Project 1: Object Oriented Programming-Part 1

Name: Zehan Wang

Date: 2025/2/11

Business Requirements

1.UserManagement: Users must register with a valid email address or phone number, and the registration process should include a verification link to ensure the authenticity of the information.

2. Personalized Plan Generation：Based on users' goals (weight loss, muscle gain, fitness maintenance), body metrics (height, weight, body fat percentage), and exercise & diet history, generate customized daily/weekly fitness and diet plans.

3. Record and Tracking：Allow users to input details of their workouts, including exercise types (weightlifting, running, yoga etc.), duration, intensity, and number of sets/reps. For diet, users can log the food they consume, with the app automatically calculating calories, macronutrients (protein, carbohydrates, fat) and micronutrients.

4. Progress Monitoring and Analysis：Visualize users' progress over time through graphs and charts for key metrics like weight, body fat percentage, muscle mass, and fitness level. Provide insights and suggestions for improvement based on the analysis of logged data.

5. Community Interaction：Build a community where users can share their achievements, challenges, workout tips, and diet recipes. Enable users to follow others, comment on posts, and form groups for mutual support and motivation.

6. Reminder and Notification：Set reminders for workouts, meal times, and water intake. Send push notifications for new community messages, plan updates, and motivational quotes.

Nouns

User Management: user, valid email address, phone number, registration process, verification link, registration information

Personalized Plan Generation: users' goals (weight loss, muscle gain, fitness maintenance), body metrics (height, weight, body fat percentage), exercise history, diet history, customized daily fitness plans, customized weekly fitness plans, customized daily diet plans, customized weekly diet plans

Record and Tracking: users, workout details, exercise types (weightlifting, running, yoga etc.), duration, intensity, number of sets/reps, food, calories, macronutrients (protein, carbohydrates, fat), micronutrients

Progress Monitoring and Analysis: users' progress, time, key metrics (weight, body fat percentage, muscle mass,

fitness level), graphs, charts, insights, suggestions for improvement, logged data
Community Interaction: community, users, achievements, challenges, workout tips, diet recipes, posts, others, comments, groups, mutual support, motivation

Reminder and Notification: reminders, workouts, meal times, water intake, push notifications, new community messages, plan updates, motivational quotes

Verbs

User Management: register, include, ensure
Personalized Plan Generation: based on, generate
Record and Tracking: allow, input, log, calculate
Progress Monitoring and Analysis: visualize, provide, analyze
Community Interaction: build, share, enable, follow, comment, form
Reminder and Notification: set, send

Target audience

-People aiming to lose weight: Such as busy office workers who have limited time but want to achieve their weight - loss goals by managing their diet and exercise.

-Fitness enthusiasts looking to gain muscle: This group has a good understanding of fitness and requires professional strength - training plans and nutritional planning.

-Beginner fitness enthusiasts lacking motivation: They need community support, easy - to - understand guidance, and regular encouragement to maintain their motivation and consistency in fitness.

-People focusing on overall health: These individuals hope to maintain a healthy lifestyle by recording and monitoring their daily activities.

Rules

-Data - related Rules

Accuracy Rules: Users are required to input accurate exercise and diet record data. For example, training duration and food intake must be filled in truthfully to ensure the reliability of subsequent progress monitoring and analysis.

Completeness Rules: When recording exercises and diets, users should fill in all necessary information completely, such as exercise type, intensity, number of sets/reps, as well as food variety and portion.

-Privacy and Security Rules

Data Protection Rules: Strictly comply with relevant data - protection regulations, encrypt and store users' personal information (such as registration information, health data, diet records, etc.) to prevent data leakage.

Access Permission Rules: Clearly define that only authorized functional modules can access user data. For example, only the personalized plan - generation module can generate plans based on users' body metrics and goals, and other unauthorized modules are not allowed to obtain relevant sensitive information.

-Community Rules

Code - of - Conduct Rules: It is prohibited for users to post offensive, insulting, false or spam information in the community. Violators may be warned, have their rights restricted or even have their accounts blocked.

Intellectual Property Rules: The content shared by users in the community (such as fitness tips, diet recipes, etc.) should ensure that it does not infringe on the intellectual property rights of others. If there is any infringement, the user shall bear corresponding responsibilities.

-Plan and Reminder Rules

Plan - Generation Rules: Personalized plan generation should be based on scientific and reasonable algorithms and professional knowledge to ensure that suitable fitness and diet plans are generated for users with different goals and physical conditions.

Reminder - Setting Rules: The exercise, meal and water - intake reminders set by users should conform to certain logic and limitations. For example, reminder times should not be too frequent to avoid disturbing users, and at the same time, the timeliness and accuracy of reminders should be ensured.

Challenge Questions

-Personalization and Precision: How can we ensure that the personalized fitness and diet plans generated for each user are highly tailored to their unique physical conditions (such as potential health issues, differences in physical functions), fitness goals (such as different stage goals for muscle gain, fat loss, and body shaping), time arrangements (such as the varying time availability of busy office workers and freelancers), and personal preferences (such as preferences for specific sports and dietary taboos) to achieve the best fitness results?

-Data Management and Analysis: With the increase in the number of users and the passage of usage time, a large amount of training records, diet data, and personal information will be accumulated. How can we efficiently store, manage, and integrate this diverse data? At the same time, how can we use advanced analysis techniques to extract valuable insights from the massive data and provide users with accurate, timely, and easy-to-understand progress feedback and improvement suggestions?

-User Retention and Activity: In the highly competitive fitness app market, how can we continuously attract users and maintain their activity and loyalty? Especially for beginners who lack the motivation to persevere in the long term, as well as users who encounter plateaus and feelings of frustration during their fitness journey, how can we stimulate their enthusiasm through the app's functions and community interactions, so that they can continue to use the app and

adhere to their fitness plans?

-Technical Implementation and Performance Optimization: Implementing various functions of the app (such as personalized plan generation algorithms, real-time data recording and synchronization, and the smoothness of community interactions) requires strong technical support. How can we ensure the richness of functions while optimizing the app's performance, ensuring that it can run quickly and stably on different devices (such as mobile phones and tablets) and in various network environments, reducing lag and loading times, and providing a good user experience?

-Community Building and Management: Building an active, positive, and healthy community is not an easy task. How can we attract users to participate in community interactions and encourage them to share real and effective fitness experiences, success stories, and challenges? At the same time, how can we effectively manage community content to prevent the appearance of inappropriate information, false propaganda, or misleading statements, and maintain a good community atmosphere?

-Regulatory Compliance and Privacy Protection: When dealing with users' personal health data, we must strictly comply with relevant laws and regulations (such as data protection laws and health information privacy laws). How can we establish a perfect privacy protection mechanism to ensure the security of user data? At the same time, in the design and operation process of the app, how can we fully meet regulatory requirements and avoid potential legal risks?

Summary of Classes, Attributes and Associations

1.User Class

Attributes:

username (string): Unique identifier for the user in the app.

password (string): For user authentication.

email (string): Contact information.

phoneNumber (string): Optional contact information.

age (integer): User's age for personalized plan generation.

gender (string): Male or female for body metric calculations.

fitnessGoal (string): Such as "weight loss", "muscle gain", "fitness maintenance".

bodyMetrics (object): Containing height (float), weight (float), bodyFatPercentage (float), etc.

timeAvailability (string): Describing the user's available time for workouts (e.g., "weekday evenings", "weekends only").

userPreferences (object): Preferences for exercise types, food likes/dislikes, etc.

2.TrainingRecord Class

Attributes:

recordID (integer): Unique identifier for the training record.

trainingDate (date): Date of the training session.

duration (float): Duration of the workout in hours or minutes.

exerciseType (string): Such as "weightlifting", "running", "yoga".

intensity (string): Light, moderate, intense.

sets (integer): Number of sets for strength training exercises.

reps (integer): Number of repetitions per set.

3.DietRecord Class

Attributes:

recordID (integer): Unique identifier for the diet record.

mealDate (date): Date of the meal.

mealType (string): Breakfast, lunch, dinner, snack.

foodItems (array of strings): List of foods consumed.

quantities (array of floats): Quantities of each food item.

calories (float): Total calories of the meal.

macronutrients (object): Containing protein (float), carbohydrates (float), fat (float).

micronutrients (object): Vitamins and minerals information.

4.PersonalizedPlan Class

Attributes:

planID (integer): Unique identifier for the plan.

planType (string): Fitness or diet plan.

fitnessPlanDetails (object): Workout schedule, exercise routines, etc.

dietPlanDetails (object): Meal plans, nutritional guidelines, etc.

startDate (date): Start date of the plan.

endDate (date): End date of the plan.

5.ProgressMonitor Class

Attributes: None (functions mainly as a service).

6.Community Class

Attributes:

communityID (integer): Unique identifier for the community.

communityName (string): Name of the community (e.g., "Weight Loss Support Group").

description (string): Description of the community's purpose.

7.CommunityPost Class

Attributes:

postID (integer): Unique identifier for the post.

author (User object reference): The user who created the post.

postContent (string): Text, images, or videos related to the post.

postDate (date): Date when the post was created.

likes (integer): Number of likes received.

comments (array of Comment objects): List of comments on the post.

8.ReminderNotification Class

Attributes:

notificationID (integer): Unique identifier for the notification.

reminderType (string): Workout, meal, water intake, etc.

reminderContent (string): Details of the reminder.

notificationTime (date and time): When the reminder will be sent.

isRead (boolean): Indicates whether the user has read the notification.

9.UserManagement Class

Attributes: None (functions as a service).

Associations

User Class

Association with TrainingRecord Class: One user (User) can have multiple training records (TrainingRecord), that is, there is a one-to-many relationship. For example, a user may conduct different trainings every day or every week, and each training will generate a training record, all of which are associated with this user.

Association with DietRecord Class: Similar to training records, one user can have multiple diet records (DietRecord), also a one-to-many relationship. The user's different meals of the day, including snacks, etc., will be recorded and correspond to this user.

Association with PersonalizedPlan Class: A user usually has only one personalized plan (PersonalizedPlan) within a certain period of time, but this plan can be updated according to the user's situation, so it is a one-to-one relationship. The personalized plan is tailored for the user based on the user's body metrics, fitness goals, etc.

Association with CommunityPost Class: A user can create multiple posts (CommunityPost) in the community and can also interact with the posts of other users. Therefore, there is a many-to-many relationship between the user and the community posts. For example, a user can post their fitness achievements, ask questions, etc., and at the same time, can like, comment on the posts of other users.

Association with ReminderNotification Class: One user can receive multiple reminder notifications (ReminderNotification), such as workout reminders, meal reminders, etc., which is a one-to-many relationship.

TrainingRecord Class

Association with User Class: Each training record belongs to a specific user, so there is a many-to-one relationship between the training record and the user. That is, multiple training records can correspond to the same user.

DietRecord Class

Association with User Class: Each diet record corresponds to one user, which is a many-to-one relationship. Multiple diet records can belong to the same user, reflecting the user's dietary situation at different times.

PersonalizedPlan Class

Association with User Class: Each personalized plan is generated for a specific user, which is a one-to-one relationship. The content of the plan will be customized according to the user's physical condition, fitness goals, etc.

ProgressMonitor Class

Association with User Class: The progress monitoring class interacts with the user to obtain the relevant data (such as training records and diet records) of this user, so as to analyze and monitor the user's fitness progress. It is an interactive relationship.

Association with TrainingRecord Class: The progress monitoring class uses the user's training record data to calculate and analyze the user's progress in training, such as changes in training intensity, training frequency, etc. It is a relationship of data acquisition and processing.

Association with DietRecord Class: By obtaining the user's diet record data, the progress monitoring class can analyze the user's dietary intake situation, such as whether the intake of calories and nutrients meets the fitness goals. It is also a relationship of data acquisition and processing.

Community Class

Association with User Class: One community can have multiple users joining, and one user can join multiple communities. So there is a many-to-many relationship between the community and the user. Users communicate, share, etc. in the community.

Association with CommunityPost Class: One community can contain multiple community posts, and one post can only belong to one community, which is a one-to-many relationship. The community provides a platform for posts to be published and communicated.

CommunityPost Class

Association with Community Class: Each post belongs to a specific community, which is a many-to-one relationship. Different posts are available for users to browse and interact with in the community to which they belong. Association with User Class: Posts are created by users, and other users can operate on the posts (such as liking, commenting). So there is a many-to-many relationship between the user and the post.

ReminderNotification Class

Association with User Class: Reminder notifications are sent to specific users, and one user can receive multiple reminder notifications, so it is a one-to-many relationship.

UserManagement Class

Association with User Class: The user management class is mainly responsible for operations such as user registration, login, account update, and deletion. It has a management and managed relationship with the user class, and realizes various management functions of user accounts by operating on the user class.

User personas (at least 2) and user stories (at least 3 per persona)

User Persona 1: The Busy Office Worker - Sarah

Age: 30

Gender: Female

Occupation: Marketing Specialist at a large corporation

Fitness Goal: Lose weight and maintain a healthy lifestyle despite a busy work schedule

User Stories:

1.As a busy office worker with a short lunch break and a busy schedule, I want to be able to log my meals quickly. I wanted an easy-to-use interface where I could select common office lunches (e.g. sandwiches, salads) and enter quantities in a few taps, making it easy to track calorie intake for the day.

2. Due to long working hours, Sarah has limited time to exercise and wants to plan short and effective workouts. I want the app to recommend 20 to 30 minute workouts that I can do at the office or at home, like quick self-weight training or simple stretching, with clear instructions and videos for each move to help me do it right.

3. I would like the app to send me regular water reminders, with customizable intervals to fit my work schedule, and set a daily water intake goal.

User Persona 2: The Fitness Enthusiast - Mark

Age: 28

Gender: Male

Occupation: Personal Trainer at a local gym

Fitness Goal: Build more muscle mass and improve overall strength

User Stories:

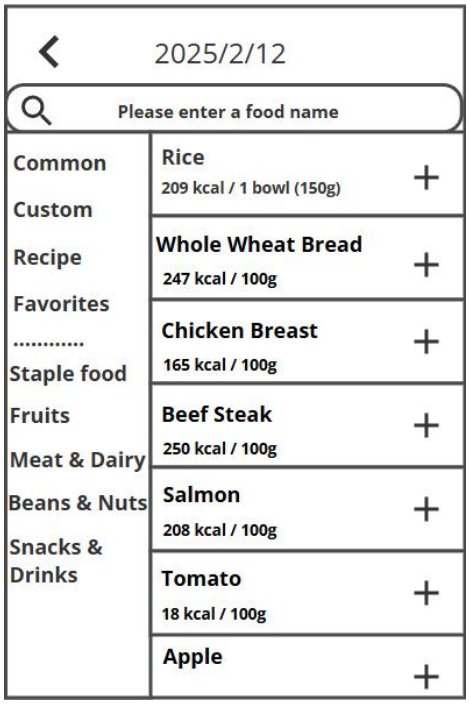
1. As a fitness enthusiast and personal trainer like Mark, I hope the app can generate highly personalized strength training plans. It should take into account my current strength level, training experience, and any injuries or limitations. The plan should include exercise progressions, appropriate rep ranges, and rest periods to help me build muscle mass.

2. As a fitness enthusiast and personal trainer Mark, who adheres to a high-protein diet for muscle growth, I hope the app can calculate my daily protein intake according to the diet info I upload.

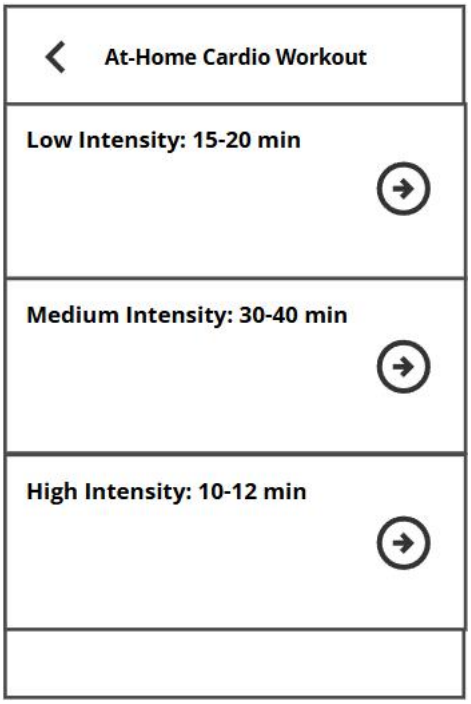
3.Mark pays attention to the latest fitness trends and techniques. I hope the community section of the app can enable me to communicate with other fitness enthusiasts and professionals. I can share my experiences, learn new knowledge, follow specific users and groups, and receive notifications when there is new content.

Interface low level mockups

Interface for recording calorie intake

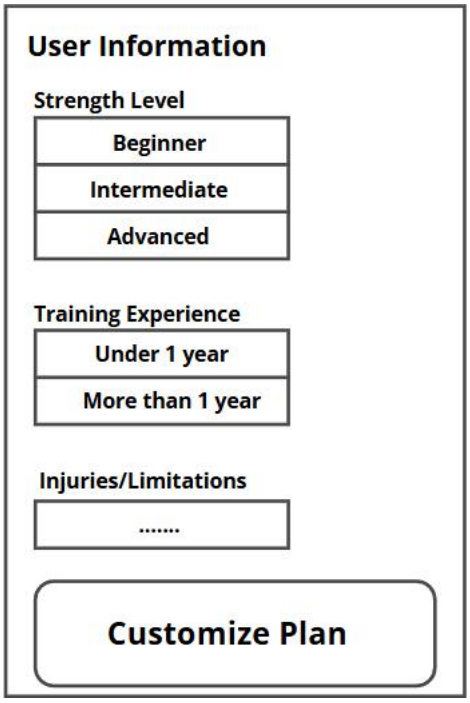


Home-based aerobic exercise selection page

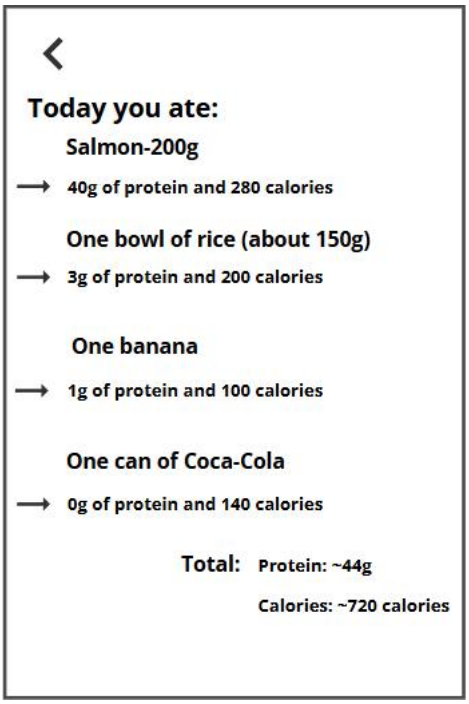


Add reminder page

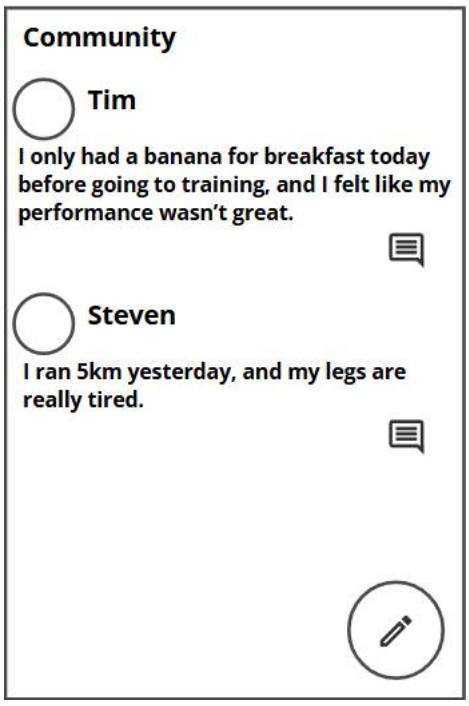
The user fills in personal information (Strength Level, Training Experience, Injuries/Limitations) and generates a suitable plan with one click.



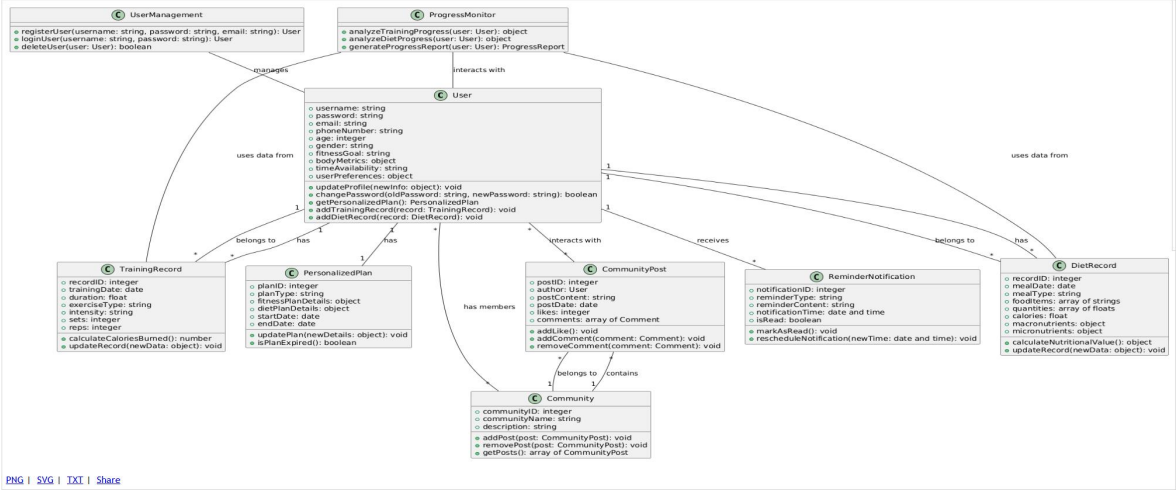
Daily Dietary Nutrition Tracking page



Community interface



UML Class Diagram



Link:

ZLTBRnGn3BvNwd-8UbgWFN2T4z0YL8bMgwfmpquyiu4a7f9CsmNntt5cvvK7i3rqk_Pdn_xyIFhEUMvzhzNvMQsuS-oB0yj-dPyn-dd3Ulfck8QAEM-bQITvHz1dj2ALW-PIPU0z6hZlzGvigk8jkPR6GmjsahLWH8vif3VWt2VacVyTYiCTaAXs5SFTDwZzfF9Im_idYedlf9B-a9g6_BOM6h1WQiZC-qvm3rkBZLImCV1yQngSGAygzeHIJDXwpqqBsv6O3IgnJKYwPEGX5PAN7Q82RfRy_HQiGyEL_0bYgxZP42gLpHGAyMYvDEJh0Ufmj1q-AXQBat39x4Q2JqmMqGp_VNuszKRiSEcIe_NjJLPAFnhS48iL2rnE6j5RxYMQYZKA-Kmul82jfOF7GvVrNF1jNA683hpBZhRGvSAQgxfN5COrLsWbk0-zDI02lMRepxZm8pbKDKg0PtLVCRFmTW8h6hZA6GdIKjeDehZre2aRRYq_C6n6X9iWFtfklFGIrfY1LxVAV4WuuLlpsgBfoHqOdxM_bl_IpcJU1qGe9bTVkUeXq5cQezFfJ7jz8RMZRmLAWxX8td5h11ytu6aFP5a8gjm_rCDwp8j5AoeNZjiYp6l8CNQOZ9ySOlhuqabxxBzv2QnfiDXQ2lyEZVIuMikSk3dyX6aU9zmchBDgME5P2KQJqAYdm6a9WptklW7x01rQ7rl4kYZoQzIwfw0FIypr92gKRzRT5-uQ0QwsiZjkY-Jw46ABpcyw-bKjXmPPmhS5ZKzm8ZXiO9AwK9TvdYA3Sh91isfL-bR8bVT-ZoE9Q-WrXltciykJL7chAVaT2kkD0ZbEQXGrYLQ4VIRJJNuPZC3DwAEQB8iy_XUyOkS1j0mN-ZrwsSXwsFeBIMObBP1bH-FIR4-wln2tzlmezKWWpO4ONWNB83u05zabhBdz_dvGfLnHnzTx4Bs2TKPXuWkdb0W9XR-ZSMmXKBQ-BrhfFDYWtwGFhylioNKPFxPUnHsbi9NcH4-9gG057cRRUTpZ3JMy4o_UNh2hAtRn-Y9z6rHiprsEMjsJ9GHzPAi_mh7Nx8WS_eW7xzWybjVUiMVfz_dfnLQimaKDyWdea2I9vJZoCJWZ5a0XNRtCu_dPAfl_GTDLJiWHbGPTG6RSKbrSQ3JE6ekwQ1CndUANzPmmEiP4ZDDx93o8N0HUg4ofhsa8ASWrVD14l2oUOdAIYLYuqOEGdBwZ2vV-IVa3 (2004×989)