# GRADUATE CERTIFICATE
# ARTIFICIAL INTELLIGENT SYSTEM (AIS)
# PROJECT REPORT

Project name: fItneSS us —— an intelligent fitness app
Group Number: Practice module group 4
Group Members' Names:

Xia Yichao – A0297377H
Huang Tianle- A0296702Y
Huang Chenyu - A0295813W
Zhu Yihang - A0297220H

Date of the Presentation: 2024

# Catalog

# 1. Project Background

## 1.1 Introduction

The fItneSS us platform is an innovative fitness app designed to provide personalized gym and workout site recommendations based on users' location, available time, and fitness goals. It offers tailored workout plans, real-time exercise suggestions, and live form correction during workouts using computer vision technology.

This project seeks to combine location-based services, AI-powered fitness recommendations, and real-time feedback to revolutionize the way individuals approach their fitness routines.

## 1.2 Background

### 1.2.1 Abstract

In today's fast-paced world, people often struggle to find time to work out and maintain a balanced fitness routine. Furthermore, accessing nearby workout facilities, creating effective workout plans, and ensuring proper form during exercise can be overwhelming.

Many existing platforms either focus on location-based services or provide generic workout plans without addressing individual needs. fItneSS us aims to fill this gap by integrating location-based services with AI-powered workout recommendations and live movement supervision to offer a truly personalized fitness experience.

### 1.2.2 Market analysis

The global fitness industry is projected to grow to over $96 billion by 2024. With the rise of digital fitness solutions due to the COVID-19 pandemic, users are increasingly seeking out home-based fitness platforms with a personalized touch. However, existing platforms like Peloton, Mirror, and MyFitnessPal lack the combination of location-based gym recommendations, dynamic workout planning, and real-time form feedback. fItneSS us addresses this need by integrating multiple functionalities into one seamless app, tapping into both the gym-goer and home-fitness markets.

### Key Market Drivers

- Increased health awareness post-pandemic.

- Growth of the at-home fitness industry.
- Demand for personalized fitness experiences.

## PEST Analysis

### Political

**WHO policy**

Global Action Plan on Physical Activity 2018-2030:

- Create Active Societies: Promote and create opportunities for physical activity through community programs and policies.
- Create Active Environments: Develop urban and rural environments that support physical activity, such as parks and recreational facilities.
- Create Active Workplaces: Encourage workplaces to offer opportunities for physical activity and promote a culture of health and well-being.
- Create Active People: Support individuals in becoming more active by providing information, resources, and support.

(https://iris.who.int/bitstream/handle/10665/272722/9789241514187-eng.pdf)

**Policies in different countries**

**USA**

- Department of Health and Human Services (HHS): The HHS publishes the Physical Activity Guidelines for Americans, which recommend that adults should engage in at least 150 minutes of moderate-intensity or 75 minutes of vigorous-intensity aerobic activity each week, along with muscle-strengthening activities. For children and adolescents, the guidelines recommend at least 60 minutes of physical activity per day.
- Healthy People 2030: This initiative sets data-driven national objectives to improve health and well-being over the next decade. It includes specific goals related to physical activity, such as increasing the proportion of adults who meet the recommended levels of physical activity and enhancing community environments that support active living.
- Community and School Programs

**China**

- National Fitness Plan (2016-2020): This plan is a comprehensive strategy aimed at increasing physical activity levels among the Chinese population. It emphasizes the development of sports infrastructure, community sports programs, and integration of physical activity into daily life. The plan promotes the construction of fitness centres, sports parks, and public exercise facilities.

- China's Sports Law: This law includes provisions to encourage public participation in sports and fitness activities. It supports the development of sports programs at the community level and aims to make playing sports more accessible to all citizens.
- Healthy China 2030: This initiative outlines goals to improve the overall health of the Chinese population, including increasing physical activity levels. It includes measures to integrate physical activity into daily routines, promote exercise among youth, and develop sports and fitness facilities.

**Policy in Singapore**

Singapore's political environment plays a significant role in shaping the fitness and healthcare sectors.

Government Health Initiatives:

- Health Promotion Board (HPB): The HPB is a government agency under the Ministry of Health (MOH) responsible for promoting healthy living. It runs various campaigns and programs to encourage physical activity, healthy eating, and preventive care. For instance, the HPB's "Healthier SG" initiative focuses on improving the overall health of Singaporeans by encouraging regular health checks, healthy lifestyles, and preventive measures.
- ActiveSG: Launched by Sport Singapore, ActiveSG aims to promote sports and fitness activities across all age groups. The initiative provides affordable access to sports facilities, offers community sports programs, and supports fitness events. ActiveSG also offers a fitness membership program that provides discounts and benefits for various sports and fitness activities.

Government strategic plans

- Singapore's Healthier Singapore Strategy: This strategy focuses on improving the nation's health by promoting healthy lifestyles and preventive care. It includes initiatives to enhance physical activity, reduce chronic diseases, and improve mental well-being. The strategy aligns with the broader goals of Singapore's 2030 Healthcare Masterplan, which aims to provide affordable and quality healthcare services while promoting a healthier nation. (https://www.healthiersg.gov.sg/about/what-is-healthier-sg/)
- National Physical Activity Guidelines: These guidelines provide recommendations for physical activity levels across different age groups and

populations. They are used to inform public health campaigns and community programs designed to increase physical activity and reduce sedentary behavior. (https://ch-api.healthhub.sg/api/public/content/a0254274ebdd40ab95c7c630a59acc31?v=dab36f97)

Conscription System (Increased Fitness Demand)

- Physical Requirements: Due to the physical training demands during National Service, Singaporean males need to maintain good physical condition. Many individuals begin physical conditioning before enlistment to improve their fitness levels and meet the service requirements.
- Continued Fitness Habits: Even after completing National Service, many individuals continue to engage in regular exercise. They become accustomed to physical activity and recognize its importance for maintaining health.

Political Stability

Singapore's political stability contributes to a favorable environment for implementing health and fitness policies. The government's long-term commitment to improving public health and supporting the fitness industry ensures consistent policy support and funding.

### Economic

Economic factors like rising disposable income and increased spending on health and fitness products make this a favorable time for the launch of fItneSS us. Many people are willing to spend money on gym memberships, fitness apps, and wellness programs, especially after the pandemic raised awareness of the importance of personal health.

**Economic Downturns**

During the COVID-19 pandemic, many people turned to free or low-cost fitness apps as gyms closed. According to a report by App Annie, global downloads of fitness apps increased by 50% in early 2020 compared to the previous year.

In cases of economic downturn, consumers may reduce spending on non-essential services, which could include fitness apps. However, if fItneSS us is competitively priced or offers value by eliminating the need for expensive personal trainers or gym memberships, it can retain a solid user base even during challenging economic periods.

For example, MyFitnessPal, a popular fitness app, saw increased user engagement during the pandemic as people sought cost-effective fitness solutions.

**Partnerships with Gyms and Fitness Centers**

Economic factors could influence partnerships with gyms and fitness centers, particularly in regions where the gym industry has suffered post-pandemic.

A survey by the International Health, Racquet & Sportsclub Association (IHRSA) reported that 27% of gyms worldwide closed permanently or temporarily during the pandemic.

Gyms might see partnering with fItneSS us as a way to bring customers back, increasing their own revenue while contributing to the platform’s growth.

## Social

### Health and Fitness Awareness

Social awareness of health, fitness, and wellness has grown significantly, particularly after the COVID-19 pandemic. According to a survey by McKinsey & Company, 79% of consumers in the U.S. have adopted healthier behaviours since the pandemic began. There is a strong societal trend toward adopting healthier lifestyles, exercising regularly, and focusing on mental well-being, making the timing for fItneSS us ideal. Consumers are looking for tools that provide personalized, accessible, and affordable solutions for maintaining fitness.

### Personalization and Convenience

Modern consumers are accustomed to personalized digital experiences, whether in e-commerce, entertainment, or fitness. fItneSS us can leverage this expectation by offering tailored workout routines, gym recommendations, and real-time feedback based on individual preferences and body status. Convenience will also be a key selling point, as users increasingly prefer platforms that fit seamlessly into their daily routines.

### Community and Gamification

Fitness platforms that include social or gamified elements have proven to be more engaging. Social factors like group motivation, leaderboards, challenges, and sharing progress with friends can drive user engagement. By integrating community features and challenges, fItneSS us can foster a sense of belonging and increase retention rates.

### Remote Work and Fitness Trends

The rise of remote work has changed fitness habits, with people now looking for flexible workout options that fit into their home or hybrid work life. A survey by the American College of Sports Medicine (ACSM) revealed that 43% of fitness consumers have increased their home workouts since the pandemic began. This has led to increased demand for apps that provide flexibility and allow users to work out at home, at the gym, or while traveling.

## Technological

### Advancements in AI, ML and DL

AI and machine learning technologies are essential to the success of fItneSS us, as they enable personalized recommendations and real-time feedback. The platform will rely on machine learning to analyze user data, track progress, and suggest exercises, while computer vision will ensure correct workout form. With ongoing improvements in these fields, the platform has the potential to offer even more sophisticated features over time.

### Wearable Technology Integration

The fitness market has seen a surge in the adoption of wearable devices like Fitbit, Apple Watch, and Garmin, which track health metrics such as heart rate, calories burned, and sleep quality. Integrating with these devices will enhance fItneSS us’s ability to provide data-driven recommendations and create a holistic fitness ecosystem. Keeping up with advancements in wearables and ensuring seamless integration will be crucial.

**Mobile and Cloud Technologies**

As fItneSS us is primarily a mobile application, it will benefit from the continuing advancements in smartphone capabilities (e.g., high-resolution cameras for form correction) and cloud computing technologies. Cloud-based platforms will allow the app to scale, provide real-time feedback, and store vast amounts of user data securely.

**Augmented Reality (AR) and Virtual Reality (VR)**

Emerging technologies like AR and VR offer future potential for fItneSS us to provide immersive workout experiences. Users could engage in virtual fitness classes or even receive real-time AR-based form corrections during exercises. While not immediately necessary, this could be an area of growth and innovation.

## STP Analysis

### Segmentation

**Age**

- Young Adults (18-30 years): Interested in fitness for lifestyle and social reasons. They might be tech-savvy and open to using apps for fitness.
- Adults (31-50 years): Focused on maintaining health and managing work-life balance. They might be looking for flexible solutions.
- Older Adults (50+ years): May require specialized fitness solutions that consider age-related physical limitations.

**Income Level**

- High-Income: Likely to invest in premium features or personalized services.
- Middle-Income: May prefer affordable or free features with high value.
- Low-Income: Interested in budget-friendly solutions or free options.

**Geographic**

- Urban: Higher demand for modern fitness solutions due to greater access to technology and fitness awareness.
- Rural: Might have limited access to gyms; therefore, remote, and home-based fitness solutions are more appealing.

**Behavioral**

- Fitness Enthusiasts: Regularly exercise and seek advanced fitness features.
- Casual Exercisers: Engage in occasional exercise and look for simple, easy-to-use solutions.
- Newcomers to Fitness: Interested in starting a fitness routine and need guided, beginner-friendly options.

**Psychographic**

- Health-Conscious Individuals: Focused on overall wellness, including physical fitness, mental health, and nutrition.
- Lifestyle-Oriented Individuals: Seek fitness solutions that integrate with their social and leisure activities.
- Goal-Oriented Individuals: Focused on specific fitness goals, such as weight loss, muscle gain, or improved endurance.

## Targeting

- Urban Health Enthusiasts: Young and middle-aged adults living in cities who are tech-savvy and prioritize health and fitness. They are likely to use a fitness app that offers convenience and personalization.
- Busy Professionals: Adults with demanding jobs who need flexible workout options that fit into their busy schedules. They will appreciate features like on-demand workouts and time-efficient exercises.
- Fitness Newcomers: Individuals who are new to fitness and need guidance, motivation, and easy-to-follow routines. They are likely to be drawn to beginner-friendly features and educational content.

## Positioning

Our market position is dynamic and evolving.
We primarily aim at personalization and convenience.

- Personalization: Offer tailored workout plans and fitness recommendations based on individual preferences and goals.
- Convenience: Provide flexible workout options that can be done at home, at the gym, or while traveling, integrating seamlessly into users' daily routines.

For the beginning period of the project, we provide our customers with:

- For Urban Health Enthusiasts: "fItneSS us offers cutting-edge, personalized fitness solutions that fit seamlessly into your busy urban lifestyle, helping you stay fit and motivated no matter where you are."
- For Busy Professionals: "Maximize your fitness with our flexible, on-demand workouts designed to fit into your hectic schedule. Achieve your health goals without compromising your professional commitments."
- For Fitness Newcomers: "Start your fitness journey with fItneSS us. Our easy-to-follow routines and motivational support are designed to guide you every step of the way, making fitness approachable and enjoyable. "As our project matures, we

will target our services at high-net-worth clients. We will also provide specialized and customized high-end services.

Finally, we will incorporate most market segments, including which rural or elderly market. We will also have specialized algorithms to help them achieve good health.

## Porter's 5 forces

### Threat of new entrants

The fitness app market has low barriers to entry, with many developers able to create and launch new apps. This increases competition and the threat of new entrants.But this industry could be data-orientated in the future, and users' fitness data and preferences are still unbelievably valuable and could be a barrier in the future in terms of algorithms.This means we should do more on differentiation by adding unique features and get advantages by using AI. We can also build brand loyalty in our customers.

### Bargaining power of suppliers

As mentioned before, the economic downturns could influence partnerships with gyms and fitness centers, particularly in regions where the gym industry has suffered post-pandemic. There is intense competition among the gyms which means their bargaining power is low.

### Bargaining power of buyers

Users have numerous fitness app options to choose from, giving them significant bargaining power. They can easily switch to a competitor if they find better value or features.

Users are price-sensitive, particularly if they are comparing subscription models or free vs. paid features.

These lead to high bargaining power from buyers.

### Threat of Substitute Products or Services

Substitutes include free fitness content on platforms like YouTube, home workout equipment, and traditional gyms. Users may choose these alternatives over fitness apps. Innovative technologies, such as virtual reality fitness experiences or advanced wearable devices, can serve as substitutes.

### Industry Rivalry

The fitness app market is highly competitive with established players like MyFitnessPal, Fitbit, and new entrants constantly emerging.

## Competitors

- Peloton (live and on-demand workouts).
- MyFitnessPal (calorie tracking, but no real-time feedback).
- Nike Training Club (personalized workouts but lacks gym recommendations and form supervision).

## SWOT Analysis

“fItneSS_us” has several strengths, including innovative features, a comprehensive solution, and strong community engagement. However, it faces challenges such as high development costs, market penetration difficulties, and dependence on data privacy.

### Strength

**Innovative Features**

- Personalization: Offers tailored workout routines and exercise recommendations based on individual preferences, body status, and goals.
- Form Supervision: Provides real-time feedback and corrections on exercise form, enhancing user experience and effectiveness.

**Comprehensive Solution**

- Integrated Platform: Combines gym recommendations, workout plans, and movement supervision in one app, offering a holistic fitness solution.
- Convenience: Allows users to work out anywhere—at home, in the gym, or while traveling—making fitness accessible and flexible.

**Weakness**

- Market Penetration: As a new entrant in a competitive market, fItneSS us may struggle with brand recognition and establishing trust compared to well-established competitors.
- Dependence on Data: Handling sensitive health data requires robust security measures and compliance with privacy regulations, which can be challenging and costly.

### Opportunity

- Growing fitness awareness & expanding market

**Technological advancements**

- Location-based Services
- Recommendation systems
- Computer vision

### Threat

**Intense Competition**

- Established Players: Competing with established fitness apps like MyFitnessPal, Fitbit, and new market entrants can be challenging.
- Rapid Innovation: Fast-paced technological advancements mean competitors are constantly introducing new features and improvements.

**Economic Downturns**

- Reduced Spending: Economic downturns may lead to reduced consumer spending on non-essential services, including fitness apps.
- Pricing Pressure: Pressure to offer competitive pricing or free features may impact profitability.

**Regulatory Challenges**

- Data Privacy Regulations: Compliance with stringent data privacy laws and regulations (e.g., GDPR, CCPA) can be complex and costly.
- Health and Safety Standards: Ensuring adherence to health and safety standards for fitness-related advice and supervision.

### 1.2.3 Technical background Review

This project aims to develop a personalized fitness plan generator that assists users in creating tailored exercise routines based on their physical characteristics and goals. By inputting personal data such as height, weight, and BMI, along with specific fitness objectives like fat loss or muscle gain, and current exercise conditions (e.g., no equipment or gym access), the system will generate a customized workout plan. The plan will detail exercise frequency, targeted muscle groups, specific exercises, sets, repetitions, and recommended weights. Additionally, the project intends to provide users with information on nearby gyms to enhance their fitness journey.

To achieve these objectives, the project will leverage the following technologies:

- Data Collection and Storage: A robust database will be designed to securely store user information, fitness goals, and exercise data. This database will serve as the foundation for generating personalized workout plans.
- Natural Language Processing (NLP): NLP techniques will be employed to interpret user inputs and preferences, allowing the system to understand and process natural language queries effectively. This will enhance user interaction and ensure accurate customization of fitness plans.
- Artificial Intelligence and Machine Learning: AI algorithms analyze user data and goals to generate optimal workout routines. These algorithms will consider a range of factors, such as fitness level and available equipment, to tailor exercises appropriately.
- Geolocation Services: Integration with geolocation APIs will enable the system to provide users with information on nearby gyms, including location, hours, and available equipment, enhancing the user's fitness experience.

By integrating these technologies, the project aims to deliver a comprehensive and user-friendly platform that empowers individuals to create and follow personalized fitness plans effectively.

## 1.3 Project Scope

### 1.3.1 Main Goal

To develop a comprehensive fitness platform offering personalized workout recommendations, gym suggestions, and real-time form feedback.

### 1.3.2 Core functions

#### Gym and Workout Site Recommendation (location-based services)

- Location-based Suggestions: Integrate Google Maps or other geo-location APIs to suggest nearby gyms, fitness parks, or workout classes.
- Time Availability: Allow users to input their available time slots and based on gym operating hours, suggest the best matches.
- Additional Filters: Add filters like gym amenities, membership type, or class availability.

#### Personalized Exercise Recommendation (AI-based recommendations)

- Previous Workouts: Track users' previous exercises, sets, and reps to suggest progressive workout routines.
- Body Status: Integrate data from fitness trackers (e.g., Fitbit, Apple Watch) to understand real-time body metrics like heart rate, recovery, or fatigue levels.
- Goals: Let users set fitness goals (weight loss, muscle gain, endurance) and tailor exercise recommendations to help them reach those goals.
- Variety: Suggest different exercises to avoid workout monotony, while keeping within the user's fitness plan.

#### Movement Supervision and Gesture Correction (real-time feedback)

- Real-time Form Analysis: Use computer vision and pose estimation algorithms (like Open Pose or Media Pipe) to analyze body posture during workouts.
- Forms Feedback: Provide instant feedback on whether users are performing movements correctly or incorrectly.
- Instructional Guidance: Offer suggestions to correct form and reduce injury risk. Use voice or visual cues to guide them mid-workout.

### 1.3.3 Advanced Features

- Gamification (social challenges, leaderboards).
- Integration with smart home devices (Google Home, Alexa).

- AI-driven injury prevention and recovery suggestions.

### 1.3.4 Opportunity

In recent years, fitness has become an increasingly popular topic for several reasons. First, with the improvement in material living standards, obesity has become more common, and fitness is an effective way to combat obesity and maintain a healthy body. Secondly, the faster pace of life makes it easier for people to accumulate stress, and exercise is an effective way to relieve this stress, benefiting individuals in multiple ways, including improving mental health and sleep quality. Additionally, with the development of the Internet, communication technologies, and societal changes, people are more willing to share details of their lives, and fitness, as both a hobby and a daily pastime, has gradually become a means of socializing. As of 2024, the fitness industry has amassed a large user base and is thriving.

However, despite the universal appeal of fitness, individual physical differences make personalized fitness planning increasingly important. The development trends in the fitness industry also show that professionalism is no longer solely the pursuit of professional athletes but is becoming a goal for more amateur fitness enthusiasts. In recent years, adopting a professional attitude and scientific approach to fitness has become widely embraced. For example, in China, the "2022 Mass Fitness Behavior and Consumption Research Report," published by the China Sporting Goods Industry Federation, shows that running is the most popular sport, accounting for 61.0% of participants. Also, in 2022, the popularity of topics related to "running specialization" reached 495,000 mentions, representing a 446.3% increase month-on-month. This reflects people's growing desire for more scientific and effective fitness methods.

In 2016, China's State Council issued the "National Fitness Plan (2016-2020)," which proposed that by 2020, 700 million people would participate in physical exercise at least once per week, 435 million people would regularly engage in exercise, and the total scale of sports consumption would reach 1.5 trillion yuan. However, data from market research firm Zhiyan Consulting Group shows that by the end of 2015, China's fitness market was valued at only $15 billion to $20 billion, and the number of fitness clubs per million people was just 4.3. This indicates that many gym-goers still lack the financial means, resources, or expertise to fully engage in fitness activities. Since the COVID-19 pandemic, the proportion of home exercise has increased, and there is a growing need for a platform that can provide professional advice across a variety of fitness environments and cater to a broad range of fitness enthusiasts.

The widespread adoption of communication devices, networks, and emerging technologies such as natural language processing is making such platforms possible. Additionally, due to the increasing computational power available for processing enormous amounts of data, these platforms can generate highly personalized fitness programs. This is an efficient way to address the issue of supply lagging behind demand, as these platforms can integrate past gym users' workout records with professional, science-based knowledge to create tailored fitness solutions for the mass market.

# 2. Data used to train model

## 2.1 Gym recommendation

- 1. Users' location data:

Latitude and Longitude of user

- 2. Gym Data Collection:

Use Google Maps Places API.
Endpoint: google-map-places.p.rapidapi.com
Two-step data collection process:

1. Nearby Search:
   Searches gyms within specified radius (default 1000m), gets basic gym information
2. Place Details:
   Fetches detailed information for each gym, includes reviews, operating hours, and additional metadata

- 3. Review Analysis

Uses NLTK for natural language processing
Processing steps:

1. Tokenization of review text
2. Removal of stopwords
3. Word frequency analysis
4. Extraction of top 10 keywords

## 2.2 Workout-recommendation

- "Gym Members Exercise Dataset" on kaggle used for hobby recommendation
- "fmendes-DAT263x-demos" on kaggle used for weight-losing recommendation

| Age | Gender | Weight (k | Height (m) | Max_BPM | Avg_BPM | Resting_B | Session_D | Calories_ | Workout_T | Fat_Percer | Water_Int | Workout_F | Experience | BMI |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 56 | Male | 88.3 | 1.71 | 180 | 157 | 60 | 1.69 | 1313 | Yoga | 12.6 | 3.5 | 4 | 3 | 30.2 |
| 46 | Female | 74.9 | 1.53 | 179 | 151 | 66 | 1.3 | 883 | HIIT | 33.9 | 2.1 | 4 | 2 | 32 |
| 32 | Female | 68.1 | 1.66 | 167 | 122 | 54 | 1.11 | 677 | Cardio | 33.4 | 2.3 | 4 | 2 | 24.71 |
| 25 | Male | 53.2 | 1.7 | 190 | 164 | 56 | 0.59 | 532 | Strength | 28.8 | 2.1 | 3 | 1 | 18.41 |
| 38 | Male | 46.1 | 1.79 | 188 | 158 | 68 | 0.64 | 556 | Strength | 29.2 | 2.8 | 3 | 1 | 14.39 |

Figure 2.1. Overview of the Gym Members Exercise Dataset

| User_ID | Gender | Age | Height | Weight | Duration | Heart_Rat | Body_Temp | Calories |
|---|---|---|---|---|---|---|---|---|
| 14733363 | male | 68 | 190 | 94 | 29 | 105 | 40.8 | 231 |
| 14861698 | female | 20 | 166 | 60 | 14 | 94 | 40.3 | 66 |
| 11179863 | male | 69 | 179 | 79 | 5 | 88 | 38.7 | 26 |
| 16180408 | female | 34 | 179 | 71 | 13 | 100 | 40.5 | 71 |
| 17771927 | female | 27 | 154 | 58 | 10 | 81 | 39.8 | 35 |

Figure 2.2. Overview of the fmendes-DAT263x-demos dataset

When we initially planned to recommend exercises, we thought about looking for data from popular fitness websites like MyFitnessPal. We expected the data to include users' basic information and their specific fitness routines. However, when we tried to implement this idea, we found that using APIs from these websites was complicated and time-consuming. Additionally, users' personal fitness plans were rarely fully uploaded online, and the lack of data samples would greatly impact the performance of machine learning models. Therefore, we turned to publicly available datasets on Kaggle. These datasets are usually larger and more standardized, making it easier to use machine learning models to find patterns. However, the data might not be as straightforward as we had hoped.

For our fitness project, we divided it into two modules. The first module targets users who are interested in fitness as a hobby. On Kaggle, we found a dataset with 973 fitness data samples, which provides a detailed overview of gym members' exercise habits, physical attributes, and health metrics. It includes key performance indicators like heart rate, calories burned, and workout duration. Each entry also includes basic user characteristics such as age, gender, height, and experience level, allowing us to analyze fitness patterns and trends.

The second module targets people who are focused on weight loss. We believe they are more concerned with calorie burn. Different user characteristics affect their fat-burning efficiency, so the dataset we selected for this part mainly includes calorie information for 15,000 users. Aside from users' basic information, the key to this dataset is that it contains the number of calories burned by different users during various workout durations. This will allow us to calculate users' calorie consumption based on their input and provide suggestions and weight loss plans accordingly.

## 2.3 Movement Supervision and Gesture Correction

Video collection of 15 fitness postures, including one standard fitness posture video and 4-5 non-standard fitness posture videos for each posture. The videos were personally recorded by project team member Huang Chenyu at the UTown gym and the Nottinghill Suites gym.

01 Dumbbell Flat Bench Press
02 Barbell Bench Press
03 Push-up
04 Bent-over Barbell Row
05 Wide-Grip Lat Pulldown
06 Butterfly Machine Chest Fly

07 Lateral Raise
08 Standing Barbell Shoulder
09 Barbell Bicep Curl
10 Barbell Curl
11 Barbell Squat
12 Leg Extension Machine
13 Leg Press
14 Plank
15 Sit-up


Figure 2.3. 15 fitness postures

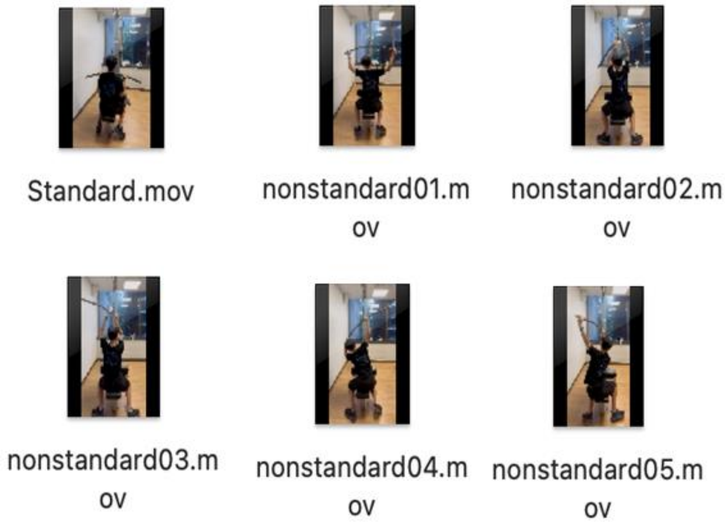



Figure 2.4. Videos in each fitness posture

Keypoint detection is performed using the YOLOv8 Pose model. A frame-skipping method is applied to the videos, capturing 5 frames every 0.5 seconds. From these 5 frames, the 1st, 3rd, and 5th frames are selected as output images.

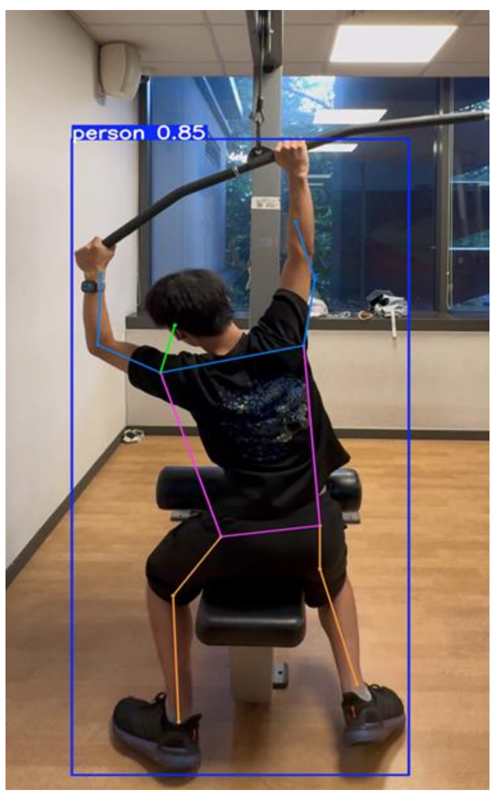



Figure 2.5. An example output image generated by Yolov8-pose model

Extract various skeletal keypoint information and define a new origin coordinate. Save all extracted keypoint coordinates in a file named “output_keypoints_data.csv” for subsequent model training.

```
# Define 12 key points to keep
keypoints_to_keep = [
    'Right Shoulder', 'Left Shoulder', 'Right Elbow', 'Left Elbow',
    'Right Wrist', 'Left Wrist', 'Right Hip', 'Left Hip',
    'Right Knee', 'Left Knee', 'Right Ankle', 'Left Ankle'
]
```

Figure 2.6. Keypoints

| | action_name | standard_type | frame_index | sequence | Nose_x | Right Eye_x | Left Eye_x | Right Ear_x | Left Ear_x | Right Shoulder_x | ... | Right Elbow_y | Left Elbow_y | Right Wrist_y | Left Wrist_y | Right Hip_y | Left Hip_y | Right Knee_y | Left Knee_y | Rig Ankle |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Dumbbell Flat Bench Press | nonstandard01 | 0 | 0 | 0.525609 | 0.598599 | 0.485133 | 0.680254 | 0.411173 | 0.791400 | ... | 0.742262 | 0.772125 | 0.871600 | 0.911305 | 0.501701 | 0.508579 | 0.261648 | 0.255344 | 0.036 |
| 1 | Dumbbell Flat Bench Press | nonstandard01 | 2 | 0 | 0.526668 | 0.600831 | 0.486696 | 0.682708 | 0.412949 | 0.791740 | ... | 0.745493 | 0.774178 | 0.876326 | 0.920597 | 0.502436 | 0.508988 | 0.261405 | 0.255176 | 0.036 |
| 2 | Dumbbell Flat Bench Press | nonstandard01 | 4 | 0 | 0.520556 | 0.595551 | 0.482698 | 0.679659 | 0.414519 | 0.789199 | ... | 0.747884 | 0.782035 | 0.876426 | 0.924035 | 0.501267 | 0.509208 | 0.259804 | 0.255130 | 0.035 |
| 3 | Dumbbell Flat Bench Press | nonstandard01 | 15 | 1 | 0.531032 | 0.598307 | 0.498048 | 0.679572 | 0.440884 | 0.801587 | ... | 0.777767 | 0.836893 | 0.918034 | 0.958946 | 0.513360 | 0.520393 | 0.274010 | 0.251076 | 0.069 |
| 4 | Dumbbell Flat Bench Press | nonstandard01 | 17 | 1 | 0.531414 | 0.604256 | 0.498173 | 0.681931 | 0.431039 | 0.797941 | ... | 0.780296 | 0.811265 | 0.920151 | 0.949701 | 0.498074 | 0.506769 | 0.260159 | 0.255756 | 0.035 |

Figure 2.7. The coordinate of keypoints saved in "output_keypoints_data.csv"

# 2.4 Post Recommendation

The MIND dataset, or Microsoft News Dataset, is a large-scale resource for research in news recommendation systems. Developed from anonymized user interaction logs on the Microsoft News website, it includes approximately 160,000 English news articles and over 15 million user impression logs from about one million users. Each news article in the dataset is accompanied by rich textual content, including titles and detailed body content, which facilitates deep learning and NLP-based recommendation research.

Key features of the MIND dataset include:

- Rich Annotations: Articles in MIND are annotated with rich information such as categories and topics, which allows for more targeted recommendation strategies.
- Diverse User Interactions: With over 15 million impression logs, the dataset captures a wide array of user interactions that can be used to train models on user preferences and click-through behaviors.
- Scale and Depth: The scale of the dataset makes it suitable for training complex machine learning models, including those using deep learning approaches that require large amounts of data.

## 2.4.1 Data understanding

```
···   Behaviors DataFrame head:
         0      1                    2  \
      0  1  U134050   11/15/2019 8:55:22 AM
      1  2  U254959  11/15/2019 11:42:35 AM
      2  3  U499841   11/15/2019 9:08:21 AM
      3  4  U107107   11/15/2019 5:50:31 AM
      4  5  U492344   11/15/2019 5:02:25 AM

                                                     3  \
      0  N12246 N128820 N119226 N4065 N67770 N33446 N10...
      1  N34011 N9375 N67397 N7936 N118985 N109453 N103...
      2  N63858 N26834 N6379 N85484 N15229 N65119 N1047...
      3  N12959 N8085 N18389 N3758 N9740 N90543 N129790...
      4  N109183 N48453 N85005 N45706 N98923 N46069 N35...

                                                     4
      0  N91737-0 N30206-0 N54368-0 N117802-0 N18190-0 ...
      1  N119999-0 N24958-0 N104054-0 N33901-0 N9250-0 ...
      2  N18190-0 N89764-0 N91737-0 N54368-0 N49978-1 N...
      3  N122944-1 N18190-0 N55801-0 N59297-0 N128045-0...
      4  N64785-0 N82503-0 N32993-0 N122944-0 N29160-0 ...

      News DataFrame head:
              0          1                 2  \
      0  N88753  lifestyle  lifestyleroyals
      ...
      1  [{"Label": "Adipose tissue", "Type": "C", "Wik...
      2                                              []
      3  [{"Label": "Ukraine", "Type": "G", "WikidataId...
      4  [{"Label": "National Basketball Association", ...
```

Figure 2.8. Data header

```
Behavior statistics:
                   0
count  376471.000000
mean   188236.000000
std    108677.960934
min         1.000000
25%     94118.500000
50%    188236.000000
75%    282353.500000
max    376471.000000

News statistics:
             0      1       2  \
count    72023  72023   72023
unique   72023     17     269
top     N88753   news  newsus
freq         1  21826    9971

                                                    3  \
count                                           72023
unique                                          70316
top     Powerball Winning Numbers For 10/26/2019 Drawi...
freq                                               21

                                                    4      5      6      7
count                                           68400  72023  72021  72018
unique                                          65871  72022  48107  51034
top     What's the weather today? What's the weather f...     []     []     []
freq                                              224      2  18734  18675
```

Figure 2.9. Data statistics

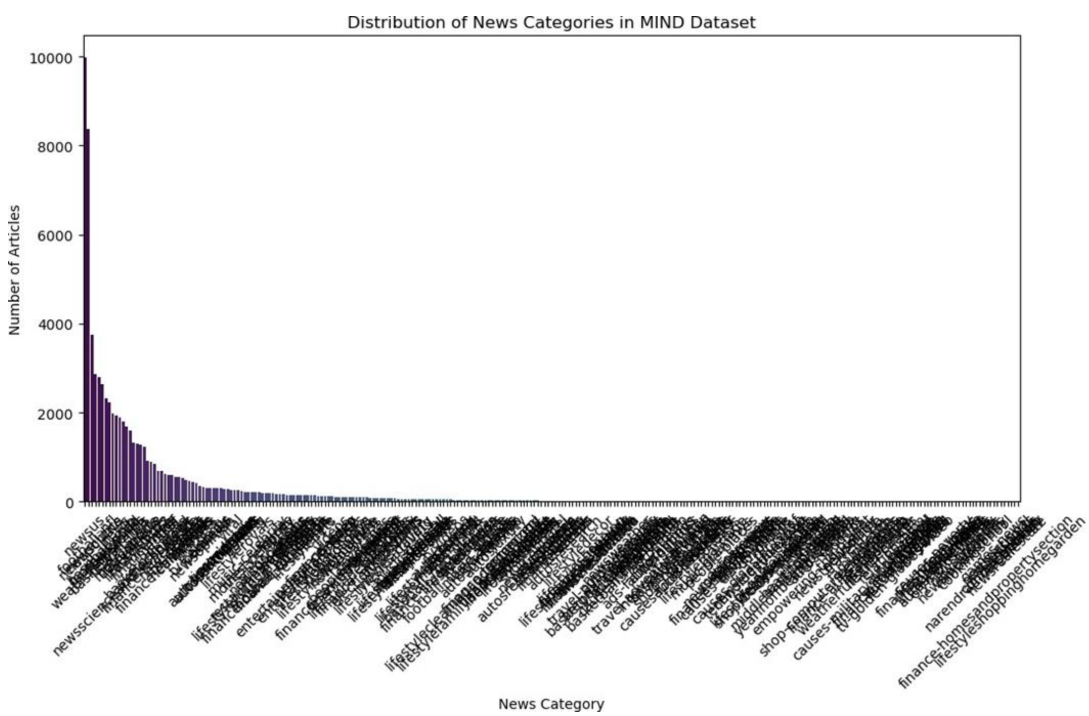



Figure 2.10. Data category

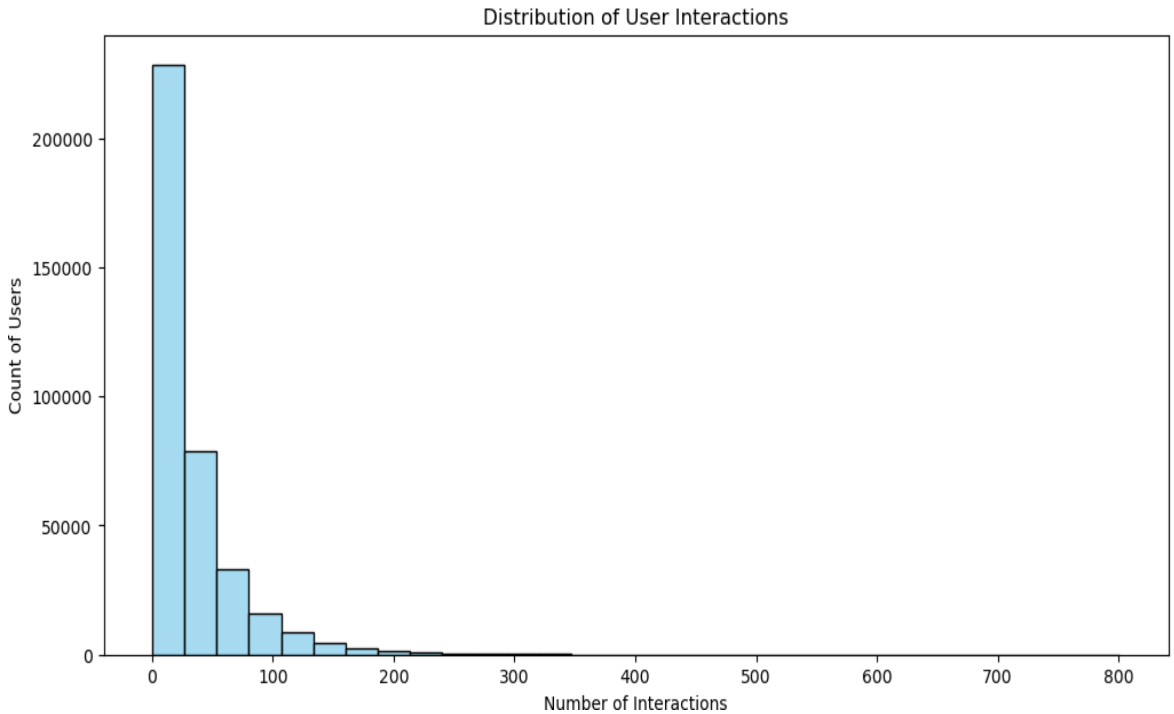



Figure 2.11. Data interaction

# 3. System Design

## 3.1 Gym recommendation

The Intelligent Reasoning System is designed to deliver straightforward gym recommendations with relevant information drawn from user reviews. The system's architecture includes a Data Collection Layer, Processing Layer, Storage Layer, and a basic Recommendation Layer.

- Data Collection Layer:

The system relies on accurate and reliable data sources for location and facility information. For user location services, it reads the users' system location, including latitude and longitude, and gives it to the data processing component. In the user interface, users can drag the map shown to them and the latitude and longitude will change based on the change of the center of the map.

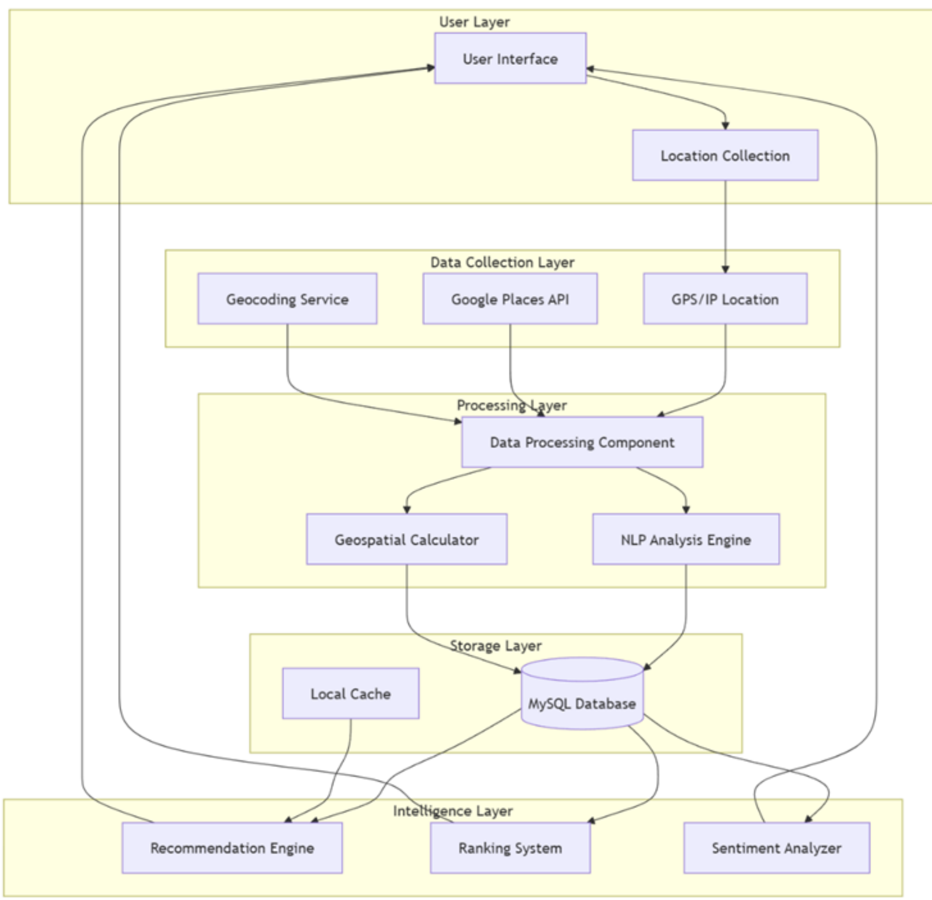



Figure 3.1. Flowchart of location-based gym finder system

- Processing Layer:

The Geospatial Calculator within this layer applies the Haversine formula to calculate the distance between the user and nearby gyms, sorting gyms by proximity. Additionally, an NLP Analysis Engine (based on NLTK) identifies the most frequently used words in user reviews. These common terms provide insights into frequently mentioned features or sentiments about each facility.

- Storage Layer:

Data from the Processing Layer is structured and stored in a MySQL Database, which maintains gym and user data with a focus on temporal consistency and optimized retrieval. Indexed and organized for rapid access, the database is supported by a Local Cache that stores frequently accessed data, reducing API call frequency and enhancing response times for users. The caching strategy also minimizes system latency, ensuring that repeated requests are handled efficiently and cost-effectively.

- Intelligence Layer:

The system's recommendation functionality is simple and proximity-based, presenting gyms sorted by distance from the user. The user interface displays the most commonly mentioned words from reviews, extracted via NLTK, to give users a quick overview of common themes or attributes associated with each facility.

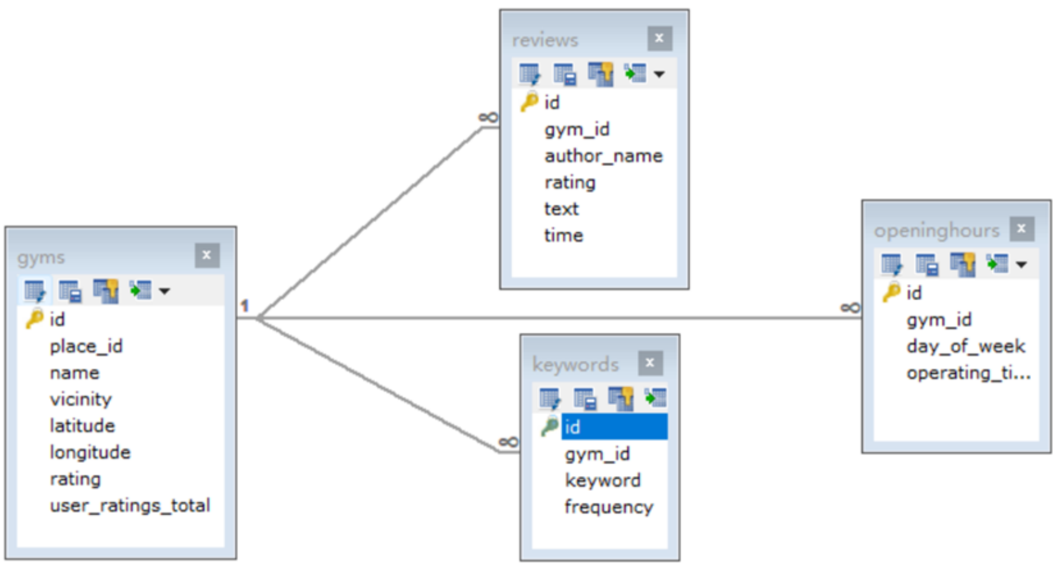



Figure 3.2. Overview of the database of location-based gym finder system

## 3.2 Workout recommendation

For users interested in fitness, our system is designed as follows: the inputs mainly include the user's age, gender, height, and experience level. This data is automatically obtained from the database based on the user's existing profile, eliminating the need for manual input. Users can update their profiles to receive recommendations that better fit their current situation. The logic here involves using a trained KNN model to find the five

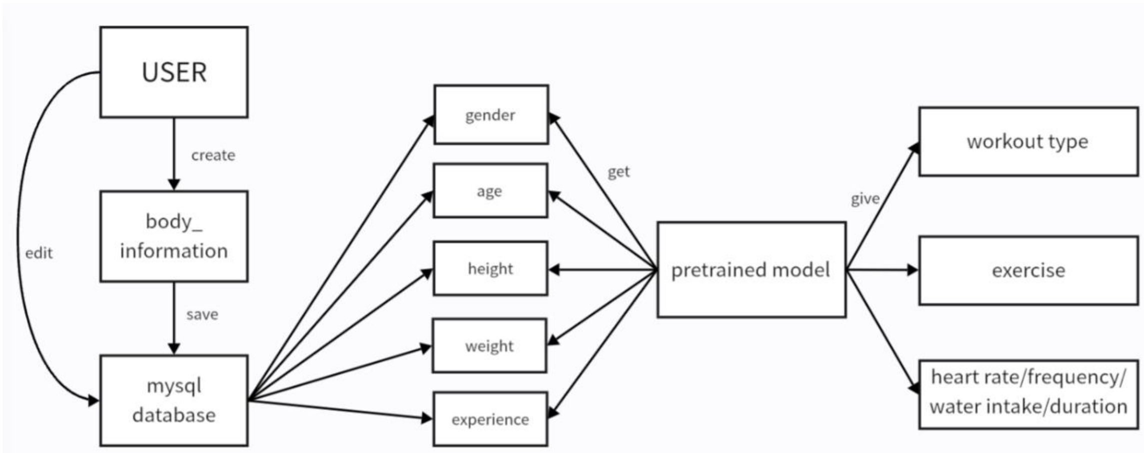



Figure 3.3. System design for fitness hobbist

closest neighbors based on the user's physical data. Among these similar users, we states their workout types and generate a recommendation exercise list based on the most frequently occurring types. We have also predefined some related exercises to facilitate the creation of workout plans. Additionally, we utilize a random forest model on the original dataset to predict various training metrics. Based on the current user's input and recommended workouts, we can provide metrics for each exercise, such as heart rate, which indicates exercise intensity, and water intake needs, helping fitness enthusiasts prepare enough water before workouts or reminding them to hydrate. The system will also suggest a recommended frequency for workouts throughout the week, with a separate random forest model trained for each metric.

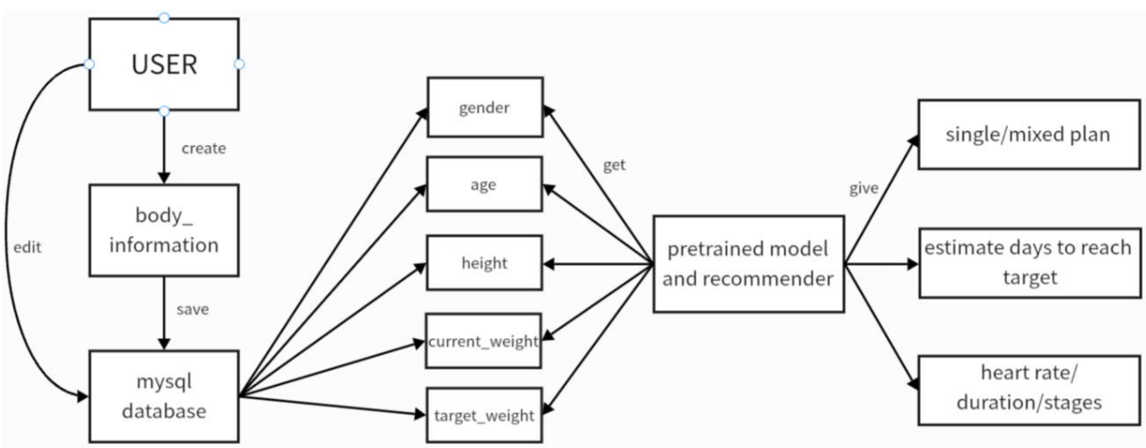



Figure 3.4. System design for weight-losing user

For users whose goal is weight loss, we recognize that they are more focused on burning fat, specifically calories. Therefore, we trained a calorie calculator using a neural network based on a large dataset. This neural network model is a fully connected deep learning structure. Its input layer matches the number of features in the training data, accepting relevant input information. The first hidden layer contains 64 neurons and uses the ReLU activation function, effectively extracting input features and capturing nonlinear relationships. Following that, a Dropout layer randomly drops neurons with a probability of 20% to prevent overfitting and enhance the model's generalization ability. The second hidden layer has 32 neurons and also uses the ReLU activation function to learn and combine information from the features extracted by the first layer. Another Dropout layer further improves model robustness. The third hidden layer contains 16 neurons and continues to use ReLU to learn more complex feature representations. Finally, the output layer consists of a single neuron without an activation function, suitable for outputting continuous values. The model employs common regularization techniques from deep learning and can predict calorie consumption fairly accurately, achieving a Train R-squared of 0.9979 and a Test R-squared of 0.9981 (with 80% of the data used for training). We also consider users' daily calorie intake, setting men at 2000 calories and women at 1500 calories, while calculating BMR (Basal Metabolic Rate) using the Mifflin-St Jeor equation. The system is designed to create a weight loss effect based on daily calorie intake and basic consumption, with increased exercise levels.

We predefined six different exercises and their corresponding intensities (the ratio of exercise heart rate to maximum heart rate): walking, jogging, swimming, sit-ups, squats, and jumping jacks. The intensity of walking is 0.5, jogging and jumping jacks are both 0.7, swimming and sit-ups are 0.6, and squats are 0.65. These exercises are common and meet the needs of those looking to lose weight. We also address the issue of decreased calorie consumption for the same exercise as weight decreases and the difference in BMR. For each exercise, we output the number of days, target weight reached, and training time for each stage, along with heart rates to clarify exercise intensity for users. Once the weight decreases to a certain extent, we provide an increased duration to achieve weight loss goals. For individual exercises, we will recalculate weight at the end of each day to arrange the next day's plan, while for mixed exercise recommendations, we have constructed a weekly workout schedule, suggesting two days of jumping jacks, two days of jogging, one day of swimming, one day of walking, and one day of sit-ups. We update weight on a weekly basis, recalculating BMR each time the weight drops by 10 kg. With these logical processes, we believe the system can create plans that are more tailored to users' situations.

## 3.3 Backend

Our backend is mainly built using Flask, and it integrates various project features while serving as a link between the frontend and the database. This backend uses the blueprint approach, making it easy to expand with new features by registering them in app.py.

We use SQLAlchemy for object-relational mapping, which maps the database into Python classes. These class definitions are stored in modules.py, making it easy to work with the database and add new classes as needed.

Basic features like user registration, login, and logout are already set up. When a user registers, we first check that their username and email aren't already in the database. Then, a verification link is sent to their email. They need to click the link to complete registration, after which their information is added to the database. This email verification is handled using flask_mail, and for added security, user passwords are stored in the database using password hashing instead of saving them directly.

For login, we generate a token to keep track of the user's session, which is needed for most functions. This token becomes invalid once the user logs out. Similarly, password resets are also done through email. The user will receive an email with a link that includes an input box to reset their password.

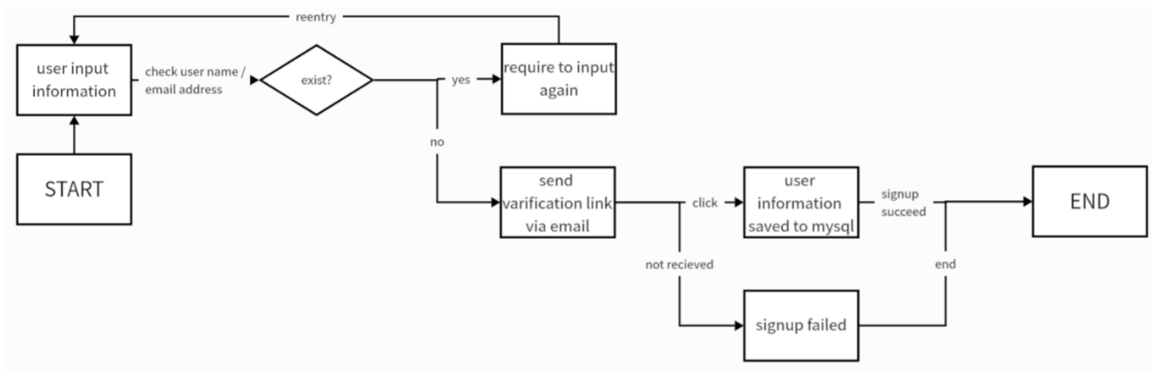



Figure 3.5. flowchart of the signup system

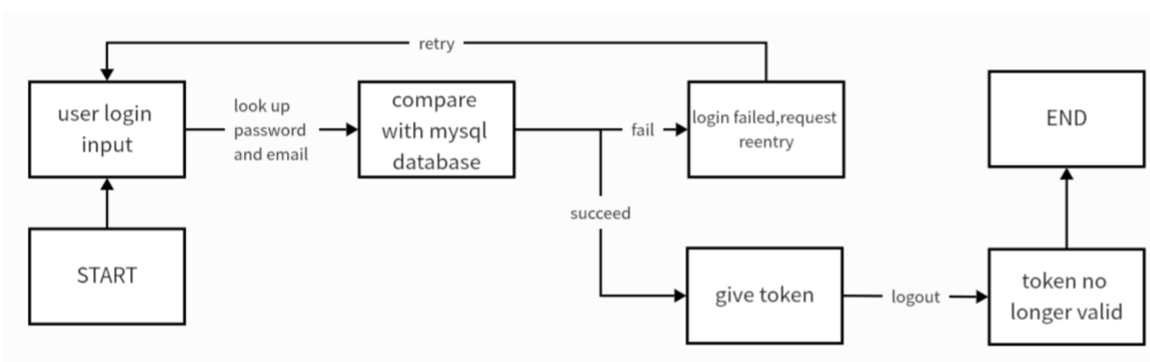



Figure 3.6. flowchart of the login system

| | id | username | email | password | created_at |
|---|---|---|---|---|---|
| ▶ | 34 | A | 1950796914@qq.com | $2b$12$4E2LlH77OkJ10kP6CKq1XuKAQw9UfG1... | 2024-10-19 14:28:48 |

Figure 3.7. password stored in mysql using password hashing

## 3.4 Movement Supervision and Gesture Correction

1 System Architecture Overview

- Input Layer:

The system first captures real-time or uploaded video/images of users performing fitness exercises.

- Processing Layer: This layer consists of two main components:
    - Pose Estimation Module: Utilizes deep learning model(YoLov8-pose) for human pose estimation to identify key points of the user's body.
    - Posture Analysis & Correction Module: Applies reasoning algorithms to compare the captured posture with ideal posture benchmarks for the exercise.
- Output Layer:

Provides real-time or post-session feedback, including visual cues, audio instructions, or detailed corrective suggestions.

2 Diagram

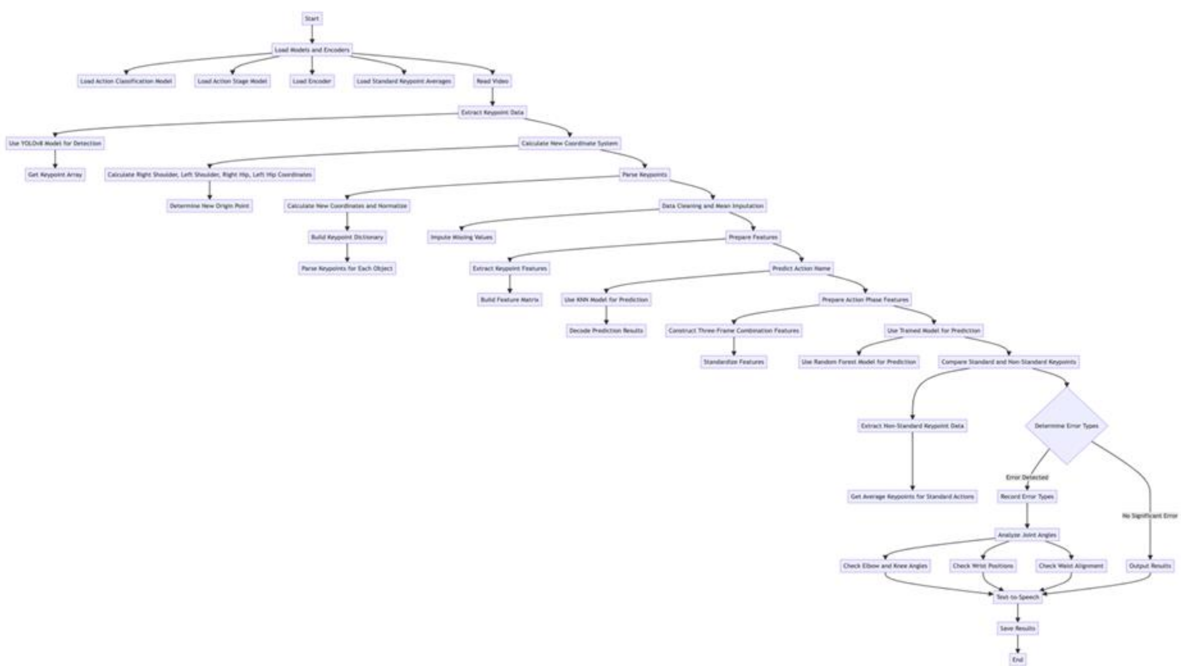



Figure 3.8. Diagram of the system

3. Reasoning Techniques and Algorithms

- Rule-Based Analysis:

Useful for scenarios with clear guidelines (e.g., squats, where knee position and back alignment are crucial).

- Machine Learning Models:

Supervised learning can be trained to recognize common form mistakes based on a labeled dataset of correct and incorrect postures.

In the entire project, the YOLOv8-Pose model is used for keypoint detection;The KNN model is employed to differentiate action types; a random forest model is utilized to determine which frame within an action sequence a single image belongs to,since the skeleton structure and movement in a single image can be coincidental and might overlap with other actions, analyzing multiple frames before and after the image helps to understand the movement's progression, enabling a more accurate classification of the action.

- Feedback Mechanism:

A feedback loop that allows corrective advice in real-time or a summary at the end of the session.

## 3.5 Post Recommendation System

The post recommendation system is developed to enhance user engagement by providing highly personalized post recommendations. This system utilizes a multi-tiered approach involving recall, rough ranking, and fine ranking phases to optimize the relevance and personalization of content presented to users.

The recommendation system comprises three key stages:

1. Recall: At this stage, embeddings generated from post content using pre-trained models (BERT， resnet) are indexed using the FAISS library for efficient retrieval. This phase quickly narrows down the potential recommendations from millions to a subset of thousands based on cosine similarity of embeddings and KNN.

Faiss is a library for efficient similarity search and clustering of dense vectors. It contains algorithms that search in sets of vectors of any size, up to ones that possibly do not fit in RAM.It also contains supporting code for evaluation and parameter tuning.Faiss is written in C++ with complete wrappers for Python (versions 2 and 3). Some of the most useful algorithms are implemented on the GPU.It is developed by Facebook AI Research.

2. Rough Ranking (Coarse Ranking): The recalled posts are then subjected to a rough ranking phase, where SVD is used. This phase utilizes basic user interaction data such as likes and views of posts to prioritize them. The objective here is to filter the subset further into hundreds of candidates.

3. Fine Ranking (Refinement): In the final stage, a more complex model takes into account detailed user profiles and following relationship graph to rank the shortlisted posts. We use node2vec to get the user embeddings and use to predict the post they like by scoring them.

# 4. Implementation

## 4.1 Gym recommendation

- Location Data Collection

This module is responsible for identifying the user's location based on their IP address. To ensure reliability, the system integrates two geolocation services: RapidAPI's IP Geolocation as the primary service and ipinfo.io as a fallback. When a user initiates a request, the system fetches their latitude and longitude, which are essential for identifying nearby gyms. This redundancy ensures location accuracy, even if one service is temporarily unavailable.

```
Current Location: Bukit Batok East Park Connector, Bukit Gombak, Bukit Batok, Southwest, Singapore, 659243, Singapore
Current Coordinate: Latitude 1.3563, Longitude 103.7542
Current Time: 2024-10-18 10:35:00.803464
```

Figure 4.1. Example of collecting user location

- Gym Data Collection

The system uses the Google Places API to gather information about gyms within a specified radius of the user's location. This data includes essential details about each gym, such as name, address, rating, and user reviews. By leveraging Google's extensive database, the system can provide users with up-to-date facility information. The information gathered here becomes the basis for recommendations and further processing in subsequent modules.

- Text Processing

In this module, natural language processing (NLP) techniques are applied to extract commonly used terms from user reviews. Using NLTK, the system processes review text to identify frequently mentioned words or phrases. These words help highlight key attributes or sentiments associated with each gym, such as "friendly staff" or "clean environment." This information is then displayed alongside each gym listing, giving users a quick summary of commonly mentioned features without deep sentiment analysis.

- Data Storage in MySQL Database

After retrieving the gym data, the system stores this information in a MySQL database. This storage process organizes the data into structured tables to facilitate quick retrieval and maintain data consistency over time. Key fields like gym name, location (latitude and longitude), rating, review text, and the most commonly mentioned review terms are stored for each gym entry.

To ensure efficient data retrieval, indexing is applied on frequently queried fields, such as gym name and location. Additionally, temporal fields are included to track when the data was last updated, which helps maintain data freshness. In cases where a gym's details change or new reviews are added, the system can check for updates by comparing timestamps, minimizing redundant data storage while keeping the database up-to-date.

```
1
Name: Bukit Gombak ActiveSG Sport Centre
Address: 810 Bukit Batok West Avenue 5
Rating: 4.3
Distance: 0.43 kilometers
Opening Hours:
  Monday: 7:00 AM – 10:00 PM
  Tuesday: 7:00 AM – 10:00 PM
  Wednesday: 7:00 AM – 10:00 PM
  Thursday: 7:00 AM – 10:00 PM
  Friday: 7:00 AM – 10:00 PM
  Saturday: 7:00 AM – 10:00 PM
  Sunday: 7:00 AM – 10:00 PM
```

Figure 4.2. Example of essential details about a gym

```
Top keywords from reviews:
  - courts: 2
  - gym: 2
  - badminton: 2
  - 6: 1
  - beside: 1
  - mrt: 1
  - station: 1
  - pretty: 1
  - good: 1
  - environment: 1
```

Figure 4.3. Example of frequently mentioned words in reviews of a gym

```
Connected Successfully
Current Database: gym_finder
Gym with place_id ChIJLzOhyEkQ2jERD-0pECpabro already exists. Updating it.
Gym with place_id ChIJKydd_kkQ2jERg2OEDQThflg already exists. Updating it.
Gym with place_id ChIJoTauEgAR2jERu2rH9GhfOAw already exists. Updating it.
Gym with place_id ChIJPyo-asoR2jERzc7PA4GB8vI already exists. Updating it.
Gym with place_id ChIJbTN-TJQR2jEROIJZ5Z6VQUw already exists. Updating it.
```

Figure 4.4. Example of updating existing gym data in the database

| id | place_id | name | vicinity | | latitude | longitude | rating | user_ra... |
|---|---|---|---|---|---|---|---|---|
| 14 | ChIJLzOhyEkQ2jERD-0pECpabro | Bukit Gombak ActiveSG Gym | 810 Bukit Batok West Avenue 5 | 29B | 1.35968840 | 103.75219390 | 4.20 | 216 |
| 15 | ChIJKydd_kkQ2jERg2OEDQThflg | Bukit Gombak ActiveSG Sport Centr | 810 Bukit Batok West Avenue 5 | 29B | 1.35966700 | 103.75222800 | 4.30 | 4 |
| 16 | ChIJoTauEgAR2jERu2rH9GhfOAw | Fitness Corner (Blk 532) | 532 Bukit Batok Street 51, Singapore | 36B | 1.35604650 | 103.74949100 | (NULL) | (NULL) |
| 17 | ChIJPyo-asoR2jERzc7PA4GB8vI | MyGymLab | 375 Bukit Batok Street 31, #01-150 | 34B | 1.35839330 | 103.74970160 | (NULL) | (NULL) |
| 18 | ChIJbTN-TJQR2jEROIJZ5Z6VQUw | Yogaful Yoga Studio | 50 Bukit Batok Street 31 | 24B | 1.35791590 | 103.74946300 | (NULL) | (NULL) |
| 19 | ChIJI6PExwoR2jERkZDsnvx4FIw | Anytime Fitness Hong Kah North CC | 30 Bukit Batok Street 31, #04-04 Hong K... | 62B | 1.35894170 | 103.74946300 | 4.50 | 19 |
| 20 | ChIJVVWV-TgQ2jERLX7b4_3Era4 | Anytime Fitness Bukit Batok CSC | 91, 05-01 Bukit Batok West Avenue 2, Bl... | 44B | 1.35263860 | 103.74982500 | 4.10 | 129 |
| 21 | ChIJWVLVSwAR2jERbD9i4Say6bo | Fitness Corner (Blk 258) | 258 Bukit Batok East Avenue 4, Singapore | 40B | 1.35137070 | 103.75916130 | (NULL) | (NULL) |
| 22 | ChIJofBPCtAR2jERbZP1ptSf9QI | Gym corner | 257 Bukit Batok East Avenue 5, Singapore | 40B | 1.34962310 | 103.75638990 | 3.00 | 1 |
| 23 | ChIJr7fxC18R2jERiFuykMURQ3g | BFT Bukit Batok | 625 Bukit Batok Central, #01-606 | 32B | 1.35148550 | 103.74869680 | 4.90 | 16 |
| 24 | ChIJganDYgAR2jERRn3Qyl_iotI | Fitness Corner (Blk 259) | 259 Bukit Batok East Avenue 4, Singapore | 40B | 1.35144220 | 103.75965090 | (NULL) | (NULL) |
| 25 | ChIJ20_P9cYR2jERsUef2pBD0So | Fitness Hub (Hong Kah North) | Bukit Batok Street 32 | 21B | 1.36063590 | 103.74818800 | 4.00 | 1 |
| 26 | ChIJ2a0-1k8Q2jERWQn7vfyPfn0 | Fighter Fitness MMA Singapore - H | 63 Hillview Avenue, #09-04 | 26B | 1.35706860 | 103.76308550 | 3.90 | 15 |

Figure 4.5. Example of gym data in the database

By storing gym data in MySQL, the system benefits from SQL's relational structure, which allows for organized data relationships and optimized querying. This approach also supports scalability as the number of users and gym data entries grows, making it practical for real-world applications.

## 4.2 Workout recommendation

For users interested in fitness, we recommend training plans based on their personal features. The recommendation is based on basic information like age, gender, height, and weight, plus an experience level. We believe different experience levels may lead to different fitness plans. Also, based on a user's height and weight, we can calculate their BMI (Body Mass Index), which is the ratio of weight to the square of height. In training the model, we first load and preprocess the workout tracking data from gym members. Preprocessing includes encoding gender (mapping "Female" to 0 and "Male" to 1) and using one-hot encoding for workout types. Then, we standardize features like age, gender, weight, height, BMI, experience level, and the one-hot encoded workout types.

After preprocessing, we train multiple Random Forest regression models. Each model predicts something different, like maximum heart rate, average heart rate, resting heart rate, workout frequency (days per week), workout duration, and daily water intake. The main part, though, is training a KNN (K-Nearest Neighbors) model to recommend similar users. All models are saved using joblib in a folder called "models" for quick access later.

Once training is done, we build the recommendation logic. Following the custom dataset with 15 exercises and 4 workout types (Strength, Cardio, HIIT, Yoga), we create a mapping table. For strength training, the exercises include dumbbell bench press, barbell bench press, barbell row, wide-grip lat pulldown, butterfly machine chest fly, lateral raise, standing barbell shoulder press, barbell bicep curl, barbell squat, leg extension machine, and leg press. For cardio and HIIT, we only include push-ups, and for yoga, we include planks and sit-ups. This setup allows a smooth connection between the modules and offers users a more complete recommendation experience.

By loading the pre-trained models, we apply the trained models during the recommendation process. After gender mapping and BMI calculation, we load the feature names from the pre-trained model and create a numpy array. Using this array, we create our DataFrame. Then, we use the trained scaler to normalize the DataFrame data, ensuring all features are on the same scale. Next, we use the trained KNN model to find the 5 most similar users and count their workout types. Since we earlier one-hot encoded the workout type features (Cardio, HIIT, Strength, Yoga), we take the last four features. By checking the values of these features, we determine the user's workout type.

In the counting process, we set all workout types to 0 at the start. Each time we get a workout type from a similar user, we increment the corresponding type by 1. Then, we sort the workout types from highest to lowest, and if the value is non-zero, we add that workout type to the recommendation list.

For each workout type in the recommendation list, we build the corresponding one-hot encoded feature and add it to the user's basic features. Then, we input this into the trained Random Forest models to predict key metrics like average heart rate, weekly workout frequency (in days), workout duration, and water intake. The predictions are returned to the front-end as object. We round the weekly workout frequency to whole numbers to make it easier to plan.

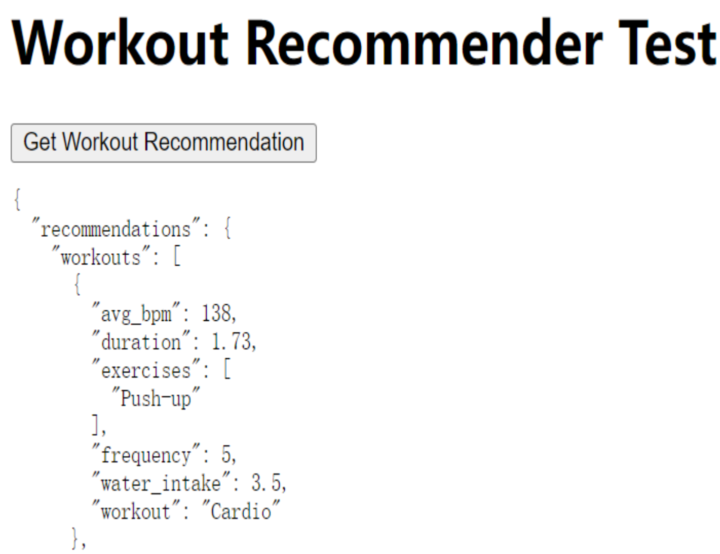



Figure 4.6. sample output of the hobbist recommender

In the module aimed at users with weight loss goals, we trained a calorie calculator. First, we read in the dataset and split it into features, including gender, height, weight, age, training duration, and heart rate (X), and the calorie values (Y). The model's logic is to predict Y based on the input X. Next, we used the ColumnTransformer class to preprocess the data differently: numerical features (age, height, weight, duration, and heart rate) were standardized using StandardScaler to ensure they had the same scale and distribution, while the categorical feature gender was encoded using OneHotEncoder for easier use by machine learning models. We then called train_test_split to divide the dataset into training and testing sets, with 80% for training and 20% for testing. After that, we built the main body of the model, which is a neural network: this model is a feedforward neural network constructed in a sequential manner, containing four layers. The first layer is a fully connected layer with 64 neurons and uses the ReLU activation function, specifying the shape of the input features (after preprocessing). The following dropout layer randomly drops 20% of the neurons to help reduce overfitting. The second layer has 32 neurons, continuing to use the ReLU activation function, followed by another dropout layer to enhance the model's robustness. The third layer has 16 neurons, still using the ReLU activation function to increase the model's non-linear expression capability. Finally, the output layer contains one neuron for predicting the regression task, without an activation function, outputting the result of a linear combination directly. The overall design aims to effectively learn patterns in the input data through stacking layers and enhance the model's generalization ability with dropout layers.

Next, we configured the learning process of the neural network: first, we used the model.compile() method with the Adam optimizer, setting the learning rate to 0.001 to balance convergence speed and stability. The loss function is the mean squared error (MSE), suitable for regression tasks to evaluate the model's prediction accuracy, and we also added the mean absolute error (MAE) as a metric to monitor model performance during training. Then, we defined two callback functions to improve the training process: the EarlyStopping callback monitors the validation loss (val_loss) and stops training early if there is no improvement for 50 consecutive epochs, restoring the best weights to prevent overfitting. The ModelCheckpoint callback saves the model with the lowest validation loss at the end of each epoch and is set to verbose=1 to output related information about the saved model.

We then started training the model using the model.fit() method, inputting the preprocessed training data X_train_preprocessed and target values y_train into the model, while using validation data X_test_preprocessed and y_test to monitor the model's performance. During training, we set the number of epochs to 1000 and specified the batch size to 32. The callback functions early_stopping and model_checkpoint were added to the training process to enable early stopping and saving the best model. By setting verbose=1, we output detailed training and validation loss information during training to monitor the learning progress and performance in real time. Finally, we saved the best training model as calories_prediction_model.keras and saved the preprocessor as preprocessor.joblib.

The initial model isn't directly designed to suggest weight loss workout plans, so we've crafted some additional logic in the recommendation system to deliver effective recommendations. First, we load a pretrained model and a preprocessor. Then, we define several exercises with their intensity levels. We assume users looking to lose weight are more focused on calories burned, as opposed to fitness enthusiasts who might have different interests. For weight-loss users, we prioritize exercises that are simple, common, and easy to do without equipment.

To guide users in picking suitable exercises, we set up a table that maps exercise types to intensity values: Walking has an intensity of 0.5, meaning it's low in energy consumption; Jogging is at 0.7, for relatively high energy burn; Swimming and Sit-ups are both at 0.6, showing a moderate intensity level; Squats are at 0.65, and Jumping Jacks are at 0.7, the same as jogging. These values help users choose workouts based on their energy levels and fitness goals, aiming to build an effective fitness plan. The recommendation system then gives training plans based on these six exercises and give out recommendations and suggestions.

To prevent burnout and avoid joint stress in heavier users, we've developed a balanced weekly exercise plan: two days of Jumping Jacks, two days of jogging, one day of swimming, one day of walking, and one day of sit-ups. Weight loss is a long-term process, so unlike programs for fitness hobbyists, we suggest users aim for daily workouts. Users can also update their physical profile anytime to adjust the plan.

We calculate each user's maximum heart rate as "220 minus their age," a common formula for estimating maximum heart rate. The recommended heart rate is then calculated by multiplying the max heart rate by the intensity of the chosen exercise. We also have a function to calculate the Basal Metabolic Rate (BMR), which is the minimum energy needed at rest. This function uses the user's gender, weight, height, and age, applying specific formulas for males and females due to physiological differences. For males, the formula is: BMR = 88.362 + (13.397 × weight) + (4.799 × height) - (5.677 × age); for females, it's: BMR = 447.593 + (9.247 × weight) + (3.098 × height) - (4.330 × age). These formulas are based on the Harris-Benedict Equation, originally proposed in 1919, and are widely used in nutrition and health management.

We also have an estimate_weight_loss_time function that estimates how long it will take to reach a target weight. If a user has already achieved or exceeded their target weight, it lets them know. Otherwise, the function calculates the calories needed per day (2000 calories for men and 1500 for women) and estimates the effectiveness of each exercise, recording the results. A pre-trained model also estimates the effects of a mixed workout plan. Finally, the function returns details about how much weight the user needs to lose to reach their goal, including the results for individual exercises and the mixed plan, which can be shown in the front end.

The calorie calculation for each exercise considers the user's characteristics and target weight, estimating the required time and other relevant data. We estimate the user's target heart rate based on exercise intensity, initialize current weight, total days, required sets for each round (where each set is 30 minutes), and record each stage. We calculate calories burned using the pretrained model, and as long as the current weight is above the target weight, we calculate the BMR and daily calorie difference. Each time the user's weight

decreases by 10 kg, we update their BMR.
For the mixed plan, we update the target weight weekly, giving the user flexibility in choosing different exercises throughout the week. To reduce processing time, we simplify calculations, like updating BMR only every 10 kg lost. We also use a basic guideline of "7700 calories burned per 1 kg of weight loss." This calorie model allows us to build a weight loss plan tailored to the user's needs.

```
Walking:
  Target heart rate: 95 bpm
  Total estimated days to reach target weight: 348
  Initial calories burned in 30 minutes: 159.10
  Exercise stages:
    At 100.0 kg: 120 minutes daily for 44 days
    At 96.3 kg: 150 minutes daily for 125 days
    At 85.2 kg: 180 minutes daily for 120 days
    At 75.1 kg: 210 minutes daily for 59 days
```

Figure 4.7. sample output of a single plan

```
Mixed Plan:
  Total estimated days to reach target weight: 210
  Weekly workout schedule:
    Jumping Jacks: 2 days per week
    Jogging: 2 days per week
    Swimming: 1 days per week
    Walking: 1 days per week
    Sit-ups: 1 days per week
  Initial calories burned in 30 minutes:
    Jumping Jacks: 304.00
    Jogging: 304.00
    Swimming: 231.74
    Walking: 159.10
    Sit-ups: 231.74
  Exercise stages:
    At 100.0 kg: 120 minutes daily for 28 days
    At 96.1 kg: 150 minutes daily for 126 days
    At 77.5 kg: 180 minutes daily for 56 days
```

Figure 4.8. sample output of mixed plan

## 4.3 Movement Supervision and Gesture Correction

- Extract specific frames from each video:

  Use YOLOv8-Pose to extract specific frames (specific images) from each video and calculate the keypoint coordinate information.

- Fitness Posture Recognition and Classification System

  Main functionalities include extracting human keypoint data from videos, normalizing these keypoint coordinates, training and testing an action classification model, and generating detailed classification results. The core of the system is to identify and classify

```
正在处理视频: /content/drive/MyDrive/fitness/dataset_fitness/13 Leg Press/nonstandard02.mov

0: 640x544 1 person, 210.5ms
Speed: 5.8ms preprocess, 210.5ms inference, 1.2ms postprocess per image at shape (1, 3, 640, 544)

0: 640x544 1 person, 214.1ms
Speed: 11.2ms preprocess, 214.1ms inference, 1.3ms postprocess per image at shape (1, 3, 640, 544)

0: 640x544 1 person, 213.8ms
Speed: 6.2ms preprocess, 213.8ms inference, 1.2ms postprocess per image at shape (1, 3, 640, 544)

0: 640x544 1 person, 228.0ms
Speed: 6.4ms preprocess, 228.0ms inference, 1.3ms postprocess per image at shape (1, 3, 640, 544)

0: 640x544 1 person, 240.1ms
Speed: 7.2ms preprocess, 240.1ms inference, 2.0ms postprocess per image at shape (1, 3, 640, 544)

0: 640x544 1 person, 381.0ms
Speed: 5.5ms preprocess, 381.0ms inference, 1.9ms postprocess per image at shape (1, 3, 640, 544)

0: 640x544 1 person, 369.9ms
Speed: 8.8ms preprocess, 369.9ms inference, 1.6ms postprocess per image at shape (1, 3, 640, 544)

0: 640x544 1 person, 359.9ms
Speed: 17.9ms preprocess, 359.9ms inference, 1.8ms postprocess per image at shape (1, 3, 640, 544)

0: 640x544 1 person, 345.2ms
Speed: 5.9ms preprocess, 345.2ms inference, 1.9ms postprocess per image at shape (1, 3, 640, 544)

0: 640x544 1 person, 463.4ms
Speed: 9.0ms preprocess, 463.4ms inference, 2.4ms postprocess per image at shape (1, 3, 640, 544)

0: 640x544 1 person, 215.9ms
Speed: 5.8ms preprocess, 215.9ms inference, 1.3ms postprocess per image at shape (1, 3, 640, 544)

0: 640x544 1 person, 230.8ms
Speed: 5.5ms preprocess, 230.8ms inference, 1.4ms postprocess per image at shape (1, 3, 640, 544)

0: 640x544 1 person, 234.8ms
Speed: 5.6ms preprocess, 234.8ms inference, 1.7ms postprocess per image at shape (1, 3, 640, 544)
```

Figure 4.9. The process of parsing the video

different fitness movements by analyzing user keypoint data, providing foundational data for posture correction.

The code utilizes the YOLOv8-Pose model to extract 17 human keypoints from videos. Each frame's keypoint data includes the x and y coordinates and confidence for each keypoint. The system processes each video frame-by-frame, capturing keypoint data every 0.5 seconds across five frames to ensure representative action data. Based on the detected keypoints, the system further calculates the average points of the shoulders and hips as a new coordinate origin, normalizing all keypoint data into a relatively consistent coordinate system.

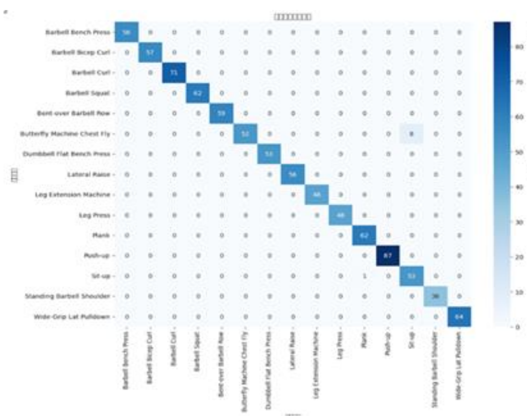
This normalization step is crucial, as it removes size and position differences between various users or videos, allowing posture data across different videos to be compared on a unified scale. During data preparation and cleaning, the code stores each video's keypoint data as a CSV file, grouped by action name and standard type, with missing values filled by the mean within each group. This approach ensures missing keypoint data is reasonably filled, improving data integrity. Next, the x and y coordinates of keypoints serve as features, and the action names as classification labels. LabelEncoder encodes the labels to make them compatible as inputs for machine learning algorithms.

In the model training phase, a KNN classifier is used. The system divides the dataset into training and testing sets, trains the model with KNN, and evaluates it on the test set, calculating classification accuracy. Additionally, it generates a confusion matrix and classification report, displaying the model's precision, recall, and F1 score across action categories, providing a detailed assessment of classification performance. As shown in the diagram, the model achieved an accuracy of 0.99.

- Fitness Movement Stage Classification Model

The code reads CSV files from the saved keypoint data. Then, it applies mean imputation to specific keypoints to reduce the impact of missing data. The data is divided into "standard movements" and "non-standard movements," so that only "standard movement" data is used for training in subsequent steps.

The code groups the standard movement data by "movement name" and "movement stage." For each group, a three-frame combination is used to extract movement features, including: keypoint positions of the middle frame, displacement differences between the first and middle frames, and displacement differences between the middle and third frames. This feature vector captures the trend of keypoint position changes over time,



动作分类报告:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Barbell Bench Press | 1.00 | 1.00 | 1.00 | 56 |
| Barbell Bicep Curl | 1.00 | 1.00 | 1.00 | 57 |
| Barbell Curl | 1.00 | 1.00 | 1.00 | 71 |
| Barbell Squat | 1.00 | 1.00 | 1.00 | 62 |
| Bent-over Barbell Row | 1.00 | 1.00 | 1.00 | 59 |
| Butterfly Machine Chest Fly | 1.00 | 0.87 | 0.93 | 60 |
| Dumbbell Flat Bench Press | 1.00 | 1.00 | 1.00 | 53 |
| Lateral Raise | 1.00 | 1.00 | 1.00 | 56 |
| Leg Extension Machine | 1.00 | 1.00 | 1.00 | 48 |
| Leg Press | 1.00 | 1.00 | 1.00 | 48 |
| Plank | 0.98 | 1.00 | 0.99 | 62 |
| Push-up | 1.00 | 1.00 | 1.00 | 87 |
| Sit-up | 0.87 | 0.98 | 0.92 | 54 |
| Standing Barbell Shoulder | 1.00 | 1.00 | 1.00 | 36 |
| Wide-Grip Lat Pulldown | 1.00 | 1.00 | 1.00 | 64 |
| accuracy | | | 0.99 | 873 |
| macro avg | 0.99 | 0.99 | 0.99 | 873 |
| weighted avg | 0.99 | 0.99 | 0.99 | 873 |

Figure 4.10. Confusion matrix and accuracy(1)

reflecting differences in movement at various stages. These feature combinations are then saved into a new data structure.

The "movement name" is one-hot encoded using OneHotEncoder to convert categorical variables into numerical features, and the encoded categorical features are combined with the keypoint feature vectors. The features are then standardized using StandardScaler to bring them to a uniform scale, avoiding inconsistencies due to differing scales among features.

The code constructs a random forest model using RandomForestClassifier to recognize different stages of standard movements. After training, the model's performance is evaluated through accuracy, classification reports, and a confusion matrix. As shown in the diagram, the model achieved an accuracy of 0.94.

Finally, the code calculates and saves the average keypoint values for standard movements as reference data. For each movement stage, the system computes the average keypoint values, resulting in reference values for keypoints at each stage of standard movements. These reference values can be used in the future to compare with users' real-time movements, identifying deviations and providing movement correction feedback.

- Error Detection

The main functionality of this part is to analyze keypoint data from fitness videos, classify movement stages, detect errors in movements, and add error prompts to the output video.

In this section, the code loads multiple pre-trained models, including a KNN action classification model, a random forest action stage classification model, as well as corresponding encoders and standardizers. Additionally, it loads the average keypoint data of standard movements, serving as a reference for detecting deviations in non-standard movements. This average keypoint data provides a reliable baseline for identifying and correcting non-standard movements accurately.

The "process_new_video" function processes new video files, including keypoint detection, action classification, and error recognition. First, the code uses the YOLO model to extract human keypoints frame-by-frame, analyzing frames at 0.5-second intervals. Next, KNN and random forest models classify actions and identify stages for each three-frame keypoint combination. Then, the code compares detected non-standard keypoints to the standard movement averages. Based on deviations in joints such as arms, knees, and waist, it identifies 13 types of errors (e.g., "left hand too far left," "left arm insufficiently bent," "waist bent") and records these error types.

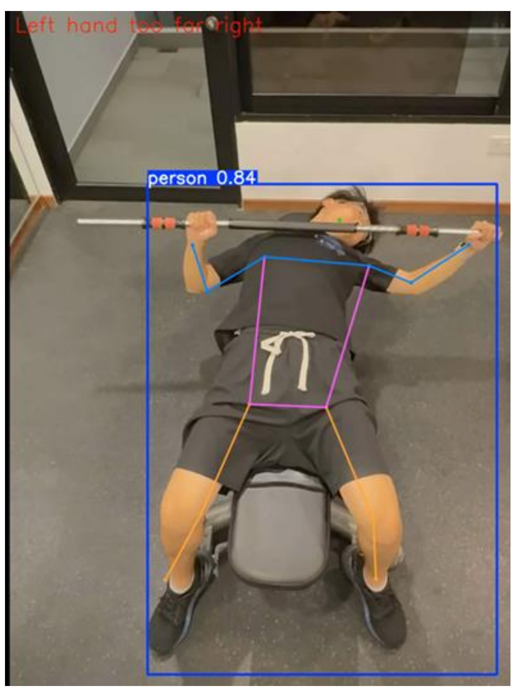



Figure 4.11. Identity and correct the movements

The “process_video_with_text” function adds error prompts to the output video. It reads the original video frame-by-frame, adding relevant error text prompts to each frame. Keypoints detected by YOLO are drawn on the video to indicate key positions. The resulting output video includes both user movements and corresponding error prompts, making it easier for users to identify and correct their movements.

Finally, error type information is saved in an Excel file (test_keypoints_error.xlsx), and the processed video file is made available for local download, enabling users to review it later. This approach not only helps users visualize their movement errors directly but also provides detailed error records, supporting movement improvement and further guidance.

| | action_name | sequence | predicted_sequence | error type |
|---|---|---|---|---|
| 0 | Barbell Bench Press | 0 | 0 | Left hand too far right |
| 1 | Barbell Bench Press | 1 | 1 | Left hand too far right |
| 2 | Barbell Bench Press | 2 | 2 | Left hand too far right |
| 3 | Barbell Bench Press | 3 | 3 | Left hand too far right |
| 4 | Barbell Bench Press | 4 | 4 | Left hand too far right |
| 5 | Barbell Bench Press | 5 | 5 | Left hand too far right |

Figure 4.12. The output excel(error type information)

# 4.4 Post Recommendation System

## 4.4.1 Data flow

```
tokenizer = BertTokenizer.from_pretrained('bert-base-uncased')
model = BertModel.from_pretrained('bert-base-uncased')
model.eval()  # Set to evaluation mode

# Tokenize and encode text
inputs = tokenizer(text, return_tensors='pt', truncation=True, padding=True)

# Get text embeddings
with torch.no_grad():
outputs = model(**inputs)
# Use the embeddings from the [CLS] token, which is commonly used for sentence-level
embeddings
text_embeddings = outputs.last_hidden_state[:, 0, :]
```

Figure 4.13. Post Embedding Generation

```
# Load a pre-trained ResNet model
model =models.resnet50(pretrained=True)
weights=models.ResNet50_Weights.DEFAULT
model =models.resnet50(weights=weights)
model.eval()  # Set to evaluation mode

# Remove the final classification layer to get feature embeddings
model = torch.nn.Sequential(*list(model.children())[:-1])
```

Figure 4.14. Post Embedding Generation

```
# Preprocess image (resize, normalize, etc.)
preprocess = weights.transforms()


img = Image.open(img path)
img_tensor = preprocess(img).unsqueeze(0)  # Add batch dimension

# Get image embeddings
with torch.no_grad():
img_embeddings = model(img_tensor)
img embeddings = img embeddings.view(img embeddings.size(0), -1)
```

Figure 4.15. Post Embedding Generation

```
# Concatenate image, title, and text embeddings
# Assuming img_embeddings and text_embeddings are both torch tensors
multimodal_embeddings = torch.cat((img_embeddings, title_embeddings, text_embeddings),
dim=1)
return multimodal_embeddings
```

Figure 4.16. Post Embedding Generation

- Post Embedding Generation:

Each post's content, whether textual or multimedia, is converted into a dense vector representation (embedding). These embeddings capture the essential semantic features of the content.

- User Embedding Generation:

```
random.shuffle(walks)
model = Word2Vec(walks, vector_size = 128, window = 10, min_count = 10, sg = 1,
workers = mp.cpu_count())
```

Figure 4.17. User Embedding Generation

- User Interaction Data:

User clicks, views, and time spent on posts are tracked and utilized in the rough and fine ranking phases to enhance recommendation relevance.

```
# Train the SVD model
model = SVD()
model.fit(trainset)
```

Figure 4.18. User Interaction Data

- Index Building:

The embeddings are stored in a FAISS index to facilitate efficient similarity-based retrieval for the recall phase.

## 4.4.2 Challenges and Solutions

- Scalability and Performance:

Managing and processing large datasets in real-time presented significant challenges. Using FAISS for the recall phase addressed scalability concerns, while distributed computing frameworks supported real-time data processing needs in the ranking phases.

- Accuracy and Personalization:

Balancing the trade-off between recommendation relevance and computational efficiency required fine-tuning of models and algorithms, particularly in the fine ranking phase where detailed user data plays a critical role.

# 5. Future plan

## 5.1 Gym recommendation

The development of our gym finder system has successfully established core functionalities including location-based gym discovery, natural language processing for review analysis, and a robust data collection infrastructure. Building upon these achievements, our future work will focus on several key directions to enhance both the system's capabilities and its practical applications.

In the immediate future, we plan to implement advanced machine learning algorithms to create a more personalized user experience. This will include collaborative filtering for recommendations based on user preferences and behavior patterns, as well as enhanced natural language processing capabilities for more nuanced review analysis. The system will be capable of understanding context-specific user requirements and providing tailored gym recommendations that consider factors beyond mere proximity.

System architecture improvements will focus on scalability and real-time capabilities. We envision transitioning to a microservices architecture deployed in the cloud, enabling better handling of increasing user loads and more efficient data processing. This will be complemented by real-time features such as live occupancy tracking and instant facility status updates, providing users with current, actionable information.

The success of these future developments will be measured not only in technical achievements but also in real-world impact, user satisfaction, and contribution to public health goals. By maintaining focus on both technological advancement and practical utility, we aim to create a system that provides lasting value to all stakeholders in the fitness ecosystem.

## 5.2 Workout recommendation

For the workout recommendations, we face some limitations. The main issues are the lack of expert knowledge and that our dataset is not specific or complete enough. When data is limited, a common approach is to get experts to give advice and create small workout plans to build a professional knowledge base, which would be more reliable than just using general online data or estimations. Also, recommending specific exercises for each workout type would usually require guidance from professionals. Unfortunately, in this course project, we don't have enough resources or time to create such a professional knowledge base.

While using our platform, though, user feedback and activity will make future improvements possible. On our platform, it will be easier to collect user feedback and gather consistent data. Many of our current predictions are limited by the original datasets, like user age ranges (e.g., gym member data ages 18-59, weight loss data ages 20-76), meaning predictions may be unreliable for ages outside these ranges. However, as we gather more user data and feedback, we'll be able to serve a broader user group. By collecting user posts and comments alongside their workout plans, we can gather ratings and feedback to adjust recommendations.

With more users—especially experienced ones—sharing their workout plans, we can build more accurate and detailed plans. In the future, we could add a rating system for workout plans and exercises, or collect feedback from posts and comments to create a more complete ratings dataset. This data could then be used to train models that provide better, more reliable workout plans.

## 5.3 Movement Supervision and Gesture Correction

The current dataset size in the project is relatively small, which limits the model's generalization ability, particularly in handling different human characteristics and exercise environments. This lack of data can lead to model overfitting on the training data, reducing the accuracy of action recognition and error detection. Future work should focus on expanding the dataset to enhance the model's robustness and adaptability. Data augmentation techniques can be applied to generate samples from various angles and poses based on the existing data, while also gathering a more diverse dataset with participants of different genders, ages, and body types, as well as videos from multiple environments. These improvements will enrich the training samples, improve the model's accuracy and resilience in diverse scenarios, and lay a stronger foundation for commercial applications.

The system currently identifies 13 common error labels but may face challenges with complex or subtle movements. Future optimizations in error detection algorithms, involving finer keypoint detection and enhanced feature extraction, could improve the precision of error classification. Accuracy in error detection is essential for the credibility of a motion correction system. A higher precision system would appeal to a broader range of users, including physical rehabilitation patients and professional athletes.

At present, the system mainly supports predefined fitness movements. Future work could expand support to a wider range of fitness and rehabilitation exercises, such as yoga, Pilates, and complex strength training movements. Additionally, incorporating multi-category detection functionality could enable recognition and classification of different actions within the same video. Supporting a broader range of movements would increase market adaptability, attracting various fitness and rehabilitation user groups. Academically, multi-category action detection is a multi-task learning challenge in computer vision, with significant research value.

The current model primarily relies on static keypoint analysis for movement detection. Future improvements could involve adding motion trajectory and dynamic sequence analysis modules. Time-series models, such as LSTM or GRU, could be used to track joint movement paths, analyzing user motion continuity and rhythm to identify dynamic features. Integrating dynamic analysis would allow the system to offer more comprehensive feedback, helping users optimize the fluidity of their movements.

Designing a more intuitive user interface with detailed analysis and visual feedback, including dynamic keypoint tracking, error highlighting, and statistical charts, would help users quickly understand feedback. Data visualization can greatly enhance user experience, making feedback easier to interpret. In the market, intuitive interface design and data presentation are key to increasing user engagement. Academically, this intersects with optimizing human-computer interaction design.

## 5.4 Post recommendation

The post recommendation system effectively employs a sophisticated multi-stage architecture to ensure high relevance and personalization of content recommendations. Through continuous optimization and integration of advanced machine learning techniques, the system aims to drive higher user engagement and satisfaction. The future plan will be as follows:

- Enhanced User Profiling:

Integrating more granular user data, such as demographic information and cross-platform behavior, to further personalize recommendations.

- Multi-modal Inputs:

Expanding the system to include more diverse data types (e.g., video, audio) in the embedding process to enrich the content understanding and recommendation diversity.

# APPENDEX

## APPENDEX A-PROJECT PROPOSAL

Our project proposal has been uploaded in the same folder as the report. You can access and review it there.

## APPENDEX B-PROJECT PROPOSAL

1. Machine Reasoning (MR)

- Model Development in Workout Recommendation:
    - MR skills are applied to refine machine learning models, especially for generating tailored workout recommendations. Iterative testing and optimization improve model accuracy, supporting decision-making in fitness recommendations.
- Calorie Estimation in Workout Recommendation:
    - The backend leverages a deep learning model to predict calorie burn based on user data, integrating MR techniques to provide precise, personalized calorie goals that help users track and manage their fitness progress.

2.Reasoning System (RS)

- Gym Recommendation System:
    - The Gym Recommendation System utilizes RS techniques to analyze location data and user reviews through NLP, offering proximity-based gym recommendations. By extracting frequently mentioned attributes in reviews, users get a quick overview of key gym features.
- Database Management:
    - In the backend, RS techniques ensure efficient data storage and retrieval, maintaining database consistency for real-time recommendations. Optimized query performance is critical for supporting the app's high-volume data transactions.
- Movement Supervision and Gesture Correction:
    - RS is applied in real-time posture assessment and movement supervision. Using rule-based analysis, the system checks user posture for accuracy, guiding correct form and reducing the risk of injury during exercises.
- Post Recommendation System:
    - The recommendation system, developed using embedding techniques and recall mechanisms like FAISS, leverages reasoning systems to interpret user behavior. By understanding user interactions, it personalizes the content recommendations to improve accuracy.

3. Cognitive Reasoning System (CGS)

- Post Recommendation System:
    - The Post Recommendation System uses CGS techniques to personalize

content based on user interactions. With a multi-stage recommendation architecture (recall, rough ranking, and fine ranking), it dynamically adapts content based on user behavior, providing highly relevant and engaging posts.

- NLP in Post Recommendation:
    - NLP techniques analyze post titles and content to interpret and process textual information, enabling the system to provide posts that match user interests and enhance personalization accuracy.
- Computer Vision (CV) in Movement Supervision:
    - In Movement Supervision, CV algorithms analyze visual content to provide real-time posture feedback during workouts. This cognitive feature helps users correct their form through context-sensitive guidance, enhancing their training experience.
- Cognitive Adaptation in Frontend:
    - The frontend interface applies cognitive reasoning by dynamically adapting content and layout based on user preferences. Using CGS techniques, the app personalizes interactions to improve usability and engagement.

Additional Skills Mapped to Knowledge Areas:

- Frontend and UI Design:
    - The frontend, designed using SwiftUI and mobile-first principles, provides a user-friendly interface that aligns with CGS by ensuring intuitive interaction and optimal usability across devices.

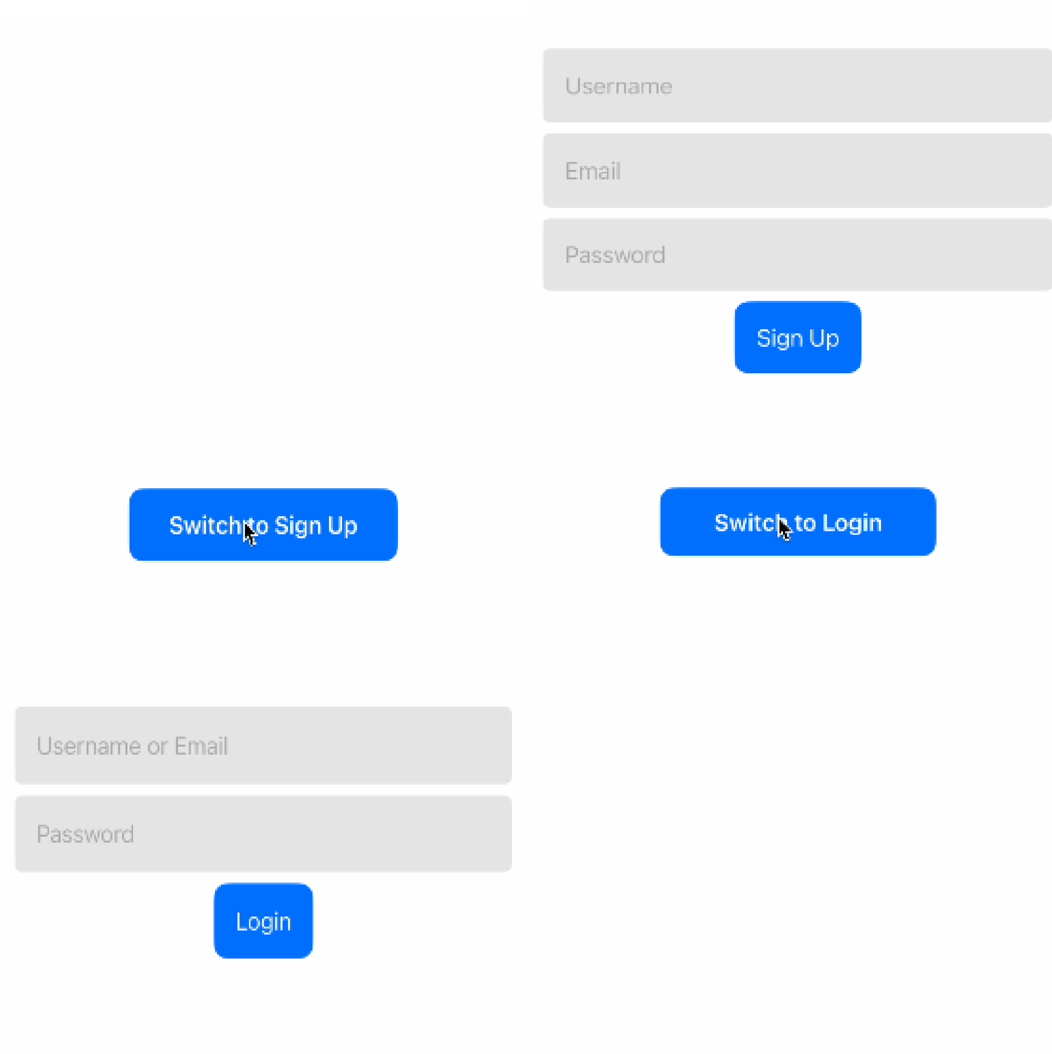
# APPENDEX C-USER GUIDE

For users of MAC and xcode, you can install our project easily.

**Login and Signup:**

Step1. Click on our icon

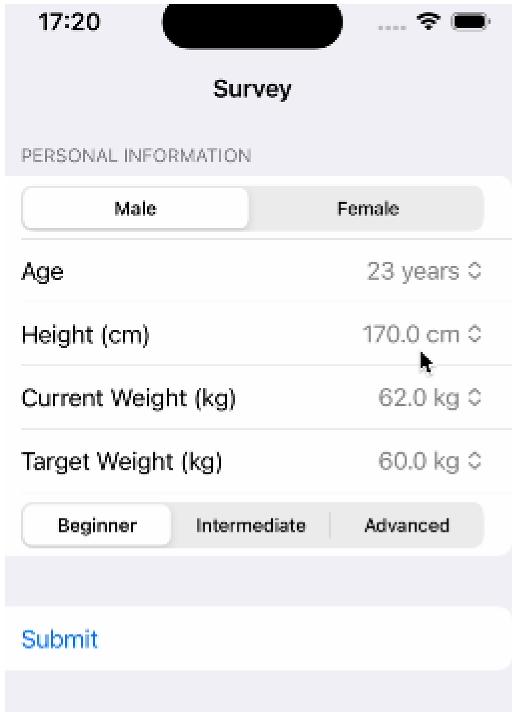
Step2. Choose to signup or login by clicking switch to button



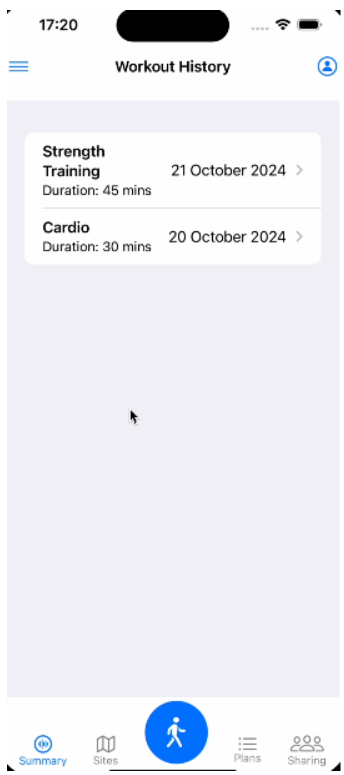

Remember, you will have to provide your real email address in order to complete a successful signup and login. If you don't click the verefication link in the email, your information won't be recorded in our database which means a failure in signup.
After login, you can now experience our various functions.

**Survey:**
In order to keep our recommendation's accuracy, you will need to complete a survey so

that we can know your body information, so please provide accurate result.

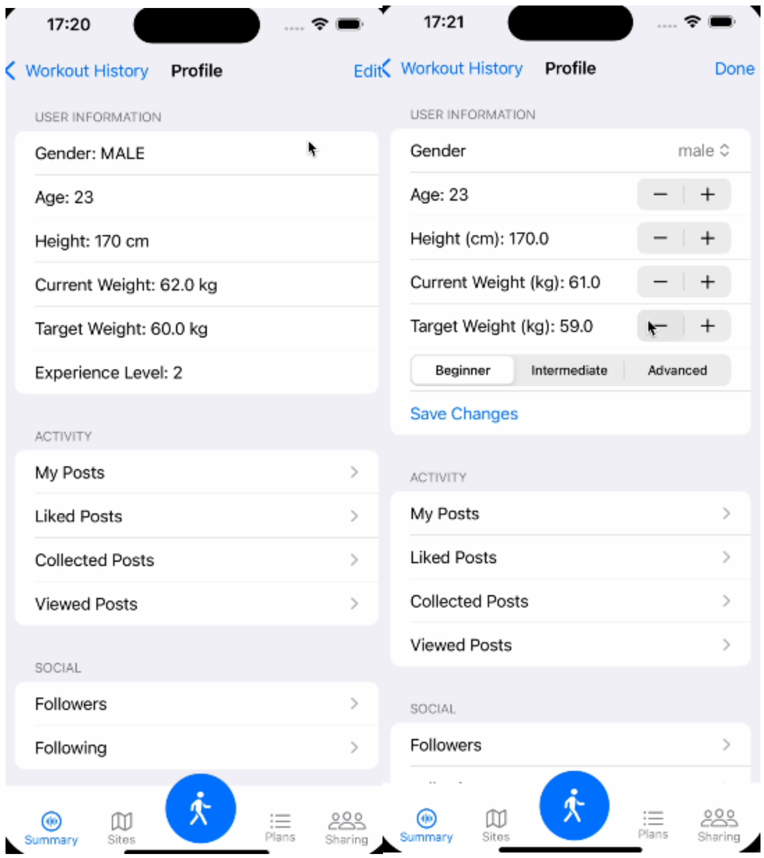



After submittion, you can now see our home page where we can see our workout history.

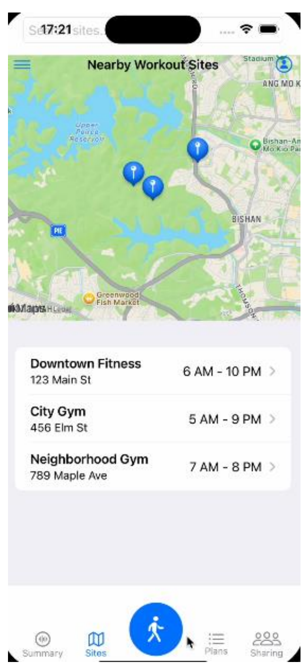



**USE GUIDE:**

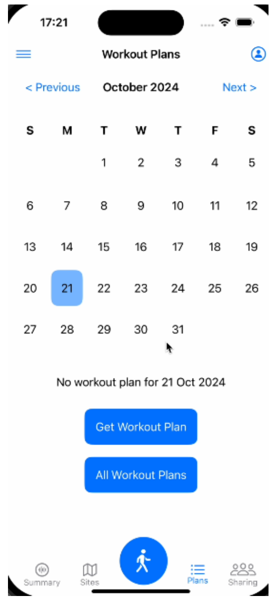
Clicking the profile icon in the top-right corner allows us to edit our profile.



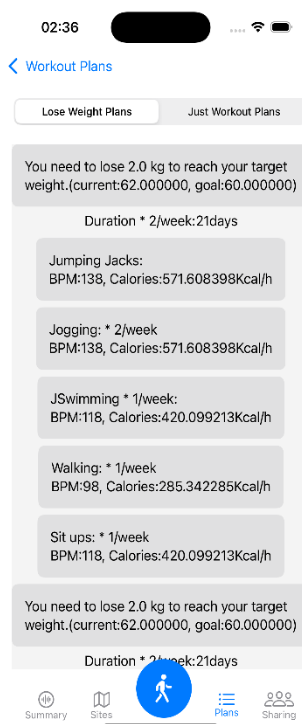

You can also see your posts history here. Next, we click the **Sites** button in the bottom toolbar.



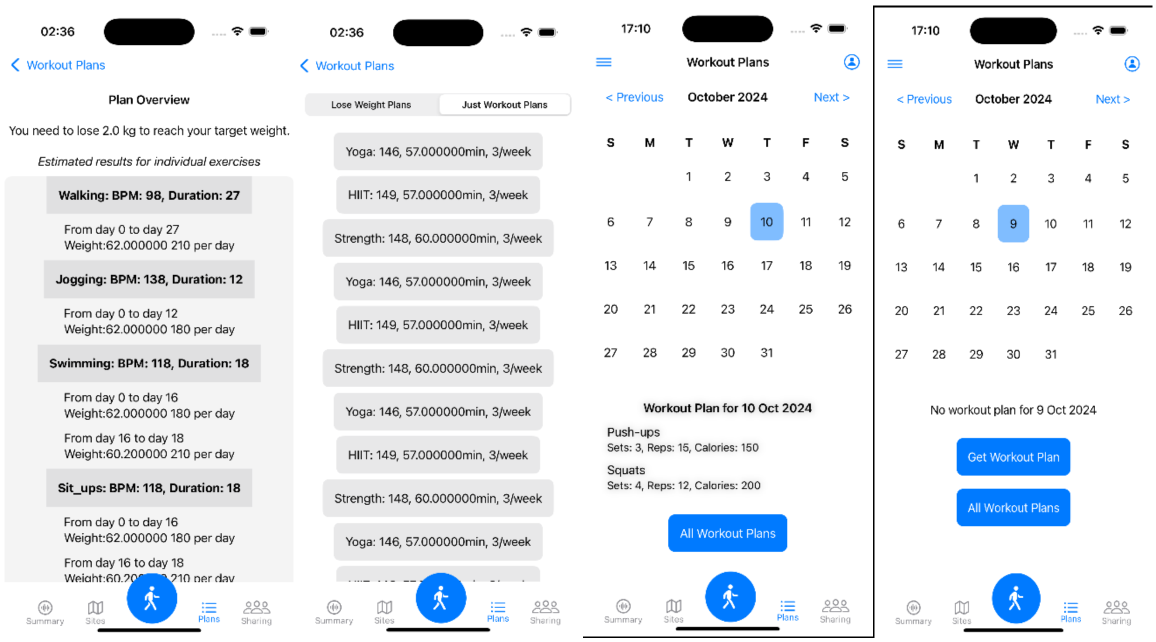

You will now see those gyms around your current loccation, and the opening hours of them.




Click on the plans in the bottom toolbar, and you can see a calendar. Press the "Get Workout Plan" to generate your workout plan.



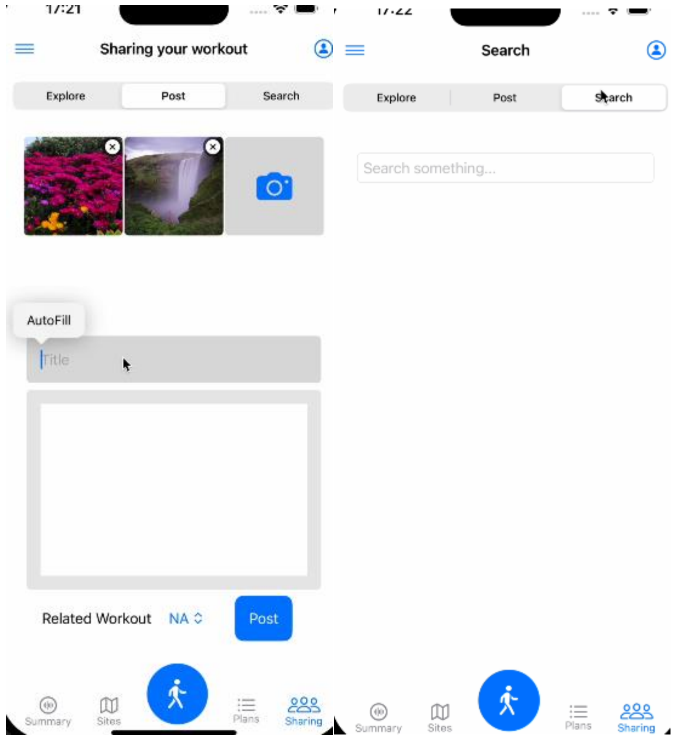

Choose the plan that suits your needs: weight loss or regular exercise? Feel free to select either one!



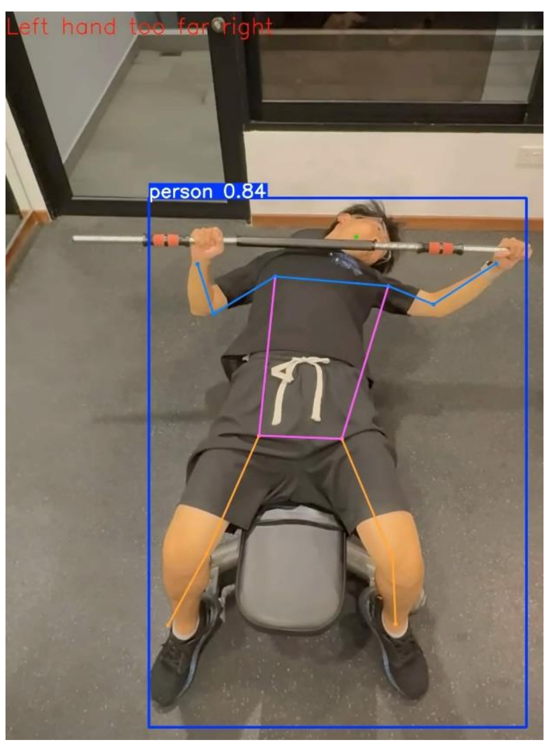

You can also view more detailed workout information or check your exercise plan on the calendar—no problem at all!


Clicking the Sharing tab in the toolbar takes you to our community feature.

The **Explore** feature at the top allows you to view posts from other users. You can also choose to click **Post** to share your own updates.While **Search** leads to more specific contents.




Click the round blue icon in the center of the bottom toolbar to view your movement guidance. Of course, your camera needs to be open.



Click the button in the top-left corner to open the quick menu, where you can log out.