# GYM MANAGER

*A*
*Case Study Report*

*Submitted in partial fulfilment of the*
*Requirements for the Course of*

**BACHELOR OF ENGINEERING**

IN

**INFORMATION TECHNOLOGY**

By

**P.Sai Venunath (1602-16-737-038)**

**K.Rohith(1602-17-737-030)**

*Under the guidance of*

**DR. K.RAM MOHAN RAO**
**Professor**

**Department of Information Technology**

**Vasavi College of Engineering (Autonomous)**

**(Affiliated to Osmania University)**

**Ibrahimbagh, Hyderabad-31**

**2019**

**Vasavi College of Engineering (Autonomous)**

**(Affiliated to Osmania University)**

**Hyderabad-500 031**

**Department of Information Technology**

# DECLARATION BY THE CANDIDATE

I, **P.SAI VENUNATH, K.ROHITH** bearing hall ticket number, **1602-16-737-038, 1602-17-737-030** respectively, hereby declare that the project report entitled **"GYM MANAGER"** under the guidance of **K.RAM MOHAN RAO,** HOD, Department of Information Technology, Vasavi College of Engineering, Hyderabad, is submitted in partial fulfilment of the requirement of MINI PROJECT of VI Semester of **Bachelor of Engineering** in **Information Technology**

This is a record of bonafide work carried out by me and the results embodied in this project.

**P.Sai Venunath**
**1602-16-737-038**

**K.Rohith**
**1602-17-737-030**

# Vasavi College of Engineering (Autonomous)

## (Affiliated to Osmania University)

## Hyderabad-500 031

## Department of Information Technology

## BONAFIDE CERTIFICATE

This is to certify that the project entitled **"GYM MANAGER"** being submitted by **P.SAI VENUNATH, K.ROHITH,** bearing **1602-16-737-038, 1602-17-737-030** respectively in partial fulfilment of the requirements for the completion of MINI PROJECT of Bachelor of Engineering, VI Semester, in Information Technology is a record of bonafide work carried out by him/her under my guidance.

**Dr. K. Ram Mohan Rao**
**HOD , IT**
**Internal Guide**

# ACKNOWLEDGEMENT

We take this opportunity with pride and enormous gratitude, to express the deeply embedded feeling and gratefulness to our respectable guide Dr K.Rammohan Rao sir, Department of Information Technology. Whose guidance was unforgettable and innovative ideas as well as his constructive suggestions has made the presentation of our project a grand success.

Finally at last but not least express our heart full thanks to the management of our college, Vasavi College of Engineering for providing the necessary arrangements and support to complete our seminar work successively.

P.Sai Venunath

1602-16-737-038

K.Rohith

1602-17-737-030

# ABSTRACT

We have created web based application used for managing a gym. This Gym manager shall enable the user to add members to a gym and manage the fee payment of the gym user. It is a very simple interface developed using python-django frame work. The user of the system shall be able to add a new gym member. The tool shall add all the necessary details like name, admission date and contact details into the system. The Gym Management System shall also monitor the timings for the member. This tool shall hold all the details of gym members.

# TABLE OF CONTENTS

# LIST OF FIGURES

# CHAPTER -1

# INTRODUCTION

Our project is **GYM MANAGER**. It is all about maintenance of the gym members, personal details, payment details and knowing about the monthly reports.

Through our designed web application software we provide the gym with the login id's through which we can easily access the member details and update.

The gym management can also get the monthly reports consisting the no. of members joined in particular month and to which batch they belong whether they have cleared the payment or regularly attendance. That is the complete details of the members can be easily accessed.

The gym management software provides lot of functions such as

- customer personal details
- fee particulars
- subscription type
- subscription period
- batch details (includes the time)

## 1.1Objective

This project is to facilitate a local gym and fitness center to automate its operations of keeping records of members

# CHAPTER-2
# SYSTEM ANALYSIS

## 2.1 **System Study**

It is always necessary to study and recognize the problems of existing system, which will help in finding out the requirements for the new system. System study helps in finding different alternatives for better solution.

**The project study basically deals with different operations:**

1: Data Gathering

2: Study of Existing System

3: Analyzing Problems

4: Studying various documents

5: Feasibility study for further improvements

**Following are the steps taken during the initial study:**

Initially, we collected all the information, which they wanted to store. Then we studied the working of the current system which is done manually. We noted the limitation of that system which motivated them to have new system. With the help of these documents we got basic ideas about the system as well as input output of the developed system

## 2.2 Existing System

An Existing system refers to the system that is being followed till now. The gym is working manually. The current system is time consuming and also it is very costly, because it involves a lot of paperwork. To manually handle the system was very difficult task. But now-a-days computerization made easy to work.

The following are the reasons why the current system should be computerized:

- To increase efficiency with reduced cost.
- To reduce the burden of paper work.
- To save time management for recording details of every member.
- To generate required reports easily

## 2.3 Proposed System

The online gym management system is user-friendly application. This automated system makes all functionality easier. It is very simple in design and to implement. The system requirements are very low. System resources and the system will work in almost all configurations.

It has the following objectives:

### Enhancement:

The main objective of Smart Gym Management System is to enhance and upgrade the existing system by increasing its efficiency and effectiveness. The software improves the working methods by replacing the existing manual system with the computer based system.

**Automation:**

The Smart Gym Management System automates each and every activity of the manual system and increases its throughput. Thus the response time of the system is very less and it works very fast.

**Accuracy:**

The Smart Gym Management System provides the uses a quick response with very accurate information regarding the users etc. Any details or system in an accurate manner, as and when required.

**Maintenance Cost:**

Reduce the cost of maintenance.

## 2.4 Architecture Used

### Front End

- HTML has been used for developing the user layout for the system
- BOOTSTRAP &CSS has been used for designing the web page of the system.
- Java Script has been used for creating all the validations and client side scripting functionality.

### Back End

- SQLLite-Xampp
- Django

# Chapter -3

# SYSTEM DESIGN

## 3.1 Use Cases

A **use case diagram** at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases .

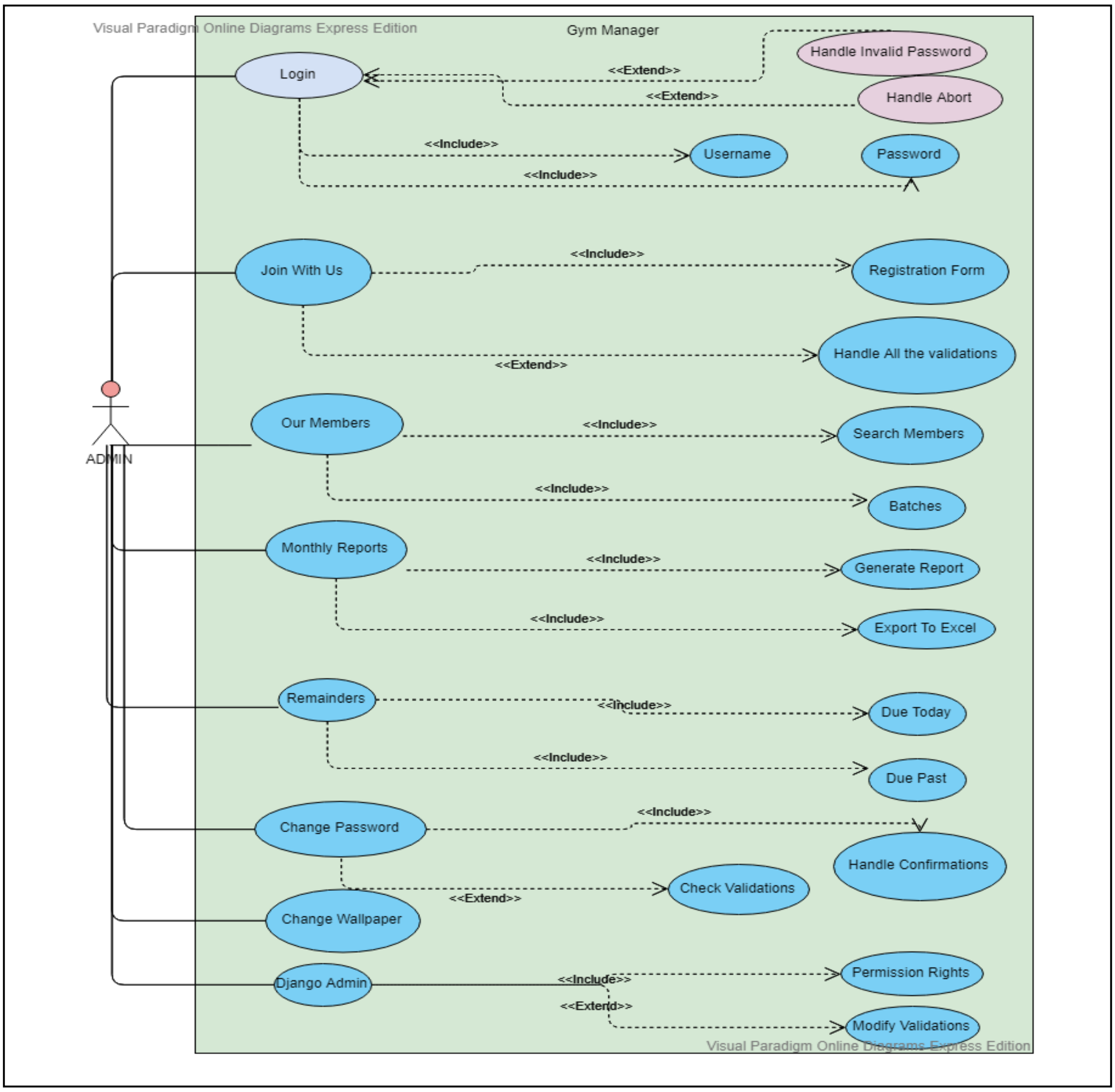


Fig 3.1 Use Case Diagram

## 3.2 UML Static Diagram

A **data-flow diagram** (DFD) is a way of representing a **flow** of a **data** of a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself.

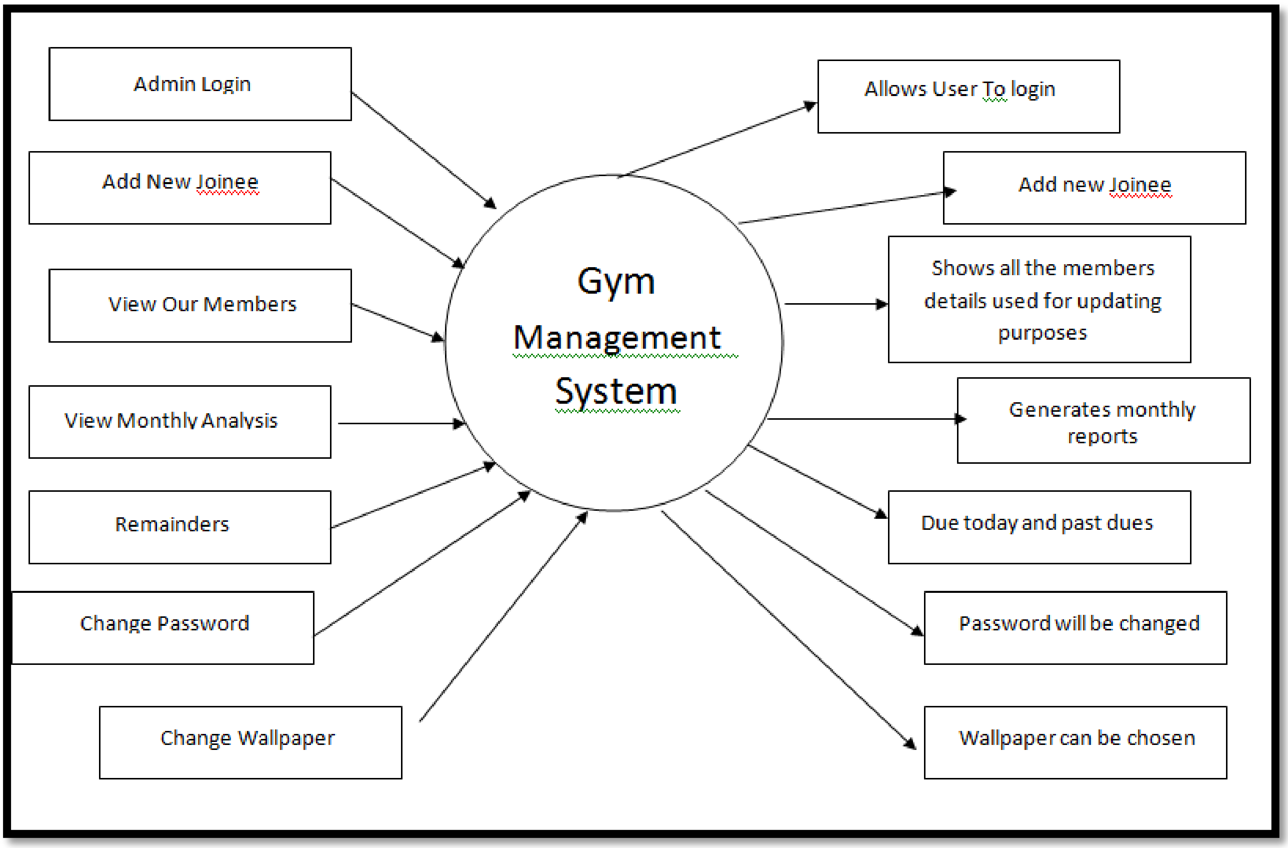


Fig 3.2 Data flow Diagram

## 3.3 UML Run time Diagram :

A **sequence diagram** shows object interactions arranged in time **sequence**. It depicts the objects and classes involved in the scenario and the **sequence** of messages exchanged between the objects needed to carry out the functionality of the scenario.

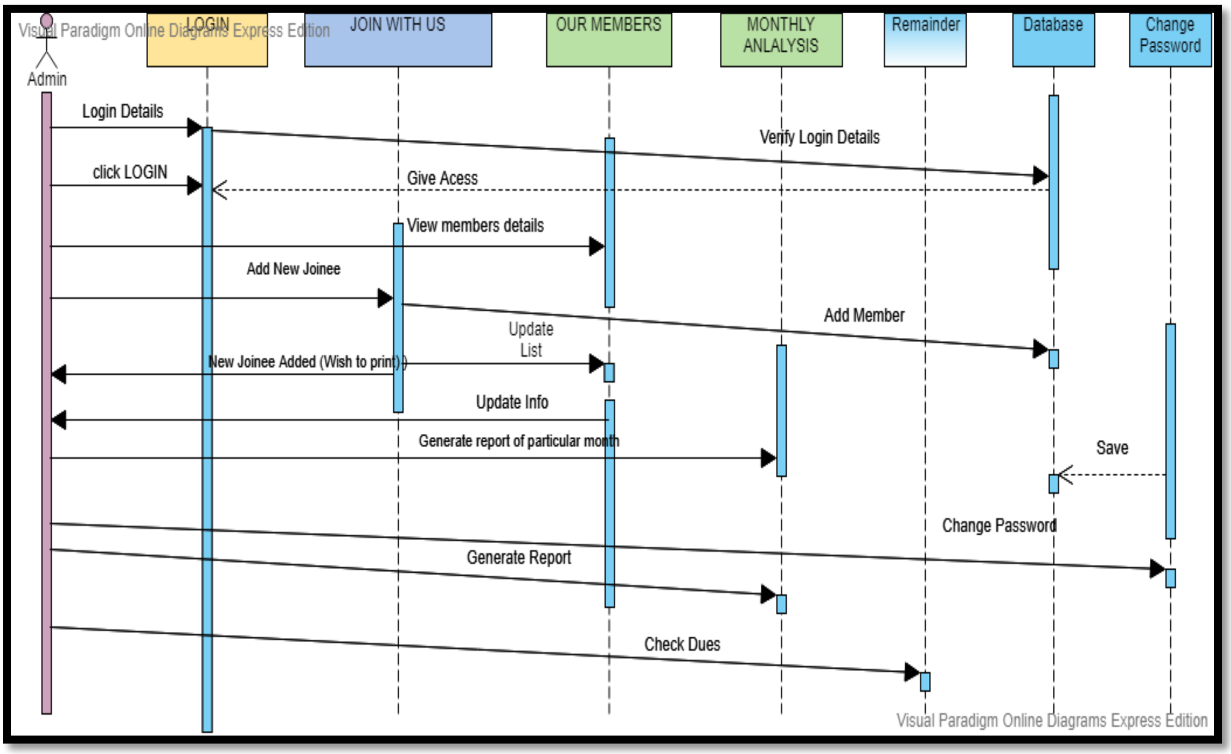


Fig 3.3 Sequence Diagram

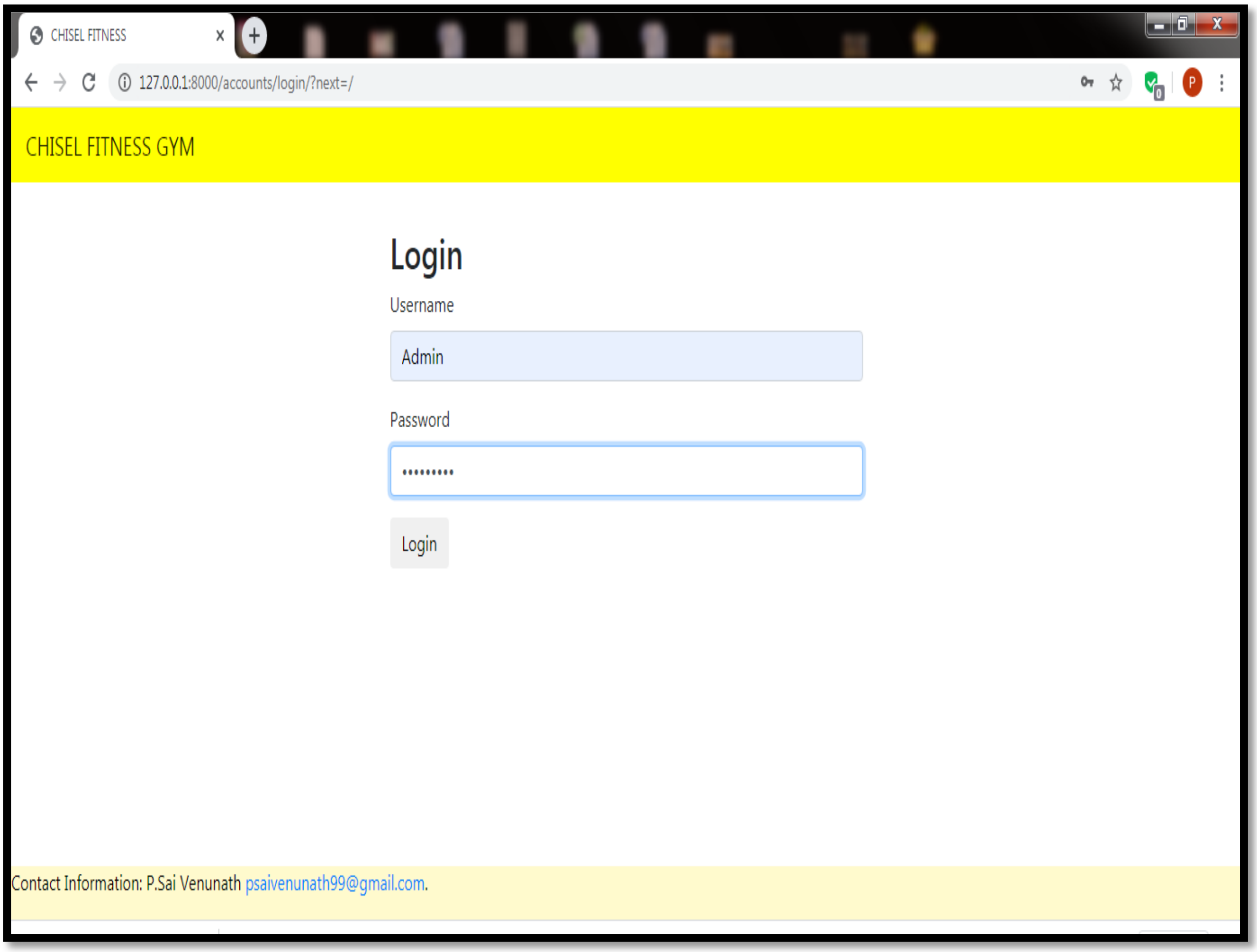
# CHAPTER 4

# IMPLEMENTATION

In our project we have the following modules

## 4 .1 Login Module

Through this module we can enter into our web application by typing username with a secured password. Password must be as per validations.



Login

Username

Admin

Password

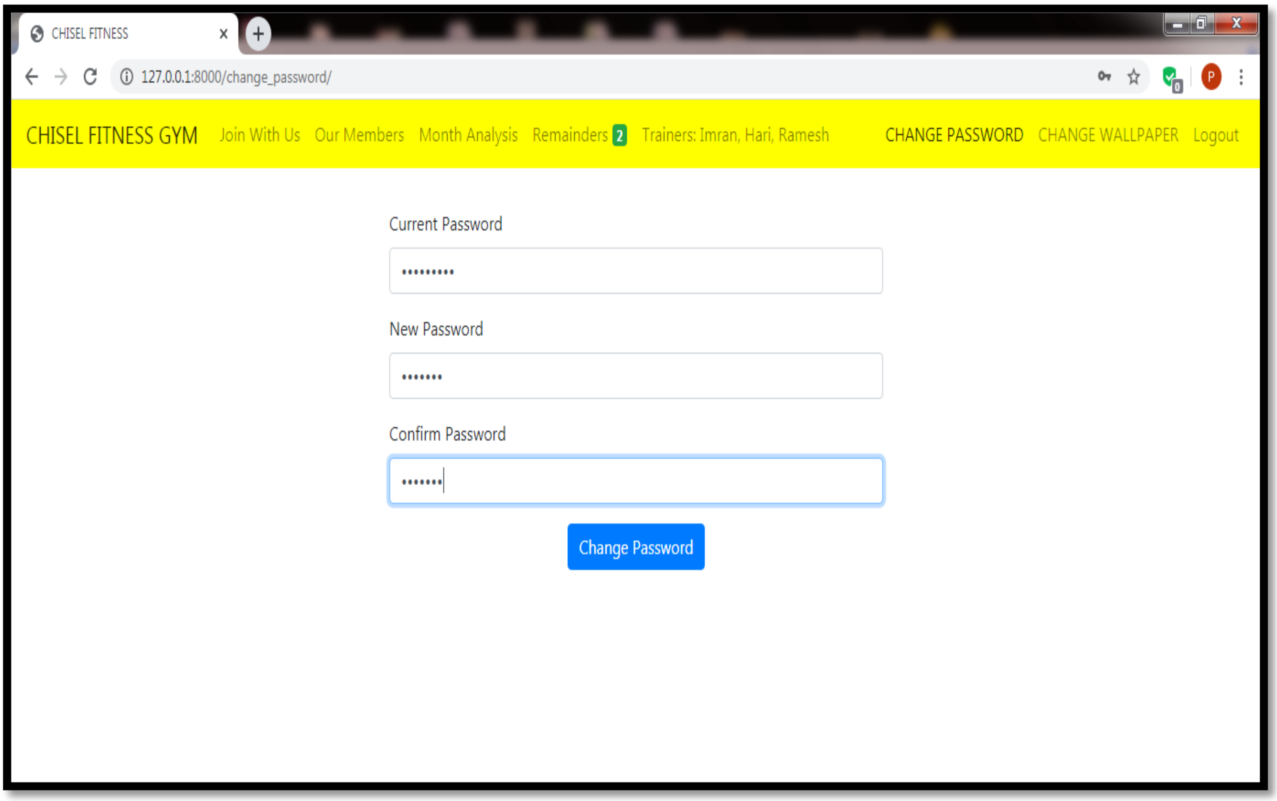
•••••••••

Contact Information: P.Sai Venunath psaivenunath99@gmail.com.

Fig 4.1 Login Module

## 4.2 Change Password

The user can change the password as per the validations.



Current Password

•••••••••

New Password

•••••••

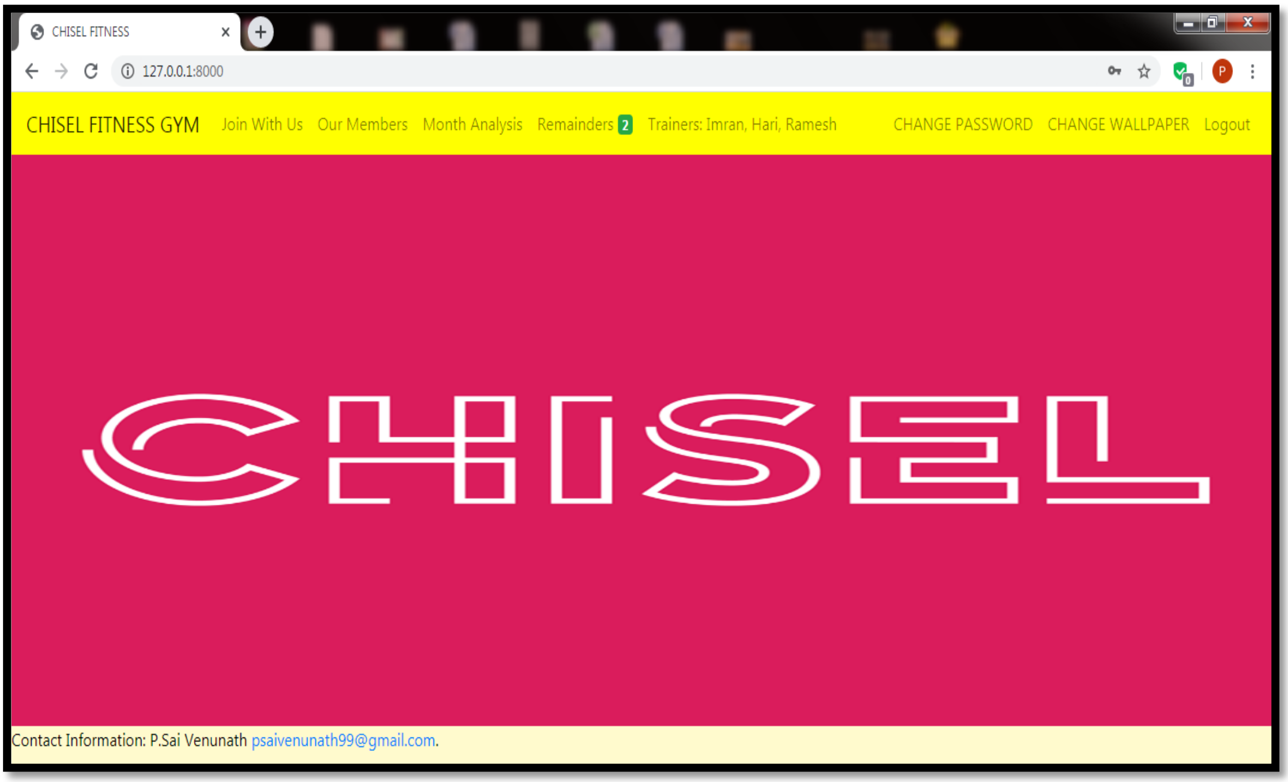
Confirm Password

•••••••

Fig 4.2 Change Password Module

## 4.3 Home Page

Once the user is logged in finds his home page consisting of many easily accessible modules. He can change the wallpaper as per his taste and choice and can navigate through the app through the options available on navigation bar.



Contact Information: P.Sai Venunath psaivenunath99@gmail.com.

Fig 4.3 Home Page Module

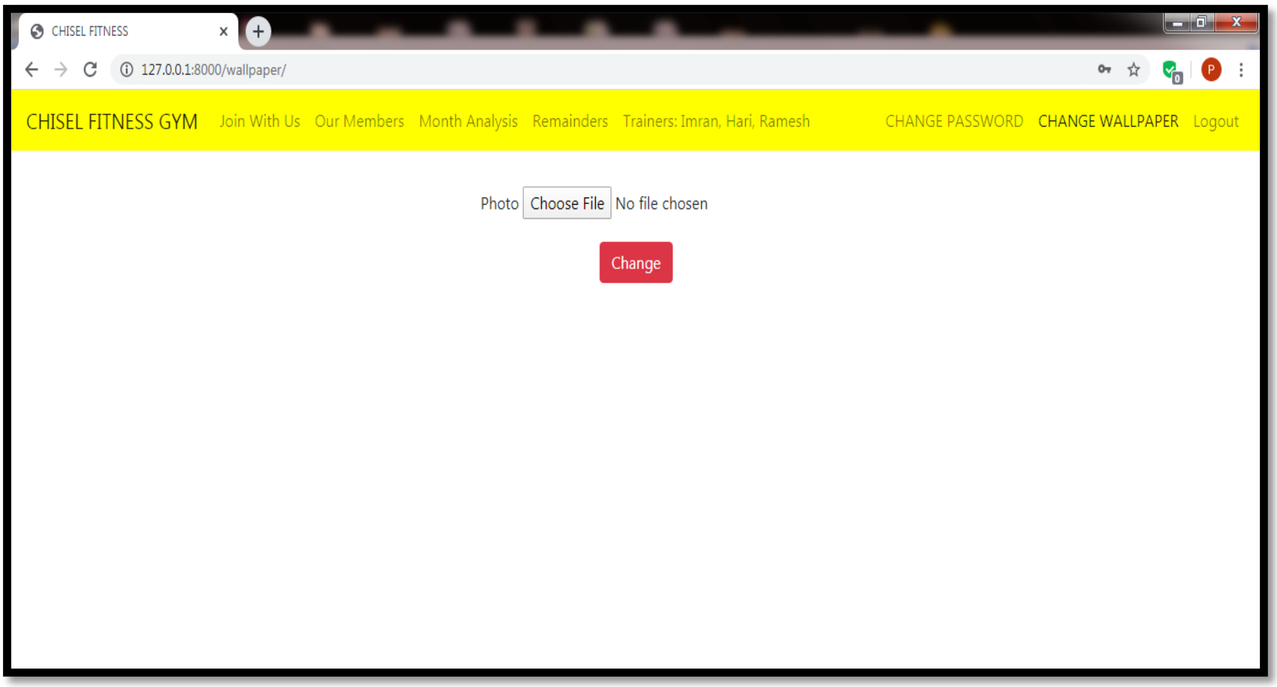
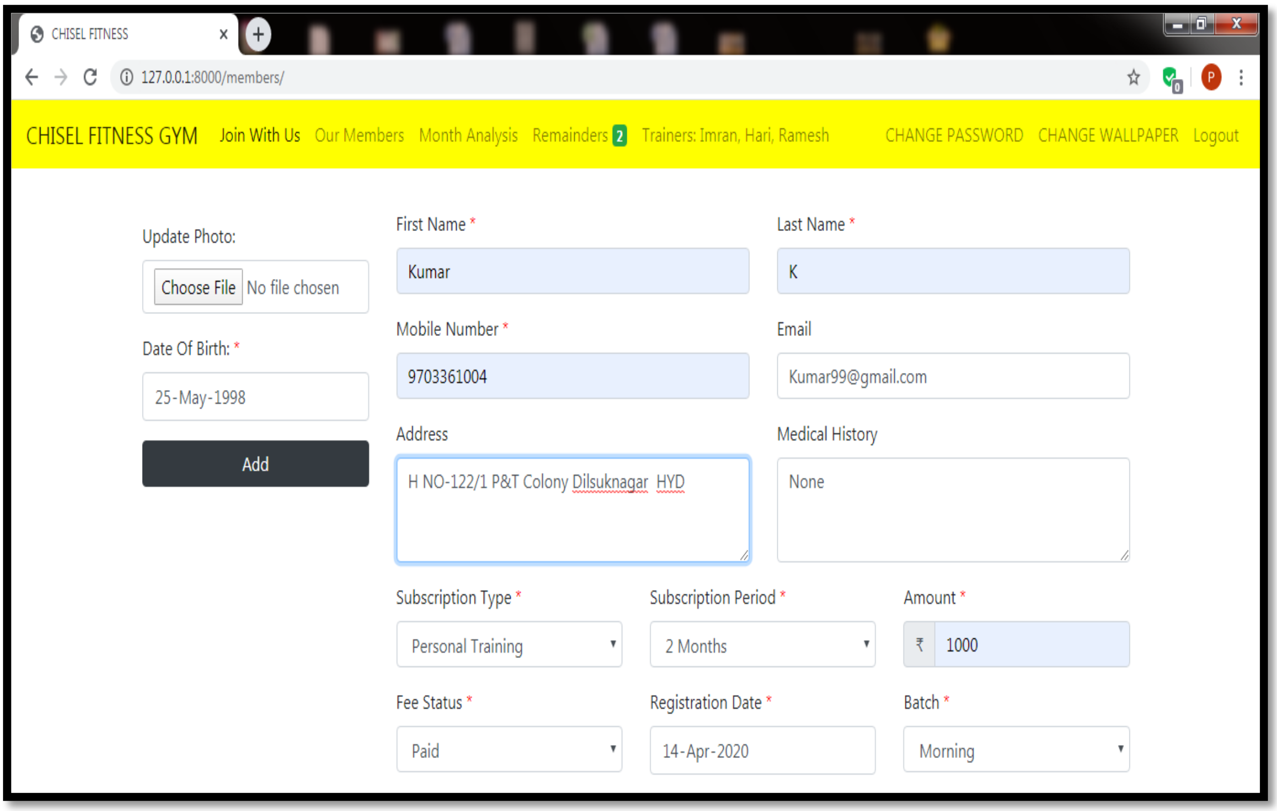


Fig 4.3.2Change Background Module

## 4.4 Join With Us

This module helps administrator to add new customers of the gym i.e the registration form. Here all the details of the customer will be taken like name, DOB, house address, subscription type, fee status.



Update Photo: No file chosen

Date Of Birth: * 25-May-1998

First Name * Kumar

Last Name * K

Mobile Number * 9703361004

Email Kumar99@gmail.com

Address H NO-122/1 P&T Colony Dilsuknagar HYD

Medical History None

Subscription Type * Personal Training

Subscription Period * 2 Months

Amount * ₹ 1000

Fee Status * Paid

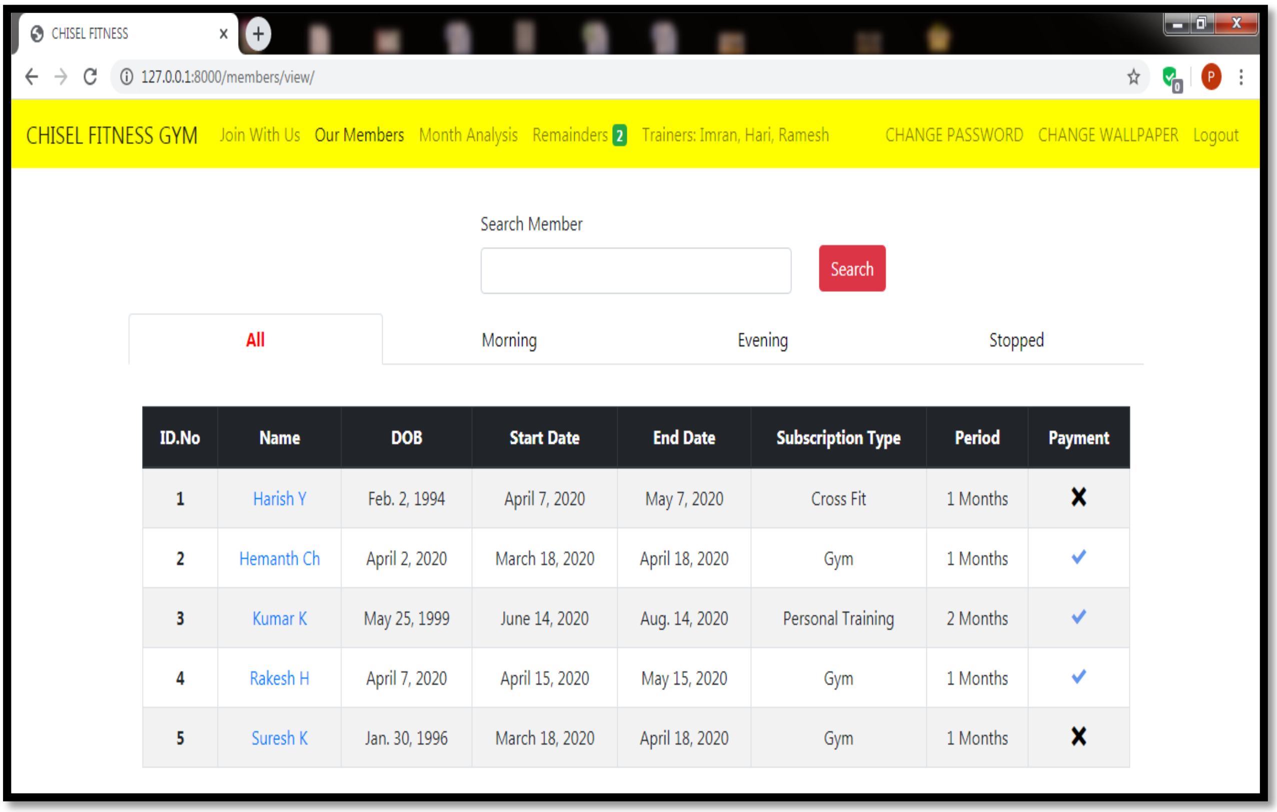
Registration Date * 14-Apr-2020

Batch * Morning

Fig 4.4 Join With Us Module

## 4.5 Our Members

In this module we will know the total members details of the gym like their joining date, completion date their subscription type and their batches. Here the user can also know about payment status of the customer and also whether is continuing or left.

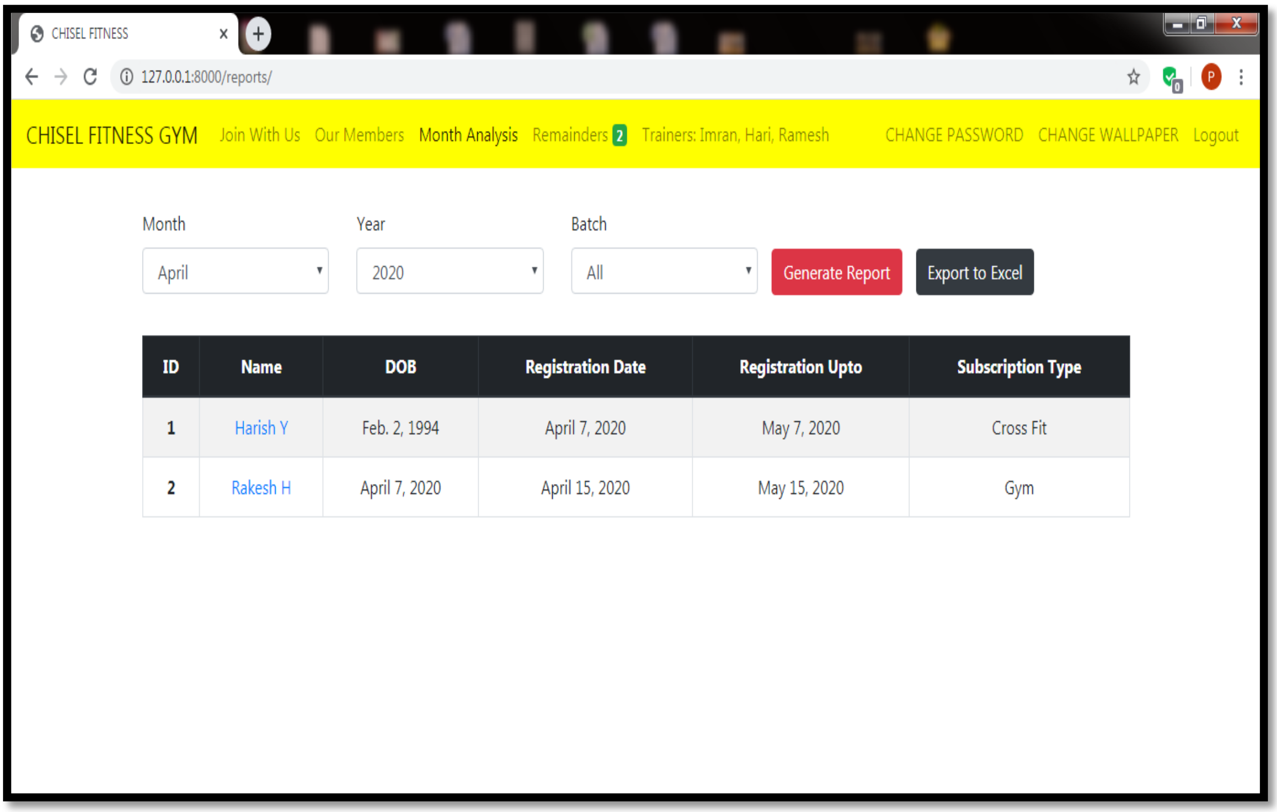


| ID.No | Name | DOB | Start Date | End Date | Subscription Type | Period | Payment |
|---|---|---|---|---|---|---|---|
| 1 | Harish Y | Feb. 2, 1994 | April 7, 2020 | May 7, 2020 | Cross Fit | 1 Months | ✗ |
| 2 | Hemanth Ch | April 2, 2020 | March 18, 2020 | April 18, 2020 | Gym | 1 Months | ✓ |
| 3 | Kumar K | May 25, 1999 | June 14, 2020 | Aug. 14, 2020 | Personal Training | 2 Months | ✓ |
| 4 | Rakesh H | April 7, 2020 | April 15, 2020 | May 15, 2020 | Gym | 1 Months | ✓ |
| 5 | Suresh K | Jan. 30, 1996 | March 18, 2020 | April 18, 2020 | Gym | 1 Months | ✗ |

Fig 4.5 Our Members Module

## 4.6 Monthly Analysis

This module helps in generating Monthly reports.



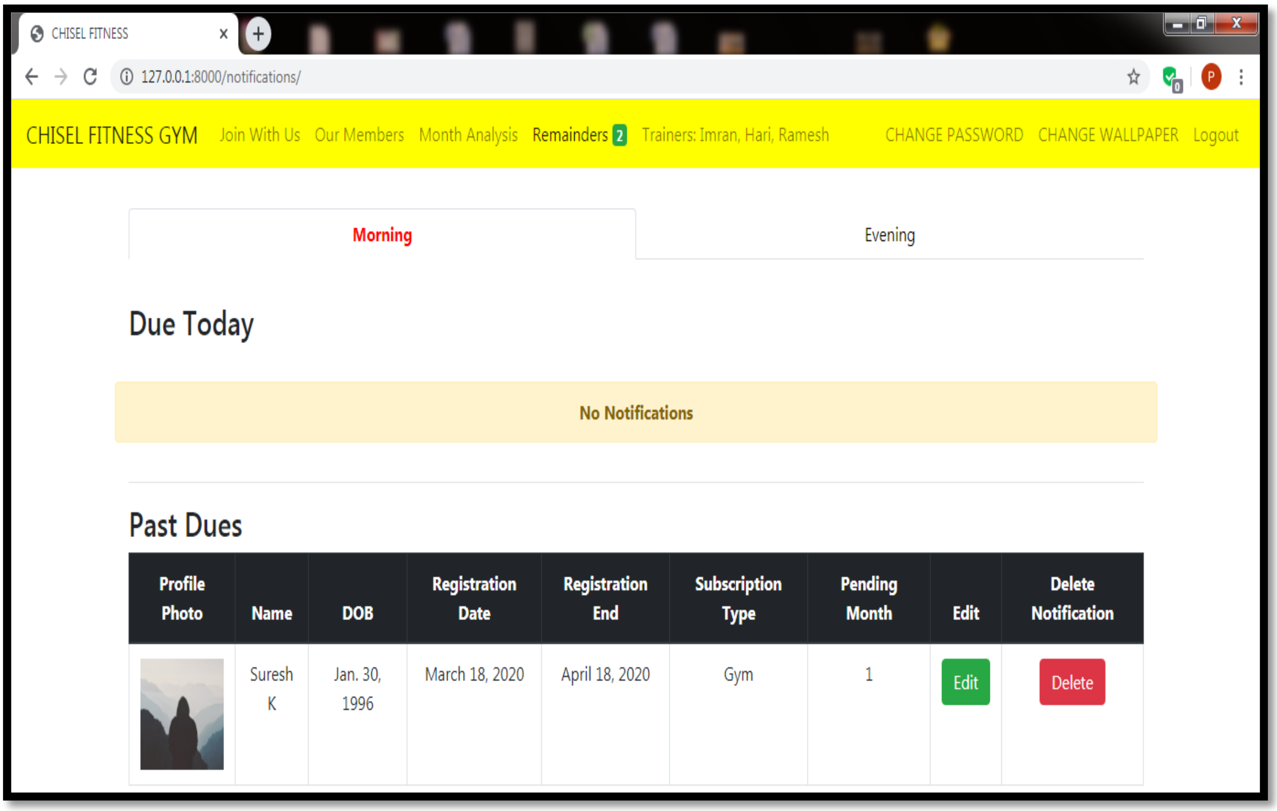
Month: April
Year: 2020
Batch: All

| ID | Name | DOB | Registration Date | Registration Upto | Subscription Type |
|---|---|---|---|---|---|
| 1 | Harish Y | Feb. 2, 1994 | April 7, 2020 | May 7, 2020 | Cross Fit |
| 2 | Rakesh H | April 7, 2020 | April 15, 2020 | May 15, 2020 | Gym |

Fig 4.6 Monthly Analysis Module

## 4.7 Remainders

This module helps in identifying the members with due list.



Due Today

No Notifications

Past Dues

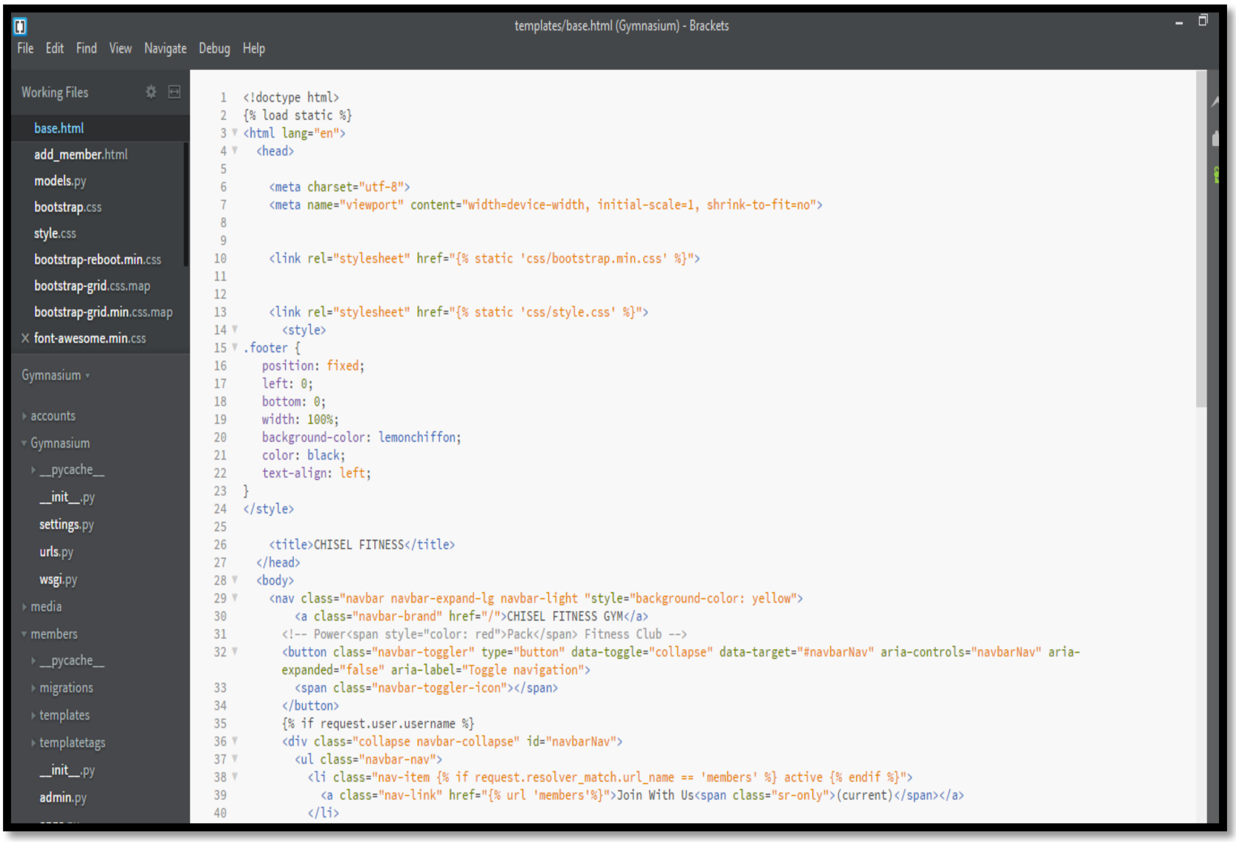
| Profile Photo | Name | DOB | Registration Date | Registration End | Subscription Type | Pending Month | Edit | Delete Notification |
|---|---|---|---|---|---|---|---|---|
| | Suresh K | Jan. 30, 1996 | March 18, 2020 | April 18, 2020 | Gym | 1 | Edit | Delete |

Fig 4.7 Remainders module

## 4.8 LOGIC

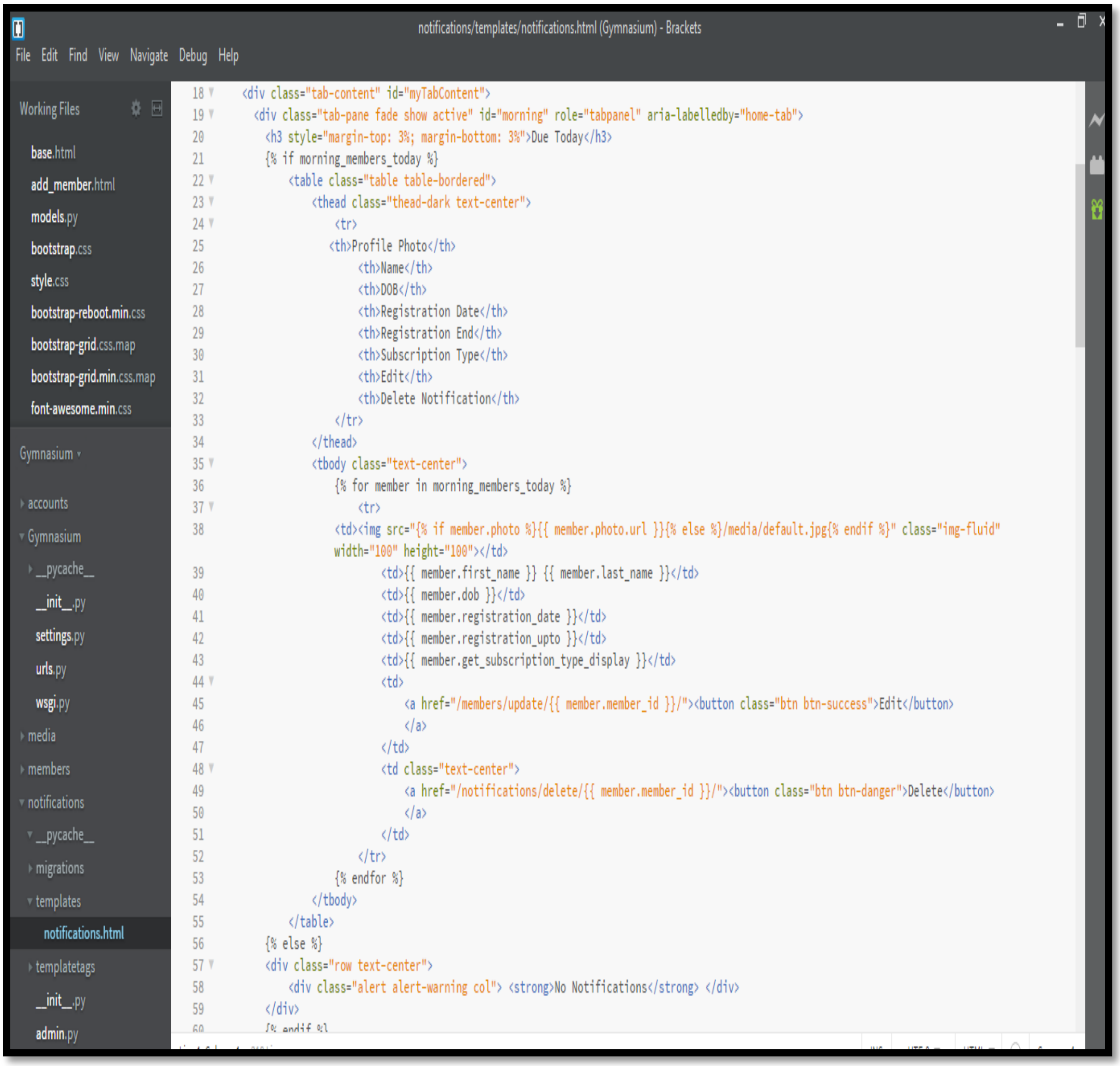
base.html

Z



```
<!doctype html>
{% load static %}
<html lang="en">
  <head>

    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">


    <link rel="stylesheet" href="{% static 'css/bootstrap.min.css' %}">


    <link rel="stylesheet" href="{% static 'css/style.css' %}">
      <style>
.footer {
   position: fixed;
   left: 0;
   bottom: 0;
   width: 100%;
   background-color: lemonchiffon;
   color: black;
   text-align: left;
}
</style>

    <title>CHISEL FITNESS</title>
  </head>
  <body>
    <nav class="navbar navbar-expand-lg navbar-light "style="background-color: yellow">
        <a class="navbar-brand" href="/">CHISEL FITNESS GYM</a>
      <!-- Power<span style="color: red">Pack</span> Fitness Club -->
      <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">
        <span class="navbar-toggler-icon"></span>
      </button>
      {% if request.user.username %}
      <div class="collapse navbar-collapse" id="navbarNav">
        <ul class="navbar-nav">
          <li class="nav-item {% if request.resolver_match.url_name == 'members' %} active {% endif %}">
            <a class="nav-link" href="{% url 'members'%}">Join With Us<span class="sr-only">(current)</span></a>
          </li>
```

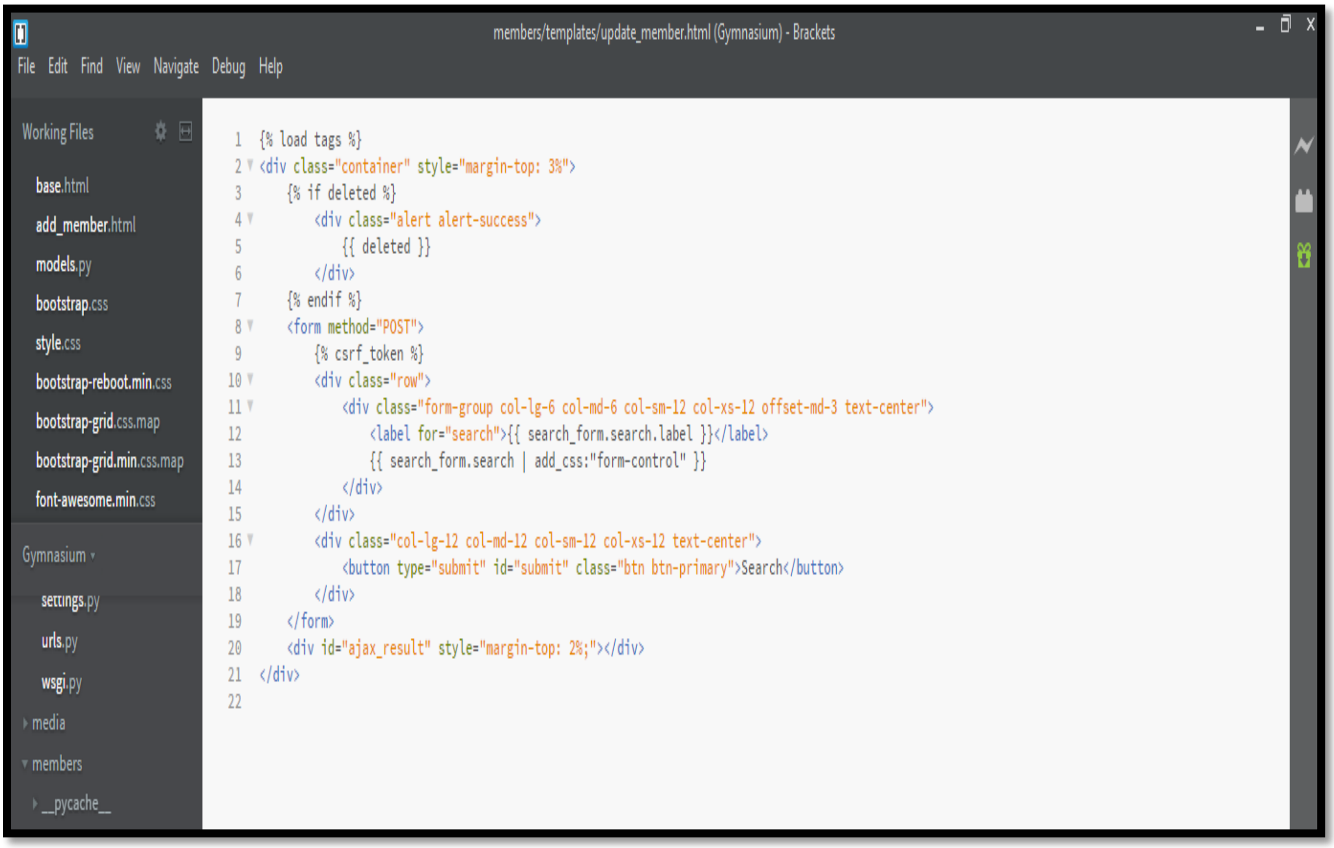
Fig 4.8.1 Base.html



```
<div class="tab-content" id="myTabContent">
 <div class="tab-pane fade show active" id="morning" role="tabpanel" aria-labelledby="home-tab">
  <h3 style="margin-top: 3%; margin-bottom: 3%">Due Today</h3>
  {% if morning_members_today %}
    <table class="table table-bordered">
      <thead class="thead-dark text-center">
        <tr>
       <th>Profile Photo</th>
          <th>Name</th>
          <th>DOB</th>
          <th>Registration Date</th>
          <th>Registration End</th>
          <th>Subscription Type</th>
          <th>Edit</th>
          <th>Delete Notification</th>
        </tr>
      </thead>
      <tbody class="text-center">
        {% for member in morning_members_today %}
          <tr>
        <td><img src="{% if member.photo %}{{ member.photo.url }}{% else %}/media/default.jpg{% endif %}" class="img-fluid"
        width="100" height="100"></td>
            <td>{{ member.first_name }} {{ member.last_name }}</td>
            <td>{{ member.dob }}</td>
            <td>{{ member.registration_date }}</td>
            <td>{{ member.registration_upto }}</td>
            <td>{{ member.get_subscription_type_display }}</td>
            <td>
              <a href="/members/update/{{ member.member_id }}/"><button class="btn btn-success">Edit</button>
              </a>
            </td>
            <td class="text-center">
              <a href="/notifications/delete/{{ member.member_id }}/"><button class="btn btn-danger">Delete</button>
              </a>
            </td>
          </tr>
        {% endfor %}
      </tbody>
    </table>
  {% else %}
  <div class="row text-center">
    <div class="alert alert-warning col"> <strong>No Notifications</strong> </div>
  </div>
  {% endif %}
```

Fig 4.8.2 Notification.html

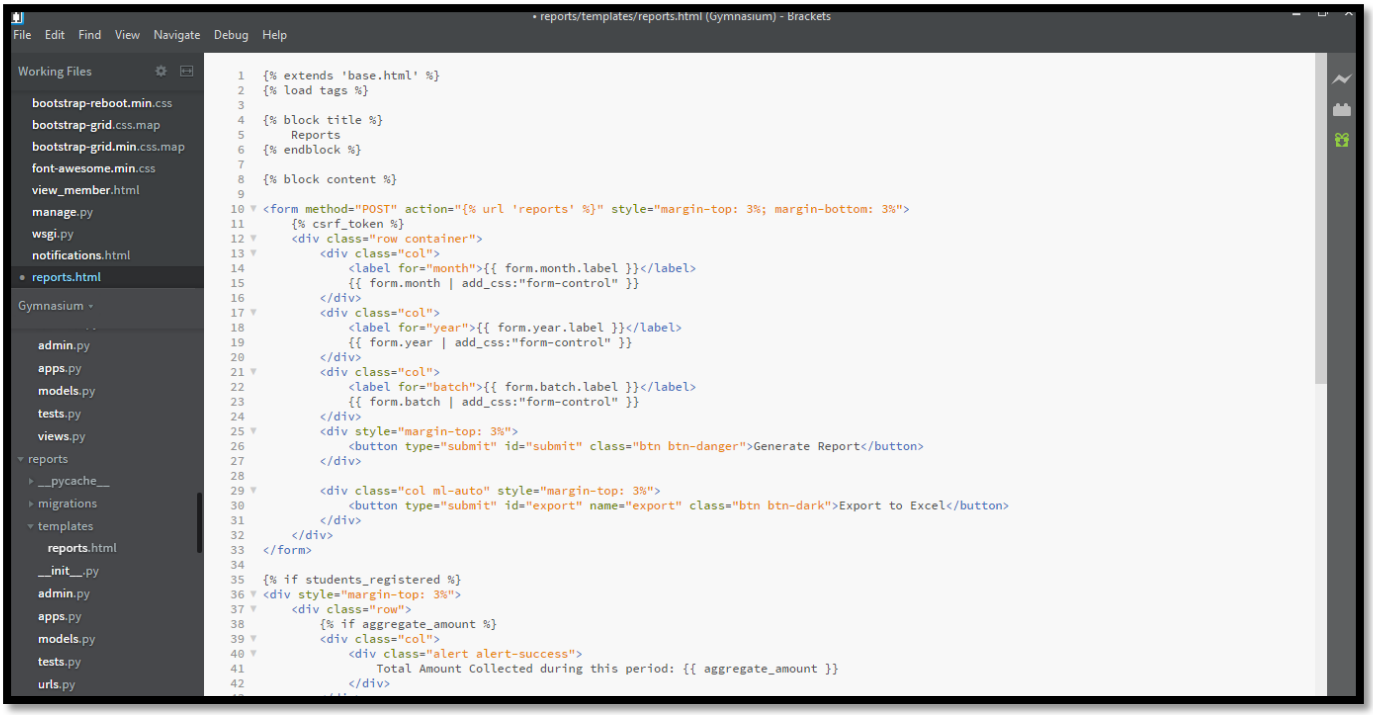
# updatemember.html



```
{% load tags %}
<div class="container" style="margin-top: 3%">
    {% if deleted %}
        <div class="alert alert-success">
            {{ deleted }}
        </div>
    {% endif %}
    <form method="POST">
        {% csrf_token %}
        <div class="row">
            <div class="form-group col-lg-6 col-md-6 col-sm-12 col-xs-12 offset-md-3 text-center">
                <label for="search">{{ search_form.search.label }}</label>
                {{ search_form.search | add_css:"form-control" }}
            </div>
        </div>
        <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12 text-center">
            <button type="submit" id="submit" class="btn btn-primary">Search</button>
        </div>
    </form>
    <div id="ajax_result" style="margin-top: 2%;"></div>
</div>

```

Fig 4.8.3 updatemember.html

# reports.html



```
{% extends 'base.html' %}
{% load tags %}

{% block title %}
    Reports
{% endblock %}

{% block content %}

<form method="POST" action="{% url 'reports' %}" style="margin-top: 3%; margin-bottom: 3%">
    {% csrf_token %}
    <div class="row container">
        <div class="col">
            <label for="month">{{ form.month.label }}</label>
            {{ form.month | add_css:"form-control" }}
        </div>
        <div class="col">
            <label for="year">{{ form.year.label }}</label>
            {{ form.year | add_css:"form-control" }}
        </div>
        <div class="col">
            <label for="batch">{{ form.batch.label }}</label>
            {{ form.batch | add_css:"form-control" }}
        </div>
        <div style="margin-top: 3%">
            <button type="submit" id="submit" class="btn btn-danger">Generate Report</button>
        </div>

        <div class="col ml-auto" style="margin-top: 3%">
            <button type="submit" id="export" name="export" class="btn btn-dark">Export to Excel</button>
        </div>
    </div>
</form>

{% if students_registered %}
<div style="margin-top: 3%">
    <div class="row">
        {% if aggregate_amount %}
        <div class="col">
            <div class="alert alert-success">
                Total Amount Collected during this period: {{ aggregate_amount }}
            </div>
```

Fig 4.8.4 Reports.html

# Urls.py

```
from django.urls import path, include

from . import views

from django.contrib.auth.decorators import login_required


urlpatterns = [

    path('', login_required(views.reports), name='reports'),

    path('export/all/', login_required(views.export_all), name='export_all'),

]
```

## Config.py

```
from members.models import Member
import datetime
from django.db.models import Q
from django.db.models.signals import post_save


def my_handler(sender, instance, created, **kwargs):
    query = Q(
            Q(registration_upto__gte=datetime.datetime.today()),
            Q(registration_upto__lte=datetime.datetime.today() + datetime.timedelta(days=1)),
            Q(Q(notification=2) | Q(notification=0)))
    members_before = Member.objects.filter(
             registration_upto__lte=datetime.datetime.today())
    members_today = Member.objects.filter(query)


     count = 0
    for member in members_today | members_before:
        if member.notification != 0:
            member.notification = 1
            member.fee_status = 'pending'
            member.save()
    return


post_save.connect(my_handler, sender=Member)
```

```
def run_notifier(**kwargs):
    query = Q(
            Q(registration_upto__gte=datetime.datetime.today()),
            Q(registration_upto__lte=datetime.datetime.today() + datetime.timedelta(days=1)),
            Q(Q(notification=2) | Q(notification=0)))



    members_before = Member.objects.filter(
        registration_upto__lte=datetime.datetime.today()).exclude(stop=1)
    members_today = Member.objects.filter(query).exclude(stop=1)


    count = 0
    for member in members_today | members_before:
        if member.notification != 0:
            member.notification = 1
            member.fee_status = 'pending'
            post_save.disconnect(my_handler, sender=Member)
            member.save()
            post_save.connect(my_handler, sender=Member)
    return
```

```
def get_notification_count():
    NOTIF_COUNT = Member.objects.filter(
        registration_upto__gte=datetime.datetime.now(),
        registration_upto__lte=datetime.date.today() + datetime.timedelta(days=1),
        notification=1).exclude(stop=1)
    PENDING_COUNT = Member.objects.filter(fee_status='pending',
notification=1).exclude(stop=1)
    return (NOTIF_COUNT | PENDING_COUNT).distinct().count()
```

## 4.9 Github Link:

https://github.com/saivenunath99/Gym-Manager

## 4.10 Folder Structure :

https://drive.google.com/drive/u/0/folders/12UlvfdK3uclMvHRadv0xTzYixtoh6diO

# CHAPTER 5

# Testing

1) Login Password must be 8 characters which should include

- One Lowercase Letter
- one Uppercase Letter
- one Special Character
- one Number.
- Username and Password must be valid

2) Join With Us :

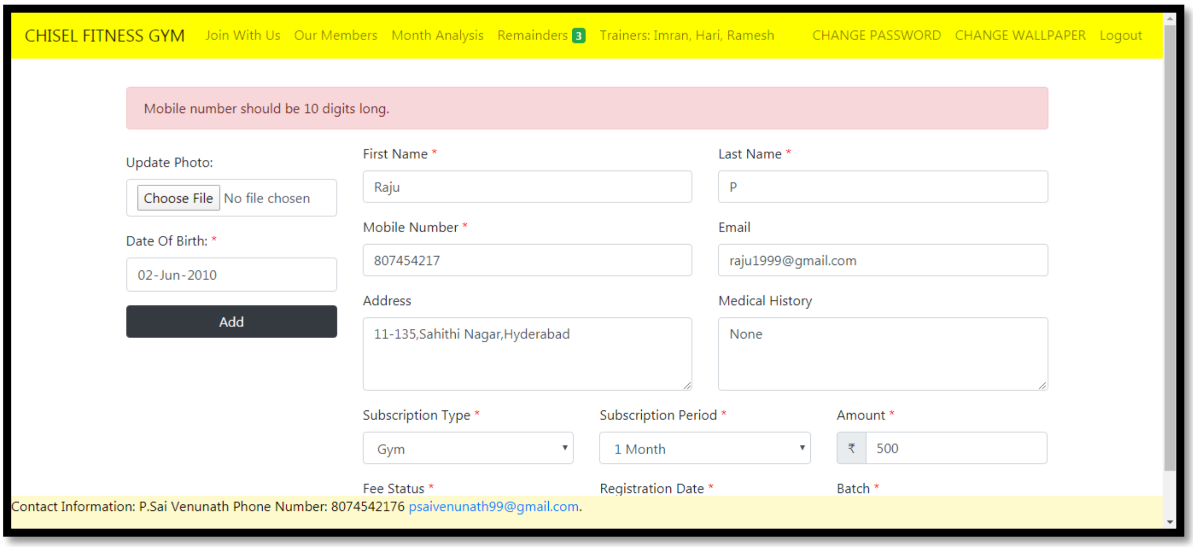
- All the text fields that have * mark are required
- Mobile Number must contain 10 digit
- Mobile Number should not be the same

3) Our Members

- If entered field in search bar is not recognized then “No Matches Found” appears.

4) Change Password

- New Password must match all the validations.



Mobile number should be 10 digits long.

Update Photo: No file chosen

Date Of Birth: * 02-Jun-2010

First Name * Raju

Last Name * P

Mobile Number * 807454217

Email raju1999@gmail.com

Address 11-135,Sahithi Nagar,Hyderabad

Medical History None

Subscription Type * Gym

Subscription Period * 1 Month

Amount * ₹ 500

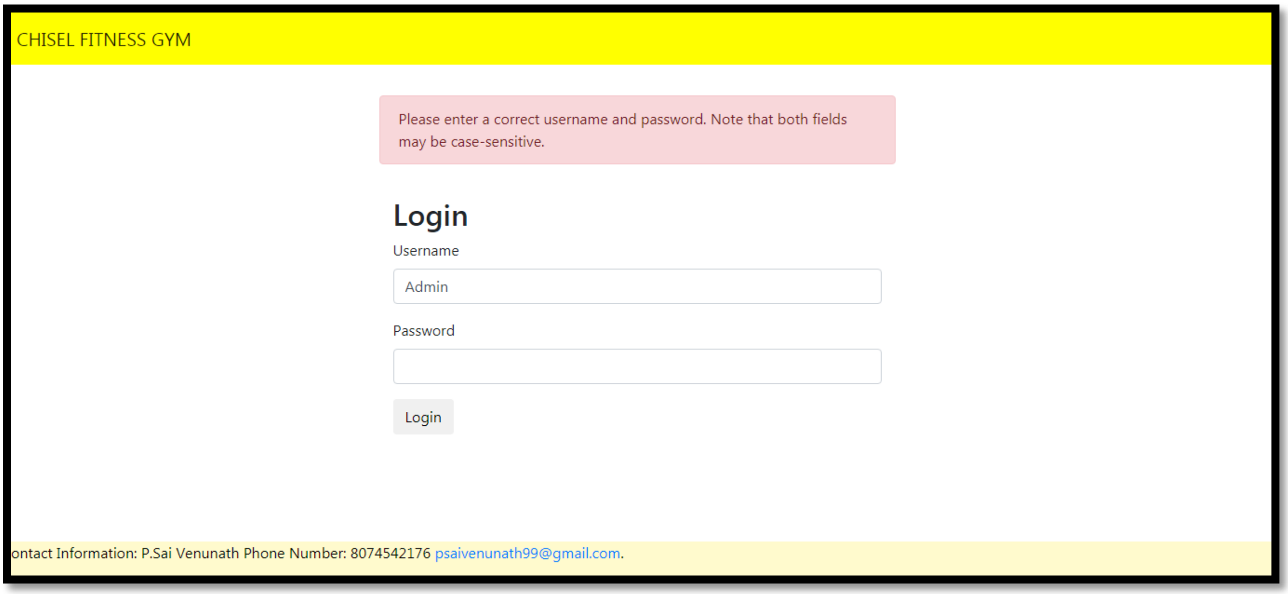
Fee Status *

Registration Date *

Batch *

Contact Information: P.Sai Venunath Phone Number: 8074542176 psaivenunath99@gmail.com.

Fig 5.1 Mobile number validation



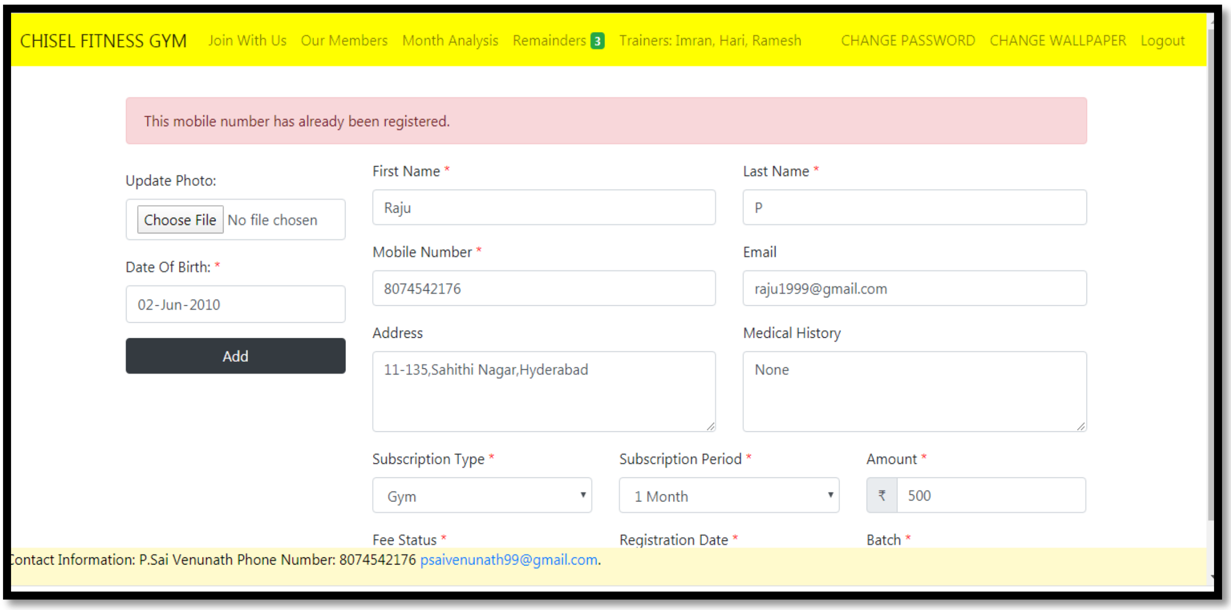
Please enter a correct username and password. Note that both fields may be case-sensitive.

Username: Admin

Password:

ontact Information: P.Sai Venunath Phone Number: 8074542176 psaivenunath99@gmail.com.

Fig 5.2 Login validation



This mobile number has already been registered.

Update Photo: Choose File No file chosen

Date Of Birth: * 02-Jun-2010

First Name * Raju

Last Name * P

Mobile Number * 8074542176

Email raju1999@gmail.com

Address 11-135,Sahithi Nagar,Hyderabad

Medical History None

Subscription Type * Gym

Subscription Period * 1 Month

Amount * ₹ 500

Fee Status *

Registration Date *

Batch *

Contact Information: P.Sai Venunath Phone Number: 8074542176 psaivenunath99@gmail.com.

Fig 5.3 Registration validation

# CHAPTER 6

# RESULTS

**Now, through this project we have developed an easy application by using django framework.**

**Accuracy and Correctness is improved by our project**

## 6.1 Project Features

1) Easy to Use
2) Login Management
3) Easy to add a new member by taking their details through **Join with us module**
4) Display all the details of members in **Our Member Module**
5) Generate **Monthly Reports** that helps Administrators of gym
6) Reminds Gym Administration about fee status of members
7) Notifies Dues on time

# CHAPTER 7

# CONCLUSION AND FUTURE SCOPE

## 7.1 Conclusion

The “ GYM MANAGER” is successfully designed and developed to fulfilling the necessary requirements, as identified in the requirements analysis phase,such as maintaining the records efficiently and accurately . The old manual system was suffering from a series of drawbacks. The present project has been developed to meet the aspirations indicated in the modern age.

## 7.2 Future Scope

- Notifications of Fee Payment can be sent to the customer’s mobile number a day before the deadline.
- Health care Chabot can be introduced in the website so that customers can clarify their issues on fitness.
- Customer heart rates are monitored by trainers while they are exercising through advancement of IOT.

# CHAPTER 8

# REFERENCES

- https://www.udemy.com/course/projects-in-django-learn-django-building-projects/
- https://docs.djangoproject.com/en/3.0/ref/contrib/admin/
- https://getbootstrap.com/docs/4.0/components/navbar/
- https://www.w3schools.com/js/
- https://www.youtube.com/watch?v=OTmQOjsl0eg
- https://www.apachefriends.org/index.html