PROJEKTDOKUMENTATION

**Projekt: MuscleLog**

Lehrveranstaltung: INF303 - Software Engineering Projekt

Semester: WS2019/20

Lehrveranstaltungsleiter/in: Dr. Burcu Yildiz

Projektmanager/in: Muhammed Zahid Bozkus

Version: 3.0

**Inhaltsverzeichnis**

Abkürzungsverzeichnis 3

Abbildungsverzeichnis 3

Tabellenverzeichnis 3

1. Lastenheft 4

1.1 Zielbestimmung 4

1.2 Produkteinsatz 4

1.3 Produktfunktionen 4

1.4 Produktdaten 4

1.5 Produktleistungen 4

1.6 Qualitätsanforderungen 4

1.7 Ergänzungen 5

1.8 Glossar 5

2. Projektplanung 6

2.1 Projektzieleplan 6

2.2 Projektmeilensteinplan 7

2.3 Projektkostenplan 7

2.4 Projektrisiken 8

3. Projektumsetzung 8

3.1 Einleitung 8

3.1.1 Problemstellung (& Motivation) 8

3.1.2 Stand der Technik 8

3.1.3 Zielsetzung 12

3.2 Anforderungsanalyse und Konzeption 12

3.3 Entwicklung und Implementierung 13

3.3.1 Anwendungsfälle 13

3.3.2 EER und UML Domänenmodell 14

3.3.3 Funktionale und nichtfunktionale Anforderungen 15

3.3.4 Systemarchitektur 16

3.3.5 Methoden und Werkzeuge 17

3.3.6 Design-Layouts 17

3.4 Zusammenfassung und Ausblick 19

4. Stundenliste 20

## Abkürzungsverzeichnis

|  |  |
| --- | --- |
| **Abkürzung** | **Bedeutung** |
|  |  |
|  |  |
|  |  |
|  |  |

## 

## Abbildungsverzeichnis

Abbildung 1: Screenshots der App Strong 9

Abbildung 2: Screenshots der App FitNotes 10

Abbildung 3: Screenshots der App Ready Gym Log 11

Abbildung 4: Screenshots der App Workit 11

Abbildung 5: Anwendungsfälle 13

Abbildung 6: EER Diagramm 14

Abbildung 7: Domain Model 15

Abbildung 8: Systemarchitektur 16

Abbildung 9.1: Application Flow 1 17

Abbildung 9.2: Application Flow 2 18

Abbildung 9.3: Application Flow 3 18

# Lastenheft

## Zielbestimmung

Ziel der MuscleLog Software ist, eine mobile Applikation zu erstellen, die als ein Trainingstagebuch dienen soll. Der Benutzer soll in der Lage sein, Trainingseinheiten zu erstellen und Übungen, sowie die dazugehörigen Informationen (Satzanzahl, Gewicht, Wiederholungsanzahl), in diese Einheiten einzutragen. Dadurch wird die Aufzeichnung einzelner Einheiten erzielt, die über längere Perioden die Entwicklung einer Person in bestimmten Übungen zeigen kann.

## **Produkteinsatz**

Die Applikation ist in erster Linie für die Benutzung in Fitnessstudios gedacht, da die Datenbank eine vorgefertigte Liste an klassischen Übungen haben wird, die in den meisten Studios durchgeführt werden können. Zielgruppe des Produktes sind Personen, die ihre Trainingseinheiten systematischer gestalten oder einfach nur ihre Progression aufzeichnen wollen. Das System ist wahrscheinlich für Neuanfänger im Krafttraining nicht geeignet, da man eine bestimmte Terminologie beherrschen muss um die verschiedenen Übungen erkennen und unterscheiden zu können.

## **Produktfunktionen**

/LF10/ Der Benutzer kann eine neue Trainingseinheit erstellen.

/LF20/ Der Benutzer kann innerhalb neu eröffneten Trainingseinheit Übungen eintragen.

/LF30/ Der Benutzer kann verschiedene Informationen (Satzanzahl, Gewicht, RPE) über die einzelnen Übungen speichern.

/LF40/ Der Benutzer kann vergangene Trainingssessions aufrufen.

## Produktdaten

/LD10/ Benutzerdaten

* Trainingsdaten
* Körpermessungen (optional, noch nicht festgelegt)
* Trainingseinheiten

## Qualitätsanforderungen

Bei dieser Applikation stehen an erster Stelle die Funktionalität und die Benutzbarkeit. Es ist besonders wichtig eine einfach bedienbare bzw. benutzerfreundliche User Interface zu gestalten, da es bei Applikationen dieser Art sehr leicht ist die Übersicht zu verlieren. Der Benutzer sollte einen genauen Überblick über sein aktuelles Training haben. Darüber hinaus sollte es einfach sein vergangene Einheiten abzurufen und die Daten zu analysieren. Bei einem Klick des Benutzers sollten Funktionen recht schnell erfolgen. Darüber hinaus ist die Zuverlässigkeit ein sehr wichtiger Aspekt. Da der eigentliche Fokus auf den gespeicherten Daten ist, sollten keine Fehler bezüglich der Verbindung zum Datenbank vorkommen.

## Ergänzungen

## Glossar

# Projektplanung

## 2.1 Projektzieleplan

|  |  |  |
| --- | --- | --- |
| **Projektzieleplan** | | |
| **Zielart** | **Projektziele** | **Adaptierte Ziele per** **<Datum> falls vorhanden** |
| **Hauptziel (Output):** | Es soll eine mobile Applikation erstellt werden, die zum Speichern und zum Analysieren von Trainingsdaten bzw. einzelnen Übungen dienen soll. |  |
| **Teilziele:** | Seite für neue Trainingseinheit:   * Neue Übung hinzufügen (Button) * Gewicht, Satzanzahl eintragen * Einheit beenden   Seite für vergangene Einheiten:   * Liste für ältere Einheiten nach Datum * Erwünschte Einheit auswählen   Seite für Einstellungen |  |
| **Nicht-Ziel:** | Es soll keine mobile Applikation erstellt werden, die persönliche Trainingspläne entwirft bzw. bestimmte Arbeitsprogramme vorschreibt. |  |
| **Projektnutzen (Outcome):** | Besonders fortgeschrittene Athleten, aber auch Anfänger machen sich nach einiger Zeit schwer die besten Resultate aus ihrem Training herauszubekommen. Einer der Gründe dafür ist die fehlende Aufzeichnung von Trainingseinheiten, da man auf längerer Dauer nicht weiß, was für einen Fortschritt man gemacht hat. Durch diese Applikation soll es möglich sein eigene Daten während des Trainings zu speichern und sie später zu analysieren. |  |

## 2.2 Projektmeilensteinplan

|  |  |  |  |
| --- | --- | --- | --- |
| **Meilenstein** | **Plantermin** | **Angepasster Plantermin** | **Ist-Termin** |
| Projektstart | 22.09.2019 | 22.09.2019 | 22.09.2019 |
| Themenrecherche, Auswahl des Projektthemas | 05.10.2019 | 05.10.2019 | 05.10.2019 |
| Zielsetzung, Projekt konkretisieren | 07.10.2019 | 07.10.2019 | 07.10.2019 |
| Stand der Technik Analyse, Motivation, Lastenheft | 13.10.2019 | 13.10.2019 | 13.10.2019 |
| Datenmodell, Mockups erstellen | 20.10.2019 | 22.10.2019 | 22.10.2019 |
| Zu benutzende remote Datenbank bestimmen | 25.10.2019 | 25.10.2019 | 25.10.2019 |
| Übungskatalog erstellen | 30.10.2019 | 30.10.2019 | 30.10.2019 |
| Design implementieren | 04.11.2019 | 25.11.2019 | - |
| Implementierung des Übungskatalogs | 07.11.2019 | 18.11.2019 | 18.11.2019 |
| Implementierung des Untermenüs New Workout | 14.11.2019 | 14.11.2019 | 14.11.2019 |
| Datenbankverbindung herstellen | 17.11.2019 | 02.11.2019 | 02.11.2019 |
| Implementierung des Untermenüs History | 25.11.2019 | 15.11.2019 | 15.11.2019 |
| Anpassungen | 05.11.2019 | 05.11.2019 | - |
| Tests, Evaluierungen | 15.11.2019 | 15.11.2019 | - |
| Doku Final | 22.11.2019 | 22.11.2019 | - |
| Projektende | 29.12.2019 | 29.12.2019 | - |

## 2.3 Projektkostenplan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Projektkostenplan** | | | | |
| **Arbeitspaket / Meilenstein** | **Kostenart** | **Menge** | **Preis / Einheit** | **Kosten [€]** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

## 2.4 Projektrisiken

|  |  |  |  |
| --- | --- | --- | --- |
| **Projektrisikoanalyse** | | | |
| **Risiko** | **Eintritts-Wahrscheinlichkeit** | **Schadensausmaß / Auswirkungen** | **Maßnahmen** |
| Probleme mit der Server- bzw. Datenbankverbindung | gering |  |  |

# 

# Projektumsetzung

**3.1Projektumsetzung**

## Einleitung

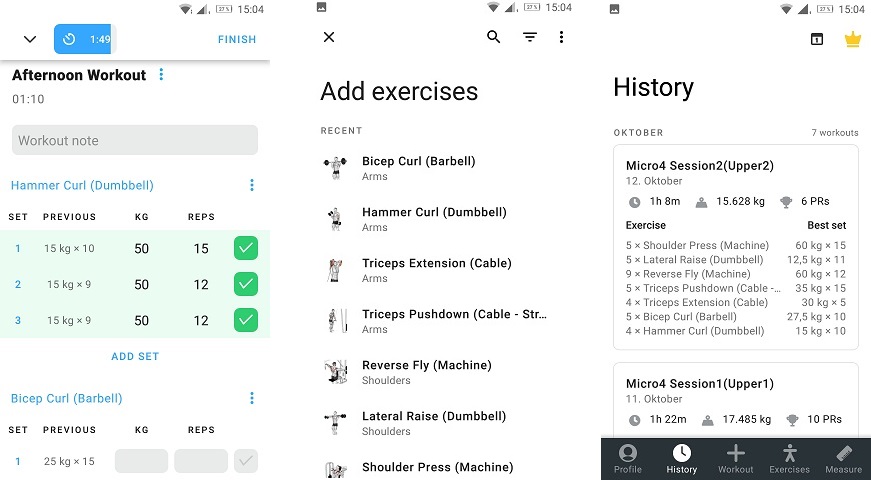
### 3.1.1 Problemstellung (& Motivation)

Die Aufzeichnung von Daten ist bei sportlichen Aktivitäten sehr wichtig. Das gilt besonders für Bereiche, die einer bestimmten Routine folgen und ähnliche Bewegungen langfristig mehrmals wiederholen. In unserem Fall geht es besonders um Kraft- und Widerstandstraining. Die Speicherung von bestimmten Informationen spielt bei solchen Trainingsarten eine wichtige Rolle. Diese Informationen ermöglichen uns, unseren Fortschritt zu sehen und geben eine Idee darüber, wie wir weitermachen können. Das Ziel ist daher eine mobile Applikation zu erstellen, die zu diesem Zweck dienen soll. Diese Applikation soll Athleten, unabhängig davon ob sie fortgeschritten sind oder nicht, ermöglichen, alle relevanten trainingstechnischen Daten einzutragen und zu speichern. Somit wird erzielt, dass der Benutzer nach mehreren Monaten bzw. Jahren seine Daten vergleicht und sich ein größeres Bild über seine Entwicklung verschafft. Natürlich muss man erwähnen, dass es schon mehrere mobile Applikationen in diesem Bereich gibt. Einige von diesen werden im Kapitel Stand der Technik näher vorgestellt. Die Funktionalitäten stimmen bei den meisten Apps überein. Ein sehr wichtiges Teilziel bei diesem Projekt wird es jedoch sein, eine Applikation zu erstellen, die ein sehr einfach bedienbares User Interface hat und somit benutzerfreundlicher ist. Außerdem soll hauptsächlich auf die wichtigsten Funktionalitäten fokussiert werden. Damit ist gemeint, dass diese Applikation z.B. keine vorgefertigten Pläne enthalten soll, sondern nur als ein Tagebuch benutzt werden soll.

### 3.1.2 Stand der Technik

Applikationen, die als Trainingstagebuch dienen sind heutzutage sehr oft zu finden. Sowohl für Android, als auch für iOS gibt es zahlreiche Apps. Die bekanntesten davon werden in diesem Kapitel analysiert.

* **Strong**

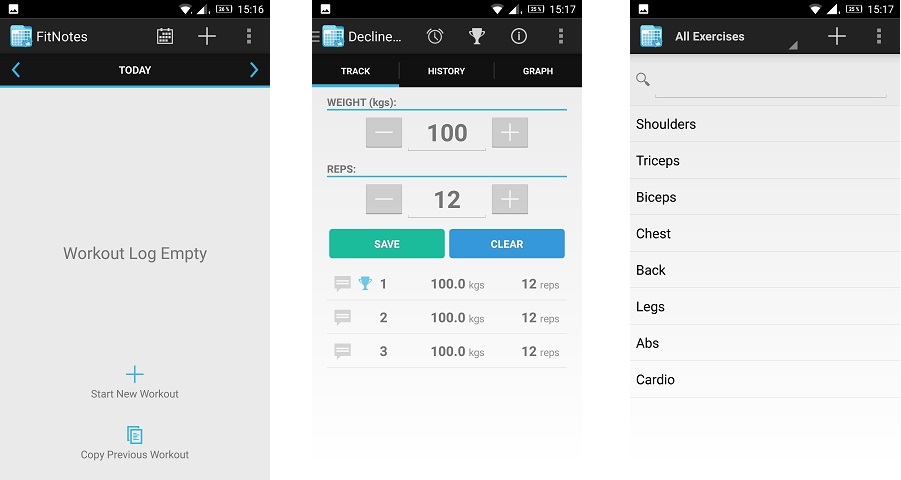
****

**Abbildung 1: Screenshots der App Strong**

Die von Strong Fitness entwickelte Applikation Strong hat mehr als 500.000 Downloads im Play Store. Die allgemeine Bewertung liegt bei 4.8. Der Benutzer hat bei Strong eine sehr gute Übersicht. Es ist sehr einfach eine neue Einheit zu erstellen und Übungen einzutragen. Man kann neben den einzelnen Sätzen auch das Gewicht und die Wiederholungsanzahl schnell eintragen. Bei den einzelnen Übungen kann man zugehörige Bilder und Erklärungen finden, was auch für Anfänger sehr nützlich ist. Darüber hinaus kann man einstellen, ob man das metrische System oder das angloamerikanische System benutzt, was sich als sehr nützlich erweist. Unter dem Menüpunkt History kann man sich vergangene Einheiten anschauen. Außerdem wird bei der Übung im aktuellen Training angezeigt, welche Gewichte man für wie viele Wiederholungen bei der letzten Durchführung dieser Übung benutzt hat. Obwohl das auch ziemlich nützlich erscheint, kann es jedoch ab und zu zu Verwirrungen führen. Einer der negativen Aspekte bei dieser Applikation ist, dass man bestimmte Einheiten nicht in einem separaten Ordner einordnen kann, was dazu führt, dass man vielleicht bisschen länger nach einer Einheit suchen muss, welche man vor einem Monat durchgeführt hat. Neben den funktionellen Seiten gibt es auch eine Profilseite und Messungsseite, wo man optional eigene körperliche Messungen eintragen kann.

Link: <https://www.strong.app>

* **FitNotes**

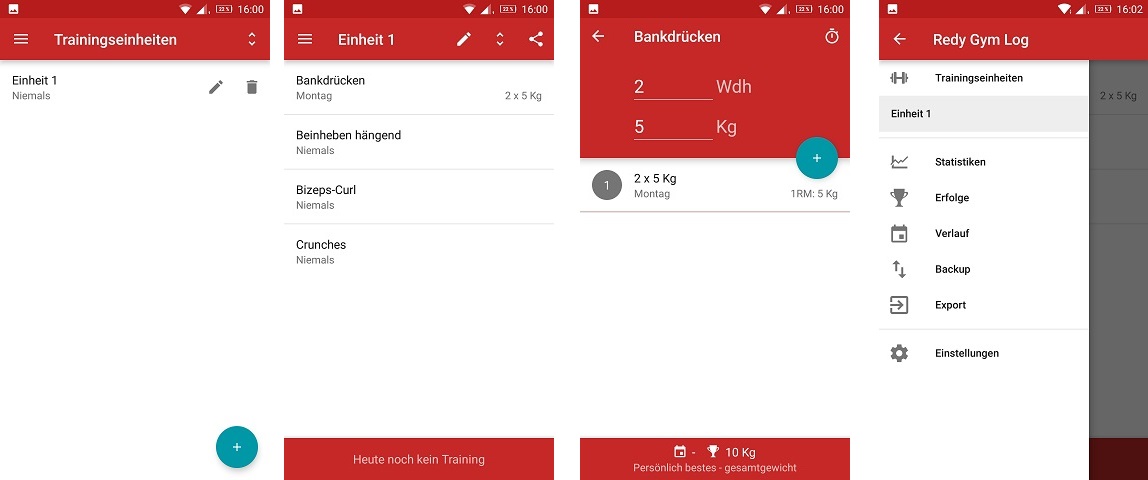


**Abbildung 2: Screenshots der App FitNotes**

FitNotes ist eine einfachere Applikation, die ein „sauberes“ Design hat. Inzwischen hat die von James Gay entwickelte Applikation mehr als 1 Mio. Downloads im Play Store. Besonders auffallend in dieser App ist die einfache User Interface. Sobald man die App geöffnet hat, kann man direkt eine neue Trainingseinheit erstellen oder eine vergangene Einheit wiederholen. Man muss sagen, dass alle nötigen Funktionalitäten zur Speicherung von Trainingsdaten vorhanden sind, ohne zusätzliche, vielleicht auch verwirrende, Funktionen. Daher wäre diese App für eine Person, die noch nie ein Trainingslog benutzt hat, sehr empfehlenswert. Im Gegensatz zu Strong werden ältere Einheiten im Kalenderansicht angezeigt. Man kann sich nach Datum alle absolvierten Einheiten anschauen und die Daten analysieren. Ein Kritikpunkt ist die limitierte Anzahl der vorhandenen Übungen in der Übungsbibliothek. Man kann zwar eigene Übungen erstellen und diese in die Übungsbibliothek eintragen, jedoch ist es in Bezug auf die Benutzbarkeit nicht wirklich effizient.

Link: <https://play.google.com/store/apps/details?id=com.github.jamesgay.fitnotes>

* **Ready Gym Log**

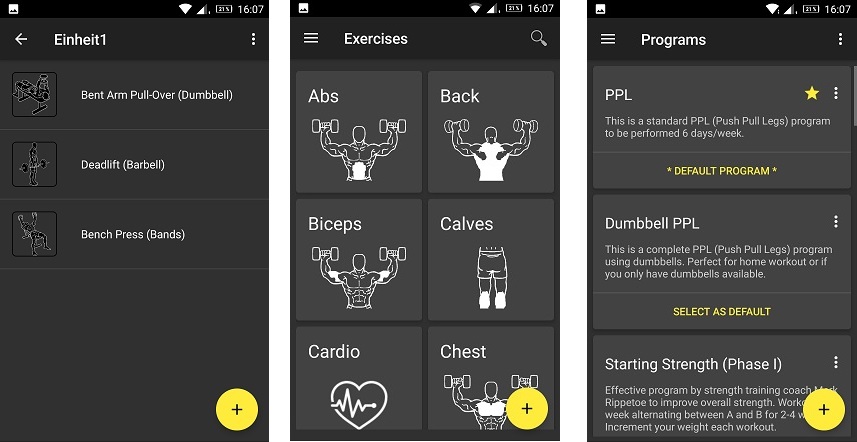
****

**Abbildung 3: Screenshots der App Ready Gym Log**

Ready Gym Log von Redy Apps wurde mehr als 100.000 Mal im Play Store heruntergeladen und hat eine Bewertung von 4.5. Auch dieses Trainingslog zeichnet sich durch die übersichtliche, einfache und intuitive Bedienung ab. Es gibt keine verwirrenden unnötigen Buttons, sobald man die App startet sieht man genau was zu tun ist. In einer Sidebar-Menu kann man Untermenüs wie Statistiken, Erfolge und Verlauf finden. Es müssen jedoch einige Kritikpunkte erwähnt werden. Wie bei FitNotes hat man hier auch eine limitierte Anzahl an Übungen. Außerdem werden bei den einzelnen Einheiten für jede Übung ein neues Fenster geöffnet. Die Übungen werden also nicht auf einer Seite gelistet, was dem Benutzer Zeit kostet und nicht sehr effizient ist.

Link: <https://play.google.com/store/apps/details?id=com.redyApps.redyGymLog&hl=de>

* **Workit**

****

**Abbildung 4: Screenshots der App Workit**

Workit von WorkIt hat eine Bewertung von 4.4 mit mehr als 100.000 Downloads im Play Store. Im Gegensatz zu den einfacheren Trainingslogs bietet Workit vorgefertigte Trainingsprogramme an, die man einfach anwenden kann. Das kann für Benutzer nützlich sein, die wenig Erfahrung mit Training haben und nach einem Startpunkt suchen. Neben den vorgefertigten Programmen kann man natürlich auch eigene Pläne erstellen und Übungen eintragen. Workit hat eine große Übungsbibliothek. Wie bei den anderen Applikationen gibt es Untermenüs wie Statistiken, Einheiten und Übungen. Die Benutzerfreundlichkeit ist nicht schlecht, jedoch ist alles nicht immer übersichtlich. Um eine neue Einheit zu erstellen muss man den Namen und die Beschreibung eingeben, was zu einem weiteren Pop-up Fenster führt, wo man wählen soll an welchen Tagen man die Einheit durchführen wird. Erst danach kommt man auf die eigentliche Seite, wo man Übungen eintragen kann. Daher ist es im Vergleich mit den anderen Apps, bei denen man mit einem Klick das Training starten kann, eher nachteilig.

Link: [www.workoutwithworkit.com](http://www.workoutwithworkit.com)

### 3.1.3 Zielsetzung

Das Produkt ist in erster Linie für die Benutzung in Fitnessstudios gedacht. Die Idee ist, dass der Benutzer seinen Satz absolviert und gleich danach die Daten über den Satz (Gewicht, Wiederholungen) einträgt. Die Zielgruppe besteht daher aus Personen, die regelmäßig in einem Studio trainieren und auch ihre Informationen speichern wollen. Ob diese Personen Anfänger oder Fortgeschrittene sind ist nicht wirklich relevant, jedoch muss der Benutzer wenigstens die Namen der Übungen kennen, damit er sie auch eintragen kann.

Die Hauptfunktionen beinhalten das Erstellen von neuen Einheiten, das Hinzufügen von Übungen bzw. das Eintragen der Informationen und das Abrufen von älteren Trainingssessions.

Als Daten sollen in der Applikation hauptsächlich die Trainingsinformationen enthalten sein. Es wird optional eine Seite geben wo man Körpermessungen eintragen und aktualisieren kann, dieses ist aber noch nicht festgelegt. Soweit es möglich ist wird das System keine Registrierung oder Login erfordern.

## 3.2 Anforderungsanalyse und Konzeption

**Produktperspektive**

Die Software wird in Android Studio mit Java geschrieben. Das Endprodukt soll auf mobilen Geräten mit einem Android Betriebssystem laufen. Es bietet Benutzern die Möglichkeit, Trainingsroutinen innerhalb der Anwendung zu erstellen oder zu speichern.

**Produktfunktionen**

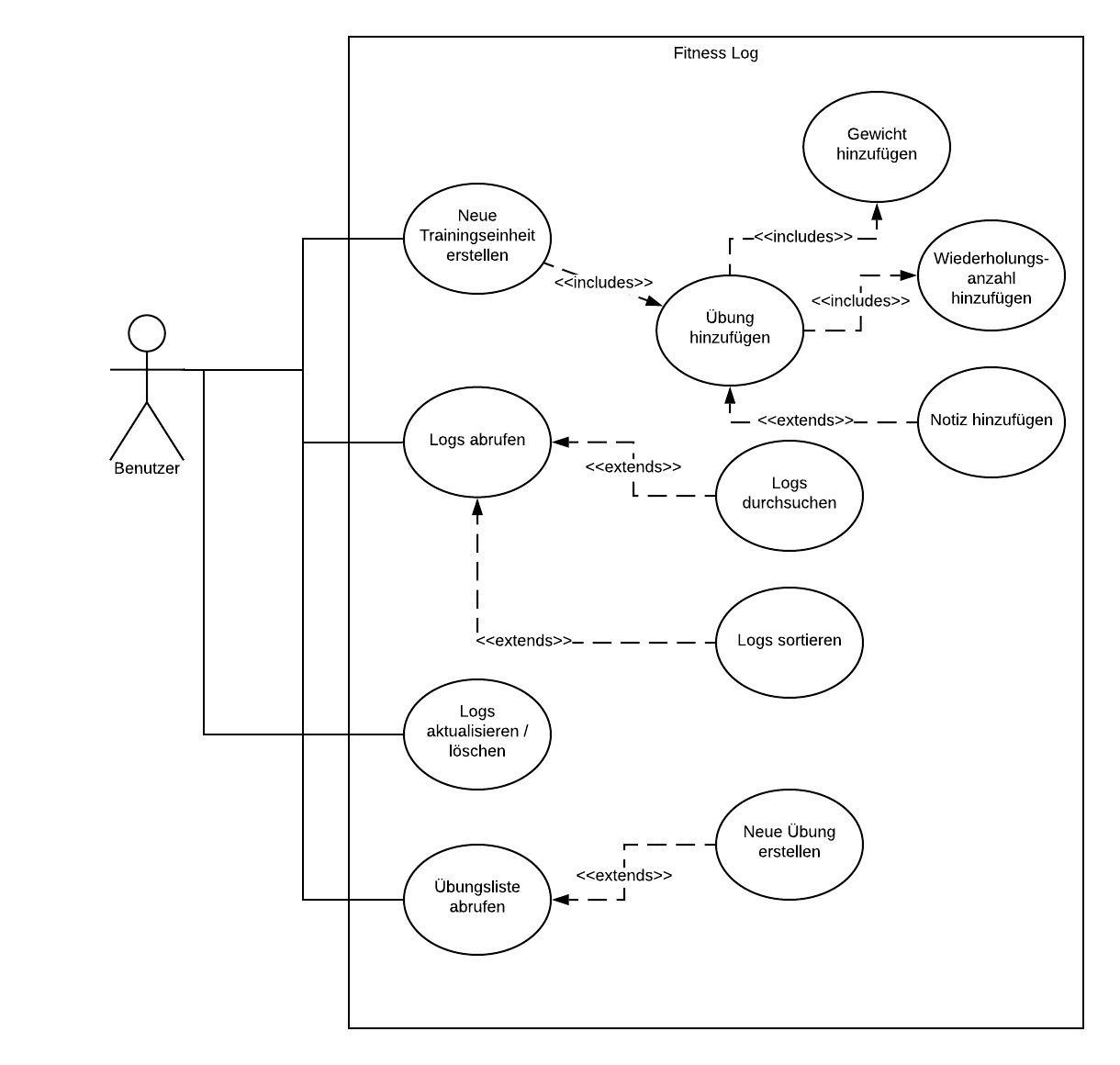
* Nachdem sie die Applikation gestartet haben können Benutzer eine leere Session erstellen.
* Benutzer können die vorgefertigte Liste von Übungen durchsuchen und die Übungen ihrer Wahl ihrer Session hinzufügen.
* Wenn eine bestimmte Übung nicht existiert, kann der Benutzer seine eigene Übung erstellen und der Liste hinzufügen.
* Nachdem sie ihre Sessions beendet haben können die Benutzer diese abspeichern.
* Benutzer können abgespeicherte Sessions später unter dem Menüpunkt History aufrufen und aktualisieren/modifizieren.

**Benutzereigenschaften**

* Normale Anwender erhalten uneingeschränkten Zugriff auf die Anwendung. Sobald ein Benutzer die Applikation startet kann er Trainingsroutinen erstellen, Trainingsroutinen anzeigen, Trainingsroutinen aktualisieren, Routinen zur Bearbeitung in seiner persönlichen Liste speichern.
* Der Administrator verwaltet die Daten in der Anwendung hinter den Kulissen. Er ist für die Verwaltung der Trainingsprogrammdaten verantwortlich. Der Administrator führt auch die tägliche Verwaltung innerhalb der Anwendung durch und stellt die Qualität des Anwendungsinhalts sicher. Darüber hinaus ist er für die Aktualisierung der Übungslisten zuständig.

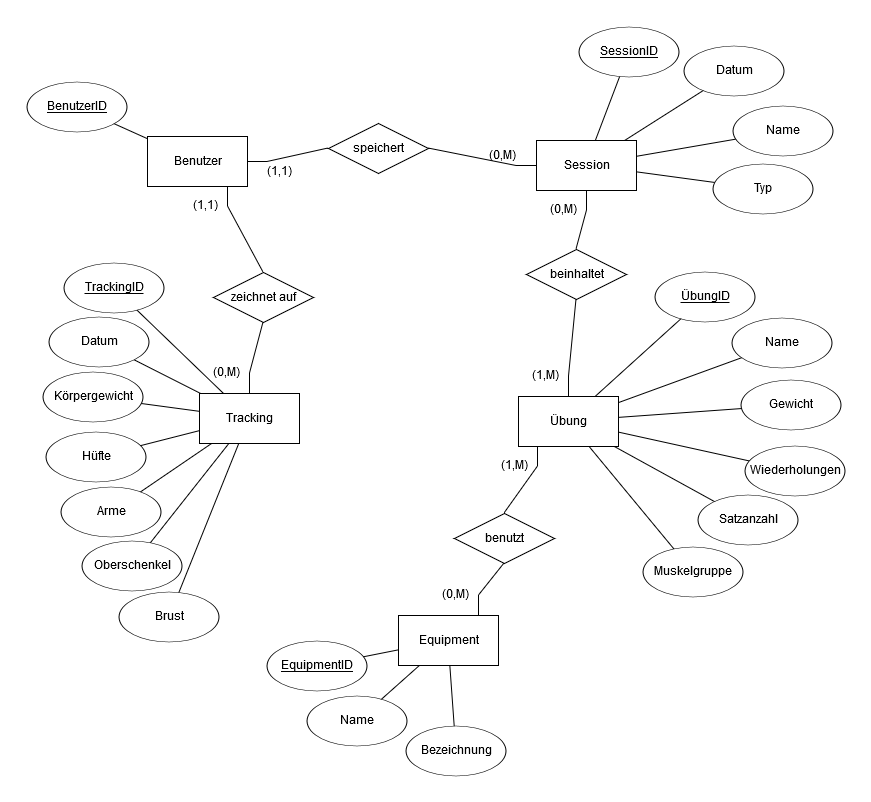
## 3.3 Entwicklung und Implementierung

### 3.3.1 Anwendungsfälle

 **Abbildung 5: Anwendungsfälle**

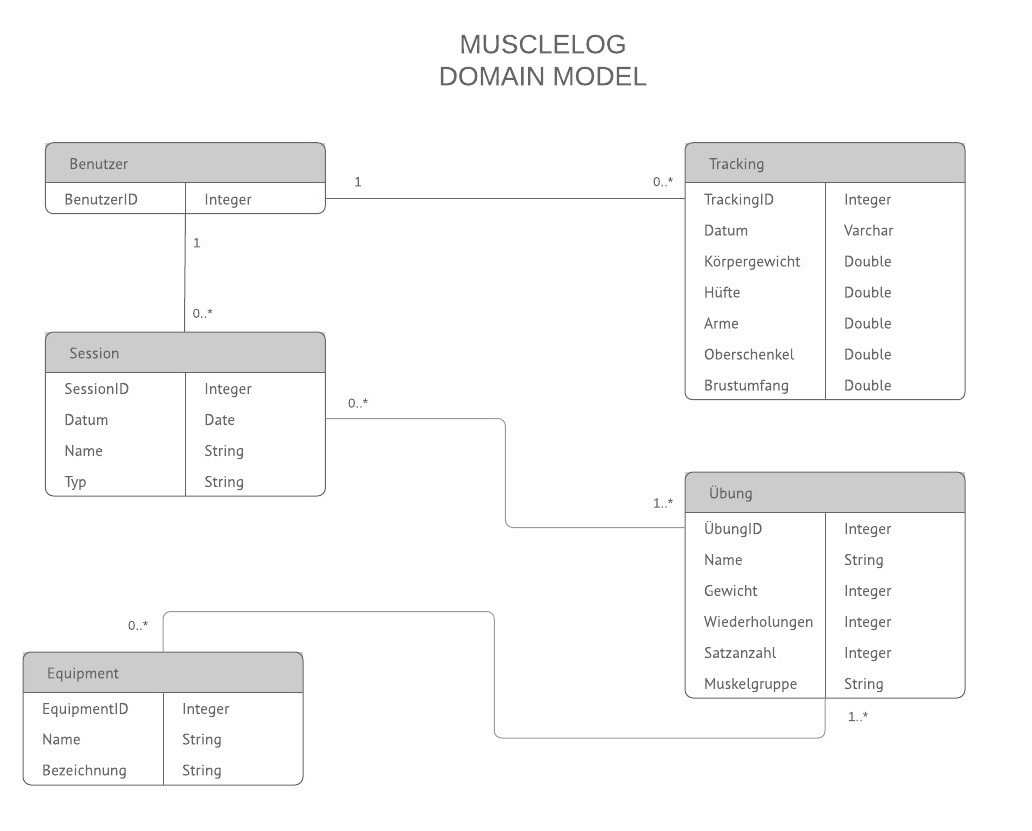
### 3.3.2 EER und UML Domänenmodell

**EER Diagramm**



**Abbildung 6: EER Diagramm**

**Domain Model**

 **Abbildung 7: Domain Model**

### 3.3.3 Funktionale und nichtfunktionale Anforderungen

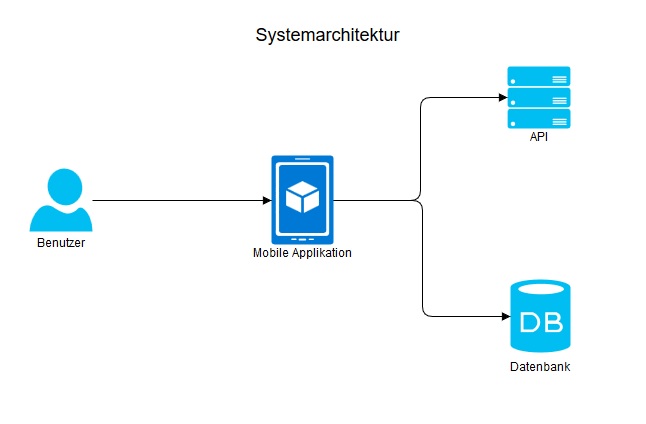
**Funktionale Anforderungen**

* Das System sollte Benutzern es ermöglichen Trainingseinheiten zu erstellen.
* Das System sollte vom Benutzer erfolgreich übermittelte Daten (Einheiten) speichern.
* Das System sollte verhindern, dass der Benutzer eine leere Einheit speichert.
* Das System sollte vom Benutzer übermittelte Daten im entsprechenden Abschnitt (History) der Anwendung anzeigen.
* Das System sollte es dem Benutzer ermöglichen, die Liste der von ihm eingereichten Einheiten zu finden.
* Das System sollte alle Details einer vergangenen Einheit anzeigen, sobald der Benutzer eine Einheit aus der Liste ausgewählt hat.
* Das System sollte eine schnelle und einfache Navigation zwischen verschiedenen Einheiten in der Liste ermöglichen.
* Das System sollte es ermöglichen, Daten von beliebigen vergangenen Einheiten zu aktualisieren.
* Das System sollte es ermöglichen, einzelne vergangene Einheiten zu löschen.

**Nichtfunktionale Anforderungen**

* Die Anwendung wird in Android Studio mit Java geschrieben und kann auf jedem Android-Gerät mit einer Android-Version 5 oder höher ausgeführt werden.
* Im Falle eines Absturzes oder eines anderen Fehlers sollte das System den Benutzer über die Probleme in Form eines Pop-ups informieren und ordnungsgemäß beenden.
* Die Anwendung sollte benutzerfreundlich und intuitiv zu bedienen sein. GUIs sowie die Navigation in der Anwendung sollten klar und einfach sein.
* Die Anwendung sollte innerhalb von 5 Sekunden nach Auswahl geöffnet und einsatzbereit sein.
* Die Benutzeroberfläche sollte schnell und flüssig sein, ohne Verzögerungen zwischen dem Drücken von Tasten und der Bildschirmreaktion.
* Alle Datenbank Lese-/Schreibvorgänge sollten nicht länger als 5 Sekunden dauern. Wenn in der Datenbank Fehler auftreten, sollte dem Benutzer eine benutzerfreundliche Warnung in Form eines Pop-ups angezeigt werden.

### 3.3.4 Systemarchitektur

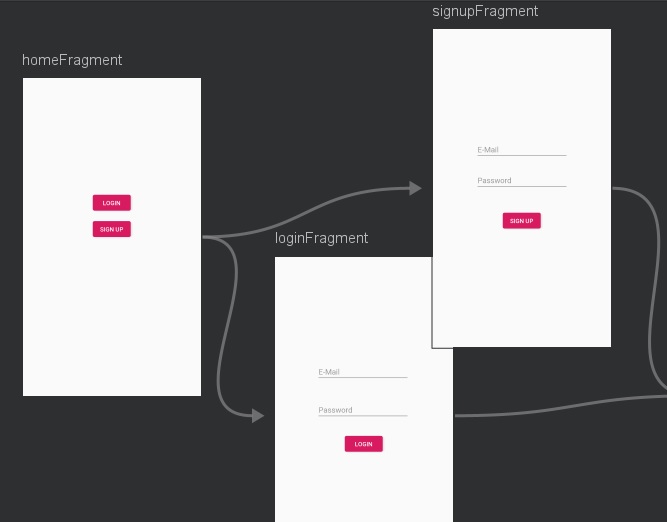
 **Abbildung 8: Systemarchitektur**

### 3.3.5 Methoden und Werkzeuge

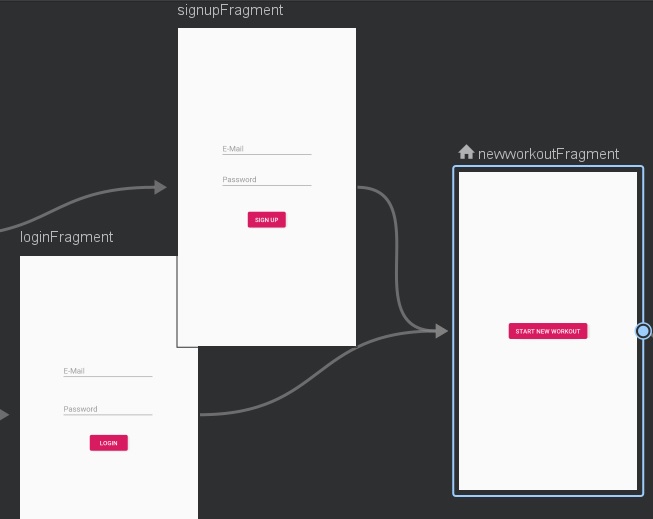
|  |  |
| --- | --- |
| **Technologie / Werkzeug** | **Beschreibung** |
| Android Studio | Basierend auf der IntelliJ IDEA-Software von JetBrains ist Android Studio eine IDE, die speziell für die Android-Entwicklung entwickelt wurde. |
| Java | Java ist die offizielle Sprache für die Android-Entwicklung und wird von Android Studio unterstützt. |
| Google Firebase | Die Firebase-Echtzeitdatenbank ist eine in der Cloud gehostete NoSQL-Datenbank, mit der man Daten zwischen den Benutzern in Echtzeit speichern und synchronisieren kann. |

### 3.3.6 Design-Layouts

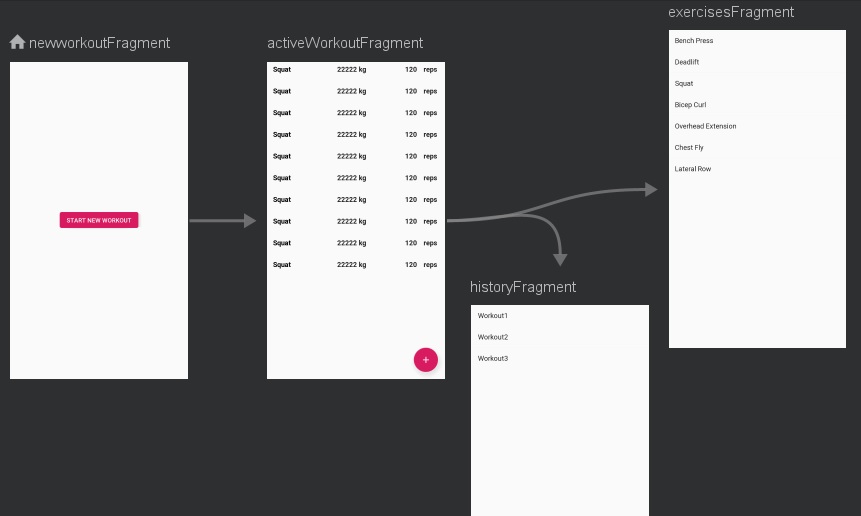
**Application Flow**

****

**Abbildung 9.1 Application Flow 1**



**Abbildung 9.2 Application Flow 2**



**Abbildung 9.3 Application Flow 3**

## 3.4 Zusammenfassung und Ausblick

# Stundenliste

|  |  |  |
| --- | --- | --- |
| 07.10.2019 | 2h 20m | Stand der Technik Analyse, Download von Apps, Eigenschaften und Funktionalitäten untersuchen |
| 09.10.2019 | 1h 30m | Lastenheft: Zielbestimmung, Produkteinsatz etc. definieren (Kapitel 1) |
| 11.10.2019 | 2h 15m | Stand der Technik: Übertragung der herausgefundenen Informationen in die Dokumentation |
| 12.10.2019 | 1h | Problemstellung und Zielsetzung definieren (Kapitel 3) |
| 13.10.2019 | 1h | Einfügen von Screenshots, fehlende Informationen ergänzen |
| 16.10.2019 | 1h 10m | Anforderungsanalyse und Konzeption |
| 18.10.2019 | 1h 35m | Funktionale und nichtfunktionale Anforderungen sowie Methoden und Werkzeuge bestimmen, Systemarchitektur |
| 19.10.2019 | 3h 40m | Design Layouts erstellen |
| 21.10.2019 | 2h 10m | Use Case Diagramm erstellen |
| 22.10.2019 | 2h 40m | EER Diagramm und Domain Model erstellen |
| 24.10.2019 | 2h 15m | Application Flow, Übergänge bestimmen |
| 27.10.2019 | 1h 25m | Übungskatalog erstellen |
| 29.10.2019 | 1h | Navgraph und zu benutzenden Fragments erstellen |
| 02.11.2019 | 1h 35m | Datenbankverbindung herstellen, Firebase Konfiguration |
| 02.11.2019 | 2h | Fragments Login und Signup implementieren |
| 04.11.2019 | 2h 15m | Erstellen der Hauptklassen Workout, Exercise, History |
| 05.11.2019 | 1h 10m | Übungskatalog Fortsetzung |
| 08.11.2019 | 3h 20m | Workout-Mode Untermenü erstellen, Debug |
| 13.11.2019 | 1h 30m | New Workout erstellen, Implementierung von Einfügen der einzelnen Übungen |

|  |  |  |
| --- | --- | --- |
| 14.11.2019 | 3h 40m | New Workout, Übergang zum History, Austausch von Objekten innerhalb der Applikation |
| 15.11.2019 | 2h 40m | Implementierung Untermenü History |
| 18.11.2019 | 2h 25m | Übungskatalog Implementieren |