Tribhuvan University

Faculty of Humanities and Social Sciences

A Project Report on

“LOGIFY TRACKING APP”

In partial fulfillment of the requirement for the degree of Bachelor in Computer Application

(BCA)

<u>Submitted To:</u>

Department of Computer Application

Kathmandu College of Technology

<u>Submitted by:</u>

Anuj Rana (11051196)

2079/04/13

Under the Supervision of

Mr. Santosh Rijal

Tribhuvan University

Faculty of Humanities and Social Sciences

Kathmandu College of Technology

SUPERVISOR’S RECOMMENDATION

I hereby recommend that this project prepared under my supervision by “**Anuj Rana**” entitled “**LOGIFY**” in partial fulfilment of the requirements for the degree of Bachelor of Computer Application is recommended for the final evaluation.

…………………….

SIGNATURE

Mr. Santosh Rijal

Supervisor

Lokanthali, Kathmandu

Tribhuvan University

Faculty of Humanities and Social Sciences

Kathmandu College of Technology

LETTER OF APPROVAL

This is to certify that this project is prepared by “**Anuj Rana”** entitled “**LOGIFY**” in partial fulfillment of the requirements for the degree of Bachelor in Computer Application has been evaluated. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

………………………………. **Mr. Santosh Rijal** **Supervisor** **Lokanthali, Kathmandu**	**……………………………….** **Mr. Santosh Rijal** **Head of Department** **Lokanthali, Kathmandu**
………………………………. **Internal Examiner**	**……………………………….** **External Examiner**

STUDENT’S DECLARATION

I hereby declare that I am the only author of this work and that no sources other than those listed here have been used in this work.

…………………………………

Anuj Rana

Date : 2022/8/6

Abstract

Fitness apps that helps users by arranging personal exercise plans, along with the trends of smartphone applications. This paper will review the literatures of logging technologies and their social implications, linking the using of fitness apps to the trends of tracking, self-surveillance of health and exercises.

Logify is an android app that uses manual inputs from data users and store it in the database. It can store weight and exercise data and show the result in few click. App can help in setting fitness goals, provide progress report. They also provide personalized result using the information one provides. This app is increasingly becoming popular among college students. The use of apps has increased due to various reasons like ease of access, time constraints, and reduced cost as compared to note and paper.

Keyword :- Fitness app, log, Tracker.

ACKNOWLEDGEMENTS

I the undersigned have great pleasure in giving our sincere thanks to those who have contributed their valuable time in helping us to achieve the success in our project work. I would like to express our gratitude to all those gave us the possibility to complete this project. I want to thank Kathmandu college of technology for giving us the opportunity for doing this project.

I am indebted and thankful to our Project Guide Sir Santosh Rijal to whom I owe her piece of knowledge for her valuable and timely guidance, co-operation, encouragement & time spent for doing this project work. Our sincere thanks to the IT staff for providing us sufficient information which helped us to complete our project successfully

Table of Contents

List of Figures

List of Tables

List of Abbreviations

APK	Android application PacKage
DBMS	Database Management System
GUI	Graphical User Interface
XML	Extensible Markup Language

NOTE :- The system is referred to as LOGIFY in the rest of the document.

CHAPTER 1

INTRODUCTION

1.1 Introduction

Everyone knows that fitness is important for your health, but how many of us actually do it? The truth is that most of us need a little bit of a push or a bit of a nudge to motivate ourselves. Post Covid has got many of us to start care for our health by engaging in different physical activities.

Studies have shown that regular exercise can have a big impact on our health, but it's also a lot easier to skip a workout when life gets busy. That's why it's so important to find that balance in life and workout. Workout is effective to keep us healthy but workout without any progress it can never be promising.

Here, to make workout easier and more effective Logify has been developed.

Logify is a fitness tracking application that tracks workout sets and weight. Fitness being a concern these days Logify will ease the tracking of progress in gym and weight.

It will keep track of your workouts, track your progress, and make it easier to stay on track.

This application will keep record of workout and weight which will make progress visible to user.

User can select varieties of workout available on our application and select the weight including repetition and sets. Also, they will be able to keep track of their weight along with the workout.

They can view their progress instantly with few taps and see their progress.

1.2 Problem Statement

It is nearly impossible to remember the workout sets and weight in your head. Also, jotting down on paper where taking pen and paper on gym seems infeasible.

So, to overcome the traditional method of tracking workout and weight 'Logify' will save time and effort.

1.3 Objectives

The main aim of our application is listed below:

- To create a simple application which helps to track workout.
- To create an application where you can see your progress of weight.
- To create an application where you can track your weight.

1.4 Scope and Limitation

Logify has its pros and cons like other applications. This application offers following scope and lacks following things:

1.4.1 Scope

- Users can easily track their workout and weight.
- Easy to use interface.
- Users can view progress on few taps.
- User friendly environment.
- Users can signup using Google

1.4.2 Limitation

- Users have to manually keep the record of their activity.

  Limited sets of workout.

1.5 Report Organization

This report document contains five chapters including this chapter.

Chapter 1: Gives brief introduction.

Chapter 2: Defines and describes Background Study and Overview of related existing systems and their pros and cons.

Chapter 3: Presents the System Analysis and Design including Requirement Analysis and Feasibility Analysis.

Chapter 4: Presents the Implementation, Testing and debugging are explained.

Chapter 5: Conclusion, Limitations and Future Enhancement are briefly explained.

CHAPTER 2

BACKGROUND STUDY AND LITERATURE REVIEW

2.1 Background Study

Smartphones users are significantly increasing in number. Since most of the athletes and gym goers already have one they can use it to track their workout and weight. This makes tracking more convenient and effective. General concept and terminologies are mentioned below:

1. **Tracking Weight :** Users can record their weight after weighing themselves. Logify keep record.
2. **Tracking Workout :** With pre build workout sets users can select the workout from the list and keep tracking sets and repetition of the same workout.

2.2 Literature Review

Technological advancements have led to increased digitization within healthcare and sports. The emergence of available smartphone applications (apps) in Google Play and the App Store (iTunes) in September 2008 and June 2009, respectively, have contributed to a better understanding of human health by allowing us to gather vast amounts of medical and fitness data. Specifically, some improvements in app technology (e.g., a built-in camera for heart rate assessment, accelerometers, etc.) have opened new opportunities for collecting relevant information in the clinical and sports settings [1].

The relationship between self-selected fitness application software use (henceforth, fitness apps) and physical activity behavior is not well studied, nor is the role that exercise identity may play in predicting fitness app use. This study examined these relationships by evaluating the ability of exercise identity to mediate a hypothesized positive association.

between fitness app use and physical activity behavior. Data were collected from college students (N = 310) who completed survey instruments assessing: physical activity behavior, exercise identity, and fitness app use. The following groups were established: apps group (those with fitness apps) and no apps (those without fitness apps). As hypothesized, the apps group participated in significantly more physical activity than the no apps group. Additionally, the apps group reported significantly greater exercise identity. Exercise identity was significantly and positively associated with physical activity [2]**.**

Nowadays, many health and fitness applications(apps) can be downloaded from app stores, changing the way people manage their health and chronic diseases. This paper reflects on 7 years of experience in mobile health and fitness app development. It analyzes the uptake of a health and fitness app, my Fitness Companion®, by the healthcare industry and end-users dealing with chronic disease management. The use of my Fitness Companion® is analyzed from an end-user perspective. The app is available via Google Play since February 2011 and the research presented is based on data collected from 5500+users over a period of7 months. The paper also discusses how mHealth apps could be distributed in the near future, as well as, the use of Personal Health Record (PHR) systems such as Microsoft Health Vault, and the impact of regulations on the future of Health apps. The conclusion highlights the challenges and opportunities for app developers in the mHealth industry [3]**.**

Physical inactivity, sedentary behavior, and lower fitness levels have been identified as risk factors for a wide range of non-communicable diseases However, it has been shown that achieving high levels of moderate to vigorous intensity physical activity can reduce the elevated risk of death associated with high sedentary behavior [4]. Unlike previous research on fitness and diet apps which have mainly studied individuals' intentions to use the apps, this study focused on the prediction accuracy of various factors that lead people to use fitness and diet apps through analysis of data collected from users as well as non-users of these apps. To examine prediction accuracy, this study applied the Random Forest algorithm. According to the findings, prediction accuracy higher than that of 70 percent was observed for nine factors: age, annual income, education, perceived obesity, dieting efforts, number of smartphone apps currently used, daily time spent with smartphone apps, perceived benefits from exercise, and social influence. A major contribution of this study is its detection of those factors predicting actual behavioral decisions regarding use of fitness and diet apps, as opposed to future intentions [5]

CHAPTER 3

SYSTEM ANALYSIS AND DESIGN

3.1 System Analysis

The Agile Methodology fits the development of this application. The main aim of using this approach is I can focus on each part of the model during development and come back to it if need be. The project can easily de broken down into different parts based on this model. Each iteration involves a team working through a full software development life cycle including planning, requirements analysis, design, coding, and testing before a working product is demonstrated to the client.

This is the model that will be used to develop the Android Application system. However, feedback loops will be allowed during the whole software development process. The model chosen for this project has to favors two developers for a project. Because I am the only ones who are going to implement this project. I find this model suitable for us to follow.

It requires that software development follows the following stages:

- Gathering requirement.
- Designing the requirements.
- Implementation of the features according to the design.
- Integration and testing of the system.
- Reviewing of the system.
- Launching of the system.
- Feedback of the system.

This process has been illustrated below on Figure showing top-down development

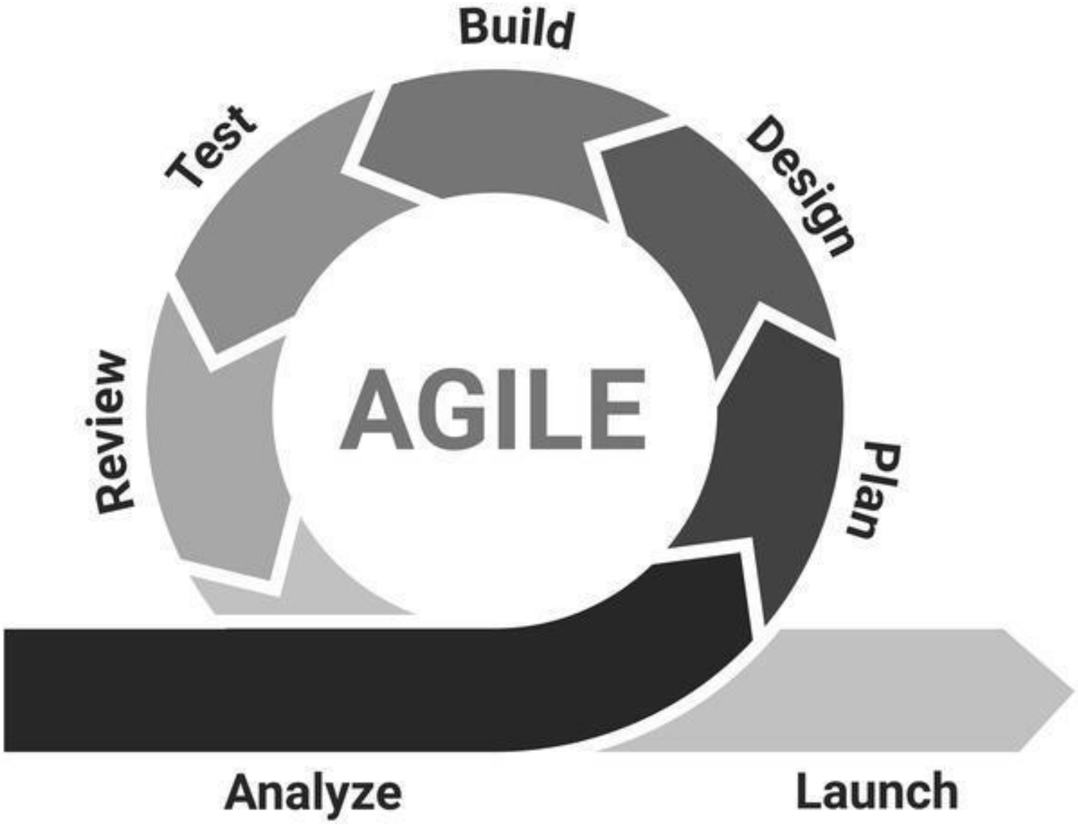


Figure 3. 1 Agile Method software development model

3.1.1 Requirements

i. **Functional Requirement**: The functional requirements in the project are mentioned below.

1. Entry Module

- Users can enter data for each activity.
- Users can create and login to the system using their own created id and password with mail verification.

2. Observation Module

- Users can look at their past data

The functional requirements are mapped using Use Case diagrams as it shows the functions of the system at the most basic level.

A use case diagram is essentially a picture showing system behavior along with the key actors that interact with the system. The use case represents complete functionality. The use case should contain all system activities that have significance to the users. A use case can be thought of as a collection of possible scenarios related to a particular goal, indeed.

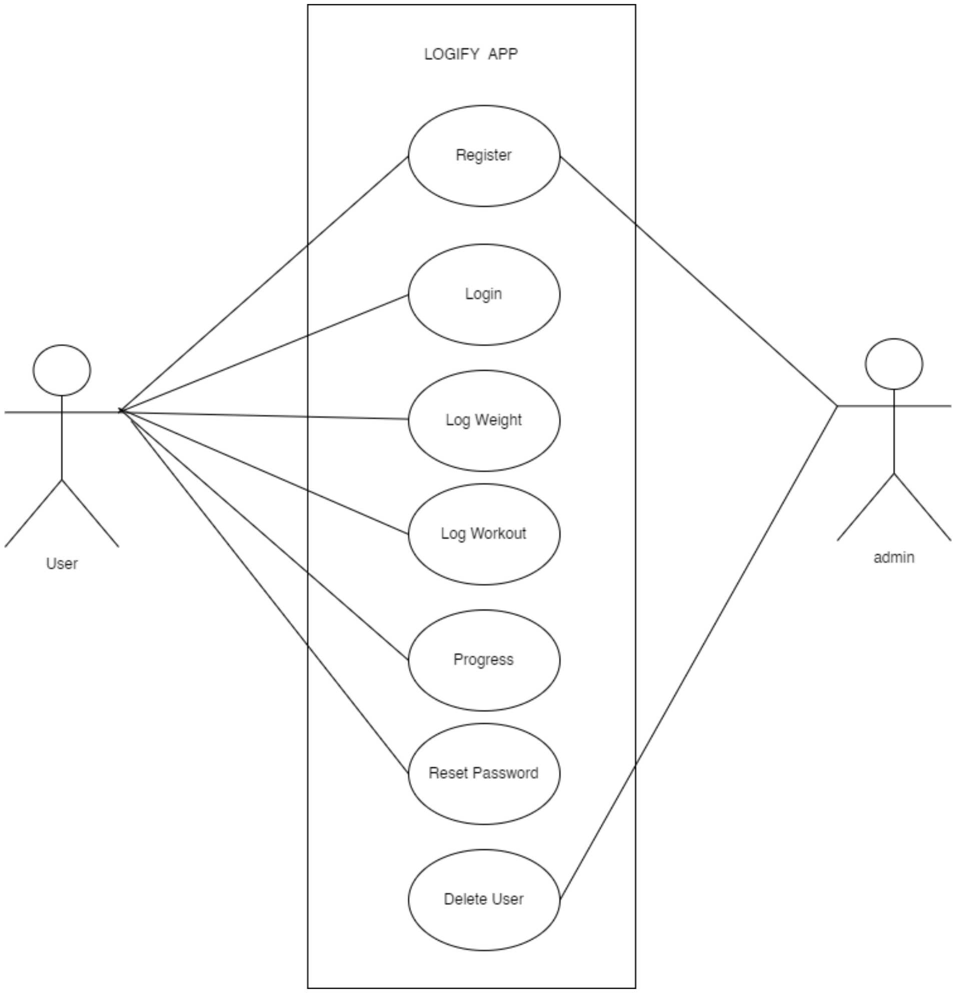


Figure 3. 2 Use Case Diagram

ii. **Non-Functional Requirement**: The functional requirements in the project are mentioned below.

1. System uses different databases for storing the attributes like name, email, exercise weight etc.

3.1.2 Feasibility Analysis

i. Technical Feasibility

- The UI of our application is very simple.
- Users will require internet connectivity and google play store to use our application.
- It will run on 98% of current mobile devices.

ii. Operational Feasibility

These include the reliability, maintainability, usability, supportability. The proposed system is operationally feasible as it is reliable for all type of user i.e. whether or not the user has the knowledge of mobile applications or not. The proposed application is supported on 99% of mobile devices. It is simple and easy to use due to simple user interface and its operational feasible.

iii. Economic Feasibility

Before the development of a system, the proposed system should be studied whether or not it is within the budget estimated by the organization. The project that I am developing is within the cost estimation of the organization. The project cost is less and no more burdens are needed. The system development does not have any requirement of expensive hardware and software. The platform are open sources and the resources required for the project are also open source. Hence the project is said to be economically feasible.

iv. Schedule Feasibility

The system that I developed is scheduling feasible as it does not require more time for the development phase. The data collection takes more time to collect the data about workout types and it optimal sets and repetitions. After data is collected, the other development phase can be within a week. Gantt charts: Gantt chart is a bar chart that provides a visual view of tasks scheduled over time. A Gantt chart is used for planning projects of all sizes, and it is a useful way of showing what work is scheduled to be done on a specific day. It can also help you view the start and end dates of a project in one simple chart. In our project, I used Gantt chart developer for developing the Gantt chart which is shown below in the figure.

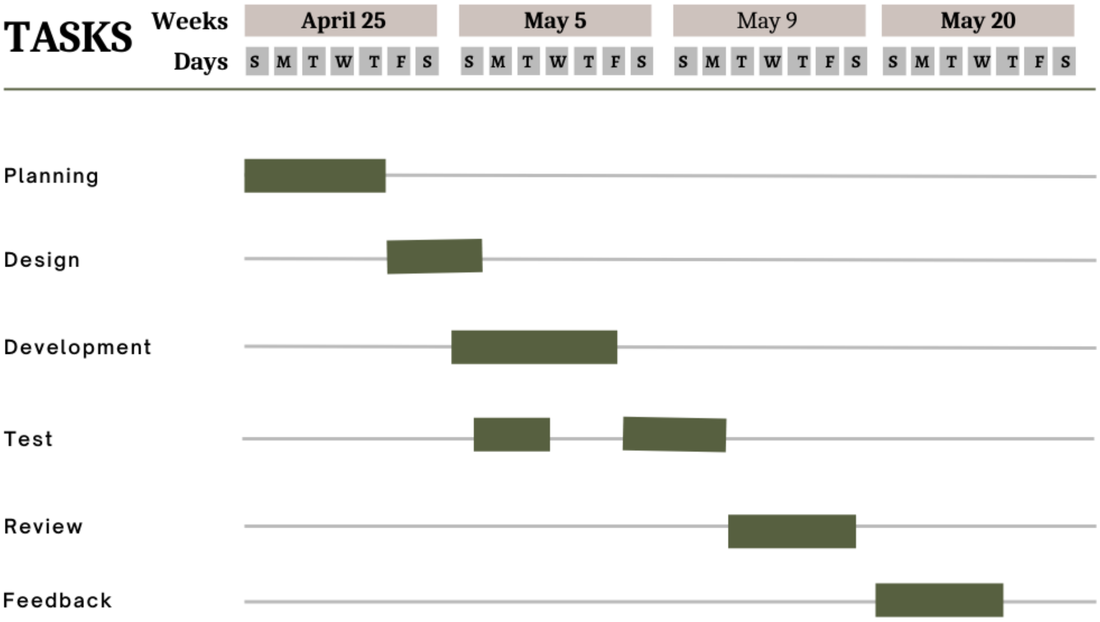


Figure 3. 3 GANTT Chart for the Logify App

3.1.3 Data Modeling

In our Firebase real-time database system user, exercise and weight are the entities which contain a UID for user, Date ID for exercise and weight as a unique identifier for each entity and other attribute to show the properties of these entities. The overall relationship between the entities is shown in the ER diagram below

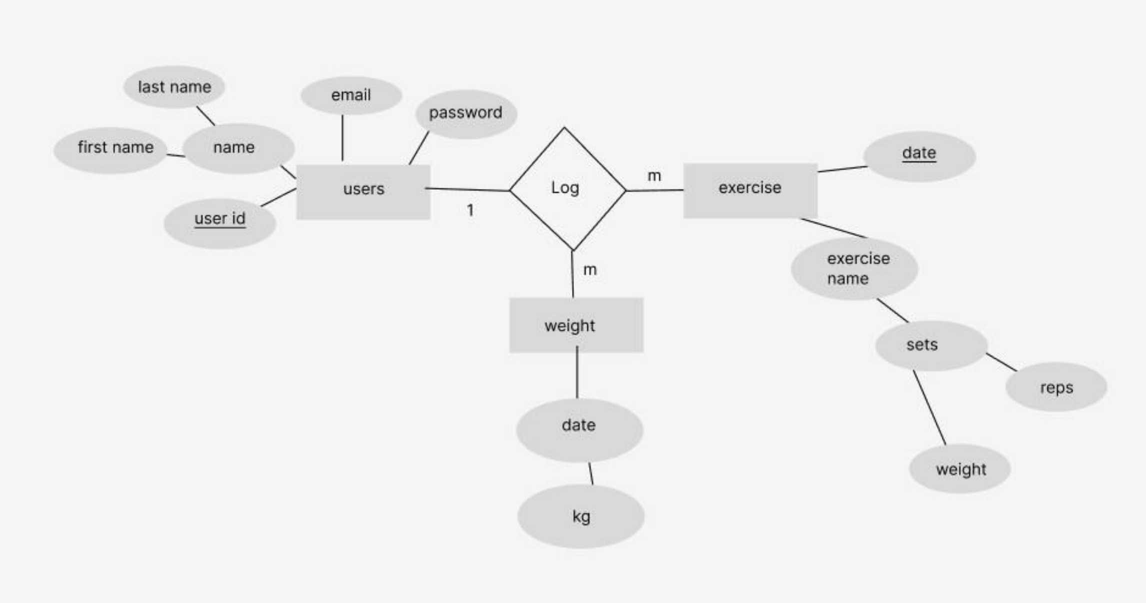


Figure 3. 4 ER-Diagram of Logify Application

3.1.4 Process Modeling (DFD)

Data Flow Diagrams show the flow of data from external entities into the system, and from one process to another within the system. Following are the Data Flow Diagrams for the current system

Context Level Diagram (0 Level)

The 0 Level DFD shows flow of data of application. DFD Level 0 is also called a Context Diagram. It's a basic overview of the whole system or process being analyzed or modeled

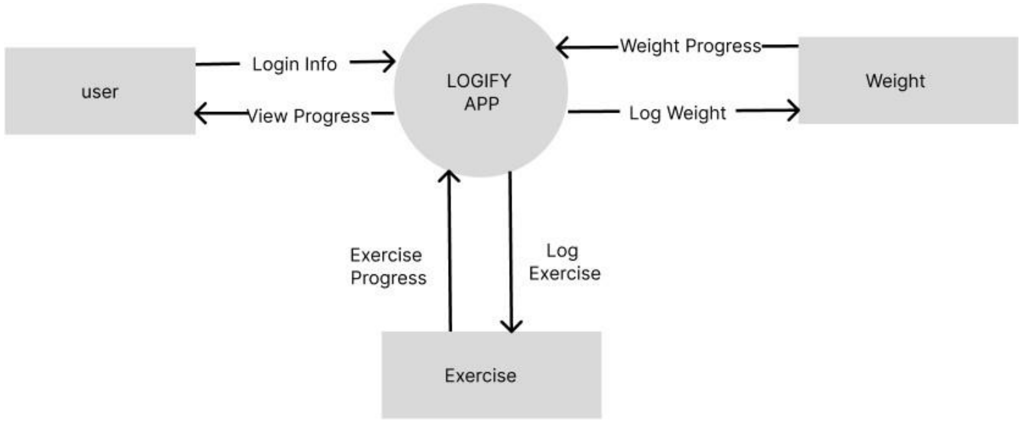


figure3. 5 Context Diagram of Logify App

DFD Level One (Level 1)

DFD Level one provides a more detailed breakout of pieces of the Context Level Diagram. This DFD describes main functions carried out by the system, as I break down the high- level process of the context Diagram into its sub-processes.

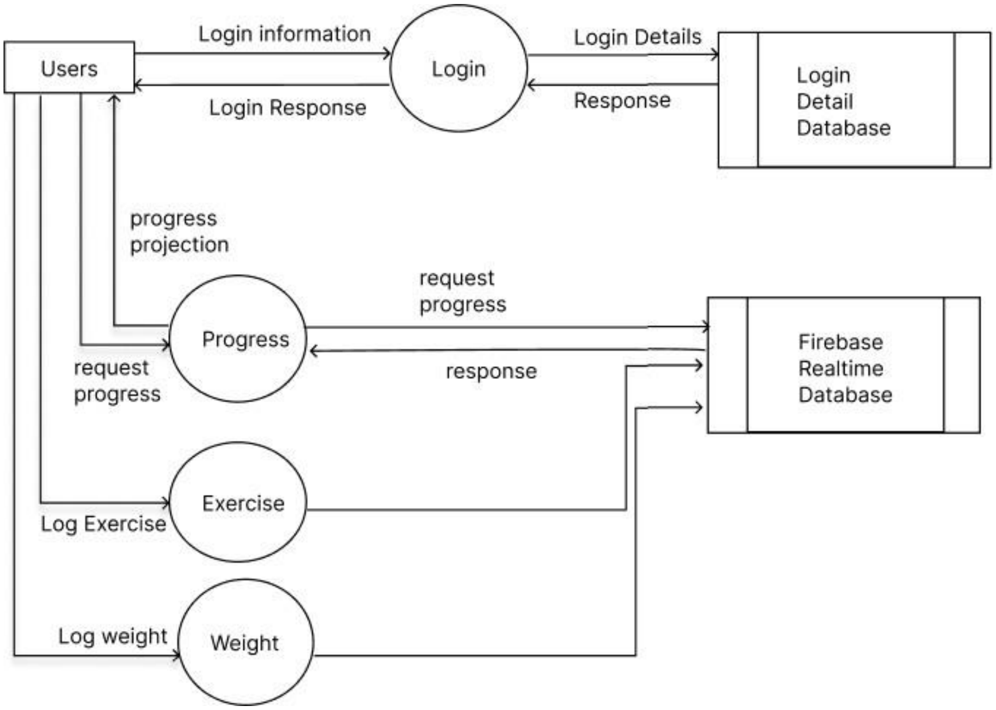


figure3. 6 DFD level 1 Diagram of the Logify

3.2 System Design

3.2.1 Architectural Design

In our application, users interact with the system through a simple user interface. Our application uses three tier architecture. The data is collected from the users and stored in the database through which the server provides the detail to the user on demand. User has a unique account number which makes them different from other users. In this way out system architecture is designed which is an abstract view of the system.

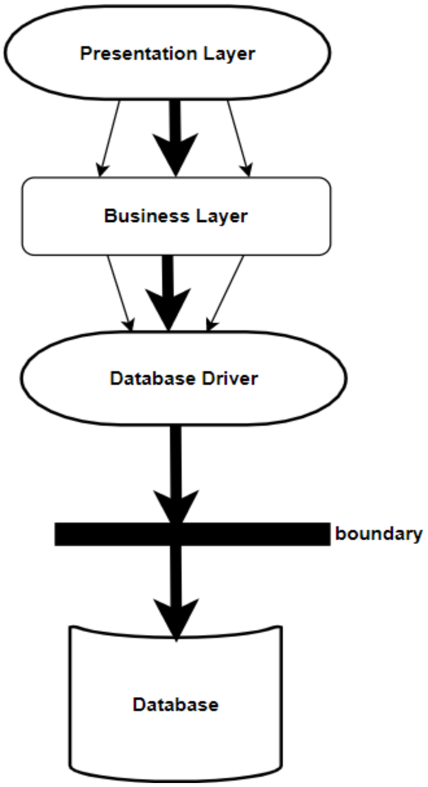


Figure 3. 7 Architectural design of the Logify

3.2.2 Interface Design (UI Interface / Interface Structure Diagrams)

Before implementing the actual design of the project, a few user interface designs are constructed to visualize the user interaction with the system as they browse Home, Register and Login pages. The user interface design will closely follow our Functional Decomposition Diagram show the initial designs of the application.

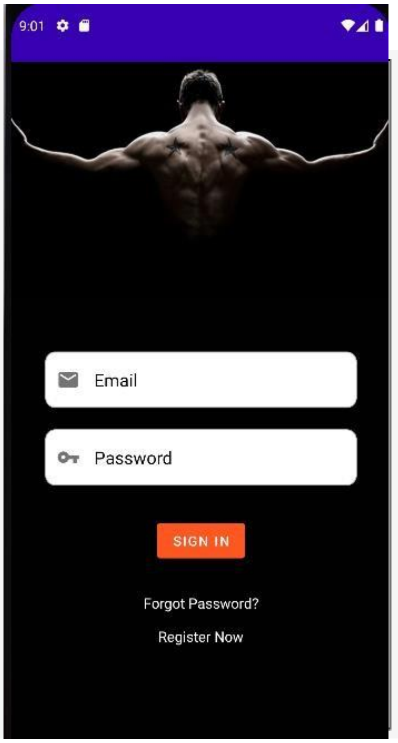


Figure 3. 8 Login page prototype

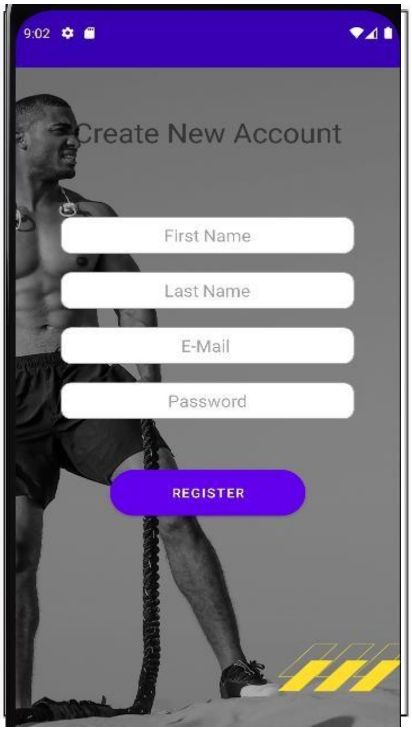


Figure 3. 9 Register Page prototype

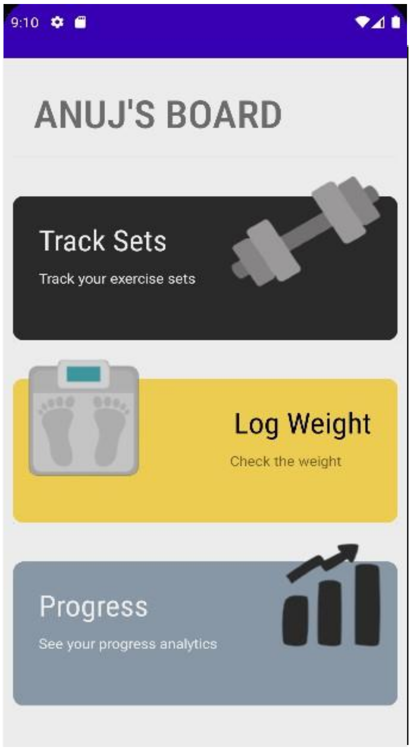


Figure 3. 10 Home Page Prototype

CHAPTER 4

IMPLEMENTATION AND TESTING

4.1 Implementation

Implementation basically means the phase where the system is actually being built. Firstly, all the information that we gathered is studied and analyzed and implemented a system in operation for users. It is one of the most important phases of any project. Implementation usually consists of coding, testing, installation, documentation, training and support. Different tools and technologies that have been used to develop the system which are already discuss in the previous chapter. It is basically converting system design specification into working software.

4.1.1 Tools Used

The various system tools that have been used in developing both the front-end and back-end of the project are being discussed in this chapter.

- **Front-end**

  XML is used for developing the front-end.

- **Back-end**

  The back-end is implemented using JAVA and FIREBASE. FIREBASE is used to design the database.

- **Java**

  Java is a popular programming language, created in 1995. Java is an object oriented language which gives a clear structure to programs and allows code to be reused, lowering development costs.

- **Firebase**

  Firebase is a platform developed by Google for creating mobile and web applications. It was originally an independent company founded in 2011. In 2014, Google acquired the platform and it is now their flagship offering for app development.

4.1.2 Algorithm

User Login Algorithm

```
Start
Enter user login details
IF (not valid)
        Display error and redirect
ELSE IF (details valid)
        Check if username and password exist in database
IF (details exist)
        Allow login and redirect to Home
ELSE
        Show error message
```

Insert Opertaion

The very first insertion creates the tree. Afterwards, whenever an element is to be inserted, first locate its proper location. Start searching from the root node, then if the data is less than the key value, search for the empty location in the left subtree and insert the data. Otherwise, search for the empty location in the right subtree and insert the data.

Algorithm

```
If root is NULL
  then create root node
return
If root exists then
  compare the data with node.data
  while until insertion position is located
    If data is greater than node.data
      goto right subtree
    else
      goto left subtree
  endwhile
  insert data
end If
```

Search Operation

Whenever an element is to be searched, start searching from the root node, then if the data is less than the key value, search for the element in the left subtree. Otherwise, search for the element in the right subtree. Follow the same algorithm for each node.

Algorithm

```
If root.data is equal to search.data
  return root
else
  while data not found
    If data is greater than node.data
      goto right subtree
    else
      goto left subtree
    If data found
      return node
  endwhile
  return data not found
```

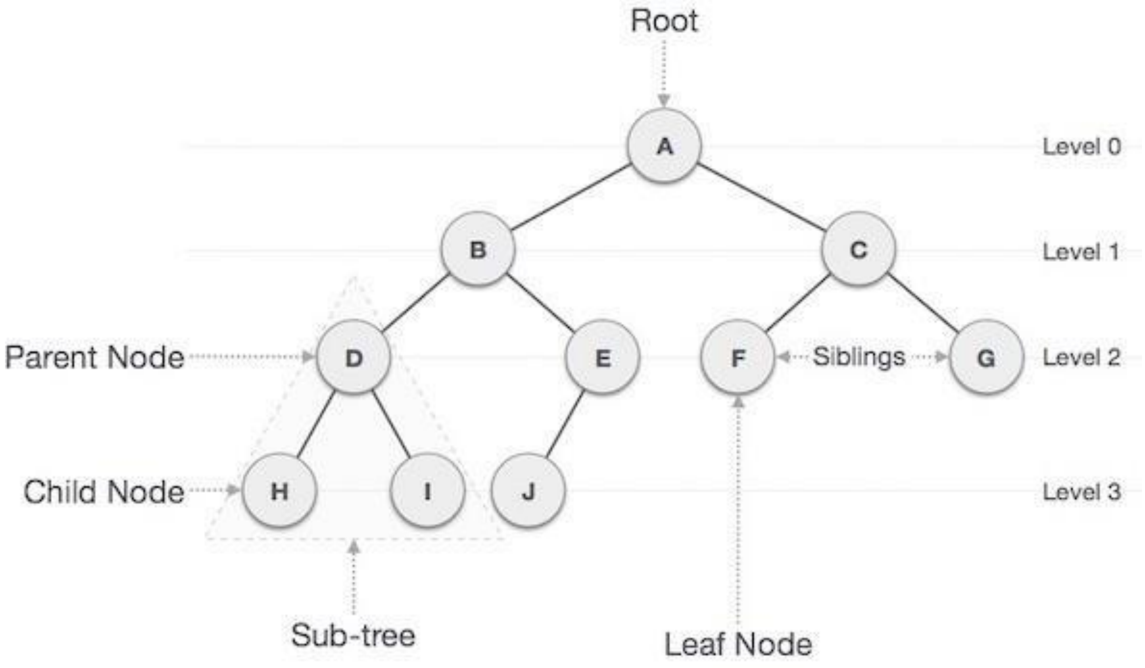


Figure 3. 11 Firebase Tree Structure

- Start by pushing the root node into the queue. Then, remove a node from the front of the queue.
- For every node removed from the queue, push all its children into a new temporary queue.
- Keep on popping nodes from the queue and adding these node’ s children to the temporary queue till queue becomes empty.
- Every time queue becomes empty, it indicates that one level of the tree has been considered.
- While pushing the nodes into temporary queue, keep a track of the sum of the nodes along with the number of nodes pushed and find out the average of the nodes on each level by making use of these sum and count values.

After each level has been considered, again initialize the queue with temporary queue and continue the process till both queues become empty.

4.1.3 Implementation Details of Modules

After the design was made and the problems arising from the design process were clarified and dealt with, it was time to start implementing the application. Implementing application of this scale requires lots of resources and explaining the whole implantation process will not be clarified in this paper. However major important aspects in the implementation will be described. Some of the modules are listed below:

- Register Form: It is used in order to register the new donor to the application. It contains the text field like email, username and other personal details. The information entered is further stored to be used in the other page.
- User Module: It provides information related to the user.

4.2 Testing

Testing is done to check the behavior of a complete and fully integrated software product based on the software requirement specification document. For the application to be deployed it has to be tested. Hence test cases will be written to test this application. We will focus the test cases on functionality, security and performance. So that various types of testing procedures were performed in order to check the working mechanism and correctness of the system. Some of the types of testing that we did are described below:

1. Verify that all the specified fields are present on the registration page.
2. Verify that for better user interface cardview and text fields are displayed wherever.
3. Verify that clicking submits button after entering all the required fields, submits the data to the server.

4. Verify that not filling the optional fields and clicking the submit button will still send data to the server without any validation error.
5. Check validation on numeric fields by entering alphabets and special characters.

Verify that after making a request to the server and then sending the same request again with the same unique key will lead to server-side validation error.

4.2.1 Test Case for System Testing

ID	**Test Case Description**	**Test Data**	**Expected Result**	**Actual Result**	**Pass/Fail**
A_LOG_1	User enters an invalid email	Firstname : anuj Lastname : rana email: anujrana.gmail.com password: 123456	Display message **Please use valid email .**	As expected,	Pass
A_LOG_2	User enters password less than 6 digit	Firstname : anuj Lastname : rana email: anujrana.gmail.com password: 12345	Display message **Invalid length **	As expected	Pass
A_LOG_3	User enters duplicate email	Email : anujrana@gmail.com Password : 123456	User already exists	As expected,	Pass

Table 4 1 Register page test case

ID	**Test Case Description**	**Test Data**	**Expected Result**	**Actual Result**	**Pass/Fail**
A_LOG_1	User enters wrong credentials	Email : babydoll@gmail.com Password : adfasdf	Display message **Couldn't find user **	As expected,	Pass
A_LOG_2	User enters wrong password	Email : anujrana@gmail.com Password : asdasd	Display message **Username or password invalid**	As expected	Pass

Table 4.2 LOGIN page Test case

CHAPTER 5

CONCLUSION AND FUTURE RECOMMENDATION

5.1 Lesson Learnt / Outcome

When this project is completed, the users will be able to register in the application. Users can easily record their data in the application. The record of the data and will be saved in the database and once the user request to view data it will be available promptly with few taps.

5.2 Conclusion

After the successful completion of the Logify Tracking Application, people can easily track and view their activities. The current application has fulfilled all the objectives. We followed the specifications strictly but enhanced some of the features when there was need for it to be done. There have been challenges especially when it came to backend and making sure that the application responses in a predictable manner.

Choosing Java for this project is because it is very simple and easy to use, it could handle a lot of data and easily manipulation compared to another scripting language, this is widely used all over the world. it is Open source; we can freely download and use.

As we came to the end of the project, we realized that there are many enhancements that can be made on the application. Some of these ideas came from those who tested the application and some of them from both of us. We decided to follow the specification because they were realistic to achieve in this given amount of time. Any other enhancements to the application can be done in future development of the application

5.3 Future Recommendation

Here is what can be added in the future on this application to increase its usability, user experience and portability. There is a lot to be done hence this application can be considered as a starting point for something big to come. It will need more time and resources for all these to be done but it is still very realistic and possible to achieve.

- Addition of graph chart for progress tracking,
- Addition of more exercise and sets,
- Access the applications on multiple devices and website,
- Advanced result set.

REFERENCES:

[1] A. Muntaner-Mas, "A Systematic Review of Fitness Apps and Their Potential Clinical and Sports Utility for Objective and Remote Assessment of Cardiorespiratory Fitness," *A Systematic Review of Fitness Apps and Their Potential Clinical and Sports Utility for Objective and Remote Assessment of Cardiorespiratory Fitness,* p. 49, 2019.

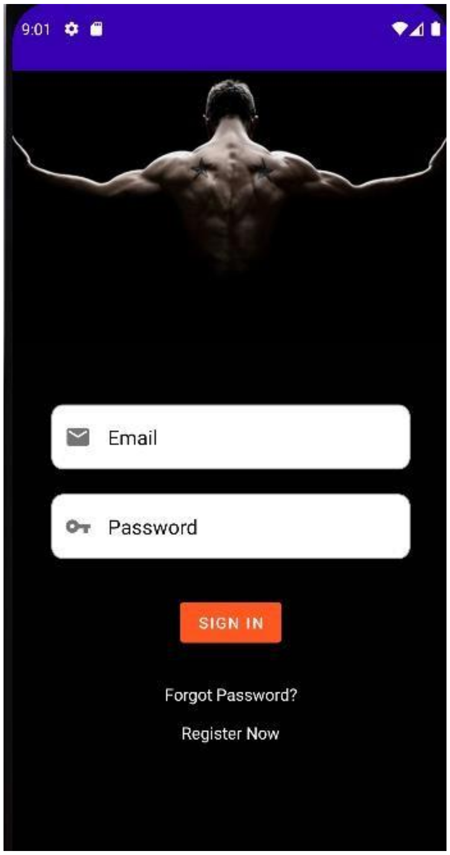
[2] J. E.Barkley, "Computers in Human Behavior," *The relationship between fitness app use and physical activity behavior is mediated by exercise identity,* vol. 108, no. 1, 2020.

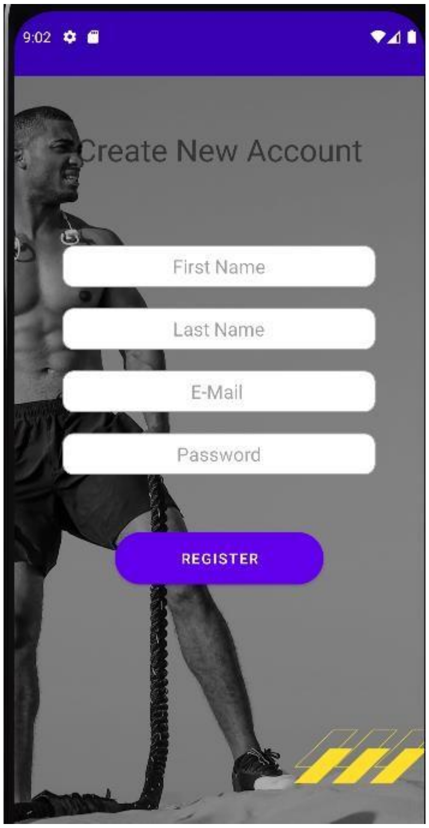
[3] P. L. &. V. Gay, "Mobile apps for chronic diseasemanagement: lessons learned frommyFitnessCompanion," *Mobile apps for chronic diseasemanagement: lessons learned frommyFitnessCompanion,* vol. 3, 2013.

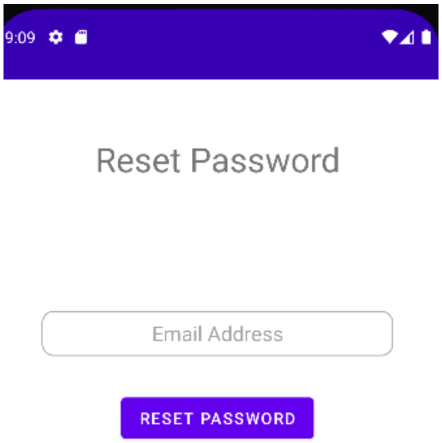
[4] A. Muntaner-Mas, "The effects of a physical activity intervention based on a fatness and fitness smartphone app for University students," *The effects of a physical activity intervention based on a fatness and fitness smartphone app for University students,* vol. 1, p. 215, 2017.

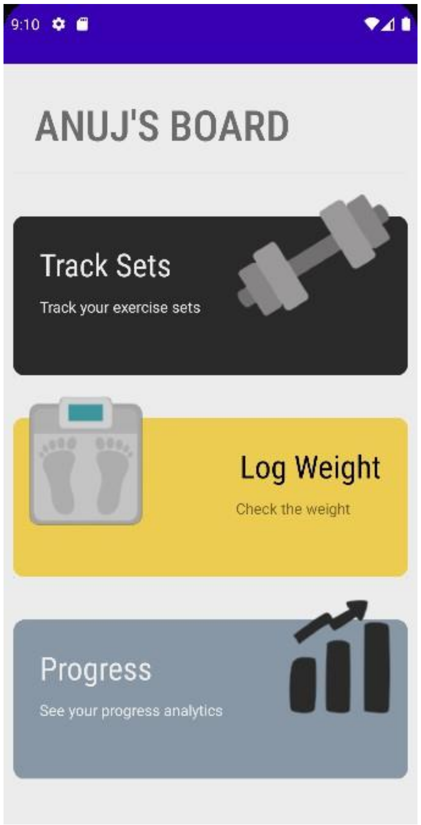
[5] T. a. Informatics, "Telematics and Informatics," *Personal and social predictors of use and non-use of fitness/diet app: Application of Random Forest algorithm,* vol. 55, 2020.

Appendices: [A]









9:10

ANUJ'S BOARD

Track Sets

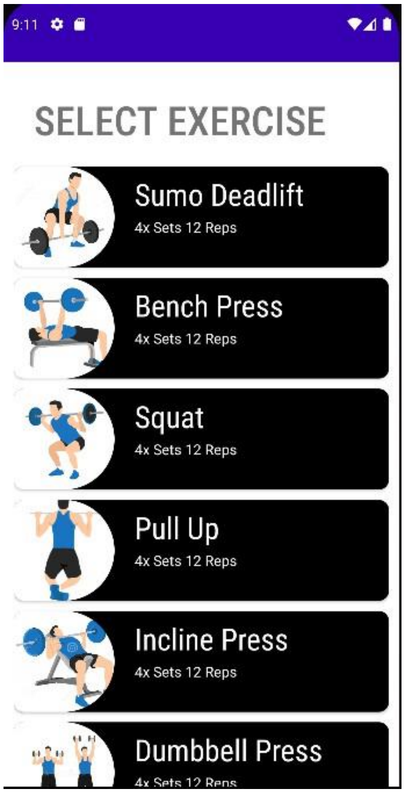
Track your exercise sets

Log Weight

Check the weight

Progress

See your progress analytics



9:11

SELECT EXERCISE

Sumo Deadlift

4x Sets 12 Reps

Bench Press

4x Sets 12 Reps

Squat

4x Sets 12 Reps

Pull Up

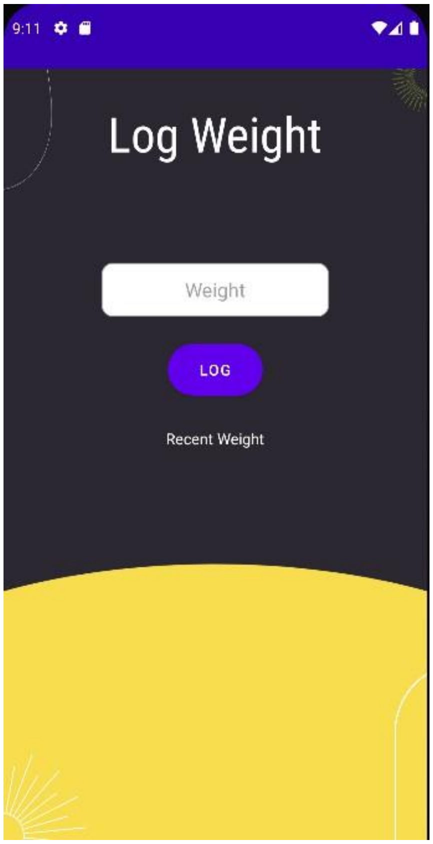
4x Sets 12 Reps

Incline Press

4x Sets 12 Reps

Dumbbell Press

4x Sets 12 Reps



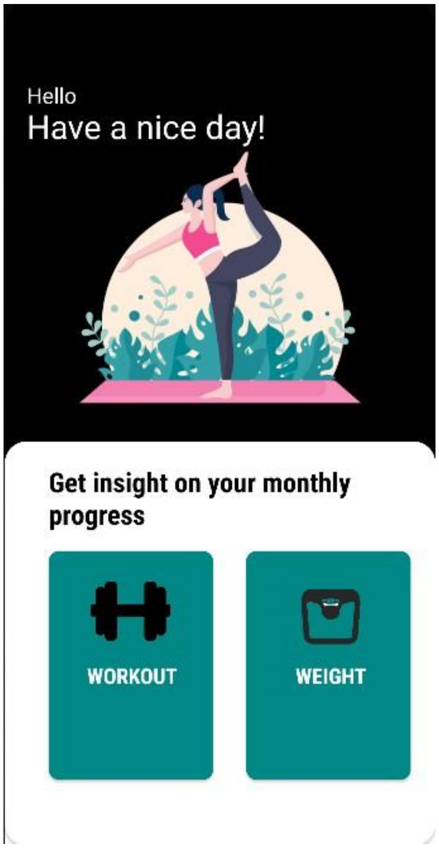
9:11

Log Weight

Weight

LOG

Recent Weight



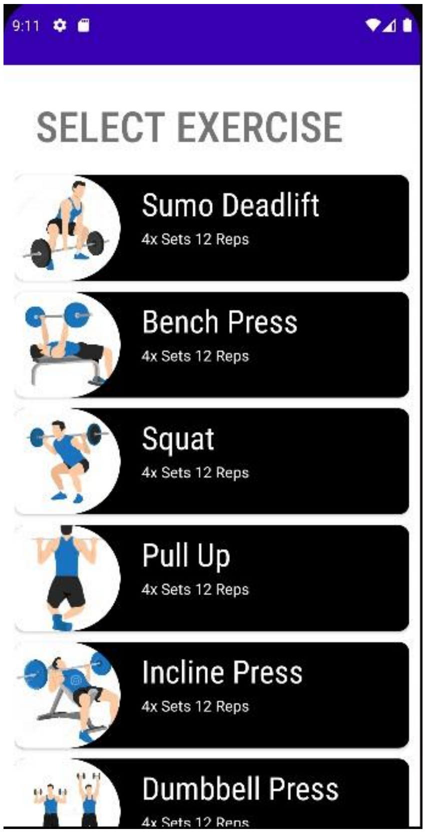
Hello

Have a nice day!

Get insight on your monthly progress

WORKOUT

WEIGHT



SELECT EXERCISE

Sumo Deadlift
4x Sets 12 Reps

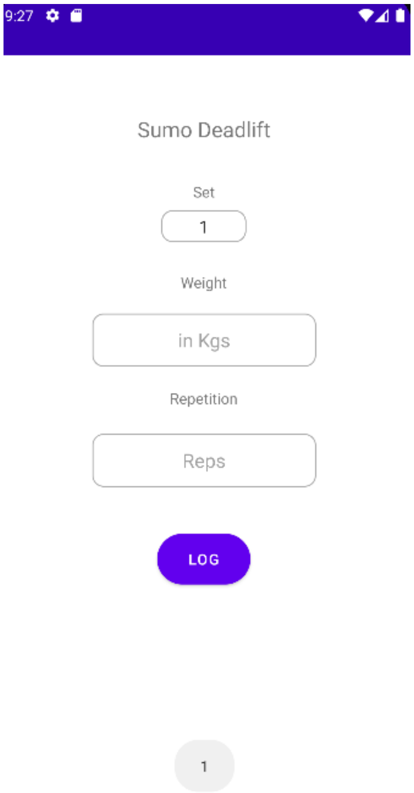
Bench Press
4x Sets 12 Reps

Squat
4x Sets 12 Reps

Pull Up
4x Sets 12 Reps

Incline Press
4x Sets 12 Reps

Dumbbell Press
4x Sets 12 Reps

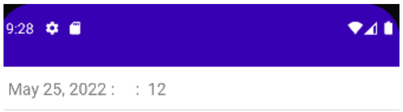


Sumo Deadlift

Set: 1

Weight: in Kgs

Repetition: Reps



May 25, 2022 : : 12