**Fitnessstan**

****

**By:**

**Zain Ul Abideen**

**35515**

**Obaid Ullah**

**35739**

**Huzaifa Khan**

**35726**

**Supervised by:**

**M. Islam Abbasi (Lecturer)**

**Faculty of Computing**

**Riphah International University, Islamabad**

**Spring/Fall 2024**

**A Dissertation Submitted To**

**Faculty of Computing,**

**Riphah International University, Islamabad**

**As a Partial Fulfillment of the Requirement for the Award of the Degree of**

**Bachelors of Science in Computer Science**

**Faculty of Computing**

**Riphah International University, Islamabad**

Date: [date of final presentation]

**Final Approval**

This is to certify that we have read the report submitted by ***Zain Ul Abideen (35515)*, *Obaid Ullah (35739), Huzaifa Khan (35726)*** for the partial fulfillment of the requirements for the degree of the Bachelors of Science in Computer Science (BSSE). It is our judgment that this report is of sufficient standard to warrant its acceptance by Riphah International University, Islamabad for the degree of Bachelors of Science in Computer Science (BsCs).

**Committee:**

|  |  |
| --- | --- |
| **1** | M. Islam Abbasi  (Supervisor) |
|  |  |

|  |  |
| --- | --- |
| **2** | Dr. Musharraf Ahmed  (Head of Department) |

**Declaration**

We hereby declare that this document “**Fitnessstan**” neither as a whole nor as a part has been copied out from any source. It is further declared that we have done this project with the accompanied report entirely on the basis of our personal efforts, under the proficient guidance of our teachers, especially our supervisor **Muhammad Islam Abbasi**. If any part of the system is proved to be copied out from any source or found to be reproduction of any project from anywhere else, we shall stand by the consequences.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Zain Ul Abideen**

**35515**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Obaid Ullah**

**35739**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Huzaifa Khan**

**35726**

**Dedication**

This work is devoted to Almighty Allah, the Most Merciful and the Most Beneficent, who granted us the knowledge, perseverance, and strength to complete this work. We also wish to extend our deepest gratitude to our parents for their unyielding love, support, and prayers along the journey. We finally dedicate this to our supervisor, “Muhammad Islam Abbasi”, with whose valuable guidance, mentorship, and encouragement we are able to make this project into reality.

**Acknowledgement**

We are grateful to Almighty Allah, the provider of all knowledge and wisdom, for giving us the strength and perseverance to bring this project to fruition.

We would like to extend our wholehearted gratitude to our supervisor, “Muhammad Islam Abbasi” who guided and encouraged us on this endeavor throughout the course of this project. His mentorship has been the backbone of our learning and growth.

We would also like to acknowledge the encouragement and constructive criticism offered by our faculty members and colleagues who guided us to correct our mistakes. Their encouraging words pushed us beyond our limits of apprehension. Last but not least, we express our profound appreciation to our families and friends for their exemplary support, prayers, and patience that provided us with strength in pursuit of excellence.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Zain Ul Abideen**

**35515**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Obaid Ullah**

**35739**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Huzaifa Khan**

**35726**

**Abstract**

In today's fast-paced world, people work tirelessly with dedication but often neglect their health and fitness, leading to a steady rise in the number of unhealthy individuals. Addressing this concern, we aim to develop a comprehensive platform, available as both a website and a mobile application, to help users manage their daily exercise and food intake effectively.

Existing apps and websites in Pakistan often fall short in providing personalized food recommendations tailored to users’ needs. While some platforms suggest exercise routines, they rarely focus on balanced nutrition or calorie management, which are essential for maintaining overall health. Moreover, these platforms lack customization based on user preferences, leaving a gap in truly effective fitness solutions.

Our solution, Fitnessstan, bridges this gap by offering personalized food and exercise recommendations. Using Flutter technology, we developed a user-friendly app that integrates an AI model to suggest appropriate calorie intake and workouts based on individual requirements. For users preferring web access, we also created a responsive website using React.js.

Fitnessstan is designed to empower users with tailored guidance, promoting healthier lifestyles through a combination of technology, innovation, and convenience. With this platform, we strive to make fitness accessible and achievable for everyone.

**Table of Content**

[Table of Contents **Error! Bookmark not defined.**](#_Toc113957278)

[List of Tables **Error! Bookmark not defined.**](#_Toc113957279)

[List of Figures **Error! Bookmark not defined.**](#_Toc113957280)

[Abstract **Error! Bookmark not defined.**](#_Toc113957281)

[Chapter 1: Introduction **Error! Bookmark not defined.**](#_Toc113957282)

[1.1 Goals and Objectives 13](#_Toc113957283)

[1.2 Scope of the Project **Error! Bookmark not defined.**](#_Toc113957284)

[Chapter 2: Literature Review **Error! Bookmark not defined.**](#_Toc113957285)

[2.1 Introduction **Error! Bookmark not defined.**](#_Toc113957286)

[2.2 Background and Problem Elaboration **Error! Bookmark not defined.**](#_Toc113957287)

[2.3 Detailed Literature Review **Error! Bookmark not defined.**](#_Toc113957288)

[2.3.1 Definitions **Error! Bookmark not defined.**](#_Toc113957289)

[2.3.2 Related Research Work 1 **Error! Bookmark not defined.**](#_Toc113957290)

[2.3.3 Related Research Work 2 **Error! Bookmark not defined.**](#_Toc113957291)

[2.4 Literature Review Summary Table **Error! Bookmark not defined.**](#_Toc113957292)

[2.5 Research Gap **Error! Bookmark not defined.**](#_Toc113957293)

[2.6 Problem Statement **Error! Bookmark not defined.**](#_Toc113957294)

[Chapter 3: Requirements and Design **Error! Bookmark not defined.**](#_Toc113957295)

[3.1 Requirements **Error! Bookmark not defined.**](#_Toc113957296)

[3.1.1 Functional Requirements **Error! Bookmark not defined.**](#_Toc113957297)

[3.1.2 Non-Functional Requirements **Error! Bookmark not defined.**](#_Toc113957298)

[3.1.3 Hardware and Software Requirements **Error! Bookmark not defined.**](#_Toc113957299)

[3.2 Proposed Methodology **Error! Bookmark not defined.**](#_Toc113957300)

[3.3 System Architecture **Error! Bookmark not defined.**](#_Toc113957301)

[3.4 Use Cases **Error! Bookmark not defined.**](#_Toc113957302)

[3.4.1 Sample Use Case Name Here **Error! Bookmark not defined.**](#_Toc113957303)

[3.5 Database Design *(Optional)* **Error! Bookmark not defined.**](#_Toc113957304)

[3.6 Class diagram *(Optional)* **Error! Bookmark not defined.**](#_Toc113957305)

[3.7 Sequence diagram *(Optional)* **Error! Bookmark not defined.**](#_Toc113957306)

[3.8 Any Other Artifact… **Error! Bookmark not defined.**](#_Toc113957307)

[3.9 Graphical User Interfaces(GUI) (*Optional)* **Error! Bookmark not defined.**](#_Toc113957308)

[Chapter 4: Implementation and Test Cases **Error! Bookmark not defined.**](#_Toc113957309)

[4.1 Implementation **Error! Bookmark not defined.**](#_Toc113957310)

[4.1.1 Implementation of First Component/Algorithm **Error! Bookmark not defined.**](#_Toc113957311)

[4.2 Test case Design and description **Error! Bookmark not defined.**](#_Toc113957312)

[4.2.1 Sample Test case No.1 **Error! Bookmark not defined.**](#_Toc113957313)

[4.2.2 Sample Test case No.2 **Error! Bookmark not defined.**](#_Toc113957314)

[4.3 Test Metrics **Error! Bookmark not defined.**](#_Toc113957315)

[4.3.1 Sample Test case Matric.No.1 **Error! Bookmark not defined.**](#_Toc113957316)

[4.3.2 Sample Test case Metric.No.2 **Error! Bookmark not defined.**](#_Toc113957317)

[4.3.3 Sample Test case Metric.No.3 **Error! Bookmark not defined.**](#_Toc113957318)

[Chapter 5: Experimental Results and Analysis **Error! Bookmark not defined.**](#_Toc113957319)

[Chapter 6: Conclusion and Future Directions **Error! Bookmark not defined.**](#_Toc113957320)

[References **Error! Bookmark not defined.**](#_Toc113957321)

[Appendix **Error! Bookmark not defined.**](#_Toc113957322)

[Appendix A: Guidelines **Error! Bookmark not defined.**](#_Toc113957323)

[Appendix B: Heading of Sample Appendix B **Error! Bookmark not defined.**](#_Toc113957324)

[Formatting Guidelines **Error! Bookmark not defined.**](#_Toc113957325)

[Chapter 1: Heading 1 **Error! Bookmark not defined.**](#_Toc113957326)

[1.1 Heading 2 **Error! Bookmark not defined.**](#_Toc113957327)

[1.1.1 Heading 3 **Error! Bookmark not defined.**](#_Toc113957328)

[Tables and Figures **Error! Bookmark not defined.**](#_Toc113957329)

[Equations **Error! Bookmark not defined.**](#_Toc113957330)

[Header/Footer **Error! Bookmark not defined.**](#_Toc113957331)

[Other Formatting Guidelines **Error! Bookmark not defined.**](#_Toc113957332)

[References **Error! Bookmark not defined.**](#_Toc113957333)

**List of Tables**

**List of Figure**

**Chapter 1:**

**Introduction**

**Chapter 1: Introduction**

**Fitnessstan** is a web-based and app-based platform designed to provide a holistic solution for fitness enthusiasts, including individuals with diabetes. In today’s world, achieving a healthy lifestyle can be challenging, especially for beginners who lack personal guidance on balancing exercise and nutrition. Neglecting either aspect often leads to unsatisfactory progress or health risks. Existing platforms fail to offer comprehensive and personalized solutions, particularly for users with specific health needs.

Fitnessstan addresses these challenges by integrating artificial intelligence to deliver customized workout and dietary plans tailored to each user’s unique health metrics, goals, and preferences. The platform specifically supports diabetic patients by incorporating features that help manage their condition through personalized exercise and nutrition guidance. This seamless integration of fitness and health ensures safe and effective progress for all users.

With features like progress tracking and adaptive recommendations, Fitnessstan ensures long-term results and user satisfaction. Whether its weight loss, muscle gain, or managing conditions like diabetes, Fitnessstan empowers individuals to take control of their health journey while promoting a culture of sustainable fitness and well-being.

* 1. **Goals and Objectives**
     1. **Goals**

1. The platform in which Provide a personalized and user-friendly fitness platform for enthusiasts.
2. To facilitate users achieve and maintain their fitness goals through AI-powered workout and dietary recommendations.
   * 1. **Objectives**
3. To offer fitness workout plans as per users' fitness levels, preferences, and health metrics.
4. To provide dietary recommendations personalized to the specific needs of diabetic patients and general users, respectively.
5. To enable users to track progress and result analysis in measurable improvements over time.
   1. **Scope of the Project**

The scopes of the **“Fitnessstan”** are as following

1. Our website will be developed on java stack development and our app will be developed on Flutter.
2. We will develop a website and app that will provide platform for diabetic patient and for those people who loss, gain and maintain their weight.
3. User will upload their information without any hesitation because we are securing his information.
4. User will get personalized diet plan on the basis of his information.
5. Problem will be solved with Machine learning techniques.

The platform would also be accessible, interactive and user-friendly, and simplified for Users.

**Chapter 2:**

**Literature Review**

**Chapter 2: Literature Review**

* 1. **Introduction**

**Fitnessstan** is a web-based and app-based platform designed to provide a holistic solution for fitness enthusiasts, including individuals with diabetes. In today’s world, achieving a healthy lifestyle can be challenging, especially for beginners who lack personal guidance on balancing exercise and nutrition. Neglecting either aspect often leads to unsatisfactory progress or health risks. Existing platforms fail to offer comprehensive and personalized solutions, particularly for users with specific health needs.

Fitnessstan addresses these challenges by integrating artificial intelligence to deliver customized workout and dietary plans tailored to each user’s unique health metrics, goals, and preferences. The platform specifically supports diabetic patients by incorporating features that help manage their condition through personalized exercise and nutrition guidance. This seamless integration of fitness and health ensures safe and effective progress for all users.

With features like progress tracking and adaptive recommendations, Fitnessstan ensures long-term results and user satisfaction. Whether its weight loss, muscle gain, or managing conditions like diabetes, Fitnessstan empowers individuals to take control of their health journey while promoting a culture of sustainable fitness and well-being.

* 1. **Background and Problem Elaboration**

While the industry of fitness is rapidly expanding, on many platforms users, especially first-time users, end up stuck in their goals due to the fact that most do not properly guide one both through exercise and nutrition. Most of them only give workout plans without connecting the critical role of nutrition for achieving fitness goals. This leads to less useful results, bad recovery, and potential health risks.

Also, many of these applications do not support different types of constraints, such as diabetes, and provide no full, personalized recommendations. There is also disconnection between guidance provided on exercise and nutrition, making the whole experience disjointed and giving the user no tools needed to be successful in the long term.

Fitnessstan offers a remedy to that problem by introducing AI-based online services combining customized workout routines, suggested diets, and monitoring to create one all-inclusive solution where every user will have the right resources for a holistic fitness journey.

* 1. **Detailed Literature Review**
     + - 1. **Definition**

Fitness refers to a state of health and well-being that allows an individual to perform daily tasks with energy and minimal fatigue. It has physical, mental, and emotional factors to it and, thus, outlines the importance of good regular exercise and lifestyle habits.

A diet, on the other hand, is a short-term measure that incorporates certain food-related restrictions designed to fit the requirements for achieving objectives like weight loss or improvement in the state of being healthy. Many diets are temporary and specific to the client's need.

On the other hand, a nutrition plan is generally a long-term approach or an eating regimen designed to maintain the body's necessary macronutrient and micronutrient inputs by proper consumption to achieve certain health and fitness goals.

* + - * 1. **Related Research Work 1**
        2. **Related Research Work 2**
  1. **Literature Review Summary Table**

**Table 2.1:** Summary of Research Paper

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No.** | **Document** | **Domain** | **Algorithm** |
| 1 | AI Fitness Trainer Applications | Fitness Tracking and Personalization | BlazePose for pose detection, CNN for image analysis, RNN for sequence data |
| 2 | Virtual Fitness Trainer Using AI | Virtual Coaching and Training | MediaPipe for motion tracking, Recurrent Neural Networks (RNN) |
| 3 | Personalized Nutrition and Fitness Platforms | Diet and Exercise Personalization | Machine Learning for meal planning, Decision Trees for analysis |
| 4 | AI-Based Quantification of Fitness Activities | Real-Time Exercise and Health Monitoring | BlazePose for pose estimation, Random Forest for movement tracking |
| 5 | Fitness Trainer Application Using Artificial Intelligence | Fitness and Health Solutions for Diabetics | Linear Regression for calorie estimation, SVM for dietary recommendations |
| 6 | Virtual Personal Trainer Platforms | Personalized Fitness Guidance | BlazePose for motion capture, Decision Trees for user customi AI Fitness Trainer Applications zation |
| 7 | AI-Based Workout Recognition Systems | Real-Time Motion and Workout Recognition | CNN for video analysis, Naive Bayes for feedback analysis |
| 8 | AI Fitness Systems for Progressive Workouts | Progress Tracking and Exercise Recommendations | K-Nearest Neighbors (KNN), Deep Neural Networks (DNN) |
| 9 | Virtual Fitness Trainer Platforms for Cardio | Specific Activity Recognition | MediaPipe for activity tracking, SVM for cardio analysis |
| 10 | Advanced AI Fitness Platforms | Integrated Fitness and Nutrition | Random Forest for data analysis, Linear Regression for diet optimization |

* 1. **Problem Statement**

The fitness industry faces significant challenges, particularly for beginners who are dedicated to workouts but lack awareness of the critical role nutrition plays in achieving their goals. This neglect of proper diet not only slows their progress but also poses health risks, including poor recovery, nutritional deficiencies, and the exacerbation of conditions like obesity and diabetes. Additionally, many individuals struggle with tailored strategies to effectively lose weight, gain muscle, or maintain their current fitness levels. Without a holistic approach that integrates both exercise and personalized nutrition, these issues continue to undermine the overall success and well-being of fitness enthusiasts.

**Chapter 3:**

**Requirements and Design**

**Chapter 3: Requirements and Design**

In this chapter, we have developed the functional requirements for our actors, i.e., **User** and **Admin**. These requirements are specifically designed for the Fitnessstan platform.

**Fitnessstan** is both a web-based and app-based platform that offers personalized fitness solutions by integrating AI-powered workout and dietary recommendations. The platform is user-friendly, easy to navigate, and provides an efficient way for users to achieve their fitness goals and for admins to manage the platform's functionality seamlessly.

We created system use cases against each functional requirement and designed use case diagrams and fully dressed use cases for our actors, i.e., **User** and **Admin**.

1. **Requirements**
   1. **User/Customer Functionalities:**

**Table 3.1: User Functional Requirement**

|  |  |
| --- | --- |
| **S.no** | **Functionality Description** |
| 1 | The user must be able to sign up. |
| 2 | The user must be able to login to their account. |
| 3 | The user must be able to recover and forget their password |
| 4 | Input personal information, like DOB, weight, height, gender, religion, sleep-hour, occupation and exercise level, to compute BMI. |
| 5 | Diet recommendations that are specifically tailored to the needs of diabetics. |
| 6 | Monitoring of progress - calories burned, workouts performed, and nutritional intake. |
| 7 | User will be read the article about features and supplements. |
| 8 | User shall be able to give feedback about our recommendation. |
| 9 | Weekly updates and alerts on progress and fitness plan. |

* 1. **Admin Functionalities:**

**Table 3.2: User Functional Requirement**

|  |  |
| --- | --- |
| **ID** | **Functionality Description** |
| 1 | Admin must be able to login to the system |
| 2 | Administrators will be able to add users. |
| 3 | Admin must be able to delete the users. |
| 4 | Admin must be able to Manages user by adding, updating, or deleting accounts. |
| 5 | Admin must be reviewed the feedback given by user. |
| 6 | Monitor the platform’s overall performance and usage statistics. |
| 7 | Provide data security and impose privacy policies. |
| 8 | Manage system configurations and updates and Database management. |
| 9 | Resolve technical issues and provide platform support. |

1. **Non-Functional Requirements**
2. **Performance:** The application should respond to the user's inputs within 1 second to provide a smooth experience.
3. **Scalability:** The system must be able to support concurrent use by different users without considerable performance degradation.
4. **Security:** All user data, personal and health-related, should be encrypted for confidentiality and integrity.
5. **Usability:** The interface must be intuitive and accessible to users with different technical
6. **Reliability:** The platform should maintain at least 99% uptime to ensure that the application is always available.
7. **Compatibility:** The application should seamlessly run on both Android and iOS devices and be accessible through web browsers
8. **Hardware and Software Requirements**
   1. **Hardware Requirement**
9. **Server:**  
   A dedicated server or cloud hosting service to host the website and app, ensuring efficient handling of user requests, AI processing, and data storage.
10. **Storage:**  
    Adequate storage capacity to store user profiles, workout and dietary data, progress reports, and AI-generated recommendations.
11. **Processing Power:**  
    Sufficient processing power to manage concurrent user requests, perform AI computations for personalized fitness plans, and handle real-time data tracking and updates.
    1. **Software Requirement**
12. **Operating System:**

The server should run a compatible operating system such as Linux, windows server or macOS server.

1. **Flutter:**

It is used for cross-platform mobile application development, which allows smooth functionality on both Android and iOS devices.

1. **React.js:**

It is a JavaScript library used for creating responsive and dynamic web interfaces.

1. **Spring Boot and Java:**

These are used for back-end development to ensure scalable and robust server-side functionality.

1. **Python:**

It is used for AI/ML tasks such as generating personalized fitness and dietary recommendations.

1. **MongoDB:**

It is a NoSQL database for managing user data, activity logs, and app-related information.

1. **Bootstrap:**

Used for creating beautiful and responsive user interfaces.

1. **VS Code:**

Primary IDE for coding and debugging.

1. **Git:**

A version control system to collaborate on development and manage source code.

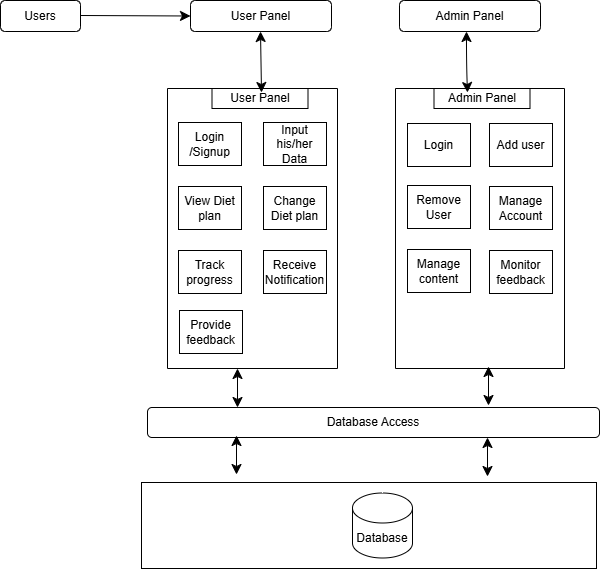
1. **Proposed Methodology**

Our solution focuses on providing personalized diet plans by leveraging advanced data clustering techniques and synthetic user data generation. To achieve this, we utilize unsupervised learning to cluster food items based on their nutritional values. This process categorizes foods into 14 distinct clusters, each representing specific nutritional characteristics, such as high protein and low fat, or high fiber and moderate carbohydrates. These clusters provide a structured representation of the nutritional profiles of various foods.

To align the dietary recommendations with individual needs, we create a synthetic dataset that simulates user information, including health metrics, dietary preferences, and fitness goals. This synthetic dataset enables the development and testing of personalized diet plans by mapping user requirements to the appropriate food clusters.

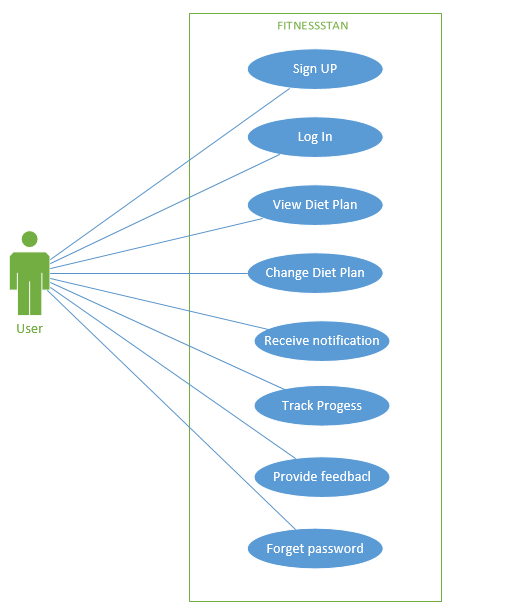
By combining nutritional clustering and synthetic user data, our approach ensures tailored, data-driven diet plans that cater to diverse dietary needs and health objectives, promoting sustainable and effective nutrition management.

1. **System Architecture**

****

**Figure 3.1: System architecture diagram**

1. **Use Cases**

* + 1. **User Use-Case**

**Figure 3.2: User use case diagram**

* + 1. **Admin Use-Case**

****

**Figure 3.3: Admin use-case diagram**

1. **Fully-Dressed Use Cases**
   * 1. **Login**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | | Login | | |
| Actors | | Admin,User | | |
| Summary | | The user must input their email address and password on the login form, following which they will be forwarded to the home page. | | |
| Pre-Conditions | | The user must be registered in to the system. | | |
| Post-Conditions | | The user will be sent to the system’s home page, and his or her session will be preserved. | | |
| Special Requirement | | Ensure password is not visible during verification. | | |
| **Basic Flow** | | | | |
| Actor Action | | | System Response | |
| 1 | The user navigates to the login page. | | 2 | The login page appears and asking for email address and password. |
| 3 | The user provide a valid email and password. | | 4 | The system verify the email and password, establishes a session and redirect to the appropriate dashboard. |
| **Alternative Flow** | | | | |
| A3 | The user enters invalid email and password. | |  | The system displays the following error message: incorrect email address and password. |

**Table 3.1: Login Fully Dressed Use-case**

* + 1. **Sign Up**

**Table 3.2: Sign-up Fully Dressed Use-case**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | | Sign Up | | |
| Actors | | Admin,User | | |
| Summary | | The user register a new account by entering the required information. | | |
| Pre-Conditions | | The user must be registered in to the system. | | |
| Post-Conditions | | New account is created and user redirect to the login page for authentication. | | |
| Special Requirement | | Validate email format and check the duplication of email | | |
| **Basic Flow** | | | | |
| Actor Action | | | System Response | |
| 1 | The user visit the sign-up page. | | 2 | The user is prompted to provide details like their name, email address and password on the sign-up page |
| 3 | The user provide their name, email address and password. | |  |  |
| 4 | The user fills out the sign-up form | | 5 | The system validate the information and send OTP to the user email. |
| 6 | The user put the OTP on the sign-up page. | | 7 | The system validate the OTP and generates a new account for user. |
|  |  | | 8 | The system display a success message, indicating that user account was successfully created. |
| **Alternative Flow** | | | | |
| A3 | If the user submit the sign-up form with incomplete or invalid information. | |  | The system display and error message, indicating specific fields that need to be corrected. |
| A6 | If the user enters invalid OTP. | |  | The system display the error message indicating you OTP is incorrect. |

**3.5.3 Recover password**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | | Recover password | | |
| Actors | | Admin,User | | |
| Summary | | The admin and user can recover their password who has forgotten or lost their login credentials. | | |
| Pre-Conditions | | The user must have FITNESSSTAN account and user must have access to email address which is associated with his account. | | |
| Post-Conditions | | The user password is successfully reset and they can log in to their account using the new password. | | |
| Special Requirement | | The reset process must be secure to prevent the unauthorized access. | | |
| **Basic Flow** | | | | |
| Actor Action | | | System Response | |
| 1 | The user navigates to the login page. | |  |  |
| 2 | User click on the “Forgot Password” button. | | 3 | A password recovery form is displayed by the system. |
| 4 | In the recovery form user enter the email which is associated with their account. | |  |  |
| 5 | The user presses the “submit” button | | 6 | The system validate the email address and create the password reset link. |
|  |  | | 7 | The system send the reset link on the email which is associated with their account. |
| 8 | The user open their email and clicks on the password reset link. | | 9 | The system redirect the user to a password reset form. |
| 10 | In the prompted field the user enters a new password and verifies it. | |  |  |
| 11 | The user fill out the password reset form. | | 12 | The new password is validated by the system and added to the user account in database. |
|  |  | | 13 | When the password is successfully reset, the system display a confirmation message. |
| **Alternative Flow** | | | | |
| A4 | If a user provide an incorrect or unregistered email address. | |  | The system prompt the user to input a valid email address which is associated with their account. |
|  |  | |  | If the password reset link is expired the system prompts the user reset button for again process. |

* + 1. **Add User:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | | Add user | | |
| Actors | | Admin | | |
| Summary | | The admin manually adds a new user to the system, by providing their details. | | |
| Pre-Conditions | | The admin must be login in to the system with appropriate permissions and the systems need user details like (name, email and password) must be available. | | |
| Post-Conditions | | The user is successfully added to the system and the user receive the email with their credentials. | | |
| Special Requirement | | The system must be validate that the email is unique and authentic. | | |
| **Basic Flow** | | | | |
| Actor Action | | | System Response | |
| 1 | The admin navigate to the “Manage user” part in the dashboard. | | 2 | The system display an option add user to the admin. |
| 3 | The admin click on that button and fill the information of user. | |  |  |
| 4 | The admin fill out the details and click on “submit” to add the user in the system. | | 5 | The system validate the uniqueness of email and authentication of information by sending OTP on email. |
| 6 | The admin enter the OTP on the OTP field. | | 7 | The system validate the OTP and show message account is “successfully created”. |
| **Alternative Flow** | | | | |
| A4 | If the user enters the incorrect details like duplication of email. | |  | The system displays the following error message: email is already used. |
| A6 | If the user enters the incorrect otp. | |  | The system display an error message incorrect otp and ask enter otp again. |

* + 1. **View Diet plan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | | View diet plan | | |
| Actors | | User | | |
| Summary | | The user will view their personalized diet plan which is recommend from ML model on the basis of his input. | | |
| Pre-Conditions | | The user must be login and navigate to user dashboard and click on the view diet button. | | |
| Post-Conditions | | The user will view their diet plan of one week. | | |
| Special Requirement | | None | | |
| **Basic Flow** | | | | |
| Actor Action | | | System Response | |
| 1 | The user must be login in the system. | | 2 | The System redirect the user on the user dashboard. |
| 3 | The user choose an option or view diet on his dashboard. | | 4 | The system will show the diet plan to the User |
| **Alternative Flow** | | | | |
| A3 | The user enters invalid email and password. | |  | The system displays the following error message: incorrect email address and password and redirect again on the login page. |

* + 1. **Remove User:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | | Remove user | | |
| Actors | | Admin | | |
| Summary | | Admin can remove a user's account permanently from the system. | | |
| Pre-Conditions | | |  | | --- | | The admin must be logged into the system with appropriate permissions and The user account must exist in the database. | | | |
| Post-Conditions | | The user account will be removed from the system and cannot be recovered. | | |
| Special Requirement | | Ensure proper authorization is in place to prevent unauthorized removal of users. | | |
| **Basic Flow** | | | | |
| Actor Action | | | System Response | |
| 1 | The admin navigates to the user management section. | | 2 | The system displays a list of all registered users. |
| 3 | The admin selects a user and clicks on the "Remove User" button. | | 4 | The system confirms the removal with a confirmation message. Upon confirmation, the user is removed from the system. |
| **Alternative Flow** | | | | |
| A3 | The admin attempts to remove a non-existing user. . | |  | The system displays an error message: "User does not exist." |

* + 1. **Manage User account:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | | Manage User Account | | |
| Actors | | Admin | | |
| Summary | | Admin can update or modify user account details such as email, name. | | |
| Pre-Conditions | | The admin must be logged into the system with appropriate permissions and The user account must exist. | | |
| Post-Conditions | | The changes to the user account are saved and reflected in the system. | | |
| Special Requirement | | Ensure proper input validation and role-based access controls. | | |
| **Basic Flow** | | | | |
| Actor Action | | | System Response | |
| 1 | The admin navigates to the user management section. | | 2 | The system displays a list of registered users. |
| 3 | The admin selects a user and edits the account details | | 4 | The system validates the changes and updates the user account successfully. . |
|  |  | | 5 | The system send update details notifications to the User email. |
| **Alternative Flow** | | | | |
| A3 | The admin enters invalid input (e.g., an email in an incorrect format). | |  | The system displays an error message: "Invalid input. Please correct the errors." |

* + 1. **Track Feedback**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | | Track Feedback | | |
| Actors | | Admin | | |
| Summary | | Admin can view feedback submitted by users and monitor trends or issues.   |  | | --- | |  | | | |
| Pre-Conditions | | The admin must be logged into the system. Feedback must exist in the database. | | |
| Post-Conditions | | Feedback is displayed and may be marked as addressed or unresolved | | |
| Special Requirement | | Provide filtering options for easier navigation of feedback | | |
| **Basic Flow** | | | | |
| Actor Action | | | System Response | |
| 1 | The admin navigates to the feedback tracking section. | | 2 | The system displays a list of user feedback with relevant details. |
| 3 | The admin selects a feedback to view details or mark it as addressed | | 4 | The system updates the feedback status accordingly. |
| **Alternative Flow** | | | | |
| A3 | The admin searches for feedback but none exists. | |  | The system displays a message: "No feedback available." |

* + 1. **Manage content**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | | Manage Content | | |
| Actors | | Admin | | |
| Summary | | Admin can create, update, or delete content displayed on the platform. | | |
| Pre-Conditions | | |  | | --- | | The admin must be logged into the system. | | | |
| Post-Conditions | | |  | | --- | | Content changes are saved and displayed on the platform | | | |
| Special Requirement | | Ensure preview functionality is available before saving changes. | | |
| **Basic Flow** | | | | |
| Actor Action | | | System Response | |
| 1 | The admin navigates to the content management section. | | 2 | The system displays existing content items. |
| 3 | The admin creates new content or updates/deletes existing content. | | 4 | The system validates the action and saves the changes. |
| **Alternative Flow** | | | | |
| A3 | The admin does not have permission to update/delete the content | |  | The system displays a warning message: "you don’t have access to update/ delete the content." |

* + 1. **change diet plan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | | Change diet plan | | |
| Actors | | User | | |
| Summary | | User will change his diet plan or specific diet if he/she do not like that diet meal. | | |
| Pre-Conditions | | The user must be login and must have already diet plan | | |
| Post-Conditions | | |  | | --- | | Diet meal is change according to user choice. | | | |
| Special Requirement | | None | | |
| **Basic Flow** | | | | |
| Actor Action | | | System Response | |
| 1 | User must be login with correct credentials. | | 2 | The system validate and redirect to the User dashboard. |
| 1 | The user navigate to the diet plan section on the dashboard. | | 2 | The system displays his/her existing diet meal and also give option to change diet meal. |
| 3 | The user click on the option of change diet meal. | | 4 | The system analyze and give meals option according to user choice. |
| 5 | The user select the meal according to his/her choice. | | 6 | The system update the meal in his diet plan of one week. |
| **Alternative Flow** | | | | |
| A1 | If the user input the incorrect email and password. | |  | The system displays a error message and ask “enter again email and password” |

* + 1. **Receive notification**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | | Receive notification | | |
| Actors | | user | | |
| Summary | | The user receive the notifications from systems about anything for example ( meal time , new content added and update/edit ) | | |
| Pre-Conditions | | |  | | --- | |  |   The user must be connected to the internet. | | |
| Post-Conditions | | Notify from the system | | |
| Special Requirement | | None | | |
| **Basic Flow** | | | | |
| Actor Action | | | System Response | |
| 1 | The user must be connected with internet. | |  |  |
|  |  | | 2 | The System sends the notification about anything e.g. (meal time, update/edit/delete, add content) to user through email. |
| 3 | The User receive an email from the system | |  |  |
|  |  | | 4 | The system validate the notifications sends to all users. |
| **Alternative Flow** | | | | |
| A1 | The user is not connected with internet. | |  |  |

* + 1. **Tracking calories**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | | Tracking calories | | |
| Actors | | user | | |
| Summary | | The user must track his calories through the system how much he/she intake calories. | | |
| Pre-Conditions | | |  | | --- | | The admin must be logged into the system. | | | |
| Post-Conditions | | |  | | --- | | View the calories intake in a day by the user. | | | |
| Special Requirement | | None | | |
| **Basic Flow** | | | | |
| Actor Action | | | System Response | |
| 1 | The user must be login. | | 2 | The system validate and redirect to the user dashboard. |
| 3 | The user move to dashboard and click the “tracking calories” option. | | 4 | The system shows the calories tracking meter where his/her calories intake are stored. |
| **Alternative Flow** | | | | |
| A3 | The user cannot register diet plan. | |  | The system display a message must be register for diet plan. |

* + 1. **Give feedback:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | | Give feedback | | |
| Actors | | User | | |
| Summary | | The user give feedback about the system progress and diet suggestion or any problem they have. | | |
| Pre-Conditions | | |  | | --- | | The admin must be logged into the system. | | | |
| Post-Conditions | | Feedback is successfully submitted now is visible for everyone. | | |
| Special Requirement | | None | | |
| **Basic Flow** | | | | |
| Actor Action | | | System Response | |
| 1 | The User access to the FITNESSSTAN website or app and login to their account. | | 2 | The system validate the user credentials and redirect to the user dashboard. |
| 3 | The User have multiple option and choose the “feedback” option. | | 4 | The system redirect user to the feedback page. |
| 5 | The user select the option of post feedback. | | 6 | The system present a feedback form to the user. |
| 7 | User enters their feedback in the provided text area. | |  |  |
| 8 | User click on the “Submit” or “post” button to publish the feedback. | | 9 | The system adds the feedback to the feedback section. |
|  |  | | 10 | The System shows a notification conforming that the feedback was successfully posted. |
| **Alternative Flow** | | | | |
| A7 | If the user enters inappropriate or abusive feedback. | |  | Then the system may have implemented AI sentiments to identify and remove the abusive feedback. |