શ્રી સ્વામિનારાયણ ગુરુકુલ રાજકોટ સંસ્થાન
શાસ્ત્રી સ્વામી શ્રી ધર્મજીવનદાસજી
સાયન્સ & IT ગુરુકુલ કોલેજ
ગુરુકુલ કેમ્પસ, કોલેજ રોડ, જૂનાગઢ

Mahamahal Food Zone

"MahaMahal" is a Django-based website with an SQLite database that provides a platform for users to log in, explore, and review food. This project report outlines the development journey of MahaMahal, detailing its technical architecture,designdecisions, user experience considerations, security measures, and scalability strategies.

Project Partners:

MR. VIVEK D. THUMMAR, BCA-6TH

::Submitted to::
BKNM University, Junagadh

::GUIDED BY::

Mr. Ripal V. Pandya

Mr. Milind V. Anandpara

શ્રી સ્વામિનારાયણ ગુરુકુલ રાજકોટ સંસ્થાન

શાસ્ત્રી સ્વામી શ્રી ધર્મજીવનદાસજી

સાયન્સ & IT ગુરુકુલ કોલેજ

ગુરુકુલ કેમ્પસ, કોલેજ રોડ, જૂનાગઢ

(Affiliated to **Bhakta Kavi Narsinh Mehta University, Junagadh**)

Project Completion Certificate

This certificate is awarded to

Mr. Vivek D. Thummar — **BCA6-2024**

in completion of project work

Django — **19/12/2024**

SQLITE — **11/03/2024**

Mr. Ripal V. Pandya

Mr. Milind V. Anandpara

Project Guide

Director

www.sssdiit.junagadhgurukul.org

A

PROJECT REPORT ON

MAHAMAHAL FOOD ZONE

Submitted in Fulfillment of Requirements

For Completion of Semester - 6 in

Bachelor of computer application

Year 2024

To

SHASHTRI SWAMI SHREE DHARMAJIVANDASJI INSTITUTE OF INFORMATION TECHNOLOGY

JUNAGADH

Guided By:

Prof. Ripal V. Pandya

Prof. Milind V. Anandpara

Prepared By:

Mr. Vivek D. Thummar

PREFACE

In an era where culinary exploration is not just a pastime but a thriving culture, the demand for platforms facilitating food exploration and critique has surged. The advent of digital technology has revolutionized the way we interact with food, transcending geographical boundaries and culinary traditions. In response to this evolving landscape, I embarked on a journey to create a digital hub where gastronomic enthusiasts could converge, share experiences, and indulge in the delightful world of flavors.

Welcome to "Mahamahal," an online platform meticulously crafted to serve as a sanctuary for food aficionados. Rooted in the ethos of community engagement and gastronomic enlightenment, Mahamahal stands as a testament to the fusion of technology and culinary arts. Built upon the robust Django framework and powered by the versatile SQLite database, Mahamahal encapsulates the essence of modern-day culinary discourse.

This project report chronicles the inception, development, and culmination of Mahamahal—a labor of love that encapsulates countless hours of coding, designing, and refining. From conceptualization to deployment, each phase of the project has been meticulously documented to offer a comprehensive insight into the intricacies of its creation.

Vivek D. Thummar

ACKNOWLEDGEMENT

We are very thankful to all whose have helped in preparing this project. We are feeling a great happiness to present this website project. First of all we would like to thank **"BKNM University"** who give me an opportunity to give a chance to prepare a project.

Before we get in to thick of the things, we would to add a few heartfelt words for the people who were part of this project numerous ways, people who give unending support right from the stage project ideas was conceived. In particular we would like to thank **Prof. Ripal V. Pandya & Prof. Milind V. Anandpara (Project Guide)**, who has always inspired us and has directed us towards the successful completion of our project. They have been the guided through the project and their encouragement has left me indebted to them.

We are very thankful to the **Director Sadhu RushikeshdashjiSwami** and the **Asst. Director Mr. Rajesh Bharad** of **Shastri Swami Shree Dharmajivandasji Institute of Information Technology – Junagadh**.

We are also thankful to**(Milind Vaghasiya SDE)** our classmate**(Himanshu Paghadar, Deven Katara)** and few other people who helped us directly or indirectly in solving problem and in making our web development project more efficientand attractive.

Thank you…

Date: 07/10/2023 Mr. Vivek D. Thummar

Place: JUNAGADH

I N D E X

PROJECT PROFILE

Project Title	Mahamahel food zone
Project Description	"MahaMahal" is a Django-based website with an SQLite database that provides a platform for users to log in, explore, and review food. This project report outlines the development journey of MahaMahal, detailing its technical architecture, design decisions, user experience considerations,securitymeasures, and scalability strategies.
Project Type	Website
Front End	Django
Back End	Sqlite
Other Tools	Bootstrap
Guide	Prof. Ripal V. Pandya Prof. Milind V. Anandpara
Submitted To	S.S.S.D.I.I.T College

USE OF SYSTEM DEVELOPMENT LIFE CYCLE MODEL

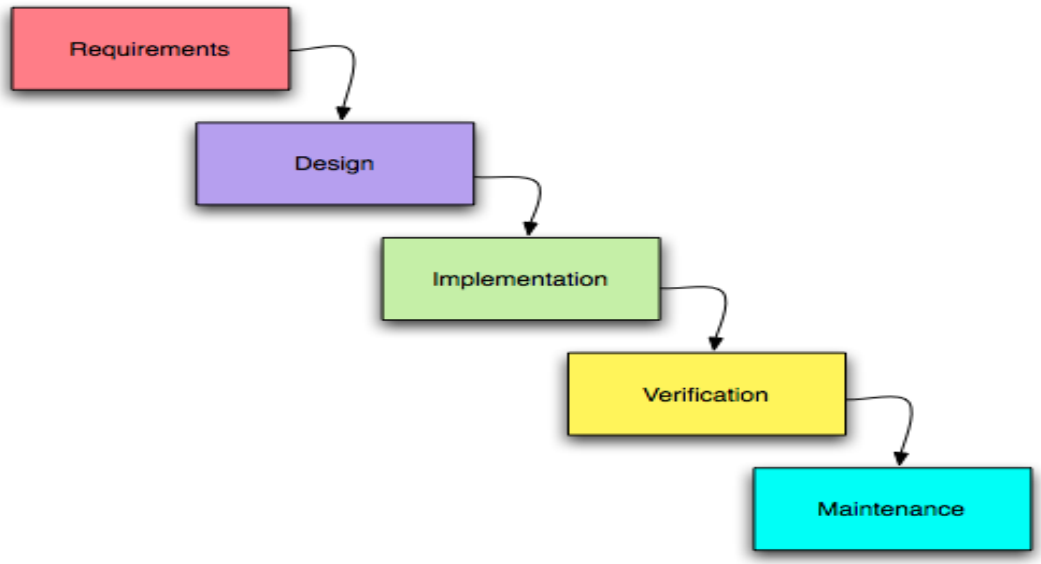
Software Development Life Cycle (SDLC) is a process for development of software. There are some steps to follow to create a software application.

In an SDLC the steps follows requirement gathering. In requirement gathering questionnaire, personal interview etc. are the method for gathering information. Analysis phase includes creating Software Requirement Specification and analyze the gathered data. In design phase, design of Software application i.e. database design and GUI design have to be prepared. In coding phase, coding is done of different modules and forms. In testing phase, the different type of testing is done like integration testing, unit testing, system testing and at last the created software is implemented and maintained.
Following are the different Life Cycle Model example.

- Waterfall model
- Iterative waterfall model
- Prototyping model
- Evolutionary model
- Spiral model
- R.A.D. model (Rapid Application Development)

WATERFALL MODEL

The waterfall model was first process model to be introduced . It is also referred to as a linear-sequential life cycle model . It also very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begins . there is no overlapping in the phases



1. Planning:

The project started with a comprehensive planning phase where we defined the project scope, objectives, and requirements. We conducted market research to understand user needs and competition, which informed our project goals.

2. Analysis:

During this stage, we conducted a detailed analysis of the hotel management processes. We identified key features and functionalities, user

roles, and data storage requirements. This phase laid the foundation for the software's architecture.

3. Design:

In the design phase, we created a detailed blueprint of the software, including database schema, user interfaces, and system architecture. We focused on creating an intuitive and aesthetically pleasing user experience.

4. Development:

The software's core functionality, including user authentication, room booking, and reviews, was implemented in this phase. Frequent code reviews and testing were conducted to ensure code quality.

5. Testing:

Rigorous testing was carried out to identify and fix any bugs or issues. We conducted unit testing, integration testing, and user acceptance testing to ensure that the software met quality standards.

6. Deployment:

After successful testing, the software was deployed to the hotel's servers, and necessary configurations were made to ensure it operated smoothly in a production environment.

7. Maintenance and Support:

Post-deployment, we continued to provide maintenance and support services. This included monitoring for any issues, releasing updates, and addressing user feedback.

FEASIBILITY STUDY

Feasibility of a project determines whether it is possible to develop the project. These are four main factor`s, which determine the feasibility of the project. They are discussed as follow.

The main aim of feasibility study is to determine whether developing the project is functionally and technically feasible or not.

The feasibility study involves analysis of the problem and collection of data which world be input to the system, the processing required to be carried out on these data, the output data required to be product by the system, as well as study of various constraint on the behavior of the system.

An initial determine in a proposal that whether an alternative system is feasible or not. To determine feasibility of candidate system in all respect I need to consider following feasibility factors:

There three types of feasibility study.

1) Technical

2) Operational

3) Economical

Technical feasibility considers whether the desired project can be completed the framework of available technology. As our project is developing the software, this is not much of problem because there are many advanced web editing tools are available.

Technical Feasibility :

The main aim of technical feasibility study is to determine whether it is possible to develop the proposed system with the present technologies available and study the technical requirements and their availability in the organization & the technical equipment availability in market.

So, in this project technical requirements is :-

Hardware:

1) Browser(chrome,opera,brave etc..

2) 8 GB RAM

3) 512 MB Hard Disk

4) Intel i5 Processor

Software:

1) Browser(chrome,opera,brave etc..)

Economical Feasibility :

The economic feasibly takes into consideration the financial matters regarding the proposed system. The organization measures the cost effectiveness of the proposed system. The economical feasibility of the proposed system is as under budget of a company or not! This is checked in economical feasibility.

Total pages : 8	
Per page cost : 1500	=20,000
Total	**= 20,000 /-**

Operational Feasibility :

The Operational feasibility deals with the matter whether the proposed system fulfills the requirements of the organization. This feasibility determines whether the proposed system covers all the aspects of the current system & gives an extra facility which is not in current system.

The project requires one person who has knowledge of basic computer fundamental.

The client has one computer operator who can handle, the software.

the operational feasibility is as follows.

- The proposed system covers all aspects of the working current manual system.
- The human sources required for proposed system.
- Staff is totally operational.
- Easy to manage with organization.

REQUIREMENT GATHERING

Questionnaire :

1) What does your business actually do ?
- Our business is a food website which is used to show the food details.

2) Which software are used to create project?
- This project is create in Django framework and used with vscode .

3) What do you want to create a software or website?
- I want create a website.

4) do you have any other functionality tool?
- Yes, I am used to Bootstrap for the make design attractive.

5) What kind of features do you need in your software?
- I want features like Add data, Update data, Delete data, Reset data.

6) How much time period will give for this site?
- I need complete website in approx 120 days.

7) In room page do you need your office map location?
- No

8) How many module do you want?
- Tow module (Admin,User).

9) how much time period will give for this software?
- 3 month to create this website.

REQUIREMENT ANALYSIS

Requirement gathering phase of software development life cycle acquires information from the organization for which we are preparing project. There are many techniques to acquire information. It's simple meaning to get a user's requirement for website which kind of facility user wants.

Following are the techniques for information gathering.

- Questionnaire
- Observation
- Personal Interview
- Record Review

From the above options, I have used "Observations" and "Personal Interview" method for requirement gathering. I have adopted Questionnaires because I can properly understand their need of software. I can also understand about different rights given to different users and the basic about software. By using Personal Interview I have understood the smallest need of their application and some idea of layout and designing.

The main requirements for the site are listed below:

- Only use for admin .
- Only admin can do product add, update, delete and view.
- Only Admin can delete website.
- Only Admin can perform user Management.
- Administrator can delete user also can change rights of food.
- Admin can get PDF of customer record.
- If the admin has forgotten his password, he can get it from the username.

REQUIREMENT ANALYSIS

Hardware Requirement:

Tools	Required
Processor	Intel i5 Processor
Hard disk	512MB or higher
RAM	8GB or higher

Software Requirement:

Browser(chrome,opera,brave)

Front – End:

Django

Back – End:

Sqlite

PROJECT ABSTRACTS

User Groups:

- **Administrator**

Administrator:

- Add, edit, or delete user , customer , employee from the Menu`s.
- Admin have full control and management of the platform
- Manage user , employee and specifications.

User:

- Update password, menu , logout , deleted from the user.
- User have full control in menu and management.
- Manage visitor specifications.

PROPOSED SYSTEM

Role of the website:

Actual role of this website is to provide the seamless experience of music listening to the User in the way of:

- food & menu Management
- Maintaining user Information

Website has been developed with a Frontend tool as Chrome Version 122.0.6261.112 (A Web Browser) and Visual Studio Code Team Edition for Software Developers (Editor).

Collaborate with Bootstrap and Django Server is on the Local machine providing Local Intranet merging two important components for this Website Development namely as under:

- Django Development Server (A Web Server)
- Sqlite (A Database Module)

Using of above all is acceptable factors for the real implementation of Website. Because in there are much number of advantage and simplicity and also security in the point of view for this Web development task. And in the Real-Life Application of this type of task above mentioned component had proved as better solution in past and present too.

ADVANTAGES & LIMITATIONS OF PROPOSED SYSTEM

ADVANTAGES:

- **User Engagement:** Mahamahal fosters active user engagement by providing a platform for users to share their food experiences, reviews, and recommendations, creating a vibrant community of gastronomic enthusiasts
- **Personalization:** The website offers personalized experiences by allowing users to create accounts, customize their profiles, and receive tailored food recommendations based on their preferences and past reviews.
- **Accessibility**: With its online presence, Mahamahal transcends geographical boundaries, enabling users from diverse locations to access a rich repository of culinary information and reviews.
- **Scalability:** Built on the Django framework, Mahamahal is inherently scalable, capable of accommodating a growing user base and expanding content without compromising performance or user experience.

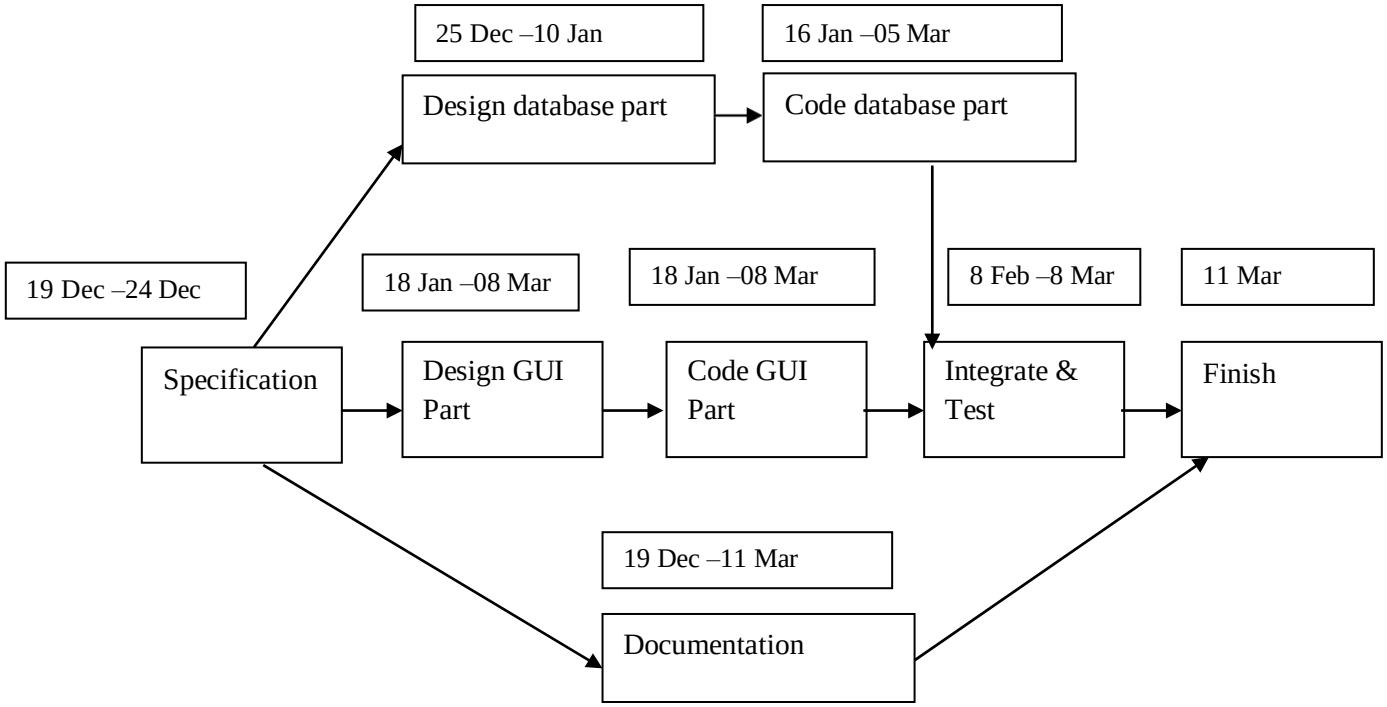
DISADVANTAGES:

- **Content Quality Control:** Maintaining the quality and authenticity of user-generated content, such as reviews and recommendations, can be challenging, as it may be susceptible to biased or spammy submissions.
- **User Privacy Concerns**: Collecting and storing user data, including personal information and preferences, raises privacy concerns, necessitating stringent data protection measures and compliance with privacy regulations.
- **Resource Intensive:** Developing and maintaining a robust online platform like Mahamahal requires significant time, effort, and resources, including web hosting, development tools, and ongoing maintenance**.**

PERT CHART AND GANTT CHART

PERT CHART :

PERT (Project Evaluation and Review Technique) charts consist of a network of boxes and arrows. The boxes represent activities and the arrows represent task dependencies. PERT charts are a more sophisticated form of activity chart. Where instead of making a single estimate for each task, pessimistic, likely and optimistic estimates are made. The boxes of PERT charts are usually annotated with the pessimistic, likely, and optimistic estimates for every task. There are thus not one but many critical paths, depending on the permutations of the estimates for each task. This makes analysis of critical path show by using shaded boxes. The PERT chart representation of the MIS problem of show follows.

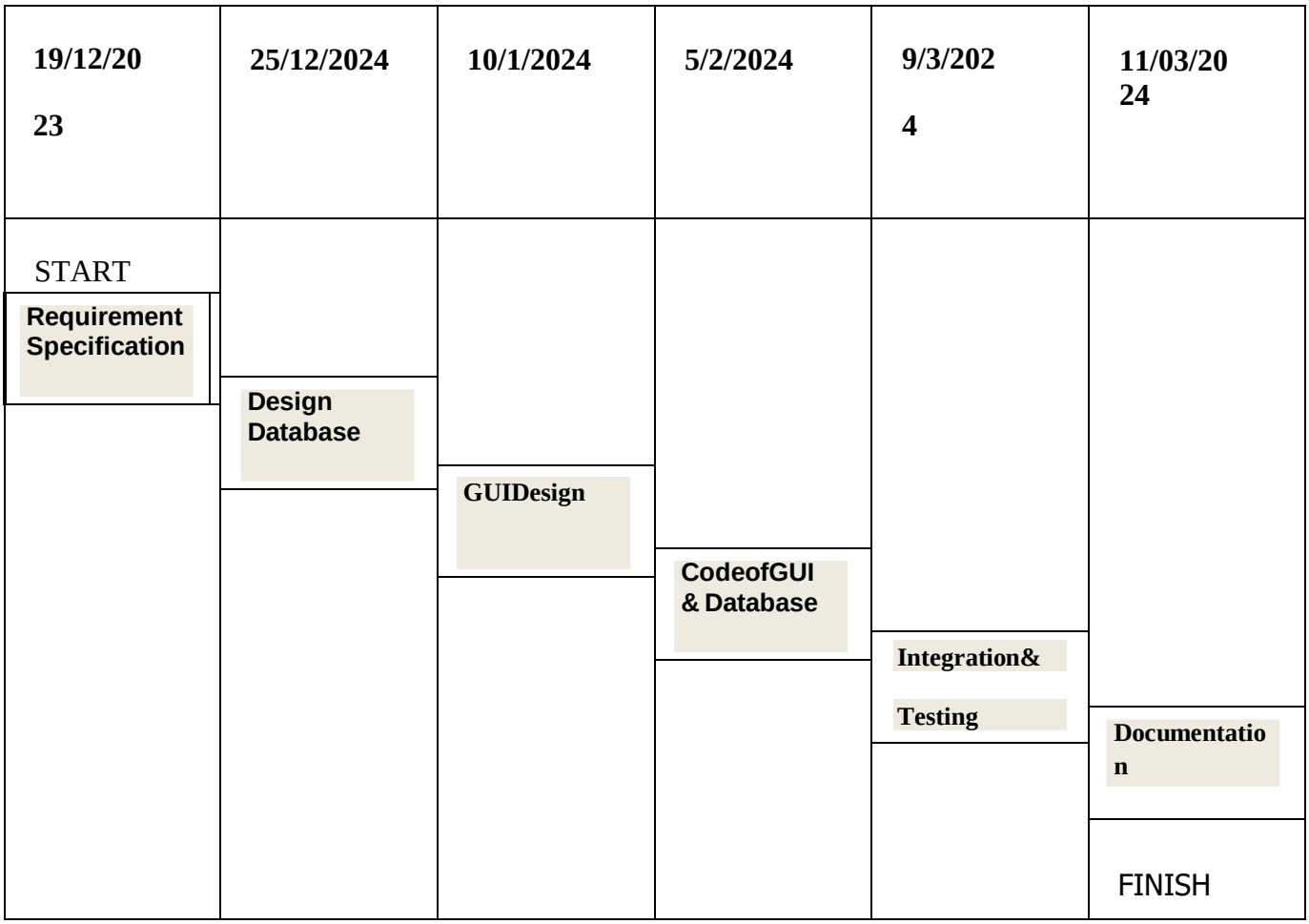


Gantt chart can be derived automatically from PERT charts. However, PERT charts cannot be automatically derived from Gantt charts because PERT charts incorporate additional information about the time when an engineer doses a task. This information is not available is helpful in planning the utilization of resources, while the PERT charts is more useful for monitoring the timely progress of activities. Also, parallel activities in a project can be easily identified using a PERT chart.

GANTT CHART

Gantt charts are mainly use of scheduling, budgeting, and resource planning. It allocates resource to activity include Staff, Hardware, Software, etc…

A Gantt chart is a special type of bar chart where each bar represents an activity. The bars are drawn along a time line. The length of each bar is proportional to the duration of time planned for the corresponding activity.



Gantt charts used in software project management are actually an enhanced version of the software project management. Each bar consists of a white part and a shaded part. The white part of the bar shows the length of time each task is estimated to take. The shaded part of the bar shows the slack time.

In order to estimate the time durations for various activities, usually managers let the engineers themselves estimate the time for an activity they might be assigned to. However, some managers prefer to estimate the time for various activities themselves. Many managers believe that an aggressive schedule motivates the engineers to do a job better and faster.

However, careful aspects, but also cause schedule compromise on intangible quality aspects, but also cause schedule delays. A good way to achieve accuracy without creating problems is to let people set their own schedules.

We can see that one engineer can do the database design and then code the database design whereas another engineer and design the GUI part, code the GUI part, and still have time left for writing the user manual. Thus, Gantt charts are very useful in scheduling resources.

So here, I have to follow the scheduling steps for my project.

Gantt chart is really useful us for planning software application resources.

DATA FLOW DIAGRAM

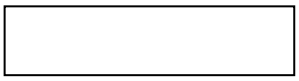
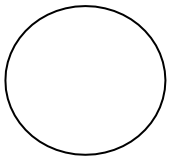
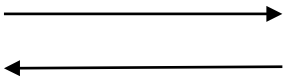
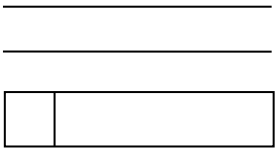
Detailed Life Cycle of Project:

In the discussion of "Detailed Life Cycle of Project" we have to concentrate on DFD (Data Flow Diagram). Here we have work on it while developing this software project.

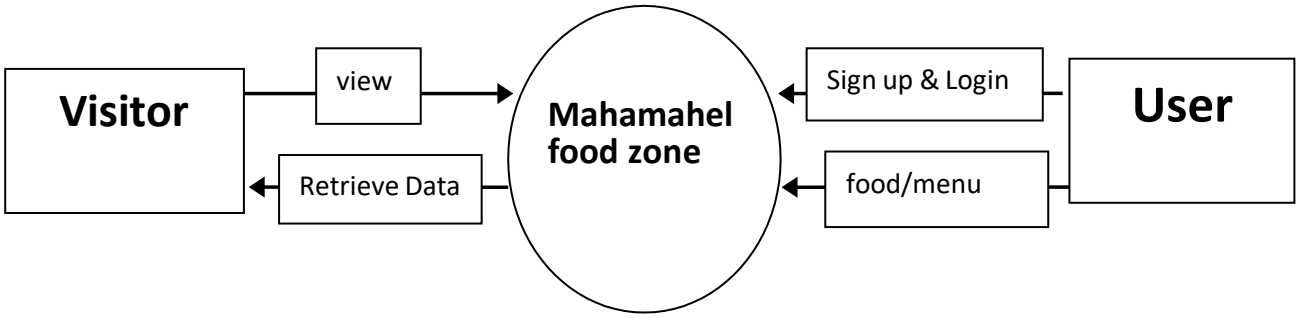
DFD (Data Flow Diagram):

DFD is a graphical view of all system processes and transactions. With the DFD an End-User also can easily understand the system in a short time period. Also it is useful to find out problems or any complications with the system we are going to develop. We can easily get that whether we have understood the system as per the requirements of the customer or not by showing them this diagram. Thus DFD is a necessary phase while developing software.

For Understanding :-

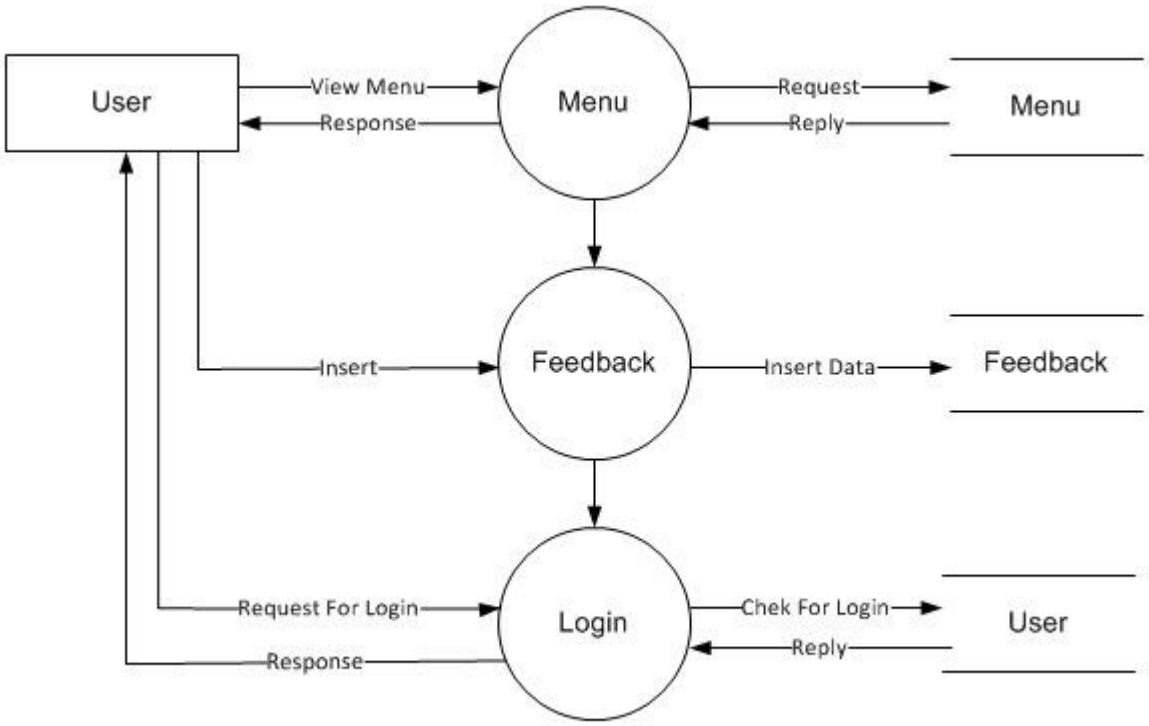
Symbol	Name	Use
	External Entity	Rectangle source and / sink destination data.
	Process / Function	Transformed, Store, or Distribute. Annotated with number and name of function.
	Data Flow	Direction of data flow single piece of data or logical collection of data.
	Data Store	Open Rectangle Parallel lines Data Structure, File, Table, Database.

Context Level Diagram :

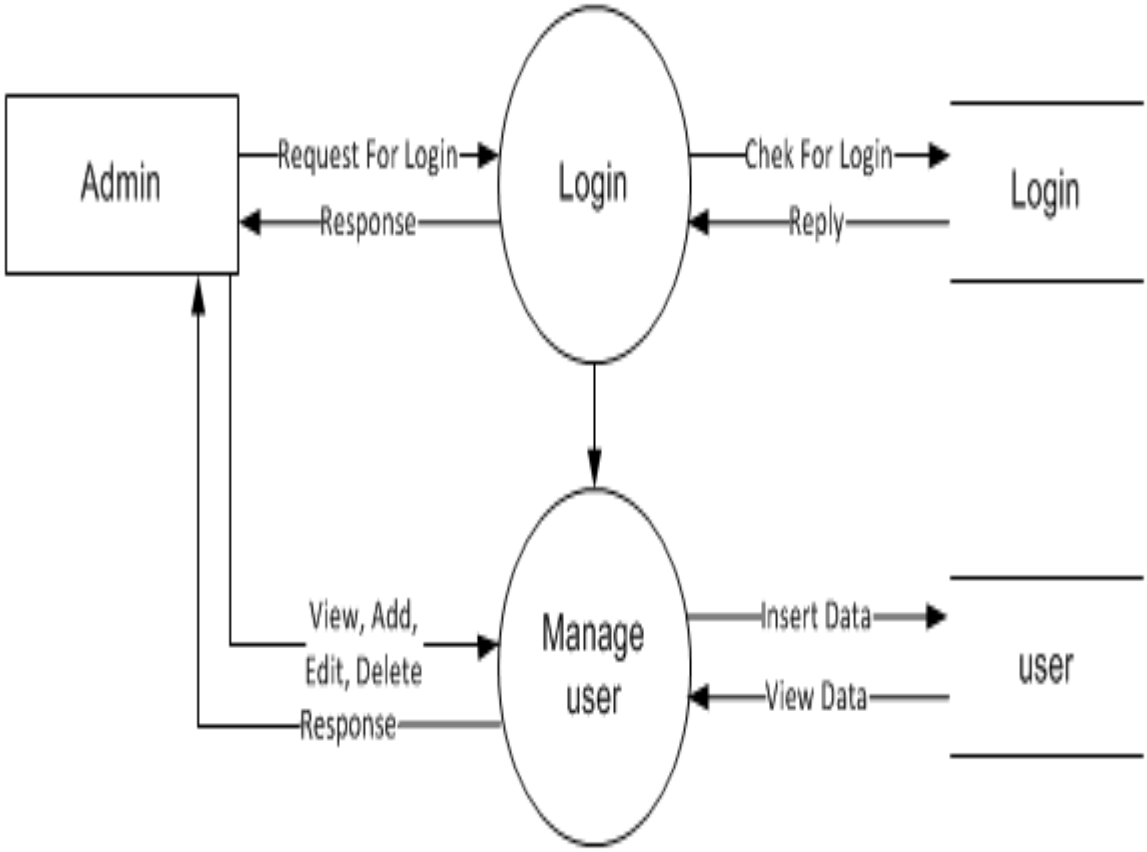


1st Level Diagram :

User 1st Level :

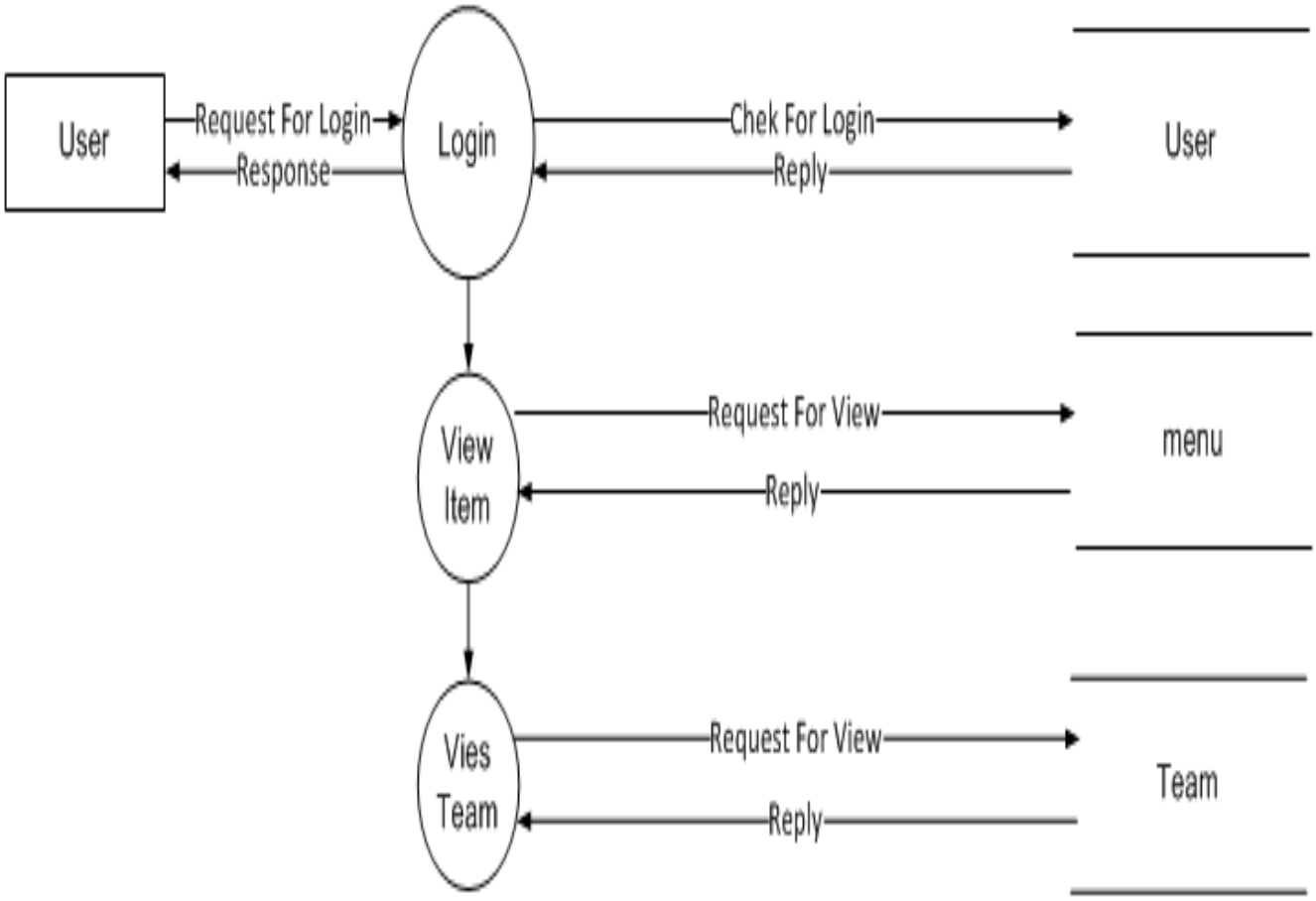


Admin 1st Level :

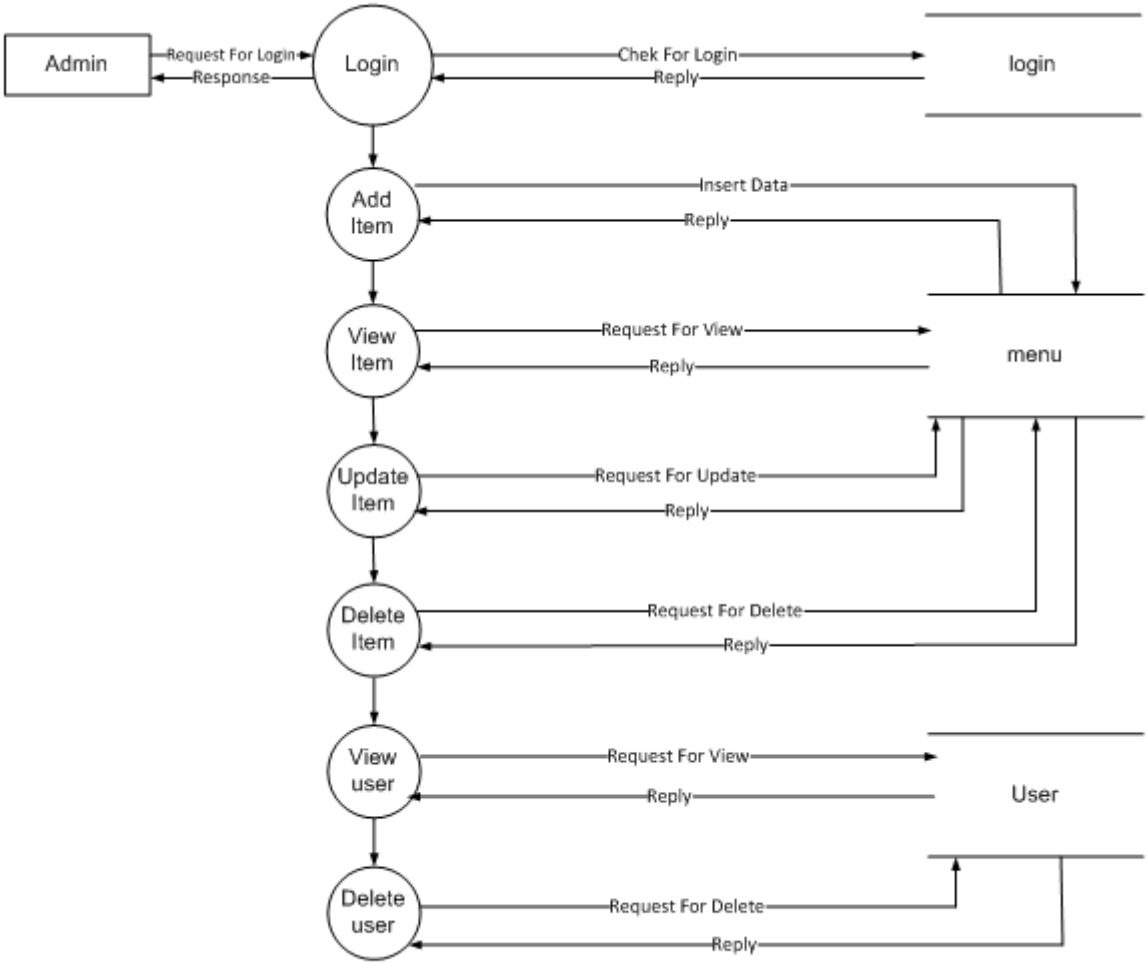


2st level Diagram:

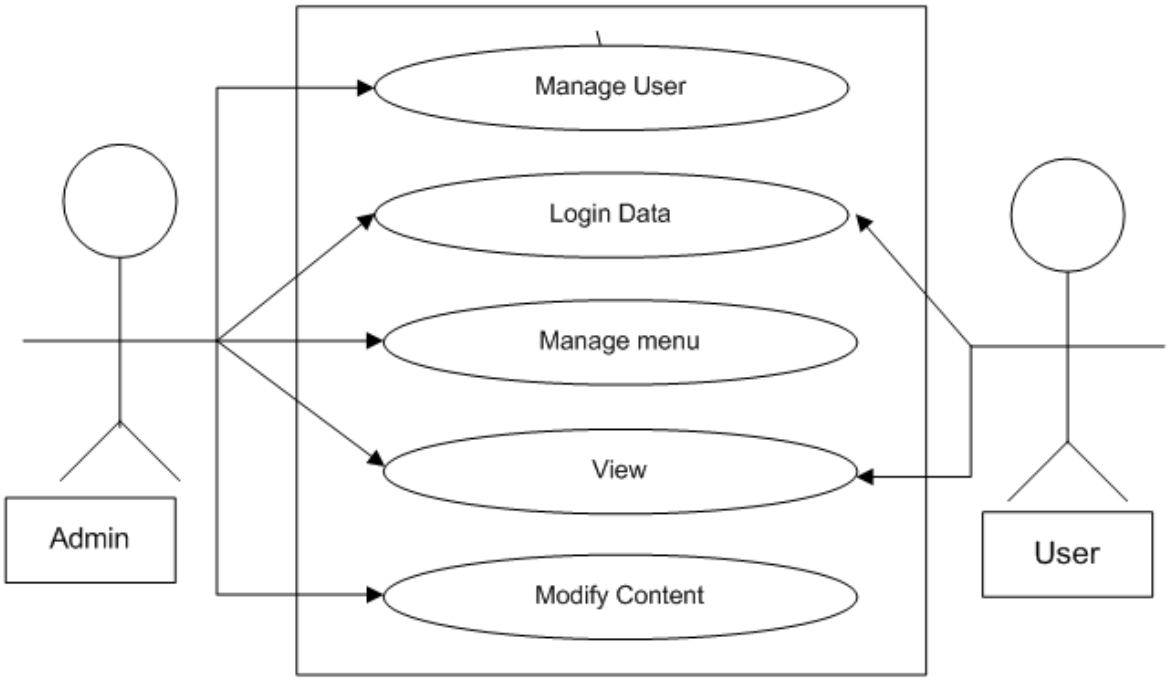
User 2st Level :



Admin 2st Level :



USE CASE DIAGRAM

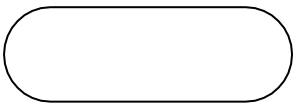


A use case is a set of scenarios that describing a system. A use case diagram the relationship among actors and use cases. The two main components of a use case diagram are use cases and actors.

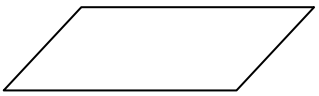
FLOW CHART

A Flow is a pictorial representation of an algorithm. Programmers often use it as a program-planning tool for visually organizing a sequence of steps necessary to solve a problem using computer. It uses boxes of different shapes to denote different type of instructions. The actual instructions are written within these boxes using clear and concise statements. Solid lines having arrow marks connect these boxes to indicate the flow of operation, that is, the exact sequence in which to execute the instructions. The process of drawing a flowchart for an algorithm is known as flow charting.

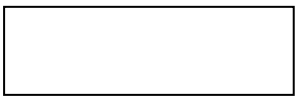
Basic Flowchart Symbols:



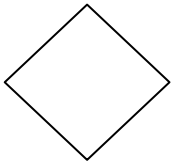
Terminal



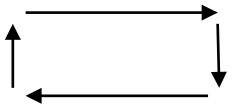
Input / Output



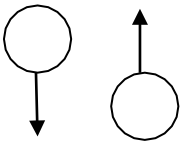
Processing



Decision

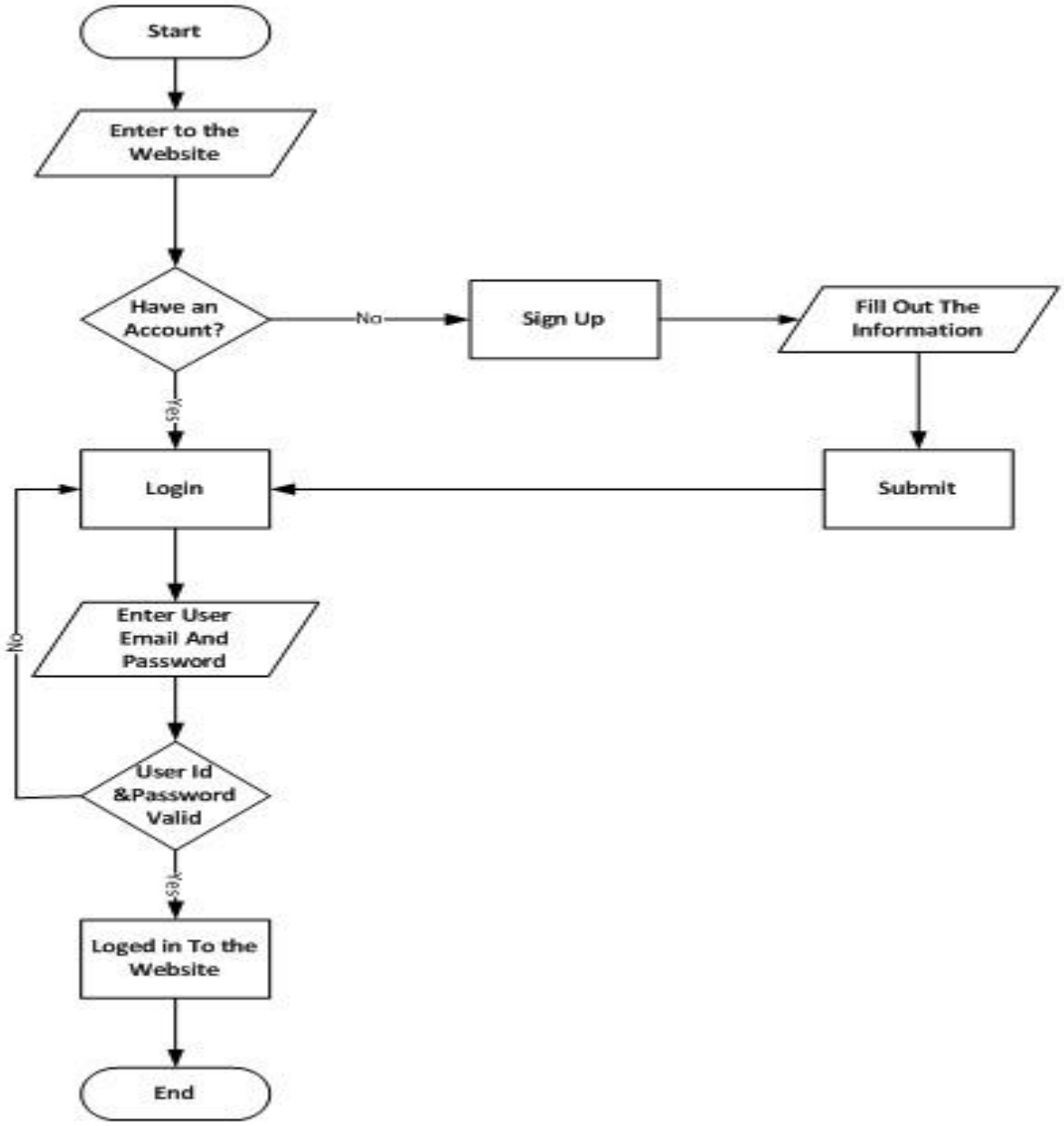


Flow Lines



Connectors

Administrative Login Flowchart:



COST ESTIMATION

Cost estimation for a software project like the "Mahamahal" website involves estimating the expenses associated with development, infrastructure, maintenance, and other aspects. Here's a general outline of the cost estimation process:

In my project, I am implementing a Module Wise to calculate and evaluate the per-page average cost. This approach allows for a comprehensive assessment of cost allocation across various components, resulting in a more accurate and informative cost analysis.

Module Wise:

Total pages : 8	
Per page cost : 1500	=20,000
Total	**= 20,000 /-**

Thus the approximation cost of this Software Project will be about Rs 20,000/-

DATA DICTIONARY & NORMALIZATION

Database Name : my hotel

Table 1: Profile

Column Name	DataType (Size)	Constraints	Remarks
user	int	A.I, P.K	
Profile_pic	Choose file		
Contact_number	Varchar(15)		
address	Text Field		
Updated_on	dateTimeField		

Table 2: Contact

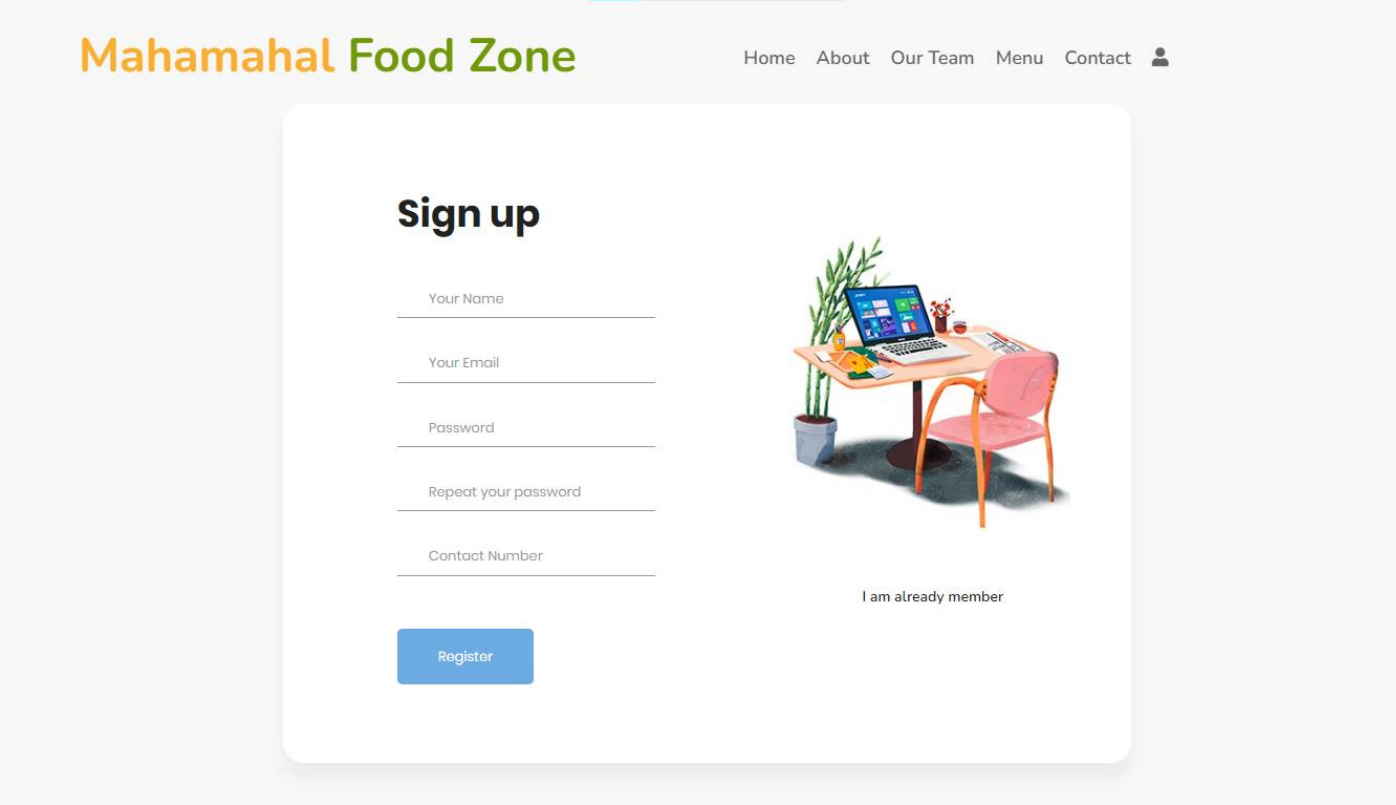
Column Name	DataType (Size)	Constraints	Remarks
name	Varchar(250)	A.I, P.K	
email	Varchar(50)		
subject	Varchar(250)		
message	Varchar(50)		
added_on	dateTimeField		
Is_approved	boolean		

Table 3: User

Column Name	DataType (Size)	Constraints	Remarks
name	int	A.I, P.K	
email	Varchar(250)		
password	Varchar(50)		

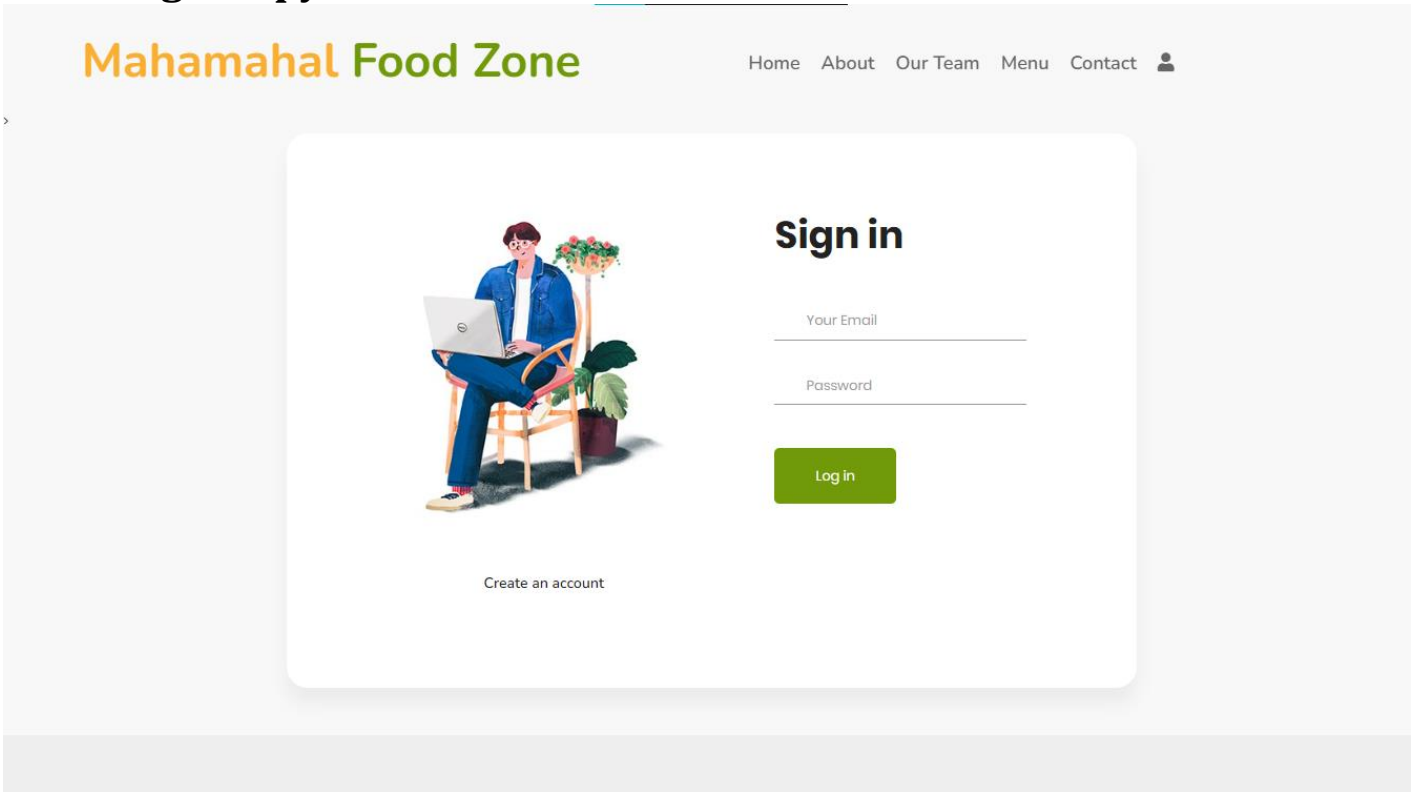
SCREEN LAYOUTS

Sign_up.py



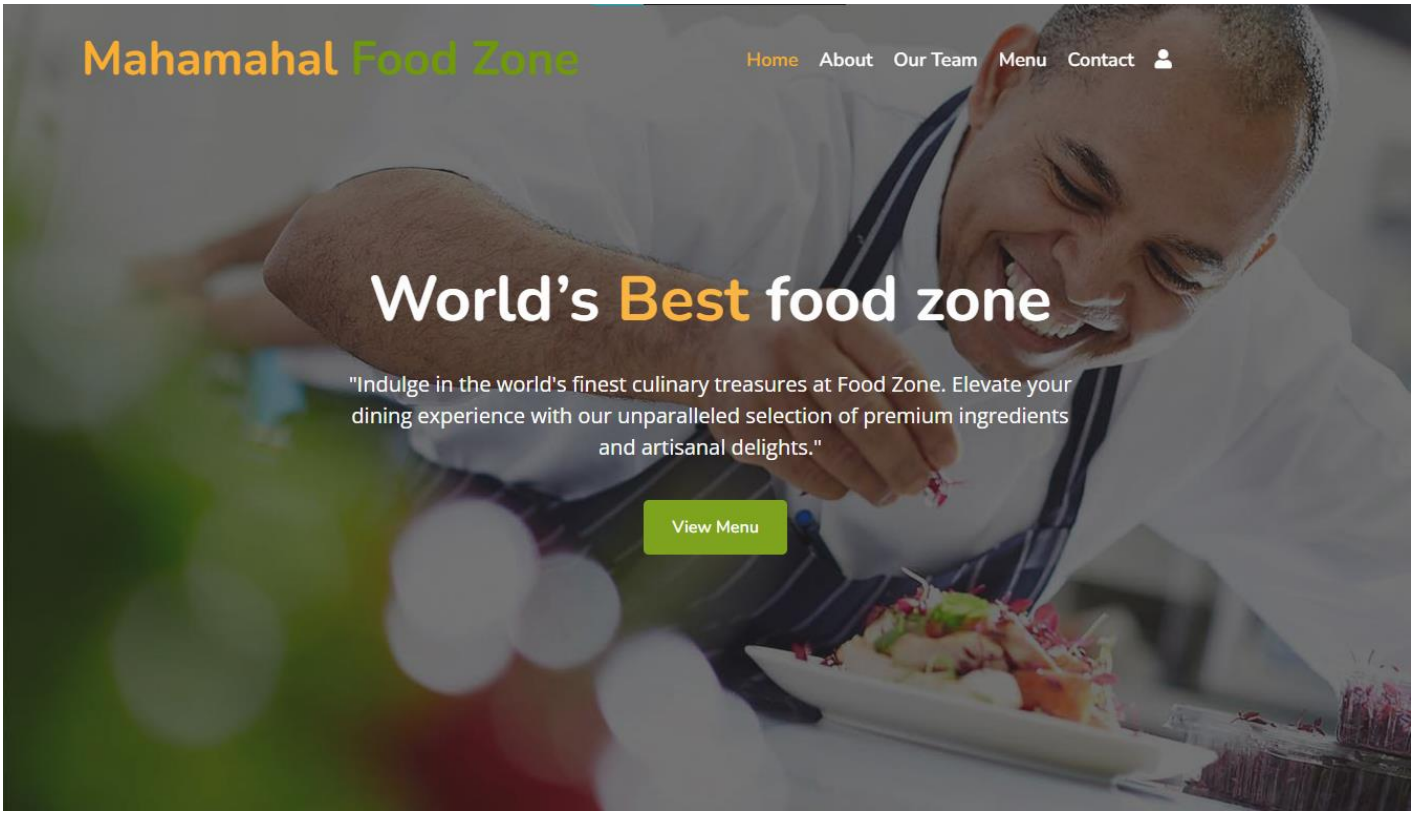

```
def register(request):
   context={}

   if request.method=="POST":
      #fetch data from html form
      name = request.POST.get('name')
      email = request.POST.get('email')
      password = request.POST.get('pass')
      contact = request.POST.get('number')
      check = User.objects.filter(username=email)

      if len(check)==0:
         #Save data to both tables
         usr = User.objects.create_user(email, email, password)
         usr.first_name = name
         usr.save()

         profile = Profile(user=usr, contact_number = contact)
         profile.save()
         context['status'] = f"User {name} Registered Successfully!"
      else:
         context['error'] = f"A User with this email already exists"

   return render(request,'register.html', context)
```

Sign_in.py




```
def signin(request):

   context={}

   if request.method=="POST":
      email = request.POST.get('email')
      passw = request.POST.get('password')

      check_user = authenticate(username=email, password=passw)

      if check_user:

         login(request, check_user)
         if check_user.is_superuser or check_user.is_staff:

            return HttpResponseRedirect('/admin/')


         return HttpResponseRedirect('/dashboard/')
      else:


         context.update({'message':'Invalid Login Details!','class':'alert-danger'})



   return render(request,'login.html', context)
```

Index.py

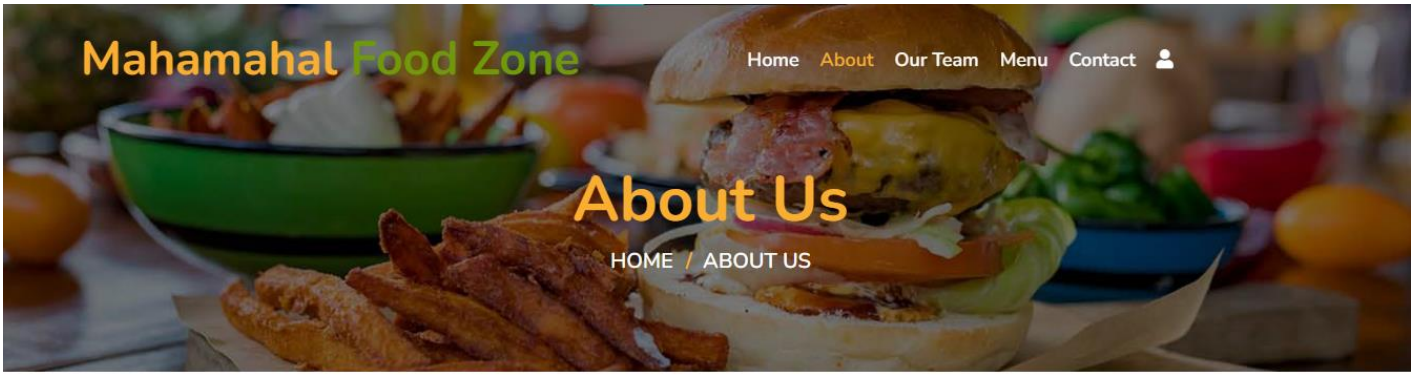



```
path('',views.index,name="index"),
```

```
def index(request):

return render(request,'index.html')
```

```
<div class="carousel-text">

        <h1>The <span>Best</span> Dishes</h1>

        <p>
          We have a proper passion for cooking. Love is the secret ingredient
          that makes all our meals taste better and magical.
        </p>
        <div class="carousel-btn">

          <a class="btn custom-btn" href="{%url 'menu'%}">View Menu</a>
        </div>
      </div>
```

About.py



About Us

Mahamahal Food Zone

Welcome to Mahamahal, your soon-to-be favorite dining destination in Ahmedabad!

flavorful dishes to a warm and inviting ambiance, Mahamahal is all about bringing people together over delicious food.

We can't wait to welcome you to Mahamahal, where good times and great flavors await!

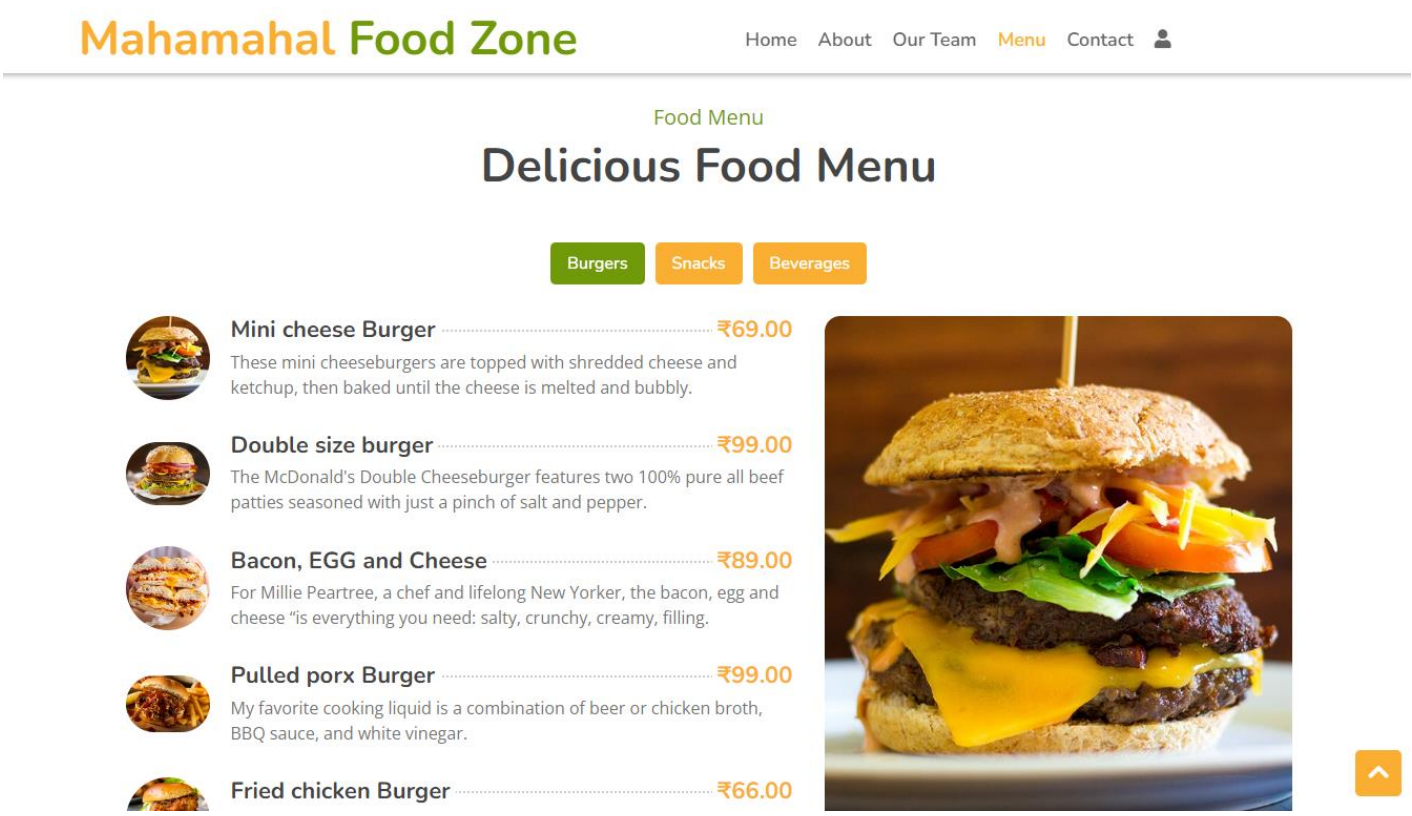
```
path('about/',views.about,name="about"),

def about(request):
   return render(request,'about.html')


<div class="col-lg-6">
      <div class="about-img">
        <img src="/static/img/mahamahal.png" alt="Image" />
      </div>
    </div>

<div class="about-text">
        <p>
         Welcome to Mahamahal, your soon-to-be favorite dining destination
         in Ahmedabad!
         <br />
         <p> flavorful dishes to a warm and inviting ambiance, Mahamahal
         is all about bringing people together over delicious food.</p>
         <br />
          <p> We can't wait to welcome you to Mahamahal, where good times and
         great flavors await!</p>
        </p>
```

Our_Team.py


```
path('team/',views.team_members,name="team"),

def team_member(request):
   return render(request,'team_members.html')
```

```
<div class="team-img">
         <img src="/static/img/team-1.jpg" alt="Image" />
         <div class="team-social">
          <a href=""><i class="fab fa-twitter"></i></a>
          <a href=""><i class="fab fa-facebook-f"></i></a>
          <a href=""><i class="fab fa-linkedin-in"></i></a>
          <a href=""><i class="fab fa-instagram"></i></a>
         </div>
       </div>

<div class="team-text">
         <h2>Adam Phillips</h2>
         <p>CEO, Co Founder</p>
       </div>
```

manu.py

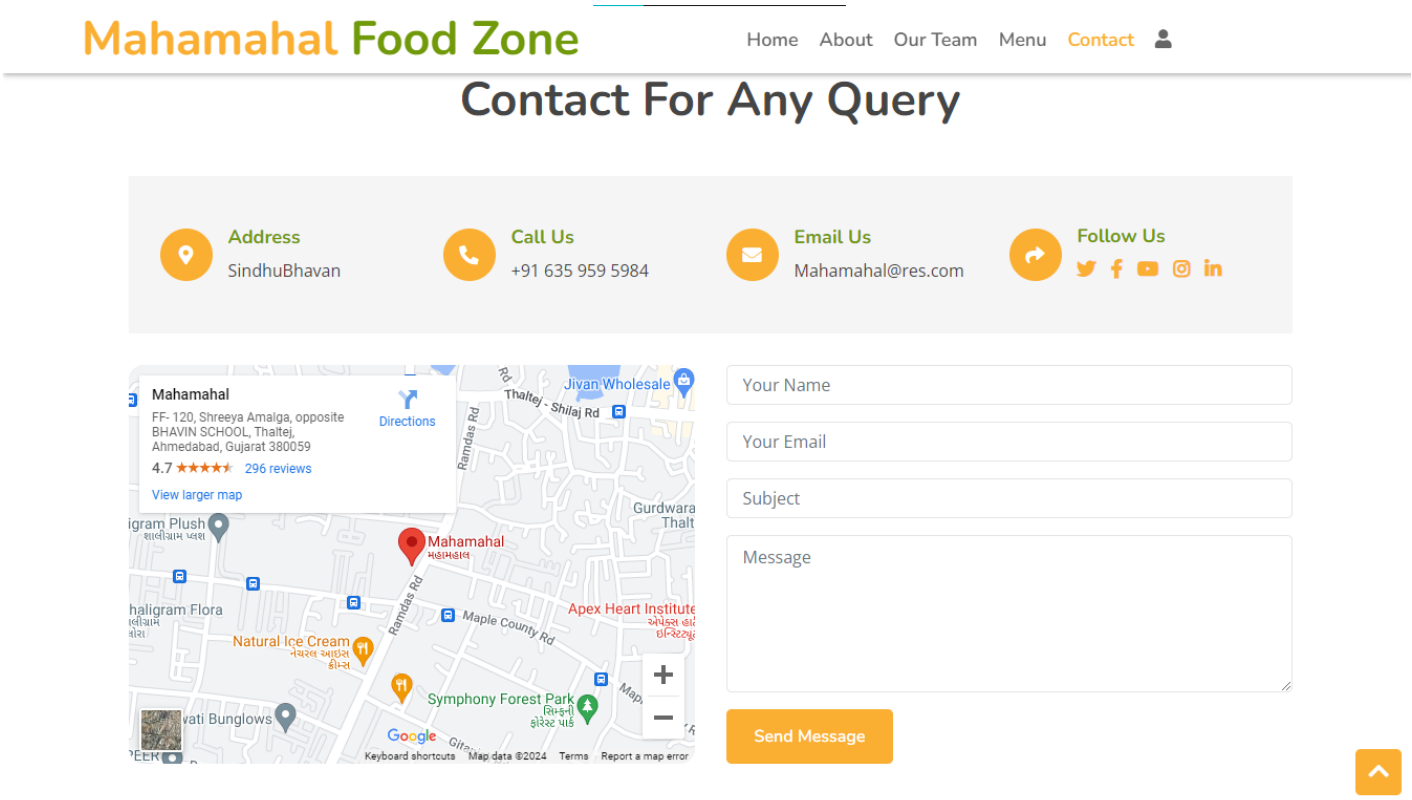


path('menu/',views.menu,name="menu"),

```
def manu(request):
   return render(request,'manu.html')
```

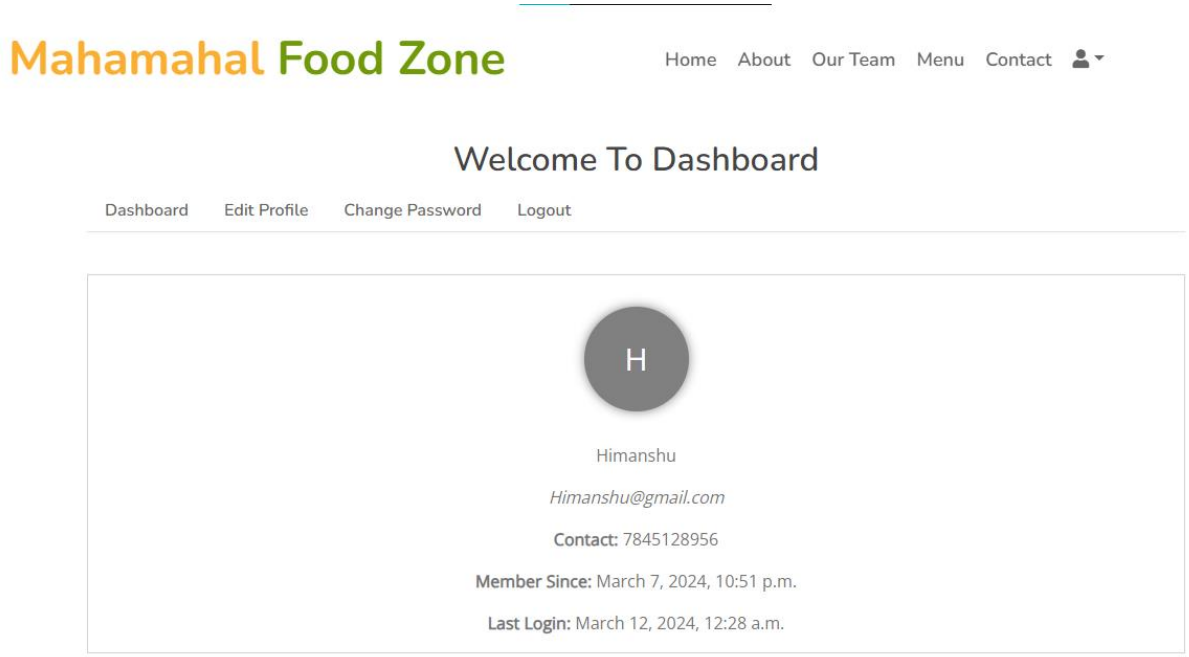
```
<div class="menu-img">
                <img src="/static/img/menu-burger.jpg" alt="Image" />
              </div>
              <div class="menu-text">
                <h3>
                  <span>Mini cheese Burger</span> <strong>₹69.00</strong>
                </h3>
                <p>
                  These mini cheeseburgers are topped with shredded cheese and
                  ketchup, then baked until the cheese is melted and bubbly.
                </p>
              </div>
```

Cantact.py



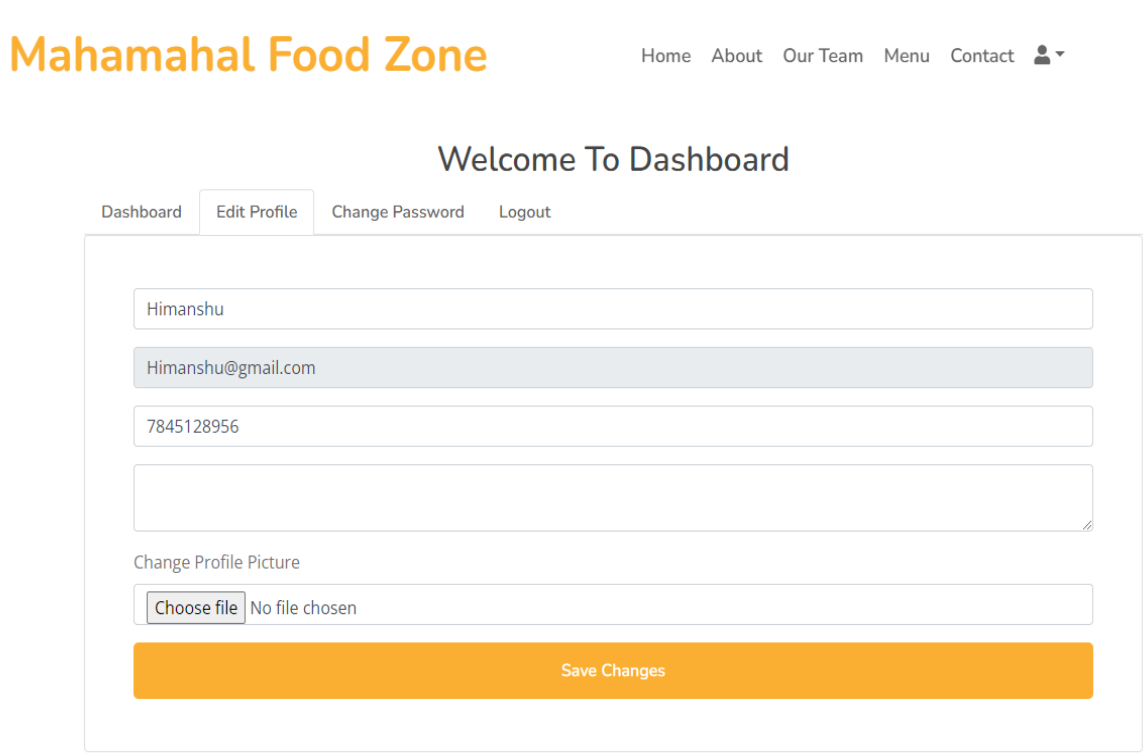

```
def contact_us(request):

    context={}

    if request.method=="POST":
        name = request.POST.get("name")
        em = request.POST.get("email")
        sub = request.POST.get("subject")
        msz = request.POST.get("message")

        obj = Contact(name=name, email=em, subject=sub, message=msz)

        obj.save()

        context['message']=f"Dear {name}, Thanks for Your time!"

    return render(request,'contact.html', context))
```

dashboard.py



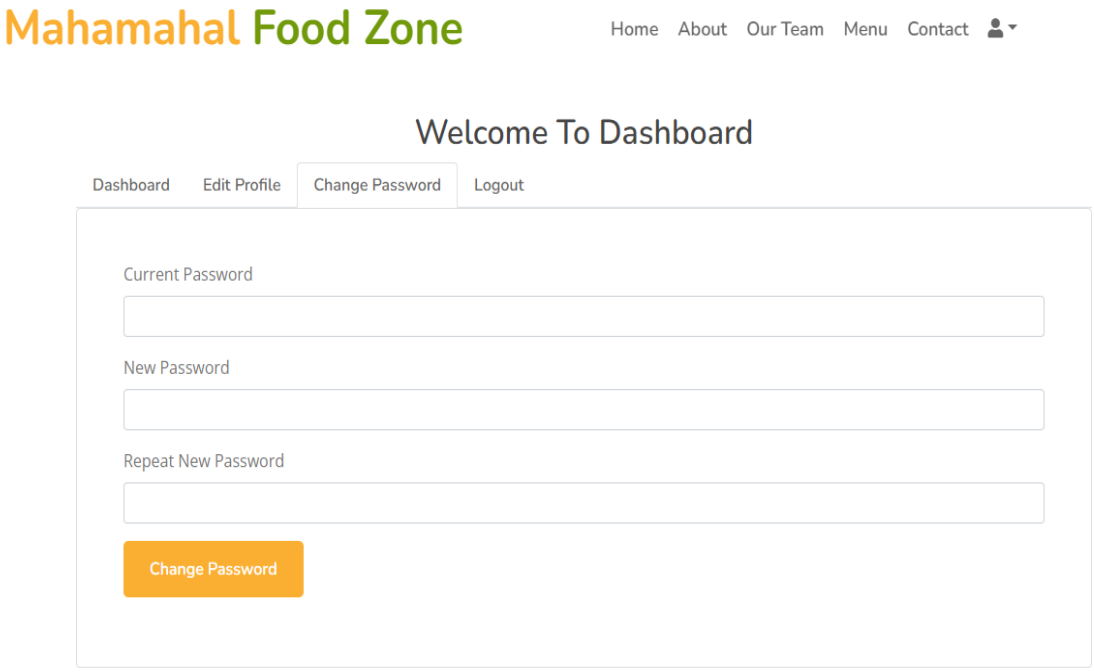

```
<div class="row">
            <div class="col-md-12 text-center dash-content">
              {% if profile.profile_pic %}
              <a href="/media/{{profile.profile_pic}}"
                ><img
                  src="/media/{{profile.profile_pic}}"
                  alt=""
                  class="pic"
                />
              </a>
              {% else %}
              <span class="pic">{{profile.user.first_name.0}}</span>
              {% endif %}

              <p>{{user.first_name}}</p>
              <p><em>{{user.email}}</em></p>
              <p><strong>Contact: </strong>{{profile.contact_number}}</p>
              <p><strong>Member Since: </strong>{{user.date_joined}}</p>
              <p><strong>Last Login: </strong>{{user.last_login}}</p>
            </div>
```

dashboard.py




```
#update profile
    if "update_profile" in request.POST:
        print("file=",request.FILES)
        name = request.POST.get('name')
        contact = request.POST.get('contact_number')
        add = request.POST.get('address')


        profile.user.first_name = name
        profile.user.save()
        profile.contact_number = contact
        profile.address = add

        if "profile_pic" in request.FILES:
            pic = request.FILES['profile_pic']
            profile.profile_pic = pic
        profile.save()
        context['status'] = 'Profile updated successfully!'
```

dashboard.py



Change Password

```
    if "change_pass" in request.POST:
        c_password = request.POST.get('current_password')
        n_password = request.POST.get('new_password')

        check = login_user.check_password(c_password)
        if check==True:
            login_user.set_password(n_password)
            login_user.save()
            login(request, login_user)
            context['status'] = 'Password Updated Successfully!'
        else:
            context['status'] = 'Current Password Incorrect!'
```

SPECIAL UTILITIES

Following are special utilities provided by Inferno Hotel.

.

- **Advanced Search Filters:**Implement advanced search filters allowing users to refine their search results based on criteria such as cuisine type, price range, dietary restrictions, and user ratings.l.

- **Geolocation Integration:** Incorporate geolocation features to enable users to discover nearby restaurants, food trucks, or specialty stores based on their current location.

- **Social Media Integration:** Integrate social media sharing and login functionalities, allowing users to easily share their reviews, recommendations, and dining experiences on platforms like Facebook, Twitter, and Instagram.

- **Bookmarking and Favorites:** Enable users to bookmark or favorite restaurants, dishes, or reviews for easy access and reference later, facilitating personalized content discovery and saving favorite dining spots.

TESTING

Software Development Life Cycle (SDLC) includes a series of production activities one of this is testing

Testing is a process of executing a program with the intent of finding an error.

Testing is the most important element to be considered for providing quality software and it represents the ultimate review of specification, design and coding.

The success or failure of the software as a system mainly depends on testing. Software Developer spends 40% to 50% of their total development time on testing.

Testing is program consists of providing the program with a set of test inputs and observing if the programs behave as expected. Under which a failure occurs are noted for debugging and correction. The following are some commonly used terms associated with testing.

A failure is manifestation of an error. But, the mere presence of an error may not necessarily lead to a failure.

A fault is an incorrect intermediate state that may have been entered during program execution. A fault may or may not lead to a failure.

A test suite is the set of all test cases with which a given software product is to be tested.

Many types of testing techniques are describes as follows.

Unit Testing:

Unit testing is under taken when a module has been coded and successfully reviewed in this section we first discuss the environment needed to perform unit testing.

Here in this project we test each and every module and forms of software application individually when it is completely coded.

There are some methods for unit testing are as follows.

Black-Box Testing:

Black Box Testing to the admin side all queries is regularly implemented and data store

This type of the all functionally is a very higher and unique data.

- Equivalence Class Partitioning
- Boundary Value Analysis

White-Box Testing:

White Box Testing to the admin error. Data Flow - Based Testing.

- Branch Coverage
- Condition Coverage
- Path Coverage
- Linearly independent Path
- Data Flow - Based Testing
- Mutation testing
- Statement coverage

Integration Testing:

The primary objectives of the integration testing is to test the module interface in order to ensure that there are no error in parameter passing when one module invokes another module.

During integration testing different module of system as per integration plane the integration plan specify the steps and the order in which module are combine to realize the full system.

After each integration test the practical integrated system is tested

Following are the integration testing Methods & Approaches:

- Big bang approach
- Top down approach
- Bottom up approach
- Mixed approach

System Testing:

In the system testing the whole application is tested and the error and failure possibility is carried out in it.

Following are the method & approach of system testing.

- Alpha testing
- Beta testing
- Acceptance testing
- Performance testing
- Error seeding

Testing is a process of executing a program with the intent of finding an error.

A good test case is one that as a high probability of finding an as yet in discovered error.

A successful test is one that uncovers a yet undiscovered error.

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. the increasing visibility of software as a system element and the attendant "Cost" associated with a software failure are motivating forces for well – planned, thorough testing. It is not unusual for a software development organization to expend between 30 to 40 percent of total project effort on testing. In the extreme, testing of human-rated software can cost three to five times as much as all other software engineering activities combined.

There are several testing techniques but we have been focused to White box testing techniques. As well as this software is concerns we have test all required testing of this site

TEST CASE

VALIDATION CHECKS TEST CASES:

A Test case is a set of conditions or variable under which a tester will determine whether a system satisfy requirement or works correctly.

The process of developing test case can also help to find the problem in requirement or design of an application.

A test case can have the following elements, not however that normally a test management tool is used by companies by the tool used.

Test Suit ID: The ID of the test suit, to which this test case belongs.

Test Case ID: The ID of the test case.

Expected Result: The expected result of the test.

Actual Result: The actual result of the test, to filled after executing the test case.

Created By: The name of the author of the test case.

Executed By: The name of the person who executing the test cases.

Pass/Fail: The result in "Pass" or "Fail" according to the test, when the expected result and the actual result is same then the result is "pass" else result is "fail".

Remark: Any comment on the test case or test execution

I create following some test cases which is executed by me.

Test Suite ID: 1Test Case ID: 1 Created By: Vivek Thummar Executed By: Himanshu PaghadarExecuted Date: 06 March 2024			Description: This test case will check the validation functionality on the login form.		
Task no	Task	Expected Result	Actual Result	Pass / Fail	Remark
1	Enter valid username and Password	Login Successfully	Login Success	Pass	
2	Enter Invalid username password	Login failed, ErrorOccurred	Generating an error message	Pass	
3	Click on login buttonwithout providing any value	An error message will generate	Generating an error message	Pass	

Test Suite ID: 1Test Case ID: 1 Created By: Himanshu Paghadar Executed By: Vivek ThummarExecuted Date: 07 March 2024			Description: This test case will check the validationfunctionality on the signup form.		
Task no.	Task.	Task no.	Actual Result	Pass/ Fail	Remark
1	Enter valid username and password	1	Allow to username & password	Pass	
2	Enter Invalid username andpassword	2	Generating an error message	Pass	
3	Click on Signup button without providing any Value	3	Generating an error message	Pass	

IMPLEMENTATION

Implementation refers to the entire effort associated with a new system. The implementation of a web application involves longer term issues after the system has been designed and installed. Implementation is a part of the design of a web application, and is an organizational change process. It is a part of the process that begins with the very first idea for a web application has been successfully integrated with the operations of the organization. We expect most of the implementation to be concerned with behavioral phenomena since people are expected to change their information processing activities.

The implementation is processed from review and reports from developer cover the following areas:

- Good working conditions.
- Useful for gathering information.
- Update website easily.
- Attractive layouts.
- Working for as per requirements.

BIBLIOGRAPHY

Websites used:

- www.DJENGO.COM
- www.google.com
- www.wikipedia.org
- https://chat.openai.com