

Projekt: FitMeUp!

Anforderungsspezifikation (Requirement Specification)

[Dokumentstruktur basiert auf RUP „Requirement Specification“]

Inhaltsverzeichnis

1. Dokumentinformationen	3
2. Allgemeine Beschreibung (Overall Description).....	4
2.1. Produkt Perspektive (Product Perspective).....	4
2.2. Produkt Funktion (Product Functions)	4
2.3. Benutzer Charakteristik (User Characteristics)	4
2.4. Einschränkungen (Constraints)	4
2.5. Abhängigkeiten (Dependenencies).....	4
2.6. Use Case Überblick (Use Case Survey)	5
3. Spezifische Anforderungen (Specific Requirements).....	5
3.1. Funktionale Anforderungen (Functionality).....	5
3.1.1. Funktionale Anforderung ‚Profil anlegen‘	5
3.1.2. Funktionale Anforderung „Daten eingeben“	5
3.1.3. Funktionale Anforderung „Challenge durchführen“	5
3.1.4. Funktionale Anforderung „Trainingsverlauf anzeigen“	5
3.2. Bedienbarkeit (Usability).....	5
3.3. Zuverlässigkeit (Reliability).....	5
3.4. Leistung (Performance)	5
3.5. Installation	6
3.6. Internationalisierung / Lokalisierung	6
3.7. Schnittstellen (Interfaces)	6
3.8. Lizenzanforderungen (Licensing Requirement).....	6
4. Use Cases	7
4.1 Use Case Diagramm	7
4.2. Aktoren & Stakeholders	7
4.3. Use Case U1: Profil erstellen	8
4.4 Use Case U2: Daten eingeben	9
4.5 Use Case U3: Challenge durchführen	10
4.6 Use Case U4: Trainingsverlauf anzeigen.....	11

1. Dokumentinformationen

<i>Datum</i>	<i>Version</i>	<i>Änderung</i>	<i>Autor</i>
29.03.22	1.0	Erstellung	Schill, Krüger, Strauß
01.04.22	1.1	Anpassung nach Besprechungstermin	Schill

2. Allgemeine Beschreibung (Overall Description)

2.1. Produkt Perspektive (Product Perspective)

Unser Produkt ist eine interaktive Fitness Software, die den Anwender zu mehr Bewegung motiviert und beim Erreichen sportlicher Ziele unterstützt. Ein Profil wird mit den eingegebenen Parametern vom Nutzer erstellt.

Unser Produkt erleichtert die schriftliche Dokumentation von Trainings- und Esseinheiten und gibt dem Nutzer einen einheitlichen und allgemeinen Überblick darüber.

Des Weiteren werden dem Nutzer Challenges und Übungen individuell vorgeschlagen, um ihn zu motivieren sein Ziel leichter und mit mehr Freude zu erreichen.

2.2. Produkt Funktion (Product Functions)

- Nutzerprofil anlegen
- Ein Profil umfasst Empfehlungen für Wasser- und Kalorienzufuhr und Trainingsübungen
- Fortlaufende Erfassung von Kalorien- und Wasserzufuhr, sowie Trainingseinheiten
- Wöchentliche graphische Darstellung der Eingabedaten
- Einsehen der Trainingshistorie
- Challenge-Funktion mit Abzeichen
- Übungserklärungen
- Erinnerungsfunktion bei fehlender Dateneingabe und bei nicht Erreichen der täglichen Empfehlung

2.3. Benutzer Charakteristik (User Characteristics)

Die App richtet sich an alle, die das Ziel verfolgen fitter zu werden bzw. sich gesünder zu Ernähren.

2.4. Einschränkungen (Constraints)

- keine mobile Software
- nur auf einem Windows 10 Rechner ausführbar
- keine explizite Datenbank für Mahlzeiten
 - Kalorien der Mahlzeiten müssen selbst bestimmt werden
- Dient nur zur Orientierung und ersetzt keine medizinische Konsultation/Rat

2.5. Abhängigkeiten (Dependenencies)

- Betriebssystem
- Datenbank

2.6. Use Case Überblick (Use Case Survey)

- Profil erstellen
- Daten eingeben
- Challenge durchführen
- Trainingsverlauf anzeigen

3. Spezifische Anforderungen (Specific Requirements)

3.1. Funktionale Anforderungen (Functionality)

Die funktionalen Anforderungen wurden am Anfang des Projekts angelegt. Die Funktionen der Konkurrenzprodukte wurden berücksichtigt.

3.1.1. Funktionale Anforderung ‚Profil anlegen‘

Der Nutzer gibt einen Benutzernamen und sein Passwort ein. Zusätzlich gibt er sein Geschlecht, Alter, Größe, Gewicht und Bewegungsgruppe an. Anschließend wählt er sein gewünschtes Ziel aus.

3.1.2. Funktionale Anforderung „Daten eingeben“

Der Nutzer gibt fortlaufend seine eingenommenen Kalorien, Wasser und seine durchgeführten Trainingseinheiten ein.

3.1.3. Funktionale Anforderung „Challenge durchführen“

Der Nutzer kann eine ihm individuell vorgeschlagene Challenge annehmen und durchführen, die auf Zeit begrenzt ist.

3.1.4. Funktionale Anforderung „Trainingsverlauf anzeigen“

Der Nutzer kann sich seine täglichen Eingaben (Wasserzufuhr, Trainingseinheiten, Kaloreinanzahl) anzeigen lassen.

3.2. Bedienbarkeit (Usability)

Die Bedienung der Software soll intuitiv und ohne große Einarbeitungszeit möglich sein. Der Benutzer gibt über die Tastatur und Maus alle Daten ein.

3.3. Zuverlässigkeit (Reliability)

Die Datensätze der FitMeUp! - Software sollen zu einem frühestmöglichen Zeitpunkt in die Datenbank ausgelagert werden, um bei einem Systemabsturz der Software eine Wiederherstellung auf den zuletzt gespeicherten Stand zu ermöglichen.

3.4. Leistung (Performance)

Die Anzahl aller gespeicherten Datensätze sollen die Geschwindigkeit der Software nicht merklich beeinflussen.

3.5. Installation

Für unsere Anwendung ist Windows10 erforderlich.

3.6. Internationalisierung / Lokalisierung

Die Anwendung soll zunächst auf Deutsch angeboten werden.

3.7. Schnittstellen (Interfaces)

3.7.1. Benutzerschnittstelle (User Interface)

Die Schnittstelle zwischen unserem Programm und dem Benutzer ist GUI.

3.7.2. Datenbankschnittstelle (Database Interface)

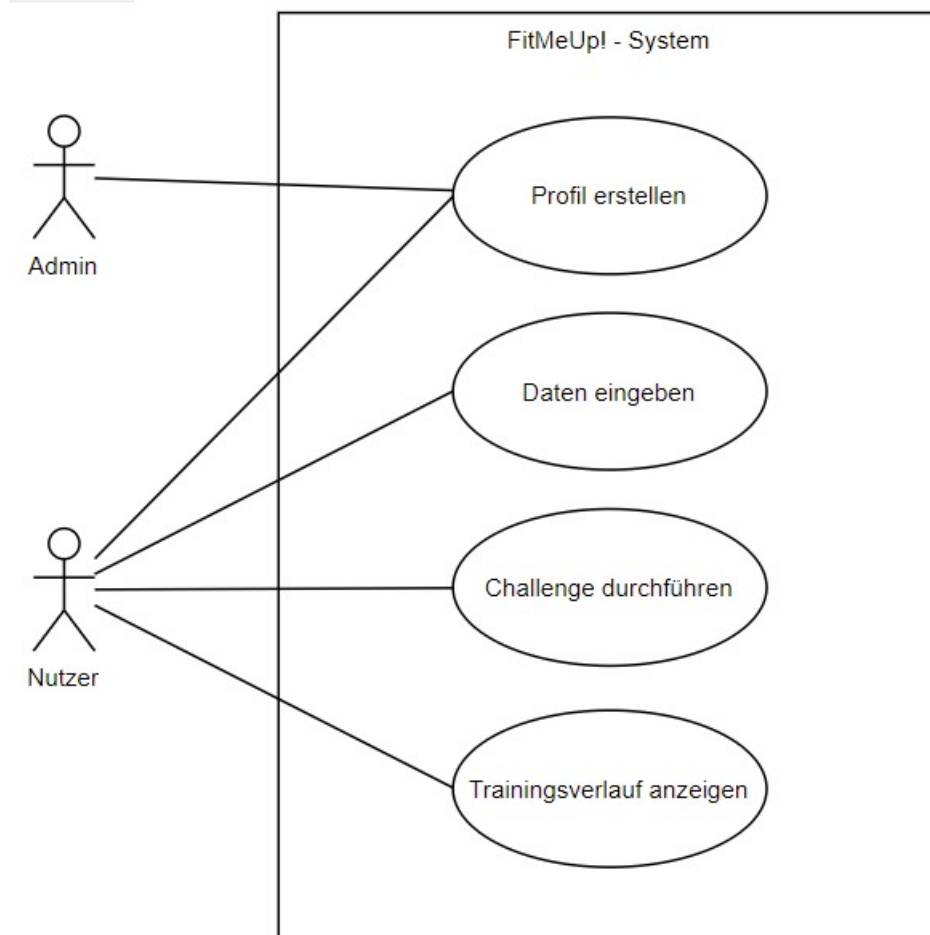
Die Datenbank wird über die virtuelle Maschine angesprochen.

3.8. Lizenzanforderungen (Licensing Requirement)

Es werden keine Lizenzen benötigt.

4. Use Cases

4.1 Use Case Diagramm



4.2. Aktoren & Stakeholders

- **Nutzer:** erstellt ein Profil, gibt seine Daten ein und wählt zwischen verschiedenen Zielen
- **Admin:** Verwaltet Nutzer und fügt neue hinzu
- **Entwickler:** Stellt eine sichere und möglichst intuitive Anwendung zur Verfügung

4.3. Use Case U1: Profil erstellen

Primary Actor: Nutzer

Stakeholders and Interests:

Nutzer: Will seine benutzerdefinierten Daten eingeben, um ein - seinen Zielen entsprechendes - Profil zu erstellen.

Preconditions:

Es existiert eine Verbindung zwischen Client und Server.

Postconditions:

Daten des Nutzers wurden erfasst und ein Profil wurde erfolgreich erstellt mit dem er sich jederzeit anmelden kann.

Main Success Scenario:

1. Der Nutzer klickt auf „Profil erstellen“
2. Das System zeigt die auszufüllenden Felder an.
3. Der Nutzer gibt sein Benutzername, Passwort, Geschlecht, Alter, Größe, Gewicht und Bewegungsgruppe ein.
4. Das System prüft die eingegebenen Daten.
5. Der Nutzer bestätigt seine Eingabe.
6. Das System speichert die Daten, legt das Profil an und zeigt dem Nutzer eine Registrierungsbestätigung an.

Extensions:

- 3a. Es werden nicht alle Felder ausgefüllt.
 1. Der Nutzer wird aufgefordert alle Felder auszufüllen.
- 4a. Der Benutzername ist bereits im System hinterlegt.
 1. Das System übermittelt eine Fehlermeldung und fordert den Nutzer auf einen neuen Benutzernamen auszuwählen.
- 4b. Der Benutzername ist unzulässig.
 1. Das System fordert den Nutzer auf einen anderen Benutzernamen zu wählen.
- 4c. Festgelegte Passwortanforderungen wurden nicht erfüllt.
 1. Das System übermittelt eine Fehlermeldung und fordert den Nutzer auf ein neues Passwort zu wählen.
- 5a. Der Nutzer bricht Anmeldevorgang ab.
 1. Das System fordert den Nutzer zum Bestätigen des Abbruchs auf.
 2. Der Nutzer bestätigt den Abbruch.
 3. Das System zeigt den Startbildschirm an.

Special Requirements:

- Benutzername Einschränkungen: mind. 2 Zeichen, max. 20 Zeichen, keine anzüglichen Benutzernamen
- Passwort Einschränkungen: Mind. 10 Zeichen, Groß- und Kleinbuchstaben, mind. ein Sonderzeichen

Frequency of Occurrence: oft

Technology and Data Variations List: Verbindung zur Datenbank

Open Issues: Aussortierung von nicht aktiven Nutzern

4.4 Use Case U2: Daten eingeben

Primary Actor: Nutzer

Stakeholders and Interests:

Nutzer: Will seine Daten eingeben, damit er diese immer im Überblick hat.

Entwickler: Will die Daten des Nutzers, um dem Nutzer Empfehlungen vorzuschlagen.

Preconditions:

Es existiert eine Verbindung zwischen Client und Server.

Der Nutzer muss angemeldet sein.

Postconditions:

Das System hat die Daten des Nutzers gespeichert.

Main Success Scenario:

1. Der Nutzer gibt jeden Tag in das vorhergesehene Feld die Daten ein.
2. Der Nutzer bestätigt seine Eingabe.
3. Das System speichert die Eingabe in der Datenbank.

Extensions:

2a. Der Nutzer bestätigt die Eingabe, ohne das Feld auszufüllen.

1. Das System zeigt eine Fehlermeldung an und fordert den Nutzer auf einen Wert einzugeben.

2b. Der Nutzer bricht den Vorgang ab.

1. Das System zeigt wieder das Hauptmenü an.

3a. Der Nutzer hat an diesem Tag bereits einen Wert gespeichert.

1. Der eingegebene Wert wird auf den bestehenden Wert darauf addiert.

Special Requirements: /

Frequency of Occurrence: täglich

Technology and Data Variations List: Verbindung zur Datenbank und zum Server

Open Issues: /

4.5 Use Case U3: Challenge durchführen

Primary Actor: Nutzer

Stakeholders and Interests:

Nutzer: Will anhand von Herausforderungen das Erreichen seines Ziels unterstützen.

Preconditions:

Es existiert eine Verbindung zwischen Client und Server.
Der Nutzer muss angemeldet sein.

Postconditions:

Der Nutzer hat die Challenge erfolgreich durchgeführt.

Main Success Scenario:

1. Das System zeigt anhand der Nutzerdaten Vorschläge für Challenges.
2. Der Nutzer startet eine Challenge.
3. Das System fordert den Nutzer auf seine täglichen Fortschritte einzugeben.
4. Der Nutzer bestätigt das Erreichen des täglichen Ziels.
5. Das System speichert die Eingabe.

Extensions:

- 2a. Der Nutzer beendet frühzeitig die Challenge.
1. Das System beendet die Challenge.
- 4a. Der Nutzer bestätigt das Erreichen des täglichen Ziels nicht.
1. Das System speichert die Challenge an diesem Tag als nicht erreicht an.

Special Requirements: /

Frequency of Occurrence: häufig

Technology and Data Variations List: Verbindung zur Datenbank und zum Server

Open Issues: /

4.6 Use Case U4: Trainingsverlauf anzeigen

Primary Actor: Nutzer

Stakeholder and Interests:

Nutzer: Will eine Übersicht über sein ganzen Trainingsverlauf einsehen.

Preconditions:

Es existiert eine Verbindung zwischen Client und Server.
Der Nutzer muss angemeldet sein.

Postconditions:

Das System zeigt dem Nutzer den Trainingsverlauf an.

Main Success Scenario:

1. Der Nutzer wählt „Trainingsverlauf ansehen“.
2. Das System holt sich die benötigten Daten und zeigt diese an.

Extensions:

Special Requirements: /

Frequency of Occurrence: häufig

Technology and Data Variation List: Verbindung zur Datenbank und zum Server

Open Issues: /

