



HTL - Perg
Höhere Abteilung für Informatik

Diplomarbeit

EasyGym



Projektteam: Paul Heimel
David Hennerbichler
Projektbetreuer: Prof. Dipl.-Ing. Christian Aberger

Bearbeitungszeitraum: 12.05.2016 – 05.04.2017



1 Eidesstattliche Erklärung

Im Zuge dieser eidesstattlichen Erklärung versichern wir, Paul Heime! und David Hennerbichler, dass diese Diplomarbeit mit dem Titel

„EasyGym“

selbstständig verfasst und ohne fremde Hilfe sowie ohne Benutzung anderer als der angegebenen Quellen angefertigt wurde.

Jene Teilbereiche, die anderen Werken beziehungsweise Internetpublikationen wortwörtlich oder dem Sinn gemäß entnommen wurden, haben wir in jedem einzelnen Fall durch eine Quellenangabe als solche kenntlich gemacht.

Perg, am 04.04.2017

Paul Heime!

David Hennerbichler



2 Gendererklärung

Für die bessere Lesbarkeit werden in dieser Diplomarbeit personenbezogene Bezeichnungen, die sich zugleich auf Frauen und Männer beziehen, generell nur in der im Deutschen üblichen maskulinen Form angeführt.

Dies soll jedoch keinesfalls eine Geschlechterdiskriminierung oder eine Verletzung des Gleichheitsgrundsatzes zum Ausdruck bringen.

Paul HeimeI

David Hennerbichler



EasyGym

Von Paul Heimel, David Hennerbichler

Diplomarbeit im Fachgebiet Projektentwicklung

HTBLA-Perg

3 Zusammenfassung

EasyGym wurde im Mai 2016 in Zusammenarbeit mit dem Verein ASKÖ-Linz gestartet. Im Verlauf der Entwicklung trennten sich jedoch ASKÖ-Linz und das Projektteam aufgrund von Zeitproblemen, sowie Passivität und Inaktivität des Vereins.

Das Projekt wurde entwickelt, um End-Nutzern Fitnessinformationen so flexibel, dynamisch und nachhaltig wie nur möglich anbieten zu können. Die Kunden sind hierbei Dienstleister, wie zum Beispiel Unternehmer oder Physiotherapeuten, die EasyGym mit Inhalten füllen und schlussendlich an ihre eigenen Kunden vertreiben.

Eine JSF-Web-Applikation sowie eine Android-Applikation bilden die beiden Kernstücke der Diplomarbeit. Die Management-Webseite wird von den Entwicklern und Dienstleistern, welche dieses Produkt erwerben, als Administrationsoberfläche verwendet. Auf dieser werden Fitnesspakete und -übungen individuell gestaltet und angelegt. Die Android-Applikation zeigt dem Endnutzer die für ihn abrufbaren Inhalte an.

Das Umsetzen einer zukunftssträchtigen Geschäftsidee sowie die Anwendung erlernter Fähigkeiten im IT-Bereiche sind die Ziele dieser Diplomarbeit.



EasyGym

From Paul Heimel, David Hennerbichler

Diploma thesis in Projectmanagement

College of engineering specialized in informatics – Perg

4 Abstract

EasyGym was created in cooperation with ASKO Linz in May 2016. However, due to issues such as timing, lack of buy-in, and inactivity from the partner organisation the project team decided to pursue the completion independently.

The project was developed to offer flexible, dynamic, and sustainable fitness information to endusers. The content is generated by 3rd-party-providers such as fitness companies or physiotherapists, who can use the app to distribute this information to their clients.

At the heart of this project sits a JSF web application and an Android app. The management side (JSF) is to be used by developers and customers who have purchased EasyGym as an admin tool to create and publish fitness packages and exercises.

The Android application is the interface on which this content is then available for the enduser.

The aim of this project was to practically apply previously gained IT skills as well as developing a futureproof business case.



5 Impressum

Schule:

HTL Perg für Informatik

Schuljahr:

2016/2017

Klasse:

5AHIF

Projekttitle:

EasyGym

Projektteam:

Paul HeimeI

David Hennerbichler

Beratungslehrer:

Herr Prof. Dipl.-Ing. Christian Aberger



6 Danksagung

An dieser Stelle möchten wir uns bei jenen Personen bedanken, die uns im Verlauf der Erstellung der Diplomarbeit in jeglicher Hinsicht unterstützt haben.

Ein besonderer Dank gilt unserer Betreuungslehrkraft, Herrn Prof. Dipl.-Ing. Christian Aberger, der uns bei der Entwicklung mit hilfreichen Tipps und stets positiver Energie zur Seite gestanden ist.

Ein weiteres Danke geht an unsere Kollegen, die uns durch Empfehlungen, Ratschläge und individuelles Wissen bei der Entwicklung und Problembeseitigung geholfen haben. Deziert bedanken wir uns bei Michael Leonhartsberger und Tobias Friedinger für die Unterstützung bei der Web- und Android-Entwicklung.

Danke auch an Frau Magistra Prof. Danner für die Hilfe bei der Kontaktaufnahme zwischen ASKÖ-Linz und dem Projektteam.

Vielen Dank auch an Familie und Freunde, die während dieser stressigen Zeit Rücksicht auf uns genommen und uns von der einen oder anderen Arbeit befreit haben.



7 Inhaltsverzeichnis

1 Eidesstattliche Erklärung	2
2 Zusammenfassung	4
3 Abstract	5
4 Impressum.....	6
5 Danksagung.....	7
6 Inhaltsverzeichnis.....	8
7 Projektumfeld	11
7.1 Team	12
7.1.1 Paul Heimel BILD	12
7.1.2 David Hennerbichler BILD.....	13
7.2 Betreuungslehrkraft	14
7.2.1 Professor Dipl.-Ing. Christian Aberger	14
7.3 Ausbildungseinrichtung.....	14
7.3.1 HTL Perg, Höhere Abteilung für Informatik.....	14
8 Einleitung	16
8.1 Motivation.....	16
8.2 Ziel der Arbeit.....	16
8.3 Technische Ziele	17
8.4 Allgemeine Ziele.....	17
9 Grundlagen.....	18
9.1 Entwicklungssysteme	18
9.1.1 Android-Studio	18
9.1.2 Eclipse	18
9.1.3 Notepad ++.....	19
9.1.4 Postman	19
9.1.5 FileZilla	20
9.1.6 MySQL Workbench.....	20
9.1.7 Firefox	21
9.1.8 Kommandozeile.....	21
9.1.9 Putty	22
9.1.10 SourceTree	22
9.2 Sonstige verwendete Software	23
9.2.1 Microsoft Office	23
9.2.2 GIMP 2.....	24
9.3 Verwendete Technologien	25



9.3.1 Java.....	25
9.3.2 Java Persistence API (JPA)	25
9.3.3 Java Server Faces (JSF)	26
9.3.4 Wildfly	26
9.3.5 HTML (Hypertext Markup Language)	26
9.3.6 CSS (Cascading Style Sheet).....	27
9.3.7 AJAX (Asynchronous JavaScript and XML).....	27
9.3.8 SQL (structured query language).....	27
9.3.9 Android.....	28
9.4 Verwendete Bibliotheken	29
9.4.1 PrimeFaces	29
9.4.2 Rocket	29
9.4.3 Gson	29
9.4.4 Volley.....	30
9.1 Verwendete Architekturen	31
9.1.1 REST.....	31
9.1.2 DAO	33
10 Realisierung.....	34
10.1 Vorgehensweise	34
10.1.1 SCRUM	34
10.2 Webanwendung.....	37
10.2.1 Grundlegende Ausgangslage.....	37
10.2.2 Layout.....	37
10.2.3 Umsetzung Management-Webseite	38
a) Entwickler-Zugang.....	39
b) Mandanten-Zugang.....	40
10.2.4 Besonderheiten Management-Webseite.....	50
10.2.5 Umsetzung REST-Schnittstelle.....	52
10.3 Android-Applikation	55
10.3.1 Idee	55
10.3.2 Erklärungen	56
10.3.3 Umsetzung	59
10.3.4 Ergebnisse	62
10.4 Datenbank	72
10.4.1 Datenlexikon	73
10.5 Probleme	78



11 Resümee.....	79
12 Aufgabenverteilung.....	80
12.1 Paul Heimel	80
12.2 David Hennerbichler	81
13 Glossar.....	82
14 Anhang	88
14.1 Projektstrukturplan	88
14.2 Meilensteine	89
14.3 Zeitlicher Ablauf	90
14.3.1 Paul Heimel	90
14.3.2 David Hennerbichler.....	93
15.1 Risikomanagement.....	96
15.1.1 Risiken	96
15.1.2 Eingetretene Risiken	100
15.1.3 Fazit aus den Problemen	101
15.2 Literaturverzeichnis.....	102
15.3 Abbildungsverzeichnis.....	106
15.4 Tabellenverzeichnis.....	108

8 Projektumfeld

Im Rahmen dieser Arbeit gab es aufgrund der Trennung vom Kooperationspartner ASKÖ zwei unterschiedliche Situation, was die am Projekt beteiligten Personen betrifft. Die Zusammenarbeit mit dem Verein ASKÖ, welche im Mai 2016 gestartet wurde, ist im Oktober 2016 aus Zeitgründen, Terminproblemen und Passivität von Seiten des ASKÖ-Linz wieder auf Eis gelegt worden.

Projektbeteiligte Mai 2016 – Oktober 2016:

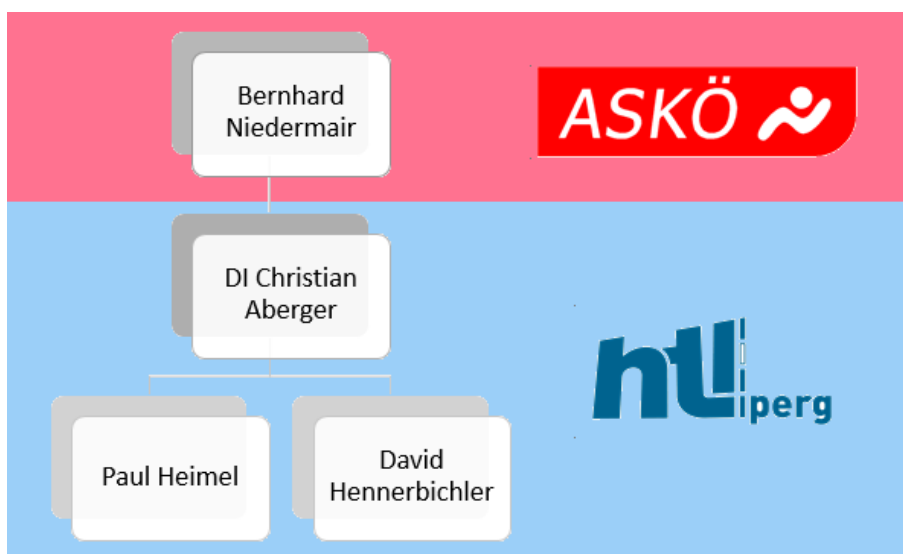


Abbildung 1: Projektumfeld Mai 2016 (htl-perg.ac.at, 2017) (askoe-linz-steg.at, 2017)

Projektbeteiligte Oktober 2016 – April 2017:

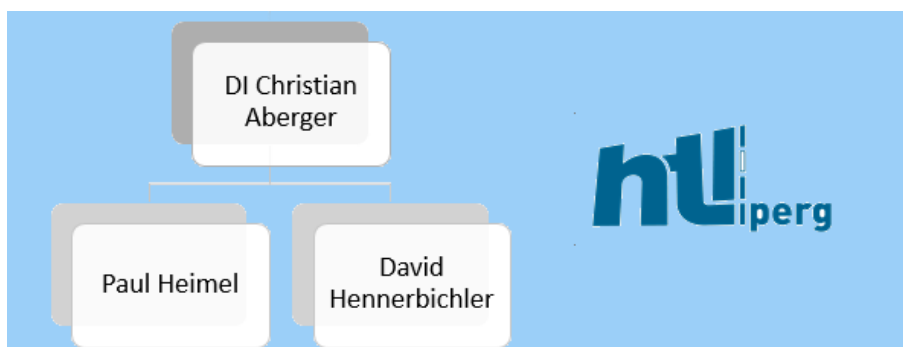


Abbildung 2: Projektumfeld Oktober 2016 (htl-perg.ac.at, 2017)

8.1 Team

8.1.1 Paul Heimel

Persönliche Daten:

Geburtsdaten:	Amstetten, 08. Juni 1998
Staatsbürgerschaft:	Österreich
Religion:	röm.-kath.
Familienstand:	ledig
Eltern:	Thomas Stöckl Gertraud Heimel
Geschwister:	Caroline Heimel Nina Heimel



Abbildung 3: Paul Heimel

Schulbildung:

2004 – 2008	Volksschule in Grein
2008 – 2012	Musikhauptschule in Saxen
2012 – 2017	HTL Perg, Höhere Abteilung für Informatik

Arbeitserfahrung:

Sommer 2013	Habau Hoch- und Tiefbaugesellschaft mbH (Baustelle)
Sommer 2014	SmilingWords Software (Softwareentwicklung)
Sommer 2015	freeWOHL GmbH (Büroorganisation und Rechnungswesen)
Sommer 2016	Uni Software Plus GmbH (Softwareentwicklung)

Kontakt:

Telefon	+43664 19 66 086
E-Mail	paul.heimel.12@gmail.com

8.1.2 David Hennerbichler

Persönliche Daten:

Geburtsdaten:	Amstetten, 17. März 1998
Staatsbürgerschaft:	Österreich
Religion:	röm.-kath.
Familienstand:	ledig
Eltern:	Johannes Hennerbichler Anita Hennerbichler
Geschwister:	Christina Hennerbichler



Abbildung 4: David Hennerbichler

Schulbildung:

2004 – 2008	Volksschule in Saxen
2008 – 2012	Musikhauptschule in Saxen
2012 – 2017	HTL Perg, Höhere Abteilung für Informatik

Arbeitserfahrung:

Sommer 2013	Habau Hoch- und Tiefbaugesellschaft mbH (Baustelle)
Sommer 2014	Habau Hoch- und Tiefbaugesellschaft mbH (IT-Abteilung)
Sommer 2015	Bezirkshauptmannschaft Perg (Verkehrsabteilung)
Sommer 2016	Hueck Folien (Fadenschneiderei)

Kontakt:

Telefon	+43664 305 65 65
E-Mail	hennerbichler.d@gmail.com

8.2 Betreuungslehrkraft

8.2.1 Professor Dipl.-Ing. Christian Aberger

Kontakt:

Aberger Software GmbH
Softwarepark 37 4232 Hagenberg
+43 720 348 450
office@aberge.at
www.aberge.at



Abbildung 5: Dipl.-Ing. Christian Aberger (aberge.at, 2017)

Unsere Diplomarbeit wurde von Herrn Prof. Dipl.-Ing. Christian Aberger betreut, der uns ab der 4. Klasse in Programmieren unterrichtete. Viele der in EasyGym verwendeten Technologien waren in seinem Lehrstoff enthalten und deshalb konnten wir uns bei Fragen jederzeit an ihn wenden.

Seinen Hauptberuf übt er jedoch nicht als Lehrer aus, sondern als Geschäftsführer der Firma „Aberger Software GmbH“ mit Sitz in Hagenberg.

8.3 Ausbildungseinrichtung

8.3.1 HTL Perg, Höhere Abteilung für Informatik

Kontakt:

HTBLA Perg
Machlandstraße 48
4320 Perg
0 72 62 539 26
office@htl-perg.ac.at
www.htl-perg.ac.at



Abbildung 6: HTL-Perg (fcp.at, 2017)

Die HTL Perg wurde 1999/2000 als Expositur der HTL Leonding gegründet. Mit dem Neubau des Schulgebäudes 2007 gab es neben der fünfjährigen Höheren Abteilung erstmals auch die Möglichkeit, eine dreieinhalbjährige Fachschule zu absolvieren.

In der höheren Abteilung liegt der Schwerpunkt im Bereich der Softwareentwicklung, wobei betriebswirtschaftliche Aspekte und Projektentwicklung ebenso eine besondere Berücksichtigung finden.



Diese Kombination beinhaltet eine umfangreiche, praxisnahe Ausbildung, welche später einerseits im Beruf direkt umgesetzt werden kann und andererseits auch die Möglichkeit zu einem Universitäts- beziehungsweise Fachhochschulstudium bietet.

Der Gegenstand Projektentwicklung und die Diplomarbeit bieten die Möglichkeit, mit Unternehmen zu kooperieren und gewähren den Schülern dadurch schon während der Schulzeit einen Einblick in das spätere Berufsleben. (vgl. HTL-Perg, 2017)

Die Fachschule für Informationstechnik bietet im Vergleich zur höheren Abteilung eine praxisorientiertere Ausbildung. Neben der Allgemeinbildung sowie der fachtheoretischen Ausbildung in der IT wird auch ein praktischer Unterricht in Werkstätten und Labors durchgeführt. Durch ein zwölfwöchiges Betriebspraktikum im letzten Schuljahr bekommen die Absolventen eine ideale Vorbereitung auf den Berufseinstieg. (vgl. HTL-Perg, 2017)

9 Einleitung

Dieses Kapitel enthält Ziele und Motivation des Projektteams.

9.1 Motivation

Das Interesse an der körperlichen Fitness sowie die Nicht-Zufriedenstellung durch diesbezüglich am Markt erhältliche Produkte waren für das Projektteam eine große Motivationsquelle, eine dynamische Lösung, welche es erlaubt, Fitnessinformationen individuell zu erstellen, zu entwickeln. EasyGym bietet eine flexible Lösung, die es verschiedensten Dienstleistern ermöglicht, eigene Information zu verpacken und diese am Markt anzubieten.

Darüber hinaus wird großes Augenmerk auf eine einfache Handhabung für Dienstleister und Endnutzer sowie die Möglichkeit der eigenständigen Einteilung von Fitnessseinheiten durch die Endnutzer gelegt.

9.2 Ziel der Arbeit

Das Ziel dieser Arbeit liegt darin, ein flexibel einsetzbares System zum Erstellen, Anzeigen und Anbieten von Fitnessinformationen zu entwickeln. Mandanten sollen über eine Web-Applikation die Möglichkeit haben, ihre Daten über Formulare in ein vordefiniertes Gerüst zu legen und dadurch Inhalte in einem einheitlichen Format für die Endnutzer abrufbar zu machen. Je nach Bedarf sollte dieses System erweiterbar sein, um eventuell in der Zukunft auch neue Funktionen hinzufügen zu können.



9.3 Technische Ziele

Die Verwendung von EasyGym sollte für alle Benutzer (Mandanten und Endnutzer) so angenehm wie möglich gestaltet sein. Da die Mandanten hauptsächlich administrative Arbeiten zu verrichten haben, wurde die Administrationsseite als einfach gestaltete Webanwendung entwickelt, welche es ermöglicht, Daten vom Server abzurufen und zu speichern. Deshalb wurde absichtlich auf komplizierte Animationen und Verschachtelungen verzichtet, um für eine leicht verständliche Bedienung zu sorgen.

Für den Client wurde das gängigste Mobile-Applikationsformat Android gewählt, das unter der Voraussetzung eines Internetzugangs vom Benutzer jederzeit und überall verwendet werden kann. Auch hier wurde der Aufbau einfach und verständlich gestaltet, um die Anwendung ohne Tutorial bedienen zu können.

9.4 Allgemeine Ziele

Das allgemeine Ziel von EasyGym lag unter anderem darin, Erfahrungen im Bereich Projektentwicklung zu erlangen. Darunter fällt, den Ablauf eines Projektes von Beginn an durchzuplanen und sich anhand der erstellten Zeitvorgaben schrittweise zur Projektfertigstellung vorzuarbeiten. In diesem Zeitraum wurde die Kommunikationsfähigkeit stark geschult, da man sich mit seinem Projektteam, dem Projektbetreuer und auch dem Auftraggeber absprechen musste. Durch die intensive Arbeit im jeweiligen Aufgabenbereich wurden wertvolle Erkenntnisse gesammelt, die im regulären Unterricht kaum möglich wären.

10 Grundlagen

Dieses Kapitel enthält Informationen zu den verwendeten Entwicklungssystemen, Technologien, Bibliotheken und Architekturen.

10.1 Entwicklungssysteme

10.1.1 Android-Studio

Android Studio ist eine kostenlose Entwicklungsumgebung von Google und wurde im Dezember 2014 mit Version 1.0 auf den Markt gebracht. Es basiert auf IntelliJ und zeichnet sich als die offizielle Entwicklungsumgebung für Android aus. Seither werden regelmäßig Updates veröffentlicht, welche Verbesserungen in Bezug auf die Performance sowie Erneuerungen enthalten. Durch einen optionalen Emulator ist ein Entwickler nicht an das eigene Smartphone gebunden, sondern kann Android-Applikation in der Entwicklungsumgebung austesten. (vgl. wikipedia.org, 2017)

Bei dieser Arbeit wurde Android-Studio zur Entwicklung der mobilen Android-Applikation verwendet.



Abbildung 7: Android Studio
(wikipedia.org, 2017)

10.1.2 Eclipse

Eclipse ist ein kostenloses Open-Source Programmierwerkzeug zur Entwicklung von Software verschiedenster Art. Durch die eigenständige „Eclipse Foundation“ kam das Produkt im Februar 2004 unter der Version 3.0 auf den Markt, zuvor ist es von IBM entwickelt worden. Ursprünglich wurde es nur als Entwicklungsumgebung für Java genutzt, es ist jedoch aufgrund seiner Erweiterbarkeit im Laufe der Zeit immer öfter für andere Entwicklungsaufgaben eingesetzt worden. Diese umfangreiche Erweiterbarkeit ermöglicht es dem Entwickler, gleichzeitig Web-Applikation, einfache Web Sites, Android Applikationen, einen Java-Server und vieles mehr zu programmieren. (vgl. wikipedia.org, 2017)

In dieser Arbeit wurde Eclipse zur Konfiguration der Web-API sowie zur Entwicklung der Administrationsoberfläche verwendet.



Abbildung 8: Eclipse (eclipse.org, 2017)

10.1.3 Notepad ++

Notepad++ ist ein kostenloser Texteditor. Im Gegensatz zu anderen Editoren, wie dem Standard-Texteditor von Windows, gehört er in die Kategorie der komplexeren Programmier-Editoren. Das wahre Potenzial kommt zum Vorschein, wenn es an die Programmierung geht. Die Software unterstützt zahlreiche Sprachen, wie zum Beispiel C, C++, Java, HTML oder PHP und bietet speziell für jede Sprache Syntax-Highlighting an, durch das Programmierer durch farbige Hervorhebung den Überblick behalten können. (vgl. Don HO, 2017)

Weitere Funktionen, wie eine Auto-Vervollständigung oder die Multi-Dokumenten Ansicht, machen Notepad++ zu einem angenehmen Entwicklertool. (vgl. wikipedia.org, 2017)

In dieser Arbeit wurde Notepad++ als Editor diverser textbasierender Dateien verwendet.



Abbildung 9: Notepad++
(wikipedia.org, 2017)

10.1.4 Postman

Postman ist eine plattformunabhängige, kostenlose Software zum Testen von Programmierschnittstellen. Die Besonderheit liegt darin, dass alle Aspekte der Schnittstellen-Programmierung berücksichtigt werden. Man kann also neben dem Wählen eines Request-Typs, wie zum Beispiel GET, PUT, POST oder DELETE, auch speziell formulierte Header oder Body-Elemente definieren. (vgl. Postdot Technologies, 2017)

In dieser Arbeit wurde Postman zum Testen verschiedener Rest-Requests des Servers verwendet, jedoch auch um aktuelle Daten in der Datenbank sowohl lokal, als auch auf dem Schulserver abzufragen.



Abbildung 10: Postman (getpostman.com,
2017)

10.1.5 FileZilla

FileZilla ist eine kostenlose Server- und Client-Software zur Dateiübertragung über FTP und SFTP. Über den FileZilla-Client ist es möglich, sich mit einem FTP-/SFTP-Server zu verbinden und in den Server-Verzeichnissen Dateien hoch- und herunterzuladen. (vgl. wikipedia.org, 2017)

Das Projektteam von EasyGym verwendete FileZilla, um Testdaten wie Bilder auf den Server zu laden, Verzeichnispfade zu erstellen oder zu kontrollieren, ob File-Uploads von der Management-Webseite funktionieren.



Abbildung 11: FileZilla
(wikipedia.org, 2017)

10.1.6 MySQL Workbench

MySQL Workbench ist in der Community Edition als kostenlose Software erhältlich. Eine erste Beta-Version gab es ab 2007. Es dient als Datenbank-Modellierungswerkzeug, das zusätzlich zur Datenmodellierung viele weitere Funktionen kombiniert. Der Einsatz dieser Software ist nur für MySQL-Datenbanken möglich. Die Hauptfunktion liegt im Instandhalten einer Datenbank durch Möglichkeiten wie zum Beispiel das Ändern eines Datenbankmodells oder das Einbinden einer bereits bestehenden MySQL Datenbank. (vgl. wikipedia.org, 2017)

In dieser Arbeit wurde die MySQL Workbench zum Erstellen, Anzeigen und Verwalten des Datenmodells verwendet. Außerdem wurden die Testdaten über die MySQL Workbench eingefügt, um sie bei Bedarf erneut auf den Server spielen zu können.



Abbildung 12: MySQL-Workbench
(quintetsolutions.com, 2017)

10.1.7 Firefox

Firefox ist ein kostenloser Web-Browser und wurde von der Mozilla Corporation entwickelt. Erstmals erhältlich war er im September 2002. Neben den Standardfunktionen eines Web-Browsers bietet Firefox die Möglichkeit, durch verschiedene Add-On's, die im Store erhältlich sind, den Web-Browser für einen Benutzer passend und einzigartig zu gestalten. Weiters gibt es Funktionalitäten, wie zum Beispiel eine Entwicklerkonsole, mit der man während der Entwicklung Inhalte einer Website ansehen oder Fehler in der Syntax des Quellcodes erkennen kann. (vgl. wikipedia.org, 2017)

In dieser Arbeit wurde Firefox zum Testen der Administrationsoberfläche verwendet, da verschiedene Browser nicht immer das gleiche Ergebnis liefern.



Abbildung 13: Firefox
(wikipedia.org, 2017)

10.1.8 Kommandozeile

Die Kommandozeile ist ein Eingabebereich und wird oft auch als Konsole oder Terminal bezeichnet. Sie dient zur Steuerung einer Software, die normalerweise im Textmodus abläuft. (vgl. wikipedia.org, 2017)

Verschiedenste Dienste, unter anderem die MySQL-Datenbank und MySQL-Workbench, haben wir durch eine Cmd-Konsole gestartet, damit Rechte und Einstellungen immer wieder gesetzt werden können und auch veränderbar sind.

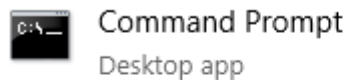


Abbildung 14: cmd-Verknüpfung

10.1.9 Putty

Putty ist eine kostenlose Software und wird verwendet, um Verbindungen zu einem Server herzustellen. Neben Windows kann Putty auch auf Linux verwendet werden. Die Funktionalität lässt sich mit der einer Kommandozeile vergleichen. (vgl. wikipedia.org, 2017)

Putty wurde von EasyGym verwendet, um über eine Kommandozeile auf den Server zuzugreifen. Verschiedene Ausgaben und Fehlermeldungen konnten so angezeigt werden. Darüber hinaus konnten auch Restarts des Servers hier durchgeführt werden.



Abbildung 15: Putty
(wikipedia.org, 2017)

10.1.10 SourceTree

SourceTree ist eine kostenlose, grafische Benutzeroberfläche, die es ermöglicht, Git- und Mercurial-Repositories zu verwalten. (vgl. wikipedia.org, 2017)

Der Hersteller Atlassian ist bekannt für Entwicklungs- und Projektmanagement-Werkzeuge. SourceTree ist für die Betriebssysteme Windows und Mac erhältlich. Es visualisiert die Aktivitäten auf einem Repository und erleichtert die Verwaltung eines solchen. Eine Bedienung durch einen Terminal ist daher nicht notwendig. (vgl. Atlassian, 2017)

In dieser Arbeit wurde SourceTree von den Entwicklern zur gemeinsamen Speicherung aller Daten der Diplomarbeit verwendet. Weiters ermöglicht SourceTree das Zusammenführen von Dokumenten und Files, wenn gleichzeitig in derselben Datei gearbeitet wird.



Abbildung 16: SourceTree
(sourcetreeapp.com, 2017)

10.2 Sonstige verwendete Software

10.2.1 Microsoft Office

Unter Microsoft Office ist ein Paket erhältlich, welches sich je nach Edition in den enthaltenen Komponenten, dem Preis sowie der Lizenzierung unterscheidet. Heutzutage sind Office-Produkte wie Word oder PowerPoint Standard in allen Schulen und Büros. (vgl. wikipedia.org, 2017)



Abbildung 17: Microsoft-Office
(wikipedia.org, 2017)

Die folgenden Office Produkte wurden für die Diplomarbeit verwendet:

Microsoft Word:

- Dokument- und Textverarbeitungssoftware
- Für jegliche Art von Dokumentation und schriftliche Ausarbeitungen



Abbildung 18:
Word
(wikimedia.org,
2017)

PowerPoint:

- Präsentationssoftware
- Für Projektpräsentation



Abbildung 19:
PowerPoint
(wikipedia.org,
2017)

Excel:

- Tabellenkalkulationssoftware
- Für Kalkulationen, Aufwandsschätzungen, usw.



Abbildung 20: Excel
(wikipedia.org,
2017)

Visio:

- Visualisierungssoftware
- Für Pläne, Diagramme



Abbildung 21: Visio
(sleepyshark.com,
2017)

10.2.2 GIMP 2

GIMP 2 ist ein kostenloses Bildbearbeitungsprogramm zum Editieren von Pixel- und Vektorgrafiken. Die erste Version von GIMP wurde im Januar 1996 veröffentlicht. (vgl. wikipedia.org, 2017)

In dieser Arbeit wurde GIMP 2 zum Erstellen des Logos, sowie für Grafiken, die erstellt, umgespeichert, gefärbt, modifiziert oder zugeschnitten werden mussten, verwendet.



*Abbildung 22: GIMP
(wikipedia.org, 2017)*

10.3 Verwendete Technologien

10.3.1 Java

Java ist eine objektorientierte Programmiersprache. Das bedeutet, dass Programme in Einheiten unterteilt werden. Eine einzelne Einheit wird Objekt genannt. Ein solches Objekt besitzt wiederum einen Zustand, welcher sich aus den Objektattributen ergibt. Durch im Objekt enthaltene Funktionen, auch Methoden genannt, lässt sich ein Objekt verwalten. Ob ein Objekt auf Methoden, die den Zustand ändern würden, reagiert, ist dem Objekt selbst überlassen und dadurch besitzt ein Objekt jederzeit einen selbstkontrollierten und wohldefinierten Zustand. Objekte können untereinander mehrfach in Beziehung stehen.

Neben dem objektorientierten Aspekt ist Java auch plattformunabhängig. Dies ermöglicht es, Anwendungen, die in Java codiert sind, auf jedem Gerät, welches Java installiert hat, ausführen zu können. (vgl. itwissen.info, 2017)

Das JRE (Java Runtime Environment, Laufzeitumgebung) ist für die Ausführung von Java-Anwendungen notwendig.

Das JDK (Java Development Kit, Entwicklungswerkzeug) wird für die Codierung und Kompilierung von Java-Programmen benötigt. Grob lässt sich die Java-Technologie in diese beiden Komponenten trennen. (vgl. ubuntuusers.de, 2017)



Abbildung 23: Java (logos-download.com, 2017)

10.3.2 Java Persistence API (JPA)

JPA ist der offizielle Standard des ORM's (Object Relational Mapping) von Java-Objekten. Durch dieses Verfahren lassen sich Objekte in Datenbanken speichern. Die aus der Java-Anwendung kommenden Klassen und Objekte werden in der Datenbank als Tabellen und Zeilen abgebildet. (vgl. itwissen.info, 2017)

10.3.3 Java Server Faces (JSF)

Java Server Faces (JSF) ist ein auf Java basierendes Web-Application Framework und gehört daher zur Kategorie der Webtechnologien in Java. Den Hauptverwendungsbereich findet JSF im Entwickeln von grafischen Benutzeroberflächen. JSF vereinfacht das Verbinden von Komponenten der Benutzeroberfläche mit Datenquellen. Es ergänzt Standardkomponenten von HTML, um auf Serverdaten zugreifen zu können. (vgl. tutorialspoint.com, 2017)

10.3.4 Wildfly

Wildfly ist ein plattformunabhängiger, kostenloser Anwendungsserver. Er arbeitet nach dem Java-Enterprise-Edition-Standard und ist in Java codiert. Die aktuelle Version ist 10.1.0.Final (August 2016). (vgl. wikipedia.org, 2017)



Abbildung 24: Wildfly
(jboss.org, 2017)

10.3.5 HTML (Hypertext Markup Language)

Unter dem Kürzel HTML steckt der Name Hypertext Markup Language. HTML ist eine textbasierte Programmiersprache, die für die Strukturierung von Online-Webseiten verwendet wird. HTML-Dokumente sind die Grundlage des World Wide Web und werden von Webbrowsern dargestellt. (vgl. wikipedia.org, 2017)



Abbildung 25: HTML5
(w3.org, 2017)

10.3.6 CSS (Cascading Style Sheet)

CSS ist eine Stylesheet-Sprache und bildet zusammen mit der Auszeichnungssprache HTML und DOM (Document Object Model) die Kernsprachen der Web-Entwicklung. Da die Sprache auf so gut wie jeder Webseite verwendet wird, zählt sie zu den sogenannten „living standards“ und wird fortwährend weiterentwickelt. Die derzeitige Version ist CSS3 und befindet sich seit dem Jahr 2000 in Entwicklung. Im Gegensatz zu den CSS-Versionen vor CSS3 ist diese Version modular aufgebaut. Dieser Aufbau ermöglicht es, Teiltechniken (Module) in jeweils eigenen Versionsschritten zu entwickeln. (vgl. wikipedia.org, 2017)



Abbildung 26: CSS3
(comsysto.com, 2017)

10.3.7 AJAX (Asynchronous JavaScript and XML)

AJAX ist ein Webservice, der auf dynamischen Webseiten verwendet wird. Durch ihn lassen sich einzelne Seitenteile neu laden, ohne die komplette Seite neu aufbauen zu müssen. Bei Anfragen, die keine Verbindung zum Server benötigen, wird die Arbeit vom Client erledigt. Bei Requests auf den Server, wird die Anfrage im Hintergrund bearbeitet und sobald eine Antwort des Servers kommt, wird der betroffene Seitenteil neu geladen. (vgl. itwissen.info, 2017)

10.3.8 SQL (structured query language)

SQL ist eine Programmiersprache und wird für Datenbankzugriffe verwendet. Im Gegensatz zu den meisten Programmiersprachen arbeitet SQL deklarativ und nicht prozessorientiert. Das bedeutet, sie beschreibt, was zu tun ist und nicht, wie etwas erledigt werden muss. Seit der Veröffentlichung 1974 ist sie bis heute der Standard für den Zugriff auf strukturierte Daten. (vgl. itwissen.info, 2017)

10.3.9 Android

Android ist ein Betriebssystem, vor allem für Smartphones und Tablets entwickelt, das erst seit 2008 am Markt zu finden ist. Google erwirtschaftete bis Jänner 2016 einen Umsatz von 31 Mrd. US-Dollar mit Android. Android wird zu 84,9 Prozent als Smartphone-Betriebssystem verwendet (Stand 2014). (vgl. wikipedia.org, 2017)

"Die Architektur von Android baute am Anfang auf den Linux-Kernel 2.6 auf, seit Android 4.x werden Kernel der Reihe 3.x verwendet. Dieser Kernel ist zuständig für die Speicher- und Prozessorverwaltung und stellt auch die Gerätetreiber des Systems zur Verfügung. In Android wird hauptsächlich in Java programmiert wobei auch andere Sprachen zum Einsatz kommen. Der Linux-Kernel selbst ist in C programmiert.

Anwendungen unter Android mit der Endung .apk (android package) werden ebenfalls in Java programmiert, greifen aber des öfteren auf Bibliotheken zu, die in C oder C++ geschrieben sind, die am Kernel liegen." – Aussagen von Martin Scholz, Gründer und Inhaber, von Tirol Tech (blogspot.co.at, 2017)



Abbildung 27: Android
(wikia.com, 2017)

10.4 Verwendete Bibliotheken

10.4.1 PrimeFaces

PrimeFaces ist eine kostenlose Bibliothek und wird im Zusammenhang mit JSF verwendet. Sie beinhaltet User-Interface-Komponenten, welche sich vor allem durch ein spezielles Layout von den Standard JSF „user-interface“ Komponenten unterscheiden. Innerhalb dieser Bibliothek gibt es Themes, welche von der Community gestaltet wurden und frei zugänglich sind. Eine Auswahl von Themes sind kostenpflichtig und können nur durch den Erwerb verwendet werden. (vgl. primefaces.org, 2017)

In unserer Arbeit verwenden wir das Community-Theme „Rocket“.



Abbildung 28: Primefaces (doubleslash.de, 2017)

10.4.2 Rocket

PrimeFaces unterstützt die von ThemeRoller präsentierten CSS Frameworks. (vgl. primefaces.org, 2017)

Dort werden verschiedene CSS-Vorlagen veröffentlicht, die eingebunden werden können. Darunter befindet sich auch Rocket, welches zu den Community-Designs zählt und frei verwendbar ist.

10.4.3 Gson

Gson ist eine Open-Source Library und auch unter dem Namen Google JSON bekannt. (vgl. wikipedia.org, 2017)

Die Java Library ermöglicht es, Java Objekte in ein JSON Format zu verwandeln. Ebenso kann sie auch verwendet werden, um einen JSON String zu einem entsprechenden Java-Objekt zu konvertieren. (vgl. google.com, 2017)



10.4.4 Volley

Volley ist eine HTTP-Bibliothek, die eine Datenübertragung vom Server zum Client und umgekehrt erleichtert und schneller abwickelt, als bisherige Varianten. Die Bibliothek wurde für Android entwickelt. (vgl. android.com, 2017)

Volley kann außerdem in Verbindung mit Strings und JSON verwendet werden, wie es auch in dieser Arbeit gemacht wird.

10.1 Verwendete Architekturen

10.1.1 REST

10.1.1.1 Allgemein

REST ist eine Schnittstelle, die eine Kommunikation unter Maschinen ermöglicht. Das World Wide Web stellt selbst eine gigantische REST Anwendung dar. Viele Suchmaschinen, Shops oder Buchungssysteme sind ohne Absicht bereits als REST basierte Web Services verfügbar. (vgl. Thomas Bayer, 2017)

EasyGym verwendet REST, um Daten vom Server aus der Datenbank zu laden und zu einem bestimmten Nutzer zu senden. Diese Datenformate werden durch das Format JSON übertragen und können dadurch plattformunabhängig ausgelesen und verwendet werden. Der Nutzer muss dafür einen Request (dt. = Anfrage) absenden, der aufgrund des Pfades bestimmte Aktionen vom Server verlangt. Der Server muss nun - je nach Request-Art - meistens Informationen aus der Datenbank auslesen und als Response (dt. = Antwort) an den Client zurücksenden. Diese Art wird GET-Request genannt.

Es gibt aber auch andere Anfragen, bei denen Daten vom Aufrufer mitgesendet werden können, die nun auf dem Server gespeichert werden müssen. Hier gibt es die zwei Arten POST und PUT:

- POST wird zum Erstellen verwendet
- PUT wird zum Erstellen und zum Updaten verwendet

Es ist außerdem möglich, durch den DELETE-Request eine Entität zu löschen.

(vgl. stackoverflow.com, 2017)



10.1.1.2 REST-Aufruf

Der Aufruf erfolgt durch eine URL, die sich aus folgenden Teilen zusammensetzt:

Er beginnt mit der Server-URL, im Fall von EasyGym mit <http://easygymaskoe.htl-perg.ac.at/>. Danach wird der Projektname angehängt mit [EasyGymAskoe/](#).

Weiters muss nun der Pfad für die REST-Applikation mit [rest/](#) angegeben werden und durch den für jede Entität eigenen Pfad, wie zum Beispiel [endusers](#), ergänzt werden.

Je nach Aufruf muss der Gesamt-Pfad eindeutig sein, um die dahinterliegenden Methoden individuell ansprechen zu können. Dafür wird nun noch der Zusatzpfad für eine bestimmte Methode aufgerufen. (zum Beispiel: [/1](#))

Die Zusammensetzung des Pfades für die Entität *EndUser*:

Server-URL	Projektname	Pfad REST	Pfad Entität	Zusatzpfad
http://easygymaskoe.htl-perg.ac.at/	EasyGymAskoe/	rest/	endusers	/1

Tabelle 1: Zusammensetzung REST-Pfad

Der gesamte Pfad für die Entität *EndUser*:

Gesamtpfad
http://easygymaskoe.htl-perg.ac.at/EasyGymAskoe/rest/endusers/1

Tabelle 2: gesamter REST-Pfad

Dieser Request fordert beispielsweise den Endnutzer mit der ID (Identifikationsnummer) 1 vom Server an. Dafür wird die Methode *findById(id)* aufgerufen. Die ID wird dabei aus dem Zusatzpfad entnommen.

Die Requests können dabei entweder vom Typ GET, POST, PUT oder DELETE sein und unterscheiden sich darin, dass bei GET Daten zurückgesendet werden, bei POST und PUT Daten in der Datenbank gespeichert werden und bei DELETE Werte aus der Datenbank gelöscht werden.



10.1.2 DAO

Ein Dao (Data Access Objekt) ist für den Zugriff von Datenquellen auf unterschiedliche Arten zuständig. Dabei ist es so gekapselt, dass die Datenquelle auch ausgetauscht werden kann, ohne dass das DAO geändert werden muss.

Das Ziel eines DAOs ist, dass die Programmlogik frei von technischen Details zur Datenspeicherung ist, sich dabei nur um den Zugriff kümmert und somit flexibler einsetzbar ist. Ein DAO ist also ein Muster für die Gestaltung von Programmierschnittstellen (APIs). (vgl. wikipedia.org, 2017)

Der Vorteil von DAOs ist unter anderem dann zu sehen, wenn eine Anwendung portiert werden muss. Zum Beispiel kann eine Anwendung vor der Einführung separat und in einem eigenständigen System entwickelt und getestet werden. Wenn dieses Programm nun auf einem anderen Speichermedium läuft, als im Gesamtsystem verwendet wird, muss sie portiert werden. Durch den Einsatz von DAOs kann das reibungsfrei erfolgen, ohne Veränderungen im Source-Code vornehmen zu müssen.



11 Realisierung

11.1 Vorgehensweise

Dieses Projekt wurde nach dem Prinzip des Vorgehensmodells „SCRUM“ durchgeführt.

11.1.1 SCRUM

Das Vorgehensmodell vieler Softwareunternehmen basiert auf Scrum, einem sehr vielfältig einsetzbaren Rahmenwerk. Es besteht aus einem Scrum-Team inklusive den damit verbundenen Rollen, Ereignissen sowie Regeln. Jede Komponente ist für ein erfolgreiches Arbeiten unentbehrlich.

11.1.1.1 Das Scrum Team:

Product Owner:

Allgemein: Der Product Owner ist für die Arbeit innerhalb des Entwicklungsteams verantwortlich. Er organisiert und lenkt das Projekt in die gewünschte Richtung.

Weiters ist er für das Management des Product Backlogs verantwortlich. Diese Art von Management beinhaltet Aufgaben wie Product Backlog-Einträge klar zu definieren und sortieren, damit ein optimales Ergebnis erreicht werden kann. Diese Aufgaben kann der Product Owner entweder selbst erledigen oder an das Entwicklungsteam weitergeben.

Teamintern: Die Rolle des Product Owners übernimmt in unserem Projekt das Entwicklungsteam.

Entwicklungsteam:

Allgemein: Ein Entwicklungsteam organisiert sich selbst und stellt am Ende eines jeden Sprints ein fertiges Inkrement zur Verfügung. Es agiert als eine Einheit, was heißen soll, dass jedes Mitglied als Entwickler angesehen wird und keine tiefere Unterteilung innerhalb eines Teams vorherrscht.

Die optimale Größe eines solchen Teams beläuft sich auf mindestens drei Personen, um alle Fähigkeiten, die in einem Projekt gebraucht werden, abdecken sowie einen Sprint effektiv nutzen zu können, und maximal neun Personen um den Überblick nicht zu verlieren.



Teamintern: Die Rolle der Entwickler übernehmen Paul Heimele und David Hennerbichler. Als Unterstützung steht das Unternehmen Aberger Software zur Verfügung, welches bei Schwierigkeiten um Rat gefragt werden kann.

Scrum Master:

Allgemein: Der Scrum Master ist für eine korrekte Ausführung des Scrum-Prozesses während der gesamten Dauer des Projekts verantwortlich. Weiters ist es seine Pflicht, allen Mitgliedern des Teams Scrum, sowohl in der Theorie, als auch in der Praxis, zu erklären.

Teamintern: Die Rolle des Scrum Masters übernimmt Paul Heimele.

11.1.1.2 Scrum Ereignisse:

Sprint: Die Dauer eines Sprints beschränkt sich auf maximal einen Monat. Am Ende jedes Sprints soll es ein nutzbares und potenziell auslieferbares Produkt-Inkrement geben (eventuell ein Projektstatus am Ende des Sprints).

Sprint Planning: Hier werden die Aufgaben für den folgenden Sprint festgelegt. Die Aufgaben werden durch das gesamte Scrum-Team beschlossen (diese werden aus dem Product Backlog beziehungsweise dem Dokument, welches die Tasks enthält, entnommen).

Daily Scrum: Kurzes Meeting an allen Entwicklungstagen, in dem die Planung der nächsten 24 Stunden stattfindet (optional kann man hierfür ein Protokoll schreiben).

Sprint Review: Meeting am Ende jedes Sprints, um das erstellte Produkt-Inkrement zu überprüfen. Falls neue Aufgaben erkannt werden, können diese im Product Backlog eingetragen werden (Projektstatus, Protokoll).

Sprint Retrospektive: Hier wird der Sprint hinsichtlich der Effektivität der Mitarbeiter sowie der Qualität des Produkt-Inkrementes kontrolliert. Es können Verbesserungsvorschläge geäußert werden, um das Arbeiten im nächsten Sprint zu verbessern. (Protokoll)



11.1.1.3 Scrum Artefakte:

Product Backlog: Ist eine Liste von allem, was in dem gewünschten Produkt enthalten sein kann. Änderungen, die am Produkt getätigt werden sollen, sind hier vermerkt.

Sprint Backlog: Enthält die gesamten Aufgaben für einen Sprint. Aus einem Sprint Backlog soll ersichtlich sein, ob das Projektteam in der Zeit liegt oder nicht. (vgl. wikipedia.org, 2017)

11.2 Webanwendung

11.2.1 Grundlegende Ausgangslage

Die Entscheidung, welche Technologien für die Webanwendung verwendet werden, fiel noch vor dem Beginn des 4. Jahrgangs. Bis zu diesem Zeitpunkt beschäftigten wir uns hauptsächlich mit Java und deswegen entschieden wir uns für eine JSF (JavaServer Faces) Anwendung. Diese Technologie wurde außerdem zu diesem Zeitpunkt im Unterricht durchbesprochen. Da die Entwicklungsumgebung Eclipse als Schulsoftware zur Verfügung gestellt wurde, ergriffen wir diese Möglichkeit und nutzten Eclipse auch zur Verwirklichung der Webanwendung. Selbiges gilt auch für den lokalen Server, der wie im Unterricht ein Wildfly Server ist, auf dem hauptsächlich getestet und entwickelt wurde. Der Schulserver läuft auf dem Betriebssystem Ubuntu. Ebenso fiel die Wahl der Datenbank auf die durch den Unterricht bekannte MySQL Datenbank mit der dazugehörigen Anwendung MySQL Workbench.

Dieses gesamte Paket von Technologien und Umgebungen erleichterte es uns, wenn auch nur zu Beginn, EasyGym zu verwirklichen und gleichzeitig weiterhin mit relevanten und aktuellen Information in der Schule versorgt werden zu können.

11.2.2 Layout

Das Design wurde mit selbst geschriebenen CSS-Files und der Layout-Library PrimeFaces erstellt. Durch CSS-Files wurden die Elemente angeordnet und auf die gewünschte Größe skaliert. Weiters wurde die Menüleiste und die Titelleiste am oberen Bildschirmrand mittels CSS formatiert. Elemente wie Tabellen, Eingabefelder, Textanzeigen und ähnliches erhielten ein Standardlayout von der Library PrimeFaces.



11.2.3 Umsetzung Management-Webseite

11.2.3.1 Login-Seite

Die Webanwendung leitet durch eine generalisierte Login-Seite sowohl Entwickler als auch Mandanten auf die entsprechenden Verwaltungsoberflächen weiter.

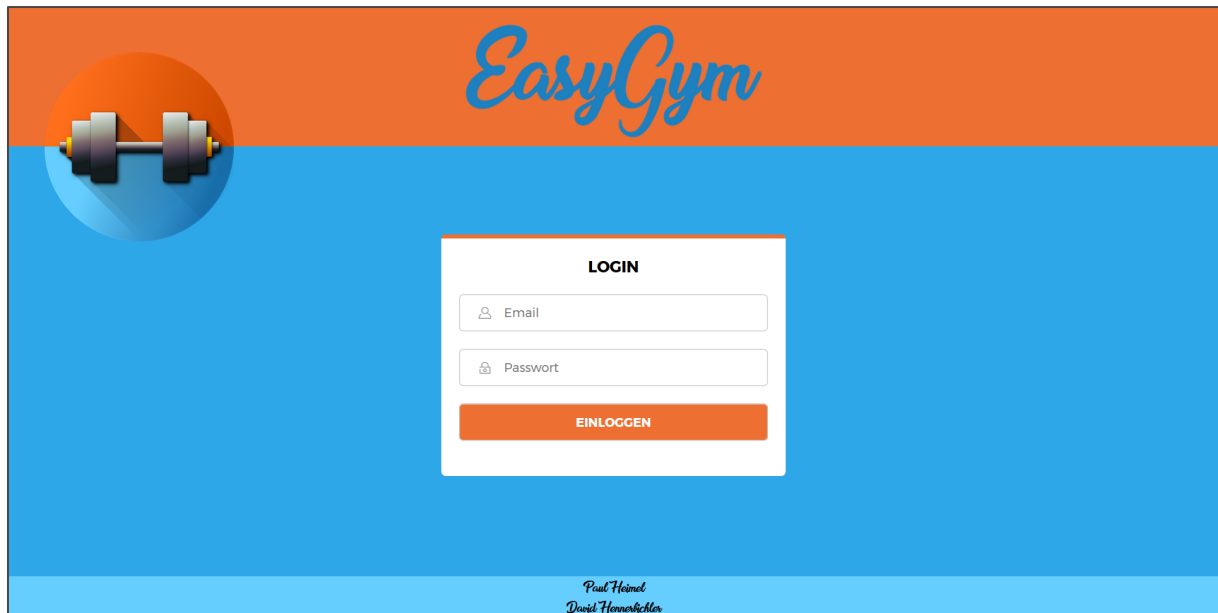


Abbildung 29: Login-Seite

Bei eingegebenen Login-Daten wird überprüft, ob die übergebenen Credentials (= Anmeldedaten) in der Datenbank gespeichert sind. Wenn diese vorhanden sind, werden nun entsprechende Zugriffsrechte ausgelesen und dem Benutzer folgende Zugänge angezeigt.

a) Entwickler-Zugang

Als Entwickler können wir einen neuen Mandanten-Zugang anlegen. Wenn ein Produktverkauf stattgefunden hat, wird ein Mandanten-Account mit dem gewünschten Benutzernamen und Passwort angelegt und per Email dem Mandant zugeschickt. Dies geschieht mithilfe eines Formulars, in dem der Name und die Anmeldedaten des Mandanten vergeben werden.



The screenshot shows the 'Mandanten' (Tenants) management page in the EasyGym system. The page has a blue header with the 'EasyGym' logo and navigation links for 'Home', 'Mandanten', and 'Ausloggen'. Below the header, there are two main sections: a table of existing tenants and a form to create a new one.

Name	E-Mail	Passwort	Bearbeiten	Löschen
Mandant1	askoe_admin	admin		
EasyGym	mcfi_admin	admin		
Injoy	injoy_admin	admin		
HappyFit	happyfit_admin	admin		
kein Mandant zugewiesen	mandant1	mandant1		

Mandant anlegen: NEU

Name:

E-Mail:

Passwort:



Abbildung 30: Entwickler-Zugang Mandanten-Seite

In einem Datenformular können nun alle Mandanten angezeigt, gelöscht und gegebenenfalls bearbeitet werden.

b) Mandanten-Zugang

Der Mandant hat nun eine einfach gehaltene Management-Oberfläche vor sich, durch die mit einem schlichten Menüband navigiert werden kann.

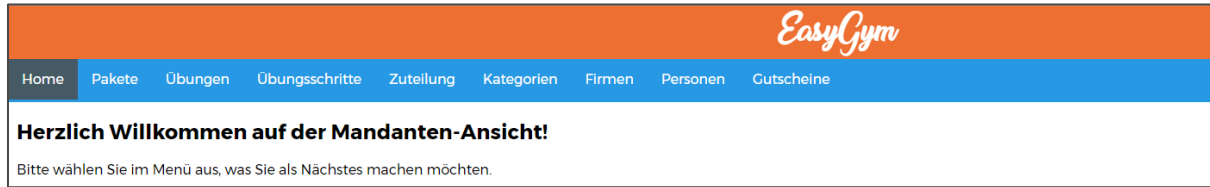


Abbildung 31: Mandanten Zugang - Menüband

Hierbei können Pakete, Übungen, Übungsschritte, Kategorien, Firmen, Personen und Gutscheine angezeigt, erstellt, bearbeitet und gelöscht werden. Dies geschieht durch verschiedene Formulare und Tabellen.

Grundsätzlich ist jede Seite, ausgenommen die Home-Seite, in zwei genau gleich große Spalten aufgeteilt.

Auf der linken Seite befindet sich immer die Anzeige der Daten, die je nach Inhalt variabel ist. Es befindet sich bei jeder umfangreicheren Tabelle ein Suchfeld. Damit kann schnell mit dem Namen nach den entsprechenden Objekten gesucht werden. Jede Zeile wird in der 1. Spalte durch eine fortlaufende Zahl durchnummeriert.

Auf der rechten Seite befindet sich ein Formular, welches auch hier wieder abhängig von den entsprechenden Informationen ist. Jedes Formular besitzt einen Absende-Button mit einer Aufschrift, je nachdem, ob gerade ein Wert bearbeitet oder erstellt wird. Weiters befindet sich ein Knopf, der das Löschen des Formulars bewirkt, rechts über dem Formular. Auf der Seite des Browserfensters rechts oben wird bei jeder Aktion eine zugehörige Mitteilung angezeigt, die eine visuelle Rückmeldung für den Benutzer bereithält.



11.2.3.2 Home-Seite

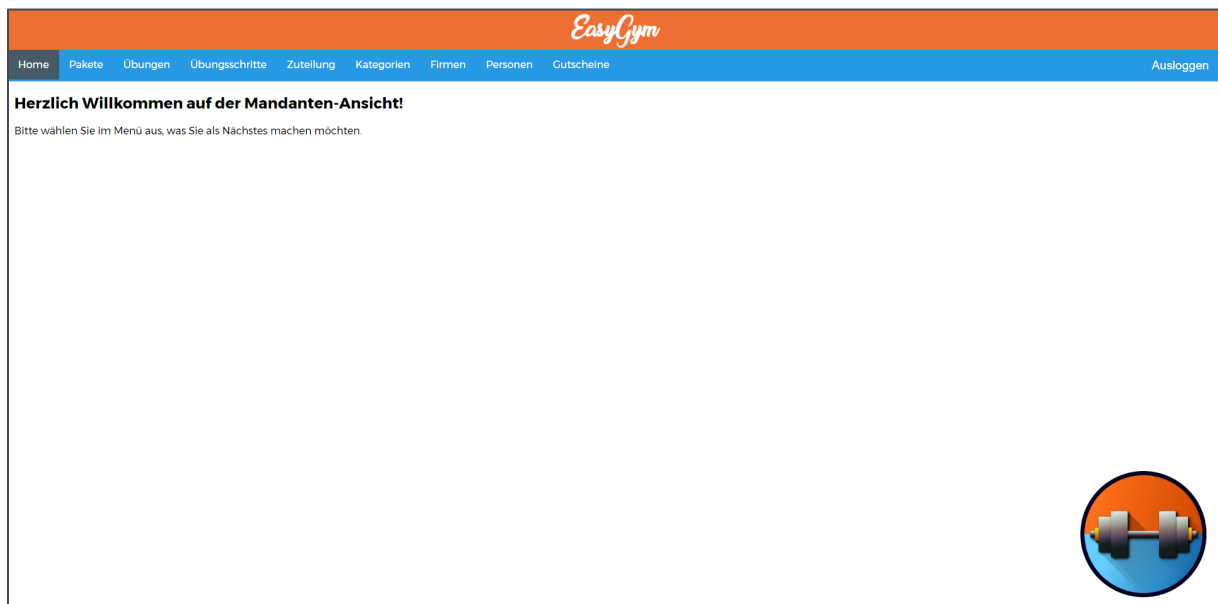


Abbildung 32: Home-Seite

Diese Seite wird nach dem erfolgreichen Login eines Mandanten angezeigt und verweist auf die Menüleiste am oberen Bildschirmrand. Das Logo von EasyGym wird rechts unten angezeigt.

11.2.3.3 Paket-Seite

Die Datentabelle der Paketseite besteht aus den folgenden Spalten:

The screenshot shows the 'Pakete' page in the EasyGym system. It features a navigation bar at the top with links to Home, Pakete, Übungen, Übungsschritte, Zuteilung, Kategorien, Firmen, Personen, Gutscheine, and Ausloggen. The main content area is divided into two sections: a table of packages and a form to create or edit a package.

Pakete Table:

Nr.	Titelbild	Paket ID	Name	Kurzbeschreibung	Credit-Kosten	Bearbeiten	Löschen
1.		8	Bauchmuskeln	FAUL war gestern!!	14		
Übungen im Paket:							
Nr.	Name	Credit-Kosten	Kurzbeschreibung	Löschen			
11.	Crunch	5	Die perfekte Übung für Anfänger.				
12.	Radfahrer Crunch	4	Ähnlich zum normalen Crunch, jedoch anspruchsvoller.				
13.	Situp	3	Der Klassiker unter den Bauchmuskelübungen.				
14.	Liegendes Beinheben	4	Auch im Liegen lässt sich der Bauch trainieren.				
15.	Seitliches Beinheben	2	Eine Übung für die wahren Profis.				
2.		9	Beinmuskulatur	Never skip leg day!!	13		
3.		10	Brustmuskulatur	Wer kämpft, kann verlieren. Wer nicht kämpft, hat schon verloren.	14		
4.		11	Armmuskulatur	Wenn du alles gibst, kannst du dir nichts	15		

Paket anlegen Form:

Name:

Kurzbeschreibung:

Detaillierte Beschreibung:

Level:

Titelbild:

Bild auswählen:

Abbildung 33: Paket-Seite

Neben der Nummer – wie zuvor beschrieben - wird das Titelbild des Pakets, gefolgt von der Paket ID, angezeigt. Daraufhin werden der Name, die Kurzbeschreibung und die Credit-Kosten des Pakets in dieser Zeile dargestellt. Mit dem Bearbeiten-Button, der das passende Material-Icon (Grafik) verwendet, kann nun ein Paket ausgewählt werden, welches rechts daneben im Formular angezeigt wird und verändert werden kann. Der Lösch-Button in der letzten Spalte löscht ein Paket unwiderruflich.

In einem Paket sind Übungen enthalten. Diese können durch den kleinen blauen Pfeil in Spalte 1 ausgeklappt werden. Alle aufgelisteten Übungen sind wiederum mit einem Löschbutton versehen, welcher die entsprechende Übung aus dem Paket entfernt.

Wünscht sich ein Mandant, ein neues Paket anzulegen, kann er dies mithilfe des rechts liegenden Formulars durchführen. In den Feldern Name, Kurzbeschreibung und detaillierte Beschreibung muss jeweils ein Textwert mit einer festgelegten Maximallänge eingegeben werden. Im Eingabefeld für das Level muss ein Zahlenwert eingegeben werden und darunter wird das aktuell gespeicherte Bild angezeigt, mit der Option, ein neues Bild auszuwählen. Ein Bild wird dann unter einem generierten Namen auf dem Server abgelegt. Bei der Erstellung des Pakets gibt es ein Standardbild, das verwendet wird, falls der Mandant kein Bild auswählt. Wenn nun ein Paket zum Bearbeiten ausgewählt wurde, können alle Werte rechts im Formular angesehen und verändert werden.

11.2.3.4 Übungsseite

The screenshot shows the 'Übungen' page in the EasyGym system. The main table lists exercises with columns for 'Nr.', 'Name', 'Credit-Kosten', 'Beschreibung', 'Paket IDs', 'Bearbeiten', and 'Löschen'. Exercise 2, 'Radfahrer Crunch', is selected. A sub-table titled 'Steps der Übung:' shows five steps with descriptions and images. To the right, the 'Übung anlegen:' form is visible, with fields for 'Name' (Radfahrer Crunch), 'Kurzbeschreibung', 'Genaue Beschreibung', and 'Credit-Kosten' (4). A 'Kategorien auswählen' list shows categories 6 through 11, with checkboxes for 'Gerader Bauchmuskel' and 'Seitlicher Bauchmuskel' checked. A 'Speichern' button is at the bottom.

Abbildung 34: Übungsseite

Auch auf der Übungsseite befindet sich eine Datentabelle. Sie beginnt wieder mit einer Nummerierung und dem Namen der Übung, gefolgt von Credit-Kosten und den IDs aller Pakete, in denen sie vorkommt. Die nächsten beiden Spalten sind wieder zum Bearbeiten und Löschen der Übung.

Wenn man die Übung wieder mit dem blauen Pfeil aufklappt, kommen alle Übungsschritte durch eine Nummerierung gereiht darunter in einer Datentabelle zum Vorschein. Bei jedem Übungsschritt wird die Beschreibung und das gespeicherte Bild angezeigt. Auch hier gibt es die Möglichkeit, einen Übungsschritt direkt zu löschen oder zu bearbeiten, um gegebenenfalls das Beschreibungsbild zu ändern.

In dem danebenliegenden Formular kann man nun eine Übung bearbeiten oder erstellen. In den dafür vorgesehenen Feldern kann man den Namen, die Kurzbeschreibung, eine ausführlichere Beschreibung und die Credit-Kosten eingeben. Unter dem Formular befindet sich eine Kategorienliste. Hier werden alle Kategorien mit dem Namen durchnummeriert aufgelistet und daneben befindet sich eine Checkbox, die es ermöglicht, der Übung eine beliebige Anzahl an Kategorien zuzuteilen.

11.2.3.5 Übungsschritt-Seite

Um nun für eine Übung Übungsschritte zu definieren, gibt es die Übungsschritt-Seite, auf der man eine Übung auswählen und Übungsschritte hinzufügen kann.

The screenshot shows the 'Übungsschritt-Seite' in the EasyGym system. The main navigation bar includes 'Home', 'Pakete', 'Übungen', 'Übungsschritte', 'Zuteilung', 'Kategorien', 'Firmen', 'Personen', 'Gutscheine', and 'Ausloggen'. The 'Übungen' section is active, displaying a table of exercises:

Nr.	Name	Credit-Kosten	Beschreibung	Paket IDs	Löschen	Schritt erstellen
1.	Crunch	5	Die perfekte Übung für Anfänger.	8		
2.	Radfahrer Crunch	4	Ähnlich zum normalen Crunch, jedoch anspruchsvoller.	8		

Below the exercise list, a 'Steps der Übung' table is shown for the selected 'Radfahrer Crunch' exercise:

Nr.	Beschreibung	Bild	Bearbeiten	Löschen
2.1.	Auf den Rücken legen und die Beine anwinkeln.			
2.2.	Beide Hände zu den Ohren halten, wobei die Ellbogen nach außen zeigen.			
2.3.	Beide Beine auf einen 90° Winkel anheben.			
2.4.	Abwechselnd ein Bein ausstrecken, ohne es auf den Boden zu legen. Zeitgleich den Oberkörper heben und jeweils zu dem Knie drehen, welches gerade angewinkelt ist, bis sich dein Ellbogen und dein Knie fast berühren.			
2.5.	Wieder zurück in die Ausgangsposition und beliebig oft wiederholen.			

On the right side, the 'Step anlegen:' form is visible. It includes a dropdown menu for 'Übung auswählen:' (currently set to 'Radfahrer Crunch'), a 'Beschreibung' field with a text area containing instructions, a 'Bild:' field with a preview image, and a 'Bild auswählen:' field with a 'Datei auswählen' button. A 'Speichern' button is at the bottom of the form.

Abbildung 35: Übungsschritt-Seite

Diese Datentabelle ist fast identisch mit der Datentabelle auf der Übungsseite, allerdings mit einem markanten Unterschied: Es befindet sich statt dem „Bearbeiten-Button“ ein „Übung-Auswählen-Button“ in der letzten Spalte. Durch diesen Button wird im Formular nun das Auswahlmenü auf die entsprechende Übung gesetzt. Alternativ ist es auch möglich, manuell im Auswahlmenü eine Übung auszuwählen. Bei dieser ausgewählten Übung ist es nun möglich, einen Übungsschritt anzulegen.

In der Datentabelle von den Übungen kann man nun die Übungsschritte durch Ausklappen betrachten, bearbeiten oder löschen.

11.2.3.6 Zuteilungsseite

Bis zu diesem Zeitpunkt sind Pakete noch leer. Das kann nun mithilfe der Zuteilungsseite geändert werden, denn hier kann man Übungen einem beliebigen Paket zuteilen.

The screenshot shows the 'Zuteilung' (Assignment) page in the EasyGym system. The page is divided into two main sections: 'Pakete' (Packages) on the left and 'Übungen' (Exercises) on the right.

Pakete (Packages): A table lists packages with columns for 'Nr.', 'Titelbild', 'Paket ID', 'Name', 'Credit-Kosten', 'Löschen', and 'Übungen zuteilen'. Package 1 is selected, showing a sub-table 'Übungen im Paket' (Exercises in Package) with columns for 'Nr.', 'Name', 'Credit-Kosten', 'Kurzbeschreibung', and 'Löschen'. The exercises listed are:

Nr.	Name	Credit-Kosten	Kurzbeschreibung	Löschen
11.	Crunch	5	Die perfekte Übung für Anfänger.	<input type="checkbox"/>
12.	Radfahrer Crunch	4	Ähnlich zum normalen Crunch, jedoch anspruchsvoller.	<input type="checkbox"/>
13.	Situp	3	Der Klassiker unter den Bauchmuskelübungen.	<input type="checkbox"/>
14.	Liegendes Beinheben	4	Auch im Liegen lässt sich der Bauch trainieren.	<input type="checkbox"/>
15.	Seitliches Beinheben	2	Eine Übung für die wahren Profis.	<input type="checkbox"/>

Übungen (Exercises): A dropdown menu 'Paket auswählen:' is set to 'Bauchmuskeln'. Below it is a table of exercises with columns for 'Nr.', 'Name', 'Credit-Kosten', and 'Im Paket'. The exercises listed are:

Nr.	Name	Credit-Kosten	Im Paket
1.	Crunch	5	<input checked="" type="checkbox"/>
2.	Radfahrer Crunch	4	<input checked="" type="checkbox"/>
3.	Situp	3	<input checked="" type="checkbox"/>
4.	Liegendes Beinheben	4	<input checked="" type="checkbox"/>
5.	Seitliches Beinheben	2	<input checked="" type="checkbox"/>
6.	Air Squat	3	<input type="checkbox"/>
7.	Bulgarische Kniebeuge	4	<input type="checkbox"/>
8.	Seitlicher Ausfallschritt	3	<input type="checkbox"/>
9.	Wadenheben stehend	1	<input type="checkbox"/>
10.	Zehenheben	2	<input type="checkbox"/>
11.	Liegestütz negativ	3	<input type="checkbox"/>
12.	Gleitender Liegestütz	3	<input type="checkbox"/>
13.	Skorpion Liegestütz	4	<input type="checkbox"/>

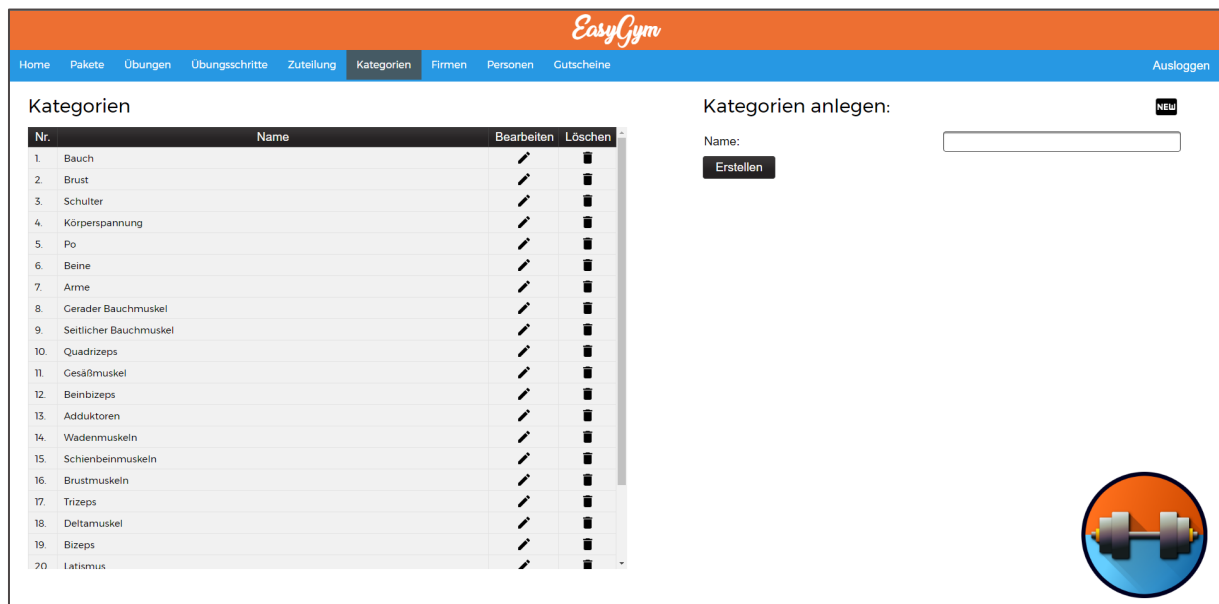
A 'Speichern' (Save) button is located at the bottom of the exercise list.

Abbildung 36: Zuteilungsseite

Auf der linken Seite befindet sich eine Datentabelle, die ähnlich wie oben, wieder eine Zuteilungsspalte besitzt, die das Paket in die rechts danebenliegende Auswahlbox überträgt. Auch hier wurde wieder der Bearbeiten-Button entfernt. Ein Paket kann man durch den blauen Pfeil ausklappen und somit alle in diesem Paket erhältlichen Übungen anzeigen.

Auf der rechten Seite unten befindet sich eine Liste von Übungen. Wählt man ein Paket aus, werden jene Übungen, die in diesem Paket enthalten sind, mit einer aktiv-gesetzten Checkbox angezeigt. Alle nicht enthaltenen Übungen können nun über diese Checkbox hinzugefügt werden, oder auch, im Falle von gesetzten Übungen, entfernt werden.

11.2.3.7 Kategorie-Seite











































Nr.	Name	Bearbeiten	Löschen
1.	Bauch		
2.	Brust		
3.	Schulter		
4.	Körperspannung		
5.	Po		
6.	Beine		
7.	Arme		
8.	Cerader Bauchmuskel		
9.	Seitlicher Bauchmuskel		
10.	Quadrizeps		
11.	Cesäßmuskel		
12.	Beinbizeps		
13.	Adduktoren		
14.	Wadenmuskeln		
15.	Schienbeinmuskeln		
16.	Brustmuskeln		
17.	Trizeps		
18.	Deltamuskel		
19.	Bizeps		
20.	Laticmus		

Abbildung 37: Kategorie-Seite

Auf der Kategorie-Seite können nun die zuvor beschriebenen Kategorien von Übungen erstellt werden. Dafür befindet sich links die Anzeige von Übungen und rechts das Formular, um eine Kategorie zu bearbeiten oder anzulegen. Eine Kategorie besitzt lediglich einen Namen, der einer Übung zugewiesen werden kann.



11.2.3.8 Firmen-Seite

Nr.	Name	Bearbeiten	Löschen
1.	Mandant1		
2.	Hueck Folien		
3.	keine Firma zugeordnet		
4.	EasyGym		

Abbildung 38: Firmen-Seite

Auch hier, auf der Firmenseite, befindet sich links eine Datentabelle zur Anzeige von erstellten Firmen. Rechts daneben befindet sich das Formular, in dem wiederum bearbeitet oder erstellt werden kann.

11.2.3.9 Personen-Seite

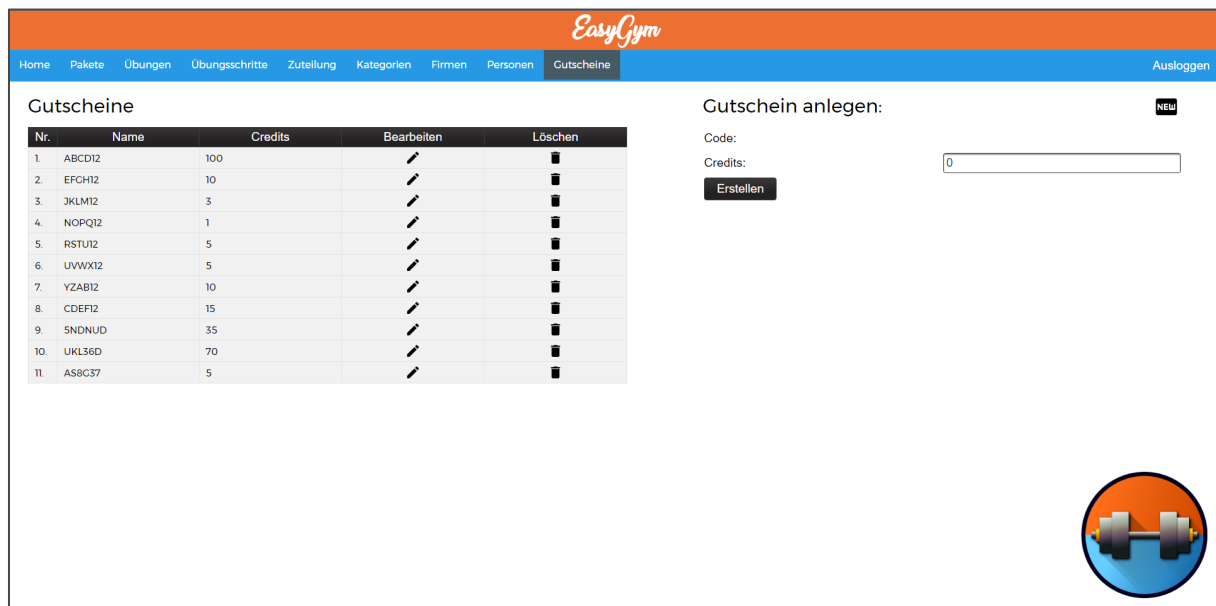
Nr.	ID	Vorname	Nachname	Email	Passwort	Bearbeiten	Löschen	Gutschein zuteilen
1.	1	David	Hennerbichler	hennerbichler	12345678			
2.	2	Michael	Leonhart	leonhartsberger	user			
3.	3	Tobias	Janko	janko	12345678			
4.	4	Tobias	Kappmüller	kappmüller	12345678			
5.	5	Paul	Heimel	heimel	12345678			

Nr.	Code	Credits	Zugewiesen
1.	ABCD12	100	<input checked="" type="checkbox"/>
2.	EFGH12	10	<input checked="" type="checkbox"/>
3.	JKLM12	3	<input checked="" type="checkbox"/>
4.	NOPQ12	1	<input type="checkbox"/>
5.	RSTU12	5	<input type="checkbox"/>
6.	UVMX12	5	<input type="checkbox"/>
7.	YZAB12	10	<input type="checkbox"/>

Abbildung 39: Person-Seite

Die links angezeigten Informationen in der Datentabelle reichen von der Nummerierung über die ID zu Vornamen, Nachnamen, Email und Passwort. In den letzten Spalten gibt es wieder die Möglichkeit, über Buttons Personen zu bearbeiten, zu löschen oder auszuwählen. Im Formular kann nun durch Eingabe von Vor- und Nachnamen, Email beziehungsweise Benutzernamen und Passwort ein Endnutzer erstellt werden. Darunter befindet sich eine Liste von Gutscheinen, die einem Endnutzer zugewiesen werden können. Auch hier werden die bereits vom Endnutzer zugewiesenen Gutscheine als gesetzt in der Checkbox angezeigt. Nun können Gutscheine wieder beliebig vom Endnutzer zugewiesen oder an diesen zugewiesen werden.

11.2.3.10 Gutschein-Seite

























Nr.	Name	Credits	Bearbeiten	Löschen
1.	ABCD12	100		
2.	EFGH12	10		
3.	JKLM12	3		
4.	NOPQ12	1		
5.	RSTU12	5		
6.	UVWX12	5		
7.	YZAB12	10		
8.	CDEF12	15		
9.	5NDNUD	35		
10.	UKL36D	70		
11.	ASBG37	5		

Abbildung 40: Gutschein-Seite

Auf der Gutscheinseite befindet sich links wieder die Datentabelle, in der Gutscheine durchnummeriert angezeigt werden. Bei einem Gutschein werden der Code, die noch verfügbaren Credits und der Bearbeiten- beziehungsweise Löschen-Button angezeigt. Rechts daneben werden im Formular der Code und die verfügbaren Credits angezeigt.

Bei der Erstellung eines Gutscheins wird ein neuer, zufälliger Code mit der Länge von sechs Zeichen generiert. Sehr ähnliche Zeichen, wie i, I, l, 1, oder o, O, 0 wurden dabei aufgrund der verbesserten Benutzerfreundlichkeit ausgespart.

11.2.4 Besonderheiten Management-Webseite

11.2.4.1 Login

Neben dem Standard-Einloggen müssen beim Login noch viele weitere Funktionen berücksichtigt werden. Sobald sich ein User einloggt, wird eine Session gestartet. Es wird ein individualisierter Authorization-Filter vor jedem Seiten-Aufruf gestartet. Beim Login wird jeweils ein Session-Attribut mit dem Benutzernamen und den Zugriffsberechtigungen gesetzt. Falls ein Benutzer noch eingeloggt und die Session noch nicht abgelaufen ist, wird dieser ohne Login sofort auf die gewünschte Seite geleitet. Diese Attribute werden nun ebenso bei jedem Seitenwechsel ausgelesen und es wird geprüft, ob die Person die nötigen Rechte für den Seitenaufruf besitzt.

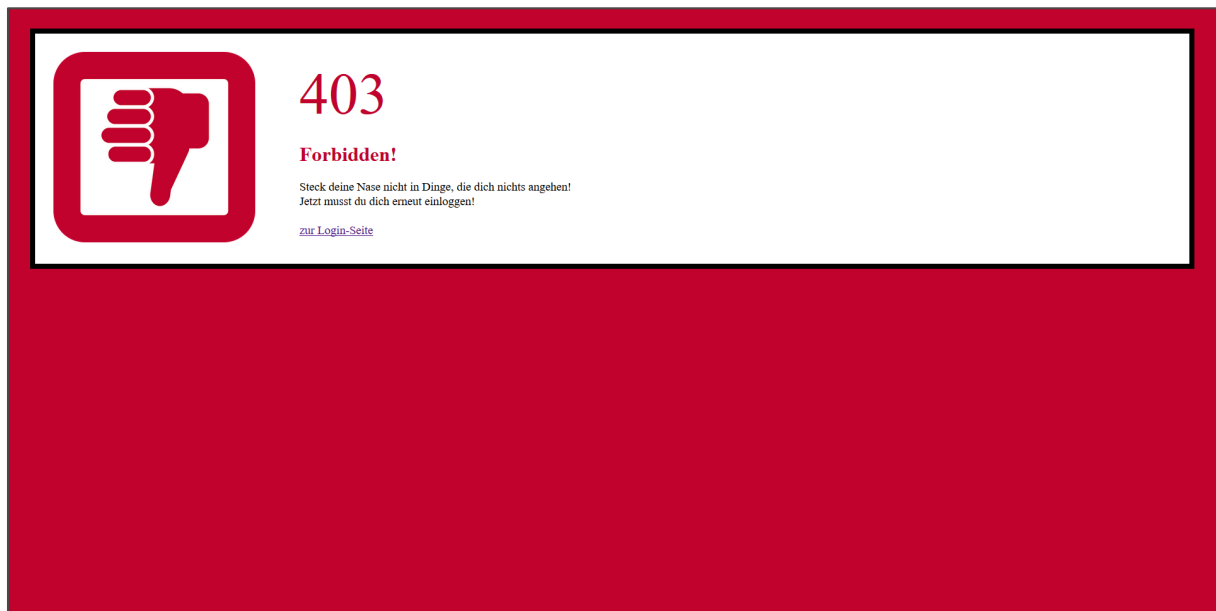


Abbildung 41: Access Denied-Seite

Möchte nun ein unzureichend qualifizierter Nutzer auf eine Seite zugreifen, wird dieser auf eine „Zugriff verweigert - Seite“ weitergeleitet, die den Benutzer benachrichtigt, dass er hier nicht mehr die nötigen Zugriffsrechte besitzt und somit gezwungen ist, sich erneut einzuloggen. Weiters läuft nach einer gewissen Zeit die Session ab und der Benutzer wird zurück auf die Login-Seite geleitet, um sich erneut einzuloggen.

Durch den auf jeder Seite zugänglichen „Ausloggen-Button“ wird die Session beendet und der Benutzer wieder auf die Login-Seite weitergeleitet.

11.2.4.2 Bildupload

Beim Bildupload werden die Bilder beim Erstellen eines Pakets oder eines Steps in das Serververzeichnis `/var/www/html/images/` gespeichert. Der Dateiname wird per Zufall generiert und besteht aus den Buchstaben a-z, A-Z und den Ziffern von 0-9. Die Dateiendung des hochgeladen Bildes wird ausgelesen und mit denen in der Datenbank abgeglichen. Falls es nicht einem herkömmlichen Dateiformat entspricht, wird dem Bild die Dateiendung `.jpg` angehängt.

11.2.4.3 Benachrichtigungsmitteilungen

Bei bestimmten Aktionen wird eine Mitteilung am rechten oberen Bildschirmrand angezeigt, um dem Nutzer ein visuelles Feedback zu geben. Eine Mitteilung wird nach jedem Erstellen, Bearbeiten oder Löschen angezeigt, ebenso, wenn das Formular geleert oder die eingegebenen Anmeldedaten inkorrekt sind.

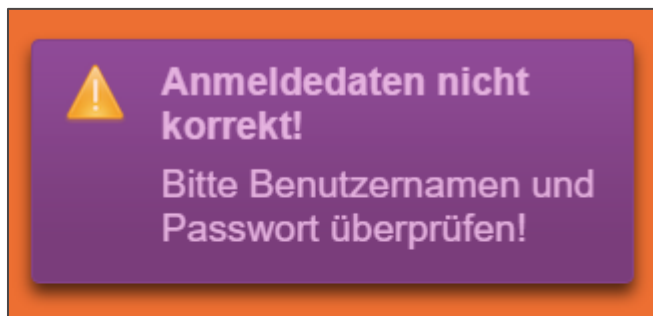


Abbildung 42: Info-Message

11.2.5 Umsetzung REST-Schnittstelle

11.2.5.1 REST-Requests bei EasyGym

Für jede Entität (Endnutzer, Firma, Pakete, Übungen, ...), die vom Server gespeichert wird, wurde ein REST-Endpoint, der die Requests empfängt, erstellt. In jedem dieser Endpoints sind DAOs (Data Access Objects) für den Zugriff auf die Daten in der Datenbank enthalten.

Folgende Aufrufe sind standardmäßig in einem Endpoint vorgeschrieben und implementiert:

- **findAll()**

Diese Methode liest alle zu dieser Entität vorhandenen Informationen aus der Datenbank aus und gibt sie dem Aufrufer zurück.

findAll() ist ein GET-Request und wird aufgerufen, sobald kein zusätzlicher Pfad angegeben wird.

Um alle Pakete zu erhalten wird zum Beispiel diese Request-URL verwendet:

<http://easygymaskoe.htl-perg.ac.at/EasyGymAskoe/rest/packs>

- **findById(id)**

Wie zuvor kurz erwähnt, sucht *findById(id)* die Entität mit der übergebenen ID aus der Datenbank und gibt sie zurück.

findById(id) ist ebenfalls ein GET-Request und wird dann aufgerufen, wenn im Zusatzpfad die ID enthalten ist.

Um ein bestimmtes Paket zu erhalten wird die folgende Request-URL aufgerufen:

<http://easygymaskoe.htl-perg.ac.at/EasyGymAskoe/rest/packs/1>

- **persist(entity)**

persist(entity) wird verwendet, um in der Datenbank die übergebene Entität zu speichern. Dabei ist die Entität noch nicht in der Datenbank vorhanden oder gespeichert.

Bei EasyGym kann der Client keine ganzen Entitäten erstellen, deshalb wurde *persist(entity)* für den Aufruf des Servers implementiert.



- **merge(entity)**

Im Gegensatz zu *persist(entity)* wird *merge(entity)* dazu verwendet, um in der Datenbank die übergebene Entität zu speichern oder upzudaten. Falls die Entität bereits existiert, werden geänderte Werte in der Datenbank gespeichert. Ist dies nicht der Fall, wird ein neuer Datensatz mit den übergebenen Werten angelegt.

Ein Endnutzer von EasyGym kann keine ganzen Entitäten erstellen oder updaten, deshalb wurde *merge(entity)* ebenso nur für den Serveraufruf implementiert.

- **delete(entity)**

delete(entity) wird dazu verwendet, dass eine Entität aus der Datenbank gelöscht wird.

Auch hier darf ein Endnutzer Entitäten nicht einfach löschen, deswegen wurde *delete(entity)* nur für den Server implementiert.

11.2.5.2 Konkrete REST-Requests von EasyGym

Diese nachfolgenden Requests werden derzeit ausschließlich vom Client abgesendet, um Daten vom Server anzufordern oder Daten am Server zu speichern.

GET-Requests

- Alle Pakete, die ein Endnutzer noch erwerben kann

Alle Pakete, die im Store angezeigt werden, hat der Benutzer noch nicht zu seinen Paketen hinzugefügt und kann sie somit kaufen.

- Alle Übungen in einem Paket

Um von der generellen Ansicht aller Pakete einzeln für ein aufgerufenes Paket die Übungen anzeigen zu können.

- Alle Pakete, die ein Endnutzer besitzt

Um Pakete spezifisch für einen bestimmten Endnutzer herauszufiltern und anzuzeigen.

- Alle Übungsschritte einer Übung

Um von der Übungsansicht bei der Auswahl einer bestimmten Übung einzelne Übungsschritte dieser Übung anzeigen zu können.

- Alle Gutscheine eines Endnutzers

Um bei der Anzeige von Gutscheinen jedem Benutzer individuell seine eigenen Gutscheine anzeigen zu können.



- Den gesamten Pfad eines Mediums

Dieser setzt sich aus Pfad am Server, sowie Dateinamen und Dateiendung zusammen.

PUT/POST-Requests

- Ein Paket für Endnutzer

Wird ein Paket vom Endnutzer erworben, wird es in die Liste der gekauften Pakete eingetragen.

- Passwort für Endnutzer ändern

Sobald ein Endnutzer sein Passwort ändern möchte, wird dies am Server durch das neue ersetzt.

- Namen eines Endnutzers ändern

Ebenso wie oben wird dieser Request benötigt, um den Namen eines Endnutzers am Server zu aktualisieren.

- Mit einem Gutschein ein Paket erwerben

Um ein Paket mit einem Gutschein zu bezahlen. Dabei wird der Gutschein um die Paketkosten verringert.

- Eine Übung als „gemacht“ markieren

Wenn ein Endnutzer eine Übung erledigt hat, kann er sie als „gemacht“ markieren. Diese Daten werden an den Server geschickt.

- Anmelddaten überprüfen

Wenn sich ein Endnutzer einloggt, werden die Daten überprüft und zurückgesendet, ob er berechtigt ist, sich einzuloggen.

11.3 Android-Applikation

Die Android-Applikation wurde in der Entwicklungsumgebung Android Studio verwirklicht. Als offizielle Entwicklungsumgebung für Android-Applikationen bringt dies Vorteile, wie zum Beispiel eine sehr große, motivierte Community oder eine ausführliche Dokumentation der Technologie Android mit sich.

11.3.1 Idee

Die mobile Applikation ist primär für die Kunden der Kunden gedacht. Ein Dienstleister erwirbt unser Produkt und befüllt es mit eigenen Informationen in Form von Paketen. Diese Informationen können die Klienten des Dienstleisters in der mobilen Applikation abrufen.

Bevor ein User auf Inhalte zugreifen kann, muss er sich über eine Login-Oberfläche in der Applikation einloggen. Die Zugangsdaten zur Applikation erhält ein Benutzer vom zuständigen Dienstleister. Im Fitnessstudio kann beispielsweise der Personal-Trainer der Dienstleister sein und sein Klient ist der Endnutzer. Ist ein User erfolgreich durch das Login-Verfahren gekommen, befindet er sich auf der Hauptseite der Applikation, der Übersicht der eigenen Pakete.

Zu Beginn besitzt ein Benutzer ein Beispielpaket, das nicht vom Dienstleister zur Verfügung gestellt wird, sondern von den Entwicklern. Es dient als Starter-Paket und soll dem Benutzer zeigen, wie diese Anwendung funktioniert. Mit einem Klick auf ein Paket bekommt man eine Übersicht der darin enthaltenen Übungen. Für den Fall, dass diese Übungen absolviert wurden, kann man diese als erledigt markieren. Wird eine Übung als erledigt markiert, wird der Fortschritt eines Pakets erhöht. Dieser lässt sich auf der Startseite aus jedem Paket auslesen. Zusätzlich zu der Möglichkeit, ein Paket als erledigt zu markieren, kann sich der Benutzer auch noch eine genauere Information anzeigen lassen. Die genauere Information beinhaltet die Kategorien der Muskelgruppen, welche durch die ausgewählte Übung beansprucht werden, sowie einen Beschreibungstext, der den Sinn und den Zweck einer Übung erläutert. Wird eine Übung ausgewählt, kommt man auf eine neue Ansicht, welche die Ausführungsschritte beinhaltet. Der User kann sich also mithilfe von Wisch-Gesten durch die einzelnen Schritte bewegen und zu jedem Ausführungsschritt ein Beschreibungsbild und einen Beschreibungstext ansehen.

Neben diesem Starter-Paket soll es natürlich auch möglich sein, weitere Pakete zu erwerben. Dazu benötigt es einen Store, in dem einem User nur die Pakete angezeigt werden, die der Benutzer noch nicht erworben hat. Als Zahlungsmittel werden Credits verwendet, die ein Dienstleister für seine Kunden erstellen kann. Um dem Benutzer den Zahlungsvorgang einfacher zu gestalten, werden ihm im Kaufdialog die möglichen Gutscheine als Auswahlmöglichkeiten präsentiert. Zusätzlich ist es möglich, im Store weitere Informationen zu einem Paket anzusehen.

Eine Übersicht über diese Zahlungsmittel (Gutscheine) erhält der Benutzer auf einer eigenen View, die alle verfügbaren Gutscheine anzeigt. Das sind also jene Gutscheine, die dem Benutzer zugewiesen sind und die mehr als null verbleibende Credits gespeichert haben.

Zusätzlich zu den bereits erwähnten Ansichten gibt es noch Profileinstellungen, in denen der Benutzer eigene Informationen, wie zum Beispiel die Zugangsdaten, verändern kann und eine Anzeige, in der nähere Informationen zum Projekt zu finden sind.

11.3.2 Erklärungen

Eine Android-Anwendung setzt sich im Normalfall aus mehreren Activities zusammen.

11.3.2.1 Activity

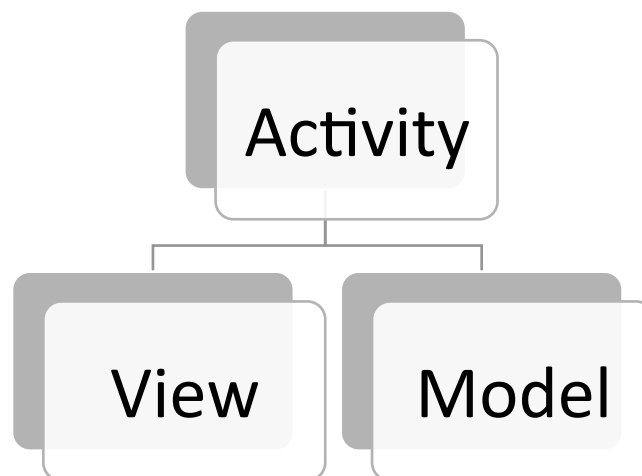


Abbildung 43: Aufbau Activity (vgl. individuapp.com, 2017)

In der mobilen Applikation stellt eine Activity eine Bildschirmseite dar. Sie besteht aus einer View und einem Model. Das Model beinhaltet die Logik hinter einer Activity und die View das Layout (Design). Die Verwaltung der Activities funktioniert wie ein Stapel. Sobald eine Activity gestartet wird, kommt sie an die Spitze des Stapels. Drückt man die Zurück-Taste, wird die zuvor aktive Activity erneut aktiv, wenn man die Standard-Aktion dieses Events nicht überschrieben und selbst konfiguriert hat. Im Fall, dass zu viele Activities die Leistung einer Anwendung beeinträchtigen, werden inaktive Activities von der VM beendet. (vgl. individuapp.com, 2017)



Methode	Beschreibung
onCreate()	Wird aufgerufen, wenn eine Activity das erste Mal gestartet wird. Hier sollten alle statischen Inhalte konfiguriert werden (zum Beispiel Daten einer Liste zuweisen, Views erstellen, etc.).
onRestart()	Eine gestoppte Activity (<i>onStop()</i>) wird erneut aufgerufen.
onStart()	Die Activity ist ab diesem Zeitpunkt für den Benutzer sichtbar.
onResume()	Die betroffene Activity befindet sich ab dem Aufruf dieser Methode ganz oben am Activity-Stapel. Das heißt, ab diesem Zeitpunkt ist es dem Benutzer möglich, mit der Activity zu interagieren.
onPause()	Wird aufgerufen, wenn man sich auf einer Activity befindet und eine bereits zuvor gestartete Activity aufruft.
onStop()	Eine bereits zuvor gestartete Activity reiht sich über die betroffene Activity. Ab diesem Zeitpunkt ist die Activity für den Benutzer nicht mehr sichtbar.
onDestroy()	Die betroffene Activity wird „zerstört“. Kann entweder absichtlich durch einen Benutzer geschehen, indem die <i>finish()</i> Methode der betroffenen Activity aufgerufen wird, oder durch die VM, die die Activity auf Grund einer Ressourcenknappheit schließt.

Tabelle 3: Activity Methoden

(vgl. android.com, 2017)

11.3.3 Umsetzung

11.3.3.1 Auflistung – Activities in der Anwendung:

- Login
- Eigene Pakete
- Übungen
- Ausführung
- Store
- Verfügbare Gutscheine
- Profil
- Passwort bearbeiten
- Über uns

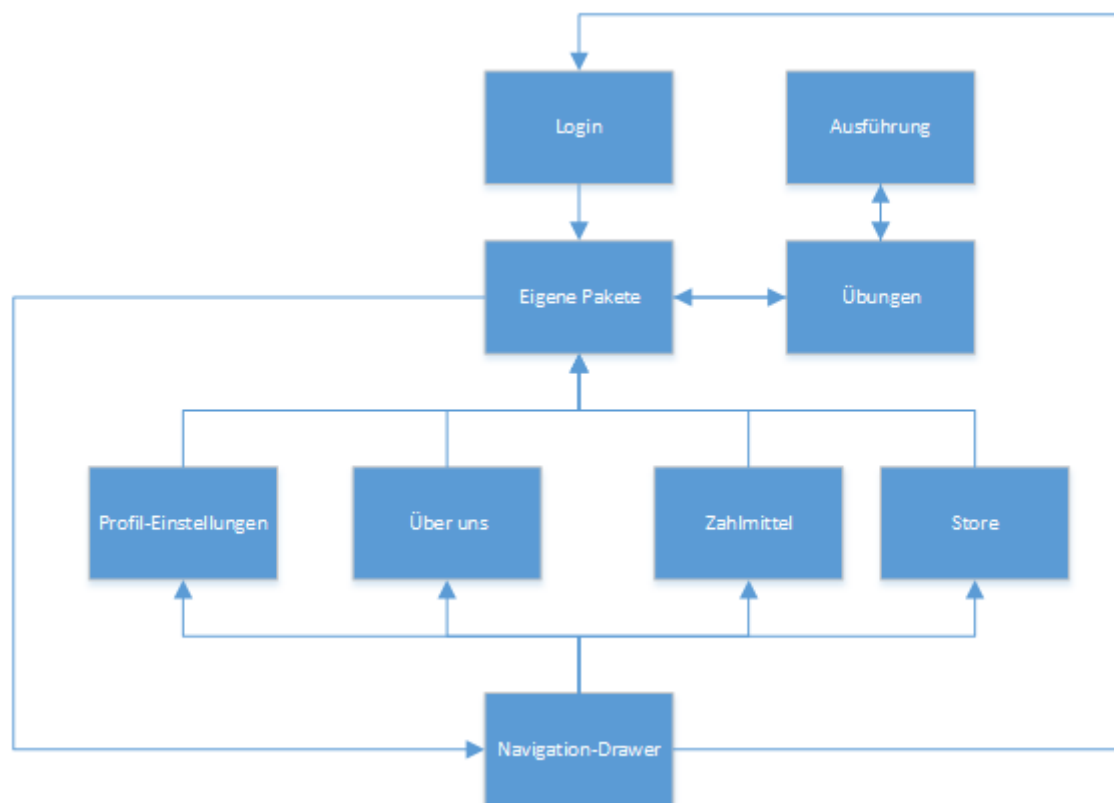


Abbildung 45: Funktionale Idee - Android Applikation

11.3.3.2 Vorkonfiguration der Anwendung

Jede Android Version besitzt ein **API Level** (eindeutige Nummer). Das heißt eine Version von Android steht in direkter Verbindung zu einem API Level. Der Name einer Version ändert sich lediglich bei größeren Updates der Versionen. (vgl. xamarin.com, 2017)

Version	Codename	API	Distribution
2.3.3 - 2.3.7	Gingerbread	10	1.0%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	1.0%
4.1.x	Jelly Bean	16	3.7%
4.2.x		17	5.4%
4.3		18	1.5%
4.4	KitKat	19	20.8%
5.0	Lollipop	21	9.4%
5.1		22	23.1%
6.0	Marshmallow	23	31.3%
7.0	Nougat	24	2.4%
7.1		25	0.4%

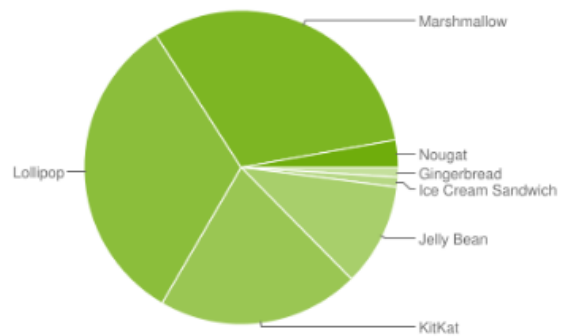


Abbildung 46: Android-Versionen Verteilung (android.com, 2017)

Die Grafik zeigt die Verteilung aller unterschiedlichen Android-Smartphone-Variationen (Samsung Galaxy S5, Samsung Galaxy S6, Sony Xperia Z1, ...), die durch eine Android-Version erreicht werden.

In den vergangenen Jahren hat sich für Android-Anwendungen im Bereich des Designs ein Standard etabliert. Ab API Level 21 (5.0 – Lollipop) ist es möglich, das Material-Design von Google zu verwenden.

11.3.3.3 Material Design

Das Material Design ist eine Designsprache, welche durch eine Anlehnung an das Flat Design sehr minimalistisch gehalten ist. Die Inhalte werden meistens in kartenähnlichen Flächen präsentiert. Durch Animationen und Schatten wird versucht, die wichtigsten Komponenten hervorzuheben. Verpackt werden die Designs in verschiedenen Themes, die der Benutzer je nach Bedarf auswählen kann. Zusätzlich zu den Themes gibt es im Material-Design auch eine RecyclerView und eine CardView. (vgl. android.com, 2017)

11.3.3.4 RecyclerView

Die RecyclerView ermöglicht es, ein eigenes Design für Listen zu entwerfen und bringt durch Wiederverwendung von nicht mehr sichtbaren Listenelementen Performance-Steigerungen mit sich. Beispielsweise könnten auf einem 5“ Smartphone sechs Listeneinträge Platz haben. Die RecyclerView erstellt ab einer gewissen Anzahl keine neuen Listenelemente mehr. Es werden die Inhalte von nicht sichtbaren Listeneinträgen verworfen und mit neuen Daten befüllt. So kann es sein, dass eine RecyclerView maximal 20 Listenelemente erstellt. Wird nun das 21. Element geladen, verwirft die RecyclerView ein nicht mehr sichtbares Element, befüllt es mit den Daten des neuen Eintrags und hängt es je nach Bedarf oben oder unten an die Liste an. (vgl. android.com, 2017)

11.3.3.5 CardView

Die CardView wird verwendet, um wichtige Informationen mit selbst definiertem Design innerhalb visuell getrennter Karten anzuzeigen. Dadurch bleiben Activities sehr übersichtlich und ansprechend für den Betrachter. (vgl. android.com, 2017)

Um die Applikation nach den derzeitigen Richtlinien gestalten zu können, wurde das API-Level 21 ausgewählt. Damit haben wir die Möglichkeit, für ~67% (API-Level 21+) aller unterschiedlichen Android-Geräte ein standardisiertes Design zur Verfügung zu stellen.

11.3.4 Ergebnisse

11.3.4.1 Navigation-Drawer

Der Navigation-Drawer hilft dem Benutzer, sich in der mobilen Applikation auf verschiedene Ansichten begeben zu können. Er ist sozusagen das Menü der Applikation.

Er besitzt Links auf die Ansicht der eigenen Pakete und der Pakete im Store, auf die Übersicht der verfügbaren Gutscheine, die Profilsansicht, den Login-Screen sowie auf die Informationen zum Projekt.

Im oberen Teil des Navigation-Drawers sind das Logo von EasyGym, die E-Mail-Adresse, sowie der Name des eingeloggtten Benutzers zu finden.

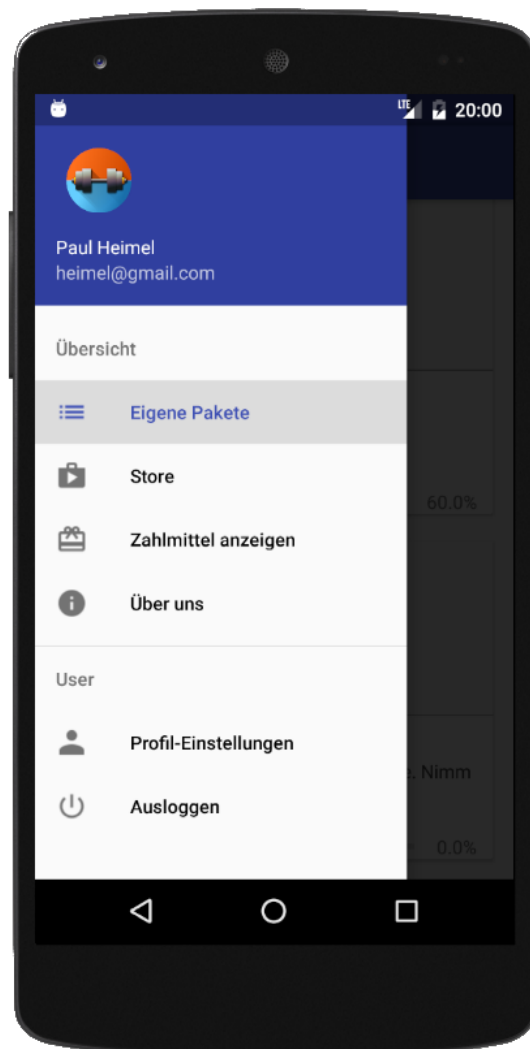


Abbildung 47: Navigation-Drawer

11.3.4.2 Login

Der Endnutzer verwendet diese Activity, um sich in der Applikation anzumelden. Durch die korrekte Eingabe einer Email inklusive zugehörigem Passwort kann sich der Benutzer in der Anwendung authentifizieren. Die Zugangsdaten erhält der Benutzer vom zuständigen Mandanten, der auch die Inhalte der Anwendung verwalten kann. Nach dem Absenden eines Login-Versuches durch den „Anmelde-Button“ wird ein Request an den Server gesendet. Dem Client wird während der Durchführung des Requests ein Progress-Dialog in Form eines Kreises angezeigt, der dem Benutzer zeigt, dass sich etwas in Arbeit befindet. Tritt ein Fehler auf (zum Beispiel eine falsche Eingabe der Zugangsdaten oder ein Server-Error), erscheint am unteren Bildschirm Rand ein Toast. Dieser Toast teilt dem Benutzer mit, dass etwas nicht funktioniert hat. Bei einer richtigen Eingabe wird der Benutzer zu den „Eigenen Paketen“ weitergeleitet.

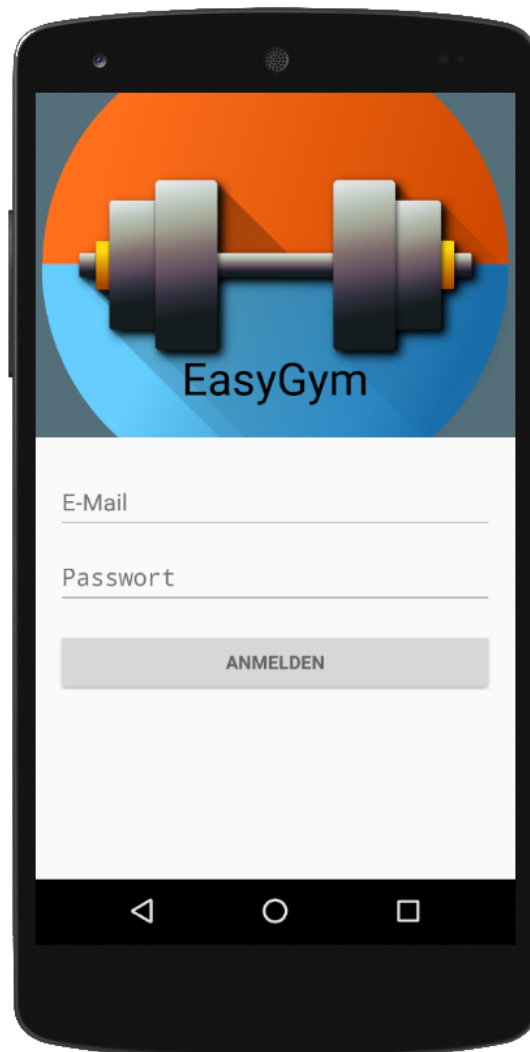


Abbildung 48: Login

11.3.4.3 Eigene Pakete

Der Benutzer sieht in dieser Activity seine Pakete. Dazu gehören das Starter-Paket, welches von den Entwicklern als Beispielpaket zur Verfügung gestellt wird und die im Store erworbenen Pakete. Die Pakete werden als einzelne Cards einer CardView dargestellt. Per Klick auf eine Card kommt man zu den im Paket enthaltenen Übungen. Um die Performance der Anwendung zu erhöhen, befindet sich diese CardView innerhalb einer RecyclerView. Während die Daten vom Server geladen werden, wird ein Progress-Dialog mit dem Text „Pakete werden geladen ...“ angezeigt.

Eine Card enthält ein Titelbild, einen Titel, einen Beschreibungstext sowie eine Fortschrittsanzeige.

Über den Fortschrittsbalken mit Prozentangabe erfährt der User, wie viele Übungen dieses Pakets bereits durchgeführt wurden.

Durch einen Klick auf die drei Balken im linken oberen Eck öffnet sich der Navigation-Drawer.

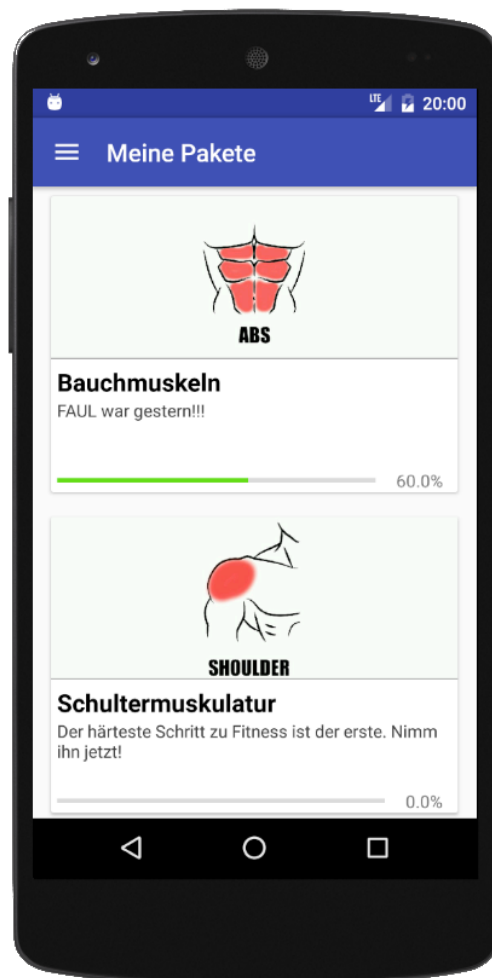


Abbildung 49: Meine Pakete

11.3.4.4 Übungen

Der Benutzer kommt durch einen Klick auf ein erworbenes Paket zu dieser Ansicht. Hier werden alle im Paket enthaltenen Übungen in einer ListView angezeigt. Die Liste befindet sich in einer RecyclerView. Während die Daten geladen werden, wird ein Progress-Dialog mit dem Text „Übungen werden geladen ...“ angezeigt.

Ein Listenelement enthält einen Titel, eine Kurzbeschreibung der Übung, einen Info-Button, sowie eine Checkbox.

Durch einen Klick auf den Info-Button öffnet sich ein Dialog. Dieser Dialog zeigt eine detaillierte Beschreibung sowie die Muskelgruppen, welche durch die Übung gestärkt werden, an.

Durch einen Klick auf die Checkbox erhöht sich der Fortschritt der absolvierten Übungen eines Pakets. Der Benutzer hat dadurch einen Überblick, welche Übungen bereits absolviert sind und welche noch gemacht werden müssen.

Der Pfeil im linken oberen Eck leitet auf die Ansicht „Eigene Pakete“ weiter.

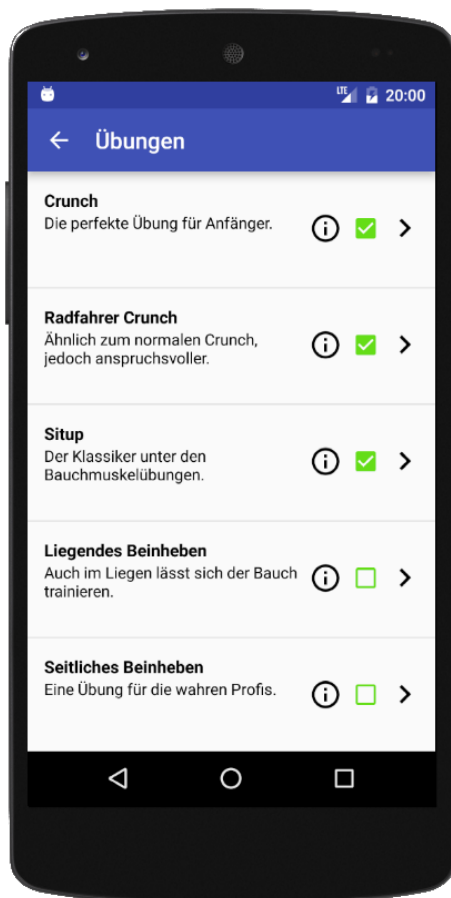


Abbildung 51: Übungen

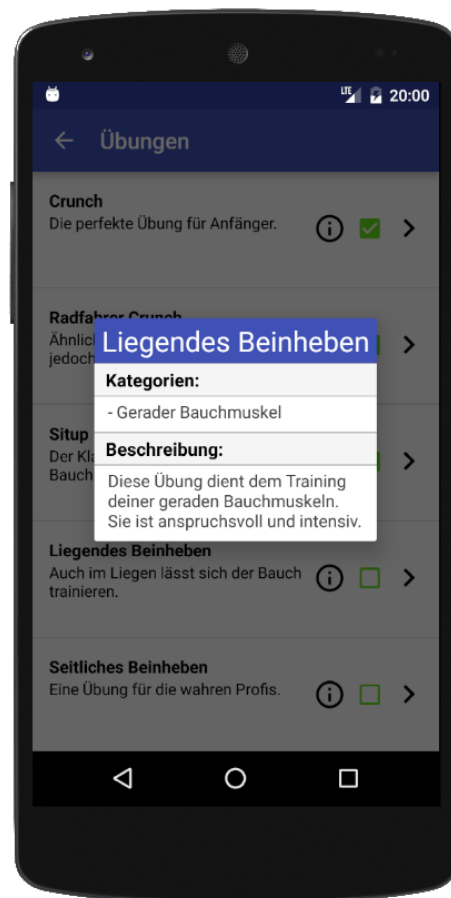


Abbildung 50: Übungen-Info

11.3.4.5 Ausführung

Der Benutzer kommt durch einen Klick auf eine Übung zu dieser Ansicht. Die einzelnen Ausführungsschritte werden in einem ViewPager angezeigt. Dieser besteht aus mehreren Fragments, die durch Wischgesten nach links und rechts geschoben werden können. Ein Fragment beschreibt in dieser Anwendung einen Ausführungsschritt und besteht aus Beschreibungsbild und -text. Während die Daten geladen werden, wird ein Progress-Dialog mit dem Text „Schritte werden geladen ...“ angezeigt.

Am unteren Ende dieser Activity befindet sich eine Anzeige in Form von Punkten, die dem Benutzer zeigt, auf welchem Ausführungsschritt er sich befindet. Gibt es zum Beispiel fünf Schritte und man ist gerade beim zweiten, so sind der erste, dritte, vierte und fünfte Punkt anders formatiert als der zweite Punkt.

Durch den Pfeil links oben kommt man auf die Ansicht „Übungen“.

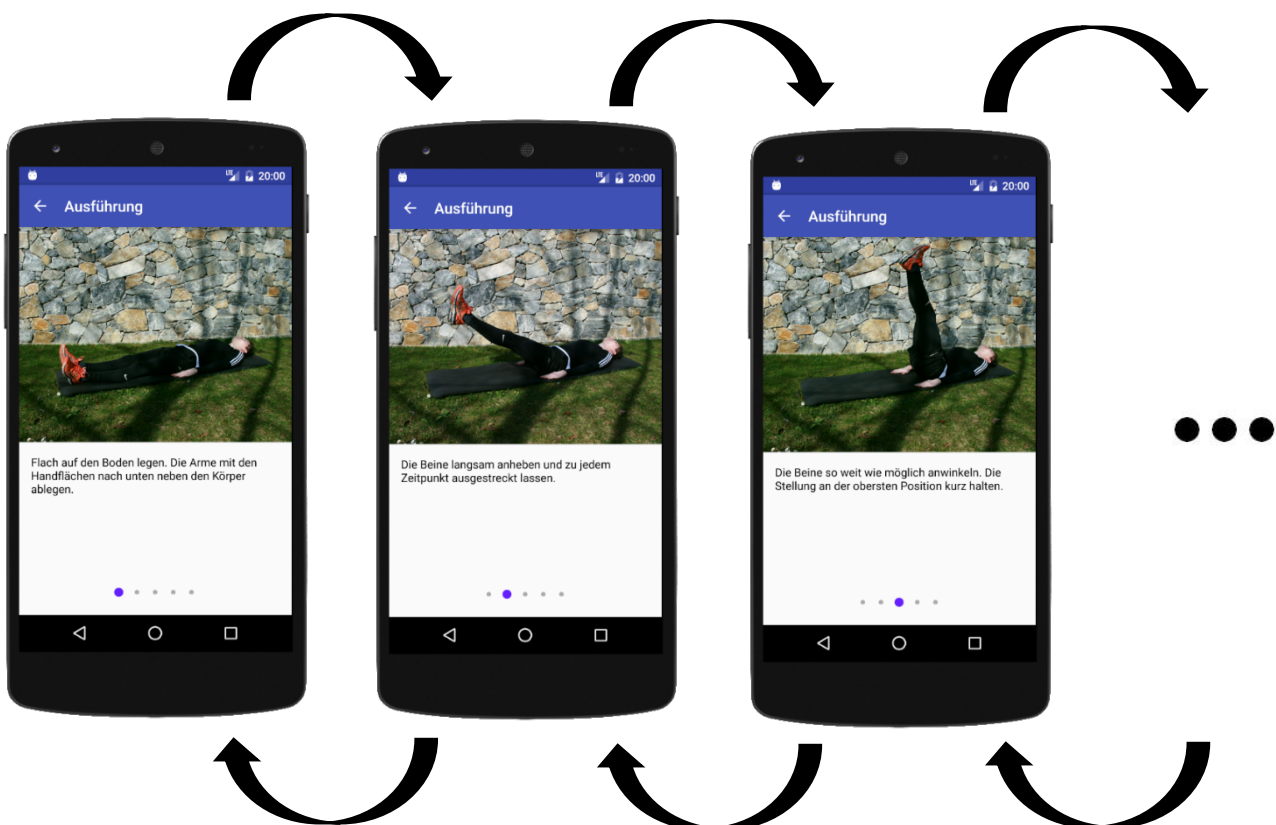


Abbildung 52: Ausführungsschritte

11.3.4.6 Store

Der Benutzer sieht auf dieser Activity alle noch nicht erworbenen Pakete, die für ihn erhältlich sind. Die Pakete werden als einzelne Card einer CardView dargestellt, welche sich wiederum in einer RecyclerView befindet. Während die Daten geladen werden, wird ein Progress-Dialog mit dem Text „Pakete werden geladen ...“ angezeigt.

Eine Card enthält ein Titelbild, einen Titel, eine Beschreibung, die Kosten für das Paket, die Anzahl der enthaltenen Übungen sowie einen Button zum Kaufen des Pakets und einen Button für weitere Informationen.

Durch einen Klick auf den Button für die weiteren Informationen erhält der Benutzer eine detaillierte Beschreibung des Pakets sowie die Namen der darin enthaltenen Übungen.

Klickt der Benutzer auf den Button, um das Paket zu erwerben, öffnet sich ein Dialog, in dem man einen seiner Gutscheine für den Erwerb auswählen kann. Sollte ein Gutschein zu wenig Credits enthalten, wird er ausgegraut und ist nicht auswählbar. Der Benutzer hat in dem Dialog die Möglichkeit, die Aktion abzubrechen oder mit einem ausgewählten Gutschein das Paket zu erwerben.

Ist der Kauf erfolgreich, wird am unteren Bildschirmrand eine Bestätigung in Form eines Toasts angezeigt. Der Pfeil im linken oberen Eck leitet auf die Ansicht „Eigene Pakete“ weiter.

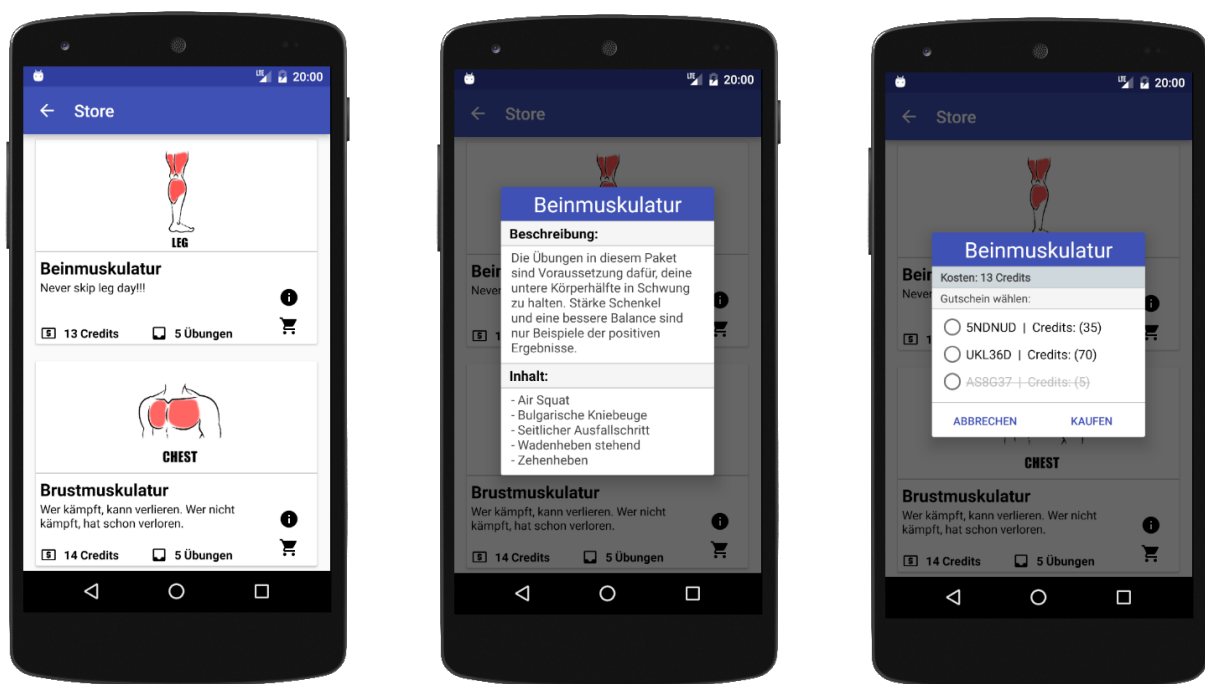


Abbildung 53: Store

11.3.4.7 Verfügbare Gutscheine

Der Benutzer erhält auf dieser Activity eine Übersicht über die ihm zugewiesenen Gutscheine. Ein Gutschein wird als Card einer CardView angezeigt. Diese CardView befindet sich wiederum in einer RecyclerView. Während die Daten geladen werden, wird ein Progress-Dialog mit dem Text „Zahlungsmittel werden geladen ...“ angezeigt.

Eine Card enthält den Namen des Gutscheins sowie die Anzahl der darauf gespeicherten Credits.

Durch einen Klick auf den Pfeil links oben wird man auf die Ansicht „Eigene Pakete“ weitergeleitet.

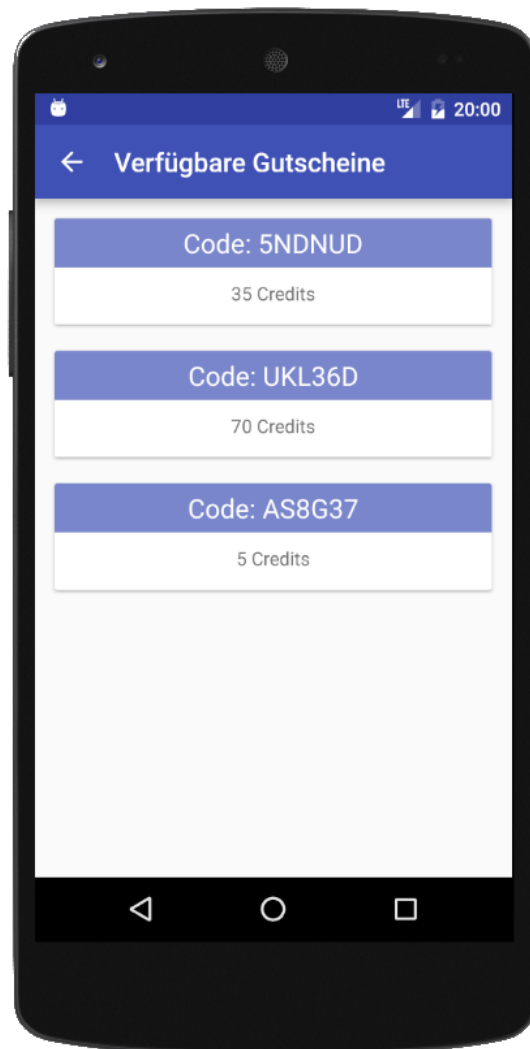


Abbildung 54: Gutscheine

11.3.4.8 Profil

Hier kann der Benutzer sein Profil verändern. In unserer Anwendung ist das sein Vor- und Nachname. Zusätzlich zu den veränderbaren Werten sieht man noch die E-Mail des eingeloggtten Users.

Rechts unten befindet sich ein Button zum Speichern der Änderungen. Nach einer erfolgreichen Speicherung wird am unteren Rand eine Bestätigung in Form eines Toasts angezeigt. Läuft bei der Speicherung etwas schief oder werden die Daten nicht geändert, ist auch eine dementsprechende Benachrichtigung zu finden.

Durch einen Klick auf den Button „Passwort ändern“ wird man auf eine neue Activity weitergeleitet.

Der Pfeil im linken oberen Eck stellt einen Link zur Ansicht „Eigene Pakete“ dar.

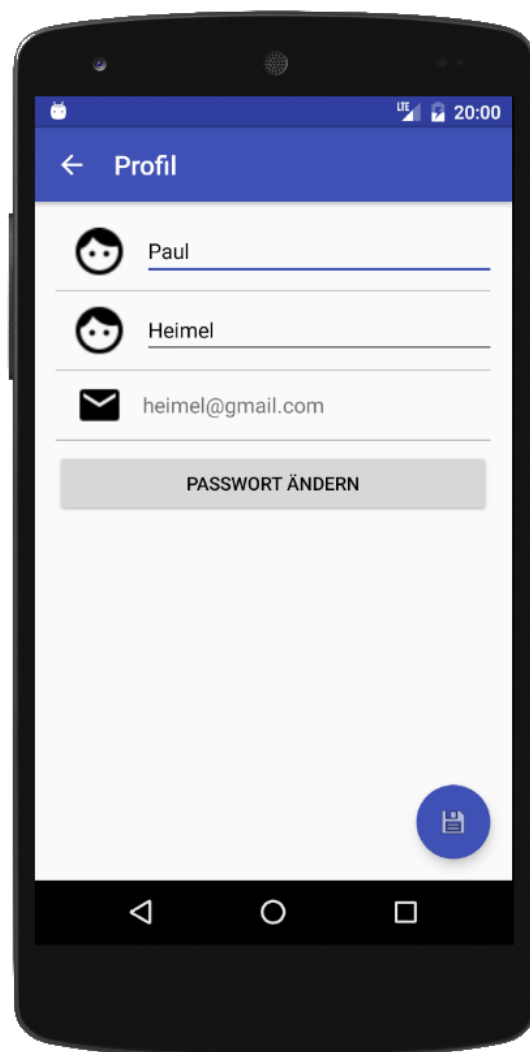


Abbildung 55: Profil

11.3.4.9 Passwort bearbeiten

Hier kann der Benutzer sein Passwort abändern. Voraussetzung dafür ist eine Eingabe des aktuellen Passworts, sowie eine doppelte Eingabe des neuen Passworts.

Durch einen Klick auf den „Speichern“ Button wird bei Erfüllung aller Kriterien das Passwort des Benutzers geändert. Je nach Erfüllung oder Nichterfüllung der gewünschten Kriterien beziehungsweise der durchgeführten Serverarbeit, wird am unteren Bildschirmrand eine entsprechende Benachrichtigung in Form eines Toasts angezeigt.

Durch das Anklicken des Pfeils links oben wird man auf die Profilansicht weitergeleitet.

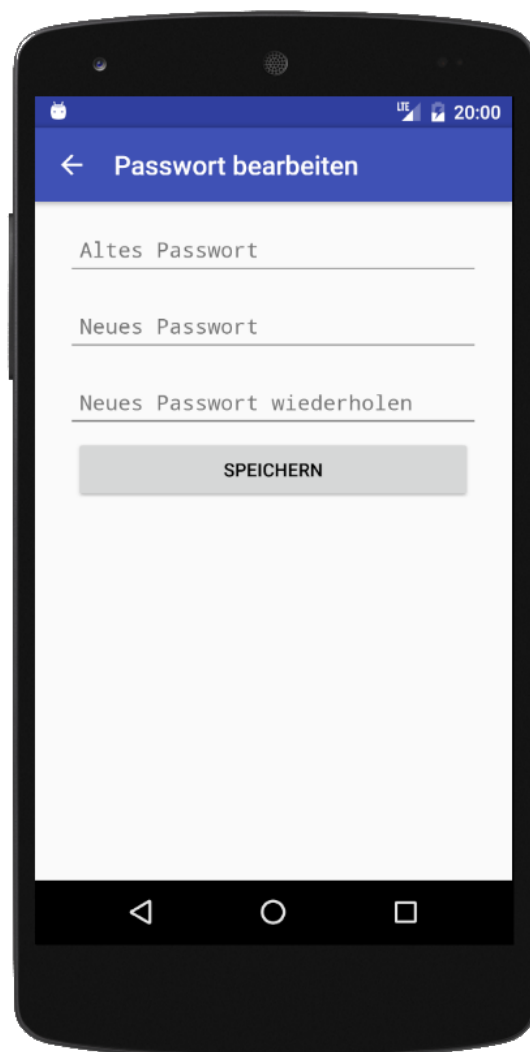


Abbildung 56: Passwort bearbeiten

11.3.4.10 Über uns

Hier findet der Benutzer Informationen zum Projekt und zum Entwicklungsteam. Ebenso befindet sich auf dieser Activity die Danksagung des Projektteams. Mit dem Pfeil links oben kann man auf die Hauptseite („Eigene Pakete“) wechseln.



Abbildung 57: Über uns

11.4 Datenbank

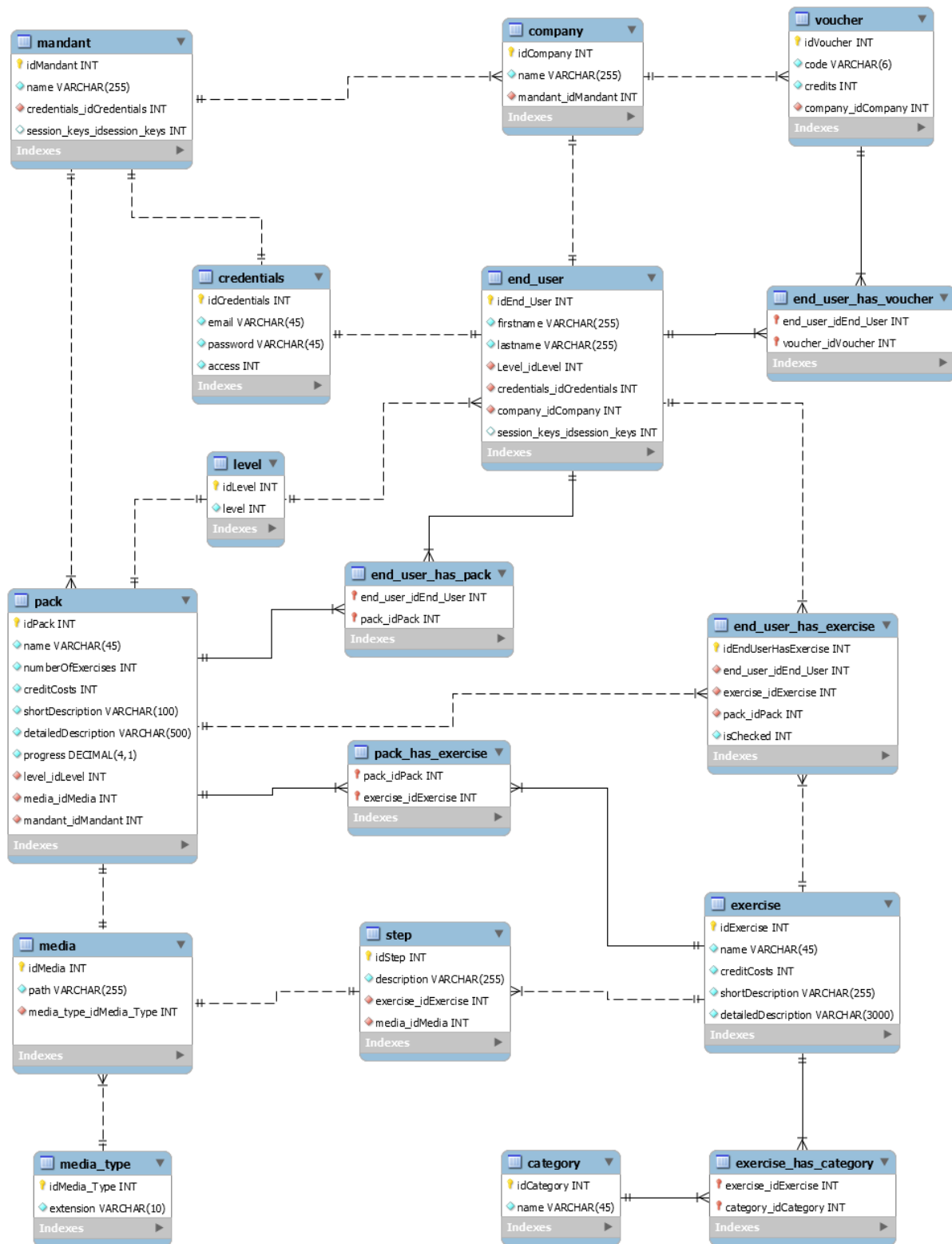


Abbildung 58: gesamtes Datenmodell von EasyGym

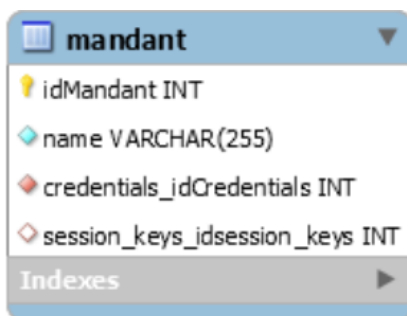
Die zuvor eingefügte Grafik ist das Datenmodell unserer Arbeit. Erstellt wurde es mit der MySQL Workbench. Auch Modelländerungen und zum Teil Tests wurden mit diesem Modellierungswerkzeug durchgeführt. Durch das Forward-Engineering wird das aktuelle Datenmodell auf eine zuvor angelegte Verbindung übertragen. Diese Verbindung verweist einerseits auf die lokale Wildfly-Instanz und andererseits auf den Server.

Über SQL-Abfragen können Daten, die in den Tabellen gespeichert sind, ausgelesen und somit alle Änderungen am Server kontrolliert werden.

Im folgenden Absatz ist ein Datenlexikon mit den wichtigsten Tabellen des Modells zu finden.

11.4.1 Datenlexikon

Mandant:



Column	Type
idMandant	INT
name	VARCHAR(255)
credentials_idCredentials	INT
session_keys_idsession_keys	INT

Abbildung 59: Tabelle Mandant

Die Mandanten-Tabelle speichert alle Mandanten, die dieses Produkt erwerben. Jeder Mandant erhält bei der Anmeldung einen Namen sowie Zugangsdaten, die er durch einen Fremdschlüssel mit der ID aus der Tabelle Credentials erhält. Mit diesen Credentials kann er sich in der Administrationsoberfläche anmelden.

Firma:



Column	Type
idCompany	INT
name	VARCHAR(255)
mandant_idMandant	INT

Abbildung 60: Tabelle Firma

Die Tabelle Company dient als Zwischenstück zwischen Mandant und Endnutzer. Vertreibt ein Mandant seine Inhalte an Firmen, die wiederum Endnutzer haben, kommt diese Tabelle ins Spiel und eine Firma wird erstellt. Vertreibt ein Mandant Inhalte direkt an Endnutzer wird ein zum Mandant identischer Company-Tabelleneintrag erstellt, der als Ersatzeintrag dient.

Endnutzer:

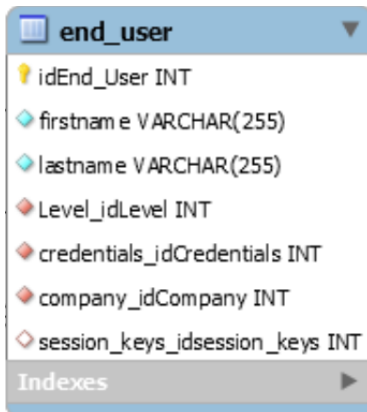


Abbildung 61: Tabelle Endnutzer

Die Endnutzer, die von einem Mandanten oder einer Firma erstellt werden, sind in dieser Tabelle gespeichert. Bei der Erstellung wird einem Endnutzer sowohl Vorname und Nachname zugeteilt, als auch die Firma beziehungsweise der Mandant zu dem der Endnutzer gehört. Weiters werden Zugangsdaten, mit denen sich der Benutzer in der mobilen Applikation einloggen kann, zugewiesen.

Gutschein:

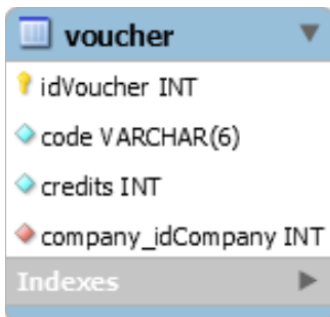


Abbildung 62: Tabelle Gutschein

Gutscheine können von einem Mandanten beziehungsweise in weiterer Folge einer Firma erstellt werden. Ein Gutschein besitzt einen einzigartigen Code und eine Anzahl von Credits. Mit diesen Credits kann der Gutschein-Besitzer Pakete erwerben. Ein Gutschein wird bei genau einer Firma beziehungsweise einem Mandanten gespeichert, da diese/r den Gutschein erstellt hat. Ein Gutschein wird einem Benutzer zugewiesen.

Paket:

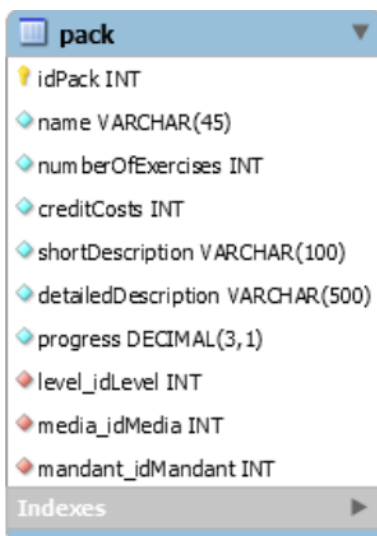
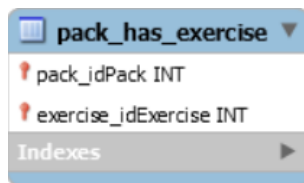


Abbildung 63: Tabelle Paket

In der Paket-Tabelle werden alle Pakete, die von verschiedenen Dienstleistern erstellt werden, gespeichert. Um die Pakete dem Ersteller zuweisen zu können, besitzen diese eine Mandanten-ID. Ein Paket wird durch einen Namen, die Anzahl der enthaltenen Übungen, die Credit-Kosten, eine Kurzbeschreibung, eine detaillierte Beschreibung, den Fortschritt im Sinne der gemachten Übungen sowie das Titelbild beschrieben.

Übungen im Paket:



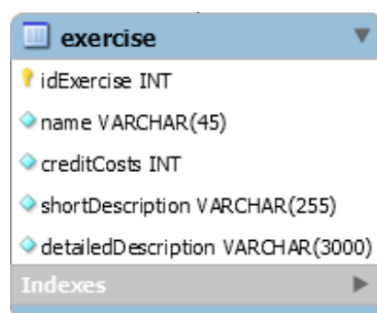
Field	Type
pack_idPack	INT
exercise_idExercise	INT

Indexes

Zwischen Paketen und Übungen befindet sich eine n:m Beziehung, die als Fremdschlüssel die beiden IDs beinhaltet und somit einem Paket eine Übung eindeutig zuweist.

Abbildung 64: Beziehungstabelle zwischen Paket und Übung

Übung:

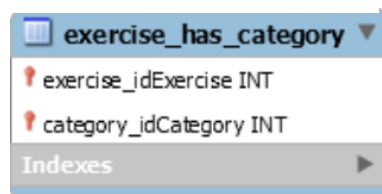


Field	Type
idExercise	INT
name	VARCHAR(45)
creditCosts	INT
shortDescription	VARCHAR(255)
detailedDescription	VARCHAR(3000)

Indexes

In dieser Tabelle werden alle Übungen gespeichert. Legt man eine Übung an, muss man einen Namen, die Kosten für diese Übung in der Einheit Credits, sowie eine Kurz- und Detailbeschreibung vergeben. Darüber hinaus werden einer Übung Muskelgruppen, die durch die Übung gestärkt werden, in Form einer Beziehungstabelle zwischen einer Übung und den Muskelgruppen, zugewiesen.

Abbildung 65: Tabelle Übungen

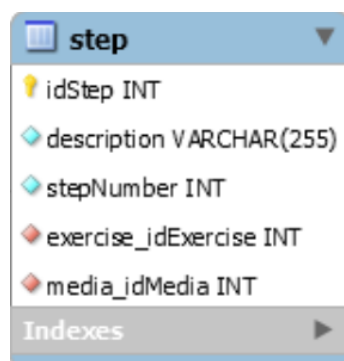


Field	Type
exercise_idExercise	INT
category_idCategory	INT

Indexes

Abbildung 66: Beziehungstabelle zwischen Übung und Kategorie

Übungsschritt:



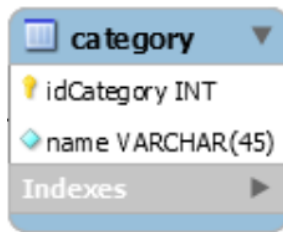
Field	Type
idStep	INT
description	VARCHAR(255)
stepNumber	INT
exercise_idExercise	INT
media_idMedia	INT

Indexes

Um eine Übung sinnvoll für einen Endnutzer darstellen zu können, benötigt sie Ausführungsschritte. Diese werden in der Tabelle mittels einer Beschreibung, einer Schrittnummer, der zugehörigen Übung sowie eines Bildes, welches einen Ausführungsschritt beschreibend darstellt, gespeichert.

Abbildung 67: Tabelle Übungsschritt

Kategorie:

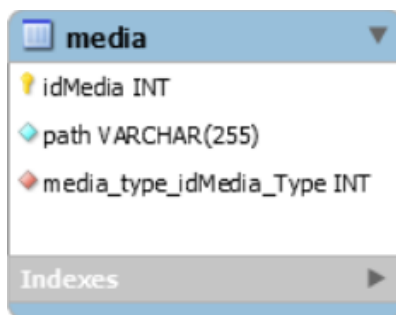


category
idCategory INT
name VARCHAR(45)
Indexes

Übungen werden immer verschiedenen Kategorien zugewiesen. In unserer Arbeit ist unter einer Kategorie ein Funktionsbereich, beispielsweise eine Muskelgruppe, zu verstehen. Eine Kategorie wird durch die ID identifiziert und durch einen Namen beschrieben.

Abbildung 68: Tabelle Kategorie

Medium:

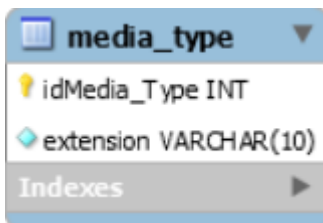


media
idMedia INT
path VARCHAR(255)
media_type_idMedia_Type INT
Indexes

Beim Erstellen von Paketen oder Übungsschritten muss der Benutzer auch Medien hochladen. Diese Bilder werden mit dem Pfad inklusive des Dateinamens gespeichert. Je nach Dateiendung (zum Beispiel .jpg, .png, ...) wird ein Fremdschlüssel zum entsprechenden Media-Typ erstellt.

Abbildung 69: Tabelle Medium

Medien-Typen:

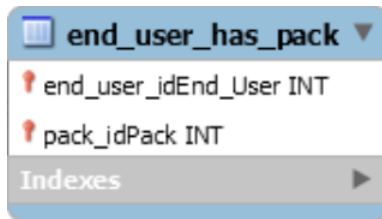


media_type
idMedia_Type INT
extension VARCHAR(10)
Indexes

Abbildung 70: Tabelle Medien-Typ

Die verschiedenen Typen, die ein Medium besitzen kann, werden in der Tabelle „Medien Typ“ gespeichert. Sie kann vom Nutzer nicht geändert werden und wird benötigt, um zu überprüfen, ob es sich dabei um ein gängiges Dateiformat handelt. Zu einem späteren Zeitpunkt wäre es möglich, EasyGym so zu erweitern, dass zum Beispiel auch Videos angezeigt werden können.

Pakete eines Endnutzers:

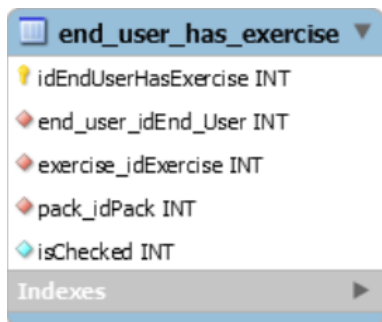


end_user_has_pack
end_user_idEnd_User INT
pack_idPack INT
Indexes

Welche Pakete ein Endnutzer nun besitzt, wird in der Beziehungstabelle zwischen einem Endnutzer und einem Paket ausgewiesen. Hier werden alle Paket-IDs, die ein Endnutzer erwirbt, mit der ID des Endnutzers gespeichert.

Abbildung 71: Beziehungstabelle zwischen Endnutzer und Paket

Übungen eines Benutzers:

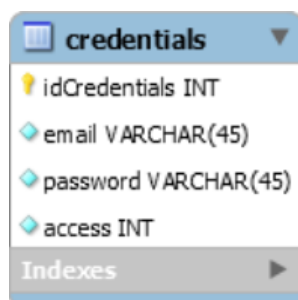


end_user_has_exercise
idEndUserHasExercise INT
end_user_idEnd_User INT
exercise_idExercise INT
pack_idPack INT
isChecked INT
Indexes

Abbildung 72: Beziehungstabelle zwischen Endnutzer und Übung

Zuvor wurde bereits erwähnt, dass Übungen Paketen zugeteilt sind und diese Pakete nach dem Erwerb einem Benutzer zugewiesen werden. Um den Fortschritt eines Pakets für einen speziellen Benutzer abspeichern zu können, musste eine neue Tabelle erstellt werden, in der das Attribut `isChecked` ausweist, ob die Übung bereits erledigt wurde oder nicht. Da Übungen in mehreren Paketen vorkommen können, müssen zum Fremdschlüssel der Übungs-ID zusätzlich noch die zwei weiteren Fremdschlüssel, Paket-ID und Endnutzer-ID, mitgespeichert werden.

Anmeldedaten:



credentials
idCredentials INT
email VARCHAR(45)
password VARCHAR(45)
access INT
Indexes

Abbildung 73: Tabelle Anmeldedaten

Die Anmeldedaten für Mandanten beziehungsweise Endnutzer, werden in dieser Tabelle gespeichert. Als Zugangsdaten wird eine E-Mail und ein Passwort verwendet. Zusätzlich gibt es noch ein Zugriffsberechtigungsattribut, das zwischen einem Endnutzer, einem Mandant und einem Entwickler unterscheidet.

11.5 Probleme

Dadurch, dass im Unterricht nur die grundlegenden Schritte durchgesprochen werden und die von EasyGym benötigten vertiefenden Bereiche wenig bis gar nicht angesprochen wurden, war vieles eigenständig zu erarbeiten und erlernen. Oft standen wir vor Problemen und Fehlermeldungen, die uns in der Form noch nicht begegnet waren. So gesehen wurden wir oft auf die Probe gestellt. Durch Einsatz, Fleiß und Motivation, aber auch durch Hilfestellungen von Kollegen und Professoren, konnten wir auch anfänglich schwierige Passagen gut meistern.

Durch die Auflösung der Arbeitsgemeinschaft mit ASKÖ-Linz, wurde das Projektteam mit Problemen konfrontiert, denen aufgrund der sehr geringen Auftrittswahrscheinlichkeit kaum mit Gegenmaßnahmen vorgebeugt wurde.

Darunter fällt, dass der Projektumfang nicht mehr vom Auftraggeber definiert oder vorgegeben wurde, sondern sich das Projektteam selbst die zu implementierenden Projektaspekte überlegen musste. Später sind viele zusätzliche Anwendungspunkte, die im Vorhinein nicht klar abschätzbar waren, aufgetaucht. Diese Umstände verursachten enormen Mehraufwand, der sich über die gesamte Dauer des Projektes erstreckte.

Zusätzlich hat das Projektteam die vereinbarten Testdaten nie bekommen und dadurch selbst die Informationen und Bilder erarbeiten und recherchieren müssen.



12 Resümee

Durch EasyGym wurde das Projektteam bereits während der Schulzeit phasenweise mit einem alltagstauglichen Arbeitsverhalten in der Informatik-Branche konfrontiert. Besonders das effiziente Arbeiten als Team wurde gestärkt und den Entwicklern ist klargeworden, dass es für die Lösung mancher Problemstellungen oft mehr als nur einen Kopf benötigt. Durch den Einsatz von sehr modernen und aktuellen Vorgehensweisen und Programmiersprachen beziehungsweise Entwicklungsumgebungen, ist das Projektteam sehr gut auf mögliche Aufgaben im späteren Berufsleben, die sich mit diesen Technologien beschäftigen, vorbereitet. Dass vor allem eine vollständige, klar strukturierte Projektplanung einen großen Einfluss auf ein gewünschtes Projektergebnis mit vordefiniertem Aufwand hat, wurde uns spätestens beim Auftreten so mancher Probleme klar. Vor allem das Einplanen von Zeitreserven aus diversen Gründen, wird den Projektmitgliedern in späteren Projekten eine große Hilfestellung sein. Darüber hinaus war diese Diplomarbeit das erste Projekt, das von Grund auf alleine durchgeführt wurde. Das hat uns geholfen, unsere Organisation besser zu gestalten und jederzeit im Blick zu haben.

Das Wichtigste, das wir während dieser Arbeit lernten, ist, dass man mit dem richtigen Partner immer jemanden an der Seite hat, der zuverlässig und hilfsbereit im Team agiert. Deshalb sind wir mit der gemeinsamen Entwicklung von EasyGym durchaus zufrieden und freuen uns, dass wir ein Projekt mit diesem Umfang zusammen gemeistert haben.



13 Aufgabenverteilung

13.1 Paul Heimel

- Zusammenfassung
- Abstract
- Danksagung
- Projektumfeld
- Motivation
- Ziel der Arbeit
- Grundlagen
 - Entwicklungssysteme
 - Sonstig verwendete Software
 - Verwendete Technologien
 - Bibliotheken
- Realisierung
 - Vorgehensweise
 - Webanwendung
 - Layout
 - Android Applikation
 - Datenbank
- Resümee
- Glossar
- Anhang
 - Projektstrukturplan
 - Meilensteine
 - Zeitlicher Ablauf Heimel
 - Risikomanagement
 - Risiken
 - Eingetretene Risiken



13.2 David Hennerbichler

- Abstract
- Danksagung
- Projektumfeld
- Technische Ziele
- Allgemeine Ziele
- Grundlagen
 - Entwicklungssysteme
 - Sonstig verwendete Software
 - Verwendete Technologien
 - Bibliotheken
 - Verwendete Architekturen
- Realisierung
 - Webanwendung
 - Grundlegende Ausgangslage
 - Umsetzung Management-Webseite
 - Besonderheiten Management-Webseite
 - Umsetzung REST-Schnittstelle
 - Datenbank
- Resümee
- Glossar
- Anhang
 - Meilensteine
 - Zeitlicher Ablauf Hennerbichler
 - Risikomanagement
 - Risiken
 - Eintretene Risiken



14 Glossar

Add-On

Unter einem Add-On versteht man eine Erweiterung, welche neben den Standardfunktionen einer Software spezielle Fähigkeiten mit sich bringt.

Authorization Filter

Ein Authorization Filter ist eine Komponente, die einen HTTP Request authentifiziert. (vgl. microsoft.com, 2017)

Bibliothek, Library (Informatik)

Eine Bibliothek ist eine Sammlung von Softwarefunktionen, die man in gewissen Fällen in eine Entwicklungsumgebung einbinden und somit die Funktionalität der Entwicklungsumgebung erhöhen kann. Sie wird zur Laufzeit verwendet, um Funktionen der Software, die die Standardsoftware nicht enthält, zu realisieren. (wikipedia.org, 2017)

Body (Web-API)

Im Body können bei Requests und Responses zusätzliche Informationen wie zum Beispiel JSON-Objekte oder Strings gespeichert werden.

Community-Designs

Ein Design wird meistens zur Veränderung des Aussehens von Standardkomponenten, wie zum Beispiel eines Eingabefeldes, verwendet. Bei einem Community-Design handelt es sich um ein Design, das nicht von den Entwicklern, welche die Standardkomponenten zur Verfügung stellen, geformt wurde, sondern von den Benutzern, die sich mit dieser Software beschäftigen.

CSS Frameworks

CSS Frameworks sind eine Sammlung von Gestaltungselementen, die in Bereichen der Webanwendung das Aussehen der Webseiten bestimmen. Es werden auch einige Schaltflächen sowie Steuer- und Eingabefelder durch dieses Framework zur Verfügung gestellt. (vgl. wikipedia.org, 2017)

Dateiendung

Je nach Dateiformat unterscheidet sich auch die Dateiendung (zum Beispiel .txt, .png oder .jpg).

**Dateiformat**

Das Dateiformat gibt Auskunft darüber, wie die Daten innerhalb der Datei strukturiert sind. Diese Struktur ist teilweise von der Dateierweiterung bestimmt. (vgl. univie.ac.at, 2017)

Datenmodellierung

Als Datenmodellierung wird der Prozess des Erstellens eines Datenmodells bezeichnet. Das Modellieren beschreibt Objekte und deren Beziehungen zueinander in der Struktur der ausgewählten Datenbank.

Dialog (Android)

Ein Dialog in Android ist eine visuelle Komponente, die über einer Activity erscheint und sich in den Vordergrund stellt.

Eclipse Foundation

Die Eclipse Foundation ist ein Zusammenschluss der Community und der Eclipse Organisation, die gemeinsam an kommerziell-freundlicher Software und Frameworks arbeiten.

Emulator

Unter einem Emulator versteht man ein System, das ein anderes System in gewissen Teilen nachbildet (zum Beispiel ein Smartphone am Computer). (vgl. wikipedia.org, 2017)

Entität

Als Entität wird in der Informatik ein eindeutig bestimmbares Objekt bezeichnet, das gespeichert und verarbeitet werden kann. Dieses kann nun in der Realität vorkommen, materiell, immateriell, konkret oder abstrakt sein. (vgl. wikipedia.org, 2017)

Expositur

Unter einer Expositur versteht man eine Zweigstelle. Bei Schulen stellt eine Expositur einen Teil einer Schule, der in einem anderen Gebäude untergebracht ist, dar. (vgl. duden.de, 2017)

Flat Design

Ist ein sehr minimalistisch gehaltenes Design. Es arbeitet sehr intensiv mit einfachen Designkomponenten wie zum Beispiel Farbflächen.

**FTP/SFTP**

Das File-Transfer-Protokoll ist ein Netzwerkprotokoll, dass zur Übertragung von Daten über IP-Netze verwendet wird.

Mit dem SSH File Transfer Protocol (SFTP) existiert eine auf SSH aufbauende, sichere Alternative zu FTP für Dateiverwaltung und -übertragung. (vgl. wikipedia.org, 2017)

Header

Als Header werden in der Informatik die Zusatzinformationen bezeichnet, die beschreiben, wie die Daten verarbeitet beziehungsweise wohin sie transportiert werden müssen. Der Header wird auch als Dateikopf bezeichnet. (vgl. wikipedia.org, 2017)

IBM

Ist ein international tätiges Unternehmen der Informatik-Branche.

Inkrement

Als Inkrement ist der festgelegte Betrag der Änderung einer Variable bezeichnet. Das Gegenteil dazu ist Dekrement (Verminderung). (vgl. wikipedia.org, 2017)

IntelliJ IDEA

IntelliJ IDEA ist eine auf Java basierende Entwicklungsumgebung.

Java

Java ist eine Insel.

JSON (JavaScript Object Notation)

JSON ist ein Datenformat in Textform. Es wird hauptsächlich zum Datenaustausch (Client-Server) verwendet. (vgl. w3schools.com, 2017)

Kernel

Ein Kernel ist der Kern eines Betriebssystems (zentraler Bestandteil). Darin sind unter anderem die Prozess- und Datenorganisation definiert. (vgl. wikipedia.org, 2017)



Kompilierung

Als Kompilierung wird der Vorgang des Übersetzens des vom Programmierer geschriebenen Source-Codes von der Programmiersprache in Maschinensprache bezeichnet. Diese Arbeit wird vom programmierspracheneigenen Compiler übernommen.

Open-source

Darunter versteht man Software, die von jedem verwendet und verändert werden kann. (vgl. opensource.com, 2017)

plattformunabhängig

Eine plattformunabhängige Software kann unter verschiedenen Hard- und Software-Konstellationen verwendet werden (zum Beispiel auf verschiedenen Betriebssystemen).

Programmierschnittstellen

Als Programmierschnittstelle wird ein Programmteil bezeichnet, der es ermöglicht, auf Datenbanken, Hardware, Festplatten oder Grafikkarten zuzugreifen, der aber auch das Erstellen von Komponenten der grafischen Benutzeroberfläche ermöglicht oder vereinfacht.

Progress-Dialog (Android)

Basiert auf einem Dialog und beinhaltet eine Fortschrittsanzeige (zum Beispiel in Form eines Kreises oder eines Balkens).

Repository

Repository bedeutet übersetzt Ablageort und beschreibt das Verzeichnis/den Speicherort, an dem digitale Dateien abgelegt werden können.

Request

Anfrage an den Server, um auf Datenstrukturen zugreifen zu können.

Response

Die Antwort eines Servers auf einen Request.

Session

Session bedeutet auf Deutsch übersetzt Sitzung und beschreibt eine stehende Verbindung zwischen Client und Server. Eine Session wird gestartet, wenn sich eine Person einloggt und beendet, wenn sie sich wieder ausloggt.



Session-Attribut

Ein Session-Attribut wird einer Session zugewiesen und ist eine Variable, die einen bestimmten Wert zugewiesen bekommt und später wieder aus der Session ausgelesen werden kann.

String (Informatik)

Ist in der Informatik ein Datentyp, der eine Zeichenkette enthält. (vgl. info-wsf.at, 2017)

Syntax

Die Syntax einer Sprache (eines Zeichensystems) beschreibt die Regeln, nach denen die Sprachkonstrukte (Zeichen des Zeichensystems) gebildet werden. (inf-schule.de, 2017)

Syntax-Highlighting

Darunter versteht man die Hervorhebung bestimmter Wörter und Zeichenkombinationen eines Textes (am Beispiel der Informatik ist der Text der Quellcode). (vgl. wikipedia.org, 2017)

Texteditor

Ein Textbearbeitungsprogramm, welches es erlaubt verschiedene, auf Text basierende Dateiformate zu verändern.

ThemeRoller

Der ThemeRoller ist von der Bibliothek PrimeFaces, die Auslage von zur Verfügung stehenden Themes, die mit Informationen beschrieben und vorgestellt werden.

Theme

Ein Theme beschreibt die Formatierung, den Farbton und das Aussehen, also generell das Design von Webseiten.

Toast (Android)

Ein Toast ist eine visuelle Komponente, die einen Text enthält und standardmäßig am unteren Bildschirmrand, nach einer Aktion die eine Bestätigung erfordert, erscheint.

User Interface

Unter User Interface versteht man die grafische Oberfläche einer Software-Anwendung, über die ein Nutzer (User) Eingaben tätigen kann.



ViewPager (Android)

Ein ViewPager kann mehrere Ansichten (Fragments) enthalten und eignet sich daher sehr gut, um Informationen die miteinander verbunden sind, jedoch eine eigene Ansicht benötigen, zu veranschaulichen.

Fragment

Ein Fragment beschreibt einen Teil einer Activity und kann verschiedene Komponenten beinhalten. (vgl. android.com, 2017)

Web-API

Das Web-API (Application Programming Interface) ist entweder für einen Web-Server oder einen Webbrowser. Es ist ein Entwicklungskonzept, dass meist nur für Webapplikation ausgelegt ist.

Web-Applikation/Webanwendung

Als Web-Applikation/Web-Anwendung wird ein Online-Anwendungsprogramm bezeichnet, das vom Server für den Zugriff eines Clients zur Verfügung gestellt wird. Diese Anwendungen können durch einen Webbrowser aufgerufen und bedient werden.

15 Anhang

15.1 Projektstrukturplan

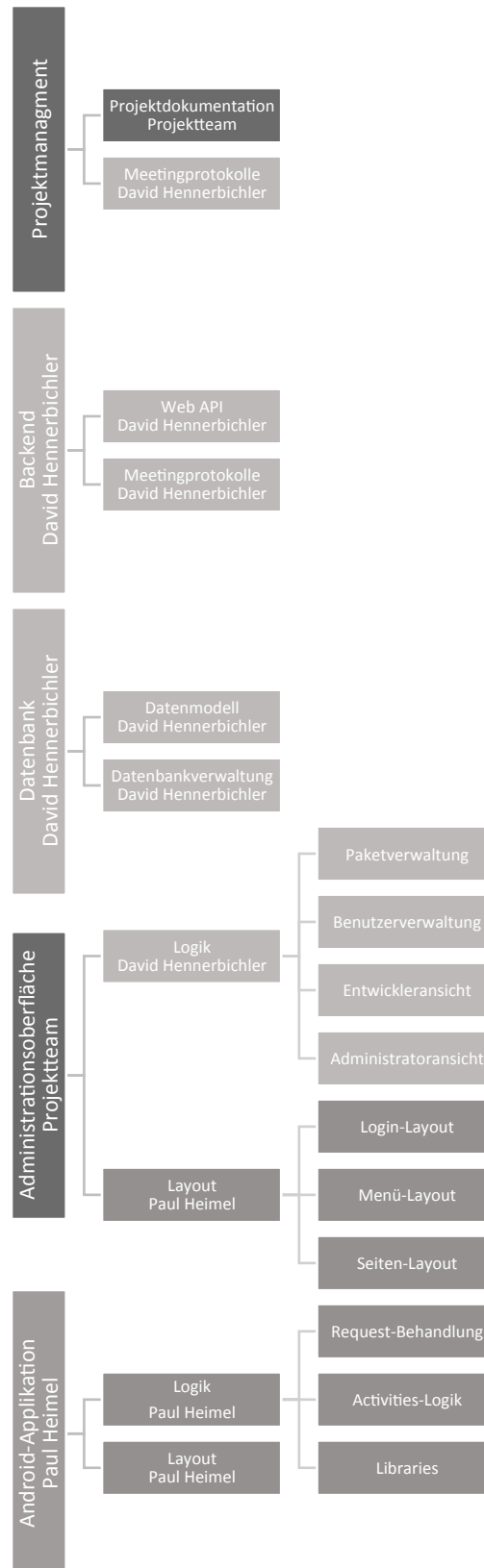


Abbildung 74: Projektstrukturplan

15.2 Meilensteine

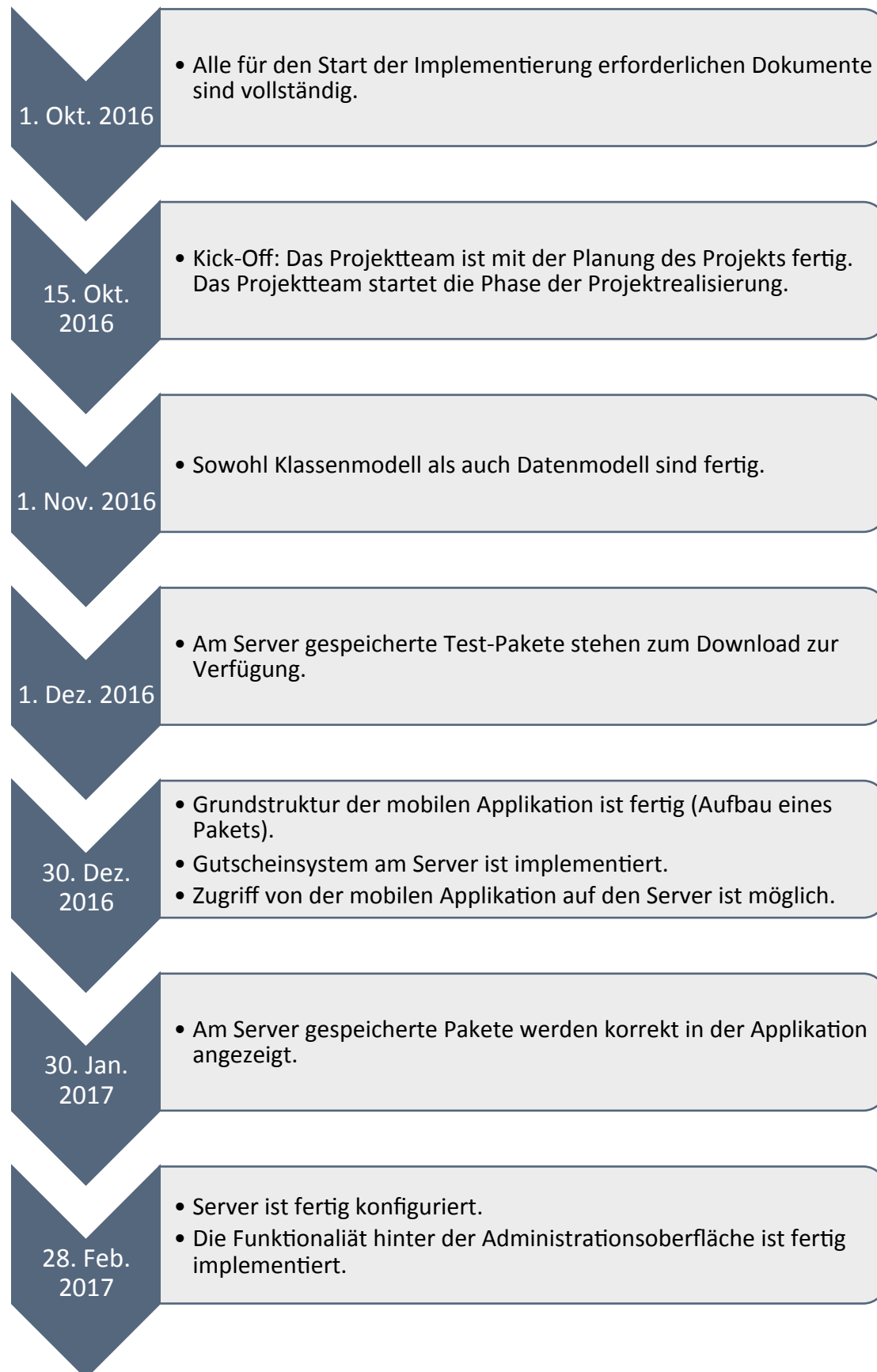


Abbildung 75:Meilensteine

15.3 Zeitlicher Ablauf

15.3.1 Paul Heimel

Eintrag	Datum	Zeit	Tätigkeit	Beschreibung
1	12.05.	3,67	Projektplanung	Meeting mit Herr Niedermaier (ASKÖ)
2	25.06.	0,50	Projektplanung	Meetingprotokoll erstellen
3	10.07.	2,50	Projektdokumentation	Start der Projektdefinition
4	22.08.	4,50	Projektdokumentation	Projektdefinition Zeitmanagementplanung
5	23.08.	1,00	Projektdokumentation	Meilensteine definieren Projektdefinition weiterschreiben
6	25.08.	1,50	Projektdokumentation	Projektdefinition Product Backlog
7	26.08.	2,00	Projektdokumentation	Projektdefinition
8	31.08.	4,00	Projektdokumentation	Verschiede Dokumente Projektdefinition
9	01.09.	2,50	Projektdokumentation	Projektstrukturplan
10	13.09.	3,00	Projektdokumentation	Risikomanagement
11	19.09.	2,00	Projektdokumentation / Projektplanung	Projekt neu definieren Klassenmodell Datenmodell
12	20.09.	2,00	Projektplanung / Projektdokumentation / Projektentwicklung / Konfiguration der Entwicklungsumgebungen	Projektplanung Konfiguration Git Diplomarbeitsdatenbank Eintrag
13	24.09.	1,50	Projektdokumentation	Änderung vom Layout der Dokumente auf den Vorlagenstandard
14	25.09.	2,50	Entwicklung Android Konfiguration der Entwicklungsumgebungen	Einrichtung der Entwicklungsumgebung Start der Entwicklung der Mobile-App - Login-Screen angefangen
15	26.09.	1,00	Entwicklung Android	Entwicklung der Mobile-App - Main Activity inkl. Navigation Drawer - Menü des Navigation Drawers ist bereits fertig o Keine Funktionen hinter den Schaltflächen
16	29.09.	3,50	Entwicklung Android	Entwicklung der Mobile-App - RecyclerView inkl. CardView für Gutscheine erstellt o Funktion noch nicht fertig
17	05.10.	1,75	Entwicklung Android	Entwicklung der Mobile-App - RecyclerView inkl. CardView für Gutscheine fertiggestellt
18	24.10.	1,75	Entwicklung Android	Entwicklung der Mobile-App - Gutschein-Card Designänderung
19	02.11.	4,50	Entwicklung Android	Entwicklung der Mobile-App -CardLayout inkl. RecyclerView für Packages erstellt
20	07.11.	4,00	Entwicklung Android	Entwicklung der Mobile-App - CardView inkl. RecyclerView für Store Packages erstellt - Design für Store Packages verändert
21	10.11.	2,00	Entwicklung Android	Entwicklung der Mobile-App -Layout-Veränderungen der Card-Views
22	13.11.	6,00	Entwicklung Android	Entwicklung der Mobile-App



				- Vervollständigung verschiedenster Layouts und Card Views Nachtrag Stunden
23	15.11.	4,50	Entwicklung Android	Entwicklung der Mobile-App - Layout-Veränderungen Zeitnachtrag der Schulstunden
24	15.11.	1,00	Entwicklung Android	Entwicklung der Mobile-App - CardView Layoutänderungen
25	16.11.	2,50	Entwicklung Android	Entwicklung der Mobile-App - CardView Logik
26	20.11.	4,00	Projektdokumentation / Entwicklung	Projektplakat Designänderungen Entwicklung der Mobile-App - Profil bearbeiten - Passwort bearbeiten
27	06.12.	8,00	Entwicklung Android	Entwicklung der Mobile-App - Layout größtenteils fertig
28	09.12.	8,00	Entwicklung Android	Entwicklung der Mobile-App - Testdaten mittels JsonObjects erstellen - Ausführungsschritte inkl. bottom dots implementiert
29	13.12.	3,00	Entwicklung Android	weitere Testdaten definieren
30	02.01.	2,83	Entwicklung Android / Projektplanung	Entwicklung der Mobile-App - Kleine Layoutveränderungen Besprechung des Layouts
31	03.01.	1,50	Entwicklung Android	Entwicklung der Mobile-App - Übersetzungs Strings für die Sprache Deutsch eingefügt
32	15.01.	7,00	Entwicklung Android	Entwicklung der Mobile-App - Start der Datenkonsumierung Zeitnachtrag
33	25.01.	10,00	Entwicklung Android	Entwicklung der Mobile-App - Requests Verarbeiten - Bearbeiten des Layouts von Login
34	31.01.	10,00	Entwicklung Android	Entwicklung der Mobile-App -Layout Änderungen für Profile Settings, Login Activity, Register Activity - Daten vom Server konsumieren
35	10.02.	10,00	Entwicklung Android Konfiguration Entwicklungsumgebungen	Entwicklung der Mobile-App - Minimale Designänderungen - neuer Dialog zum Auswählen von Gutscheinen beim Kauf Git Reparieren Zeit Nachtrag
36	15.02.	7,00	Entwicklung Android Entwicklung Datenbank	Entwicklung der Mobile-App - Designänderungen - Layoutüberlegungen Datenbankdesign änderungen Zeitnachtrag
37	16.02.	2,75	Entwicklung Android	Entwicklung der Mobile-App - Layout Änderungen
38	17.02.	4,00	Konfiguration Entwicklungsumgebungen	Konfiguration der Entwicklungsumgebung für den Server
39	18.02.	4,00	Entwicklung Webanwendung	Webanwendung - Entwickleransicht Startseite - Login
40	19.02.	4,50	Entwicklung	Webanwendung



			Webanwendung	- Layout Login
41	20.02.	6,50	Entwicklung Webanwendung	WebAnwendung - Layout Login - Layout Entwickleransicht
42	21.02.	3,75	Entwicklung Webanwendung	Webanwendung - Primefaces Theme Download - Layout Mandantenansicht
43	22.02.	6,00	Entwicklung Webanwendung	Webanwendung - Layout Mandantenansicht
44	23.02.	2,75	Entwicklung Webanwendung Konfiguration der Entwicklungsumgebung en	Webanwendung - Mandantenansicht - Neue Seiten hinzugefügt - Eclipse von Projektpartner David reparieren
45	24.02.	3,00	Entwicklung Android	Entwicklung der Mobile-App - neuer Dialog für Informationen beim Kauf
46	27.02.	10,00	Sonstiges	Vorbereitung der Präsentation für den P@BS-Award
47	28.02.	4,00	Entwicklung Android	Kleine Veränderungen in der mobilen Applikation
48	01.03.	4,00	Sonstiges	Diplomarbeit schreiben
49	02.03.	3,50	Sonstiges	Diplomarbeit schreiben
50	08.03.	9,00	Sonstiges	Projektpräsentations-Vorbereitung Diplomarbeit schreiben Zeitnachtrag
51	14.03.	7,00	Sonstiges	Diplomarbeit schreiben Zeitnachtrag
52	21.03.	5,00	Entwicklung Android Entwicklung Webanwendung Sonstiges	Entwicklung Diplomarbeit schreiben
53	27.03.	15,00	Sonstiges	Zeitnachtrag für Verschriftlichung und Entwicklung
54	27.03.	6,00	Entwicklung Android	Entwicklung mobile App - Layout-Veränderungen - Request-Veränderungen (JSON <-> String)
55	29.03.	4,50	Sonstiges	Diplomarbeit schreiben
56	30.03.	5,00	Sonstiges	Diplomarbeit schreiben - Abbildungsverzeichnis - Literaturverzeichnis
57	31.03.	2,00	Projektdokumentation	Zeitlicher Ablauf des Projektes
58	01.04.	3,50	Sonstiges	Korrekturlesen der Diplomarbeit intern
59	02.04.	4,00	Sonstiges	Korrekturlesen der Diplomarbeit extern
60	03.04.	1,00	Sonstiges	Diplomarbeit schreiben - Layout ausbessern
61	04.04	10,00	Sonstiges	Diplomarbeit schreiben und DVD brennen

Aufwand gesamt: 262,75 Stunden

Tabelle 4: Zeitaufzeichnung HeimeI

**15.3.2 David Hennerbichler**

Eintrag	Datum	Zeit	Tätigkeit	Beschreibung
1	12.05.	4	Meeting	Meeting in Linz, Niedermaier ASKÖ
2	20.06.	1,75	Design	Logo-Entwurf
3	21.06.	2,58	Design	Logo-Entwurf erweitert
4	21.06.	2	Design	Logo fertiggestellt
5	31.07.	1,5	Projektmanagement	Meeting
6	31.08.	2	Dokumentation	Dokumentation vervollständigen
7	13.09.	3	Dokumentation	Dokumentation vervollständigen
8	18.09.	4,67	Projektmanagement	Pauschalnachtrag für Besprechungen, Anträge, Dokumente
9	23.09.	0,92	Server Anwendung programmieren	Datenbankmodell erstellen
10	24.09.	1,5	Server Anwendung programmieren	Datenbankmodell erstellen
11	26.09.	0,83	Projektmanagement	Diplomarbeitsdatenbank vervollständigen
12	27.09.	1,67	Server Anwendung programmieren	Datenbankmodell erweitern
13	02.10.	2	Server Anwendung programmieren	Serveranwendungsimplementation initialisieren
14	03.10.	0,83	Server Anwendung programmieren	Server-Anwendung implementieren
15	17.10.	1,67	Server Anwendung programmieren	erste Pages, Beans und Felder
16	01.11.	6	Server Anwendung programmieren	Grundgerüst, DB, Server erstellen
17	07.11.	1	Server Anwendung programmieren	weitere Serverarbeiten
18	13.11.	3,5	Server Anwendung programmieren	DB-Modell Änderung
19	14.11.	3	Server Anwendung programmieren	Serveranwendung, Ordnerstruktur am angelegt
20	15.11.	3	Server Anwendung programmieren	Login ohne Session fertiggestellt
21	21.11.	2	Server Anwendung programmieren	Exercise hinzufügen und begonnen
22	22.11.	2	Dokumentation	Projektplakat erstellen A1 und A3
23	23.11.	0,75	Dokumentation	Projektplakat erstellen A1 und A3
24	24.11.	7,5	Server Anwendung programmieren	Formular für Exercise fertiggestellt und Funktion ausgebessert/getestet
25	05.12.	1,67	Entwurf	Entwickler und Mandanten Oberfläche handschriftlich planen und skizzieren
26	12.12.	2,25	Entwurf	Entwickler und Mandanten Oberfläche handschriftlich planen und skizzieren
27	15.12.	2	Server Anwendung programmieren	Restschnittstelle für Enduser hinzugefügt
28	02.01.	2,75	Server Anwendung programmieren	Rest erweitert
29	06.01.	2,5	Besprechung	Besprechung
30	07.01.	5	Serverinstallation-Konfiguration	Server mit FileZilla und Putty erreichen
31	15.01.	2,5	Server Anwendung programmieren	Rest-Endpunkte überarbeitet



32	23.01.	4	Server Anwendung programmieren	Datenmodelländerungen und Entitätsänderungen
33	24.01.	4	Server Anwendung programmieren	Fehler behoben, Mapping-Annotations
34	25.01.	3	Server Anwendung programmieren	Richtigstellung der Server-URL
35	31.01.	11	Server Anwendung programmieren	Nachtrag der letzten Arbeitstage Ausbessern von Warnungen in den Daos Update der Rest Requests Media hinzugefügt
36	01.02.	6,25	Server Anwendung programmieren	Veränderung im Datenmodell - Carts hinzugefügt Anpassung der Entities
37	06.02.	5	Server Anwendung programmieren	Git-crash aufgrund von Merge-Conflicts Serverreparaturen
38	08.02.	2	Server Anwendung programmieren	Git Reset complete
39	12.02.	7	Server Anwendung programmieren	Rest Request Dokument angepasst VoucherEndpoint und Dao hinzugefügt
40	14.02.	3	Server Anwendung programmieren	Neue Requests hinzugefügt
41	16.02.	2,72	Server Anwendung programmieren	SessionKey in DB, Servermodells hinzugefügt, Dao, Entpoint
42	17.02.	5	Server Anwendung programmieren	Serveranwendung, Pages, Daos, Entites
43	18.02.	4	Server Anwendung programmieren	MandantenPage verändert und ExercisePage richtiggestellt
44	19.02.	2	Server Anwendung programmieren	Mandantenpage richtiggestellt
45	21.02.	1	Server Anwendung programmieren	Icons hinzugefügt
46	23.02.	10	Server Anwendung programmieren	Nachtrag der letzten Tage: Mandanten anzeigen Mandanten löschen Mandanten bearbeiten Mandanten erstellen *Fehler behoben! Gutscheine anzeigen Gutscheine löschen Gutscheine bearbeiten Gutscheine erzeugen (zufällig aus der definierten Zeichenkette ohne o,0, l,i,l, usw...) Info-Messages für Formulare Butten für das Löschen des Formulars mit provisorischen Icon, des messma si sowieso generell nu auschaun Menü
47	24.02.	2	Dokumentation	Gitreparatur Gitversionen-Management
48	25.02.	9	Server Anwendung programmieren	ServerAnwendung, Checkboxes, Logik, Allocation-Seite Logikänderung in allocation und exercise
49	28.02.	7	Server Anwendung programmieren	Serveranwendungsänderung, grobe Datenmodelländerung, Testdaten eingefügt Bilder
50	01.03.	4	Server Anwendung programmieren	Buttontexte geändert, Login SessionKey erstellt Datenmodelländerungen
51	05.03.	9	Server Anwendung programmieren	kurzzeitiger Nachtrag: Datenmodelländerung Löschen von Exercises in Paketen Pakete löschen (funktioniert noch nicht) Person Page teilweise umgesetzt
52	14.03.	13	Server Anwendung programmieren	Bilddateiupload realisiert für lokalen und Schul-Server done: - clear form after create - paket bearbeiten implementieren - veeeele Bugfixes... - Bilder nun auch bei edit anzeigbar und leere Eingaben abgefangen
53	16.03.	11	Server Anwendung programmieren	Checkboxes fertig implementiert, Fehler ausgebessert Von Grund auf erneuert, durch viele Fehler aufgehalten worden



54	20.03.	13	Server Anwendung programmieren	Nachtrag der letzten Tage: Firmen anlegen Kategorien anlegen minor bug fixes steps per exercises anlegen und anzeigen Datenmodellveränderungen Filename Generierung statt Datenbankspeicherung Datenmodelländerung Step per Exercise auf eigene Seite verlegt Category for Exercise implementiert Benachrichtungen rechts oben hinzugefügt
55	22.03.	4	Server Anwendung programmieren	Exercise-Seite verändert Category-Seite vollendet Bessere Version für das Bearbeiten von Exercises
56	25.03.	4	Server Anwendung programmieren	Login fertig, ohne Accessüberprüfung
57	27.03.	5	easyGym ASKÖ	Datenmodell ausgebessert Entitäten im Projekt angepasst Askoe aus Titel gelöscht, kleine Änderungen im Code
58	28.03.	5	easyGym ASKÖ	Login Accessprüfung hinzugefügt Gitverzeichnis finalisiert
59	01.04.	3	easyGym ASKÖ	Code Refactoring
60	02.04.	11	easyGym ASKÖ	Nachtrag: Diplomarbeit verschriftlichen, Korrekturlesen
61	04.04.	14	Sonstiges	Diplomarbeit finalisieren

Aufwand gesamt: 262,31 Stunden

Tabelle 5: Zeitaufzeichnung Hennerbichler

16.1 Risikomanagement

*Wahrscheinlichkeits-Skala (1 = unwahrscheinlich, 5 = sehr wahrscheinlich)

16.1.1 Risiken

Ungenaue Definition des Projektumfangs

Projektphase	Initialisierung
Wahrscheinlichkeit	1
Auswirkungen	<ul style="list-style-type: none">• Probleme während der Projektimplementierung
Maßnahmen	<ul style="list-style-type: none">• Wird durch bereits erlernte Kenntnisse beinahe vollständig eliminiert

Tabelle 6: Risiko Projektumfang

Zeitmanagement fehlerhaft

Projektphase	Initialisierung
Wahrscheinlichkeit	3
Auswirkungen	<ul style="list-style-type: none">• Kann sich negativ auf die Benotung auswirken
Maßnahmen	<ul style="list-style-type: none">• Zeitmanagement mit Betreuungslehrer besprechen• Erfahrungswerte aus vorherigen Projekten miteinbeziehen

Tabelle 7: Risiko Zeitmanagement

Know-how nicht ausreichend

Projektphase	Initialisierung, Implementierung
Wahrscheinlichkeit	2-3
Auswirkungen	<ul style="list-style-type: none">- Projekt kann nicht abgeschlossen werden- Es muss mehr Zeit investiert werden
Maßnahmen	<ul style="list-style-type: none">• Betreuungslehrer sowie Partner bei Schwierigkeiten um Rat bitten• Schwierigkeiten im Zeitmanagement berücksichtigen

*Tabelle 8: Risiko know-how***Ausfall eines Entwicklers**

Projektphase	Initialisierung, Implementierung
Wahrscheinlichkeit	2-3
Auswirkungen	<ul style="list-style-type: none">- Projekt kann nicht abgeschlossen werden- Die restlichen Entwickler müssen Arbeit des Arbeitsunfähigen bewältigen
Maßnahmen	<ul style="list-style-type: none">• Sonderfälle wie diesen im Zeitmanagement berücksichtigen• Zeitreserven aufbauen, um Ausfall überbrücken zu können

Tabelle 9: Risiko Ausfall eines Entwicklers

Product Owner springt ab

Projektphase	-
Wahrscheinlichkeit	1
Auswirkungen	<ul style="list-style-type: none">- Gibt keinen Abnehmer mehr- Entwickler entgeht möglicher Gewinn- Entwicklerteam ist auf sich alleine gestellt
Maßnahmen	<ul style="list-style-type: none">• Gutes Verhältnis zum Product Owner aufbauen

Tabelle 10: Risiko Product Owner springt ab

Hoher Zeitdruck

Projektphase	Entwicklung
Wahrscheinlichkeit	3
Auswirkungen	<ul style="list-style-type: none">- Probleme und möglicher Verlust verschiedener Funktionen im Produkt
Maßnahmen	<ul style="list-style-type: none">• Gut durchdachtes Zeitmanagement

Tabelle 11: Risiko Zeitdruck

App-Interesse am Markt nicht vorhanden

Projektphase	Release
Wahrscheinlichkeit	3
Auswirkungen	<ul style="list-style-type: none">- Gewinnverlust für Entwickler und Product Owner
Maßnahmen	<ul style="list-style-type: none">• App durch Design interessant gestalten• Gratis Pakete beim Download anbieten• Große Auswahl an verschiedenen Paketen• Kostenpflichtige Pakete im leistbaren Bereich halten

Tabelle 12: Risiko am Markt vorhanden

**Lizenzen & Programme kostenpflichtig**

Projektphase	Gesamte Projektdauer
Wahrscheinlichkeit	1-2
Auswirkungen	- Projektteam muss Entwicklungsumgebung bezahlen oder wechseln
Maßnahmen	• Laufend die Preisentwicklung der Entwicklungsumgebungen kontrollieren

Tabelle 13: Risiko Lizenzen kostenpflichtig



16.1.2 Eingetretene Risiken

Ungenaue Definition des Projektumfangs

Eintrittsgrund:

Dieses Risiko ist indirekt eingetreten, nachdem der Product Owner nicht mehr Teil des Projektes war. Das Projektteam musste sich nun selbst überlegen, welche Funktionalitäten für die Umsetzung nötig waren.

Ursprünglich waren Vereinbarungen wie zum Beispiel das Erhalten von Testdaten zu fixierten Terminen ausgemacht, welche jedoch vom Product Owner nicht erfüllt wurden.

Auswirkungen:

- Projektumfang ist nicht wie geplant vom Auftraggeber definiert worden
- Am Projektbeginn definierte Meilensteine konnten teilweise nicht eingehalten werden.
- Zeitmanagement nicht mehr aussagekräftig
- Zusätzlicher Aufwand für die Entwickler

Maßnahmen:

- Detaillierte Definition des Projektumfangs durch die Entwickler
- Mehraufwand betreiben

Product Owner springt ab

Eintrittsgrund:

Das Arbeitsverhältnis zwischen dem Verein ASKÖ-Linz und dem Entwickler-Team wurde aus Zeitgründen des Product Owners aufgelöst.

Auswirkungen:

- Die Projektidee ist veraltet und nicht umsetzbar, da die Überlegungen des ehemaligen Product Owners nur durch dessen Unterstützung realisierbar gewesen wären
- Dokumentation veraltet
- Zeitmanagement nicht mehr aussagekräftig

Maßnahmen:

- Projektidee neu definiert und die gesamte Dokumentation darauf ausgelegt
- Aufgaben des Product Owners ans Entwickler-Team übertragen

Hoher ZeitdruckEintrittsgrund:

Dieses Risiko ist aufgrund der nicht vorhersehbaren Trennung vom Auftraggeber sowie der Unterschätzung des schulischen Aufwandes eingetreten.

Auswirkungen:

- Neben dem Schulstress entstand auch der Druck, die Diplomarbeit fertig zu stellen.

Maßnahmen:

- Gutes Zeitmanagement des Entwicklerteams, um neben dem Schulstress auch die Diplomarbeit verfassen zu können.

16.1.3 Fazit aus den Problemen

Dem Projektteam wurde klar, dass auch Risiken die nur mit sehr geringer Wahrscheinlichkeit eintreten können, durchaus einen beträchtlichen Mehraufwand mit sich bringen können. Die Maßnahmen zur Bewältigung der Risiken sollten daher ernst genommen und im Vorhinein durchdacht werden. Das Einplanen von Zeitreserven ist Voraussetzung dafür, ein Projekt im gewünschten Zeitrahmen durchführen zu können.



16.2 Literaturverzeichnis

- aberger.at. (2017, März). Retrieved from <http://www.aberger.at/blog/assets/images/abs.jpg>
- android.com. (2017, März). Retrieved from <https://developer.android.com/training/volley/index.html>
- android.com. (2017, März). Retrieved from <https://developer.android.com/reference/android/app/Activity.html>
- android.com. (2017, März). Retrieved from <https://developer.android.com/design/material/index.html>
- android.com. (2017, März). Retrieved from <https://developer.android.com/training/material/lists-cards.html>
- android.com. (2017, März). Retrieved from <https://developer.android.com/about/dashboards/index.html>
- android.com. (2017, März). Retrieved from <https://developer.android.com/guide/components/fragments.html>
- askoe-linz-steg.at. (2017, März). Retrieved from <http://www.askoe-linz-steg.at/>
- Atlassian. (2017, März). Retrieved from <https://www.heise.de/download/product/sourcetree-90675>
- blogspot.co.at. (2017, März). Retrieved from <http://tiroltech.blogspot.co.at/2014/11/android-und-die-funktionsweise.html>
- comsysto.com. (2017, März). Retrieved from <https://comsysto.com/leistungen/werkzeugkasten>
- Don HO. (2017, März). Retrieved from <https://www.heise.de/download/product/notepad-26659>
- doubleslash.de. (2017, März). Retrieved from <https://blog.doubleslash.de/primefaces/>
- duden.de. (2017, März). Retrieved from <http://www.duden.de/rechtschreibung/Expositur>
- eclipse.org. (2017, März). Retrieved from <https://eclipse.org/artwork/>
- fcp.at. (2017, März). Retrieved from <http://www.fcp.at/de/projekte/htbla-perg>
- getpostman.com. (2017, März). Retrieved from <http://blog.getpostman.com/2014/07/17/postmans-got-a-new-look/>
- google.com. (2017, März). Retrieved from <https://sites.google.com/site/gson/gson-user-guide>
- HTL-Perg. (2017, März). Retrieved from <http://htl-perg.ac.at/hoehere-abteilung>
- HTL-Perg. (2017, März). Retrieved from <http://htl-perg.ac.at/fachschule>
- htl-perg.ac.at. (2017, März). Retrieved from <https://dspace.htl-perg.ac.at/>
- individuapp.com. (2017, März). Retrieved from <http://individuapp.com/de/android-kurs/die-klassen-activity-und-intent>
- info-wsf.at. (2017, März). Retrieved from http://www.info-wsf.de/index.php/Datentyp_String



- inf-schule.de. (2017, März). Retrieved from http://www.inf-schule.de/sprachen/sprachenundautomaten/formalesprachen/einfuehrung_zeichensystem/syntaxsemantikpragmatik
- itwissen.info. (2017, März). Retrieved from <http://www.itwissen.info/Java-Java.html>
- itwissen.info. (2017, März). Retrieved from <http://www.itwissen.info/JPA-Java-persistence-API.html>
- itwissen.info. (2017, März). Retrieved from <http://www.itwissen.info/AJAX-asynchronous-JavaScript-and-XML.html>
- itwissen.info. (2017, März). Retrieved from <http://www.itwissen.info/AJAX-asynchronous-JavaScript-and-XML.html>
- itwissen.info. (2017, März). Retrieved from <http://www.itwissen.info/SQL-structured-query-language.html>
- jboss.org. (2017, März). Retrieved from <http://design.jboss.org/wildfly/>
- logos-download.com. (2017, März). Retrieved from <http://logos-download.com/10695-java-logo-download.html>
- microsoft.com. (2017, März). Retrieved from <https://docs.microsoft.com/en-us/aspnet/web-api/overview/security/authentication-filters>
- opensource.com. (2017, März). Retrieved from <https://opensource.com/resources/what-open-source>
- Postdot Technologies. (2017, März). Retrieved from <https://www.getpostman.com/>
- primefaces.org. (2017, März). Retrieved from <https://www.primefaces.org/documentation/>
- primefaces.org. (2017, März). Retrieved from <https://www.primefaces.org/themes/>
- quintetsolutions.com. (2017, März). Retrieved from <http://quintetsolutions.com/services/mysql-workbench-consultation/>
- sleepyshark.com. (2017, März). Retrieved from <http://kb.sleepyshark.com/article.php?id=12>
- sourcetreeapp.com. (2017, März). Retrieved from <https://www.sourcetreeapp.com/>
- stackoverflow.com. (2017, März). Retrieved from <http://stackoverflow.com/questions/630453/put-vs-post-in-rest>
- stackoverflow.com. (2017, März). Retrieved from <http://stackoverflow.com/questions/630453/put-vs-post-in-rest>
- Thomas Bayer. (2017, März). Retrieved from <http://www.oio.de/public/xml/rest-webservices.htm>
- tutorialspoint.com. (2017, März). Retrieved from https://www.tutorialspoint.com/jsf/jsf_overview.htm
- ubuntuusers.de. (2017, März). Retrieved from <https://wiki.ubuntuusers.de/Java/>
- univie.ac.at. (2017, März). Retrieved from <https://www.univie.ac.at/video/grundlagen/dateiformate.htm>
- w3.org. (2017, März). Retrieved from <https://www.w3.org/html/logo/>



w3schools.com. (2017, März). Retrieved from https://www.w3schools.com/js/js_json_intro.asp

wikia.com. (2017, März). Retrieved from <http://fallout.wikia.com/wiki/File:Android-logo.png>

wikimedia.org. (2017, März). Retrieved from https://commons.wikimedia.org/wiki/File:Android_Studio_icon.svg

wikimedia.org. (2017, März). Retrieved from https://commons.wikimedia.org/wiki/File:Microsoft_Word_logo.png

wikimedia.org. (2017, März). Retrieved from https://commons.wikimedia.org/wiki/File:Microsoft_Word_logo.png

wikipedia.org. (201). Retrieved from <https://de.wikipedia.org/wiki/CSS-Framework>

wikipedia.org. (2017, März). Retrieved from https://de.wikipedia.org/wiki/Android_Studio

wikipedia.org. (2017, März). Retrieved from [https://de.wikipedia.org/wiki/Eclipse_\(IDE\)](https://de.wikipedia.org/wiki/Eclipse_(IDE))

wikipedia.org. (2017, März). Retrieved from <https://de.wikipedia.org/wiki/Notepad%2B%2B>

wikipedia.org. (2017, März). Retrieved from <https://de.wikipedia.org/wiki/FileZilla>

wikipedia.org. (2017, März). Retrieved from https://de.wikipedia.org/wiki/MySQL_Workbench

wikipedia.org. (2017, März). Retrieved from https://de.wikipedia.org/wiki/Mozilla_Firefox

wikipedia.org. (2017, März). Retrieved from <https://de.wikipedia.org/wiki/Kommandozeile>

wikipedia.org. (2017, März). Retrieved from <https://de.wikipedia.org/wiki/PuTTY>

wikipedia.org. (2017, März). Retrieved from <https://de.wikipedia.org/wiki/Atlassian>

wikipedia.org. (2017, März). Retrieved from https://de.wikipedia.org/wiki/Microsoft_Office

wikipedia.org. (2017, März). Retrieved from <https://de.wikipedia.org/wiki/GIMP>

wikipedia.org. (2017, März). Retrieved from <https://de.wikipedia.org/wiki/WildFly>

wikipedia.org. (2017, März). Retrieved from https://de.wikipedia.org/wiki/Hypertext_Markup_Language

wikipedia.org. (2017, März). Retrieved from https://de.wikipedia.org/wiki/Cascading_Style_Sheets

wikipedia.org. (2017, März). Retrieved from [https://de.wikipedia.org/wiki/Android_\(Betriebssystem\)](https://de.wikipedia.org/wiki/Android_(Betriebssystem))

wikipedia.org. (2017, März). Retrieved from https://de.wikipedia.org/wiki/Data_Access_Object

wikipedia.org. (2017, Oktober). Retrieved from <https://de.wikipedia.org/wiki/Scrum>

wikipedia.org. (2017, März). Retrieved from <https://en.wikipedia.org/wiki/Notepad%2B%2B>

wikipedia.org. (2017, März). Retrieved from <https://de.wikipedia.org/wiki/FileZilla>

wikipedia.org. (2017, März). Retrieved from https://de.wikipedia.org/wiki/Mozilla_Firefox

wikipedia.org. (2017, März). Retrieved from <https://de.wikipedia.org/wiki/PuTTY>

wikipedia.org. (2017, März). Retrieved from https://de.wikipedia.org/wiki/Microsoft_Office



- wikipedia.org. (2017, März). Retrieved from https://de.wikipedia.org/wiki/Datei:Microsoft_PowerPoint_2013_logo.svg
- wikipedia.org. (2017, März). Retrieved from https://de.wikipedia.org/wiki/Datei:Microsoft_Excel_Logo.svg
- wikipedia.org. (2017, März). Retrieved from https://de.wikipedia.org/wiki/Datei:The_GIMP_icon_-_gnome.svg
- wikipedia.org. (2017, März). Retrieved from <https://en.wikipedia.org/wiki/Gson>
- wikipedia.org. (2017, März). Retrieved from <https://de.wikipedia.org/wiki/Laufzeitbibliothek>
- wikipedia.org. (2017, März). Retrieved from <https://de.wikipedia.org/wiki/Emulator>
- wikipedia.org. (2017, März). Retrieved from [https://de.wikipedia.org/wiki/Kernel_\(Betriebssystem\)](https://de.wikipedia.org/wiki/Kernel_(Betriebssystem))
- wikipedia.org. (2017, März). Retrieved from <https://de.wikipedia.org/wiki/Syntaxhervorhebung>
- wikipedia.org. (2017, März). Retrieved from [https://de.wikipedia.org/wiki/Entitat_\(Informatik\)](https://de.wikipedia.org/wiki/Entitat_(Informatik))
- wikipedia.org. (2017, Marz). Retrieved from <https://de.wikipedia.org/wiki/CSS-Framework>
- wikipedia.org. (2017, Marz). Retrieved from https://de.wikipedia.org/wiki/File_Transfer_Protocol
- wikipedia.org. (2017, Marz). Retrieved from <https://de.wikipedia.org/wiki/Header>
- wikipedia.org. (2017, Marz). Retrieved from https://de.wikipedia.org/wiki/Inkrement_und_Dekrement
- xamarin.com. (2017, Marz). Retrieved from https://developer.xamarin.com/guides/android/application_fundamentals/understanding_android_api_levels/

16.3 Abbildungsverzeichnis

Abbildung 1: Projektumfeld Mai 2016 (htl-perg.ac.at, 2017) (askoe-linz-steg.at, 2017)	11
Abbildung 2: Projektumfeld Oktober 2016 (htl-perg.ac.at, 2017)	11
Abbildung 3: Paul Heimel	12
Abbildung 4: David Hennerbichler	13
Abbildung 5: Dipl.-Ing. Christian Aberger (aberge.at, 2017)	14
Abbildung 6: HTL-Perg (fcp.at, 2017)	14
Abbildung 7: Android Studio (wikimedia.org, 2017)	18
Abbildung 8: Eclipse (eclipse.org, 2017)	18
Abbildung 9: Notepad++ (wikipedia.org, 2017)	19
Abbildung 10: Postman (getpostman.com, 2017)	19
Abbildung 11: FileZilla (wikipedia.org, 2017)	20
Abbildung 12: MySQL-Workbench (quintetsolutions.com, 2017)	20
Abbildung 13: Firefox (wikipedia.org, 2017)	21
Abbildung 14: cmd-Verknüpfung	21
Abbildung 15: Putty (wikipedia.org, 2017)	22
Abbildung 16: SourceTree (sourcetreeapp.com, 2017)	22
Abbildung 17: Microsoft-Office (wikipedia.org, 2017)	23
Abbildung 18: Word (wikimedia.org, 2017)	23
Abbildung 19: PowerPoint (wikipedia.org, 2017)	23
Abbildung 20: Excel (wikipedia.org, 2017)	23
Abbildung 21: Visio (sleepyshark.com, 2017)	23
Abbildung 22: GIMP (wikipedia.org, 2017)	24
Abbildung 23: Java (logos-download.com, 2017)	25
Abbildung 24: Wildfly (jboss.org, 2017)	26
Abbildung 25: HTML5 (w3.org, 2017)	26
Abbildung 26: CSS3 (comsysto.com, 2017)	27
Abbildung 27: Android (wikia.com, 2017)	28
Abbildung 28: Primefaces (doubleslash.de, 2017)	29
Abbildung 29: Login-Seite	38
Abbildung 30: Entwickler-Zugang Mandanten-Seite	39
Abbildung 31: Mandanten Zugang - Menüband	40
Abbildung 32: Home-Seite	41
Abbildung 33: Paket-Seite	42
Abbildung 34: Übungsseite	43
Abbildung 35: Übungsschritt-Seite	44
Abbildung 36: Zuteilungsseite	45
Abbildung 37: Kategorie-Seite	46
Abbildung 38: Firmen-Seite	47
Abbildung 39: Person-Seite	48
Abbildung 40: Gutschein-Seite	49
Abbildung 41: Access Denied-Seite	50
Abbildung 42: Info-Message	51
Abbildung 43: Aufbau Activity (vgl.individuapp.com, 2017)	56
Abbildung 44: Activity Lifecycle	57
Abbildung 45: Funktionale Idee - Android Applikation	59
Abbildung 46: Android-Versionen Verteilung (android.com, 2017)	60
Abbildung 47: Navigation-Drawer	62



Abbildung 48: Login	63
Abbildung 49: Meine Pakete	64
Abbildung 50: Übungen-Info	65
Abbildung 51: Übungen	65
Abbildung 52: Ausführungsschritte	66
Abbildung 53: Store	67
Abbildung 54: Gutscheine	68
Abbildung 55: Profil.....	69
Abbildung 56: Passwort bearbeiten	70
Abbildung 57: Über uns	71
Abbildung 58: gesamtes Datenmodell von EasyGym.....	72
Abbildung 59: Tabelle Mandant.....	73
Abbildung 60: Tabelle Firma	73
Abbildung 61: Tabelle Endnutzer	74
Abbildung 62: Tabelle Gutschein	74
Abbildung 63: Tabelle Paket	74
Abbildung 64: Beziehungstabelle zwischen Paket und Übung.....	75
Abbildung 65: Tabelle Übungen.....	75
Abbildung 66: Beziehungstabelle zwischen Übung und Kategorie	75
Abbildung 67: Tabelle Übungsschritt	75
Abbildung 68: Tabelle Kategorie	76
Abbildung 69: Tabelle Medium	76
Abbildung 70: Tabelle Medien-Typ	76
Abbildung 71: Beziehungstabelle zwischen Endnutzer und Paket.....	77
Abbildung 72: Beziehungstabelle zwischen Endnutzer und Übung	77
Abbildung 73: Tabelle Anmeldedaten.....	77
Abbildung 74: Projektstrukturplan.....	88
Abbildung 75:Meilensteine	89



16.4 Tabellenverzeichnis

Tabelle 1: Zusammensetzung REST-Pfad	32
Tabelle 2: gesamter REST-Pfad	32
Tabelle 3: Activity Methoden	58
Tabelle 4: Zeitaufzeichnung Heimel	92
Tabelle 5: Zeitaufzeichnung Hennerbichler	95
Tabelle 6: Risiko Projektumfang.....	96
Tabelle 7: Risiko Zeitmanagement	96
Tabelle 8: Risiko know-how	97
Tabelle 9: Risiko Ausfall eines Entwicklers	97
Tabelle 10: Risiko Product Owner springt ab	98
Tabelle 11: Risiko Zeitdruck	98
Tabelle 12: Risiko am Markt vorhanden	98
Tabelle 13: Risiko Lizenzen kostenpflichtig	99