A Mini Project Report

on

STAY FIT - A WEB APPLICATION WHICH TRACKS YOUR FITNESS

By

A.V.L.Likhita

(1602-18-733-058)

Department of Computer Science & Engineering

Vasavi College of Engineering (Autonomous)

(Affiliated to Osmania University)

Ibrahimbagh, Hyderabad-31

2020-21

ACKNOWLEDGEMENT

We would like to express our heartfelt gratitude to Mr. Sashi Kumar,(Department of CSE) our project guide, for his valuable guidance and constant support, along with her capable instruction and persistent encouragement.

We are grateful to our Head of Department, Dr T. ADILAKSHMI, for her steady support and the provision of every resource required for the completion of this project.

We would like to take this opportunity to thank our Principal, Dr S. V. RAMANA, as well as the management of the institute, for having designed an excellent learning atmosphere.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

16	Count
17	Shows the day of workout
18	Bar graphs of sets,reps
19	List of all workouts
20	Representation of muscles
21	Selection of metrics
22	Go to recommendation page
23	User registration and login page
24	Selection of subscription plans(After registration)
25	After successful login
26	Basic plan tutorials
27	Youtube modal for workout tutorials
28	Standard plan home page
29	Premium plan home page
30	Workout recommendations home page
31	Food recommendations home page

48	Add tutorials
49	Add a new subscription plan
50	List of all workout
51	List of all food items available in the database
52	List of tutorials
53	Available subscription plans other than basic ones
54	User’s details
55	User’s feedback
56	Payment gateway(paypal)
57	Cart (plans)
58	Payment successful

1. STAY FIT

1.1 Introduction

Stay Fitis a web application for analysing and logging strength training and body building data.Stay Fit has two different portals i.e., exercise portal and recommendations portal. Apart from this, it also includes an admin portal.

Exercise portal aims to present workout data in a way that highlights and encourages progressive overload and long-term commitment. There are many workout tracking apps already available, what sets this one apart is the focus on data. If you want a workout application that guides you through a routine this is not the tool for you. Exercise portal has no server and works completely offline using Indexed DB.

On the other hand, if you manage the routine yourself and track your workouts in a spreadsheet with a bunch of formulas and charts, you are probably the kind of person who would like this application.

Recommendations portal provides 3 different subscription plans (Basic, Standard, Premium) where the user can choose their plan according to their requirement. These plans provide basic workout tutorials, workout recommendations and food recommendations according to the user's weight.

These plans recommend a particular type of workout according to his/her weight and the same follows with food recommendations i.e., for Breakfast, Lunch and Dinner based on the calorie intake.

1.2 Motivation

Statistics have shown that the fitness apps influence the gym athletes a lot, these give them motivation to work out a lot and compare it with their friends and family. So such websites have a lot of scope for people who tend to set their goals daily and workout and diet is as important as the workout you do. Many people don't really know how to have a balanced diet and it becomes very difficult for the people to reach their goal.

Hence, building such a web application would be very useful and can get many real time users.

1.3 Objective

This STAY FIT is a web application intended to provide a complete solution for the users. It will enable users to create an account and add their workouts for a particular day and check the progress of their workouts. They can also download their list of workouts at the end of the day. Workouts are broken down into sets of exercises.

Each set is associated with *exactly one* exercise. Each set is performed at one specific weight. Each set consists of one or more reps (repetitions) of an exercise at that weight. Not all exercises are based on lifting weight for certain reps. Some are time-based, but they are still grouped into sets.

The main objective of this recommendation's portal is to make it easy for the users to choose a perfect diet and proper workouts. It provides 3 different subscription plans (Basic, Standard, Premium) where the user can choose their plan according to their requirement. These plans provide basic workout tutorials, workout recommendations and food recommendations according to the user's weight.

1.4 Product Scope

The purpose of this web application is to ease tracking of the fitness of a particular user and to create a platform for the users who want to keep track of their fitness and diet. So, it has two phases i.e., portals one is exercise portal and another one is recommendations portal. In the exercises portal the users can add their workouts each day and keep track of which muscles they worked on; it also helps you to compare your day-to-day progress. In the recommendation's portal, the user can get the workout recommendations and diet to be followed based on his/her weight.

Meet all the objectives and implementation of all the modules in a planned way.

2. FEATURES

2.1 Exercise portal

1. Simple workout data entry supporting reps, sets, weight, duration, warmup and failure sets
2. Exercise Search
3. Supports multiple workouts per day
4. Track and visualise personal records
5. Lots of Charts
6. Musculature visualizations (using SVG files)
7. Intensity and volume calculations
8. Import from CSV
9. Export to JSON
10. Proper time zone support (important for people who travel)
11. Workouts portal works offline, *no network connection required*

2.2 Recommendation's portal

1. Provides different subscription plans for users (Basic, Standard, Premium) etc., paid through payment gateway (Integrated PayPal).
2. Basic plan provides workout tutorials
3. Standard plan provides workout recommendations including features of basic plan.
4. Premium plan includes food recommendations including features of standard
5. A chatbot for conducting an online chat conversation via text.

2.3 Admin portal

1. Admin portal for providing recommendations on daily basis.
2. Dashboard
3. Perform certain actions like insertion, deletion and updating.
4. Workout and food recommendations, tutorials.

3. LITERATURE SURVEY

Healthy eating and fitness web applications like STAY FIT are designed to promote healthier living. However, for young people, body dissatisfaction is commonplace, and these types of web applications/apps can become a source of maladaptive eating and exercise behaviours. Furthermore, such applications are designed to promote continuous engagement, potentially fostering compulsive behaviours.

As our project has two phases the first one is workouts progress and the second one being recommendations part (both food i.e., a proper diet plan and workouts). We looked at researches on both the scenarios.

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6604512/

The above research given was conducted in two phases. Through a survey and 2 workshops, this paper contains information on understanding of young people's perceptions of healthy eating and fitness apps and any potential harm that their use might have and also explored these further through interviews with experts in eating disorder and body image. Using insights drawn from this initial phase. This paper also have a review of the top 100 healthy eating and fitness apps on the Google Play Store to find out whether or not apps on the market have the potential to elicit maladaptive eating and exercise behaviours.

The results of this paper shows that participants were aged between 18 and 25 years and had current or past experience of using healthy eating and fitness apps. Almost half of our survey participants indicated that they had experienced some form of negative experiences and behaviours through their app use. The app review identified logging functionalities available across the apps that are used to promote the sustained use of the app. However, a significant number of these functionalities were seen to have the potential to cause negative experiences and behaviours. This paper highlights necessity for careful considerations around the design of apps that promote weight loss or body modification through fitness training and provide a proper diet, especially when they are used by young people who are vulnerable to the development of poor body image and maladaptive eating and exercise behaviours.

4. SOFTWARE REQUIREMENT SPECIFICATION

4.1 Definitions, Acronyms and Abbreviations

SQL -> Structured Query Language

ERD -> Entity Relationship Diagram

UML -> Unified Modelling Language

SRS -> Software Requirement Specification

DFD -> Data Flow Diagram

4.2 Overall Descriptions

4.2.1User Classes and Characteristics

The portal displays data based on user entitlements. Here userscan have accessto both exercise and recommendation's portal. Specific subscription plan to be selected for recommendation's portal. User can get few graphs along with duration time, intensity, and volume calculations. Other than these, users can do the following

- Look at the features in the home page
- Users can sign up if they are using Stayfit for the first time. Regular users can just login.
- Users can view muscular visualisations
- Users can view duration time, intensity, and volume calculations in form of graphs.
- Users can select from different subscription plans and pay using paypal gateway.
- Users can use the users friendly chat application for clarifying user's doubts.
- Admin can make changes regularly.
- Admin can add/delete/update recommendations and plans.
- Admin can view the list of users and their feedbacks.
- Admin can delete users if they use the website in the wrong way or delete users if they try to post malicious content.

4.2.2 Operating Environment

This application will be operating in webpages through browser. This web application can be operated through any browser supporting java script. The only requirement to use this website would be an internet connection and a device which has access toa browser which supports java script and Indexed DB.

4.2.3 Assumptions and Dependencies

The assumptions are:

- The coding should be error free.
- The application should be user-friendly so that it is easy to use.
- Valid information of every user must be stored in database.
- The system should provide more storage capacity and provide fast access to the database.
- The sever running 24 hours a day.
- Users may access from any browser that has Internet browsing capabilities and supports java script and Indexed DB.

The dependencies are:

- The specific hardware and software due to which the product will be run.
- Based on listing requirements and specification the project will be developed and run.
- The end users should have proper understanding of the product.
- The information of all the users must be stored in a database that is accessible by the college campus system.
- Any update regarding the user profile and any other updation is to be recorded to the database and the data entered should be recorded.

4.2.4 Requirements

4.2.4.1 Software Requirements

Front-end:HTML 5

HTML 5

HTML5 is a markup language used for structuring and presenting content on the World Wide Web. It is the fifth and last major HTML version that is a World Wide Web Consortium (W3C) recommendation.

jQuery

jQuery is a JavaScript library designed to simplify HTML DOM tree traversal and manipulation, as well as event handling, CSS animation, and Ajax.

Javascript (for validations)

JavaScript(often shortened to **JS**) is a lightweight, interpreted, object-oriented language with first-class functions, and is best known as the scripting language for Web pages, but it's used in many non-browser environments as well.

Styling: CSS, Bootstrap

Bootstrap

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates

Server-side: PHP (For server-side connection)

PHP is a server scripting language, and a powerful tool for making dynamic and interactive Web pages.PHP is a widely used, free, and efficient alternative to competitors such as Microsoft's ASP.PHP 7 is the latest stable release.

Database: MySQL Database

MySQL Database

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Wideness's daughter, and "SQL", the abbreviation for Structured Query Language

Tools & IDE's: Sublime IDE, Xampp for MySQL

4.2.4.2 Hardware Requirements

Operating System: Windows 10

Processor: Core i5 Processor

Clock speed: 2.5GHz

Monitor: 1920* 1080 Resolution Colour

Keyboard: QWERTY

RAM: 8 GB

4.2.5 Data Requirement

The inputs consist of the query to the database and the output consists of the solutions for the query. The output also includes the user receiving the details of their accounts. In this project the inputs will be queries as fired by the users like create an account. Now the output will be visible when the user requests the server to get details of their profile and all the posts.

4.3 Other Non-functional Requirements

4.3.1 Performance Requirement

The proposed system that we are going to develop is expected to target huge audience. Therefore, it is expected that the database is functional and can handle all the requests given by the users.

The performance of the system should be fast and accurate.

The System shall handle expected and non-expected errors in a way that prevent loss in information and long downtime period. Thus, it should have inbuilt error testing to identify search or data check/fetch.

The system should be able to handle large amount of data. Thus, it should accommodate large number of requests without any fault.

4.3.2 Satisfy Requirement

The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take a database backup.

4.3.3 Security Requirement

- System will use secured database.
- Normal users can just read instruction and use this system, but they cannot edit or modify anything except their personal and some other information.
- System will have two type of users and user has access constraints.
- Proper user authentication should be provided.
- No one should be able to hack user's details or any other information.
- There should be separate part for users that no users can access the database and only admin has the rights to update the database.

4.3.4 Requirement attributes

- There may be multiple admins creating the project, all of them will have the right to create changes to the system. But the users cannot do the changes.
- The project should be open source.
- The quality of the database is maintained in such a way so that it can be very user friendly to all the users of the database.
- The user be able to easily access this system from anywhere at any time.

4.3.5 Business Rule

A business rule is anything that captures and implements business policies and practices. A rule can enforce business policy, decide, or infer new data from existing data. This includes the rules and regulations that the System users should abide by. This includes the cost of the project and the discount offers provided. The users should avoid illegal rules and protocols. Neither admin nor member should cross the rules and regulations.

4.3.6 User Requirement

The users are assumed to have basic knowledge of the computers and internet browsing. The administrators of the system should have more knowledge of the internals of the system and is able to rectify the small problems that may arise due to disk crashes, power failures and other catastrophes to maintain the system. The proper user interface, user manual, online help, and the guide to install and maintain

the system must be sufficient to educate the users on how to use the system without any problems.

5. SYSTEM ARCHITECTURE

5.1 Schema Diagram

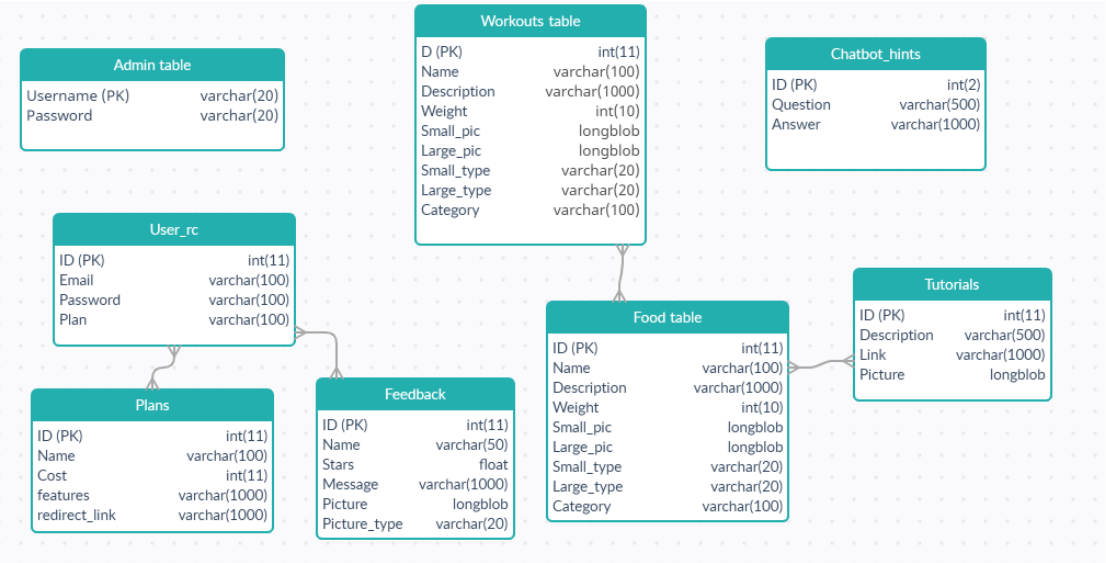

Fig 1: Database scheme

5.2 Database Tables

Table 1: Admintable

COLUMN NAME	DATATYPE & SIZE	CONSTRAINTS	DESCRIPTION
Username	Varchar(100)	Primary Key	Username of admin
Password	Varchar(20)	Not null	Password of admin

Table 2: User_rc

COLUMN NAME	DATATYPE & SIZE	CONSTRAINTS	DESCRIPTION
ID	int(11)	Primary Key	The ID accepts the integer which identify the user

Email	varchar(100)	Not null	Email of the user
Password	varchar(100)	Not null	Password of the user
Plan	varchar(100)	Not null	The subscription which the user opted.

Table 3: Food table

COLUMN NAME	DATATYPE & SIZE	CONSTRAINTS	DESCRIPTION
ID	int(11)	Primary Key	The ID accepts the integer which identify the food item
Name	varchar(100)	Not null	Name of food item
Description	varchar(100)	Not null	Description of the food item
Weight	int(100)	Not null	Weight of the person who can eat the food item
Small_pic	longblob	Not null	Small size pic of food
Large_pic	longblob	Not null	Large picture of food item
Small_type	varchar(20)	Not null	Type of small picture
Large_type	varchar(20)	Not null	Type of large picture
category	varchar(100)	Not null	Category of food item

Table 4: Workouts table

COLUMN NAME	DATATYPE & SIZE	CONSTRAINTS	DESCRIPTION
id	int(11)	Primary Key	The ID accepts the integer which identify the workout

name	varchar(100)	Not null	Name of the workout
description	varchar(100)	Not null	Description of the workout
weight	int(100)	Not null	Weight of the person who can perform the workout
small_pic	longblob	Not null	Small size pic of workout
large_pic	longblob	Not null	Large picture of workout
small_type	varchar(20)	Not null	Type of small picture
large_type	varchar(20)	Not null	Type of large picture
category	varchar(100)	Not null	Category of workout

Table 5: Tutorials

COLUMN NAME	DATATYPE & SIZE	CONSTRAINTS	DESCRIPTION
ID	int(11)	Primary Key	The ID accepts the integer which identify a specific tutorial
Description	varchar(500)	Not Null	Short note of youtube video
Link	varchar(10000)	Not Null	iframe link from youtube
Picture	Longblob	Not Null	Sample picture for youtube video
Picture_type	varchar(25)	Not Null	Type of picture

Table 6: Plans

COLUMN NAME	DATATYPE & SIZE	CONSTRAINTS	DESCRIPTION
ID	int(11)	Primary Key	The ID accepts the

			integer which identify a specific plan
Name	varchar(100)	Not Null	Name of the subscription plan
Cost	int(11)	Not Null	Price of the subscription plan
Features	varchar(1000)	Not Null	Features related to the plan
Redirect_link	varchar(1000)	Not Null	Home page of specific plan

Table 7: Feedback

COLUMN NAME	DATATYPE & SIZE	CONSTRAINTS	DESCRIPTION
ID	int(11)	Primary Key	The ID accepts the integer which identify a specific plan
Name	varchar(10)	Not null	Name of the user who have given feedback
Stars	float	Not null	Number of stars
Message	varchar(1000)	Not null	Feedback given by the user
Picture	longblob	Null	Profile picture of the user
Picture_type	varchar(20)	Not null	Type of picture(jpg/png)

Table 8: Chatbot_hints

COLUMN NAME	DATATYPE & SIZE	CONSTRAINTS	DESCRIPTION
id	int(2)	Primary Key	The ID accepts the

			integer which identify chatbot question and answers
question	varchar(500)	Not null	Questions related to fitness and diet
reply	varchar(1000)	Not null	replies to the respective questions

Indexed DB:

Key: id
Value: all other metrics

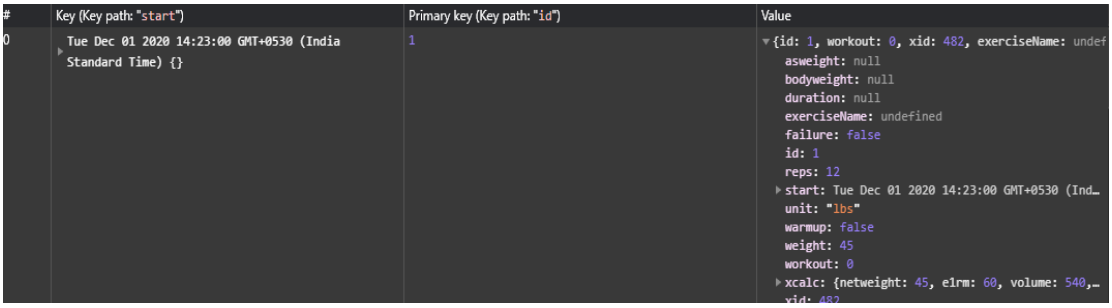

Fig 2: Indexed DB

5.3 Unified Modelling Language (UML) Diagram

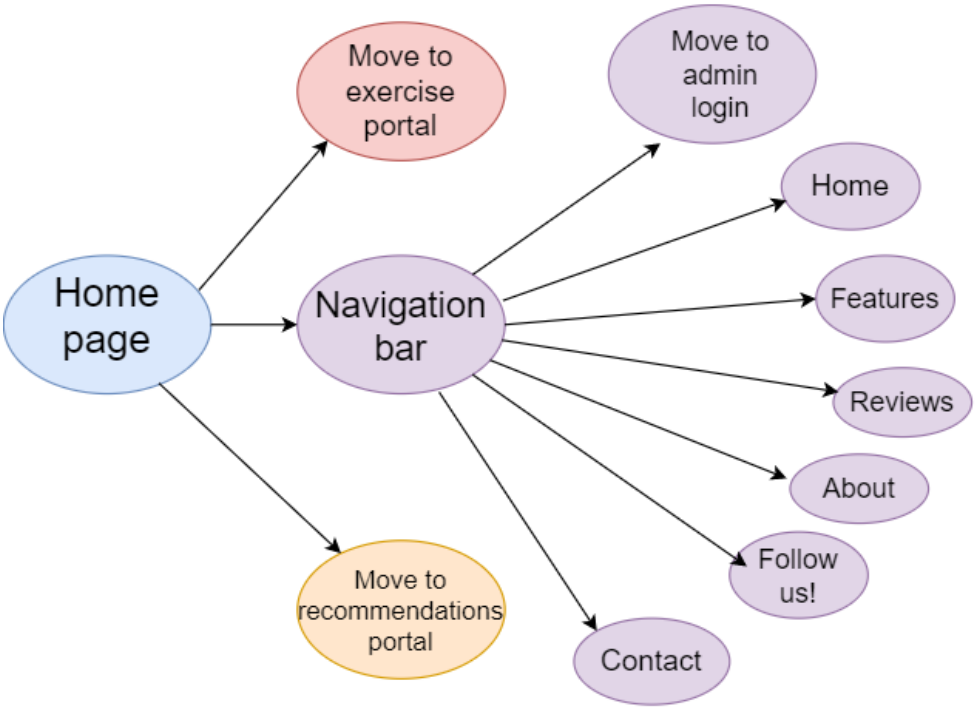

Fig 3: Home page

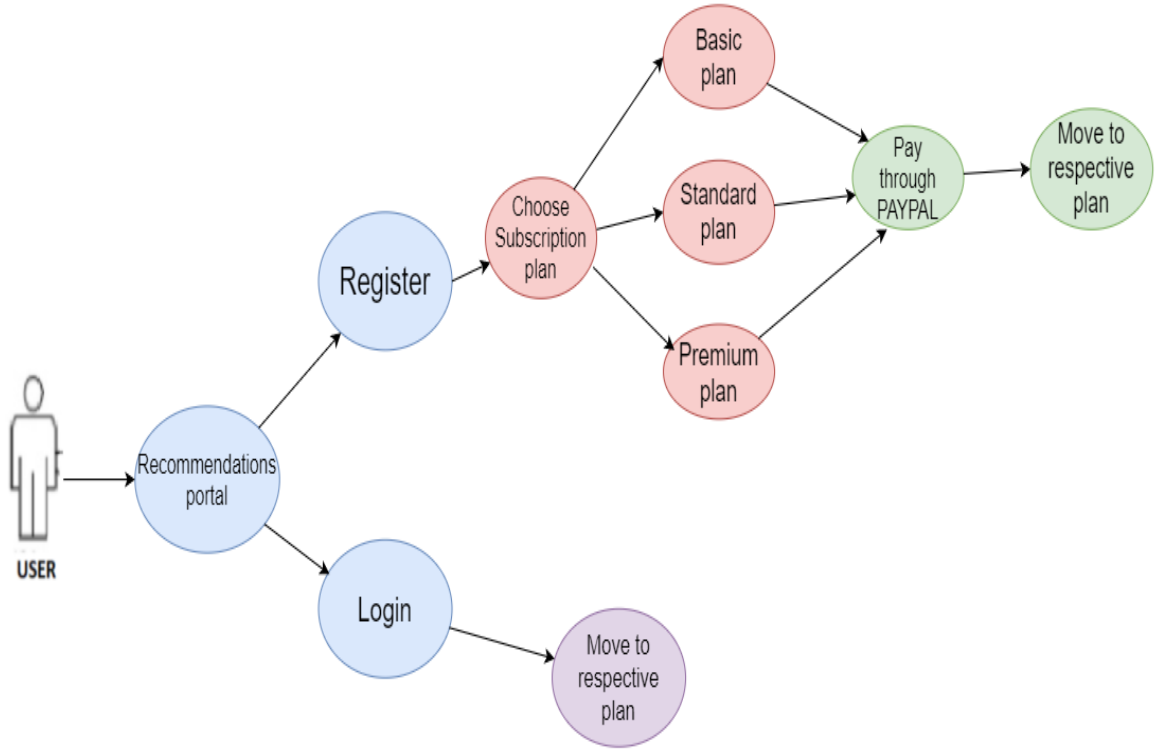

Fig 4: Recommendations portal

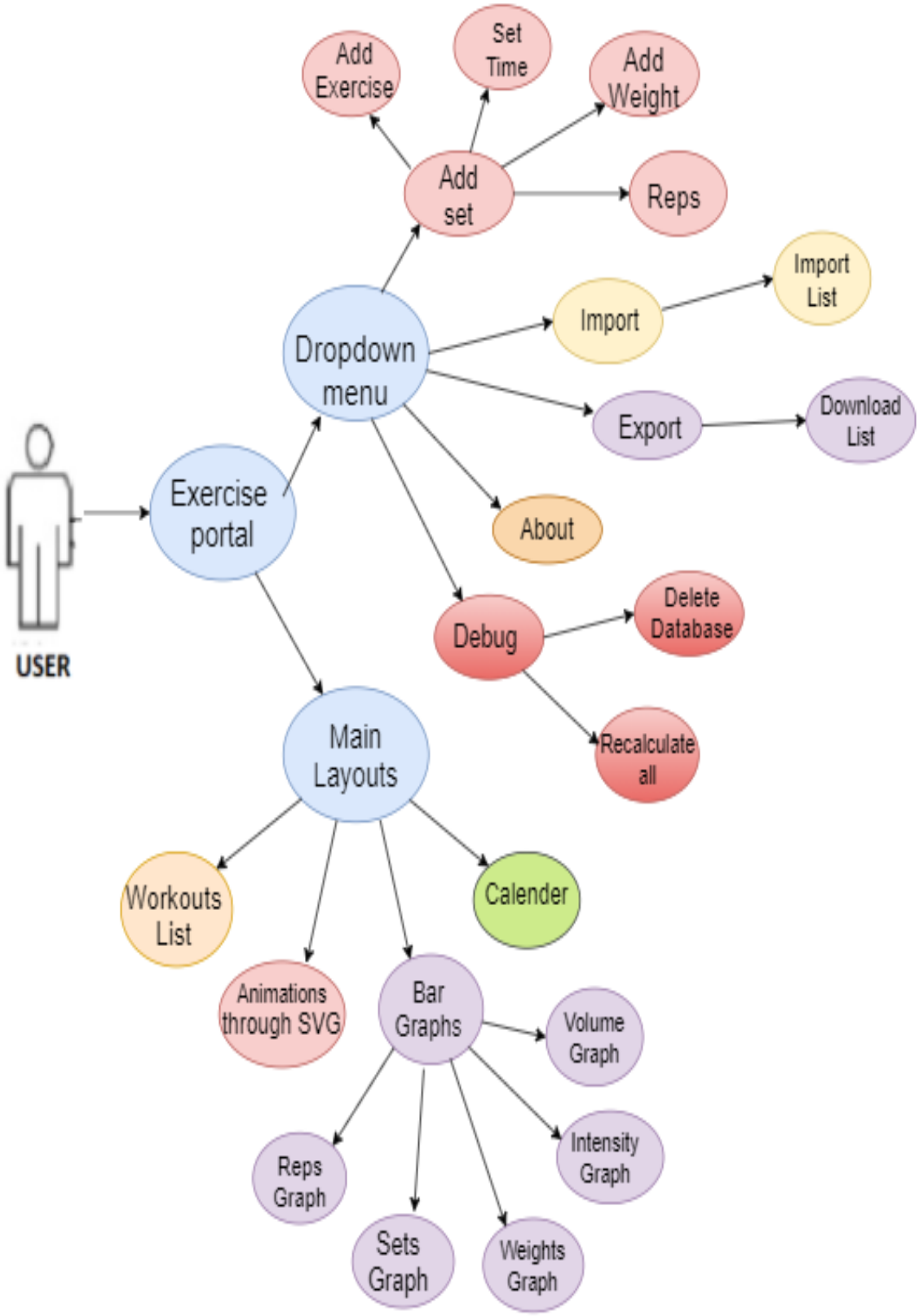

Fig 5: Exercise Portal

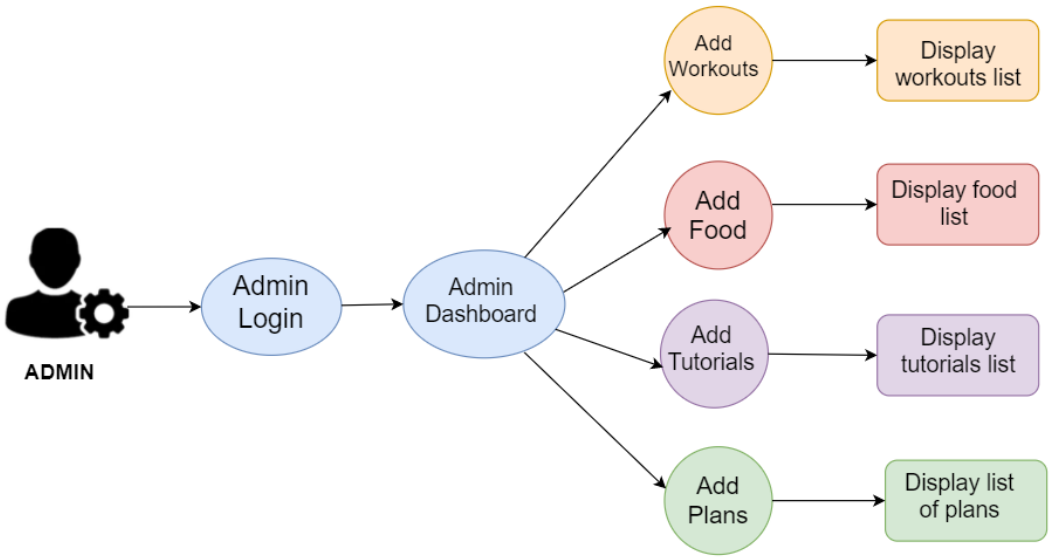

Fig 6: Admin portal

6. CODE IMPLEMENTATION

Link for github repository: https://github.com/likhitaavl2k/Stay-fit--A-web-application-which-tracks-your-fitness.git

Access.html (To choose admin/user)

```
<!DOCTYPE html>
<html>
<head>
	<title>Admin/User</title>
	<link rel="stylesheet" type="text/css" href="css/access.css">
	<link rel="stylesheet" type="text/css"
href="https://stackpath.bootstrapcdn.com/font-awesome/4.7.0/css/font-
awesome.min.css">
</head>
<!-- background-image: url(images/accessbg2.jpg) -->
<header style="position: absolute;top: 1%;">
	<a href="index.php"><i class="fa fa-home fa-5x" ></i></a>
</header>

<body style="background-repeat: no-repeat;background-size: cover;">
	<div style="padding-left: 300px;padding-top: 100px;" id="admindiv">
		<div class="vl" ></div>
		<a href="adminLogin.html"><imgsrc="images/female.png" ></a>
		<br><br>
		<label style="position: absolute;left:27%;font-size:
50px;">ADMIN</label>
	</div>
	<div style="position: absolute;top: 18%;left: 60%" id="userdiv">
		<a href="userlogin.php"><imgsrc="images/user1.png"
width="64%"></a>
		<br>
		<label style="position: absolute;left: 20%;font-size:
50px;">USER</label>
	</div>
</body>
</html>
```

Open or create Indexed DB @ mb.js

```
function loadFromDB() {

console.log("loading db...");

constdbOpen = window.indexedDB.open("myDatabase", 1);

dbOpen.onerror = event => {
```

```
        alert(event.target.errorCode);};
}
```

Add object to Indexed DB @ mb.js

```
function calculateE1RM(reps, weight) {
if (reps < 0) throw new Error("invalid reps: %s", reps);
if (reps === 0) return 0;
   if (reps == 1) return weight;
   if (reps < 10) return Math.round(weight / (1.0278 - 0.0278 * reps));
   else return Math.round(weight / 0.75);
}
function calculateEntry(entry) {
entry.xcalc.netweight = calculateNetWeight(entry);
entry.xcalc.e1rm = calculateE1RM(entry.reps, entry.weight);
entry.xcalc.volume = calculateVolume(entry);
}
```

Delete all entries from the database @ mb.js

```
function deleteAllEntries() {
db.close();
   let req = indexedDB.deleteDatabase("myDatabase");
req.onsuccess = function () {
alert("Deleted database successfully");
  };
}
```

Database connection @database.inc.php

```
<?php
$con=mysqli_connect('localhost','root','','admindb');
```

```
?>
```

User Registration @ insertuser.php

```
if (isset($_POST['signup'])) {

$email = $_POST['email'];
$password = $_POST['password'];
}

$query="SELECT * FROM USERSRC WHERE EMAIL='$email'";
$data=mysqli_query($conn,$query);
$num=mysqli_num_rows($data);

$uppercase = preg_match('@[A-Z]@', $password);
$lowercase = preg_match('@[a-z]@', $password);
$number    = preg_match('@[0-9]@', $password);

$hashpassword = md5($password);

                        $query1="insert into usersrc(email,password)
values('$email','$hashpassword')";

                        $data1=mysqli_query($conn,$query1);
```

Delete using ID @ deleteuser,php

```
<?php
$id = $_POST['id2'];
echo $id;
$query  = "DELETE FROM USERSRC WHERE ID='$id'";
mysqli_query($con,$query);

?>
<META  HTTP-EQUIV="Refresh" CONTENT="0; URL=includeusers.php">
```

Store weight in cookies @StoreWeightInCookie.php

```
<?php

session_start();

$cookie_name = "weight";
$cookie_value = $_POST['Weight'];
setcookie($cookie_name, $cookie_value, time() + (86400 * 30), "/");
```

```
echo "<script>location.href='workoutrecom_stand.php'</script>";
?>
```

Validation @uservalidate.php

```
if (isset($_POST['login'])) {

$email = $_POST['email'];
$password = $_POST['password'];
}
$query="SELECT * FROM USERSRC WHERE EMAIL='$email'";
$data=mysqli_query($conn,$query);
$result=mysqli_fetch_assoc($data);
if(md5($password))!== $result['password']){
echo “success”; }
```

Update food details @editfood.php

```
if (isset($_POST['id'])) {

$id=$_POST['id1'];
$name=$_POST['name'];
$description=$_POST['description'];
$weight=$_POST['weight'];
$category=$_POST['category'];
$smallfile = addslashes(file_get_contents($_FILES['file1']['tmp_name']));
$largefile = addslashes(file_get_contents($_FILES['file2']['tmp_name']));
$smalltype=$_FILES['file1']['type'];
$largetype=$_FILES['file2']['type'];
}
$query = "UPDATE FOODTABLE SET NAME='$name'
,DESCRIPTION='$description',WEIGHT='$weight',CATEGORY='$category'
WHERE ID='$id'";
$set=mysqli_query($con,$query);

if(!empty($smallfile)){
$query1 = "UPDATE FOODTABLE SET
SMALL_PIC='$smallfile',SMALL_TYPE='$smalltype' WHERE ID='$id'";
mysqli_query($con,$query1);
}

if(!empty($largefile)){
$query2 = "UPDATE FOODTABLE SET LARGE_PIC='$largefile',
LARGE_TYPE='$largetype' WHERE ID='$id'";
mysqli_query($con,$query2); }
```

7. RESULTS/SCREENSHOTS

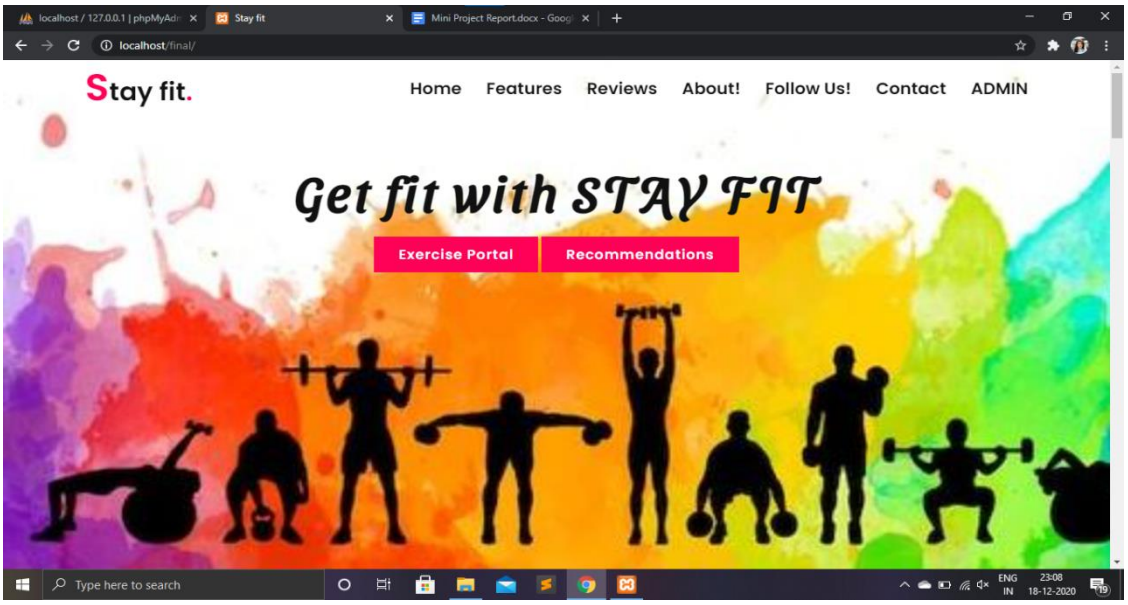

Fig 7: Home page

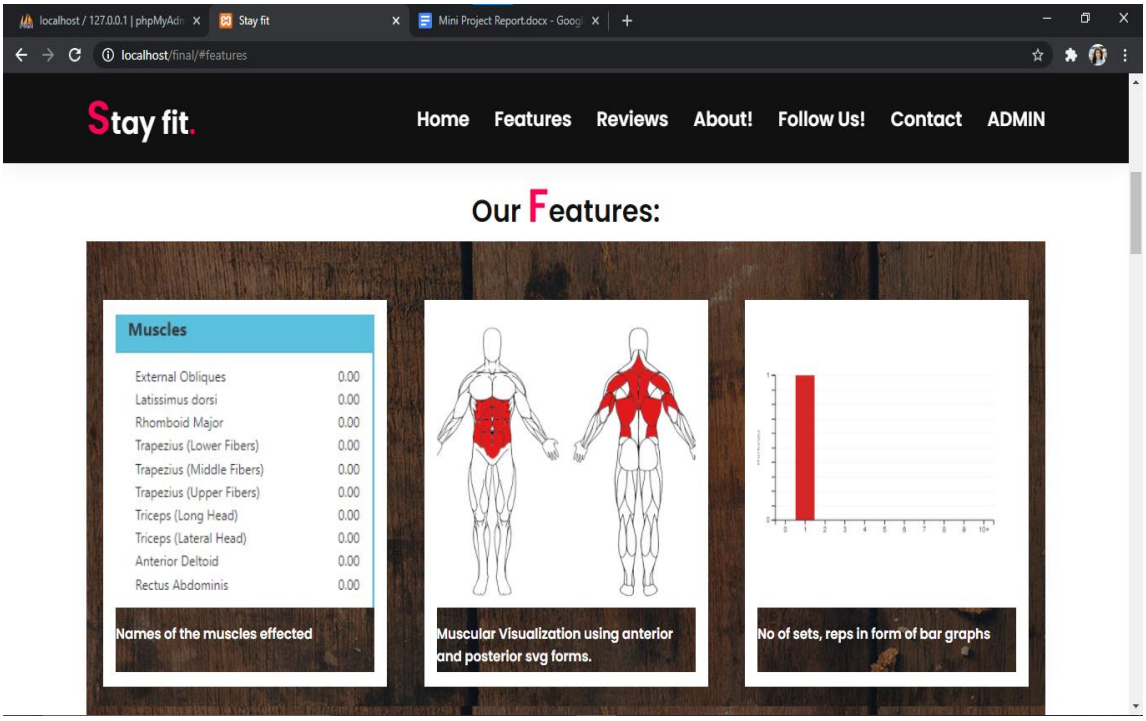

Fig 8: Features

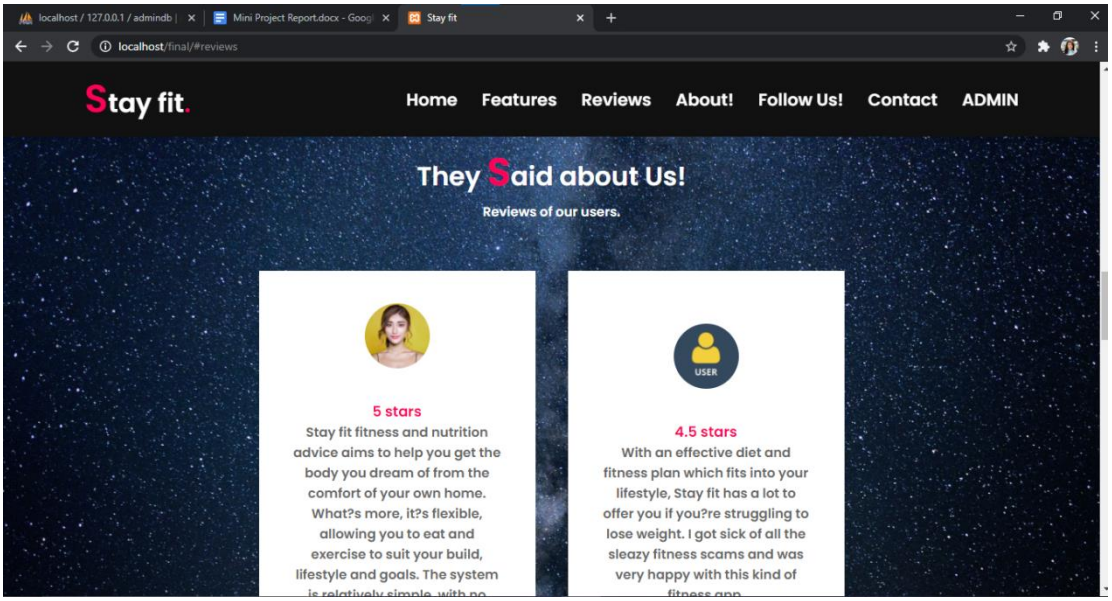

Fig 9: Reviews

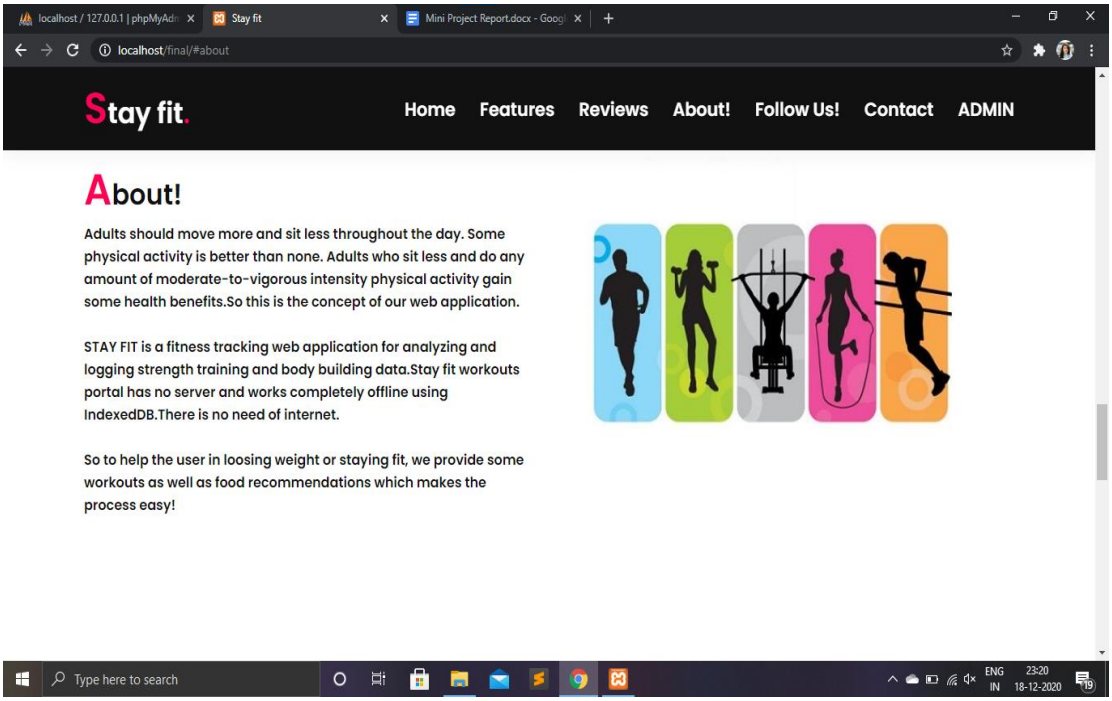

Fig 10: About

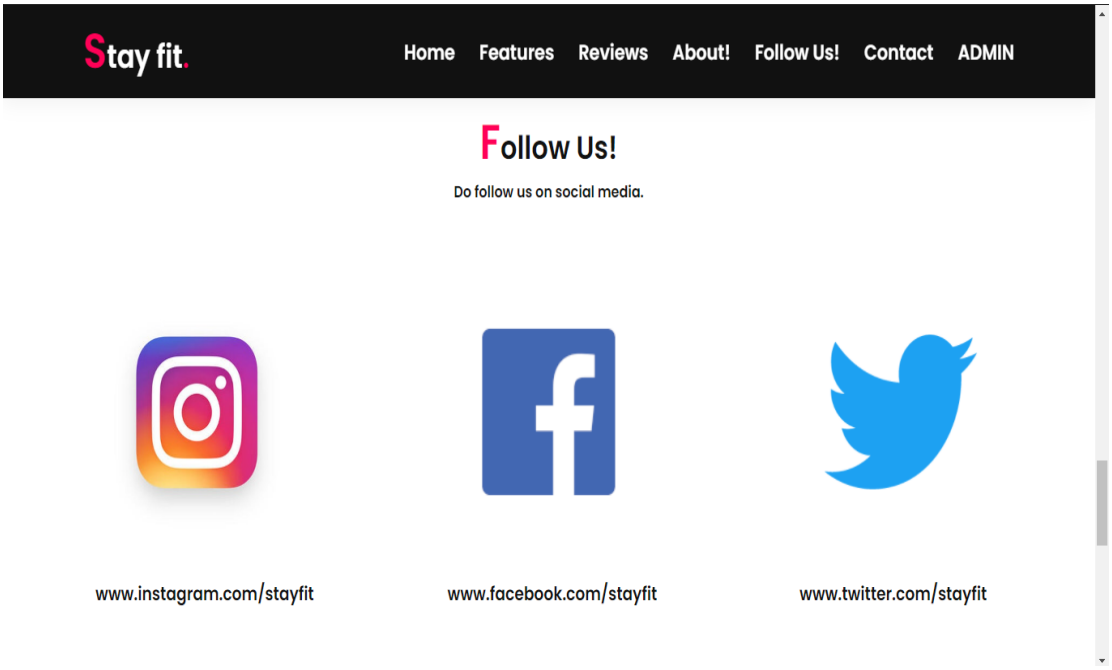

Fig 11: Follow on social media

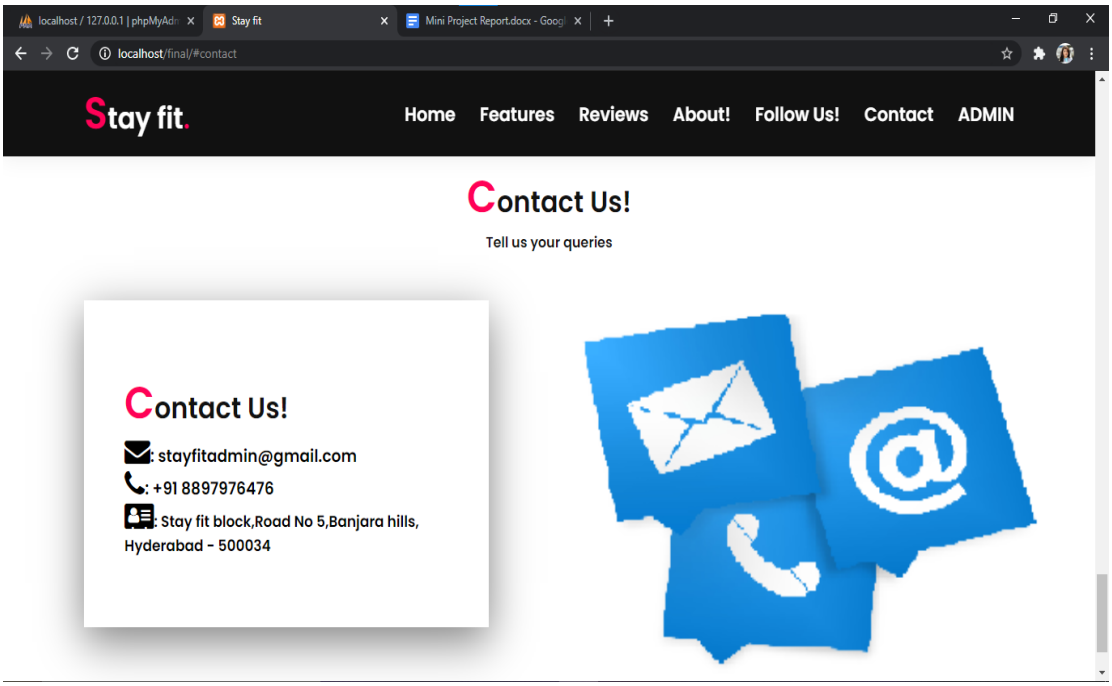

Fig 12: Contact Us page

7.1 Exercise portal:

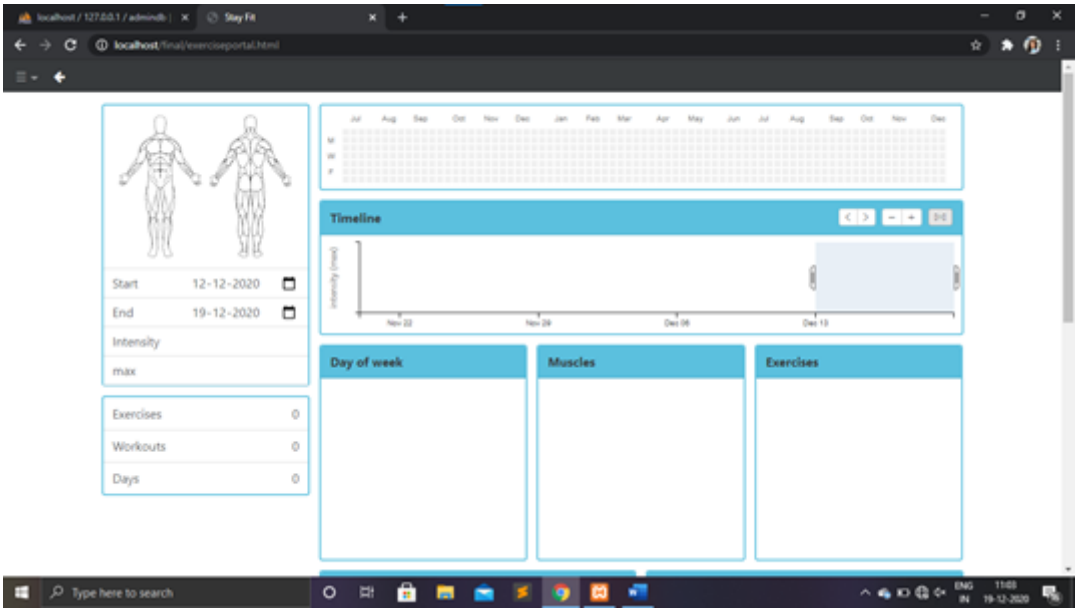

Fig 13: Exercise portal home page

Fig 14: Bar graphs columns of sets, reps and weight

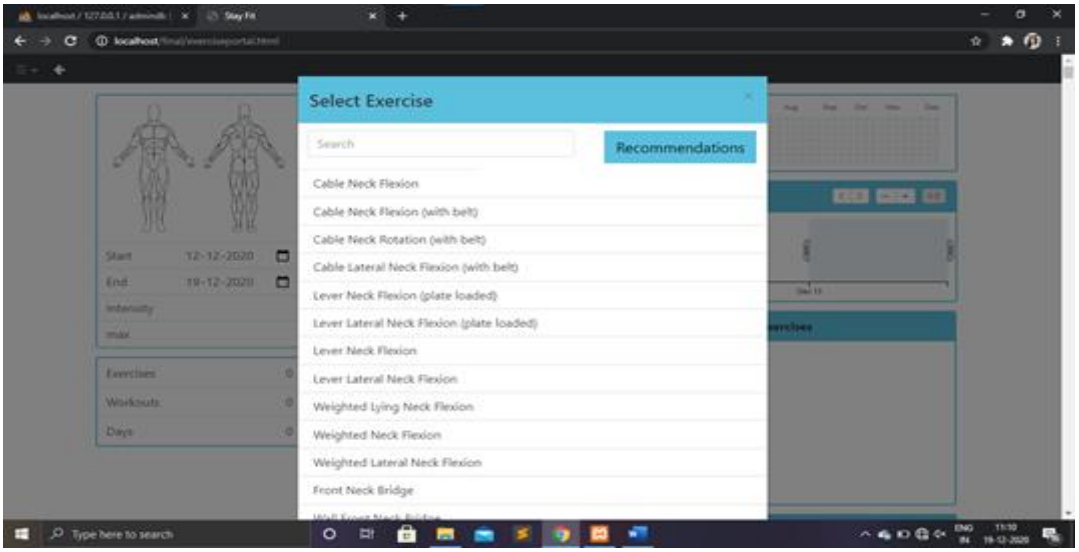

Fig 15: Select exercise modal

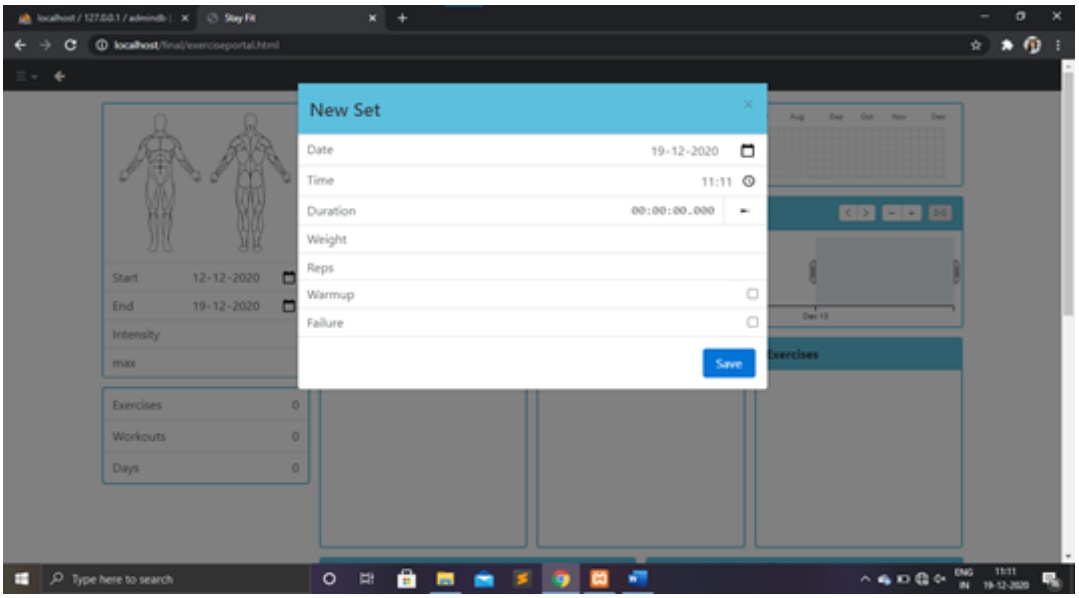

Fig 16: Add new set of exercise

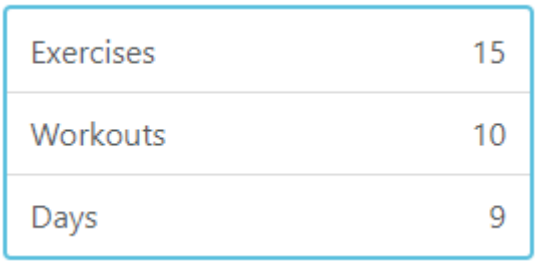

Fig 17: Count of Exercises, Workouts and Days

Fig 18: After performing few sets of exercises (Timeline, Days of week, Muscles and Exercises performed)

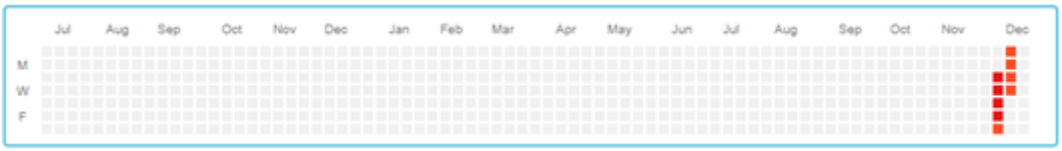

Fig 19: Shows the days of workout in the calendar

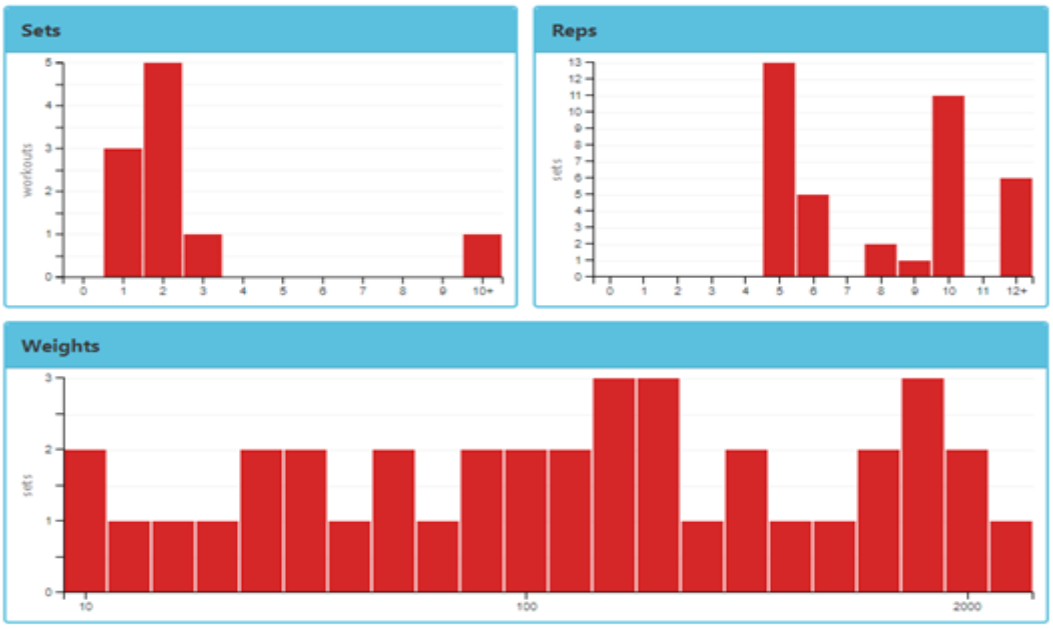

Fig 20: Bar graphs of sets and reps using dc.js predefined library

Exercise	Reps	Weight	Duration
Workout 9: Wed, Dec 9, 2020			
Weighted Neck Harness Extension	20	90	11.3s
Suspended Y Lateral Raise	5	40	6.1s
Lever Shoulder Press (parallel grip)	5	10	0.0s
Workout 8: Tue, Dec 8, 2020			
Suspended Front Raise	5	100	0.0s
Workout 7: Mon, Dec 7, 2020			
Cable Standing Shoulder Press	5	500	0.0s
Workout 6: Sun, Dec 6, 2020			
Smith Behind Neck Press	5	200	0.0s
Dumbbell One Arm Lateral Raise	5	200	0.0s
Workout 5: Sat, Dec 5, 2020			
Weighted Neck Flexion	5	2,000	0.0s
Weighted Lying Neck Flexion	20	500	0.0s
Workout 4: Fri, Dec 4, 2020			
Dumbbell Lying Rear Delt Row	5	20,000	0.0s
Dumbbell Lying Rear Delt Row	5	2,000	0.0s
Workout 3: Thu, Dec 3, 2020			
Dumbbell Lying Rear Delt Row	5	500	0.0s
Workout 2: Thu, Dec 3, 2020			
Dumbbell Lying Rear Delt Row	5	100	0.0s
Dumbbell Lying Rear Delt Row	5	80	0.0s
Workout 1: Wed, Dec 2, 2020			
Lever Lateral Neck Flexion (plate loaded)	20	198	0.0s
Lever Lateral Neck Flexion (plate loaded)	20	155	0.0s
Workout 0: Tue, Dec 1, 2020			
Barbell Bench Press	5	175	0.0s
Barbell Bench Press	12	45	0.0s
Cable Pulldown	10	35	0.0s
Cable Pulldown	10	25	0.0s
Bench Dip	9	169	0.0s
Bench Dip	10	169	0.0s
Barbell Bench Press	6	155	0.0s
Barbell Bench Press	6	155	0.0s
Barbell Bench Press	6	135	0.0s
Barbell Bench Press	6	135	0.0s
Barbell Bench Press	6	135	0.0s
Barbell Bench Press	8	115	0.0s

Fig 21: List of all the workouts in a specific week

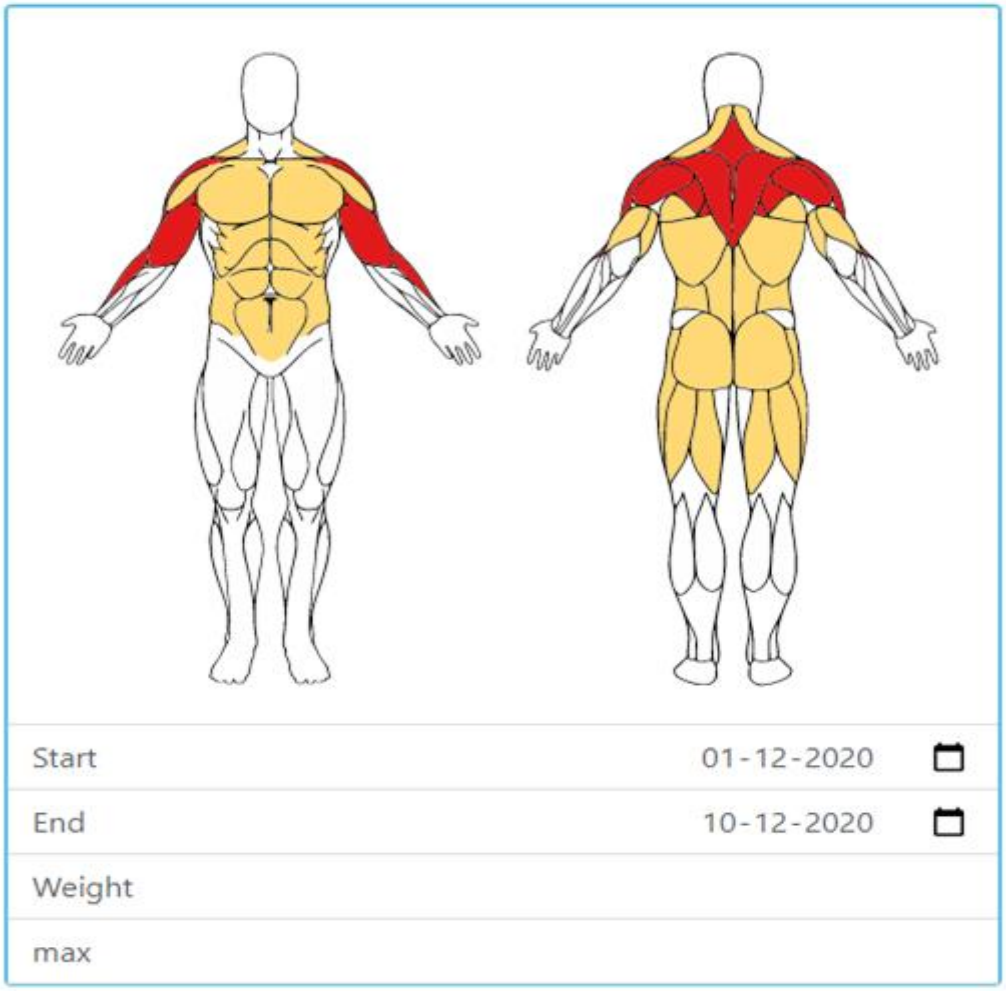

Fig 22: Representation of muscles affected

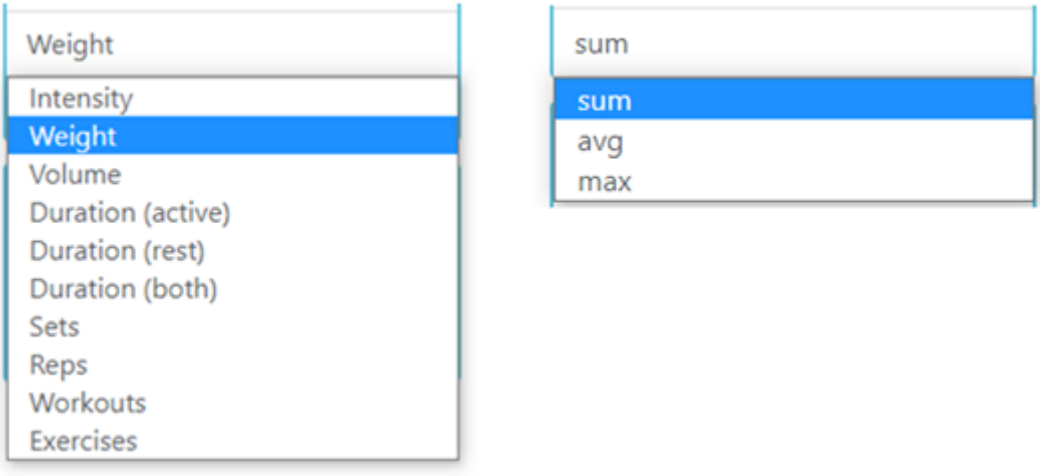

Fig 23: Dropdown for selection of metrics

7.2 Recommendations portal:

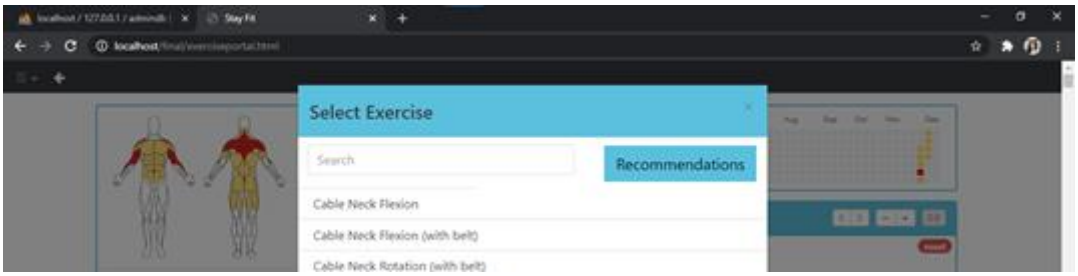

Fig 24: Navigate to recommendations portal

Fig 25: User Registration and Login page

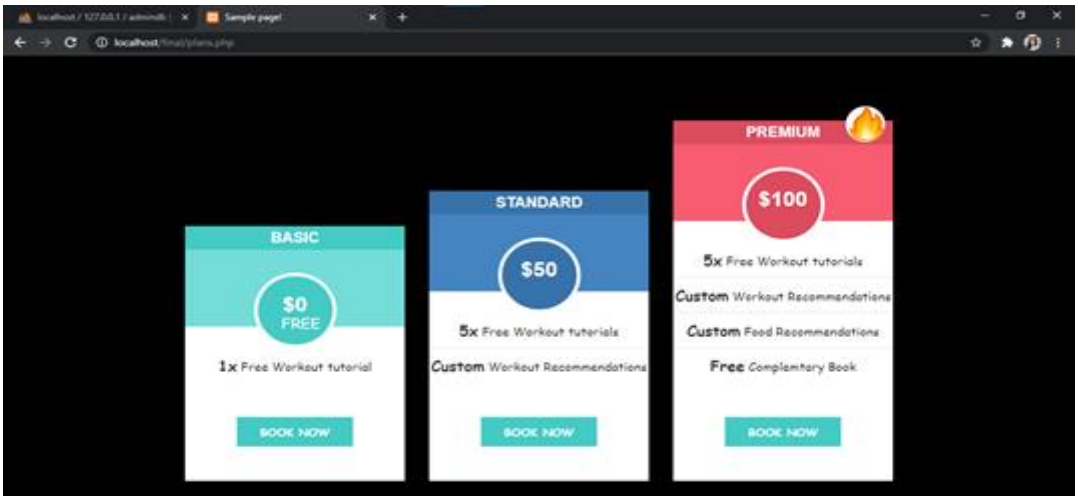

Fig 26: Selection of subscription plans (After registration)

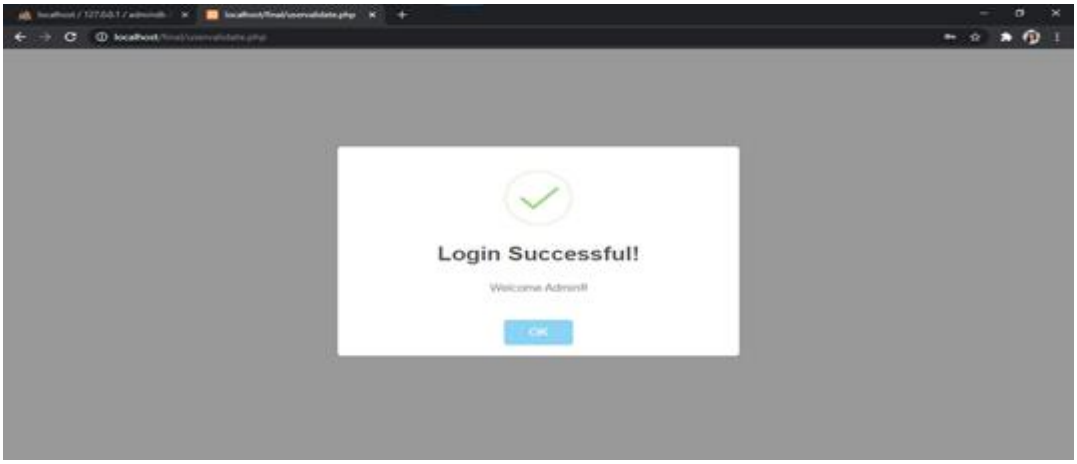

Fig 27: After successful login

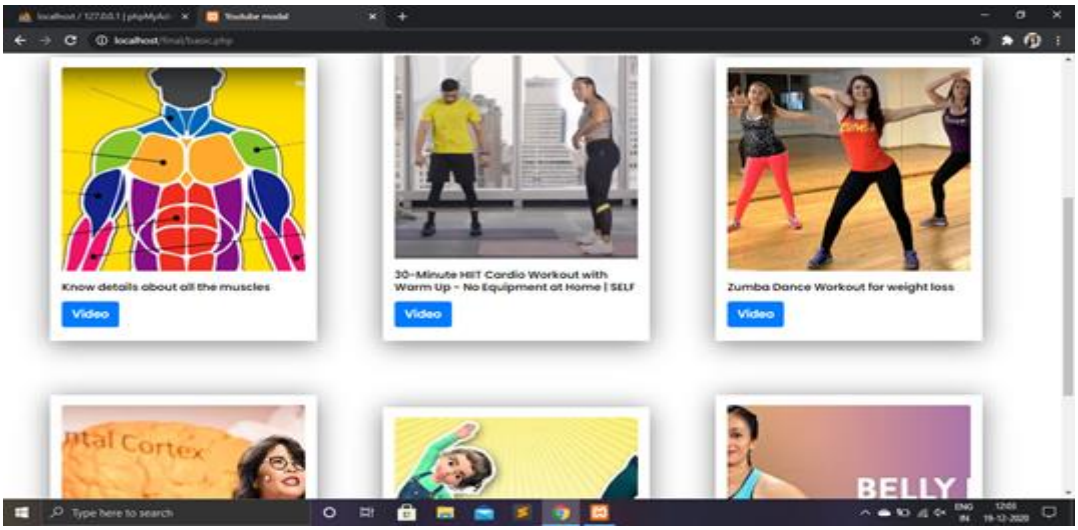

Fig 28: Basic plan tutorials

Fig 29: Youtube modal for workout tutorials

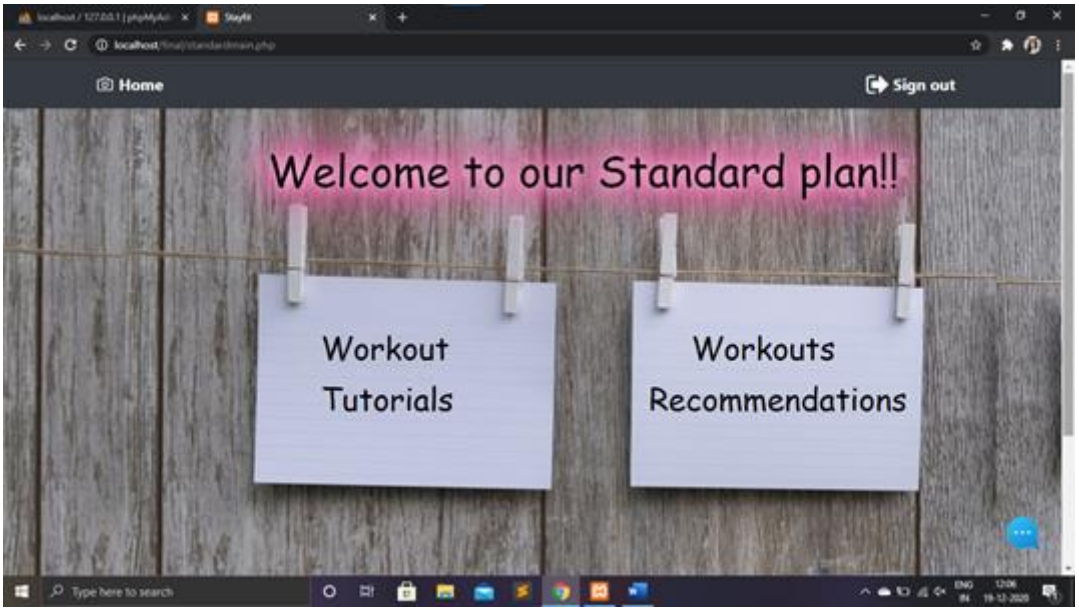

Fig 30: Standard plan home page

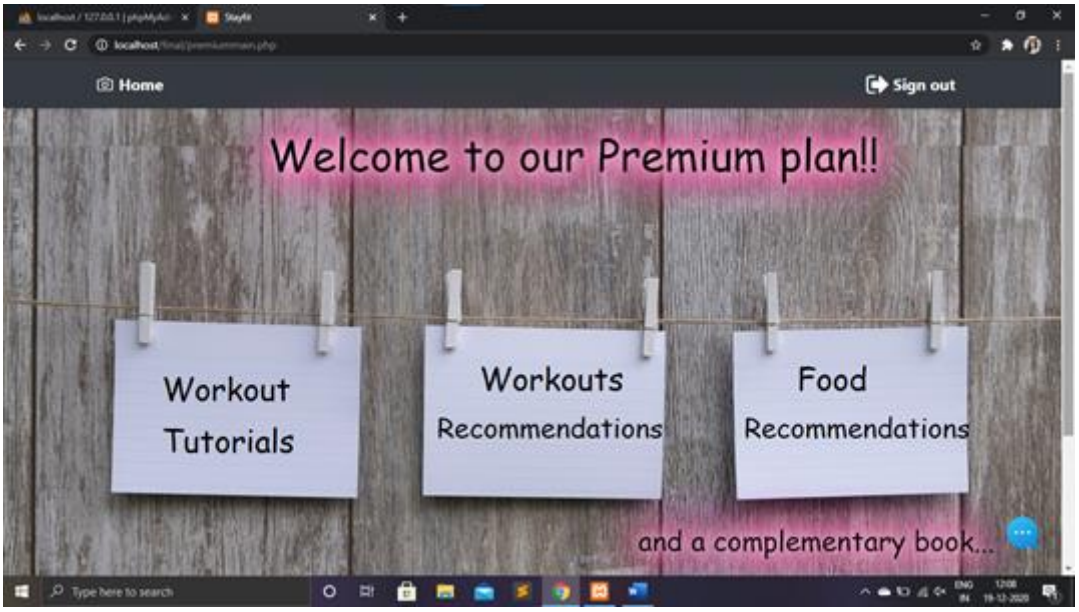

Fig 31: Premium plan home page

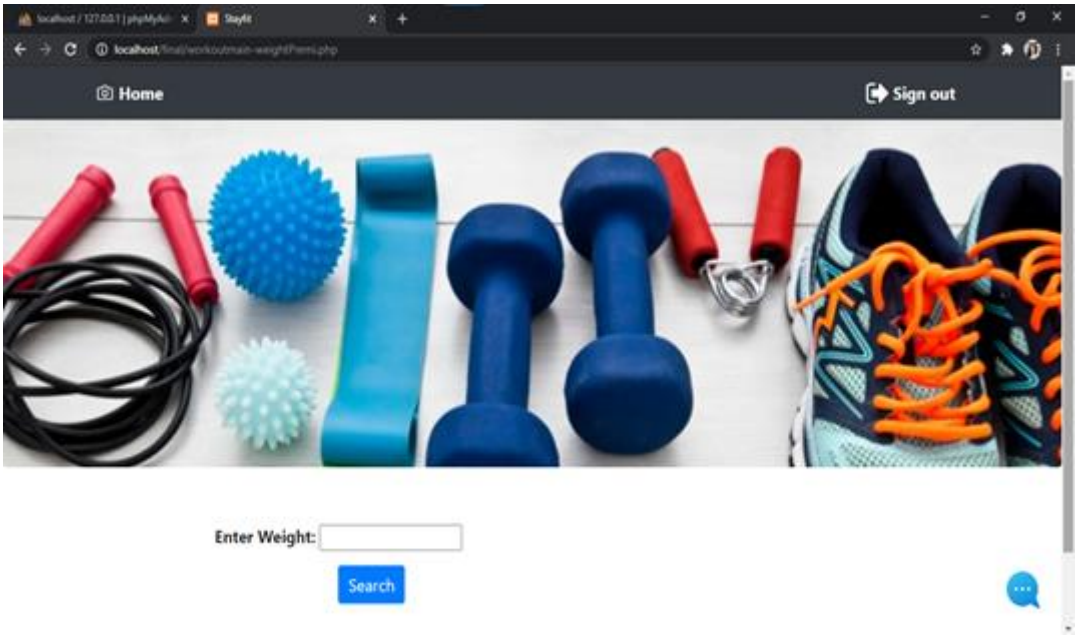

Fig 32: Workout recommendations home page

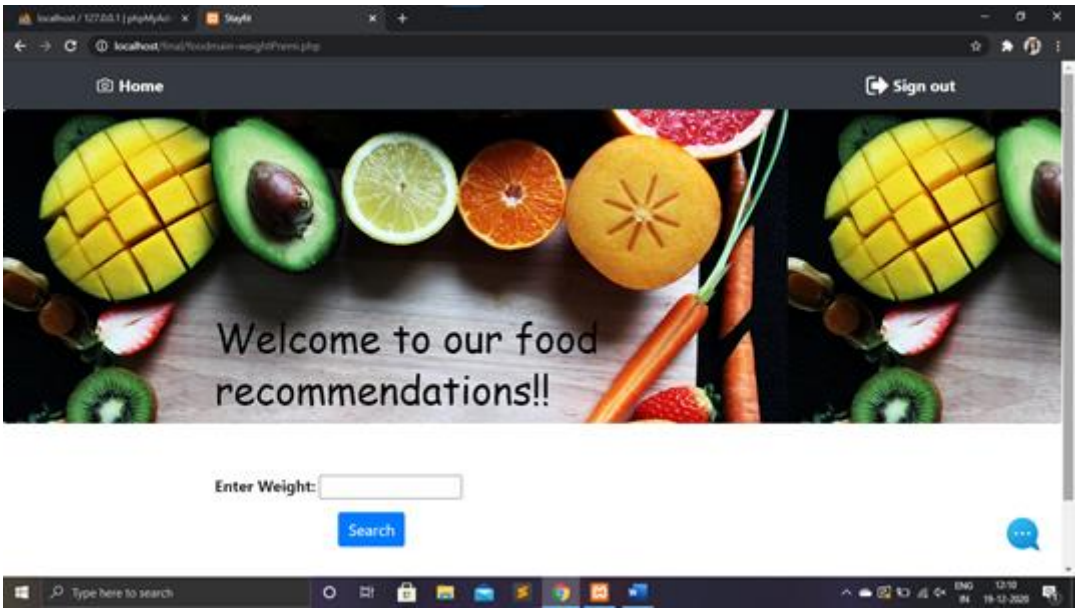

Fig 33: Food recommendations home page

Fig 34: Selection of workout recommendations

Fig 35: Selection through category

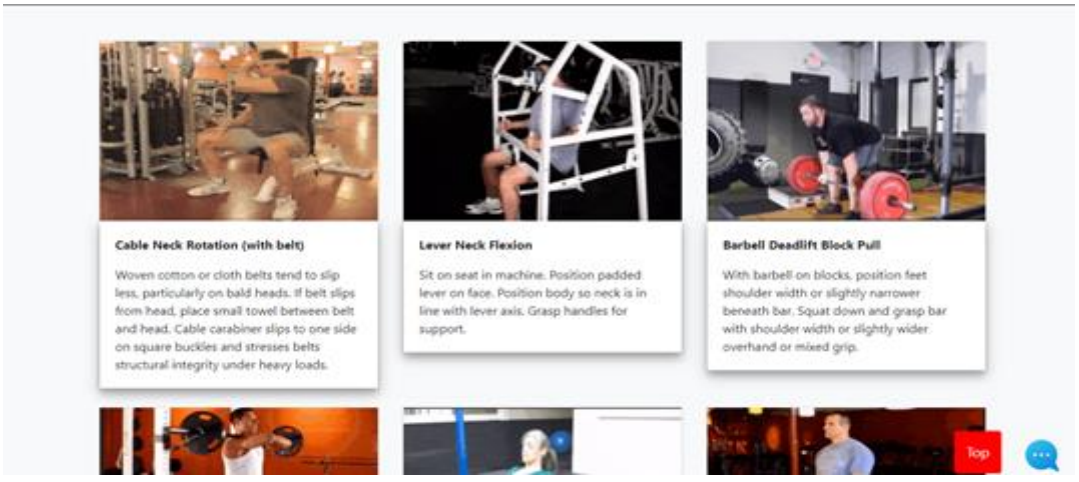

Fig 36: Display all workout recommendations

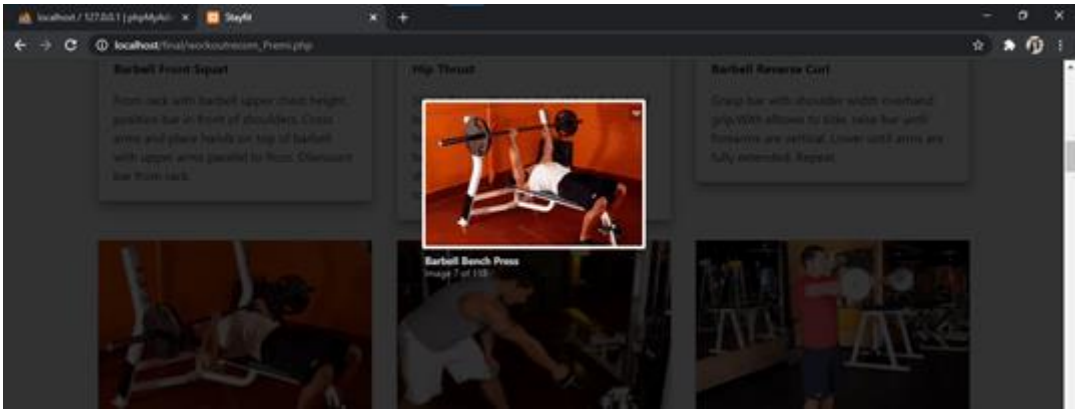

Fig 37: How to do a specific workout? (GIF File)

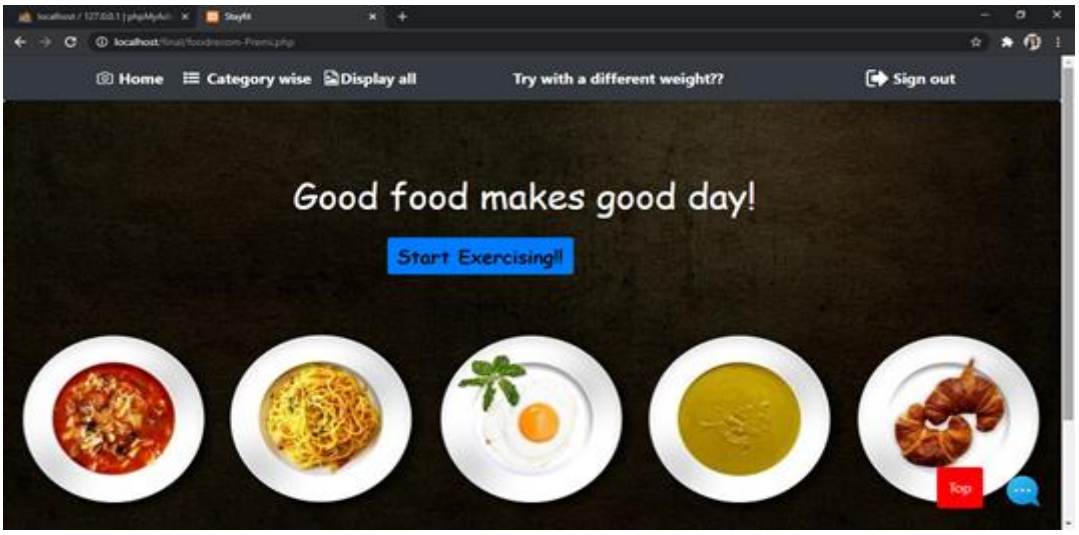

Fig 38: Selection of food recommendations

CATEGORIES:

- Breakfast
- Lunch
- Dinner

Fig 39: Selection through category

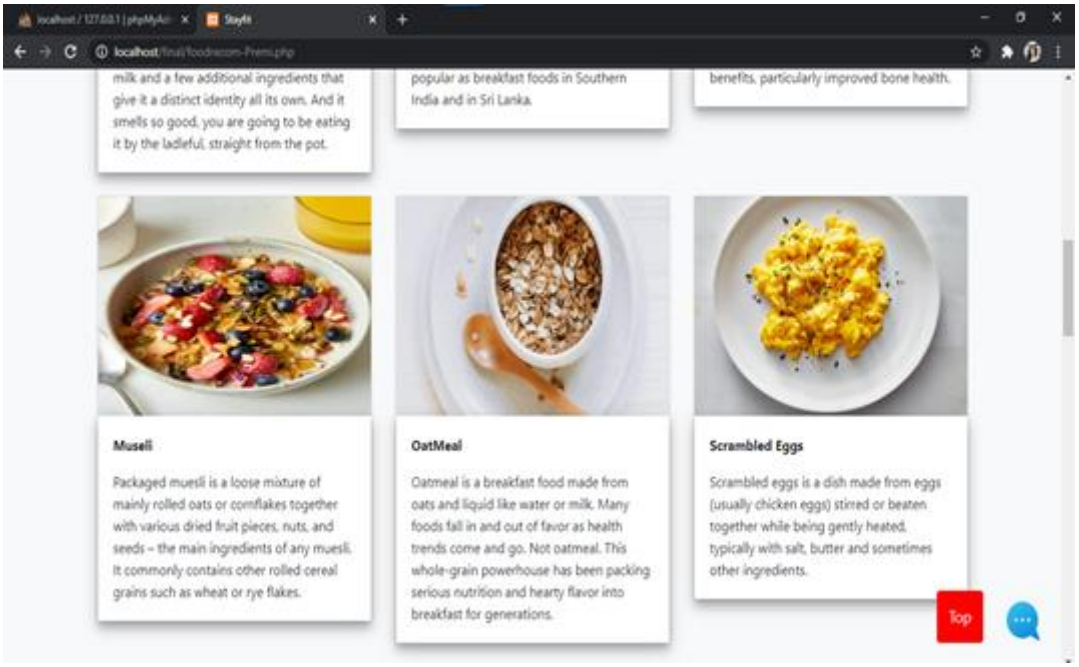

Fig 40: Display all food recommendations

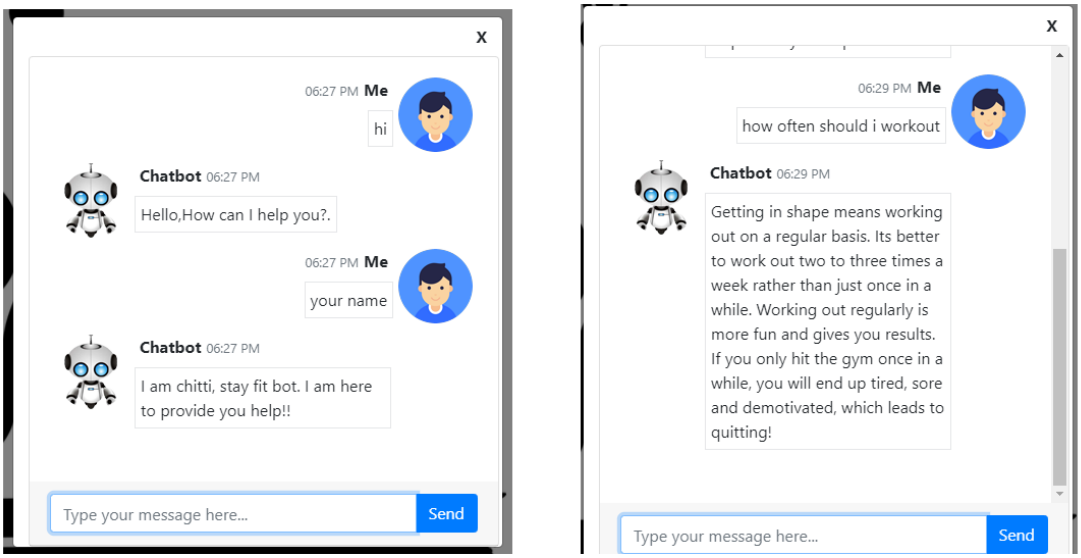

Fig 41: Chatbot

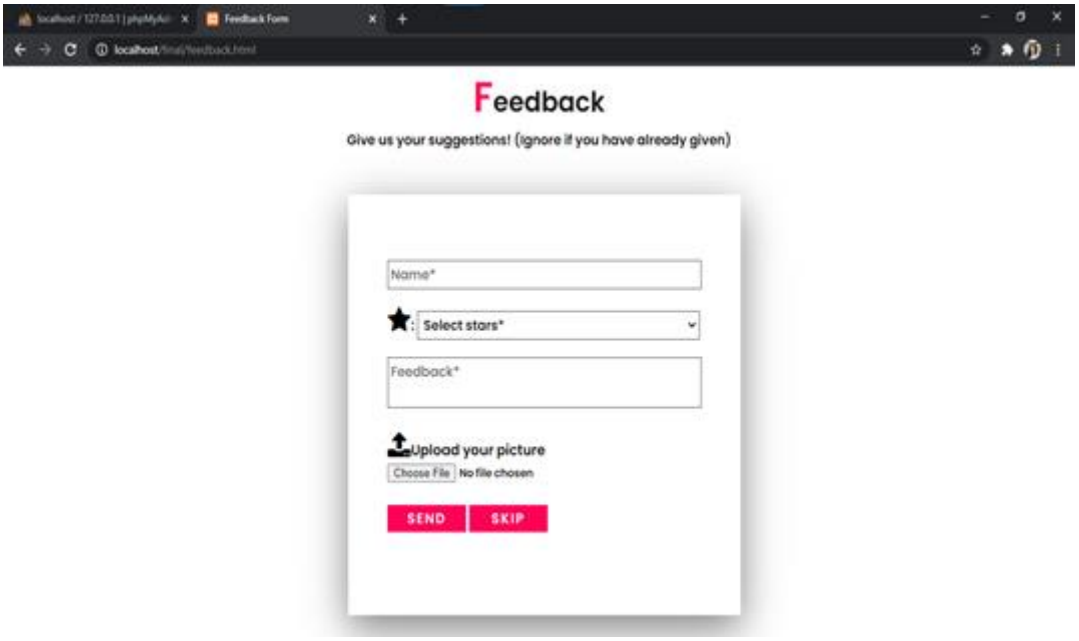

Fig 42: Feedback form

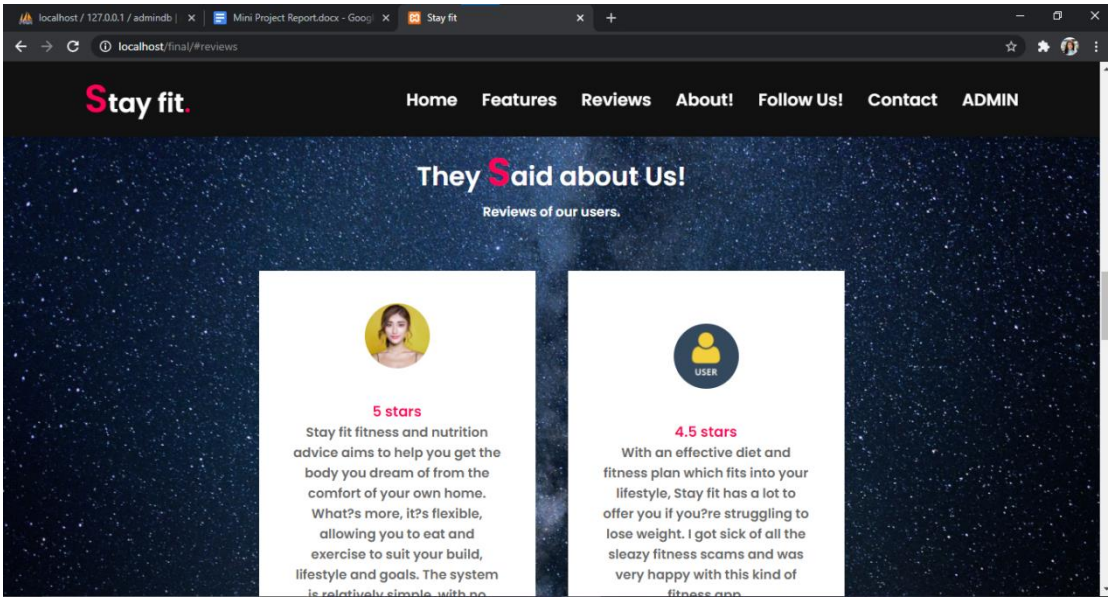

Fig 43: Review of some of our users

7.3 Admin portal:

Fig 44: Navigation to admin portal

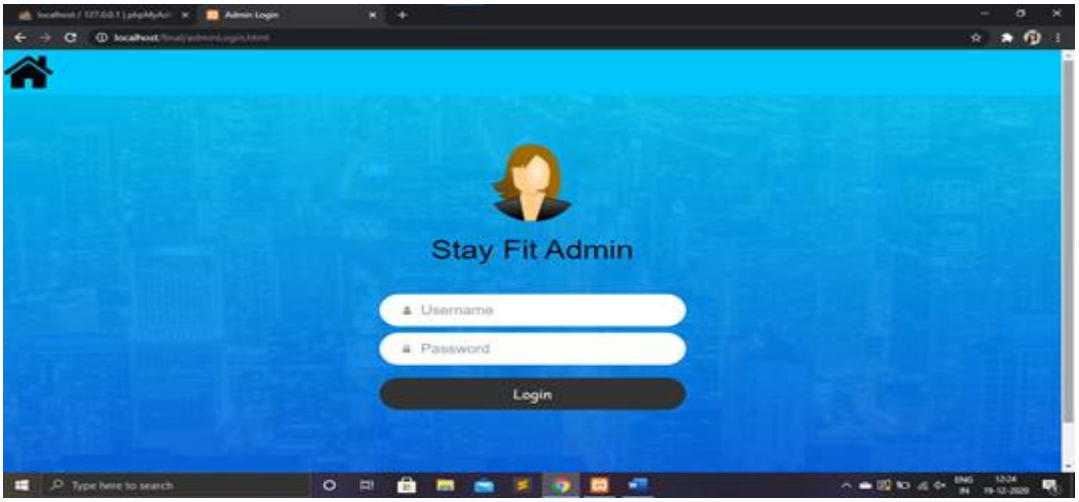

Fig 45: Admin login page

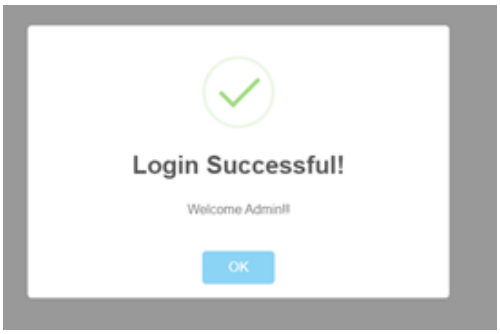

Fig 46: Login successful dialog box

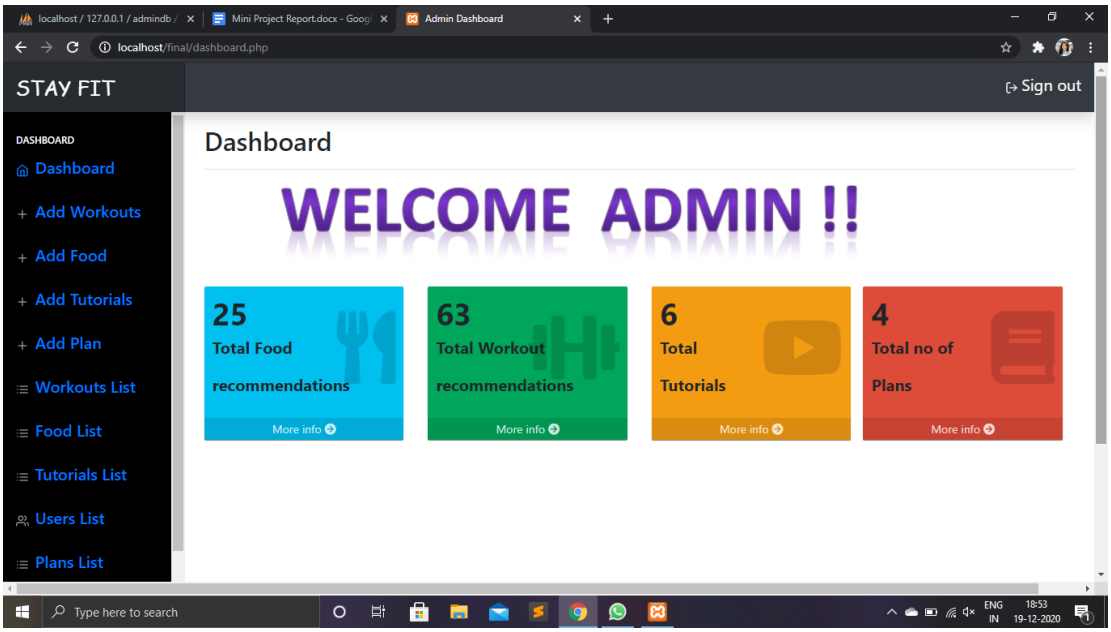

Fig 47: Admin Dashboard

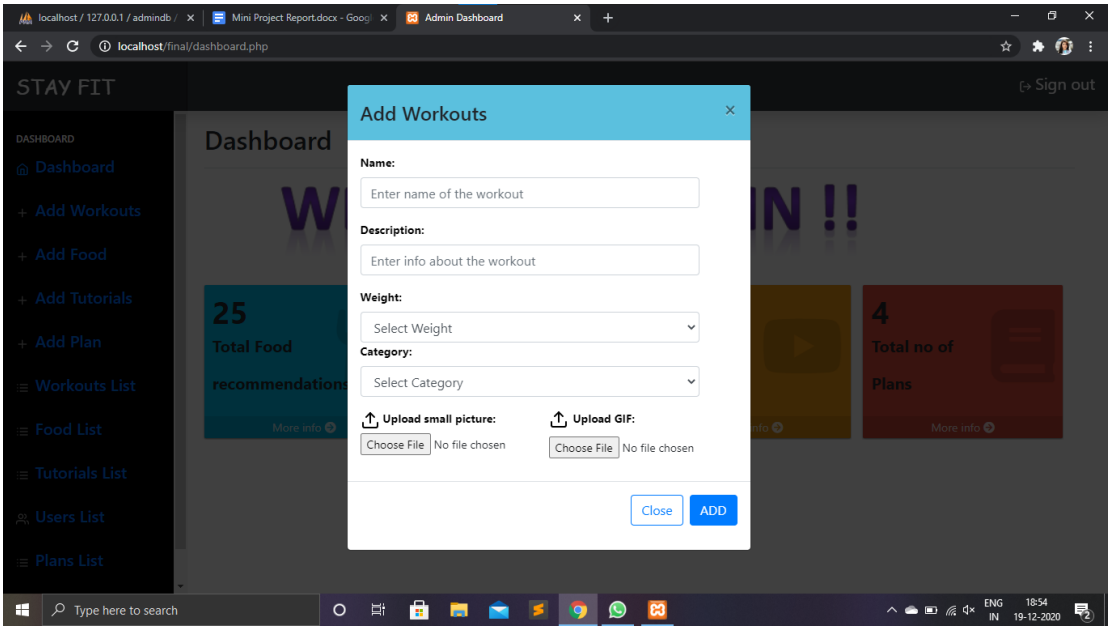

Fig 48: Add Workouts dialog box

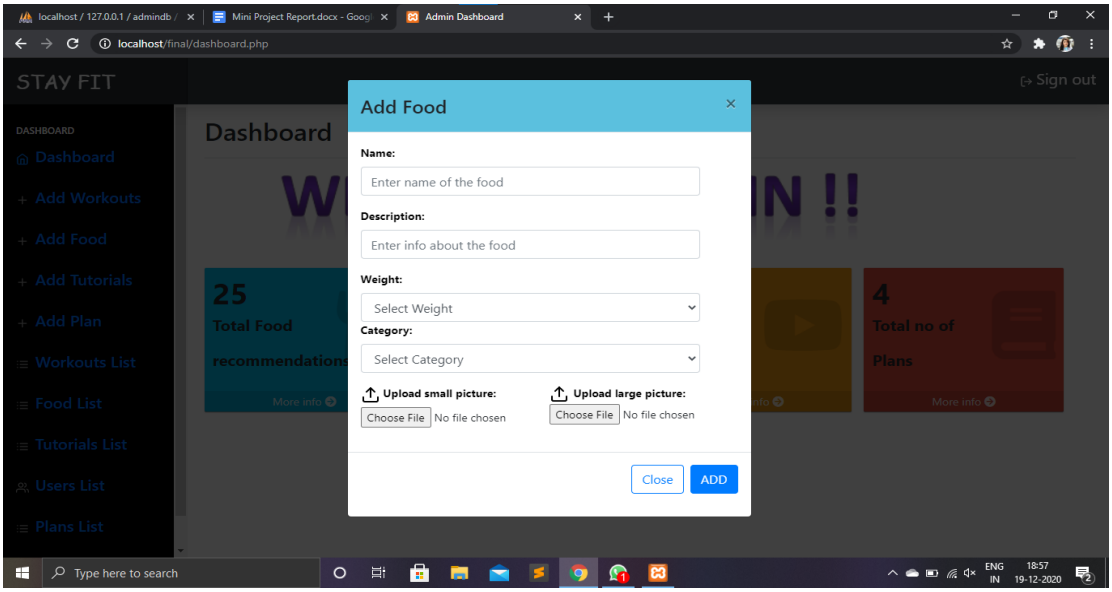

Fig 49: Add food dialog box

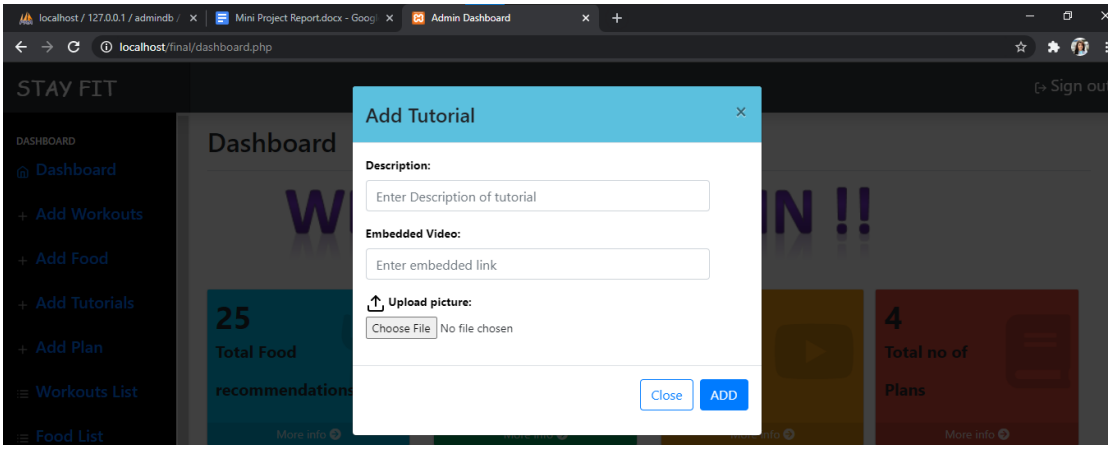

Fig 50: Add tutorials

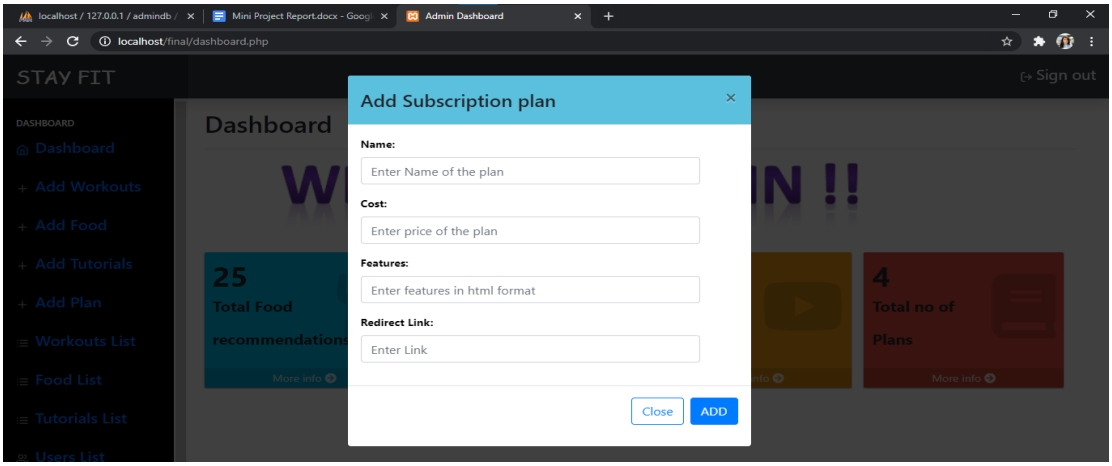

Fig 51: Add a new subscription plan

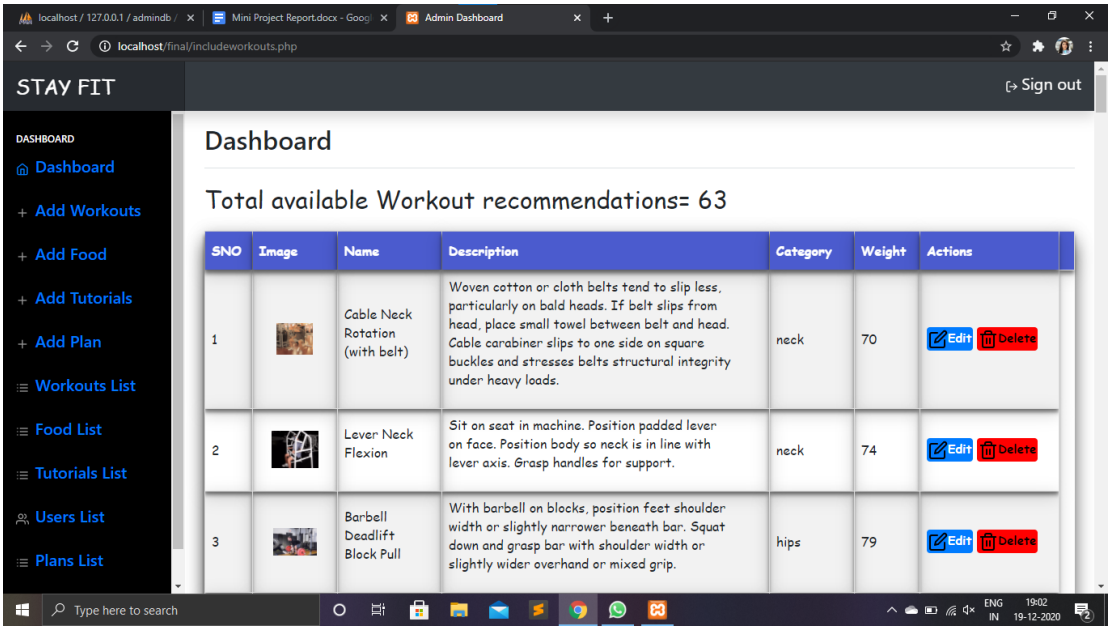

Fig 52: List of all the workouts

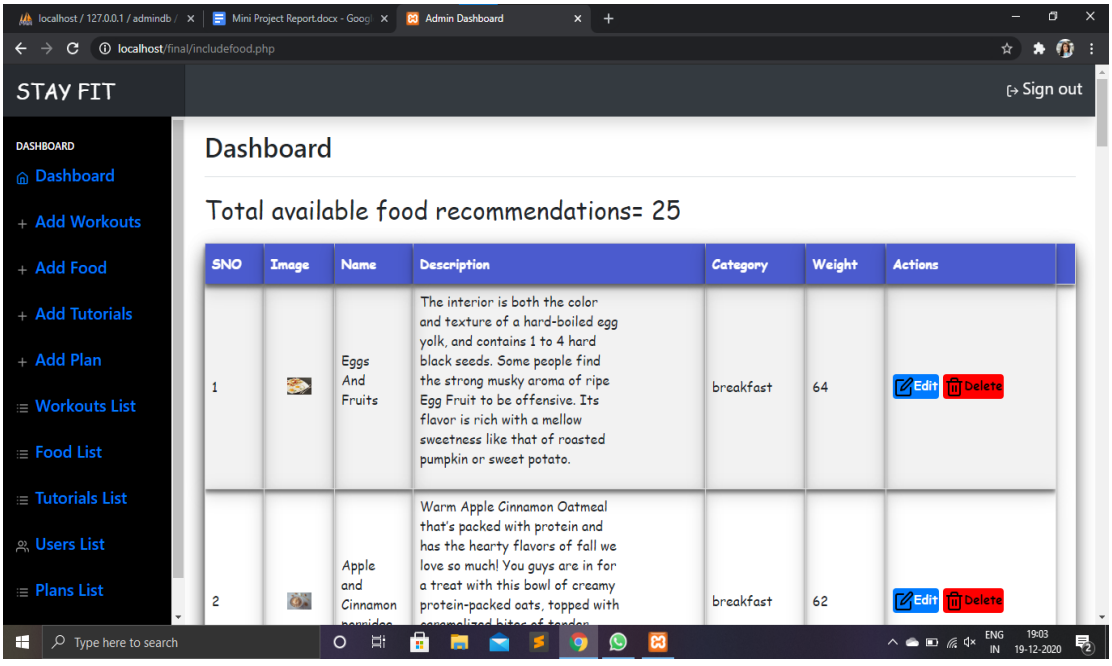

Fig 53: List of all food items available in the database

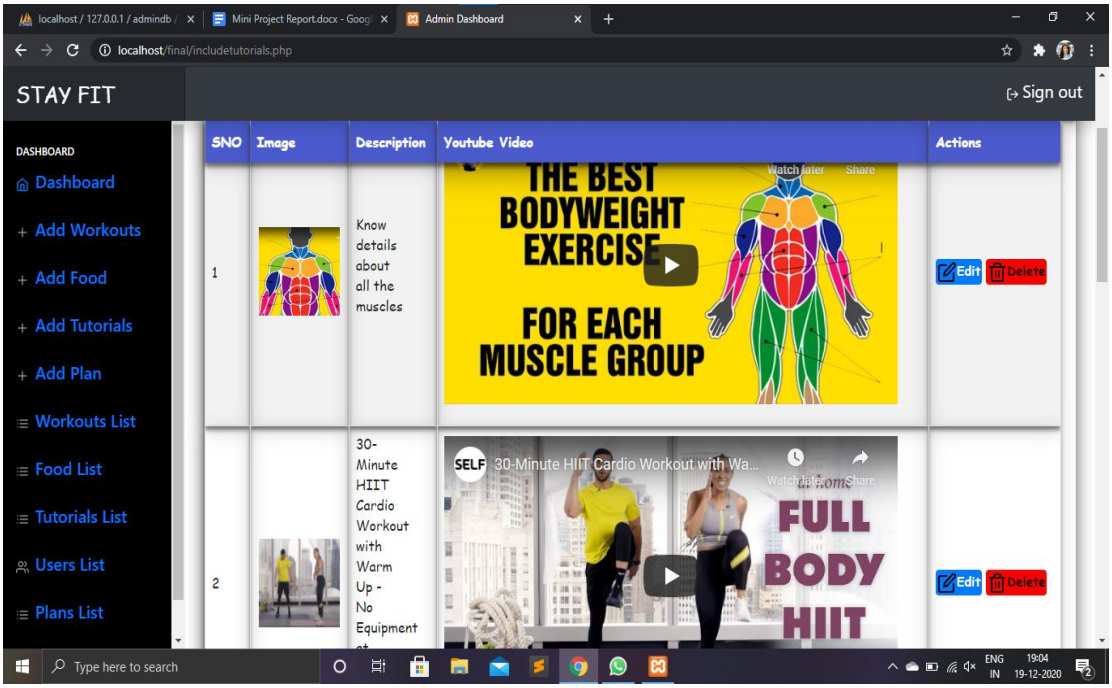

Fig 54: List of tutorials

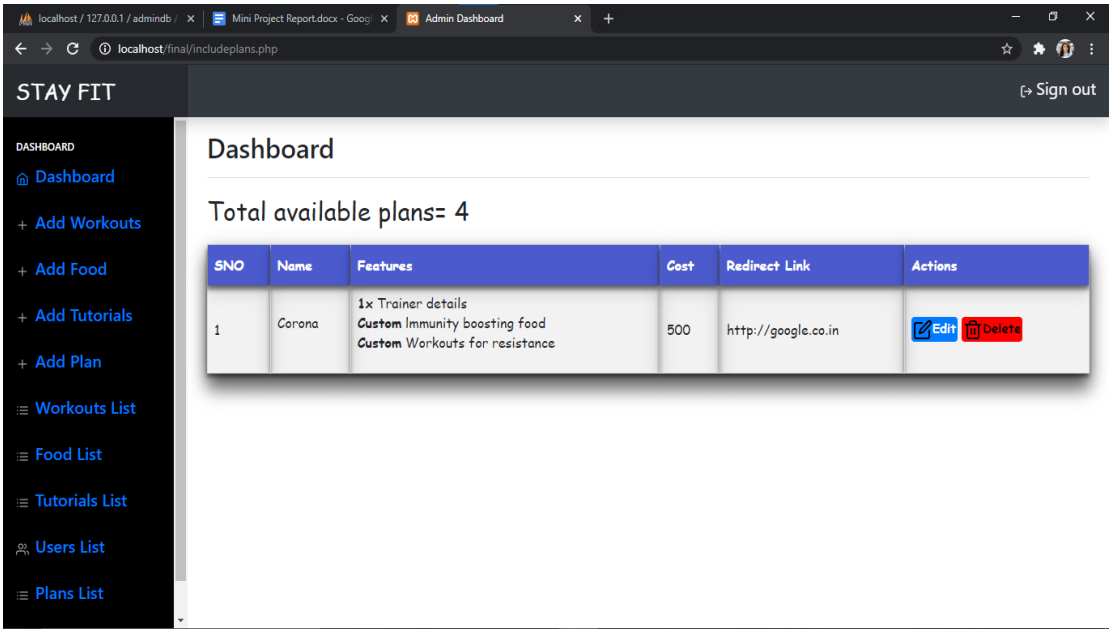

Fig 55: Available subscription plans other than basic ones

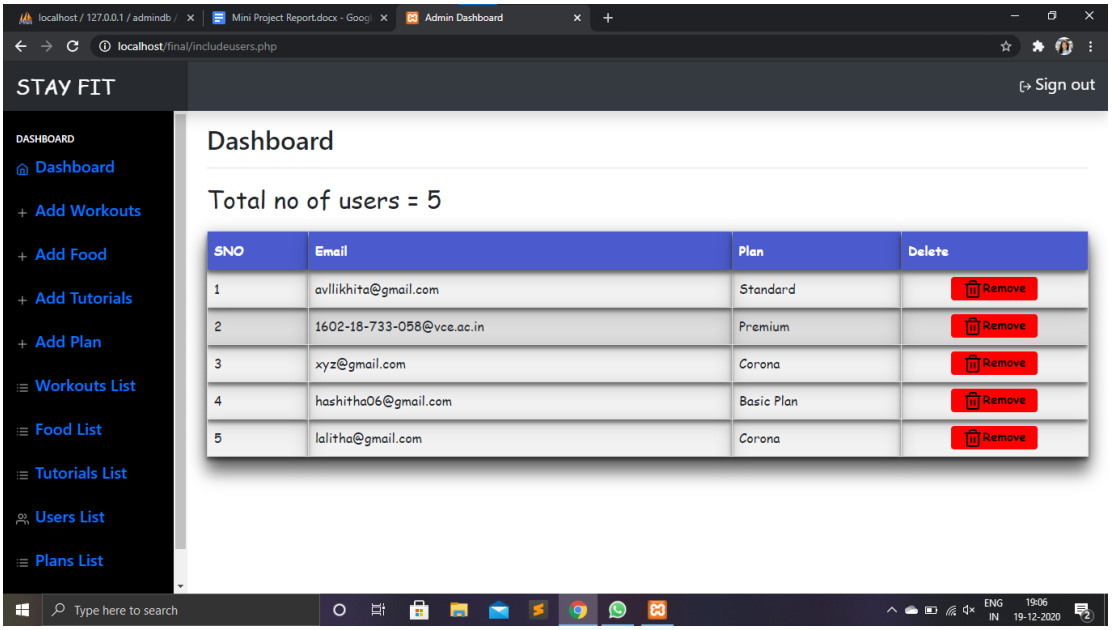

Fig 56: User's details

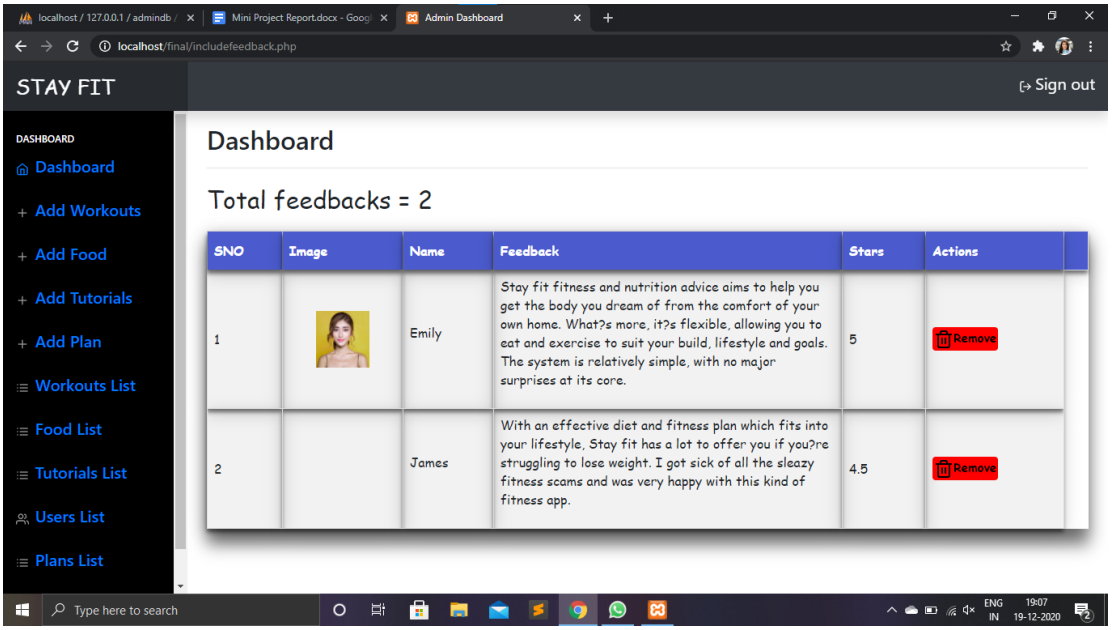

Fig 57: Users feedback

7.4 Payment Gateway

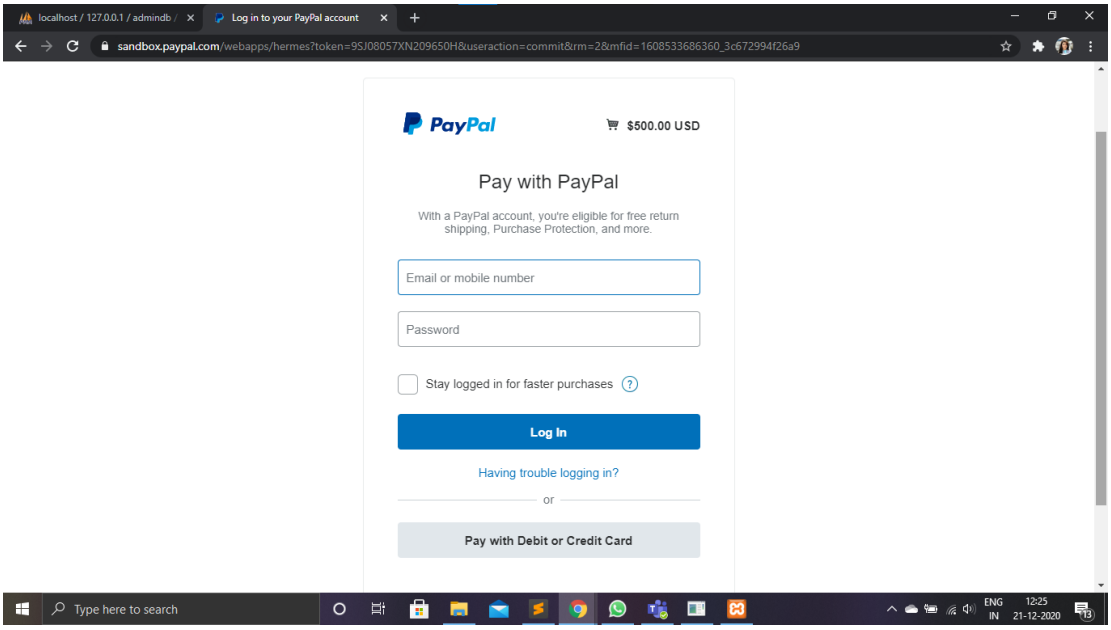

Fig 58: Paypal payment gateway

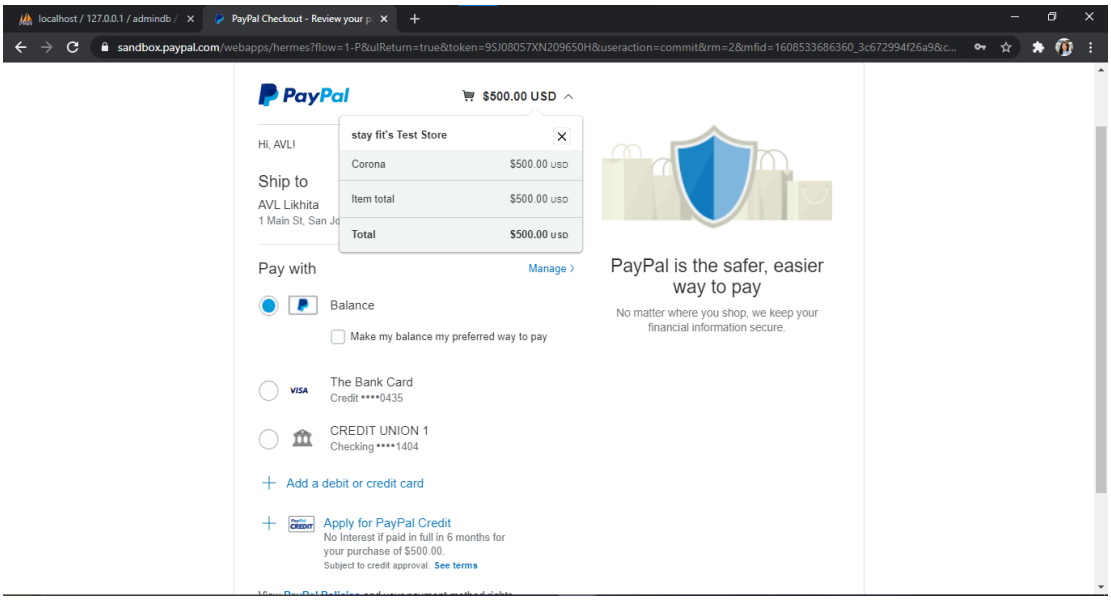

Fig 59: Cart (plans)

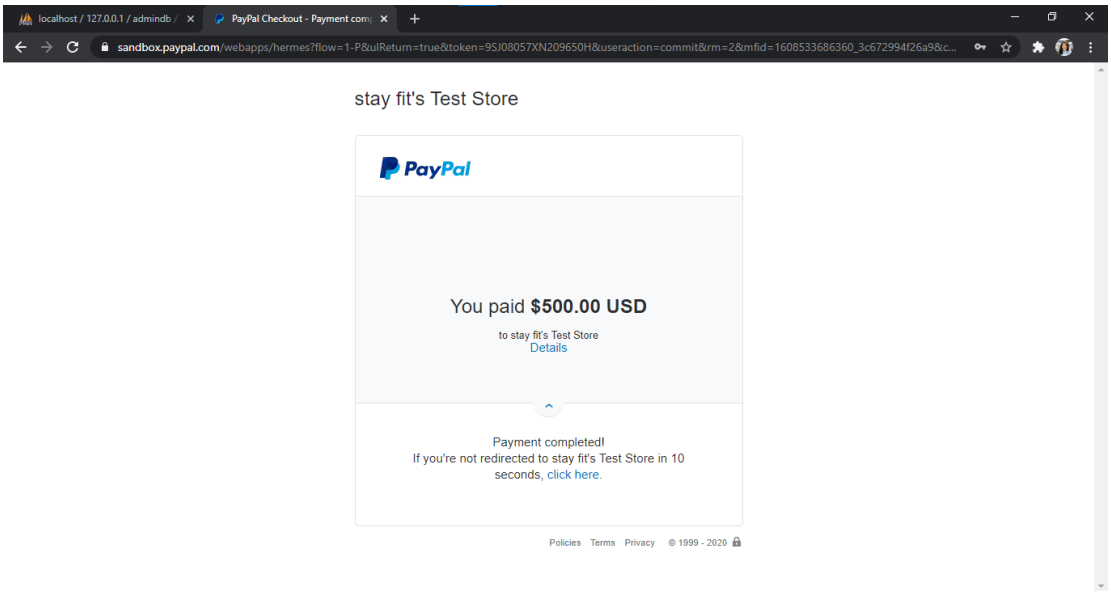

Fig 60: Payment successful

8. SYSTEM TESTING

System testing is a level of testing that validates the complete and fully integrated software product. The purpose of a system test is to evaluate the end-to-end system specifications.

Our project has undergone two levels of testing.

8.1 Unit Testing

Unit testing is undertaken when a module has been created and successfully reviewed.

Unit testing was done on every module present in our application.

1. Test for Admin Module: Admin is able toadd, delete and modify all workout and food recommendations, subscription plans and tutorials without any error and delay. Necessary validations were implemented to prevent SQL injection.

2. Test for Exercise Module: User is able to add sets. Graphs and muscular visualisations are shown perfectly. Duration time, intensity and volume calculations are accurate.Necessary validations were implemented to preventSQL injection.

3. Test for Recommendation's Module: User is able toget recommendations (food and workouts) according to his/her weight without any errors.

4. Test for Chat Module: User is able to clarify the doubts with the bot. String matching is working well without any exceptions.

8.2 Integration Testing

In this type of testing, we test various integration of the project module by providing the input. The primary objective is to test the module interfaces to ensure that no errors are occurring when one module invokes the other module.

All the changes by admin module, is being reflected in user and chat module.

Therefore,

The system can integrate flawlessly and function well.

9. CONCLUSION & FUTURE WORK

STAY FIT web application hence acts as an ultimate fitness tracker for the users who wants to keep track of their daily or weekly progress of workouts like their intensity, number of sets, number of reps, time duration etc...For the beginners who are new to any form of workouts will have a separate portal where they can select their plans and workout accordingly. Staying fit does not only mean doing workouts, it must also include a proper diet. This web application also provides a diet (Breakfast,Lunch and Dinner)based on the user's weight.

This project helped us to gain insight into full stack web application development and learn a new framework that is JavaScript. It also helped us to improve our knowledge regarding JavaScript language and HTML scripting.

The website can be further extended to an app in the future as a part of development of our project. Moreover additional functionalities can be added like online training sessions and monthly/yearly food subscriptions of your diet delivered to the user.

10. REFERENCES

For exercise portal:

[1]BrewerColor Schemes available at http://www.graphviz.org/doc/info/colors.html

[2]How can I get query string values in JavaScript? http://stackoverflow.com/a/2880929/952123

[3]Filtering disables, https://github.com/dc-js/dc.js/issues/991

[4]Justify buttons with Bootstrap v4, http://stackoverflow.com/questions/34450586

[5]JavascriptDate.toJSON don't get the timezone offset, https://stackoverflow.com/questions/11382606/javascript-date-tojson-dont-get-the-timezone-offset

[6]GitHub Commit Calendar Clone using D3, https://bl.ocks.org/peterbsmith2/a37f2b733a75a6f348c2

[7]Add days to JavaScript Date, https://stackoverflow.com/questions/563406/add-days-to-javascript-date

[8]For all the sets of exercises, https://exrx.net/

For recommendations portal:

[9]Integratingpaypal (payment gateway), https://developer.paypal.com/docs/integration/web/

[10]https://www.youtube.com/watch?v=rFSVoGRXcCY

[11] Paypal home, https://www.paypal.com/us/home

[12]Signup for paypal, https://www.paypal.com/us/signin

[13] Temporary hosting of a page to redirect from paypal using ngrok, https://ngrok.com/docs

[14]Bootstrap alerts, https://demos.creative-tim.com/argon-dashboard/docs/components/alerts.html
https://mdbootstrap.com/docs/jquery/modals/additional/

Icons:

[15]Feather icons: https://feathericons.com/

[16]Font awesome : https://fontawesome.com/ (fas)

https://fontawesome.com/icons/desktop(fa)

[17]Regex for strong password, https://stackoverflow.com/questions/8141125/regex-for-password-php

[18] Selection of colors, https://encycolorpedia.com/

[19]downloading gifs : https://exrx.net/

[20]screen recording : https://www.freescreenrecording.com/

[21]convert to gifs : https://ezgif.com/video-to-gif

[22]resize and compress : https://compress-or-die.com/gif

Food images:

[23]http://google.co.in

[24]https://pixabay.com/

[25]https://www.pexels.com/

Tutorial videos:

[26]https://www.youtube.com/watch?v=QBxKlqGgb8o

[27]https://www.youtube.com/watch?v=ml6cT4AZdqI

[28]https://www.youtube.com/watch?v=HRkNfdlm5Qs

[29]https://www.youtube.com/watch?v=BHY0FxzoKZE

[30]https://www.youtube.com/watch?v=oc4QS2USKmk

[31]https://www.youtube.com/watch?v=8MAtXXXUvqo

Google fonts

[32]https://fonts.google.com/specimen/Poppins

[33]https://fonts.google.com/specimen/Ubuntu

[34]Queries related to css/bootstrap/html : https://www.w3schools.com/

[35]Queries related to js, https://www.tutorialspoint.com/index.htm

[36]Sql queries and php, https://stackoverflow.com/

[37]Table template, https://www.w3schools.com/css/css_table.asp

Github repository:

[38] Repo: https://github.com/likhitaavl2k/Stay-fit--A-web-application-which-tracks-your-fitness.git