Bachelor MKI

“Mobile Computing”

SS 2019

- Projektdokumentation -

Fitness Applikation – „Stay Fit“

Vorgelegt von: Kontaktadresse:

Philipp Gerhard [philipp.gerhard@student.reutlingen-university.de](mailto:philipp.gerhard@student.reutlingen-university.de)

Marvin Holm [marvin.holm@student.reutlingen-university.de](mailto:marvin.holm@student.reutlingen-university.de)

Daniel Schott [daniel.schott@student.reutlingen-university.de](mailto:daniel.schott@student.reutlingen-university.de)

Arina Markova [arina.markova@student.reutlingen-university.de](mailto:arina.markova@student.reutlingen-university.de)

Tim Mangliers [tim.mangliers@student.reutlingen-university.de](mailto:tim.mangliers@student.reutlingen-university.de)

Christian Michels [christian\_alexander.michels@student.reutlingen-university.de](mailto:christian_alexander.michels@student.reutlingen-university.de)

Eingereicht bei: M.Sc. Daniel Stiefel [daniel.stiefel@reutlingen-university.de](mailto:daniel.stiefel@reutlingen-university.de)

Eingereicht am: 26.06.2019

**Zusammenfassung**: Das Ziel der Fitness Applikation „Stay Fit“ ist es Trainierende beim fit werden zu unterstützen und durch ihre Praktikabilität im Trainingsalltag zu mehr Training zu motivieren. Die Applikation ermöglicht das Erstellen eigener, an individuelle Trainingspraktiken angepasste Trainingspläne. Beim Durchführen eines vorgefertigten oder selbst erstellten Trainingsplans werden Daten wie z.B. die Ausführungsdauer der ausgeführten Übungen gespeichert um eine umfassende Statistik des Trainierenden zu erstellen. Mithilfe der in der Statistik bereitgestellten Informationen kann das Training des einzelnen noch effektiver und effizienter gestaltet werden. Diese und weitere Features machen die Fitness Applikation „Stay Fit“ zu einem unverzichtbaren Tool das im Trainingsalltag nicht mehr wegzudenken ist.

Inhaltsverzeichnis

1. Einführung 1

1.1 Motivation 1

1.2 Ziele 1

1.3 Zielgruppe 1

2. Anforderungsanalyse 1

2.1 Must-Have 1

2.2 Should-Have 1

2.3 Nice-to-Have 1

2.4 Anforderungen Überblick 1

3. Entwurf 1

3.1 Mockups 1

3.2 ER-Diagramm 1

3.3 Aktivitätsdiagramm 1

3.4 Szenario 1

4. Ergebnis 1

4.1 Login/Registrierung 11

4.2 Traininspläne(Excesises) 11

4.3 Statistik 11

4.4 Stoppuhren 11

4.5 „…..“ 11

6. Links 11

6. Evaluierung 11

7. Declaration on oath 11

# Einführung

Die Ziele des Moduls Mobile Computing sind

* Software engineering im Bereich Mobiles App Development erlernen
* Die Grundlegende Fertigkeit erwerben für die Android Plattform zu entwickeln
* In einem Team eine dedizierte Applikation umsetzen

Zum Erfüllen der Lernziele des Moduls hat die Gruppe die an der Umsetzung der zum vorliegenden Dokument gehörenden Applikation beteiligt war eine Fitness App umgesetzt.

Bei der Wahl des Projektes hatten die Gruppe freie Wahl. Nach einem gemeinsamen Mindstorming standen zwei gegensätzliche Ideen zur Auswahl

* Eine Kalender Applikation
* Eine Fitness Applikation

Diese anfänglichen Ideen sollten an einem speziell dafür bereitgestellten Termins den anderen Teams in Form einer kurzen Präsentation vorgestellt werden. In einer Anschließenden Diskussion mit den anderen Teams sollte dann eine Idee ausscheiden. Die Idee der Kalender Applikation schied schlussendlich aus. Das Ausscheiden der Kalender Applikation hatte den Grund des fehlenden Mehrwertes. Ein Mehrwert könnte durch eine längere Entwicklungszeit entstehen, jedoch stände uns diese in einem Semester nicht zur Verfügung. Mit der uns zur Verfügung stehenden Zeit könnten wir eher eine Sinnvolle Applikation im Bereich Fitness, als im Kalender Bereich. Dies ist der inhärenten Funktionalität der zwei Bereiche geschuldet. Beispielsweise wäre das Eintragen von Terminen in einen Kalender als umgesetztes Key-Feature weniger spektakulär als das Abarbeiten von bestimmten Abläufen einer Trainingseinheit. Beide Features hätten aber ungefähr den gleichen Zeitaufwand bezüglich ihrer Implementierung. Da es zum Schluss jedoch durchaus schön wäre, wenn nicht nur Wissen im Programmieren einer Android Applikation erworben wird sondern auch Funktionalität besteht die vorzeigbar ist, haben wir die Vorschläge der anderen Teams angenommen und uns für das Umsetzen der Fitness Applikation entschieden.

**Danksagung**

Wir möchten uns bei Herr Daniel Stiefel für die Bereitstellung des Theoretischen Wissens durch die Vorlesung, für das verkürzen der Einarbeitungsphase in Android Studio durch die Entsprechenden Code Beispiele die ebenfalls während der Vorlesung behandelt wurden, sowie für die allzeit gegebenen nützlichen Tipps und Tricks bedanken.

## Motivation

## Warum sollte überhaupt eine neue Applikation die das Training unterstützt umgesetzt werden? Eine Anfrage im Google Play Store ergibt immerhin eine satte Anzahl von rund 250 Fitness Applikationen (Siehe Abbildung 1). Diese Applikationen sind jedoch für Sportprofis oder für absolute Anfänger konzipiert. Meist sind sie außerdem mit sehr teuren In-App-Käufen verbunden. Zusätzlich lässt die Bedienbarkeit der Applikationen beim aktiven Sportmachen zu wünschen übrig. Diese Probleme wurden von einigen Teammitgliedern persönlich erlebt. So wurden von mehreren Teammitgliedern bereits verschiedene Fitness Applikationen ausprobiert. Jedoch war keine dazu in der Lage den Nutzer länger zu binden. In den meisten Fällen scheiterte die Nutzbarkeit der Applikation an der Bedienung während des Trainings. Viele Benutzer wechseln deshalb auf Lösungen im Bereich der Smart Watches. Unsere Fitness Applikation setzt genau bei diesen Problemen an. Konzeptuell ist die Applikation für alle Sporttreibenden gedacht. Geplant ist eine kostenlose, intuitiv während dem aktiven Sport zu bedienende Applikation, die endlich das Sporttreiben einfacher machen soll.

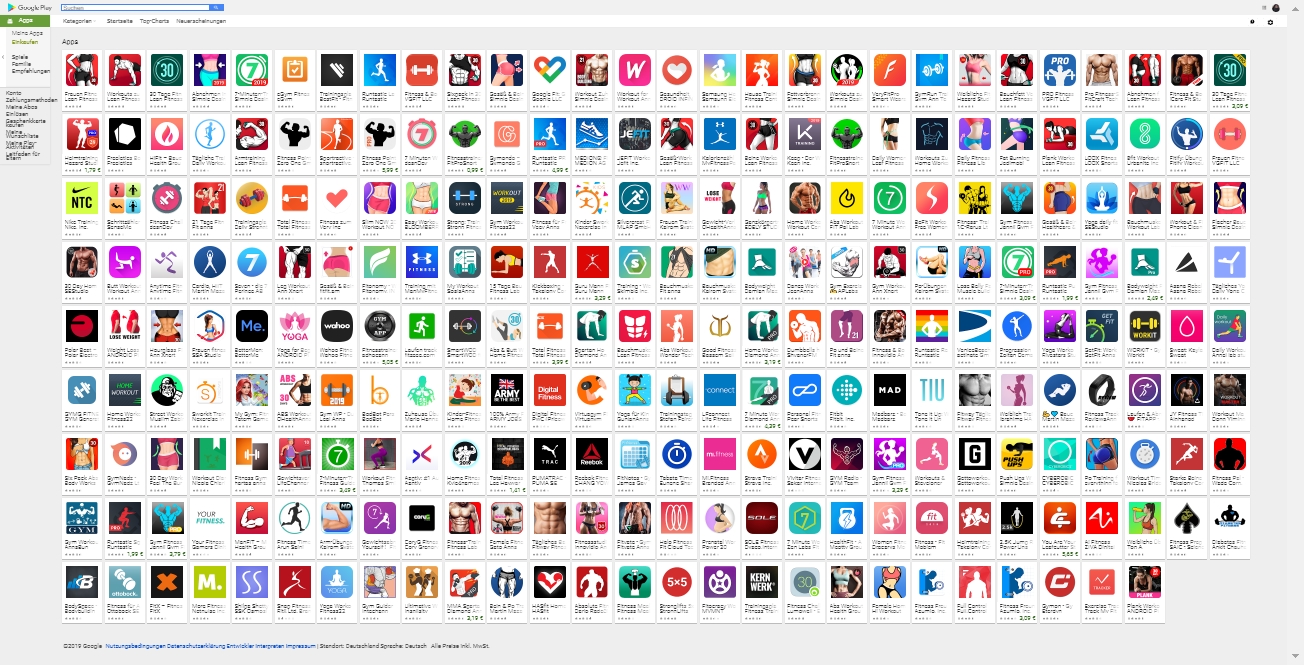


Abbildung 1: Fitness Applikationen im Google Play Store

## Ziele

Das Ziel ist es eine benutzerfreundliche, auf die Bedürfnisse von Trainierenden aller Art zugeschnittene Applikation zu entwerfen und zu implementieren. Die Fitness Applikation „Stay Fit“ soll das alltägliche Training unterstützen und durch dadurch verbesserte Trainingserfolge dazu motivieren kontinuierlich das eigene spezifische Training zu verfolgen. Die Applikation soll am Schluss der letzten Iteration als .apk Datei vorliegen um einfach auf den verschiedensten Android Geräten installiert werden zu können.

# Anforderungen

## Ideenfindung

## Zielgruppe

Die Zielgruppen der Applikation sind sehr breit gefächert. Durch die intuitive Bedienbarkeit sowie der ansprechenden Oberfläche wird eine Benutzbarkeit auch von Benutzern mit geringer Technikaffinität gewährleistet. Ist es den Benutzern möglich ein Handy zu bedienen wird es ihnen auch möglich sein die Fitness Applikation „Stay Fit“ zu verwenden. Jeder der gewillt ist Sport zu treiben und sich fit zu halten ist ein potenzieller Benutzer der Applikation. Ob Neueinsteiger, Freizeit- oder Profisportler, mit der Fitness Applikation „Stay Fit“ wird das Training mit Garantie nicht nur Effektiver und Effizienter sondern auch spaßiger als das Training ohne.

Die im Planungsprozess erstellten Personas sind im Folgenden dargestellt.



Abbildung 1: Zielgruppe Persona 1

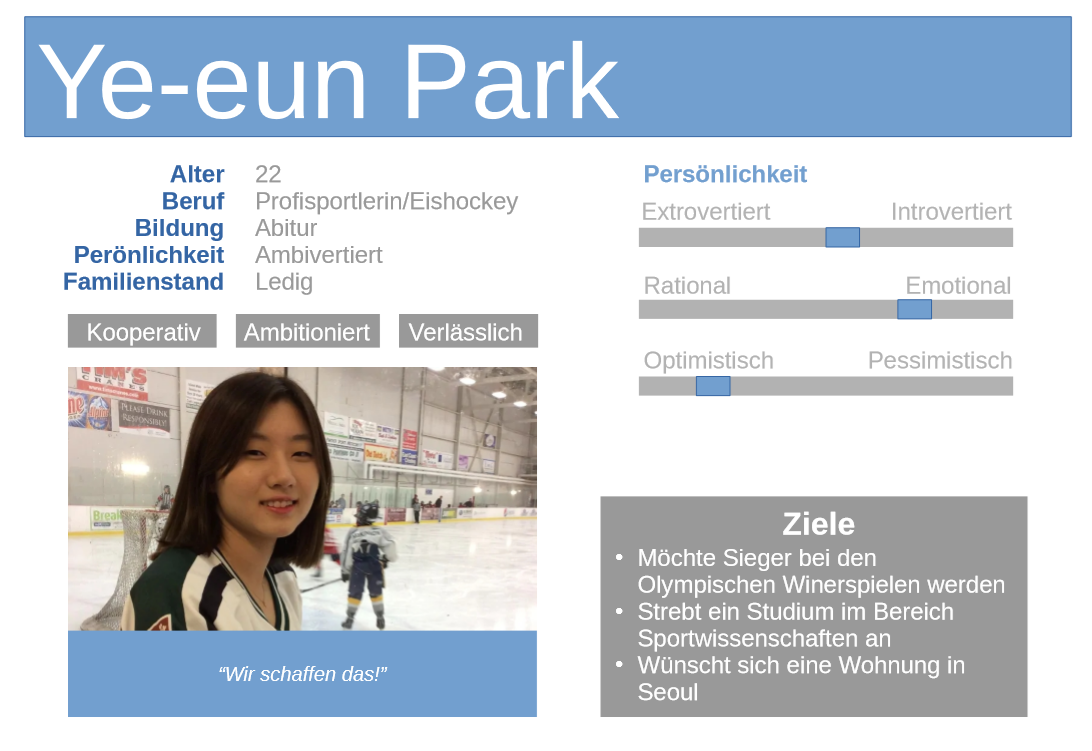


Abbildung 2: Zielgruppe Persona 2



Abbildung 3: Zielgruppe Persona 3

## Anforderungsanalyse

Die Anforderungen sind in „Must-Have“, „Should-Have“ und „Nice-to-Have“ Anforderungen unterteilt. Hierbei wird weiters zwischen funktionalen- („F“) und nichtfunktionalen- („NF“) Anforderungen unterschieden.

### Must-Have Anforderungen

|  |  |  |  |
| --- | --- | --- | --- |
| Nr. | Typ | Anforderung | Beschreibung |
|  | F | Registrierung | Falls ein Benutzer kein Account besitzt, registriert er sich einmalig mit seinem Namen und einem Passwort. |
|  | F | Login | Falls der Benutzer einen Account bereits besitzt und in der App noch nicht angemeldet ist, meldet er sich beim Start der App mit seinem Namen und seinem Passwort an. |
|  | F | Eigenen Trainingsplan erstellen | Es muss möglich sein einen, aus beliebig vielen Trainingseinheiten bestehenden Trainingsplan zu erstellen. |
|  | F | Laden von Trainingsplänen aus der Datenbank | Die Datenbank muss eine Reihe von bekannten und vorbereiteten Trainingsplänen anbieten. |
|  | F | Einträge der Leistungen | Alle erreichten Leistungen müssen in die Trainingseinheit eingetragen werden, welche aus den folgenden Punkten besteht:  -Gewicht  -Anzahl Wiederholungen  -Anzahl Sätze  -Dauer |
|  | NF | Schneller Zugriff | Die Applikation sollte in der Praxis einen schnellen Zugriff gewährleisten. |
|  | NF | Einfache Bedienung | Die Applikation soll einfach und intuitiv Bedienbar sein. |
|  | NF | Sicherheit der Daten | Aktuelle Datenschutzrichtlinien sollen eingehalten werden. |
|  | NF | Ansprechendes Design | Die Applikation soll ein ansprechendes Design besitzen. |

### Should-Have Anforderungen

|  |  |  |  |
| --- | --- | --- | --- |
| Nr. | Typ | Anforderungen | Beschreibung |
|  | F | Statistik | Statistische Auswertung der erbrachten Leistungen über Zeit. |
|  | F | Kalorienverbrauch | Eintragen des Kalorienverbrauchs für die Trainingseinheit. |
|  | F | Erfolg analysieren | Erstellen eines Datensatzes für den Erfolg des Trainings nach den folgenden Kriterien:  - Datum  - Gewicht  - Körperfett  - Muskelanteil. |
|  | NF | Geringer Akkuverbrauch | Die Applikation soll wenig Akku verbrauchen um ein breites Spektrum an Android Geräten anzusprechen. |

### Nice-to-Have Anforderungen

|  |  |  |  |
| --- | --- | --- | --- |
| Nr. | Typ | Anforderungen | Beschreibung |
|  | F | Ernährungsplan | Benutzer kann den eigenen Kalorienverbrauch analysieren, indem er seine tägliche Ernährung einträgt. |
|  | F | Statische Auswertung zur Ernährung | Die Ernährungsweiße wird analysiert. |
|  | F | Vergleiche Ernährung mit dem Training | Die Ernährung kann auf das Training angepasst werden. |
|  | F | Hinweis auf Defizite in der Ernährung | Defizite in der Ernährung sollen erkannt werden. Basierend darauf sollen dem Nutzer Verbesserungen vorgeschlagen werden. |
|  | F | Erstellen von Meilensteinen | Meilensteine sollen den Benutzer Motivieren. |
|  | F | Verbindung zu Spotify um Musik hören zu können | Dies berücksichtigt ebenfalls den motivierenden Aspekt der Applikation. |
|  | F | Passwort wiederherstellen | Benutzer kann sein Passwort wiederherstellen, falls er es vergessen hat. |

## Anforderungen im Überblick

In Abbildung q ist die Verteilung der anfänglich gestellten Anforderungen dargestellt. Deutlich ist, dass der größte Anteil an Anforderungen Must-Have Anforderungen sind. Diese Verteilung änderte sich jedoch in weiteren Iterationsschritten.

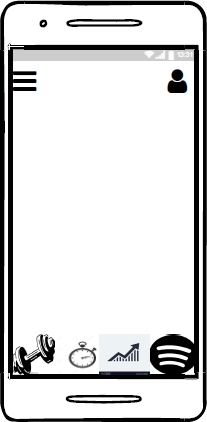
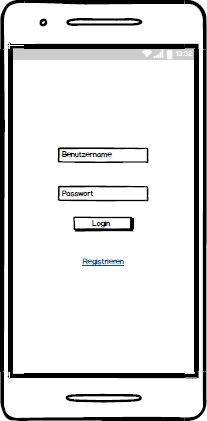
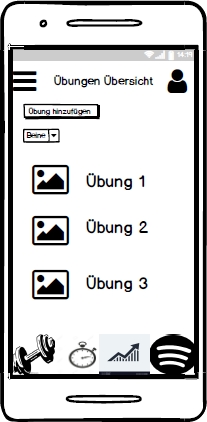
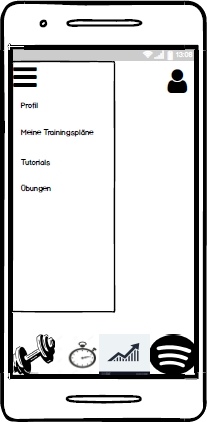
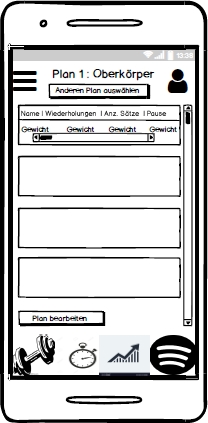
Abbildung q: Anforderungsverteilung im Überblick

# Entwurf

Ein erster Entwurf der Applikation fand mit Mockup sowie einem ER- und Aktivitätsdiagramm statt. Bereits im Entwurfsprozess wurden erste User Stories angelegt.

## Mockups

Die ersten Mockups sind im Folgenden in Abbildung m abgebildet.



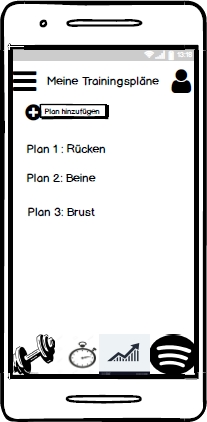
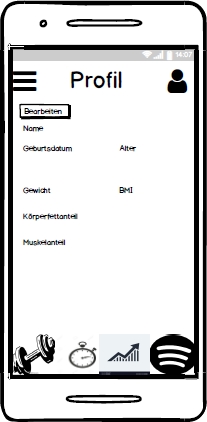


Abbildung m: Die ersten Entwürfe der Ansichten

Da sich im Verlaufe des Entwicklungsprozesses die Anforderungen geändert haben wurden auch dementsprechend die einzelnen Ansichten anders umgesetzt als ursprünglich geplant. Für einen detaillierten Vergleich können die Abbildungen im Kapitel Ergebnis herangezogen werden.

## ER-Diagramm

Das ER-Diagramm, dass hauptsächlich aus Must-Have Anforderungen abgeleitet ist im Folgenden dargestellt.

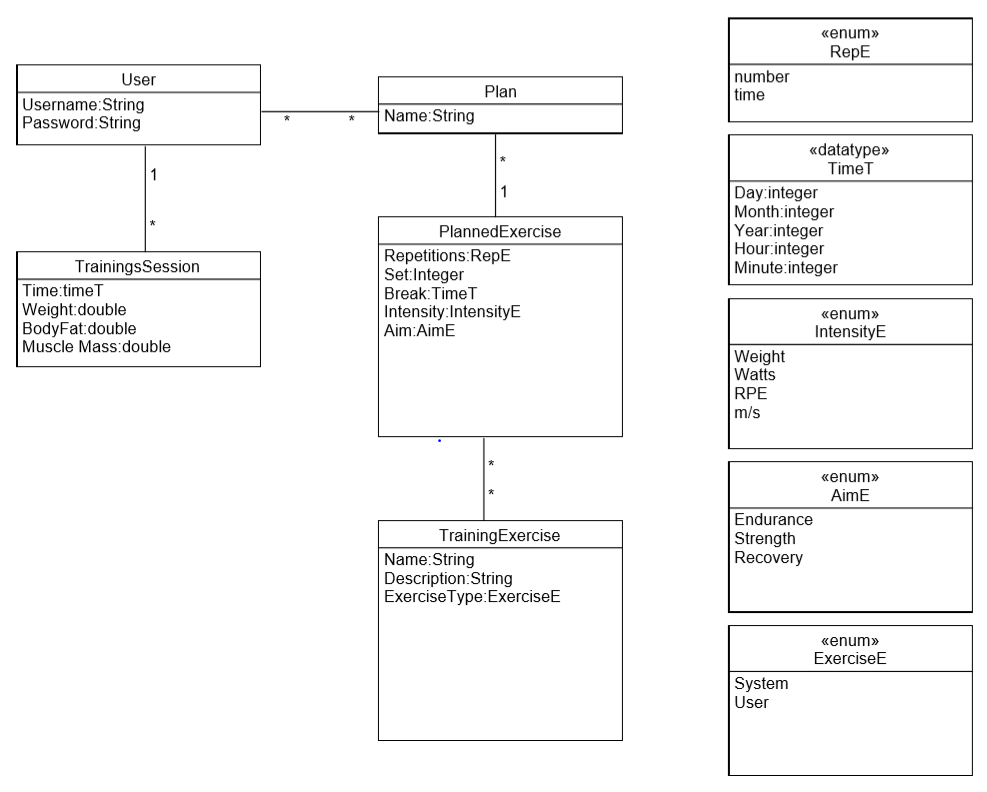


Abbildung e: Erster Entwurf eines ER-Diagrammes

## Aktivitätsdiagramm

## User Storys

Die Folgenden User Storys, teilweiße aus den Anforderungen abgeleitet sind im Entwurfsprozess erstellt worden. Trotzt der Veränderung der Anforderungen konnten einige User Storys im Wortlaut implementiert werden.

* Als Sportanfänger möchte ich mich nach dem Start der Applikation einloggen können, um auf mein gespeichertes Profil zugreifen zu können.
* Als Sportanfänger möchte ich die einzelnen Übungen in meinem Trainingsplan anschauen können, da ich manche Übungen noch nicht auswendig kann und sie ansonsten gar nicht ausführen kann.
* Als Sportanfänger möchte ich eine Zeitanzeige während der Ausgeführten Übung, da ich die Übung korrekt ausführen möchte
* Als wenig erfahrener Applikationsnutzer möchte ich ein wenig Überladenes Design der Applikation da ich sonst mit der Benutzung überfordert wäre.
* Als erfahrener Applikationsnutzer möchte ich ein Ansprechendes Design, da es mir dann mehr Spaß bereitet die Applikation zu benutzen.
* Als Freizeitsportler möchte ich eigene Trainingspläne erstellen können, da ich beginne mein Training zu individualisieren.
* Als Freizeitsportler möchte ich meine Leistungen der Trainingseinheit eintragen und speichern können, da ich meinen Fortschritt sichern möchte.
* Als Freizeitsportler möchte ich darüber informiert werden wenn ich mich Verbessere, da dies mich motiviert.
* Als Profisportler möchte ich eine umfassende Statistik, da ich an Längerfristigen Fortschritten interessiert bin.
* Als Profisportler möchte ich eine Nahrungsempfehlung für meinen Trainingsplan da ich die Trainingseffektivität und Trainingseffizienz steigern möchte.

# Projektprozess

Wir entschieden uns dazu für unser Projektmanagment das Vorgehensmodell Scrum umzusetzen. Das Ziel ist es eine lauffähige inkrementell verbesserte Software nach definierten Zeiteinheiten des Entwicklungsprozesses zu erhalten. Diese definierten Zeiteinheiten werden Sprints genannt. Das Team organisiert sich dabei selbst. Es gibt dabei keinen Projektleiter im klassischen Sinn. Ein Scrum Team besteht aus drei Rollen, dem Product Owner, dem Scrum Master, sowie dem Entwicklungsteam. Wir entschieden uns für dieses agile Vorgehen aus Gründen der

* Transparenz (Für alle Projektbeteiligten sind Fortschritte sowie Hindernisse nach jedem Sprint sichtbar)
* Überprüfung (Die Ergebnisse des Projektes werden Regelmäßig - ebenfalls nach jedem Sprintende - überprüft)
* Anpassung (Die Anforderungen werden kontinuierlich angepasst)

Im Folgenden werden die einzelnen Komponenten die wir während des Projektprozesses umgesetzt haben kurz erläutert.

**Product Owner**

Unter anderem besteht die Aufgabe des Product Owners darin, die Umsetzungen der einzelnen Phasen zu beurteilen. Er entscheidet über die Implementierung und hat ein Einfluss auf die Zeitpläne.

**Scrum Master**

Neben anderen Aufgaben sorgt der Scrum Master für eine entsprechende Moderation während den Team-Meetings. Er sorgt dafür dass keine störenden Einflüsse das Team beim Entwicklungsprozess hindert. Weiterhin sorgt er im Allgemeinen für eine gute Kommunikation sowie für die Bereitstellung der notwendigen Ressourcen für die Umsetzung des Projektes.

**Entwicklungsteam**

Das Entwicklungsteam arbeitet selbstständig und setzt die geforderten Funktionalitäten um.

**Sprints**

Die vordefinierten Zeiteinheiten in denen Funktionalität vom Entwicklungsteam umgesetzt wird werden Sprints genannt. Die Sprints sind dabei als zyklische Iterationen angedacht. Am Ende eines jeden Sprints soll ein voll Funktionsfähiges Produkt Inkrement stehen. In Abbildung x ist unser Sprintplan dargestellt. Insgesamt waren 5 Sprints vorgesehen.

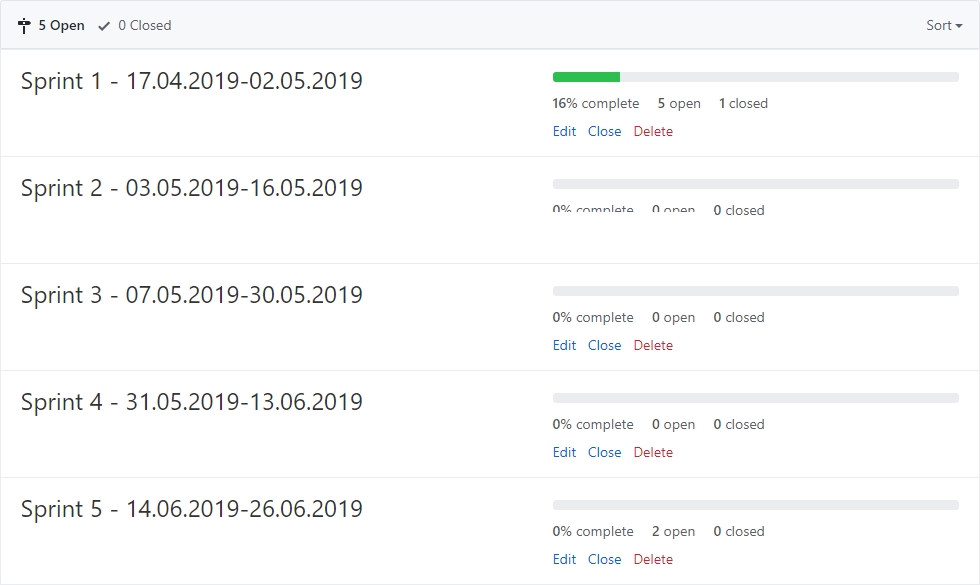


Abbildung x: Sprintplan während des ersten Sprints

**Kommunikation**

Am Anfang des Projektes kommunizierten wir über die Chat Plattform Discord. Wir entschieden uns dazu weil Discord Sprach- wie Videokonferenzfähig ist. Ein Auszug der Kommunikation ist in Abbildung xx dargestellt. Hauptsächlich nutzen wir Discord für organisatorische Belange wie z.B. das aufzeichnen von Terminen.

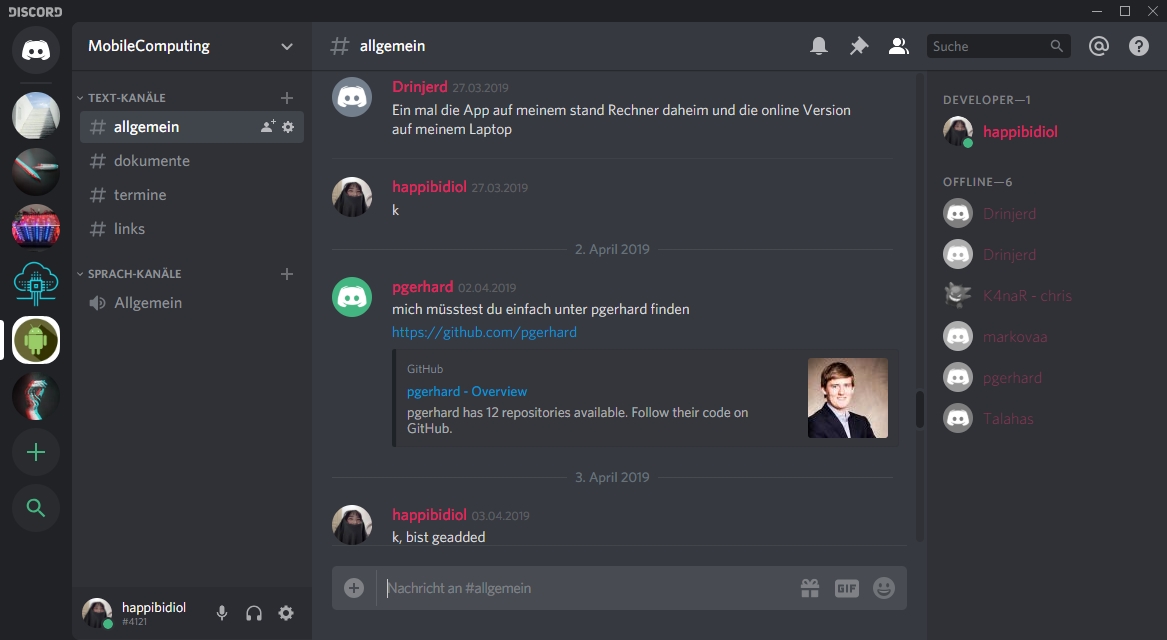


Abbildung xx: Kommunikation über Discord

Ebenfalls kommunizierten wir über den Messenger Dienst WhatsApp. Wir entschieden uns zusätzlich dafür, da dort eine schnellere Erreichbarkeit der einzelnen Teammitglieder gegeben war. Ein Auszug dieser Kommunikation ist in Abbildung xxx dargestellt.



Abbildung xxx: Kommunikation über WhatsApp

Weiterhin kommunizierten wir über GitHub. Hier wurde ausschließlich über Belange die direkt mit der Implementation des Projektes zusammenhingen kommuniziert. Ein Ausschnitt der Kommunikation in GitHub der mittels Issues erfolgte ist in Abbildung i dargestellt.

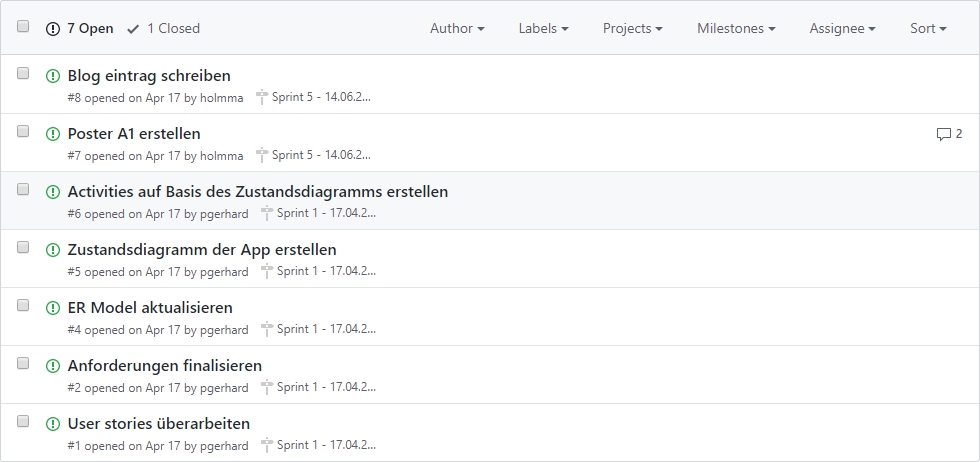


Abbildung i: GitHub Issues

**Umsetzung/Probleme**

Die Rollenverteilung in den Einzelnen Sprints ist in Tabelle 1 dargestellt.

Sprint 1-6: 17.04.19 – 26.06.19

|  |  |  |  |
| --- | --- | --- | --- |
|  | Product Owner | Scrum Master | Entwickler |
| Phillip Gerhard | x |  | x |
| Marvin Holm |  |  | x |
| Daniel Schott |  |  | x |
| Arina Markova |  | x | x |
| Tim Mangliers |  |  | x |
| Christian Michels |  |  | x |

Tabelle 1: Verteilung der Scrum Rollen

Das Vorgehensmodell Scrum benötigt aufgrund seines Aufbaus einen hohen Kommunikations- und Abstimmungsaufwand. Da die einzelnen Mitglieder der Gruppe sehr unterschiedliche Stundenpläne hatten waren gemeinsame Treffen für alle nicht immer möglich. Dieses Problem wurde aber durch ein schneiden der Komponenten, was eine simultane Bearbeitung unterschiedlicher Mitglieder möglich machte, behoben.

# Implementierung

## Konzeptuelles Modell

Konzeptuell ist unsere Applikation in zwei Teile aufgeteilt.

## Verwendeter Technologiestack

## Aktivitätsdiagramme

## Sequenzdiagramme

## ER-Diagramme

## Code-Ausschnitte

# Ergebnis

## Activity „Login/Registrierung“

Ein neuer Benutzer muss sich bei der Login Seite registrieren, wenn er die App nutzen möchte. Dabei muss er einen Benutzernamen mit mindestens 3 Zeichen und ein Passwort mit mindestens 5 Zeichen angeben.

Wenn der eingegebene Name nicht bereits von einem anderen Benutzer verwendet wird, kann ein neuer Eintrag in der Datenbank angelegt werden und der Benutzer wird erfolgreich registriert.

Wenn der Benutzer schon ein eigenes Konto hat, kann er sich einfach mit seinen persönlichen Daten einloggen.

## Activity „Trainingspläne“

## Activity „Statistik“

## Activity „…“

## Activity „……“

# Tests und Evaluierung

## User Interface Testing

Prinzipiell gibt es bei Android Applikation zwei unterschiedliche Arten des Testings

* Unit Tests
* Instrumentation Tests

Die Unit Tests erfolgen lokal auf der Entwicklungsumgebung. Hierbei werden die einzelnen Methoden und Funktionen getestet. Beispielsweiße ob eine Funktion foo(…) mit Parameter x die korrekte Rückgabe y liefert.

Instrumentation Tests haben die Aufgabe Android Frameworks zu testen. Das kann z.B. das User Interface sein. Diese Tests werden direkt auf einem Endgerät oder in einem Emulator ausgeführt.

Hier kommt das Instrumentation Testing Framework Espresso ins Spiel. Mit Espresso lassen sich einfach automatische Tests für das User Interface generieren. Espresso wird einfach den Dependencies hinzugefügt. Anschließend können einzelne zu testende Komponenten mit bestimmten Annotationen versehen werden. Diese Annotationen spezifizieren das Verhalten der einzelnen Tests. Im weiteren können die einzelnen Komponenten dann mit Testcode versehen werden. Alternativ kann der Espresso Test Recorder genutzt werden. Dieser erstellt automatisch User Interface Tests. Dies geschieht indem der Tester ein eigens erstelltes und selbst durchgeführtes Testszenario mit dem Recorder aufnimmt. Aus diesem Aufgezeichnetem Szenario generiert der Espresso Test Recorder dann einen entsprechenden User Interface Test. Unsere User Interface Tests wurden mit dem Espresso Test Recorder durchgeführt. Einzelne Auszüge des Prozesses sind in den Abbildungen 11-14 dargestellt.

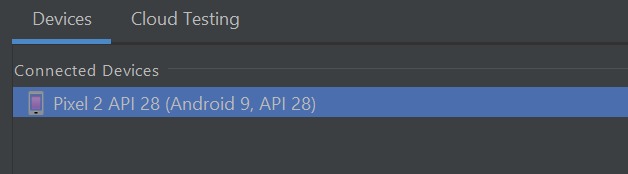


Abbildung 11: Espresso Tests wurden auf dem Pixel 2 durchgeführt

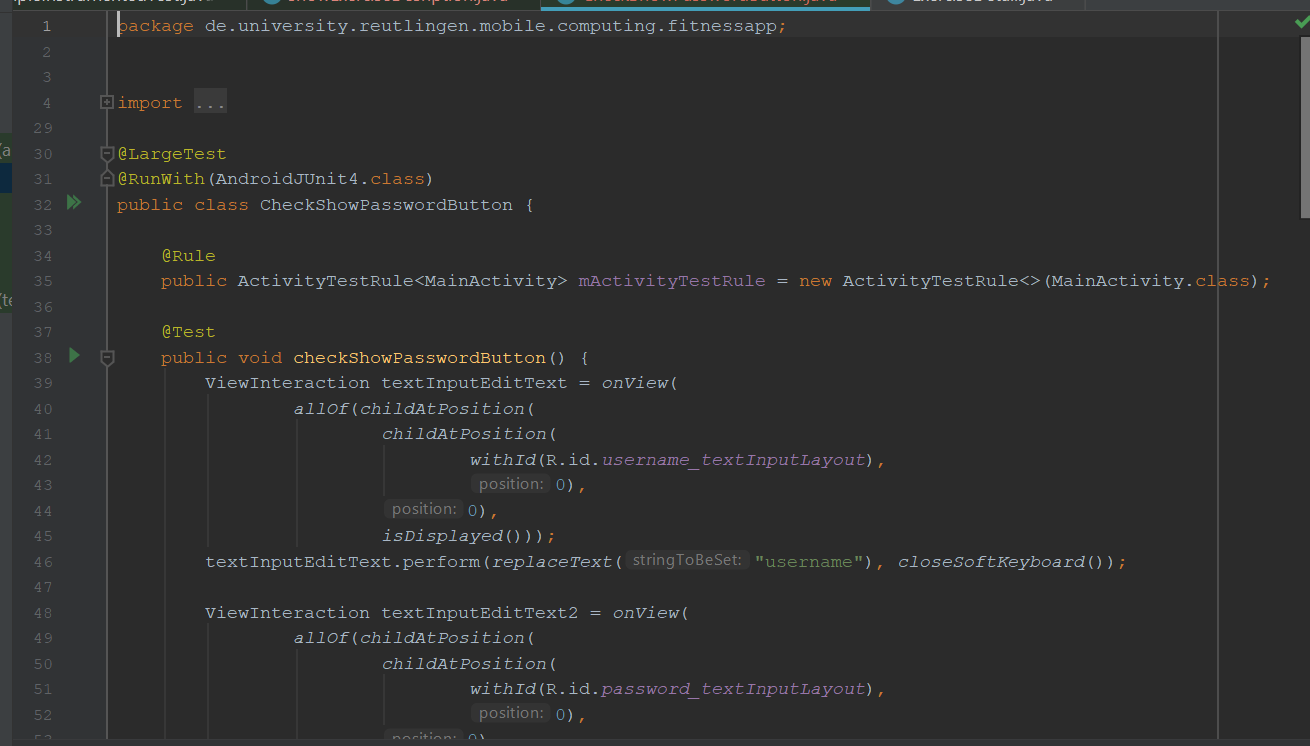


Abbildung 12: Von Espresso autogenerierter Code

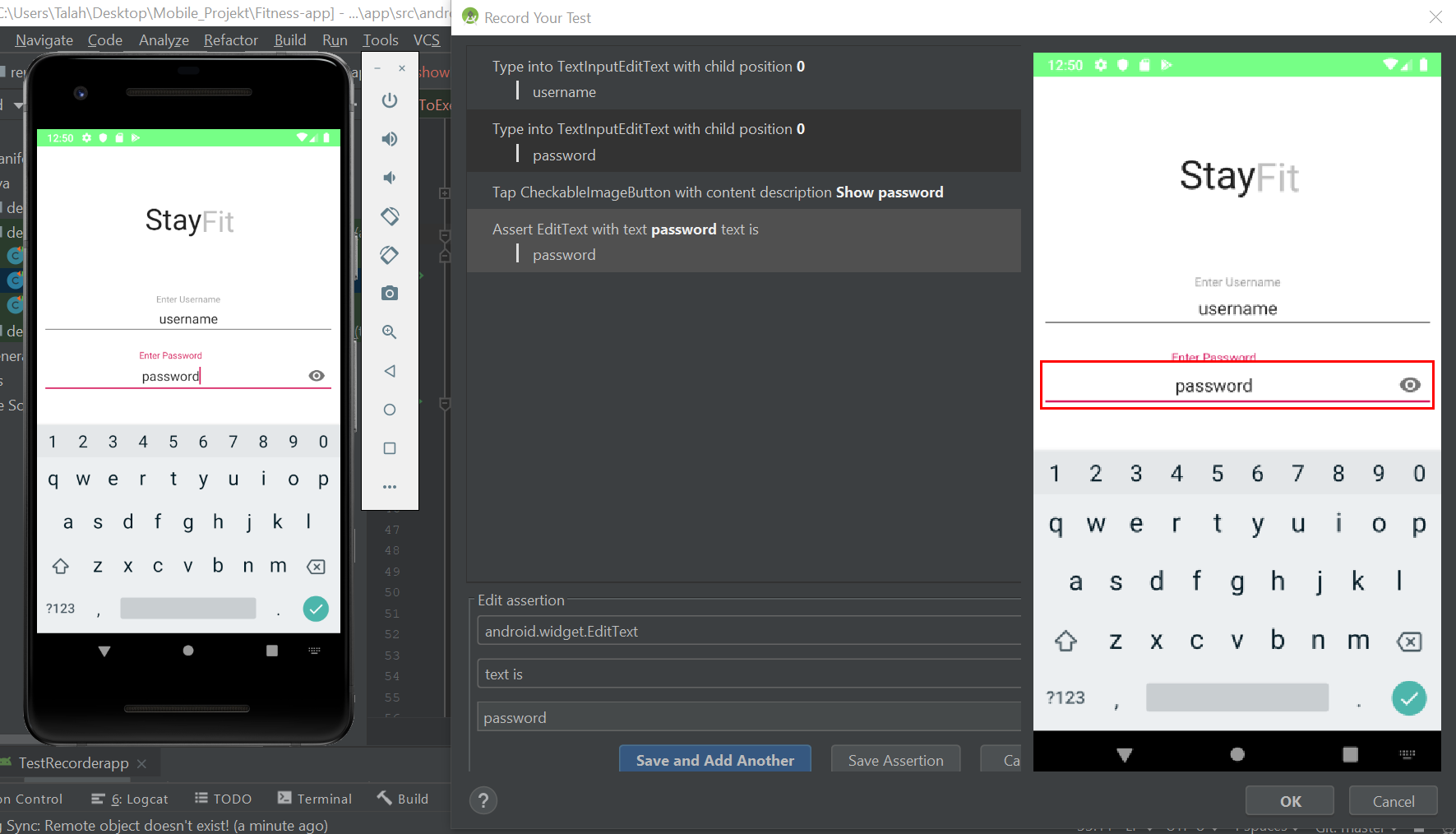


Abbildung 13: Der Espresso Test Recorder bei der Aufnahme

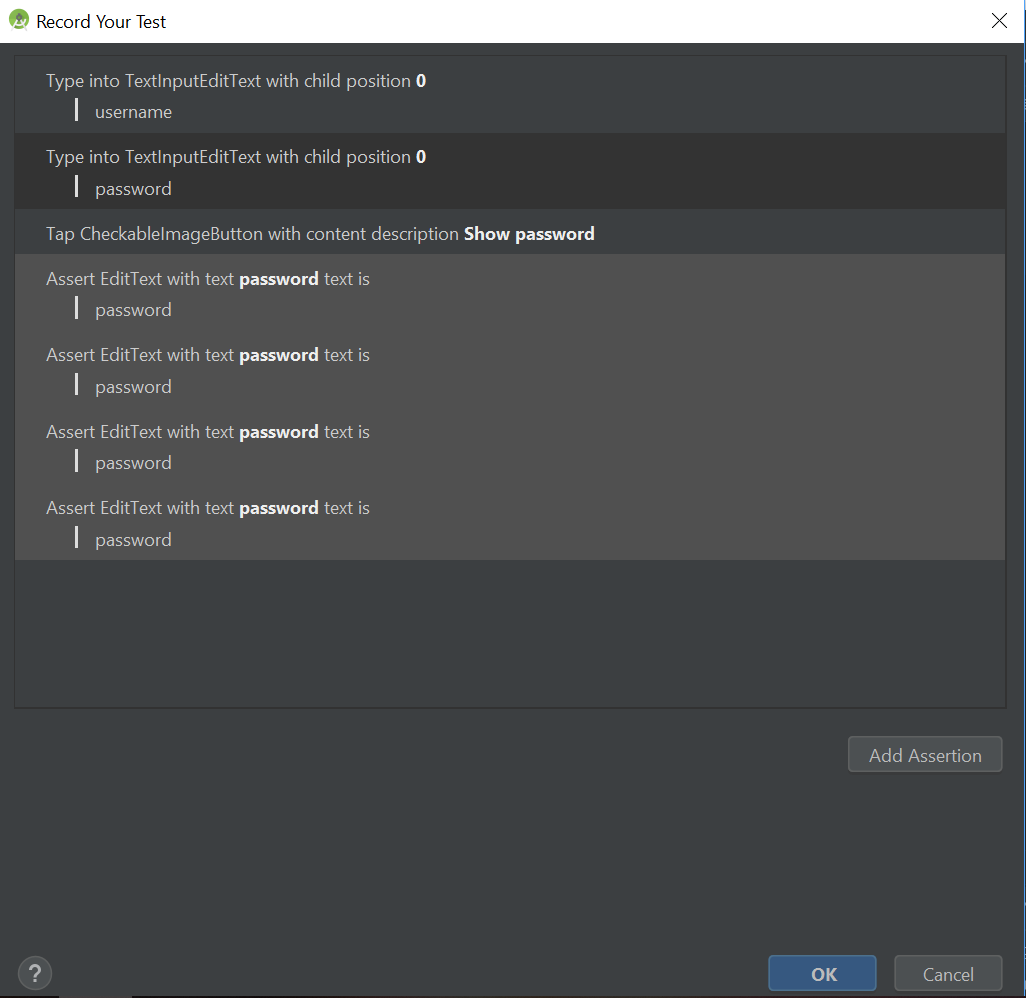


Abbildung 14: Log des Espresso Recorders

In Abbildung 12 ist der autogenerierte Code des Recorders dargestellt. Sollte die Applikation in einem Iterationsschritt ändern, so kann der Test Code einfach angepasst werden. Der Recorder ist einfach zu bedienen und erleichtert das Testing ungemein. Jedoch gibt es spezielle Restriktionen. Es kann nur eine bestimmte Menge an Assertions verwendet werden. Werden speziellere Assertions benötigt, so müssen diese händisch implementiert werden. In unserem Fall war dies jedoch nicht nötig. Es kam jedoch zu Problemen beim Testen asynchroner Abläufe. Hier kommt der Recorder an seine Grenzen. In diesem Fall können Espresso idling resources genutzt werden. Diese repräsentieren Asynchrone Operationen welche das User Interface verwenden kann.

## Evaluierung

Die Applikation wurde von 10 Personen mithilfe der folgenden 5 Fragen evaluiert.

1. Wie oft Trainierst du? (Einfachauswahl)
   1. Weniger als zweimal die Woche
   2. Zwei bis dreimal die Woche
   3. Mehr als dreimal die Woche
2. Hast du schon einmal eine Fitness Applikation benutzt? (Einfachauswahl)
   1. Ja
   2. Nein
3. Würdest du die fertige Applikation anhand dem Eindruck den du gerade gewonnen hast im Trainingsalltag verwenden? (Einfachauswahl)
   1. Ja
   2. Nein
4. Was stört dich gerade an der Applikation? (Mehrfachauswahl)
   1. Sie ist unfertig
   2. Das Design
   3. Nichts
5. Welches Feature ist dir am wichtigsten? (Einfachauswahl)
   1. Statistik
   2. Ernährungsplan
   3. Spotify

Die Befragung ergab ein in Abbildung s dargestelltes Ergebnis.

Abbildung s: Befragungsergebnisse

Unsere Applikation würde von der Mehrheit der User wieder benutzt werden. Dabei hat die Mehrheit der Benutzer schon einmal eine Fitness Applikation benutzt. Bei niedriger Nutzerakzeptanz ist Hauptsächlich darauf zurückzuführen, dass die Applikation noch nicht vollständig ist.

# Konklusion

# Links

* GitHub Repositorien des Projektes
  + <https://github.com/eightfour/Fitness-app>
  + <https://github.com/pgerhard/fitness-app-server>

# Declaration on oath

I hereby declare that I have written the present work on my own, that I have not used any external help other that declared and that all helping material and sources used fort he work have been properly named and referenced.

I am aware that an untruthful declaration will be considered as a deceit.

Reutlingen 26.06.2019 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Place, Date Signature

(Philipp Gerhard)

Reutlingen 26.06.2019 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Place, Date Signature

(Marvin Holm)

Reutlingen 26.06.2019 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Place, Date Signature

(Daniel Schott)

Reutlingen 26.06.2019 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Place, Date Signature

(Arina Markova)

Reutlingen 26.06.2019 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Place, Date Signature

(Tim Mangliers)

Reutlingen 26.06.2019 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Place, Date Signature

(Christian Michels)