden oder in der Nähe von hohen Gebäuden, könnte eine GPS Verbindung nicht verfügbar sein. Eine Messung wäre in diesem Fall nicht möglich. Um dem entgegenzuwirken könnte AGPS verwendet werden. AGPS oder auch Assisted GPS nutzt sowohl den GPS-Chip im Smartphone als auch den Mobilfunkanbieter um die Position zu lokalisieren. Dieser schickt dem Empfänger Hilfsdaten um die Positionsbestimmung schneller und genauer durchzuführen. Dies funktioniert jedoch nur wenn der Mobilfunkanbieter das zulässt. Ein weiterer Vorteil wäre der geringere Stromverbrauch im Vergleich zum reinen GPS.

### **8.2.4 Netzwerkkommunikation**

In diesem Projekt wurde Volley zur Netzwerkkommunikation verwendet. Jedoch gibt es auch Alternativen, wie z.B. Retrofit [8]. Retrofit's Code eignet sich für REST-API.

Retrofit besitzt eine sehr gute API Dokumentation und die Performance von Retrofit ist besser im Vergleich zu Volley.

# Tabellenverzeichnis

3.1 Funktionale Anforderungen . . . . .	7
3.2 Nicht-funktionale Anforderungen . . . . .	8
7.2 Abgleich funktionaler Anforderungen . . . . .	37
7.3 Abgleich nicht-funktionaler Anforderungen . . . . .	37

# Abbildungsverzeichnis

2.1	Volley Request . . . . .	3
2.2	Aufbau JSON-Objekt (oben) und JSON-Array (unten) . . . . .	5
4.1	Übersicht der Architektur . . . . .	10
4.2	Genereller Ablauf . . . . .	12
4.3	Fragebögen anzeigen und beantworten . . . . .	13
4.4	Übung durchführen . . . . .	14
4.5	Model-View-Controller . . . . .	15
5.1	Splashscreen . . . . .	17
5.2	Erste Ansicht . . . . .	17
5.3	Registrierung in der App . . . . .	18
5.4	Login in der App . . . . .	19
5.5	Passwort vergessen . . . . .	20
5.6	Verifikation . . . . .	21
5.7	Statische Fragebögen . . . . .	22
5.8	Drawer . . . . .	23
5.9	Startseite . . . . .	24
5.10	Aktuelle Geschwindigkeit . . . . .	25
5.11	Zielgeschwindigkeit . . . . .	25
5.12	Eigentliche Übung . . . . .	26
5.13	Alle Studien Tab . . . . .	27
5.14	Alle Frageboegen . . . . .	28
5.15	Frage . . . . .	28
5.16	Ergebnisse . . . . .	29
5.17	Benutzerkonto . . . . .	30
5.18	Passwort ändern . . . . .	30
5.19	Benachrichtigungen . . . . .	31
5.20	Einstellungen . . . . .	31



# Literaturverzeichnis

- [1] KABAT-ZINN, Jon: *Full catastrophe living : using the wisdom of your body and mind to face stress, pain, and illness*. Stress Reduction Clinic at the University of Massachusetts Medical Center : delta, 1990. – ISBN 9780385303125, 0385303122
- [2] KHOURY, B ; SHARMA, M ; RUSH, S. E. ; FOURNIER, C: Mindfulness-based stress reduction for healthy individuals: A meta-analysis. In: *Journal of Psychosomatic Research* 78, 2015, S. 519–528
- [3] SEGAL, Z. V. ; WILLIAMS, J. M. G. ; TEASDALE, J. D.: *Mindfulness-based cognitive therapy for depression – A new approach to preventing relapse*. New York : Guilford Press, 2002. – ISBN 9781462507504, 1462507506
- [4] PRYSS, Rüdiger ; REICHERT, Manfred ; HERRMANN, Jochen ; LANGGUTH, Berthold ; SCHLEE, Winfried: Mobile Crowd Sensing in Clinical and Psychological Trials—A Case Study. In: *Computer-Based Medical Systems (CBMS), 2015 IEEE 28th International Symposium IEEE*, 2015, S. 23–24
- [5] *Ecma International, Transmitting Network Data Using Volley*. <http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf>, . – Zugriff: 2018-01-12
- [6] *Android, The JSON Data Interchange Syntax*. <https://developer.android.com/training/volley/index.html>, . – Zugriff: 2018-04-20
- [7] *Android, Location and context overview*. <https://developer.android.com/training/location/>, . – Zugriff: 2018-01-12
- [8] *Square, Retrofit*. <http://square.github.io/retrofit/>, . – Zugriff: 2018-05-03

Name: Dennis Kozlowski

Matrikelnummer: 791052

**Erklärung**

Ich erkläre, dass ich die Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Ulm, den .....

Dennis Kozlowski