

**Philosophische** Fakultät III

**Sprach-, Literatur- und Kulturwissenschaften**

**Institut für Information und** Medien**, Sprache und Kultur (I:IMSK)  
Lehrstuhl für Medieninformatik**

Einführung in die Anwendungsprogrammierung

Modul: MEI-M03.3

Sommersemester 2017

URGaining

Jannik Bikowski, Thomas Fischer, Lisa Sanladerer, Titus Tissot

1768542, …, …, …

Medieninformatik / Informationswissenschaft, Medieninformatik / Informationswissenschaft, Medieninformatik / Kunstgeschichte, Medieninformatik / Geschichte,

5. Semester B.A., 5. Semester B.A., …, 4. Semester B.A.

Jannik Bikowski

84184 Ast

Tel.: 015780466999

Email: jannik.bikowski@stud.uni-regensburg.de

Abgegeben am 24.09.2017

Inhalt

1. Kurzbeschreibung 4
2. Technische Voraussetzungen 5
3. Problemstellung und Lösungsvorschlag 6
4. Design, Implementierung & User Interface 8
5. Testing 11
6. Finaler Zustand und Ausblick 12
7. Projektmanagement 13

# Kurzbeschreibung

URGaining ermöglicht dem Nutzer ein übersichtliches und intuitives Trainings- und Workout-Management. Die leichte Bedienung und das vertraute Interface helfen fitnessbegeisterten

Nutzern, ihre Erfolge beim Training zu speichern und gibt mit entsprechenden Tabellen stets einen guten Überblick über absolvierte Workouts. Neue Trainingeinheiten mit zugehörigen Übungen sind schnell und einfach erstellt und werden für individuelle Statistiken auf einer lokalen Datenbank gespeichert. Mit der Facebook-Einbindung lassen sich Trainingserfolge mit Freunden teilen.

# Technische Vorrausetzungen

*URGaining* ist eine speziell für mobile Android-Geräte entwickelte Anwendung, die ohne eigene Server-Infrastruktur funktioniert. Für den Betrieb der Anwendung reicht eine (manuelle) Installation der Anwendung auf einem Android Gerät aus. Unterstützt werden ausschließlich Android-Smartphones, die mindestens die Version 4.4 des Android-Systems nutzen.

Um auf die Sharing-Funktion nutzen zu können, ist sowohl ein Facebook-Account, die neuste Version der offiziellen Facebook App, als auch eine Internetverbindung notwendig.

Benutzte Bibliotheken sind zum einen „android.support.v7“ um Rückwärtkompatibilität für ältere Android Versionen zu bieten, sowie „android.support.percent.PercentRelativeLayout“ um ein adaptives Layout zu gewährleisten. Außerdem wurde die „com.facebook.share“ Bibliothek verwendet um Trainingserfolge auf Facebook zu teilen und „com.github.mikephil.charting“ um Trainingsdaten in Form von Graphen zu visualisieren.

# Problemstellung und Lösungsvorschlag

Oft sieht man im Fitnessstudio Trainingsenthusiasten mit Notizblock und Stift umherlaufen, damit sie ihre Fortschritte genauestens dokumentieren können. Doch da in der heutigen Zeit beinahe jeder ein eigenes Smartphone besitzt, bietet es sich an diese Fortschritte innerhalb einer App festzuhalten.

Mit URGaining hat der Nutzer während des Trainings stets eine Übersicht über seine Leistungen aus dem vergangenen Training und hat so einen Anreiz diese bei seinem aktuellen Workout zu übertreffen. Sollte ihm das Workout einmal zu schwer fallen kann er sich mit der Deload Funktion über einen vorher festgelegten Wert ein passendes Arbeitsgewicht für die einzelnen Übungen errechnen lassen. Dies erspart im Zeit, die er sonst in externen Apps aufwenden würde.

Außerdem bietet die App dem Nutzer eine Übersicht über seinen Gesamtfortschritt der einzelnen Übungen, gemessen an der errechneten Maximalkraft, in Form einer Grafik. Sollte der Nutzer eine befriedigende Steigerung erzielt haben, kann er diese Grafik direkt mit seinen Freunden auf Facebook teilen.

Desweiteren hat der Nutzer die Möglichkeit eine oder mehrere Übungen in die Berechnung seiner kombinierten Maximalkraft miteinzubeziehen. So haben z.B. Powerlifter auf der Startseite eine Übersicht über ihre Maximalkraft bestehend aus den Disziplinen „Kniebeugen“, „Bankdrücken“ und „Kreuzheben“, oder bei Gewichthebern „Reißen“ und „Stoßen“.

# Design, Implementierung & User Interface

Die Anwendung besteht aus sieben Activities. Der Einstiegspunkt der App ist die Fortschrittübersicht (ProgressActivity) von der aus der Nutzer einzelne eingespeicherte Übungen anwählen kann, um in einer Grafik eine Übersicht über seine Fortschritte zu erhalten. Unterhalb der Grafik befindet sich ein Facebook-Share-Button, mit dem der Nutzer die Grafik auf Facebook teilen kann. Ist keine Übung ausgewählt tritt an Stelle der Übersichtsgrafik eine Textanzeige, die die kombinierte Maximalkraft, der vorher festgelegten Übungen, des Nutzers anzeigt. Übungen werden der kombinierten Maximalkraftberechnung über einen onItemLongClickListener hinzugefügt.

Über die untere Navigationsleiste gelangt der Nutzer in die Workoutübersicht (WorkoutsActivity) von hier aus hat er die Möglichkeit ein zuvor eingespeichertes Workout auszuwählen oder über den FloatingActionButton ein neues Workout hinzuzufügen (AddWorkoutPopup). Der Name des erstellten Workouts wird in der Datenbank gespeichert und im ListView der Workoutübersicht dargestellt. Über einen onItemLongClickListener kann der Nutzer gespeicherte Workouts wieder löschen, dabei werden ebenso alle Übungen und gespeicherte Daten der Übungen aus der Datenbank gelöscht.

Klick der Nutzer kurz auf ein Workout wird er zur Übungsübersicht (ExercisesActivity) weitergeleitet, dabei wird der Name des Workouts dem Intent als Extra übergeben, damit das ListView der Übungsübersicht mit den entsprechenden Übungen befüllt werden kann. Auch hier können Übungen über einen langen Klick darauf wieder gelöscht werden. Außerdem befinden sich direkt über der unteren Navigationsleiste zwei Buttons. Mit dem „Reset Workout“-Button werden die gespeicherten Daten aller Übungen dieses Workouts, mit Ausnahme der jüngst gespeicherten, aus der Datenbank gelöscht. Durch einen Klick auf den FloatingActionButton, rechts des Reset-Buttons öffnet sich ein Popup-Fenster (AddExercisePopup). Im EditText kann der Nutzer nun ein Namen für die Übung, sowie über einen Spinner die Anzahl der Sätze für diese Übung angeben. Klickt er nun auf den Create-Button, wird ein neues Exercise-Objekt (bestehend aus: Name der Übung, Anzahl der Sätze, Name des zugehörigen Workouts) erzeugt und in der Datenbank gespeichert. Der Name dieser Übung wird nun dem ListView in der Trainingsübersicht hinzugefügt.

Durch Klick auf eine Übung in der Trainingsübersicht öffnet sich eine Detailansicht für diese Übung (ExerciseDetails). Hier er eine Übersicht über jeden Satz dieser Übung aus dem letzten Training, d.h. wieviel Gewicht er in jedem Satz verwendet hat und wieviele Wiederholungen er damit geschafft hat. Direkt darüber befinden sich einige EditTexts, um Daten aus dem aktuellen Training einzutragen. Trägt der Nutzer ein Gewicht für z.B. den dritten Satz ein, wird diese Zahl für alle späteren Sätze automatisch übernommen, um Tipparbeit zu sparen. Über den „Deload“-Button wird, anhand einem in den Einstellungen festgelegten Wert und dem verwendeten Gewicht der ersten Satzes aus dem letzten Training, ein neues Arbeitsgewicht errechnet. Der Nutzer kann die Eingabe abbrechen, indem er an eine beliebige Stelle außerhalb des Popups klickt, oder die Eingabe bestätigen, indem er auf den Finish-Button oben rechts klickt. Dabei werden alle eingetragenen Werte einer ArrayList hinzugefügt und in der Datenbank gespeichert.

Klickt der Nutzer in der unteren Navigationsleiste auf Settings, gelangt er in die Einstellungen (SettingsActivity). Hier hat er die Möglichkeit einzustellen, um welchen Prozentbetrag das Arbeitsgewicht über den Deload-Button reduziert werden soll, sowie wieviele Trainingseinheiten rückwirkend in der Datenbank gespeichert bleiben sollen. Diese Daten werden mit Hilfe von SharedPreferences persistent gespeichert. Ganz oben im Einstellungsmenü befindet sich zudem ein Facebook-Login-Button.

# Testing

Beschreiben Sie die wesentlichen Probleme, die durch den Usability-Test aufgedeckt wurden. Erläutern Sie kurz, welche Maßnahmen ergriffen wurden, um diese Problem zu beheben und geben Sie an, im welchem Umfang die identifizierten Schwachstellen ausgebessert wurden. Beschreiben Sie, wie Sie Ihre Anwendung angepasst haben, um die identifizierten Problem zu lösen. Dabei sollten Sie erwähnen welche Komponenten oder Klassen Ihren App betroffen sind und wie sich die Änderungen auf Ihr ursprüngliches (Interaktions-)Konzept ausgewirkt haben.

# Finaler Zustand und Ausblick

Im finalen Zustand der App wurden alle festgelegten Kernfeatures wie in den Anforderungen beschrieben, bzw. leicht abgewandelt implementiert. Trainingseinheiten mit den zugehörigen Übungen und deren Daten werden in einer Datenbank persisten gespeichert. Und der Nutzer hat auf der Startseite eine Übersicht über seine Trainingsfortschritte. Er erhält zwar wie anfangs beschrieben nicht regelmäßig eine Übersicht, allerdings kann er diese zu jedem Zeitpunkt aus Eigeninitiative einsehen. In den Einstellungen kann zudem festgelegt werden wieviele Trainingseinheiten rückwirkend gespeichert werden und um welchen Prozentbetrag das Arbeitsgewicht der Übungen beim Deload reduziert werden soll. Außerdem wird in der „ProgressActivity“ (Startseite) beim Starten der App die kombinierte Maximalkraft des Nutzers angezeigt, dies ist ebenso der Fall, wenn momentan keine Übung zur grafischen Visualisierung ausgewählt ist. Außerdem lässt sich die Fortschrittsgrafik einer Übung in der „ProgressActivity“ über Facebook teilen.

Die App könnte noch durch eine Stoppuhr Funktion in der Übungsdetailansicht erweitert werden, damit der Nutzer nach dem Eintragen der Werte für den aktuellen Satz die Pausezeiten bis zum nächsten Satz messen kann. Außerdem könnten alle Activity als Fragments realisiert werden, um einen schöneren Übergang zwischen den Tabs der unteren Navigationsleiste zu erzielen.

# Projektmanagement

Beschreiben Sie kurz, wie Sie bezüglich der gruppeninternen Abstimmung vorgegangen sind. Erwähnen Sie Tools und Kommunikationswege, die Sie für die Gruppenarbeit eingesetzt haben. Sie können hier auch einen kurzen Überblick über die Arbeitsaufteilung geben und explizit erwähnen, welche Bestandteile von welchen Teammitgliedern umgesetzt wurden.