

StayHealthy

Projektplan

Team 6

Khader AlHamed, Marco Klein
Denis Manherz, Andreas Wirth

15. Mai 2022

Software Praktikum

Inhaltsverzeichnis

1	Dokumentinformationen	4
1.1	Änderungsgeschichte	4
2	Einführung	4
2.1	Definitionen und Abkürzungen	4
2.2	Referenzen	4
2.3	Übersicht	4
3	Projektübersicht	4
3.1	Zweck und Ziel	5
3.2	Annahmen und Einschränkungen	5
3.3	Arbeitsergebnisse	5
4	Projektorganisation	5
4.1	Organisationsstruktur	5
4.2	Externe Schnittstellen	6
5	Management Abläufe	6
5.1	Projekt Kostenvoranschlag	6
5.2	Projektplan	6
5.2.1	Zeitplan	6
5.2.2	Arbeitspakete Allgemeine Aufwendungen	7
5.2.3	Arbeitspakete Use-Cases	7
5.2.4	Zeitaufwand Gesamt	8
5.2.5	Iterationsplanung/Meilensteine	8
5.2.6	Besprechungen	9
5.2.7	Abgabe	9
6	Risiko Management	9
7	Arbeitspakete	10
7.1	Arbeitspaket 01 Visiondokument	10
7.2	Arbeitspaket 02 Anforderungsspezifikation	10
7.3	Arbeitspaket 03 Projektplan	10
7.4	Arbeitspaket 04 Use Cases	10
7.5	Arbeitspaket 05 UML-Diagramme	11
7.6	Arbeitspaket 06 Installation und Konfiguration	11
7.7	Arbeitspaket 07 Einarbeitung	11
7.8	Arbeitspaket 08 Datenbank aufsetzen	11
7.9	Arbeitspaket 09 Softwaredokumentation	11

7.10	Arbeitspaket 10 Code Review	11
7.11	Arbeitspaket 11 Tests	12
7.12	Arbeitspaket 12 Meetings	12
7.13	Arbeitspaket 13 Abschlusspräsentation	12
7.14	Arbeitspaket 14 Ansichten erstellen	12
7.15	Arbeitspaket 15 Benutzer registrieren	12
7.16	Arbeitspaket 16 Benutzer anmelden	13
7.17	Arbeitspaket 17 Profil bearbeiten	13
7.18	Arbeitspaket 18 Profil aufwerten	13
7.19	Arbeitspaket 19 Grundumsatz berechnen	13
7.20	Arbeitspaket 20 Mahlzeit eingeben	13
7.21	Arbeitspaket 21 Kalorien berechnen	14
7.22	Arbeitspaket 22 UC Trainingsplan erstellen	14
7.23	Arbeitspaket 23 Auto Trainingsplan erstellen	14
7.24	Arbeitspaket 24 Trainingseinheit erstellen	14
7.25	Arbeitspaket 25 Ernährungsplan erstellen	14
7.26	Arbeitspaket 26 Statistik erstellen	15
7.27	Arbeitspaket 27 Erinnerung senden	15
7.28	Arbeitspaket 28 Termine bearbeiten	15
7.29	Arbeitspaket 29 Training durchführen	15
7.30	“Arbeitspaket” 30 Puffer	15
8	Infrastruktur	15
9	Unterstützende Prozesse	16

1 Dokumentinformationen

1.1 Änderungsgeschichte

Datum	Version	Änderung	Autor
25.03.2022	0.0	Erstellung	Manherz
27.03.2022	1.0	Grobe Ausarbeitung	Manherz
31.03.2022	1.1	Ergänzungen allgemein	Wirth
02.04.2022	1.2	Ergänzung Arbeitspakete	Wirth
04.04.2022	1.3	Anpassungen nach Rücksprache	Wirth
26.04.2022	1.4	Meilensteine überarbeitet	Wirth
28.04.2022	1.5	Ergänzung unter 3.2, 6 und 9	Wirth
29.04.2022	1.6	Anpassungen unter 5.2 und 7	Wirth

2 Einführung

2.1 Definitionen und Abkürzungen

Ernährungsverhalten - Ernährungsbezogene Handlungen die Menschen im Alltag vollziehen.

Gesundheitsverhalten - Handlungen von gesunden Menschen die das Risiko von Erkrankungen nachweislich senken oder welche die Gesundheit positiv beeinflussen.

2.2 Referenzen

scweb.uhcl.edu - Software Development Plan

2.3 Übersicht

Der Inhalt dieses Dokuments beschreibt den Umfang des Projekts StayHealthy, sowie die zeitlichen Rahmenbedingungen und die generelle Organisation.

3 Projektübersicht

Mit Hilfe der StayHealthy Software kann der Benutzer Trainings- und Ernährungspläne erstellen. In ihrem Zeitplan können sie diese dann ansehen bzw. durchführen. Im allgemeinen soll dies dem Benutzer dabei helfen sich an eine Ernährungs- bzw. Trainingsroutine zu halten.

3.1 Zweck und Ziel

StayHealthy soll dazu dienen das Gesundheitsverhalten des Benutzers zu verbessern indem es sein Trainings- und Ernährungsverhalten überwacht und ihm sportliche Aktivitäten vorschlägt.

3.2 Annahmen und Einschränkungen

Eine funktionsfähige Datenbankanbindung ist für dieses Projekt unabdinglich, um den Fortschritt des Benutzer über die Zeit nachvollziehen zu können und um entsprechende Vorschläge anzubieten.

Der Projektverlauf wird stetig in den wöchentlichen Meetings überprüft (Vergleich Soll - Ist) und wird ggf. angepasst.

Jedes Teammitglied ist für die individuelle Zeiterfassung seiner aufgetragenen Arbeitszeit selbst zuständig. Diese wird jedoch in einem gemeinsamen Dokument gespeichert, um den nötigen Überblick zu behalten.

3.3 Arbeitsergebnisse

Arbeitsergebnis	Beschreibung
StayHealthy	Die fertige Software
StayHealthy-Code	Kommentierter Code
Projektdokumentation	Beschreibt wie die Software entwickelt werden soll
Softwaredokumentation	Erklärt die Funktionen der Software sowie ihre Anwendung
UML-Diagramme	Alle Diagramme die für das Verständnis der Architektur notwendig sind
Zeitpool	Die vollständige Zeiterfassung
Testdokumente	Strukturierte Tests der Software

4 Projektorganisation

4.1 Organisationsstruktur

Teamübersicht:

Khader Alhamed	khader1.alhamed@st.oth-regensburg.de
Marco Klein	marco.klein@st.oth-regensburg.de
Denis Manherz	denis.manherz@st.oth-regensburg.de
Andreas Wirth	andreas2.wirth@st.oth-regensburg.de

Verantwortungsbereich	Verantwortlicher
Projekt Management	Klein
Systemadministration	Manherz
Software Architektur	AlHamed
Qualitätsmanagement	Wirth
User Interface Design	Alle
Implementierung	Alle
Code Review	Alle

4.2 Externe Schnittstellen

Stetiger Ansprechpartner und Betreuer des Projekts ist Prof. Dr. Axel Doering. Als Team nehmen wir bei Prof. Dr. Doering Beratungstermine wahr und besprechen Projektergebnisse in Reviewterminen.

5 Management Abläufe

5.1 Projekt Kostenvoranschlag

Für das Projekt entstehen vorerst keine Kosten, da es lokal auf einem System installiert werden kann. Falls die Datenbank später ausgelagert wird benötigen wir einen Backendserver. Dieser steht über die virtuelle Maschine, auf die wir im Rahmen des Projekts Zugriff haben, zur Verfügung. Nach Abschluss des Projekts können Kosten für das Betreiben dieses Servers anfallen.

5.2 Projektplan

Die zeitliche Dauer des Projekts umfasst 16 Wochen, wobei die Schlussabgabe am 07.07.2022 ist. Ein genauer Termin der Abschlusspräsentation ist zur Zeit noch nicht bekannt. Die genaue Zeiteinteilung der einzelnen Phasen und Iterationen werden im Zeitplan aufgeführt

5.2.1 Zeitplan

Der angestrebte Aufwand pro Teammitglied ist 150 Stunden, folglich eine gesamte Arbeitszeit von 600 Stunden. Daraus ergibt sich im Durchschnitt eine wöchentliche Arbeitszeit von 9-10 Stunden für jeden Teilnehmer. In allen Arbeitspaketen ist Fehlermanagement mit inbegriffen. Die Angaben im Zeitplan beziehen sich auf Stunden.

5.2.2 Arbeitspakete Allgemeine Aufwendungen

#	Arbeitspaket	DM	MK	AW	KA	Gesamt
01	Visiondokument		5			5
02	Anforderungsspezifikation	10				10
03	Projektplan	3		3	3	9
04	Use Cases		8		8	16
05	UML-Diagramme		8	4	4	16
06	Installation und Konfiguration	2	2	2	2	8
07	Einarbeitung	5	5	5	5	20
08	Datenbank aufsetzen	4		4		8
09	Softwaredokumentation		8		8	16
10	Code Review	10	10	10	10	40
11	Tests	15		15		30
12	Meetings	32	32	32	32	128
13	Abschlusspräsentation	2	2	2	2	8
	Gesamt	83	80	77	74	314

5.2.3 Arbeitspakete Use-Cases

#	Arbeitspaket	DM	MK	AW	KA	Gesamt
14	Ansichten erstellen		10		10	20
15	Benutzer registrieren	4		4		8
16	Benutzer anmelden	4		4		8
17	Profil bearbeiten	4		4		8
18	Profil aufwerten	4		4		8
19	Grundumsatz berechnen		2		2	4
20	Mahlzeit eingeben		6		6	12
21	Kalorien berechnen		4		4	8
22	UC Trainingsplan erstellen		10		10	20
23	Auto Trainingsplan erstellen	10		10		20
24	Trainingseinheit erstellen	10		10		20
25	Ernährungsplan erstellen		10		10	20
26	Statistik erstellen	10		10		20
27	Erinnerung senden	4		4		8
28	Termine bearbeiten	4		4		8
29	Training durchführen		8		8	16
	Gesamt	54	50	54	50	208

5.2.4 Zeitaufwand Gesamt

#	Arbeitspaket	DM	MK	AW	KA	Gesamt
30	Puffer	25	25	25	25	100
	Allgemeine Aufwendungen	83	80	77	74	314
	Use-Cases	54	50	54	50	208
	Gesamt	162	155	156	149	622

5.2.5 Iterationsplanung/Meilensteine

Die Entwicklung erfolgt iterativ. Angestrebte Meilensteine werden ständig überprüft und wenn nötig agil angepasst. Einzelne Code Module werden stetig dokumentiert und getestet.

Deadlines orientieren sich nach den vorgegebenen Abgabeterminen/Meilensteinen von Herrn Doering.

- Meilenstein 00 “Projektantrag” 19.03.2022
 - Projektantrag wurde eingereicht
- Meilenstein 01 “Projektplan” 04.04.2022
 - Ausgearbeiteter Projektplan liegt vor
- Meilenstein 02 “Requirements” 22.04.2022
 - Alle nötigen UML-Diagramme wurden erstellt
 - Datenbank wurde aufgesetzt, ist funktionsfähig und CRUD-Funktionalitäten sind implementiert
 - GUI wurde mit nötigen (noch nicht funktionsfähigen) Elementen (Buttons usw.) erstellt
- Meilenstein 03 “Architektur” 19.05.2022
 - Benutzer kann Profil erstellen/bearbeiten/löschen
 - Profil kann aufgewertet werden
 - Mahlzeiten können eingegeben werden
 - Trainingsplan erstellen
 - Testplan für Code wurde erstellt
- Meilenstein 04 “Design” 10.06.2022

- Training kann vorgeschlagen werden.
- Benutzer kann ein vorgeschlagenes Training durchführen.
- Benutzer kann Ernährungs-/Trainingsplan erstellen
- Benutzer kann Statistik anzeigen
- Benutzer kann Zeitplan bearbeiten
- Meilenstein 05 “Schlussabgabe” 07.07.2022
 - Finale Projektversion ist fertiggestellt
 - Software Dokumentation ist vorhanden
- Meilenstein 06 “Präsentation” 19.07.2022
 - Abschlusspräsentation ist vorbereitet

5.2.6 Besprechungen

Das gesamte Team trifft sich regelmäßig zu Teammeetings und zu den verpflichtenden Reviewterminen an der OTH. Bei Notwendigkeit können außerplanmäßige Teammeetings auch über eine Videokonferenz stattfinden. Die Sitzungen belaufen sich auf ca. zwei Stunden wöchentlich. Anstehende Aufgaben, bis zum nächsten Teammeeting werden verteilt, Unklarheiten sowie Fragen werden besprochen und gelöst.

5.2.7 Abgabe

Art	Abgabe spätestens	Inhalt
Prototyp	19.05.22	siehe Meilenstein 03
Vorläufiges Endprodukt	24.06.22	siehe Meilenstein 04

6 Risiko Management

Mögliche Risiken und Problemlösungen

Risiko	Lösungsansatz
Datenbankanbindung nicht funktionsfähig	Datenbankanbindung rechtzeitig und ausgiebig testen, bei nicht-Funktionalität alternative DB integrieren
Unerwarteter (temporärer) Ausfall von Teammitglied (z.B. durch Krankheit)	Aufgaben an andere Mitglieder verteilen und ggf. Projektumfang anpassen

7 Arbeitspakete

Die Arbeitspakete werden im Laufe des Projekts angepasst.

7.1 Arbeitspaket 01 Visiondokument

Zeit	5
Ressource	Marco Klein
Inhalt	Visiondokument
Abhängigkeit	keine

7.2 Arbeitspaket 02 Anforderungsspezifikation

Zeit	10
Ressource	Manherz
Inhalt	Anforderungsspezifikation
Abhängigkeit	Use Cases 7.4

7.3 Arbeitspaket 03 Projektplan

Zeit	9
Ressource	AlHamed, Manherz, Wirth
Inhalt	Ausgearbeiteter Projektplan
Abhängigkeit	keine

7.4 Arbeitspaket 04 Use Cases

Zeit	16
Ressource	Klein, AlHamed
Inhalt	Ausarbeitung der Use Cases
Abhängigkeit	keine

7.5 Arbeitspaket 05 UML-Diagramme

Zeit	16
Ressource	AlHamed, Klein, Wirth
Inhalt	Use Case Diagramm, Entity Relationship Modell
Abhängigkeit	keine

7.6 Arbeitspaket 06 Installation und Konfiguration

Zeit	8
Ressource	Alle
Inhalt	Installieren und Konfigurieren von benötigten Tools
Abhängigkeit	keine

7.7 Arbeitspaket 07 Einarbeitung

Zeit	20
Ressource	Alle
Inhalt	Einarbeitung in Benötigte Tools und Technologien
Abhängigkeit	keine

7.8 Arbeitspaket 08 Datenbank aufsetzen

Zeit	8
Ressource	Manherz, Wirth
Inhalt	Erstellen von Tabellen
Abhängigkeit	keine

7.9 Arbeitspaket 09 Softwaredokumentation

Zeit	16
Ressource	Manherz, Klein
Inhalt	Softwaredokumentation
Abhängigkeit	Geschriebener Code

7.10 Arbeitspaket 10 Code Review

Zeit	40
Ressource	Alle
Inhalt	Überprüfung des Codes anhand vorhandener Spezifikationen und Coding-Guidelines
Abhängigkeit	Geschriebener Code

7.11 Arbeitspaket 11 Tests

Zeit	30
Ressource	Manherz, Wirth
Inhalt	Test der Funktionsfähigkeit von einzelnen Komponenten sowie Fehlersuche
Abhängigkeit	Geschriebener Code

7.12 Arbeitspaket 12 Meetings

Zeit	128
Ressource	Alle
Inhalt	Regelmäßiges Treffen zur Besprechung über das Projekt, beinhaltet auch Beratungs- und Reviewtermine
Abhängigkeit	Funktionsfähiges Internet/Fortbewegungsmittel

7.13 Arbeitspaket 13 Abschlusspräsentation

Zeit	8
Ressource	Alle
Inhalt	Ausarbeitung und Einstudieren der Präsentation
Abhängigkeit	Fertiges Projekt

7.14 Arbeitspaket 14 Ansichten erstellen

Zeit	20
Ressource	AlHamed, Manherz
Inhalt	Asichten mit Button/Feld Dummys erstellen
Abhängigkeit	keine

7.15 Arbeitspaket 15 Benutzer registrieren

Zeit	8
Ressource	Manherz, Wirth
Inhalt	Verknüpfung von Button mit Funktionalität, Datenbankbindung, wie UC Benutzer registrieren
Abhängigkeit	Datenbank einsatzfähig

7.16 Arbeitspaket 16 Benutzer anmelden

Zeit	8
Ressource	Manherz, Wirth
Inhalt	Verknüpfung von Button mit Funktionalität, Datenbankbindung,
Abhängigkeit	Datenbank einsatzfähig

7.17 Arbeitspaket 17 Profil bearbeiten

Zeit	8
Ressource	Manherz, Wirth
Inhalt	Verknüpfung von Buttons/Eingabefeldern mit Funktionalität, Datenbankbindung,
Abhängigkeit	Datenbank einsatzfähig

7.18 Arbeitspaket 18 Profil aufwerten

Zeit	8
Ressource	Manherz, Wirth
Inhalt	Verknüpfung von Button mit Funktionalität, Datenbankbindung,
Abhängigkeit	Datenbank einsatzfähig

7.19 Arbeitspaket 19 Grundumsatz berechnen

Zeit	4
Ressource	AlHamed, Klein
Inhalt	Grundumsatz eines Benutzers berechnen und speichern
Abhängigkeit	Datenbank einsatzfähig

7.20 Arbeitspaket 20 Mahlzeit eingeben

Zeit	12
Ressource	AlHamed, Klein
Inhalt	Verknüpfung von Buttons/Eingabefeldern mit Funktionalität, Datenbankbindung
Abhängigkeit	Datenbank einsatzfähig

7.21 Arbeitspaket 21 Kalorien berechnen

Zeit	8
Ressource	AlHamed, Klein
Inhalt	Berechnung und Speicherung von Kalorien einer vom Nutzer eingegebenen Mahlzeit
Abhängigkeit	Datenbank einsatzfähig

7.22 Arbeitspaket 22 UC Trainingsplan erstellen

Zeit	20
Ressource	AlHamed, Klein
Inhalt	Verknüpfung von Buttons/Feldern mit Funktionalität, Datenbankbindung
Abhängigkeit	Datenbank einsatzfähig

7.23 Arbeitspaket 23 Auto Trainingsplan erstellen

Zeit	20
Ressource	Manherz, Wirth
Inhalt	Automatische Erstellung von Plänen mithilfe der dazugehörigen Nutzerdaten, Datenbankbindung
Abhängigkeit	Datenbank einsatzfähig

7.24 Arbeitspaket 24 Trainingseinheit erstellen

Zeit	20
Ressource	Manherz, Wirth
Inhalt	Verknüpfung von Buttons/Feldern mit Funktionalität, Datenbankbindung
Abhängigkeit	Datenbank einsatzfähig

7.25 Arbeitspaket 25 Ernährungsplan erstellen

Zeit	20
Ressource	AlHamed, Klein
Inhalt	Verknüpfung von Buttons/Feldern mit Funktionalität, Datenbankbindung
Abhängigkeit	Datenbank einsatzfähig

7.26 Arbeitspaket 26 Statistik erstellen

Zeit	20
Ressource	Manherz, Wirth
Inhalt	Erstellung einer Statistik aus den Benutzerdaten in der Datenbank, Datenbankankbindung
Abhängigkeit	Datenbank einsatzfähig

7.27 Arbeitspaket 27 Erinnerung senden

Zeit	8
Ressource	Manherz, Wirth
Inhalt	System sendet Erinnerung an bevorstehende Trainingseinheiten
Abhängigkeit	Datenbank einsatzfähig

7.28 Arbeitspaket 28 Termine bearbeiten

Zeit	8
Ressource	Manherz, Wirth
Inhalt	Verknüpfung von Buttons/Feldern mit Funktionalität, Datenbankankbindung
Abhängigkeit	Datenbank einsatzfähig

7.29 Arbeitspaket 29 Training durchführen

Zeit	16
Ressource	AlHamed, Klein
Inhalt	Verknüpfung von Buttons/Feldern mit Funktionalität, Datenbankankbindung
Abhängigkeit	Datenbank einsatzfähig

7.30 “Arbeitspaket” 30 Puffer

Zeit	100
Ressource	Alle
Inhalt	Puffer für wahrscheinliche Änderungen im Zeitplan
Abhängigkeit	keine

8 Infrastruktur

- Visual Studio Community 2022 IDE

- GitLab
- Qt Creator
- Overleaf
- Lucidchart

9 Unterstützende Prozesse

Als Tool zur Modellierung der Software Architektur in UML benutzen wir Lucidchart. Zur Implementierung benutzen wir die Programmiersprache C++. Als Entwicklungsumgebung verwenden wir Microsoft Visual Studio, da diese uns den Windows Build Prozess erleichtert und Integrationen bzw. Tools für Git, Qt und SQLite zur Verfügung stellt.

Für das GUI verwenden wir das Qt Framework mit dem Qt Designer. SQLite wird für die permanente Datenhaltung verwendet.

Um sicherzustellen, dass alle Dateien, sowie Entwicklungsstände wiederherstellbar sind, benutzen wir das von der OTH Regensburg bereitgestellte GitLab Repository. Sobald ein Arbeitspaket abgeschlossen ist wird es committed und reviewed. Fehler bzw. Bugs werden in unserem GitLab als Issues dokumentiert und frühestmöglich behoben.

Um die Qualität des Codes sicherzustellen richten wir uns nach dem Google C++ Style Guide. Außerdem wird der gesamte Code kommentiert, um sicherzustellen, dass er für alle Gruppenmitglieder verständlich ist.

Der Code für das Projekt wird in folgendem Git-Repository verwaltet:

https://gitlab.oth-regensburg.de/IM/SWP_IM4/swp_sose22/team-6

Die repository Struktur baut sich folgendermaßen auf.

Ausgehend vom branch "main" der branch "Development".

Auf dem branch "main" findet keinerlei Entwicklung statt. Dieser ist für releases reserviert.

Für einzelne Features/Arbeitspakete wird vom branch "Development" ein neuer branch abgezweigt.

Nach Fertigstellung wird dieser wieder zurück in den "Development" branch gemerged.

Tests können sowohl auf diesen sub-branches, als auch auf dem Development branch selbst durchgeführt werden.

Releases/Meilensteine/Prototypen werden auf den branch "main" zurück gemerged und werden entsprechend mit einem tag versehen.