Università degli studi di Roma Tor Vergata

Facoltà di ingegneria

Ingegneria del software e progettazione web

Fitness Helper

Professore:
Davide Falessi

Professore:
Guglielmo De Angelis

Samuele Costantinopoli
matricola: 0292589

Leonardo Pompili
matricola: 0293305

---

anno accademico 2022/2023

Project

1. Introduzione

1.1. Aim of the document

The Fitness Helper application guides the user through the 360-degree world of fitness. You can create your own personalized workout plan or buy them from personal trainers. In fact, the application provides the possibility of logging in with a "personal trainer" account to sell your services. Furthermore, the user can view a list of the gyms closest to him and track his progress via a training diary. Finally there is the possibility of knowing your TDEE(total daily energy expenditure) based on your objectives.

1.2. Overview of the defined system

The application provides the following features:

- Registration and login to the system: The user will be able to create an account or access an existing account. You can decide whether to create a normal account or an account dedicated to personal trainers.
- Create a workout and save it in the dedicated section. If you are a personal trainer user, in addition to saving it in your area, you can also sell it in the Buy workout plan area.
- Find a nearby gym.
- Find the personal trainer that is best for you.
- Calculate your TDEE(total daily energy expenditure), and show the user the calories he needs and the partition of macronutrients, with the possibility of customizing the plan.
- Buy a workout, a not logged-in user will be able to find out about the workout and have a preview of it. Only once you log in will you be able to purchase the workout through different payment methods.
- Access your reserved area where you can find all the information on purchases made

The System will connect to a database to store related information to your account and your tdee information. Operations can be carried out immediately thanks above all to the easy-to-use graphical interface that accompanies those who use it from starting to closing the application.

1.3. HW e SW requirements

- ➔ Operation System:
- ➔ Windows: 7 or higher
- ➔ Linux: Ubuntu 19 or higher
- ➔ Mac: OS Sierra or higher
- ➔ Processor: Minimum 1 GHz
- ➔ Ram: Minimum 2 GB
- ➔ Secondary Memory: Minimum 1 GB
- ➔ Network connection: LAN o Wi-Fi
- ➔ Camera/microphone: Not necessary

1.4. Related systems:

A related system is MyWorkoutPlan, an application that saves training plans from which we took inspiration to keep training plans saved.
Another related system is the website TDEE Calculator, which provides the possibility of calculating your TDEE based on a questionnaire.

2. Deliverable

- Codice:

  https://github.com/samueleCostantinopoli/ISPW

- SonarCloud link:

  https://sonarcloud.io/summary/new_code?id=samueleCostantinopoli_ISPW

- Video:

  Samuele Costantinopoli:
  linkVideoGit
  linkVideoYT

  Leonardo Pompili:
  linkVideoGit
  linkVideoYT

- Storyboard

  Samuele Costantinopoli: linkStoryboaed

  Leonardo Pompili: linkStoryboard

- BCE

  Samuele Costantinopoli: linkBce

  Leonardo Pompili: linkBce

- BCE refined

  Samuele Costantinopoli: linkBCErefined

  Leonardo Pompili: linkBCErefined

- MVC

  Samuele Costantinopoli: linkMVC

  Leonardo Pompili: linkMVC

- Activity diagram

  Samuele Costantinopoli: linkActivityDiagram

  Leonardo Pompili: linkActivityDiagram

- Sequence diagram

  Samuele Costantinopoli: linkSequenceDiagram

  Leonardo Pompili: linkSequenceDiagram

- State diagram

  Samuele Costantinopoli: linkStateDiagram

  Leonardo Pompili: linkStateDiagram

3. User Stories

Costantinopoli Samuele:

1. As a user, I want to buy a workout plan, so that I have a targeted training plan.
2. As a user, I want to save a workout, so that I can see my progress.
3. As a personal trainer user, I want to insert my contacts, so that I can sell online coaching services..

Pompili Leonardo:

1. As a user, I want to find a gym near me, so that I can start training.
2. As a personal trainer, I want to publish my workout plan, so that I can sell my service.
3. As a user, I want to write a *training diary, so that I can keep track of my progress.

Dictionary:

- Training diary: notes where the user can write.

4. Functional Requirements

Costantinopoli Samuele:

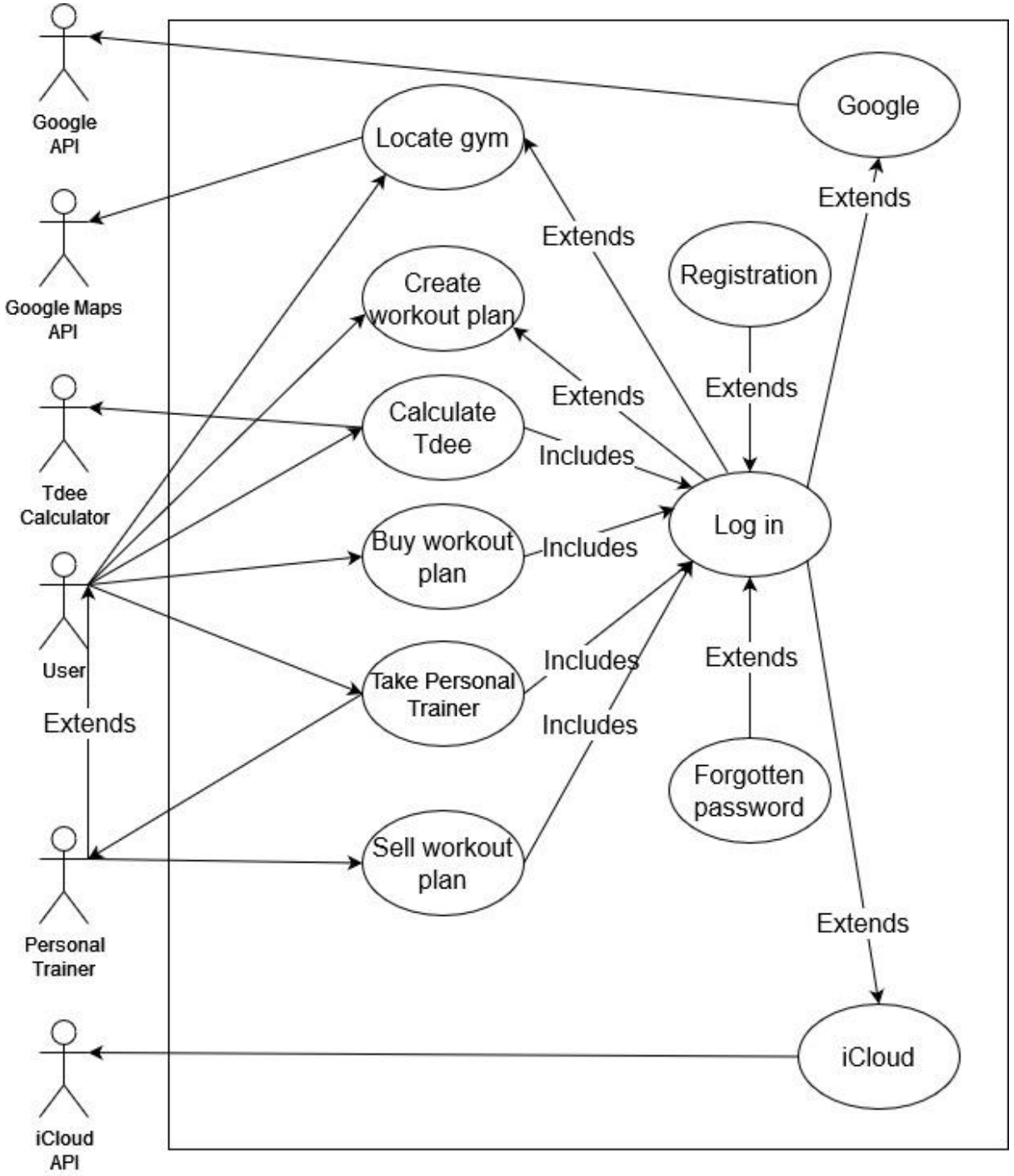
1. The system calculates the TDEE (Total Daily Energy Expenditure).
2. The system shall insert the personal trainer’s workout plan to sell.
3. 3. The system shall save the user’s annotations to the training diary.

Pompili Leonardo:

1. The system shall show on a map the list of gyms near the user.
2. The system shall notify the user of the confirmation of the purchase of a workout plan.
3. The system shows on the main page the list of workouts of the week.

5. Use Cases

Use Cases Diagram:



Use Cases internal steps:

Costantinopoli Samuele:

Name: Buy workout plan.

1. The user requests to buy a workout plan.
2. The system checks that the user is logged in.
3. The system shows the form for the selected payment methods.
4. The user chose payment methods.
5. The system verifies the correctness of the data entered.
6. The user confirms the purchase.
7. The system notifies the user of the payment made.
8. The system adds a workout plan on the main page.

Extensions:
2a. *The user is not logged in:* The system redirects the user to the login page.

4a. *The server of the platform chosen for the payment does not respond*: The system notifies the user and tries again to contact the server.

5a. *The data entered by the user are not valid:* The system prompts the user to check the data entered.

5b. *The user does not confirm the purchase:* The system notifies the user and asks if it wants to try again.

6a. *The payment was rejected:* The system notifies the user of the payment not made.

Pompili Leonardo:
Name: Calculate Tdee

1. Login.
2. The user requests to calculate the tdee.
3. The system shows the start page.
4. The user starts a questionnaire.
5. The system prepares a questionnaire.
6. The user answers questions.

7. The user confirms the answers given.
8. The system elaborates the tdee.
9. The system shows the result.
10. The user selects a carbohydrate amount.
11. The user saves the tdee calculated.
12. The system shows the list of the user’s tdee.

Extensions:

6a. The user did not fill all the fields of the questionnaire. The system warns the user to fill in all fields.

9a. The user has not selected the carbohydrate amount. The system warns the user to select one.

6. Features not implemented

The system has several dummy features:

1. The user cannot search for a nearby gym
2. The user cannot search for a personal trainer
3. The user cannot make the purchase.

Features present in the diagrams but not implemented:

1. Login via iCloud and Google is not present in the system (Use case diagram)
2. The user will not use the Google Maps API (Use case diagram)