CHAPTER 1: INTRODUCTION

1.0 INTRODUCTION:

In today's fast-paced world, maintaining good health and fitness has become increasingly challenging due to irregular lifestyles, unhealthy dietary habits, and a lack of personalized guidance. Sedentary routines, work stress, and insufficient knowledge about effective exercise and nutrition further exacerbate these issues, leading to a rise in lifestyle-related health problems. In response, technological advancements in Artificial Intelligence (AI) have opened new possibilities for health and fitness management. AI-driven systems can analyze large volumes of user data, detect patterns, and provide personalized recommendations tailored to individual needs. These systems have the potential to help users achieve fitness goals more efficiently and adopt healthier routines.

However, most traditional AI models operate as "black boxes," offering predictions or suggestions without revealing the reasoning behind them. Such lack of transparency can undermine user trust, reduce engagement, and limit the practical adoption of AI-based recommendations. Users may feel hesitant to follow suggestions when they cannot understand how or why a particular diet plan, workout, or lifestyle recommendation is generated. This creates a significant gap in the adoption of AI-driven health solutions.

To overcome these limitations, the proposed system—XAI-Enabled Health and Fitness Recommendation System—integrates Explainable Artificial Intelligence (XAI) techniques into the recommendation process. The system collects and analyzes user health data, including physiological parameters, activity patterns, dietary preferences, and wearable sensor data, to generate customized fitness and nutrition plans. By employing XAI methods such as LIME and SHAP, the system provides interpretable explanations for each recommendation, helping users understand the reasoning behind suggested workouts or diets. This enhances transparency, improves decision-making, and fosters greater trust in AI-driven solutions.

1.1 PROJECT OVERVIEW

A super-smart fitness buddy living inside your app—one who actually pays attention to what you eat, how you move, and what goals you set for yourself. That's basically the heart of this system. Users log in through a clean, friendly mobile or web interface where they can track meals, workouts, steps, and sleep. They can even hook up wearable devices to feed in activity data automatically. Everything you do during the day becomes raw material for your personalized health story.

Once the data is in, Firebase steps in like a super-organized librarian—authenticating users, storing logs, managing files, and keeping everything synced across devices. Then the real magic happens: Google Gemini, your AI powerhouse, analyzes the patterns in your diet, lifestyle, and goals. It cooks up diet plans, exercise routines, and habit-building suggestions tailored just for you. But the system doesn't stop at "Here's what you should do." Through XAI modules like SHAP or LIME, it actually explains why it gave those suggestions—almost like the AI is sitting next to you saying, "Hey, I recommended strength training today because your recent activity shows low muscle engagement.

Finally, everything gets packaged into a clean, visual dashboard that feels less like reading a medical report and more like checking your daily horoscope—but accurate. You get personalized meal plans, workout flows, activity summaries, and explanation panels that break everything down in plain language. Instead of leaving you guessing, the system shows you how today's choices influence tomorrow's goals, helping you stay motivated and understand the "why" behind every recommendation. It's like having a transparent fitness coach who actually respects your curiosity.

1.2 PURPOSE OF THE PROJECT

The purpose of this project is to create an intelligent and transparent health and fitness system that provides personalized diet and exercise recommendations while clearly explaining the reasoning behind each suggestion. By combining user health data, daily activity tracking, and Explainable AI techniques, the system aims to help individuals make informed lifestyle choices, build trust in AI-driven guidance, and stay consistent with their fitness goals through simple, understandable, and meaningful insights.

1.3 SCOPE OF PROJECT

1. **User Profiling:** Collect key health details, fitness goals, activity levels, and dietary needs. Auto-calculate BMI, BMR, and TDEE for accurate personalization.

2. **Personalized Recommendations:**Generate tailored meal plans, calorie targets, macro splits, and weekly workout routines based on each user's unique profile.

3. **Explainable AI:** Every recommendation comes with simple, human-friendly explanations using feature importance, reasoning steps, and confidence insights.

4. **Daily Tracking Integration:** Connect user logs and activity data to continuously update and refine recommendations.

5. **Smooth User Experience:** Combine all modules — data, AI, explanations, tracking — into one easy, supportive system.

CHAPTER 2: LITERATURE SURVEY

2.1 REVIEW OF EXISTING WORK

The recent wave of AI-based health and diet systems shows a growing effort to make nutrition guidance smarter and more personalized. For example, HealthGenie (Gao et al., 2025) blends Knowledge Graphs with large language models to give structured dietary suggestions and visual explanations. It's a powerful approach, but it also comes with the challenge of building and maintaining a complex knowledge graph that doesn't scale easily in real-world environments.

Other works focus on deeper personalization using more advanced reasoning. ChatDiet (Yang et al., 2024) uses causal models along with LLM orchestration to understand why certain foods suit different users, aiming for more accurate recommendations. At the same time, Purrfessor (Lu et al., 2024) brings multimodal AI into the mix by combining image and text analysis to interpret meals directly from photos. These systems push the boundaries of personalization, but they also demand heavy computational resources and face issues like dataset bias, which can affect fairness and reliability.

Alongside these domain-specific tools, researchers have also explored the explainability side of AI in health. Allen (2024) provides a broad review of XAI techniques like SHAP, LIME, and attention methods, showing how important transparency is becoming in digital health—yet many of these methods still lack real-world implementation. Similarly, OBESEYE (Roy et al., 2023) demonstrates how interpretable ML models can support obesity management, but its narrow focus limits general use. Overall, the existing studies reveal strong progress but also clear gaps in scalability, computation efficiency, fairness, and real-world user experience—gaps your project is perfectly positioned to bridge.

2.2 REVIEW IN TABLE FORMAT

SL. No.	Author(s)	Year	Title	Method Used	Gap / Disadvantage
1	Fan Gao; Xinjie Zhao; Ding Xia; Zhongyi Zhou; Rui Yang; Jinghui Lu; Hang Jiang; Chanjun Park; Irene Li	2025	HealthGenie: Empowering Users with Healthy Dietary Guidance through Knowledge Graph and Large Language Models	Combined Knowledge Graph (KG) + Large Language Model (LLM) for structured dietary recommendations and visual explanations	Requires complex KG construction; limited scalability beyond test cases
2	Zhongqi Yang; Elahe Khatibi; Nitish Nagesh; Mahyar Abbasian;Iman Azimi; Ramesh Jain; Amir M. Rahmani	2024	ChatDiet: Personalized NutritionOriented Food Recommender Chatbots through an LLMAugmented Framework	Integrated personal and population models + orchestrator + LLM; causal discovery & inference for personalization	Computationally intensive; personalization depends on causalmodel quality
3	Linqi Lu; Yifan Deng; Chuan Tian; Sijia Yang; Dhavan Shah	2024	Purrfessor: A Fine-Tuned Multimodal LLaVA Diet Health Chatbot	Fine-tuned LLaVA (vision+language) with human-in-the-loop; multimodal food-image + text analysis	High compute requirements; dataset bias may affect fairness

4	Ben Allen	2024	The Promise of Explainable AI in Digital Health for PrecisionMedicie: A Systematic Review	Systematic review of XAI methods in digitalhealth(SAP, LIME,attention, Rule extraction, etc.)	Lacked practical implementation examples; limited user-centric evaluation
5	Saman Forouzandeh; Mehrdad Rostami; Kamal Berahmand; Razieh Sheikhpour	2024	Health-Aware Food Recommendation System with Dual Attention in Heterogeneous Graphs (HFRSDA)	Dual hierarchical attention over heterogeneous graphs (users, foods, health factors) for healthaware recommendation	High complexity and training cost; explainability mainly for expert users
6	Mrinmoy Roy; Srabonti Das; Anica Tasnim Protity	2023	OBESEYE: Interpretable Diet Recommender for Obesity	MLmodels (Linear Regression, Random Forest, LightGBM,XGBoost) for nutrient prediction +	Focused on obesity cases; limited generalization to healthy population
7	Matteo Magnini; Giovanni Ciatto; Furkan Cantürk; Reyhan Aydoğan; Andrea Omicini	2023	Symbolic Knowledge Extraction for Explainable Nutritional Recommenders	Neural model + symbolic rule extraction integrated with expert prescriptions	Complex rule maintenance; limited adaptability to changing datasets

8	Juan LopezBarreiro; Jose Luis GarciaSoidán; Luis Alvarez-Sabucedo; Juan M. Santos-Gago	2023	Practical Approach to Designing and Implementing a Recommendatio System for Healthy Challenges	Modular recommender framework: user profiling, behavioral goals, evidence-based challenge selection	Limited explainability; lacked long-term adherence validation
9	Yi Chen; Yandi Guo; Qiuxu Fan; Qinghui Zhang; Yu Dong	2023	Health-Aware Food Recommendation Based on Knowledge Graph and Multi-Task Learning	Knowledge Graph + Multi-Task Neural Network learning user preferences and health effects (FKGM model)	Focused on nutrientlevel explainability only; needs broader context integration
10	Mehrdad Rostami; Vahid Farahi; Kamal Berahmand; Saman Forouzandeh; Sajad Ahmadian; Mourad Oussalah	2022	A Novel Explainable and Health-aware Food Recommender System	Hybrid rule-based + preference learning system with explicit explainability (conference paper/KDIR proceedings)	Small dataset (Allrecipes crawl); rule-based parts may struggle with coldstart/new items
11	Flavio Di Martino; Franca Delmastro	2022	Explainable AI for Clinical and Remote Health Applications: A Survey on Tabular and Time Series Data	Survey of XAI approaches on time-series health data; covers methods use-cases	Did not cover realtime recommender systems; largely theoretical

12	Hung-Kai Chen; Fueng-Ho Chen; Shien-Fong Lin	2021	An AI-Based Exercise Prescription Recommendation System	Neural network predicting exercise type/duration from demographic & physiological data	Lacked explainability; dataset diversity limited

2.3 LITERATURE REVIEW SUMMARY

The reviewed literature highlights the integration of Explainable Artificial Intelligence (XAI) in personalized health and recommendation systems. Several studies focus on dietary and nutrition recommendations using knowledge graphs, large language models (LLMs), and multimodal approaches to deliver personalized, interpretable guidance. Systems like HealthGenie and ChatDiet combine structured knowledge with AI-driven reasoning to support users in making healthy choices, though they face challenges in scalability, computational cost, and dataset bias. Graph-based attention mechanisms and multi-task learning methods enhance health-aware recommendations by modeling complex relationships between users, foods, and health factors, but often at the cost of high training complexity.

Other works explore interpretable machine learning models, including linear regression, tree-based models, and symbolic rule extraction, demonstrating how XAI can increase trust and transparency in recommendations. Systematic reviews emphasize popular XAI methods such as SHAP, LIME, attention mechanisms, and rule-based explanations, but note a lack of practical, user-centered implementations. Modular frameworks for health challenges and AI-based exercise recommendation systems show the potential of personalized health guidance, yet often lack sufficient explainability or generalization across diverse datasets.

Overall, these studies indicate that XAI is essential for building trustworthy and effective health recommendation systems. Promising strategies combine AI, knowledge graphs, multimodal data, and rule-based reasoning, but challenges remain in scalability, adaptability, and user-friendly interpretability. Future research should focus on developing efficient, explainable, and user-centric systems that support personalized health decisions and improve overall outcomes.

2.4 GAPS IDENTIFIED

1. **Scalability & Generalization:** Several systems (e.g., HealthGenie, ChatDiet) rely on complex knowledge graphs, multimodal models, or rule-based reasoning, which can be difficult to scale to larger user bases, diverse diets, or different cultural contexts. Their performance is often limited to the datasets used for testing, reducing real-world adaptability.

2. **Explainability & User Interpretability:** While many papers implement XAI methods (SHAP, LIME, attention mechanisms, symbolic rules), the explanations are often technical or expert-focused. End users may find them difficult to understand, limiting trust and adoption in practical health decision-making.

3. **Personalization & Context Awareness:** Some recommendation systems provide generic guidance or focus narrowly on specific health aspects (e.g., obesity or nutrient intake). They often lack integration of broader context such as lifestyle, exercise habits, or multi-factor health conditions, reducing the accuracy and relevance of recommendations.

4. **Computational Efficiency:** Systems using LLMs, multimodal models, or dual-attention networks require high computational resources, which may prevent deployment on low-power devices or in real-time scenarios, limiting accessibility.

5. **Integration with Health Ecosystems:** Few studies consider linking recommendations with other health platforms (fitness trackers, electronic health records, or hospital apps). Data management and interoperability challenges remain largely unaddressed.

6. **Validation & Long-Term Adoption:** Most systems are validated on small datasets or short-term studies. Long-term adherence, user satisfaction, and real-world health outcomes are rarely evaluated, which limits understanding of their practical impact.

2.5 PROBLEM DESCRIPTION

Many health and fitness applications claim to offer personalized guidance, yet most still rely on generic workout routines and broad dietary plans that fail to match users' unique needs or medical conditions. Users often receive recommendations without understanding the reasoning behind them, which reduces clarity and trust. Most existing systems also do not integrate important data such as daily activity, calorie burn, fitness goals, and dietary preferences in a meaningful way, resulting in limited personalization. This lack of adaptability makes it difficult for users to stay motivated and consistent. Another major issue is the absence of Explainable AI (XAI), as many AI models act like black boxes and provide suggestions without revealing how they were generated. Without clear explanations for calorie targets or exercise plans, user engagement and long-term adherence decrease.

2.6 PROBLEM STATEMENT

The problem this project tackles is the lack of personalized and transparent fitness recommendation systems. Most existing apps give generic plans without explaining why they are suitable, making users confused and less motivated. Without Explainable AI or meaningful use of real-time data, current systems fail to offer trustworthy, user-centric guidance. This creates a need for an intelligent, explainable solution that clearly justifies every diet and exercise recommendation.

2.7 OBJECTIVES

1. To develop a cloud-synced system that manages user profiles and tracks daily activities using Firebase.

2. To design a recommendation engine that gives personalized diet and exercise plans with clear SHAP-based explanations.

3. To implement a user feedback loop that collects ratings, suggestions, and feasibility inputs to improve trust and transparency.

2.8 SYSTEM REQUIREMENTS

1. Hardware Requirements:

- **Processor (CPU):** Intel i5/i7 or AMD Ryzen 5/7 (preferably 10th generation or higher) for efficient model training and inference.
- **Graphics Card (GPU):** NVIDIA GTX 1660 / RTX 2060 or higher for running deep learning and LLM-based models.
- **RAM**: Minimum 16 GB (32 GB recommended for large datasets or multimodal models).
- **Storage:** 512 GB SSD (1 TB preferred for storing datasets, models, and logs).
- **Display:** Full HD monitor for clear visualization of dashboards and recommendation outputs.
- **Network:** Stable internet connection for API access, cloud-based model usage, and updates.

2. Software Requirements:

- **Operating System:** Windows 10/11, macOS, or Linux (Ubuntu 20.04 recommended).
- **Programming Languages:** Python 3.8+ optionally R for statistical analysis.
- **Libraries & Frameworks:** Machine Learning & Deep Learning: TensorFlow, PyTorch, Scikit-learn, XGBoost, LightGBM
- **Natural Language Processing:** Hugging Face Transformers, SpaCy, NLTK
- **Explainable AI:** SHAP, LIME, ELI5, Captum
- **Knowledge Graphs:** RDFlib, Neo4j, NetworkX
- **Database:** MySQL, PostgreSQL, Firestore, or MongoDB for storing user profiles, dietary data, and recommendations.
- **Web/GUI Frameworks:** Flask, Django, or Streamlit for building user interface dashboards.
- **Visualization Tools:** Matplotlib, Seaborn, Plotly, or Dash for charts and graphs.
- **Version Control:** Git and GitHub/GitLab for source code management.

CHAPTER 3: METHODOLOGY

3.1 RESEARCH DESIGN

- **Define the goal** — Build an XAI-powered health & fitness system that gives personalized diet/workout plans and explains why each suggestion fits the user.

- **Recruit users & set baseline** — Bring in 60–120 participants, collect consent, health profiles, fitness goals, and initial activity levels.

- **Build the system** — Set up the mobile app, Firebase backend, daily trackers, Google Gemini recommendation engine, and the XAI module for simple human explanations.

- **Assign study groups** — Split users into Control (no explanations) and Treatment groups (text-based explanations, visual explanations, or example-based).

- **Run the study** — Let users use the system for 8–12 weeks: log meals, workouts, sleep, and follow recommendations while the app adapts daily.

- **Collect data** — Store daily activity, diet logs, wearable data, weekly surveys, and short end-of-study interviews about trust, clarity, and usability.

- **Analyze results** — Compare groups for adherence, progress, and understanding using stats (t-tests/ANOVA) + qualitative feedback to find what explanation style works best.

- **Validate & conclude** — Check whether explanations are faithful to the model, summarize insights, refine the system, and deliver the final report + improved XAI design.

3.2 TOOLS AND TECHNOLOGIES

3.2.1 Hardware Requirements

Processor: Intel i5 or higher (recommended for development and agent simulation)

RAM: Minimum 8 GB (16 GB recommended for multi-agent tasks)

Storage: 256 GB SSD or higher

GPU (Optional): Required only for AI model training (e.g., intel iRIS or above)

3.2.2 Software Requirements

Operating System: Windows 10/11 / Linux (Ubuntu 20.04+) / macOS

IDE: VS Code / PyCharm / Qoder IDE

Web Browser: Chrome / Firefox / Edge / Safari

Runtime Environment: Python 3.8+ (for Flask-based development)

Package Manager: pip / conda

3.2.3 Development Tools

Frontend: HTML5/CSS3/JavaScript

Backend: Flask 3.0 (Python)

Language: Python

AI/XAI: SHAP 0.49, scikit-learn 1.3

Data: pandas 1.5

Deployment: Local server/Heroku/Python Anywhere

3.3 SYSTEM WORKFLOW

The system begins by collecting user details such as age, weight, activity level, fitness goals, and dietary restrictions, then automatically calculates BMI, BMR, and TDEE to create a personalized health profile. This data is securely stored and synced in Firebase, while real-time activity tracking continuously updates the user's daily progress. The processed data is then sent to the Google Gemini AI engine, which generates tailored diet and workout recommendations. Before these suggestions reach the user, an Explainable AI layer breaks down the reasoning behind each choice, ensuring the guidance feels transparent and trustworthy. The user receives clear recommendations, understands why they are suitable, and provides feedback that the system uses to refine future suggestions—creating a dynamic, adaptive, and user-centric fitness journey.

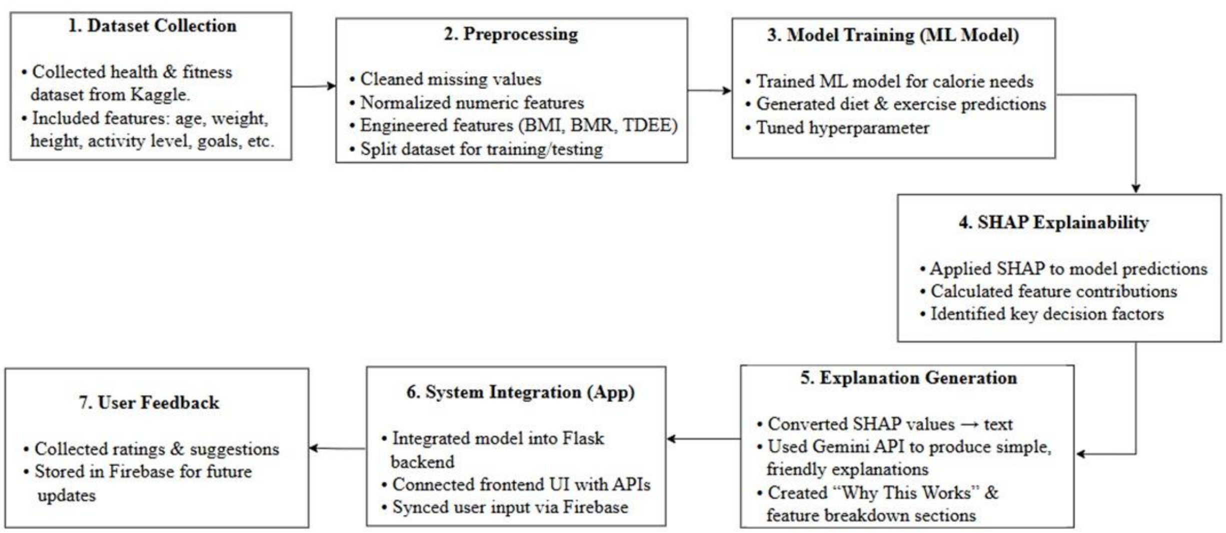

Fig 3.3.1: Methodology Diagram.

CHAPTER 4: DESIGN AND IMPLEMENTATION

4.1 SYSTEM ARCHITECTURE

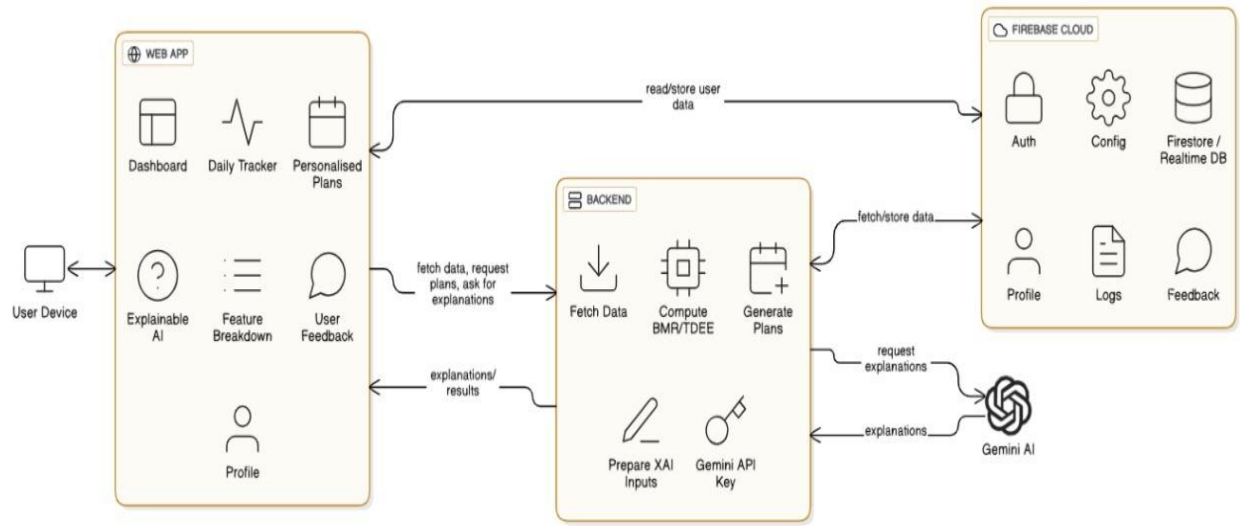

Fig 4.1.1: System architecture diagram.

The diagram represents a complete workflow connecting the User Device, Web App, Backend, and Firebase Cloud. Users interact with the web app features such as the dashboard, daily tracker, personalized plans, explainable AI, and profile management. The web app sends requests to the backend to fetch user data, compute metrics like BMR/TDEE, and generate personalized plans. The backend also prepares XAI inputs and communicates with Gemini AI to request explanations for recommendations. All user data, logs, and feedback are securely stored and retrieved from Firebase Cloud, which manages authentication and real-time database operations. The backend returns results and explanations to the web app, which displays them to the user in an understandable form. Overall, the diagram shows a smooth flow of data between the user interface, backend processing, cloud storage, and AI explanation engine.

4.2 IMPLEMENTATION (PSEUDO CODE)

1. User Authentication Flow

```
FUNCTION start_application()

LOAD environment variables
INITIALIZE Flask app
SET secret_key

INITIALIZE:
    - system = HealthFitnessXAISystem()
    - local_db = UserDatabase()
    - firebase_service = get_firebase_service()

SET use_firebase = firebase_service.is_initialized()

CALL migrate_profiles()
APPLY security headers to every response

END FUNCTION
```

2. Profile Creation and Management

```
FUNCTION create_user_profile(user_id, profile_data)
    VALIDATE profile_data (age, height, weight, etc.)
    CALCULATE BMI = weight / (height/100)²
    CALCULATE BMR using Mifflin-St Jeor equation
    CALCULATE TDEE = BMR × activity_multiplier
    STORE profile in user data
    RETURN success
```

```
END FUNCTION

FUNCTION get_user_profile(user_id)
    LOAD profile from users_db.json
    RETURN profile data
END FUNCTION
```

3. Diet Recommendation Engine

```
FUNCTION generate_diet_plan(user_profile)
    DETERMINE primary_goal (weight_loss, muscle_gain, maintenance)
    CALCULATE daily_calorie_target based on TDEE and goal
    CALCULATE macro_distribution (protein, carbs, fats)
    INITIALIZE empty meals dictionary
    FOR each meal_type in [breakfast, lunch, dinner, snacks]
        SELECT foods based on:
            - Dietary restrictions (vegetarian, vegan, gluten_free)
            - Goal-specific requirements
            - Nutritional targets
        ADD selected foods to meals[meal_type]
    END FOR
    CALCULATE total nutritional values
    GENERATE XAI explanations for recommendations
    RETURN diet_plan with meals and explanations
END FUNCTION
```

4. Exercise Recommendation Engine

```
FUNCTION generate_exercise_plan(user_profile)
    DETERMINE primary_goal (weight_loss, muscle_gain, endurance)
    ASSESS fitness_level based on BMI and activity level
    CALCULATE weekly_calorie_burn_target
    INITIALIZE weekly_plan dictionary
    FOR each day in week
        SELECT exercises based on:
            - Goal requirements
            - Fitness level
        ADD workout to weekly_plan[day]
    END FOR
    CALCULATE expected weekly_calorie_burn
    GENERATE XAI explanations for plan
    RETURN exercise_plan with schedule and explanations
END FUNCTION
```

5. Explainable AI (XAI) Component

```
FUNCTION generate_xai_explanations(user_profile, recommendations)
    CALCULATE feature_importance:
        - age_weight = age / 100 × 0.15
        - weight_weight = weight / 150 × 0.25
        - height_weight = height / 200 × 0.10
        - activity_weight = 0.30
        - goal_weight = 0.20
```

```
    GENERATE decision_factors:
        - BMI category impact
        - Activity level influence
        - Fitness goal priority
        - Sleep quality effect
    IF SHAP explainer available
        GET shap_values for predictions
        INCLUDE shap_explanations in output
    END IF
    RETURN explanations with feature importance and decision factors
END FUNCTION
```

6. Dashboard Data Processing

```
FUNCTION calculate_dashboard_stats(user_id)
    LOAD user_profile
    LOAD weekly_tracking_data
    CALCULATE daily_averages:
        - avg_steps = sum(weekly_steps) / 7
        - avg_water = sum(weekly_water) / 7
        - avg_sleep = sum(weekly_sleep) / 7
    CALCULATE progress_percentages:
        - steps_progress = current_steps / goal_steps × 100
        - water_progress = current_water / goal_water × 100
        - sleep_progress = current_sleep / goal_sleep × 100
```

```
    CALCULATE weekly_totals:
        - total_exercises_completed
        - total_calories_burned
        - current_streak_days
    RETURN stats dictionary
END FUNCTION
```

7. Meal Editor System

```
FUNCTION edit_meal(meal_name, new_foods, user_profile)
    LOAD current_diet_plan
    REPLACE current_meal[meal_name] with new_foods
    IDENTIFY subsequent_meals based on meal_order
    FOR each subsequent_meal
        ADJUST portions by 5% reduction
        ENSURE nutritional_balance maintained
    END FOR
    UPDATE diet_plan with modified meals
    RECALCULATE nutritional_totals
    GENERATE updated_XAI_explanations
    RETURN updated_plan
END FUNCTION
```

8. Data Persistence

FUNCTION save_user_data(user_email, profile_data)

```
IF user_email NOT in session
```

```
    RETURN error "User not logged in"

CALL local_db.update_user_profile(user_email, profile_data)

IF firebase_is_enabled
    CALL firebase.update_user_profile(user_email, profile_data)

RETURN success

END FUNCTION

 FUNCTION load_user_data(user_email)

IF user_email NOT in session
    RETURN error "User not logged in"

profile = local_db.get_user_profile(user_email)

IF profile EXISTS
    RETURN profile

IF firebase_is_enabled
    profile = firebase.get_user_profile(user_email)
    IF profile EXISTS
        CALL local_db.update_user_profile(user_email, profile)
        RETURN profile

RETURN empty_profile_structure

END FUNCTION

FUNCTION sync_tracking_data(user_email, tracking_entry)
```

CALL DailyTracker(user_email).save(tracking_entry)

IF firebase_is_enabled

   CALL firebase.store_tracking_data(user_email, tracking_entry)

RETURN success

END FUNCTION

9. Web Server Routes

ROUTE /login

DISPLAY login.html

IF request is POST
   CALL login_user()
   PROCESS login result

ROUTE /signup

DISPLAY signup.html
ROUTE /dashboard

IF user_authenticated
   DISPLAY dashboard.html with user statistics
ELSE
   REDIRECT to /login
ROUTE /

IF user_authenticated
   IF user_profile_exists
      CALL generate_recommendations()

```
        DISPLAY index.html with recommended plan
    ELSE
        DISPLAY profile_creation_form
ELSE
    REDIRECT to /login
ROUTE /api/create_profile

RECEIVE profile_data from POST request
CALL create_user_profile(profile_data)
RETURN success or failure result
ROUTE /api/get_recommendations

CALL generate_diet_plan()
CALL generate_exercise_plan()
RETURN combined recommendations to caller
```

4.3 IMPLEMENTATION DETAILS

The system is developed using **Flask** with **Python** as the backend framework, providing the core logic for processing user requests and managing the interaction between system components. A clean and responsive interface is created using **HTML, CSS, and JavaScript**, ensuring smooth navigation and an intuitive user experience. All backend operations are executed through well-defined Flask routes, which interact with **JSON file storage** to save and retrieve user information, health data, and analysis outputs efficiently.

AI-driven functionalities—including **personalized diet recommendations, exercise planning, and explainable AI (XAI) insights**—are implemented using the **SHAP** libraries. When users input their health metrics, the system processes the data through these machine learning models to generate customized health guidance. The SHAP module further enhances transparency by explaining how each input feature contributes to the recommendations.

To ensure secure access control, the system utilizes **Flask session management**, which handles login and signup processes while protecting user credentials. Additionally, sleep tracking and physical activity details are continuously stored in JSON files, allowing the system to deliver more accurate and improved recommendations over time.

The application is deployed and executed locally, making it lightweight and easy to run without complex server setups. All sensitive configurations, including secret keys and environment variables, are stored securely within the application settings to maintain data protection and system reliability. Overall, this implementation approach ensures a robust, scalable, and user-friendly health recommendation platform.

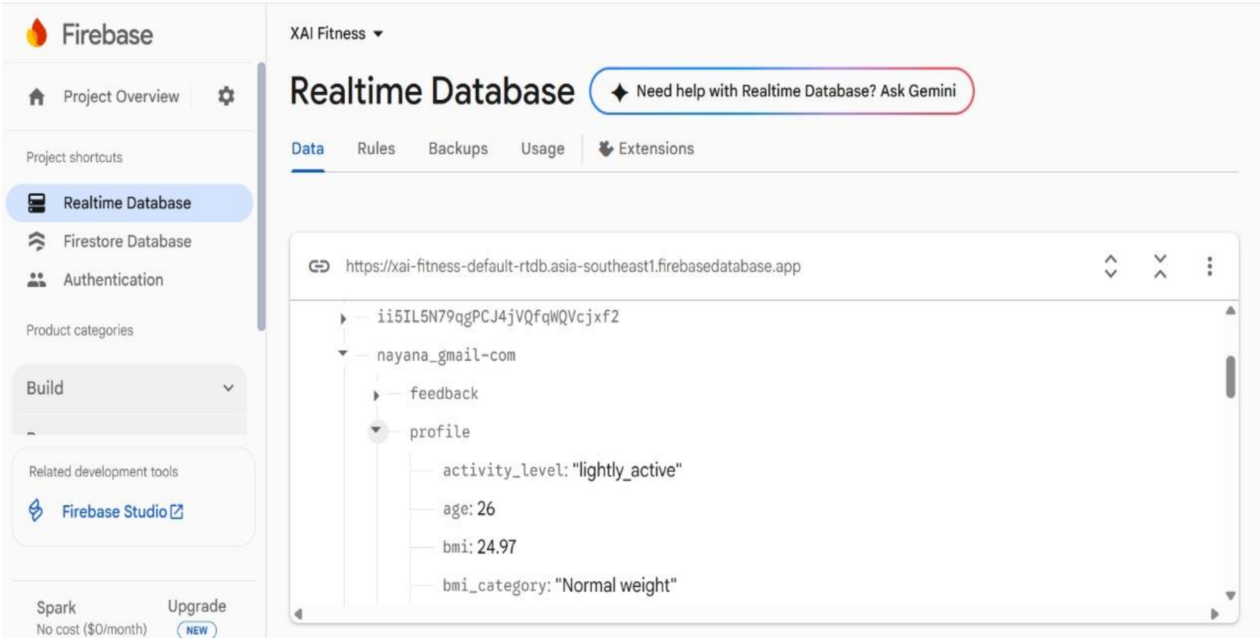

Fig 4.3.1: Realtime Database

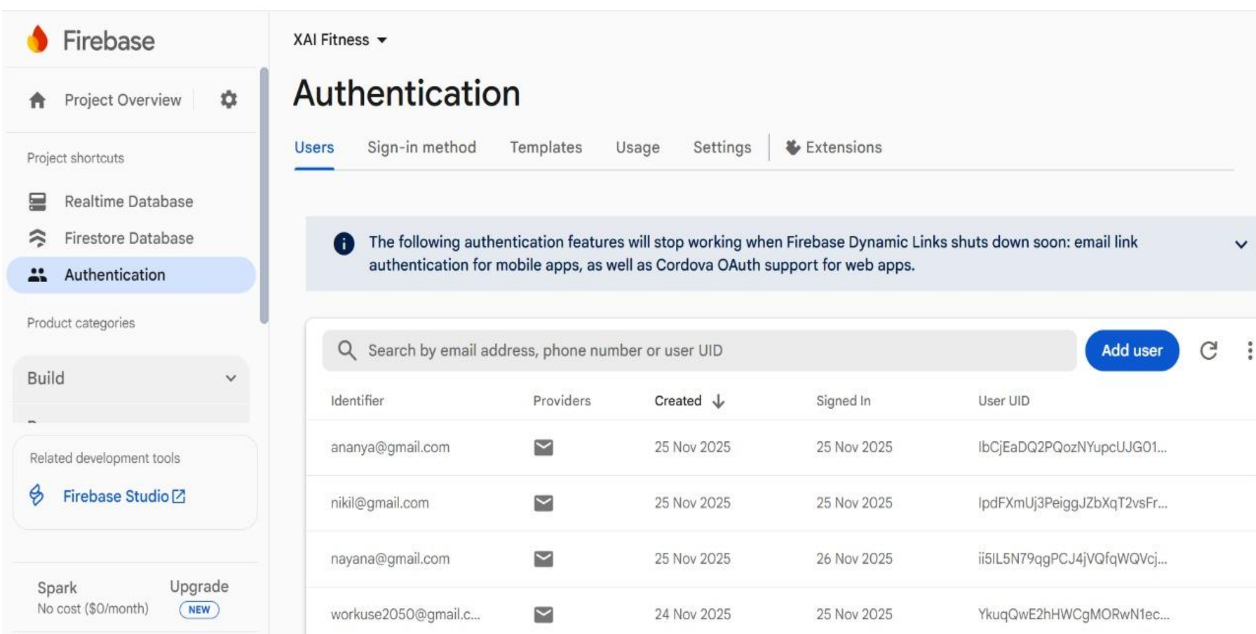

Fig 4.3.2: Authentication

CHAPTER 5: RESULTS

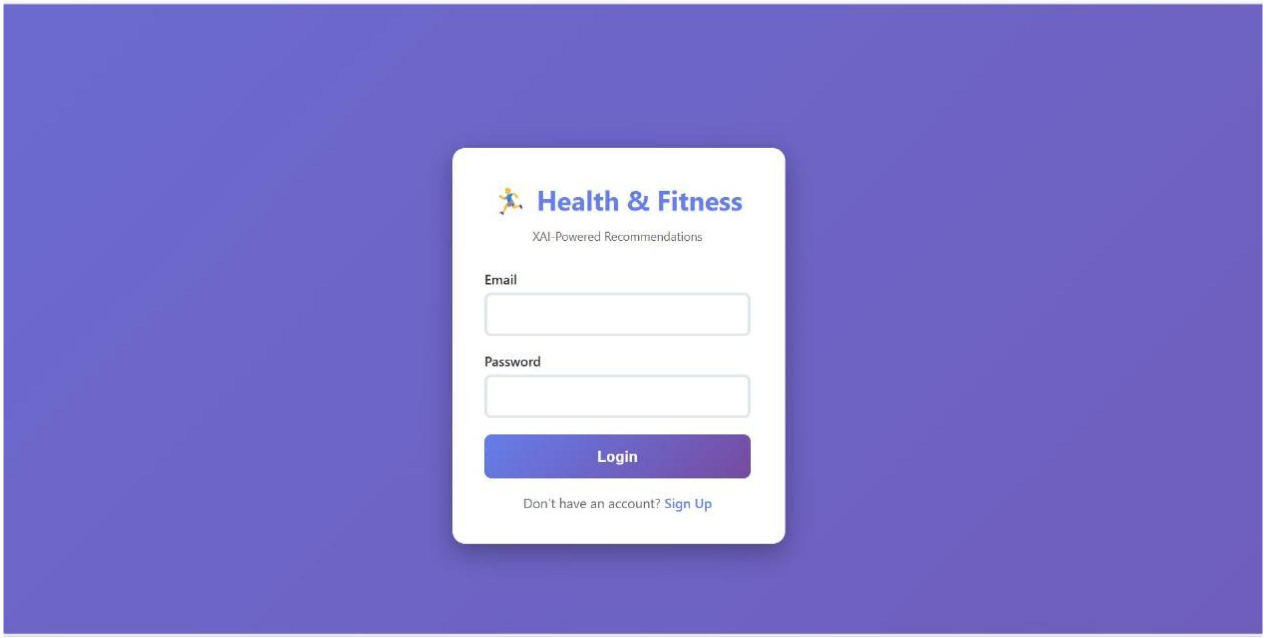

Fig 5.1: Login/Sing up page

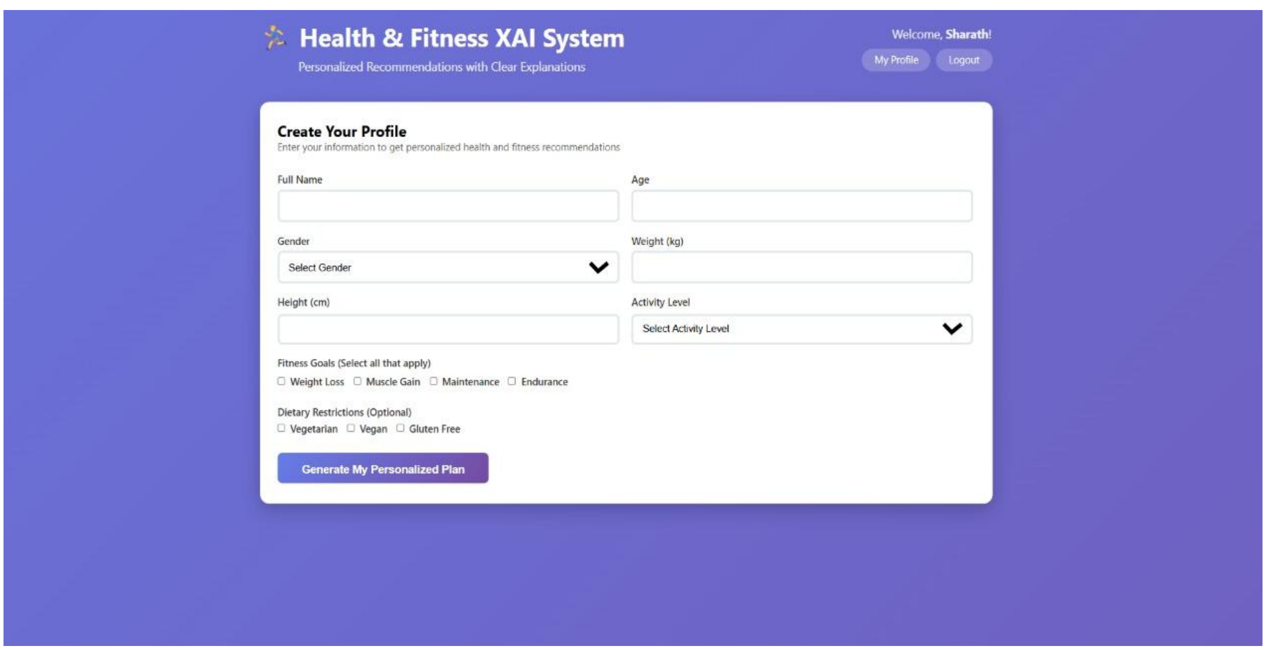

Fig 5.2: Create Profile page

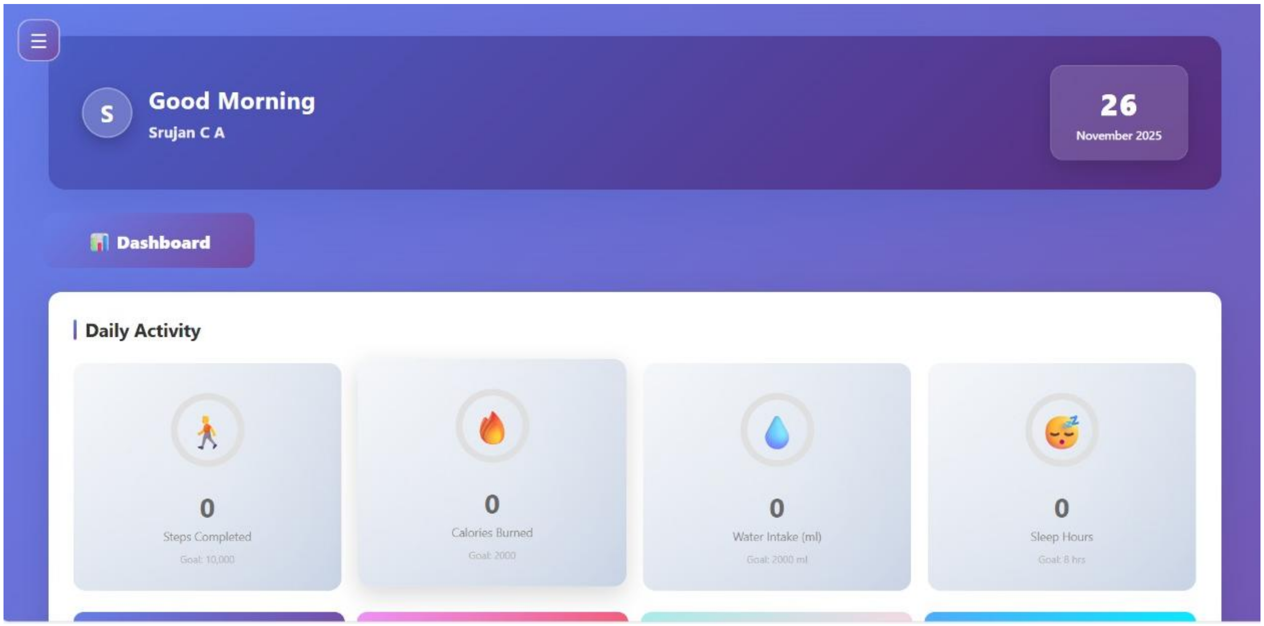

Fig 5.3: Dashboard

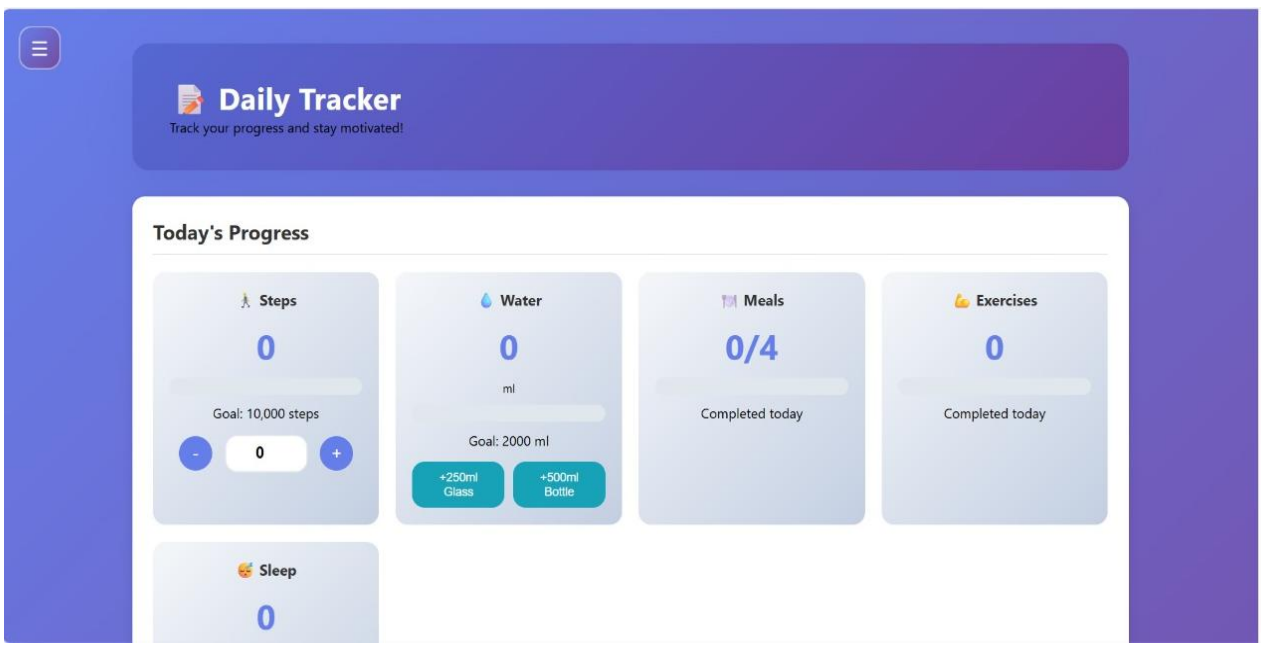

Fig 5.4 : Daily tracker

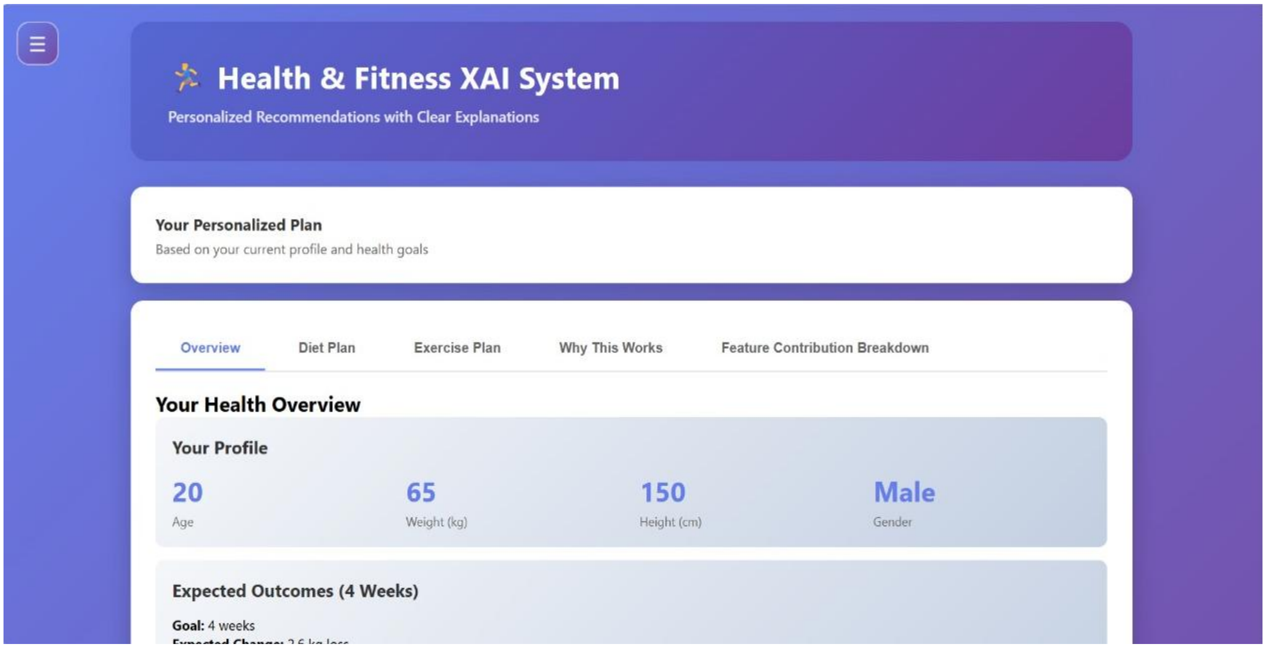

Fig 5.5: Personalized Recommendations with Clear Explanations

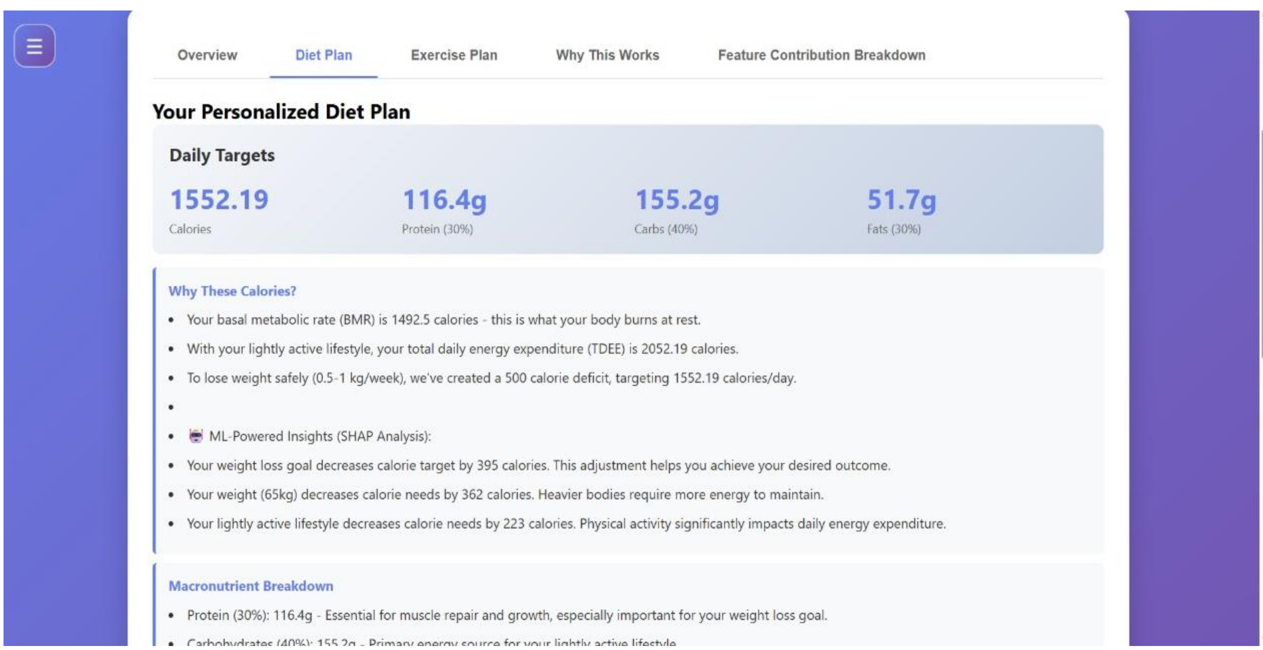

Fig 5.6: Diet Plan

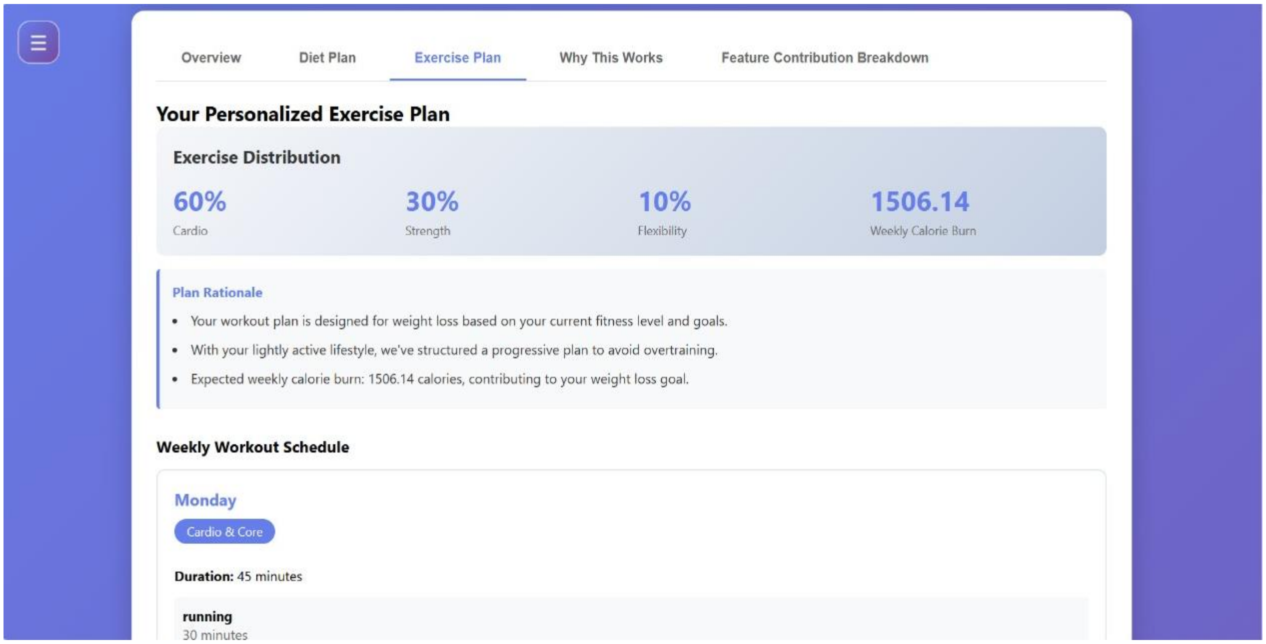

Fig 5.7: Exercise Plan

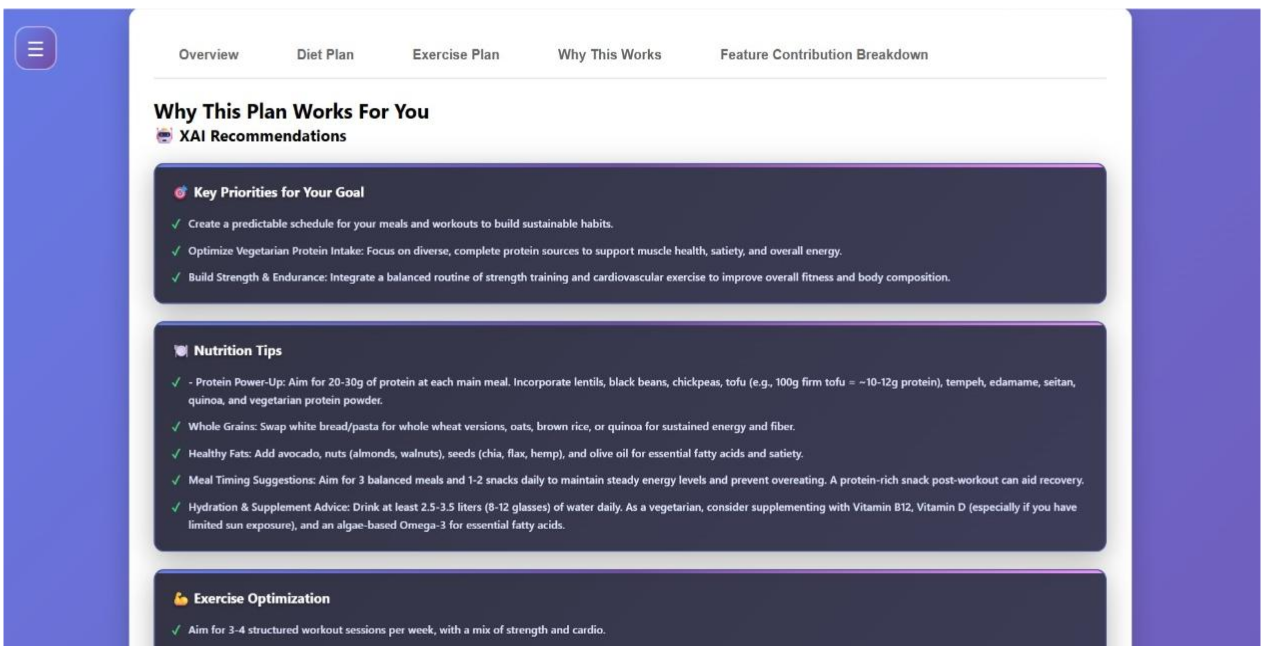

Fig 5.8: Why this works

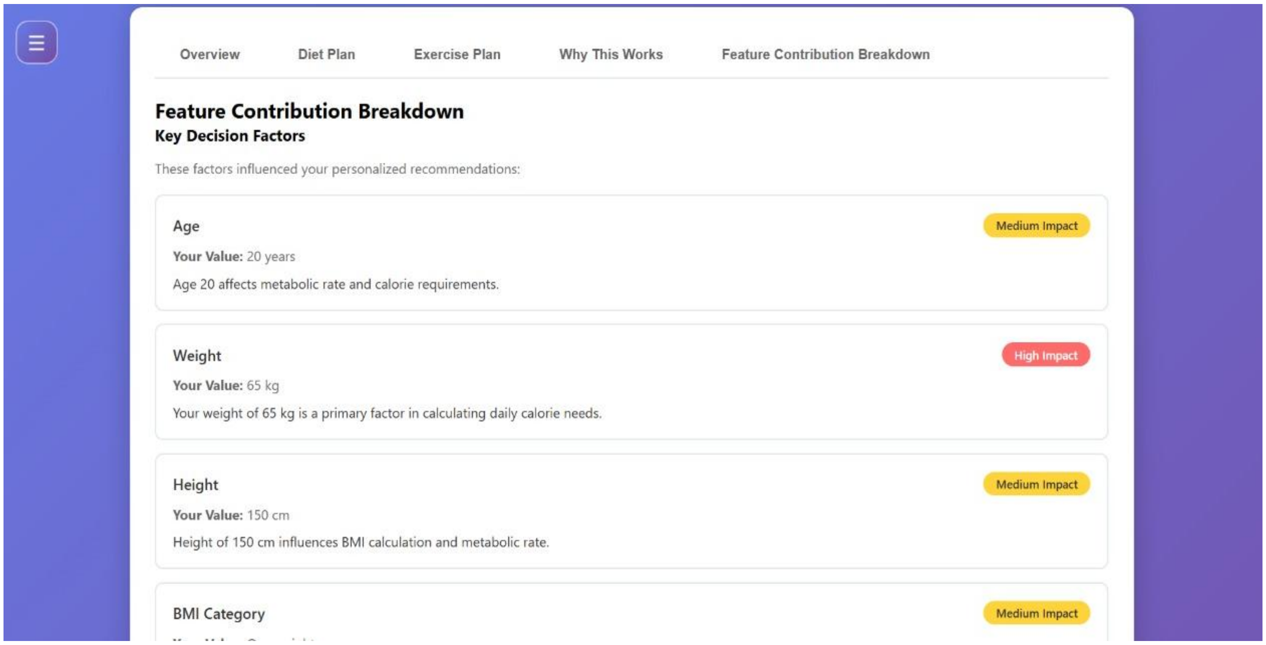

Fig 5.9: Feature contribution breakdown

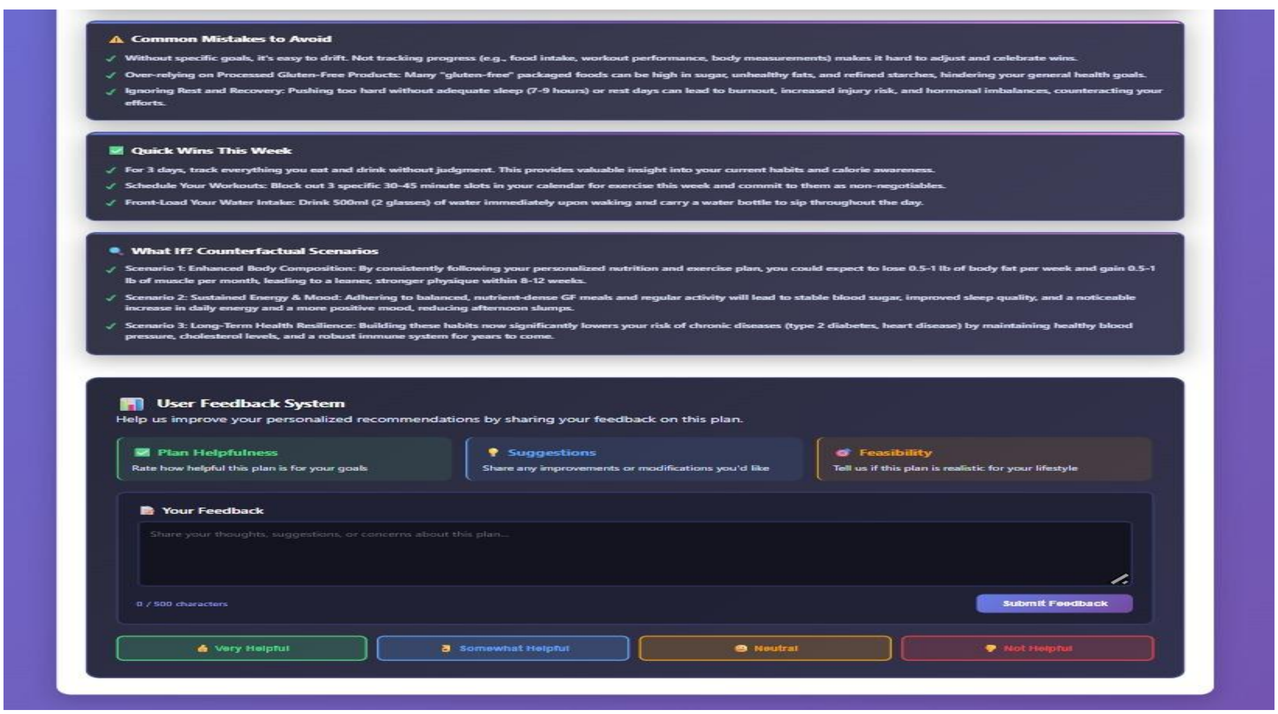

Fig 5.10: User Feedback System

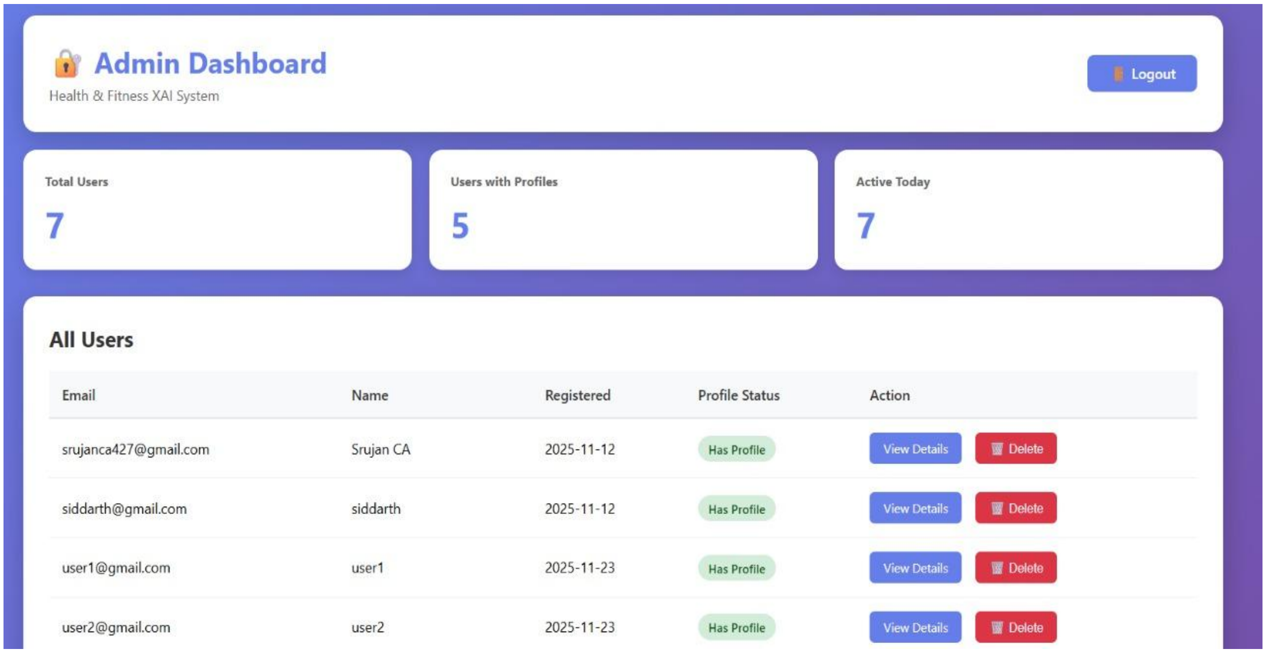

Fig 5.11: Admin Dashboard