**Pflichtenheft**

# **Inhaltsverzeichnis**

[**Inhaltsverzeichnis**](#_6csznqt6l0j8) **1**

[**Beschreibung**](#_t3llgmghkctz) **2**

[**Team**](#_6d3z2jdw5irf) **2**

[**Ziele & Nicht-Ziele**](#_izjrxueclgpp) **2**

[**Technologien & Tools**](#_ujtl14221zpv) **3**

[**Meilensteinliste**](#_1fk66yhh7cm4) **3**

[User Stories](#_hqym959ecz0v) **4**

# **Beschreibung**

Fitervari Health kann als Erweiterung zu Fitervari ([Pflichtenheft](https://docs.google.com/document/d/1XDzwYeK8wlecgJfIaNlGApOXHXopa-42w715cagviKk/edit?usp=sharing)) bei der Erstellung digitaler Trainingspläne helfen, indem es Gesundheitsdaten wie Puls und den Blutsauerstoffgehalt über Sensoren auf Smartwatches liest und diese dem Trainer zur verfügung stellt. Die Trainingspläne werden auf Basis der Kundendaten bzw. aufgrund von medizinischen Einschränkungen, unter Berücksichtigung der verfügbaren Geräte und gemessenen Gesundheitsdaten durch Fachpersonal erstellt. Der Kunde kann seinen Trainingsplan einfach auf seinem Smartphone abrufen und dabei gleichzeitig auch die Trainingsresultate und Gesundheitsdaten erfassen.

# **Team**

| **Projektname** | Fitervari Health | | |
| --- | --- | --- | --- |
| **Projektleiter** | Kern Tobias | +43 650 6740369 | kern.tobias@protonmail.com |
| **Projektteam** | Anzinger Stefan | Rausch-Schott Manuel | Steininger Markus |

# 

# **Ziele & Nicht-Ziele**

Die Ziele des Projekts sind folgende:

* Auslesung der Gesundheitsdaten des Kundens über Smartwatch
* Zurverfügungstellung der Gesundheitsdaten an den Trainer
* Sinnvolle Aufbereitung und Darstellung der Daten

Dabei **NICHT** implementiert werden folgende Punkte (Nicht-Ziele):

* Kompatibilität mit allen Smartwatch-Herstellern

# **Technologien & Tools**

| **iOS App** | **Flutter App** | **Web-App** | **Backend** |
| --- | --- | --- | --- |
| Swift | Flutter | Angular | Quarkus |
| SwiftUI | Android Studio | WebStorm | PostgreSQL |
| Xcode |  |  | IntelliJ IDEA |
| Sketch |  |  |  |

# 

# 

# **Meilensteinliste**

| **Bezeichnung** | **Termin** | |
| --- | --- | --- |
| Pflichtenheft | 01.10.2021 | |
| Detaillierte Projektplanung mit PSP & Tasks | 14.10.2021 | |
| Implementierung | 10.12.2021 | |
| Testung | 31.01.2022 | |
| Fertigstellung | 28.02.2022 | |

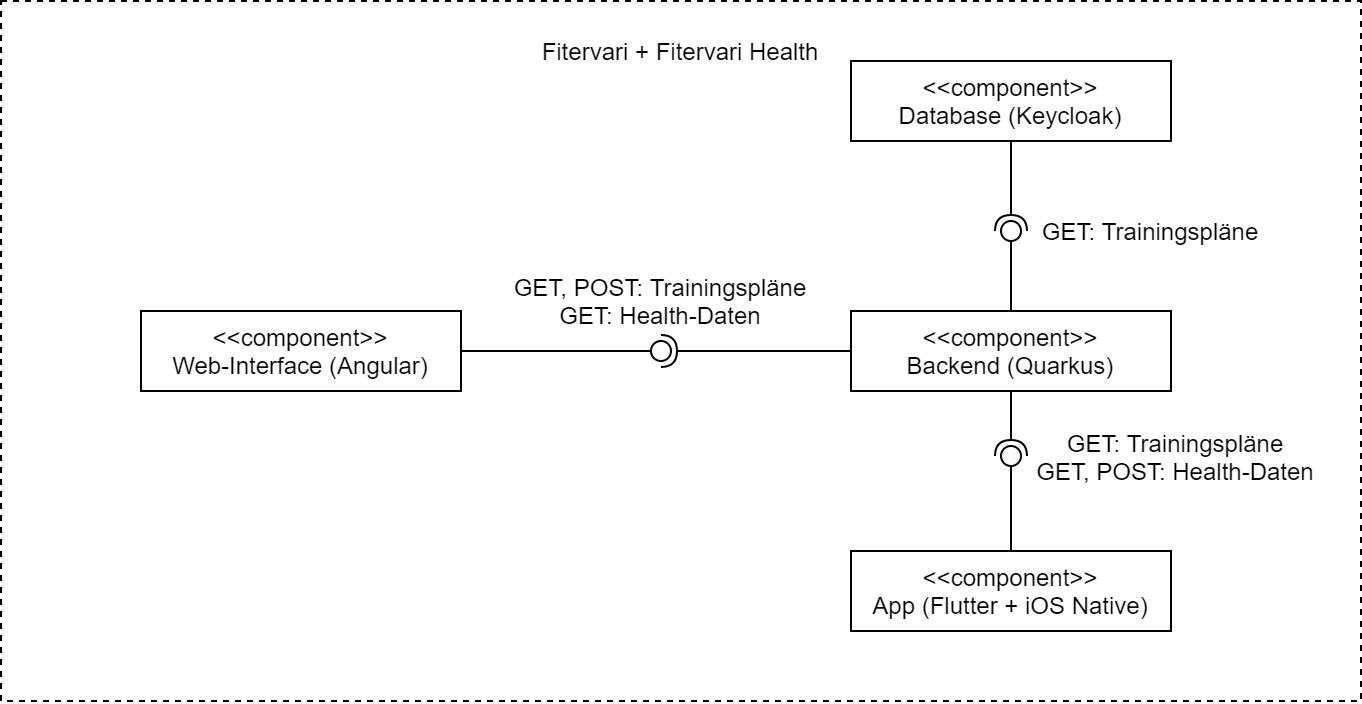
# User Stories

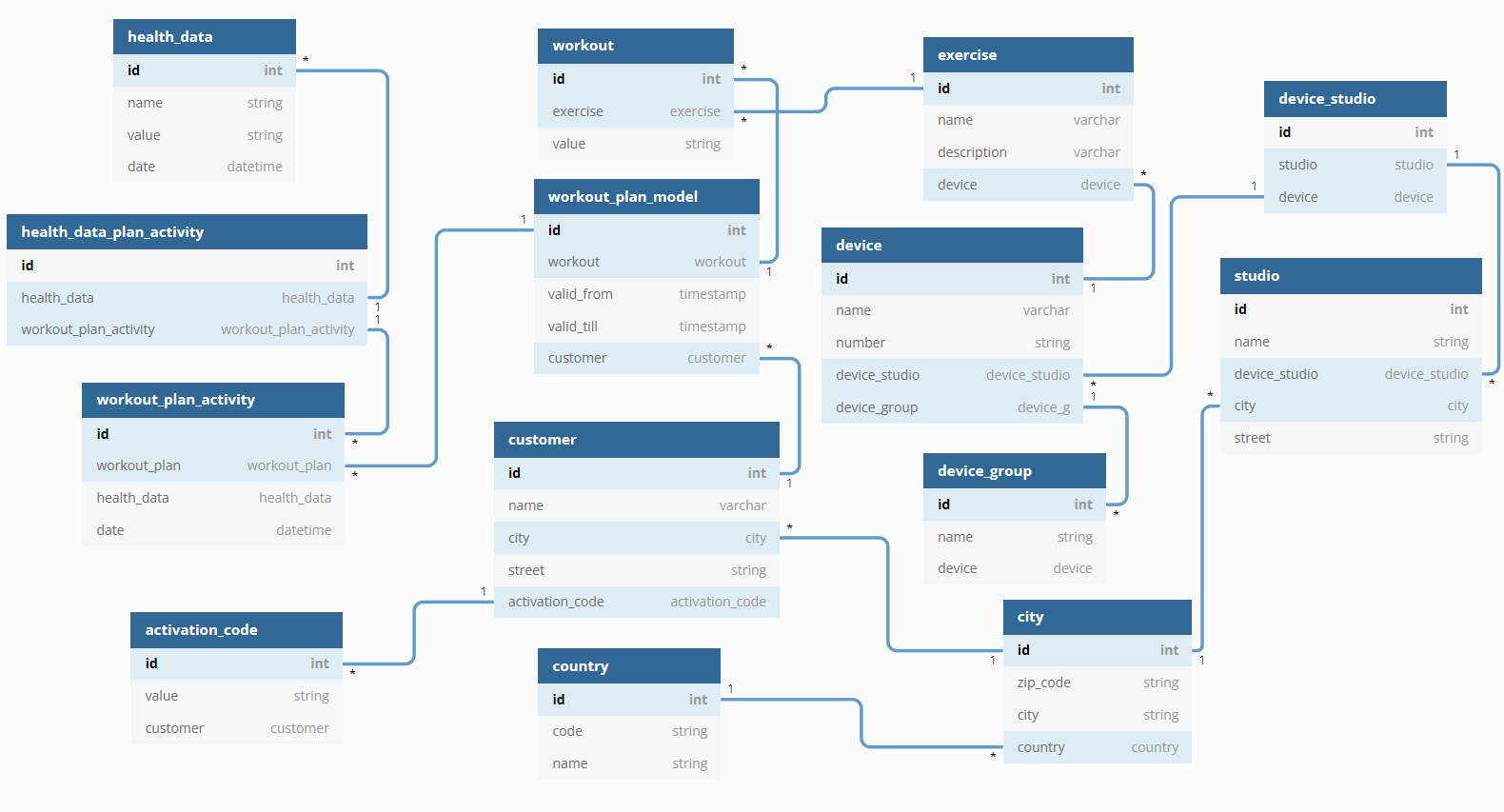
## Trainer

1. Als Trainer kann ich die Gesundheitsdaten eines Trainierenden ansehen.
2. Als Trainer kann ich erkennen, wenn Daten an bestimmten Punkten stark vom „Normalwert“ abweichen.
3. Als Trainer kann ich bestimmet Daten betrachten. (Herzfrequenz, Stresslevel, …)
4. Als Trainer kann ich die Daten einer bestimmten Zeitspanne betrachten (z.B. letzte 2 Wochen, …)
5. Als Trainer kann ich die Daten zu einem bestimmten Abschnitt im Trainingsplan (z.B. Oberkörpertraining, …) betrachten.

## Trainierender

1. Als Trainierender kann ich meine Gesundheitsdaten einsehen.
2. Als Trainierender kann ich meine Gesundheitsdaten dem Trainer zur Verfügung stellen bzw. zustimmen, dass er diese einsehen darf.
3. Als Trainierender kann ich erkennen, wenn die Daten an bestimmten Punkten stark vom „Normalwert“ abweichen.
4. Als Trainierender kann ich die Daten zu einem/r bestimmten Zeitraum/Zeitspanne (z.B. letzte 2 Wochen, …) einsehen.
5. Als Trainierender kann ich meine Daten zu einem bestimmten Abschnitt im Trainingsplan (z.B. Oberkörpertraining, …) einsehen.

**Komponentendiagramm:**

**ERD (Entity Relationship Diagram):**

**Taskliste**

| **PSP** | **Beschreibung** | **Stunden** |
| --- | --- | --- |
| 1.1 GUI-Implementierung | Navigation | 10 |
| 1.1 GUI-Implementierung | Darstellung der Gesundheitsdaten | 15 |
| 1.1 GUI-Implementierung | Auslesen von Gesundheitsdaten | 5 |
| 1.2 Verbindung | Verbindung zwischen Flutter-App und Backend-API herstellen | 5 |
| 1.2 Verbindung | Verbindung zwischen Flutter-App und Smartwatch herstellen | 20 |
| 2.1 GUI-Implementierung | Navigation | 10 |
| 2.1 GUI-Implementierung | Darstellung der Gesundheitsdaten | 15 |
| 2.1 GUI-Implementierung | Auslesen von Gesundheitsdaten | 5 |
| 2.2 Verbindung | Verbindung zwischen Flutter-App und Backend-API herstellen | 5 |
| 2.2 Verbindung | Verbindung zwischen Flutter-App und Smartwatch herstellen | 20 |
| 3.1 | Navigation | 20 |
| 3.1 | Darstellung der Gesundheitsdaten | 20 |
| 3.2 | Verbindung zum Backend | 10 |
| 4.1 Datenbank | Erweiterung um Tables für Gesundheitsdaten | 5 |
| 4.1 Datenbank | Implementierung der CRUD-Operationen | 10 |
| 4.2 REST-API | Verarbeitung von API-Requests | 25 |
| 4.2 REST-API | Bereitstellung der nötigen Endpoints | 10 |