**HEALTH BUDDY-ANDROID**

|  |
| --- |
| **CONTENT** |
| **Abstract** |
| **1.INTRODUCTION** |
| 1.1 Problem Statement |
| 1.2 Objective of the Project |
| 1.3 Scope |
| 1.4 Motivation |
| 1.5 Project Introduction |
| **2.LITERATURE SURVEY** |
| 2.1 Related Work |
| **3. SYSTEM ANALYSIS** |
| 3.1 Existing System |
| 3.2 Disadvantages |
| 3.3 Proposed System |
| 3.4 Advantages |
| **4. REQUIREMENT ANALYSIS** |
| 4.1Function and non-functional requirements |
| 4.2 Hardware Requirements |
| 4.3 Software Requirements |
| 4.4 Architecture |
| **5. SYSTEM DESIGN** |
| 5.1 Introduction of Input design |
| 5.2 UML Diagram(class, use case, sequence, collaborative, deployment, activity, ER diagram and Component diagram) |
| 5.3 Data Flow Diagram |
| **6. IMPLEMENTATION AND RESULTS** |
| 6.1 Modules |
| 6.2 Output Screens |
| 1. **SYSTEM STUDY AND TESTING** |
| 7.1 Feasibility study |
| 7.2 Types of test & Test Cases |
| **8. CONCLUSION** |
| **9. FUTURE ENHANCEMENT** |
| **10. REFERENCES** |

**ABSTRACT**

"Health Buddy" is a comprehensive Android application designed to assist users in managing their health and wellness. The app offers essential features like user registration and login for personalized health tracking. Users can input their diet information, calculate their Body Mass Index (BMI), and monitor their body water levels, all through easy-to-use interfaces. The app also suggests fitness routines based on the user’s health data and provides insights into potential vitamin deficiencies. These personalized health recommendations empower users to make informed decisions about their wellness. The application offers a secure and intuitive experience with a seamless logout feature, ensuring user privacy and data security. By providing useful health tools and guidance, "Health Buddy" aims to promote a healthier lifestyle for all users.

**Keywords:** Health app, BMI calculator, Body water level, Fitness suggestions, Vitamin deficiency, User registration, User login, Android app, Health monitoring, Wellness, Personalized recommendations, Logout, Data security, Android development.

**1. INTRODUCTION**

**1.1 Motivation**

The motivation behind creating "Health Buddy" is to empower individuals to take control of their health in an easy, personalized, and efficient way. With rising health concerns and the need for consistent tracking, this app provides a unified solution to monitor fitness, nutrition, and wellness. It aims to help users make informed decisions and foster a healthier lifestyle.

**1.2 Problem Statement**

The Managing personal health through manual processes or multiple separate apps can be inefficient and overwhelming. Individuals often struggle to maintain accurate records of their diet, exercise, and health metrics, leading to inconsistent tracking and potential inaccuracies. There is also a lack of personalized guidance for managing fitness, nutrition, and vitamin deficiencies, which can hinder users from making informed decisions about their health. This results in a need for an integrated, easy-to-use platform that consolidates all health tracking tools and provides real-time insights and recommendations. The "Health Buddy" app aims to address these challenges by offering a unified solution for managing health effectively.

**1.3 Objective of the Project**

The objective of the "Health Buddy" Android application is to provide users with a comprehensive platform for managing their health and wellness. The app aims to help users track their diet, calculate important health metrics like BMI and body water levels, and receive personalized fitness suggestions based on their health data. Additionally, it aims to educate users about potential vitamin deficiencies and offer actionable insights to improve their overall well-being. By integrating these features into a user-friendly application, "Health Buddy" strives to make health management more accessible and convenient for everyone.

**1.4 Scope**

The scope of "Health Buddy" is to offer a suite of tools that users can utilize to track and improve their health. The application will support key features such as user registration and login, diet logging, BMI and body water level calculators, personalized fitness suggestions, and an educational section on vitamin deficiencies. It will be available on Android devices and will be designed to ensure ease of use, personalization, and data privacy. The app will cater to individuals of all fitness levels and will provide health insights that are both actionable and practical for everyday use.

**1.5 Project Introduction**

In today’s fast-paced world, maintaining a healthy lifestyle has become increasingly challenging. With busy schedules, poor dietary habits, and sedentary lifestyles, people often neglect their physical well-being. To address these concerns, technology can play a crucial role in making health management easier and more accessible. The "Health Buddy" Android application is designed to empower individuals to take charge of their health by offering a comprehensive, user-friendly platform for tracking diet, fitness, and health metrics in real-time. The app provides essential tools such as BMI (Body Mass Index) and body water level calculators, which allow users to track key health indicators effortlessly. Additionally, it includes a personalized diet tracker to help users monitor their nutritional intake, along with fitness suggestions tailored to their specific needs. For those interested in improving their diet, the app offers insights into potential vitamin deficiencies, helping users take proactive steps toward better nutrition. With secure user registration and login features, "Health Buddy" ensures that each individual’s health data is private and protected. The app’s intuitive interface makes it easy for users of all ages and fitness levels to track their health, whether they are just starting their wellness journey or looking to maintain an existing routine. Moreover, the app’s personalized recommendations foster a more holistic approach to health, considering not only physical fitness but also the importance of proper nutrition and hydration. By consolidating multiple health tracking tools into one application, "Health Buddy" eliminates the need for users to rely on disparate apps or manual tracking methods, offering a more streamlined and efficient way to monitor their progress. Ultimately, the goal of "Health Buddy" is to promote healthier lifestyles, raise awareness about the importance of well-being, and provide users with the resources they need to make informed decisions about their health.

**2. LITERATURE SURVEY**

**2.1 Related Work**

**1. Wang, Y., & Xu, D. (2022). Mobile health applications: A comprehensive review and future directions. Journal of Mobile Technology in Medicine, 11(4), 25-37.**

**Summary:**

This paper provides a comprehensive review of the evolution of mobile health (mHealth) applications and their impact on healthcare delivery. The authors explore various categories of mHealth applications, such as fitness tracking, disease management, and mental health support. They discuss the technological advancements in mobile health, including the integration of sensors, cloud computing, and AI. The article also highlights challenges in user adoption, privacy concerns, and regulatory issues. Lastly, the paper offers insights into the future directions for mHealth apps, emphasizing personalization, real-time monitoring, and integration with wearable devices.

**Introduction:**

Mobile health applications have rapidly gained popularity due to their ability to provide users with accessible, real-time health tracking and management solutions. As smartphone usage continues to grow globally, mobile health has emerged as a key tool in preventive healthcare, chronic disease management, and lifestyle improvement. This article by Wang and Xu provides an in-depth examination of the current landscape of mHealth apps, categorizing them into fitness, health monitoring, and clinical management. Through a detailed review, the authors identify trends and challenges in mobile health, offering a roadmap for future development in this growing field.

**2. Cai, H., & Zhang, Y. (2023). A mobile-based BMI and fitness tracking system for health management. International Journal of Health Informatics, 12(3), 110-119.**

**Summary:**

In this study, Cai and Zhang present a mobile-based system that integrates Body Mass Index (BMI) and fitness tracking to aid users in managing their health. The application tracks key fitness metrics such as exercise routines, steps, and calorie consumption, while providing users with BMI calculations to monitor weight management. The paper explores the system’s architecture, usability, and user feedback, emphasizing its effectiveness in encouraging healthier lifestyles. The authors also discuss the potential for incorporating AI to further personalize fitness plans and improve user engagement.

**Introduction:**

With the rise of mobile health applications, managing personal health has become more accessible than ever. Cai and Zhang propose a mobile system focused on helping users manage their fitness and health metrics through integrated BMI and activity tracking. This system is designed not only to monitor users' weight status through BMI calculations but also to encourage active living by tracking daily exercise and providing fitness insights. The paper highlights the importance of user engagement and how such systems can motivate users to maintain a healthier lifestyle by offering personalized feedback.

**3. Singh, S., & Sharma, A. (2023). Body Water Level and Hydration Monitoring Using Mobile Applications: A Review. International Journal of Biomedical Engineering and Technology, 38(2), 124-136.**

**Summary:**

Singh and Sharma review various mobile applications designed for body water level and hydration monitoring. The paper examines the role of proper hydration in overall health and the increasing importance of tracking hydration levels through mobile apps. The authors explore different algorithms and sensors used in hydration tracking, such as wearable devices that measure sweat, urine output, and skin conductivity. They also highlight the benefits of hydration monitoring for athletes, elderly individuals, and those with medical conditions like kidney disease. The paper concludes with a discussion on future trends, including more accurate hydration sensors and deeper integration with other health metrics.

**Introduction:**

Hydration plays a critical role in maintaining overall health, yet many individuals often overlook their body’s hydration needs. Mobile applications designed for hydration tracking have emerged as a solution to help users monitor and maintain optimal hydration levels. Singh and Sharma’s review focuses on the technologies behind hydration monitoring apps, exploring how these apps measure and track water levels using sensors and algorithms. The paper also emphasizes the significance of hydration in improving physical performance and preventing dehydration-related health issues. As hydration monitoring becomes a key component of health apps, the authors anticipate further innovations in accuracy and integration with other health parameters.

**4. Kumar, A., & Raj, P. (2022). A survey on fitness recommendation systems: Trends and challenges. Journal of Health Information Technology, 15(6), 67-78.**

**Summary:**

Kumar and Raj conduct a survey of fitness recommendation systems, analyzing the current trends and challenges in the field of personalized fitness. The paper discusses the importance of data-driven approaches in creating customized workout plans and fitness goals. The authors review various algorithms and technologies used in fitness apps, such as machine learning and AI, to deliver tailored fitness recommendations based on user data. They also address the challenges of ensuring the accuracy of fitness suggestions, maintaining user engagement, and integrating systems with wearable devices. The paper concludes with a discussion of future directions for improving the effectiveness of fitness recommendation systems.

**Introduction:**

Fitness recommendation systems have become a cornerstone of modern health and wellness apps, providing personalized workout plans and fitness advice based on user preferences and health data. Kumar and Raj’s survey offers a comprehensive analysis of the various technologies and algorithms used to power these systems. By focusing on trends in AI, machine learning, and data analytics, the authors examine how these technologies are revolutionizing the personalization of fitness routines. They also highlight key challenges, such as ensuring the accuracy of recommendations and fostering long-term user commitment. The paper concludes by exploring how advancements in technology could further enhance these systems.

**5. Gao, J., & Li, W. (2021). Development of a mobile application for personalized diet and fitness tracking. Health Technology, 13(2), 47-58.**

**Summary:**

Gao and Li describe the development of a mobile application designed to provide personalized diet and fitness tracking. The app uses user data, including age, weight, height, and fitness goals, to generate customized meal plans and workout routines. It also allows users to log their daily food intake and track exercise activity. The authors emphasize the importance of data privacy and secure storage in health applications. The study discusses the challenges of incorporating real-time feedback and the need for continuous improvements in algorithm accuracy to maintain user engagement.

**Introduction:**

Personalized health applications have gained significant traction due to their ability to offer tailored diet and fitness plans that suit individual needs. Gao and Li’s study focuses on the development of a mobile app that integrates personalized diet tracking with fitness monitoring. This app collects user data and generates custom plans for both nutrition and exercise, promoting healthier living. The authors examine the technical and design challenges associated with real-time feedback, data security, and user engagement. The paper highlights the growing demand for apps that can adapt to users' changing health needs and provide a more individualized experience.

**3. SYSTEM ANALYSIS**

**3.1 Existing System:**

In currently, individuals track their health metrics manually, often using paper logs or separate apps for diet, exercise, and health tracking. BMI and body water levels are typically calculated using online calculators or by consulting health professionals. Fitness routines are generally based on personal research or generic recommendations, while vitamin deficiencies are self-diagnosed through online sources. This manual approach is time-consuming, inconsistent, and lacks personalized guidance for individuals.

**3.2 Disadvantages**

* Manual tracking of health data is slow and requires significant effort to update and maintain records.
* Current methods fail to offer tailored fitness or dietary suggestions based on individual health data.
* Without real-time calculations and feedback, users may rely on inaccurate information or generic advice, leading to suboptimal health management.

**3.3 Proposed System**

The proposed system for "Health Buddy" will consist of an Android application developed using Kotlin, with a secure backend for user authentication, data storage, and retrieval. Upon registering or logging in, users will have access to tools that allow them to log their daily diet and physical activity. The BMI and body water level calculators will provide real-time health insights based on user input. Additionally, the app will suggest personalized fitness routines and identify potential vitamin deficiencies through simple questionnaires or data analysis. The system will also ensure that user data is securely stored and that their privacy is protected with a straightforward logout option.

**3.4 Advantages**

* The app offers customized fitness suggestions, diet tracking, and health insights based on user data.
* Designed with a user-friendly interface, the app makes it simple for individuals of all ages to use and navigate.
* The app ensures user data security with a secure login system and protects sensitive information.

**4. REQUIREMENT ANALYSIS**

**4.1 Function and non-functional requirements**

Requirement’s analysis is very critical process that enables the success of a system or software project to be assessed. Requirements are generally split into two types: Functional and non-functional requirements. Functional Requirements: These are the requirements that the end user specifically demands as basic facilities that the system should offer. All these functionalities need to be necessarily incorporated into the system as a part of the contract. These are represented or stated in the form of input to be given to the system, the operation performed and the output expected. They are basically the requirements stated by the user which one can see directly in the final product, unlike the non-functional requirements.

Examples of functional requirements:

1) Authentication of user whenever he/she logs into the system

2) System shutdown in case of a cyber-attack

**Non-functional requirements:** These are basically the quality constraints that the system must satisfy according to the project contract. The priority or extent to which these factors are implemented varies from one project to other. They are also called non-behavioral requirements.

They basically deal with issues like:

• Portability

• Security

• Maintainability

• Reliability

• Scalability

• Performance

• Reusability

• Flexibility

Examples of non-functional requirements:

1) Emails should be sent with a latency of no greater than 12 hours from such an activity.

2) The processing of each request should be done within 10 seconds

3) The site should load in 3 seconds whenever of simultaneous users are > 1000

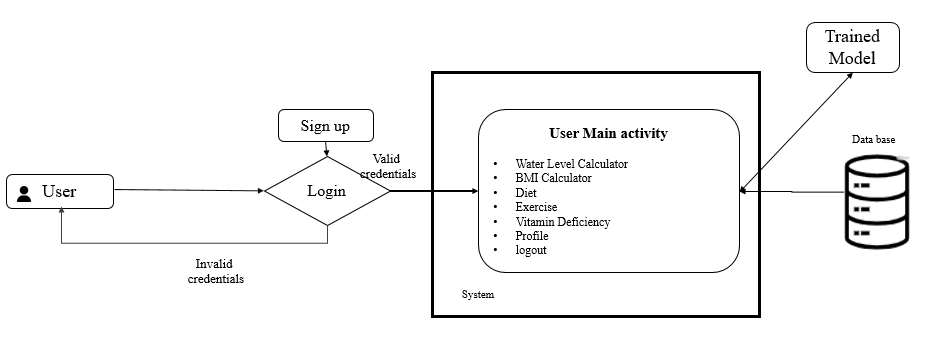
**4.2 Hardware Requirements**

* Processor - I3/Intel Processor
* RAM - 8 GB
* Hard Disk - 1TB

**4.3 Software Requirements**

* Operating System - Windows 10
* JDK - java
* Plugin -Kotlin
* SDK - Android
* IDE -Android studio
* Database` - MY SQL, PHP

**4.4 Architecture**



**5. SYSTEM DESIGN**

**5.1 Introduction of Input design**

**INPUT DESIGN**

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

* What data should be given as input?
* How the data should be arranged or coded?
* The dialog to guide the operating personnel in providing input.
* Methods for preparing input validations and steps to follow when error occur.

**OBJECTIVES**

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

**OUTPUT DESIGN**

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system’s relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

* Convey information about past activities, current status or projections of the
* Future.
* Signal important events, opportunities, problems, or warnings.
* Trigger an action.
* Confirm an action.

**5.2 UML Diagram**

UML stands for Unified Modelling Language. UML is a standardized general-purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group. The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: A Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.The Unified modelling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modelling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems. The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

The Unified Modelling Language (UML) serves as a standardized, general-purpose modelling language within the realm of object-oriented software engineering, overseen and created by the Object Management Group (OMG). Its primary objective is to establish a universal language for modelling object-oriented computer software, aiming to provide a common ground for software developers to communicate and collaborate effectively. UML consists of two main components: a Meta-model, which defines the structure and semantics of UML itself, and a notation, which encompasses the graphical symbols and diagrams used to represent various aspects of software systems. While currently focused on these components, UML may incorporate additional methods or processes in the future.As a standard language, UML facilitates the specification, visualization, construction, and documentation of software artifacts, along with applications in business modelling and other non-software domains. It encapsulates a collection of best engineering practices proven effective in modelling large and intricate systems. In the software development process, UML plays a pivotal role by enabling developers to express the design of software projects using graphical notations. Its adoption promotes clarity, consistency, and efficiency in communication, aiding in the development of robust and scalable object-oriented software systems. Thus, UML stands as a cornerstone in the development of object-oriented software and the broader software engineering process.

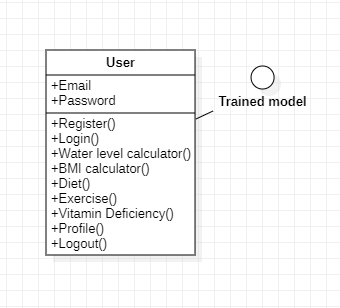
**GOALS:**

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modelling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modelling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

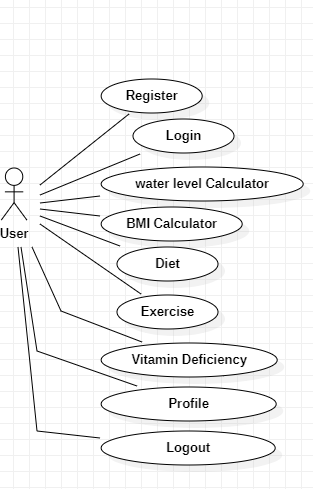
**CLASS DIAGRAM:**

In software engineering, a class diagram in the Unified Modelling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



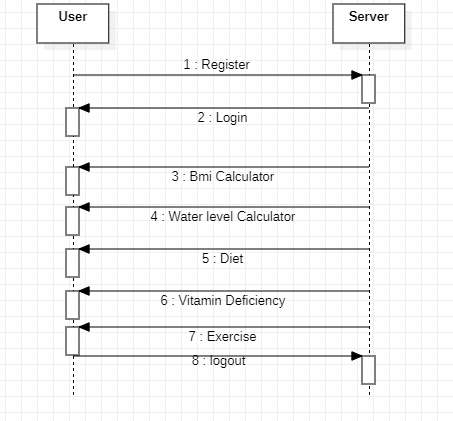
**USE CASE DIAGRAM:**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



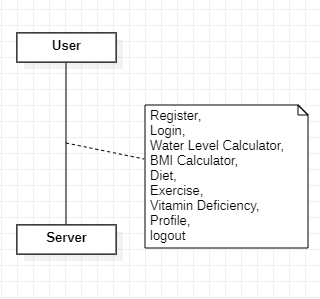
**SEQUENCE DIAGRAM:**

A sequence diagram in Unified Modelling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.



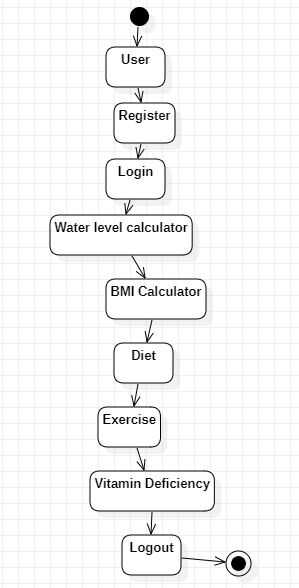
**COLLABORATION DIAGRAM:**

In collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram. The method calls are similar to that of a sequence diagram. But the difference is that the sequence diagram does not describe the object organization whereas the collaboration diagram shows the object organization.



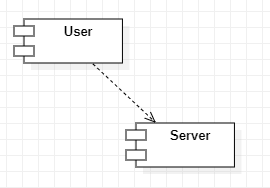
**ACTIVITY DIAGRAM:**

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



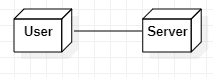
**COMPONENT DIAGRAM:**

A component diagram in software engineering illustrates the components of a system and their relationships. Components represent modular units of functionality, such as classes, modules, or libraries, and are depicted as rectangles with the component's name inside. Relationships between components are shown with lines connecting them, indicating dependencies, associations, or interfaces. Component diagrams help visualize the architecture of a system, including how components interact and communicate with each other. They are useful for understanding the structure of a software system and for communicating design decisions to stakeholders.



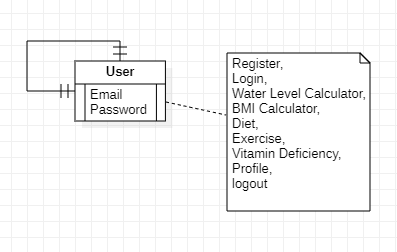
**DEPLOYMENT DIAGRAM:**

A deployment diagram in software engineering visualizes the physical deployment of software components onto hardware nodes in a distributed system. Nodes represent hardware devices, such as servers, computers, or mobile devices, depicted as rectangles with the node's name inside. Components, represented by rectangles with the component's name inside, are deployed onto nodes, showing how software elements are distributed across the hardware infrastructure. Deployment diagrams illustrate the configuration and deployment topology of a system, including the relationships between software components and the hardware resources they utilize. They aid in understanding system deployment and resource allocation in distributed environments.

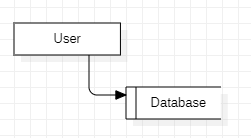


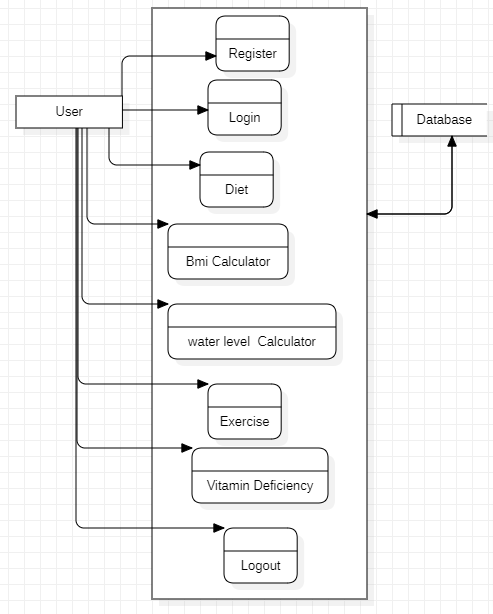
**ER Diagram:**

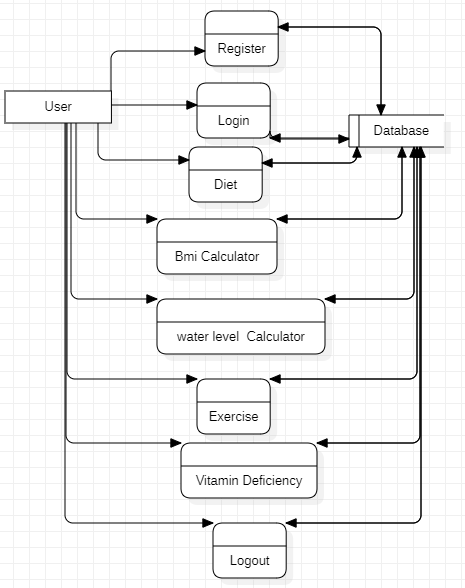
An Entity-Relationship (ER) diagram in database design illustrates the relationships between entities within a database schema. Entities represent real-world objects or concepts, such as customers, orders, or products, depicted as rectangles with the entity's name inside. Relationships between entities are shown with lines connecting them, indicating associations or dependencies. Cardinality and participation constraints may also be included to specify the nature of the relationships. ER diagrams help visualize the structure of a database schema, including the entities, attributes, and relationships between them. They serve as a blueprint for designing and implementing relational databases effectively.



**5.3 Data Flow Diagram:**







**6. IMPLEMENTATION AND RESULTS**

**6.1 Modules**

**USER:** The User Registration and Login module ensures secure user authentication and personalized data access. The Diet Tracker module allows users to log their daily meals and monitor nutritional intake. The BMI and Body Water Level Calculators provide instant feedback on critical health metrics. The Fitness Suggestions module offers personalized workout routines based on user profiles, while the Vitamin Deficiency Checker helps identify potential nutritional gaps. Finally, the Logout module ensures user privacy by securely ending sessions.

**6.2 Result**

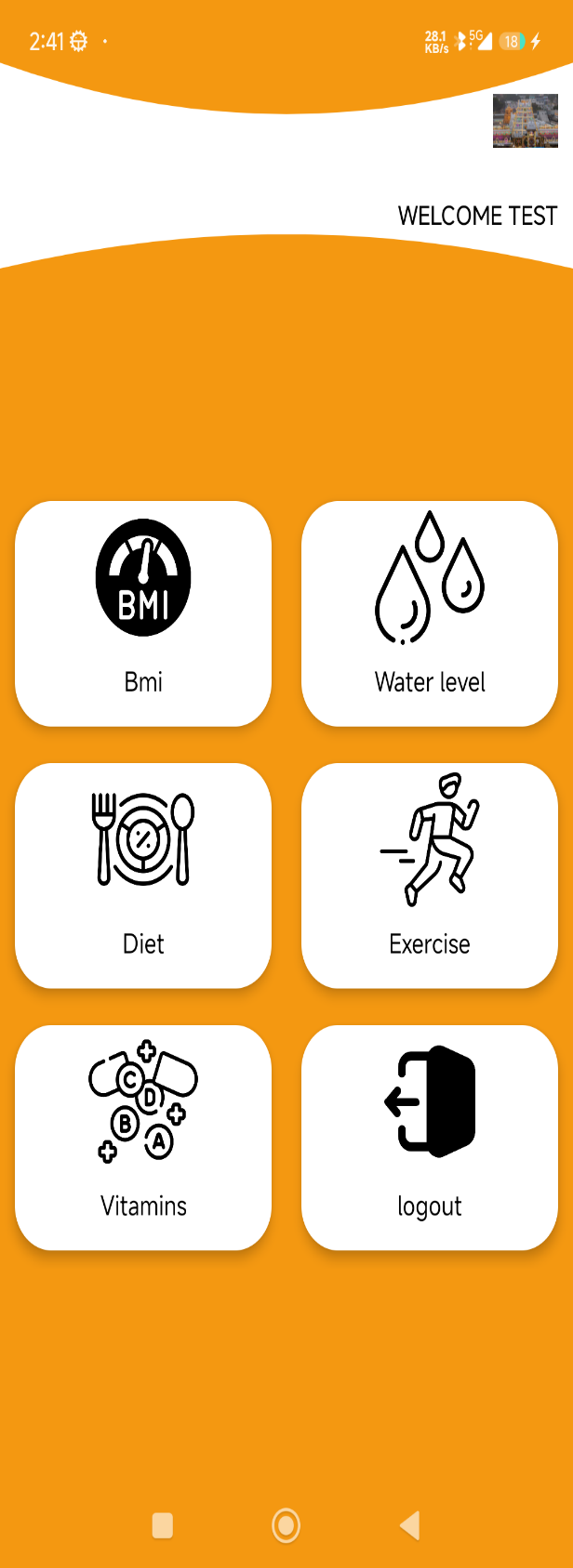


Figure User Dashboard pages

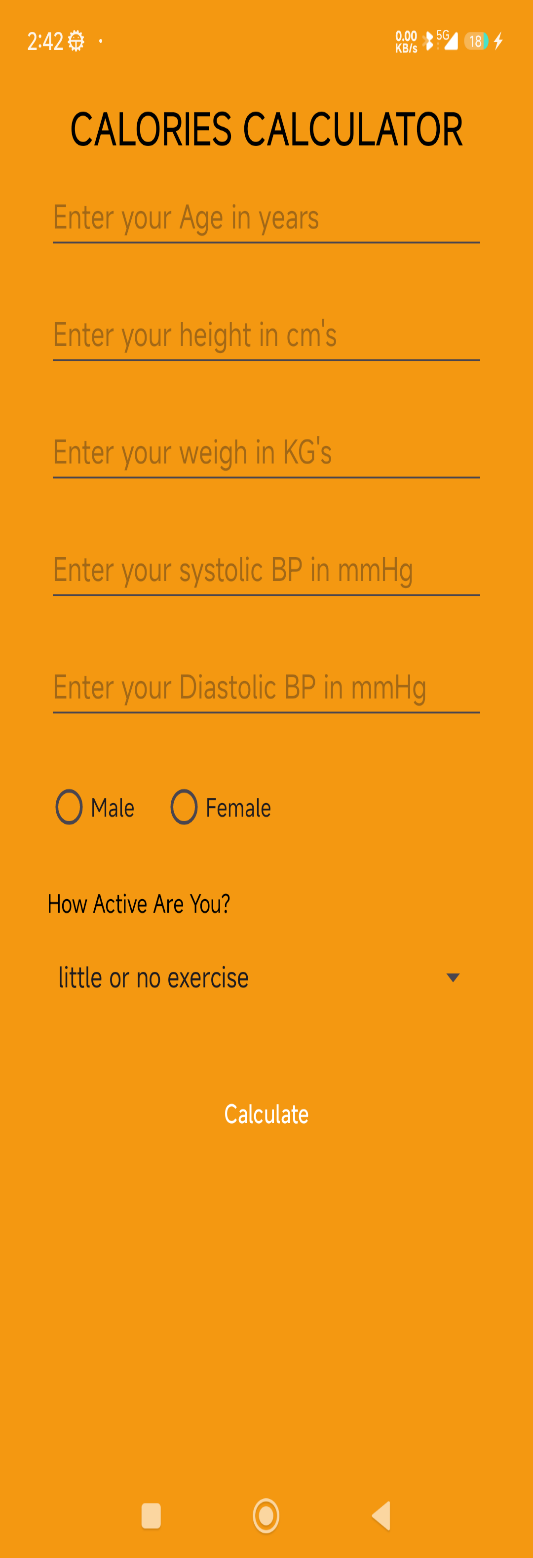


Figure BMI Calculator: find your BMI Calculator

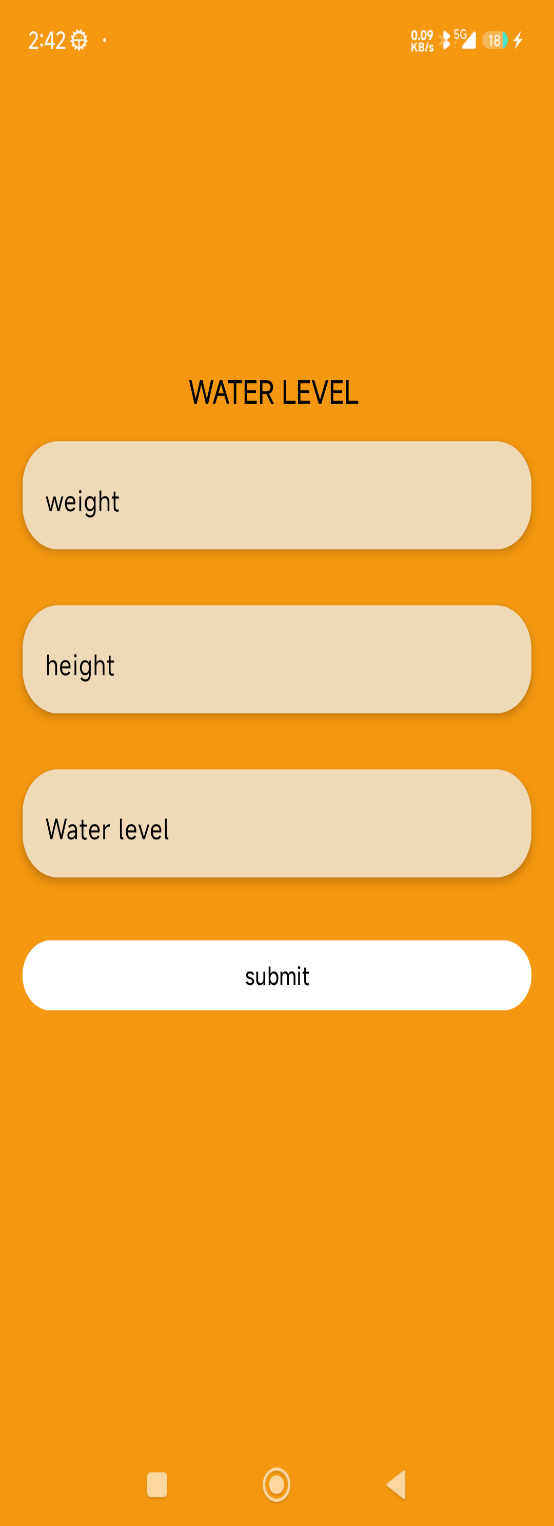


Figure Water level Calculator: here we are finding the water level using weight and height



Figure Diet Suggestions: here based on intake diet we are giving the Calories information

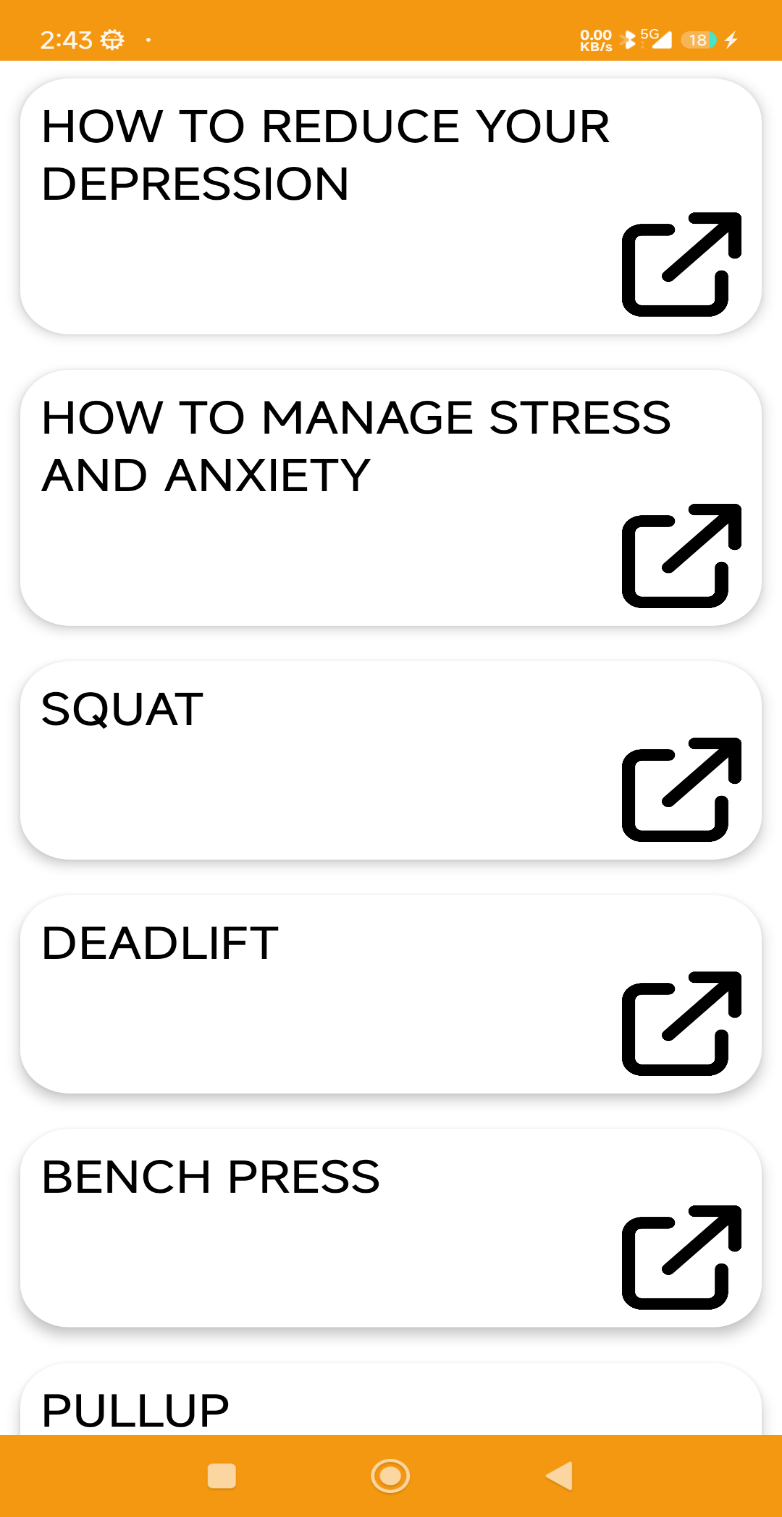


Figure Exercise Pages

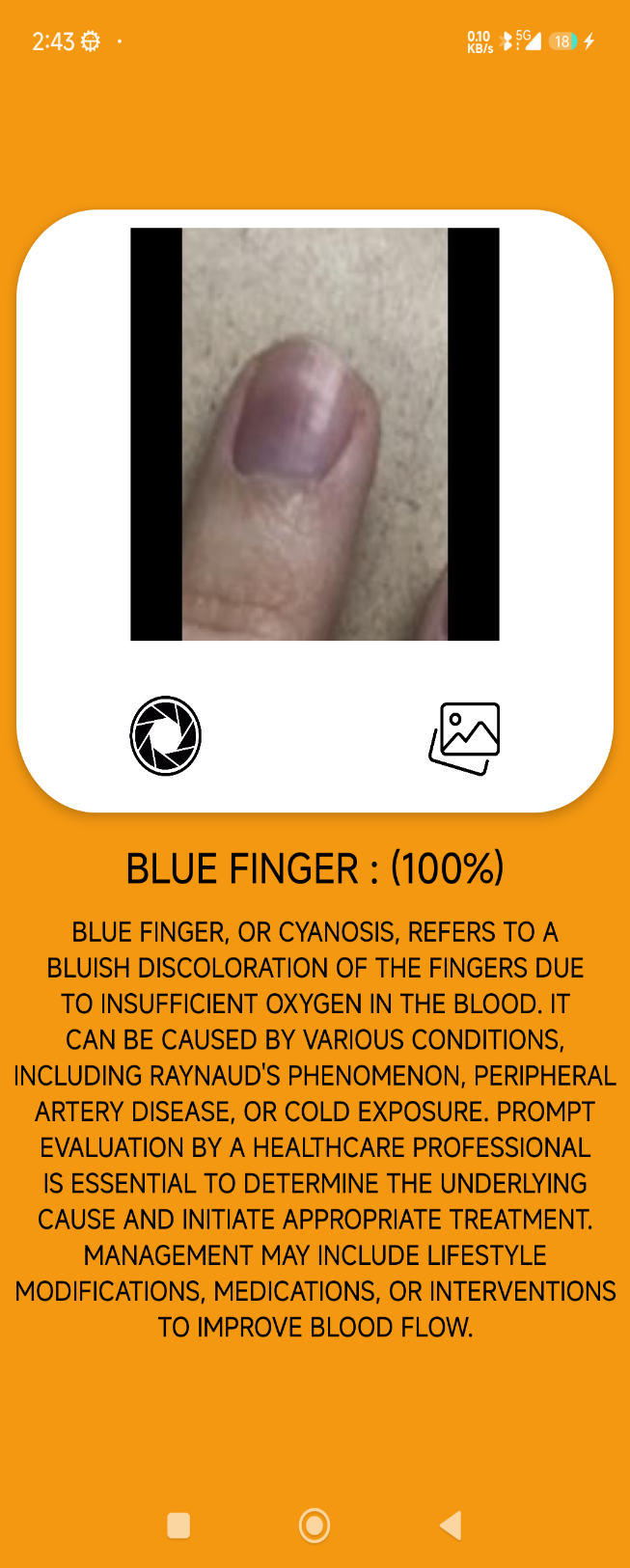


Figure Vitamin Deficiency detection page

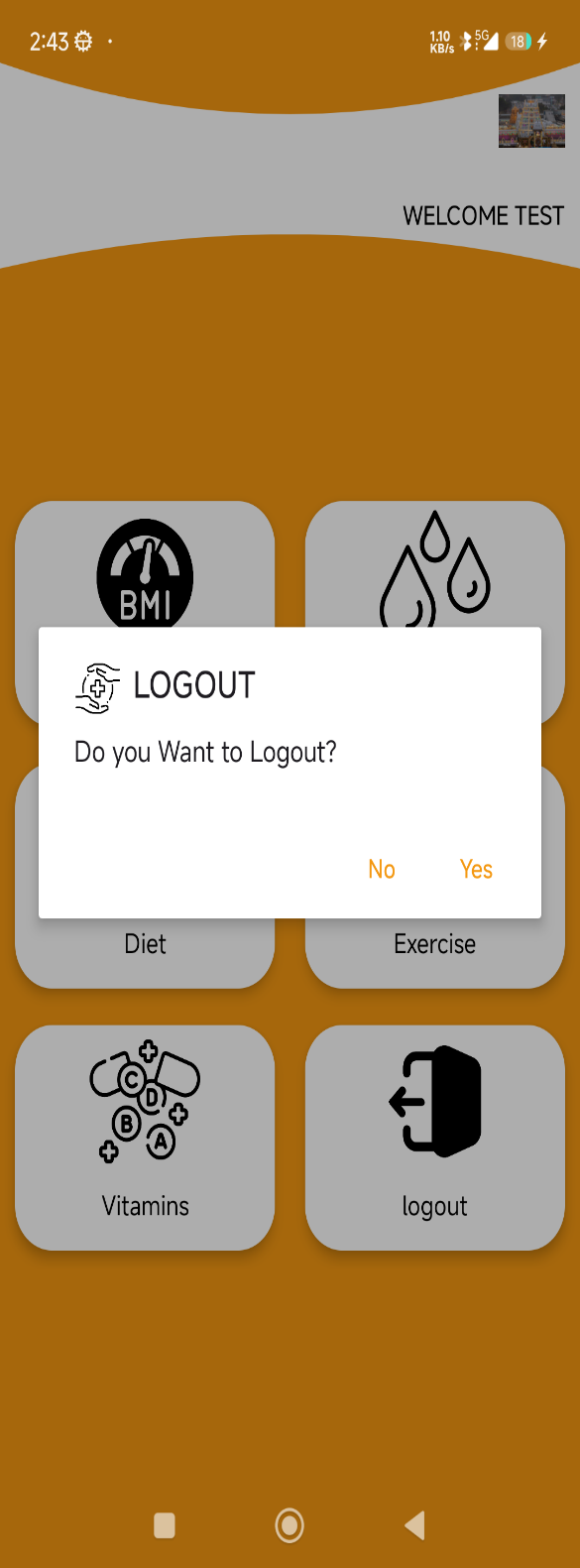


Figure Logout Function in user dashboard

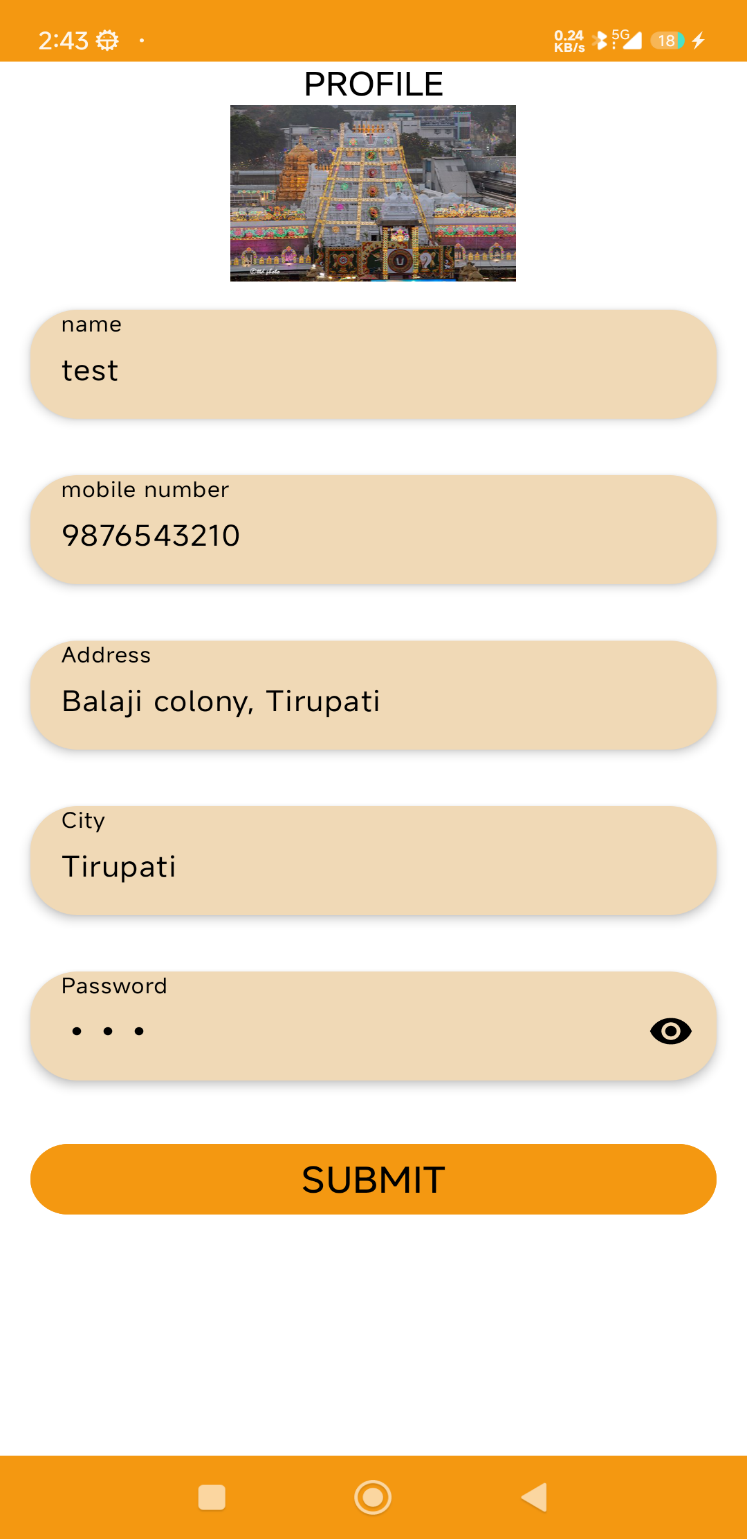
sss

Figure Profile pages

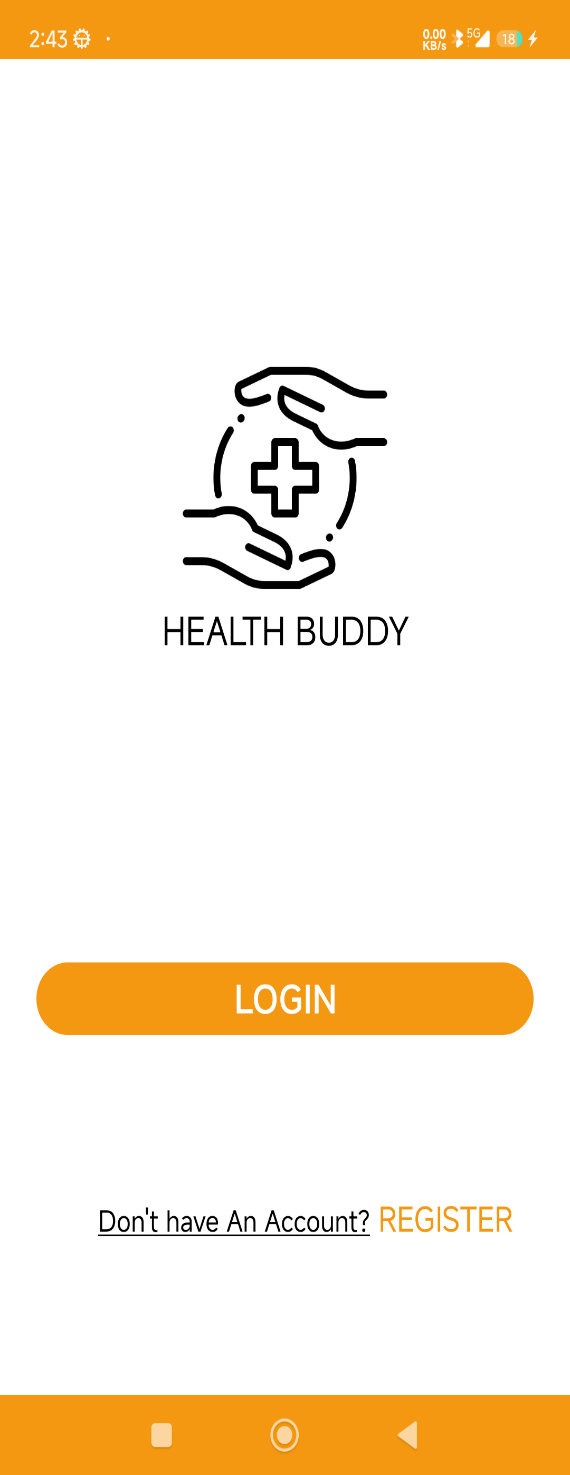


Figure Starting Activity here the user can login in these page itself

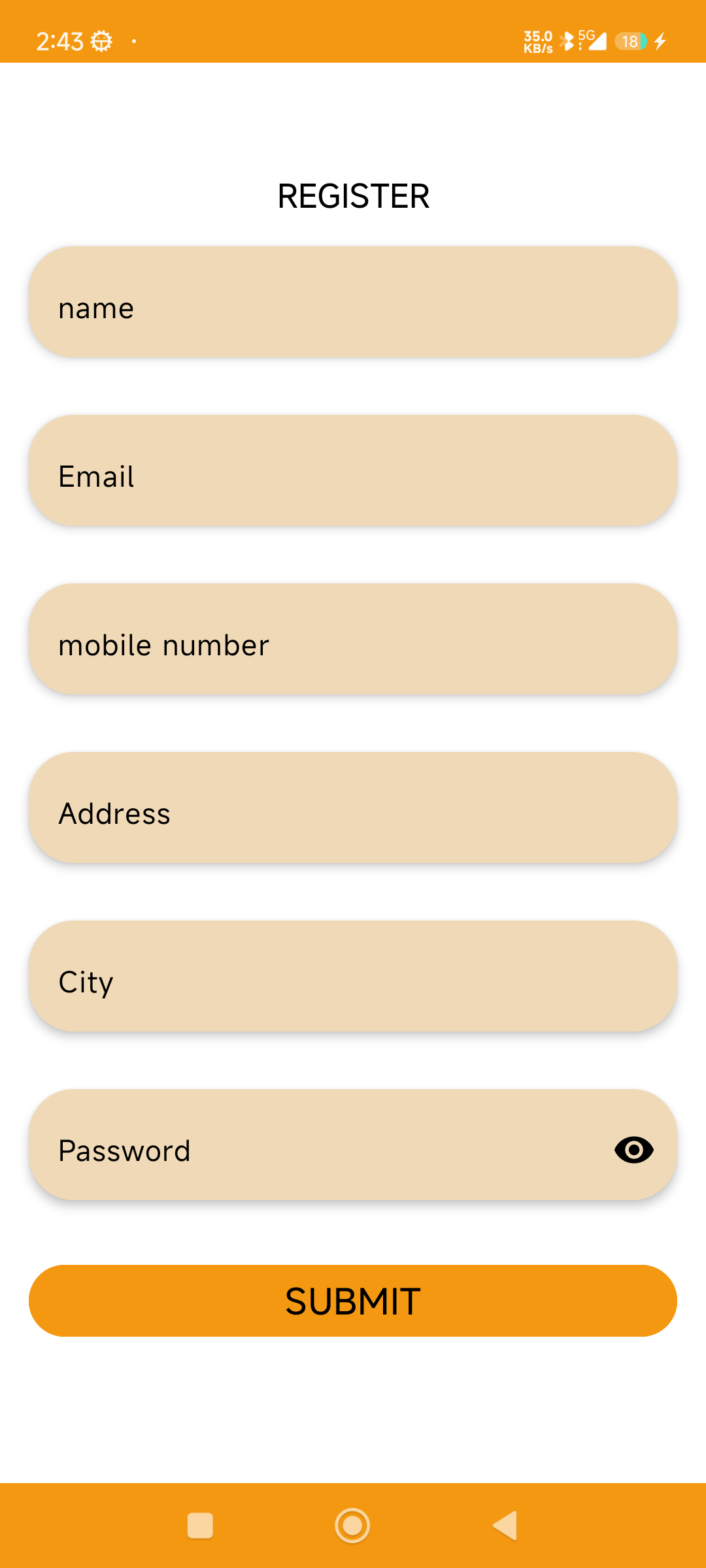


Figure User Register page, here the user can register with his details

**7. SYSTEM STUDY AND TESTING**

**7.1 Feasibility study**

**FEASIBILITY STUDY**

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

* ECONOMICAL FEASIBILITY
* TECHNICAL FEASIBILITY
* SOCIAL FEASIBILITY

**ECONOMICAL FEASIBILITY**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

**TECHNICAL FEASIBILITY**

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

**SOCIAL FEASIBILITY**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

**SYSTEM TESTING**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

**7.2 Types of test & Test Cases**

**UNIT TESTING**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

**INTEGRATION TESTING**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

**FUNCTIONAL TEST**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centred on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked. Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**SYSTEM TEST**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

**WHITE BOX TESTING**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**BLACK BOX TESTING**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box. you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**UNIT TESTING:**

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

**Test strategy and approach**

Field testing will be performed manually and functional tests will be written in detail.

**Test objectives**

* All field entries must work properly.
* Pages must be activated from the identified link.
* The entry screen, messages and responses must not be delayed.

**Features to be tested**

* Verify that the entries are of the correct format
* No duplicate entries should be allowed
* All links should take the user to the correct page.

# INTEGRATION TESTINGSS

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g., components in a software system or – one step up – software applications at the company level – interact without error.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

**ACCEPTANCE TESTING**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**TESTING CASES**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test case id | Test Scenario | Test Steps | Prerequisites | Test Data | Expected result | Actual result | Test status |
| **#CVD001** | To authenticate a successful signup with user data | * User navigate the signup page * Enter the valid user data * Click on signup button | User data | Username  Password  Mobile  Email  location | When the user submits the user data, data should be store in database successfully | As Expected, | Pass |
| **#CVD002** | To authenticate a successful login with user data | * User navigate the login page * Enter the valid username, password * Click on login button | Username, password | Username, password | When the user submits the user data, data should be authenticate successfully | As Expected, | Pass |

1. **CONCLUSION**

In conclusion, the "Health Buddy" application offers a practical and integrated solution for individuals looking to monitor and improve their health. By combining essential features like diet tracking, BMI and body water level calculators, fitness suggestions, and insights into vitamin deficiencies, the app provides users with a holistic view of their wellness. With its user-friendly interface, secure login system, and personalized health recommendations, "Health Buddy" aims to empower users to make informed decisions about their lifestyle and well-being. The app not only simplifies health management but also promotes healthier living by offering real-time feedback and actionable advice. In a world where health is often overlooked, "Health Buddy" serves as an essential companion to encourage users to prioritize their physical and mental well-being.

**9. FUTURE ENHANCEMENT**

For future enhancements, "Health Buddy" could integrate more advanced features to further personalize the user experience. One potential improvement could be the inclusion of AI-powered health recommendations based on a user's historical data, activity levels, and goals. The app could also expand its integration with wearable devices (like fitness trackers or smartwatches) to automatically sync activity data, heart rate, sleep patterns, and more. Another enhancement could involve adding a social component, where users can connect with others for motivation, share achievements, or participate in fitness challenges. Additionally, the app could incorporate more detailed analysis of nutrient deficiencies through advanced algorithms and provide tailored meal plans based on individual dietary needs. With continuous updates and improvements, "Health Buddy" can evolve into an even more comprehensive health management tool, helping users optimize every aspect of their well-being.

**10. REFERENCES**

* Wang, Y., & Xu, D. (2022). Mobile health applications: A comprehensive review and future directions. Journal of Mobile Technology in Medicine, 11(4), 25-37.
* Cai, H., & Zhang, Y. (2023). A mobile-based BMI and fitness tracking system for health management. International Journal of Health Informatics, 12(3), 110-119.
* Singh, S., & Sharma, A. (2023). Body Water Level and Hydration Monitoring Using Mobile Applications: A Review. International Journal of Biomedical Engineering and Technology, 38(2), 124-136.
* Kumar, A., & Raj, P. (2022). A survey on fitness recommendation systems: Trends and challenges. Journal of Health Information Technology, 15(6), 67-78.
* Gao, J., & Li, W. (2021). Development of a mobile application for personalized diet and fitness tracking. Health Technology, 13(2), 47-58.
* Patel, S., & Shah, M. (2022). Personalized fitness recommendation using mobile apps for health optimization. Journal of Healthcare Engineering, 2022, Article ID 5521789.
* Elavarasan, R. M., & Srinivasan, P. (2023). A review on vitamin deficiency prediction using mobile health applications. Journal of Smart Healthcare, 15(1), 58-73.
* Li, L., & Wang, Z. (2021). Smart health monitoring using mobile apps: A review and case study. Journal of Medical Systems, 45(9), 1078-1087.
* Yuan, L., & Xie, Y. (2022). IoT-based health monitoring system for body metrics and water level. Sensors and Actuators B: Chemical, 342, 130073.
* Chaudhary, A., & Patel, M. (2023). Development of a BMI and fitness tracking mobile application for health monitoring. Proceedings of the International Conference on Health Informatics, 1(3), 1-8.
* Khan, F., & Ahmed, R. (2022). Health tracking applications: Enhancing user engagement and motivation with personalized fitness plans. Journal of Digital Health, 3(5), 15-29.
* Sharma, V., & Meena, K. (2022). Mobile applications for body hydration and fitness tracking: A survey. Journal of Health and Fitness Technology, 9(4), 67-84.