

# StayHealthy

## Software Architektur Spezifikation

Team 6

Andreas Wirth, Marco Klein, Khader AlHamed, Denis  
Manherz

15. Mai 2022

Software Praktikum

# Inhaltsverzeichnis

<b>1</b>	<b>Dokumentinformationen</b>	<b>3</b>
1.1	Änderungsgeschichte . . . . .	3
<b>2</b>	<b>Einführung</b>	<b>3</b>
2.1	Definitionen und Abkürzungen . . . . .	3
2.2	Referenzen . . . . .	3
2.3	Übersicht . . . . .	3
<b>3</b>	<b>Architektonische Darstellung</b>	<b>3</b>
<b>4</b>	<b>Architektonische Ziele und Einschränkungen</b>	<b>3</b>
<b>5</b>	<b>Logische Architektur</b>	<b>4</b>
5.1	Übersicht . . . . .	5
5.2	Design Pakete . . . . .	6
5.2.1	Package GUI . . . . .	6
5.2.1.1	Beschreibung des Package . . . . .	6
5.2.1.2	Diagramme . . . . .	6
5.2.1.3	Schnittstellen . . . . .	6
5.2.1.4	Operationen . . . . .	6
5.2.1.4.1	Operation 1 . . . . .	6
5.2.1.4.2	Operation 2 . . . . .	6
5.2.2	Package Logik . . . . .	6
5.2.2.1	Beschreibung des Package . . . . .	6
5.2.2.2	Diagramme . . . . .	7
5.2.2.3	Schnittstellen . . . . .	7
5.2.2.4	Operationen . . . . .	7
5.2.2.4.1	Benutzer anmelden . . . . .	8
5.2.2.4.2	Profil bearbeiten . . . . .	9
<b>6</b>	<b>Physikalische Sicht</b>	<b>9</b>
<b>7</b>	<b>Prozesse und Threads</b>	<b>9</b>
<b>8</b>	<b>Datenspeicherung</b>	<b>10</b>
<b>9</b>	<b>Größen und Leistung</b>	<b>10</b>
<b>10</b>	<b>Risiken</b>	<b>11</b>
<b>11</b>	<b>Off-the-shelf Software</b>	<b>11</b>

# 1 Dokumentinformationen

## 1.1 Änderungsgeschichte

Datum	Version	Änderung	Autor
25.04.2022	0.0	Inhaltsverzeichnis	Denis Manherz
31.03.2022	1.0	Erster Entwurf	Denis Manherz
01.04.2022	1.1	Use Cases Fully Dressed	Marco Klein
01.04.2022	1.2	Use Cases Fully Dressed	Khader Alhamed
14.05.2022	1.3	UML Diagramm	Marco Klein

## 2 Einführung

### 2.1 Definitionen und Abkürzungen

GUI

bzw. beziehungsweise

### 2.2 Referenzen

-Unterlagen aus der Vorlesung SE von Prof. Dr. Doering

### 2.3 Übersicht

Dieses Dokument beschreibt die Art und den Aufbau der Softwarearchitektur von StayHealthy. Es werden einzelne Packages und deren Schnittstellen beschrieben. Zusätzlich werden Sequenzdiagramme für die jeweiligen Use Cases geliefert.

## 3 Architektonische Darstellung

Die Softwarearchitektur teilt sich auf in 4 Bereiche: Dateneingabe, Ausgabe, Datenhaltung und Analyse der Daten.

## 4 Architektonische Ziele und Einschränkungen

Der Arbeitsaufwand ist durch diese Aufteilung möglichst gleich auf die Teammitglieder verteilt. Um einen raschen Arbeitsfortschritt gewährleisten zu können werden Abhängigkeiten zwischen Teilbereichen der Architektur möglichst

klein gehalten. Des weiteren können die Teammitglieder mit den einzelnen Levels schon von Anfang an bis zu einer gewissen Stufe mit ihrer Arbeit beginnen. Erst bei Überschneidung müssen auf ein oder mehrere andere Teammitglieder gewartet werden.

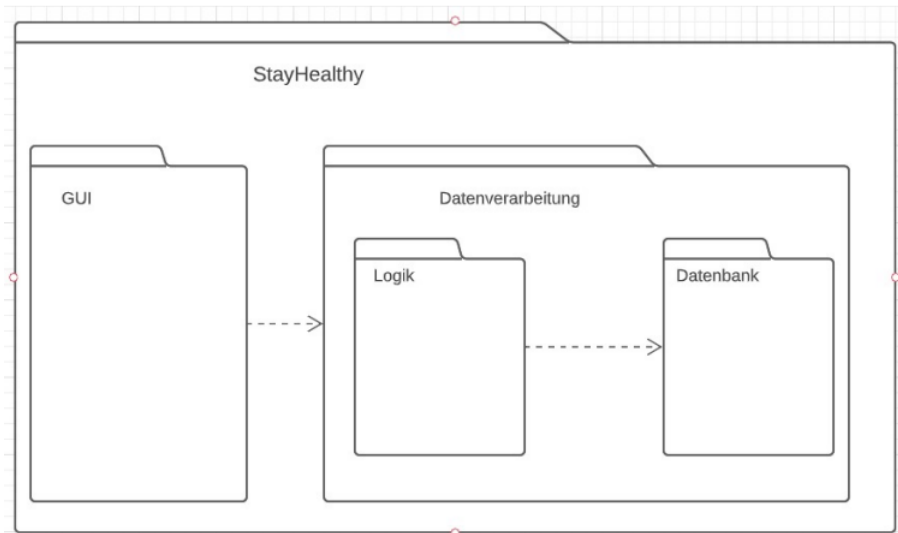
## **5 Logische Architektur**

Die Architektur ist in die Pakete GUI und Datenverarbeitung unterteilt. Die einzelnen Pakete enthalten mehrere Operationen.

Die GUI besteht aus der Anmeldung/ Registrierung, Abmeldung, dem Anlegen eines Profils/ einer Mahlzeit/ einer Trainingseinheit und außerdem die Auswahl der Ansicht.

Es können der Zeitplan bearbeitet sowie direkte Trainings- bzw. Ernährungspläne bearbeitet werden. Außerdem ist die Bearbeitung des Profils möglich.

## 5.1 Übersicht



Klasse	Nutzen
Registrierung	Beinhaltet die persönlichen Daten der Nutzer für die Registrierung.
Anmeldung	Dient zum Anmelden im Programm.
Profil	Enthält alle Attribute der persönlichen Daten, die in der Datenbank gespeichert werden.
Benutzer	Dient zur Betrachtung eines Benutzers.
Sportler	Dient zur Unterscheidung zwischen einem Trainer und normalem Benutzer.
Personaltrainer	Dient zur Unterscheidung zwischen einem Trainer und normalem Benutzer.
Plandurchführung	Dient zur Durchführung von Trainingseinheiten im Trainingsplan.
Statistik	Zeigt Statistik über den Kalorienverbrauch und sportliche Aktivität.
Trainingsplan	Dient zur Darstellung mit Attributen einer oder mehrerer Trainingseinheiten.
Trainingseinheit	Beschreibt eine oder mehrere Übungspositionen.
Übungsposition	Beinhaltet eine oder mehrere Übungen
Übung	Dient zur Beschreibung mit Attributen einer elementaren Übung.
Zeitplan	Beinhaltet Trainingsplan und Ernährungsplan.
Ernährungsplan	Dient zur Darstellung mit Attributen einer oder mehrerer Mahlzeiten.
Mahlzeit	Beinhaltet eine oder mehrere Speisepositionen.
Speiseposition	Beinhaltet eine oder mehrere Speisen.
Speise	Dient zur Beschreibung mit Attributen einer elementaren Speise.

## **5.2 Design Pakete**

### **5.2.1 Package GUI**

#### **5.2.1.1 Beschreibung des Package**

Die GUI bietet dem Benutzer die Möglichkeit durch die Anwendung zu navigieren und sich Ansichten anzeigen zu lassen oder Eingaben zu tätigen.

#### **5.2.1.2 Diagramme**

#### **5.2.1.3 Schnittstellen**

Datenverarbeitung

#### **5.2.1.4 Operationen**

##### **5.2.1.4.1 Operation 1**

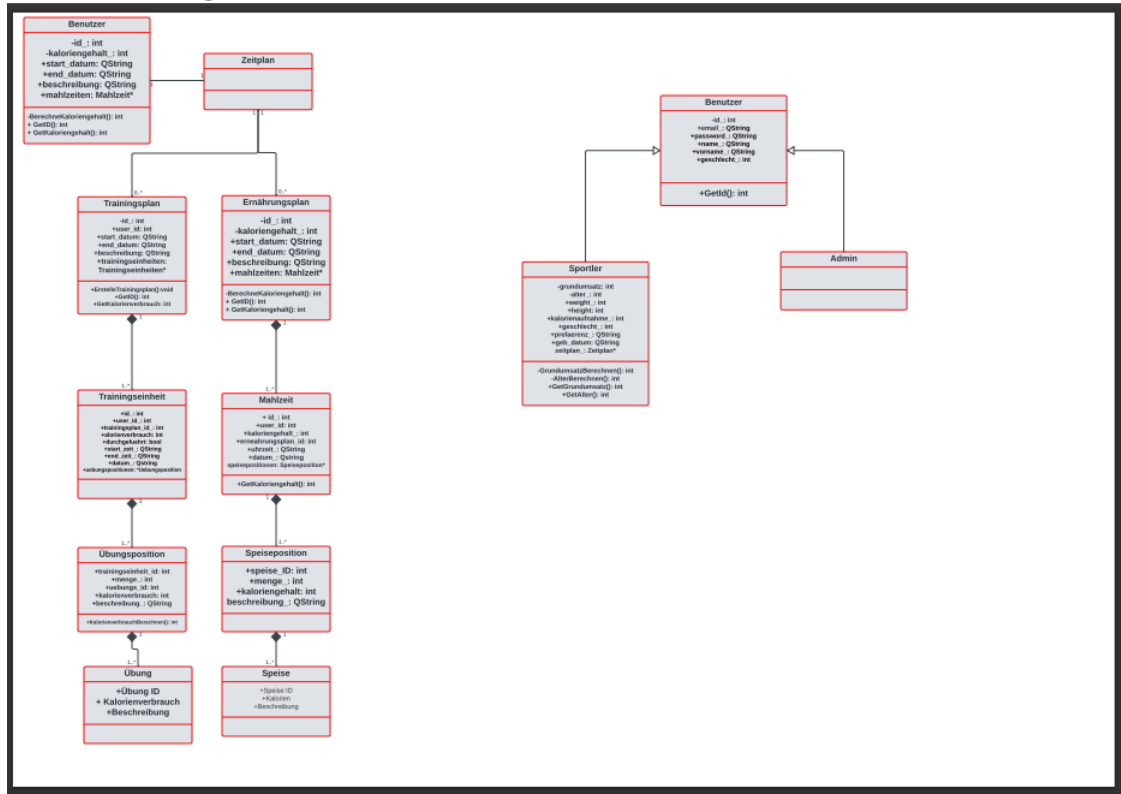
##### **5.2.1.4.2 Operation 2**

### **5.2.2 Package Logik**

#### **5.2.2.1 Beschreibung des Package**

Das Paket Logik erzeugt oder bearbeitet Objekten der Klassen Benutzer, Trainingsplan, Trainingseinheit, Übungsposition sowie Ernährungsplan, Mahlzeit und Speiseposition. Diese Klassen sind als logischer Teil der Software notwendig und erleichtern den Speichervorgang sowie das Abrufen der eingepflegten Daten in der Software.

### 5.2.2.2 Diagramme

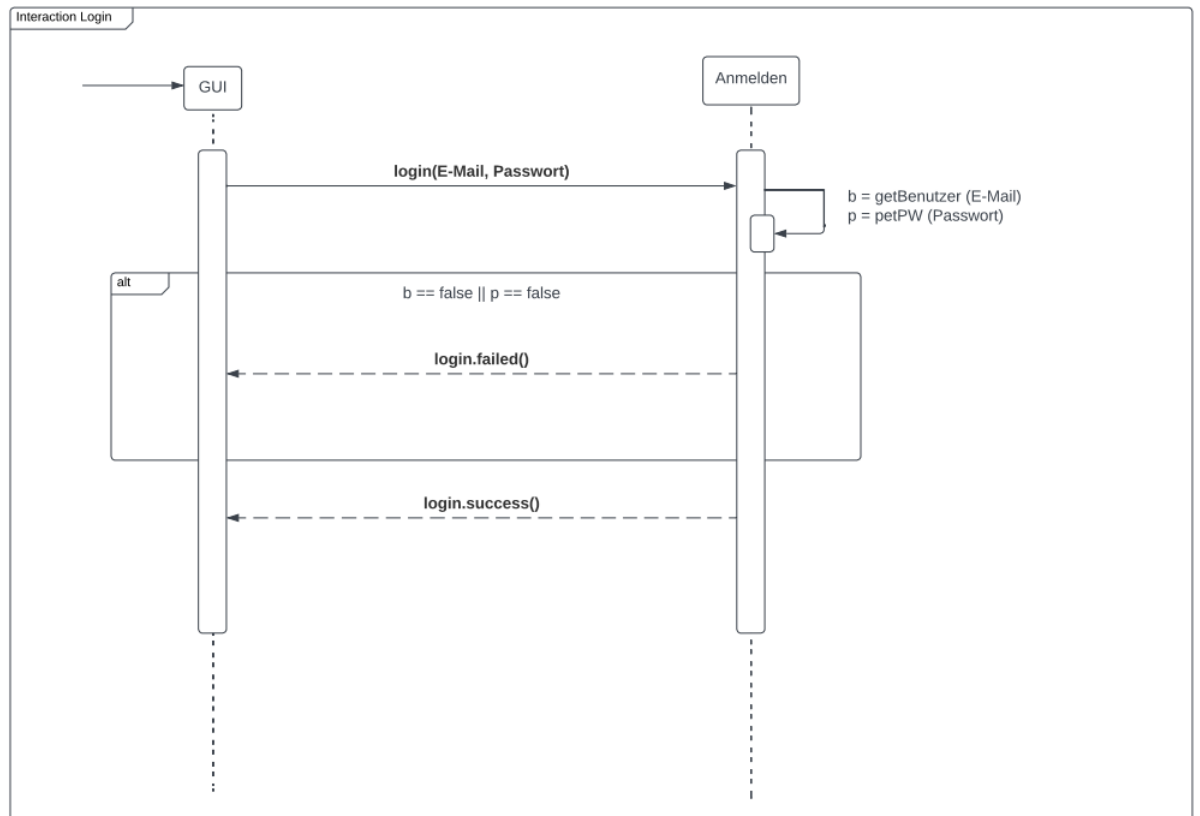


### 5.2.2.3 Schnittstellen

Das Paket Logik hat nur eine Schnittstelle zur Datenbank. Diese wird benötigt, um Daten aus der Datenbank in logische Klassen zu übersetzen.

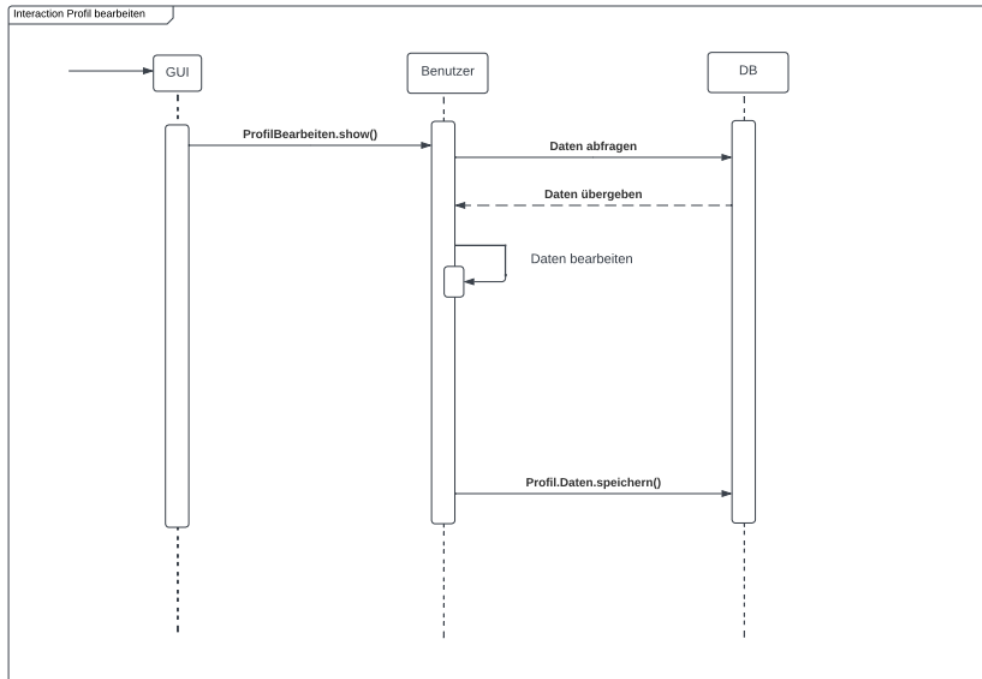
### 5.2.2.4 Operationen

### 5.2.2.4.1 Benutzer anmelden





#### 5.2.2.4.2 Profil bearbeiten



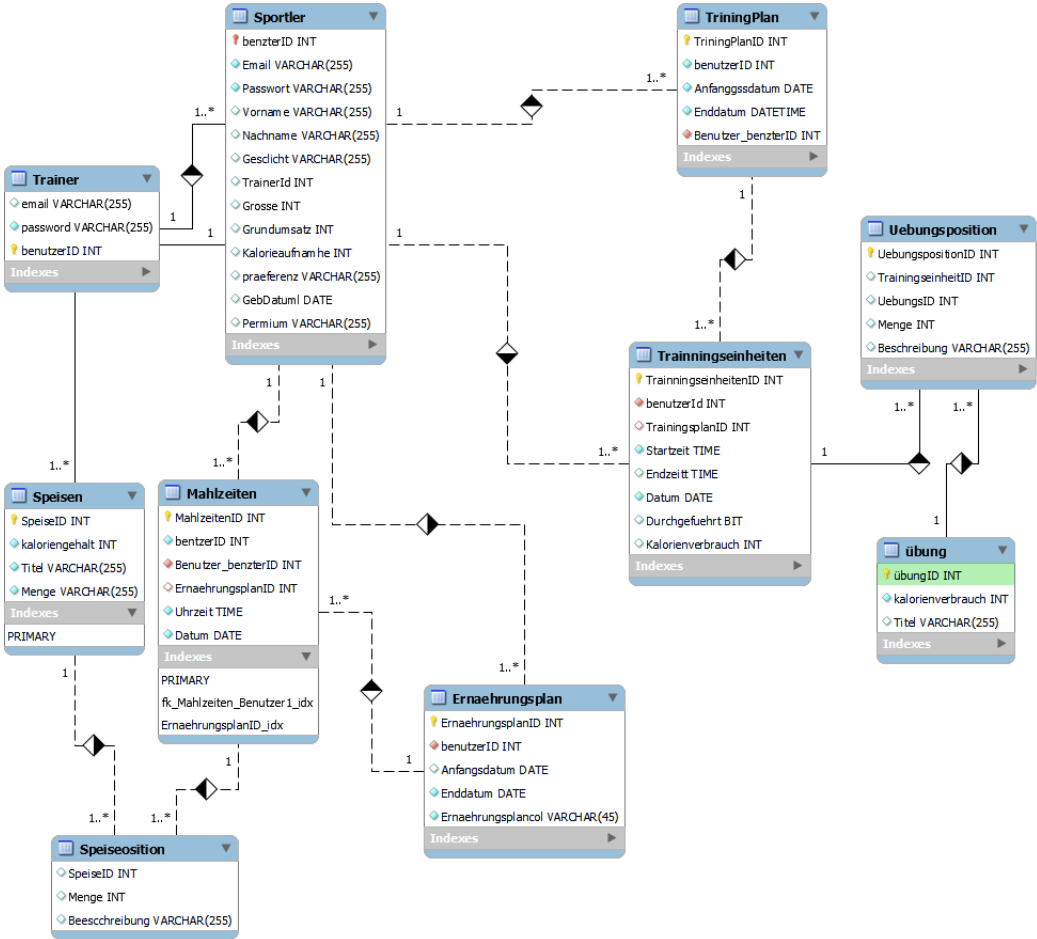
## 6 Physikalische Sicht

Für die Benutzung der StayHealthy Software wird lediglich ein Computer mit dem Betriebssystem MS Windows 10 oder neuer benötigt. Vorinstallierte Programme werden nicht benötigt. Auch spezielle Hardwareteile, die an den PC angeschlossen werden, werden nicht gebraucht. Der Benutzer interagiert mit der Anwendung über das GUI mit Maus und Tastatur. Internetverbindung ist benötigt um Die Daten auf den Datenbank hochzuladen.

## 7 Prozesse und Threads

Der Benutzer gibt in der GUI EMail, Passwort ein diese werden als QStrings an die Datenverarbeitung weitergeleitet, wo der Benutzer in der Datenbank registriert wird. Analog zur Registrierung kann sich der Benutzer anmelden. Nach dem anmelden hat der Benutzer zugriff auf alle Funktionalitäten der Anwendung. Er kann seine Daten ändern und sich seine Daten ansehen. Die Daten die aus der GUI hervorgehen werden in der Datenverarbeitung verarbeitet und wenn nötig in der Datenbank hinterlegt, gelöscht, verändert oder ausgegeben.

## 8 Datenspeicherung



## 9 Größen und Leistung

- Pro Anwendungsinstanz kann sich nur ein Benutzer anmelden.
- Benutzer und deren Profile sind einzigartig.
- Benutzerdaten werden maximal 1 Jahr gespeichert.
- Ein Benutzer kann pro Woche einen Plan erstellen diesen aber beliebig ändern.
- Ein Benutzer kann pro Tag maximal 30 Mahlzeiten bzw. Trainingseinheiten speichern.

## 10 Risiken

Nachfolgend werden alle nicht trivialen Risiken des Projekts aufgeführt:

- Aufgrund mangelnder Kommunikation auftretende Programmierfehler, die zu Speicherlecks und Sicherheitslücken führen können
- Probleme auf anderen Betriebssystemen
- zu große Menge an Daten die in der Datenbank gespeichert werden und einen daraus resultierenden Programmabsturz.
- Die Eingaben der Daten sind nur auf Deutsch verfügbar.

## 11 Off-the-shelf Software

**Bestriebssystem:**

- Windows

**Bibliotheken:**

- QtGui
- QtWidgets
- QSql
- QtCore