30MINRECIPES

A Project Report

Submitted in partial fulfillment of the

Requirements for the award of the Degree of

BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)

By

Weqar Fatima Sayed Abedi

UID / Roll No. : 18BIT046

Under the esteemed guidance of

Ms. Bertilla Fernandes

Assistant Professor

DEPARTMENT OF INFORMATION TECHNOLOGY

JAI HIND COLLEGE

(Autonomous)

MUMBAI, 400020

MAHARASHTRA 2020-21

JAI HIND COLLEGE

(Autonomous)

MUMBAI-MAHARASHTRA-400020

DEPARTMENT OF INFORMATION TECHNOLOGY

CERTIFICATE

This is to certify that the project entitled,”**30MinRecipes**”, is bonafied work of **WEQAR FATIMA SAYED ABEDI** bearing UID / Roll No. **: (18BIT046)** submitted in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATION TECHNOLOGY from Jai Hind College Autonomous (University of Mumbai).

Internal Guide **Coordinator**

External Examiner

Date: **College Seal**

ABSTRACT

A platform made for those users who want suggestions to home cook simple meals in less than 30 minutes with available ingredients. The main objective of this website is to create a platform where people can get ideas to prepare healthy meals in very less time. If they choose the option to select meals for the whole week, it would help them in preparing grocery shopping list in advance if they shop/order on weekends. This would cater to those students or executives who move to different places away from their hometown miss out on home food and end up eating take away meals or struggle to get recipes People will be able to share their recipes. People could be able to share their many things that would enable one to learn from the one and vice versa.

This project is built for students or executives to have a platform to cook easy and healthy quick meals and share their thoughts. Users will be able to add the recipes to favorite, create a recipe menu plan for an entire week, create a recipe plan for 2 or 3 days, add ingredients to their own pantry, and users can also comment on the recipes. Also, there will be a subscription service provided to the users. There would be sections where recipes will be divided into categories according to their likes. This project also aims to raise interest in healthy meals to develop interest in cooking.

ACKNOWLEDGEMENT

I would like to express my thanks to the people who have helped me most throughout my project. I am grateful to my Prof. Ms. Bertilla Fernandes for nonstop support for the project. I can’t say thank you enough for her tremendous support and help.

I owe my deep gratitude to our HOD of Information Technology Department Mr. Wilson Rao who took keen interest on our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system.

At last but not the least I want to thank all of my friends who helped/treasured me out in completing the project, where they all exchanged their own interesting ideas, thoughts and made this possible to complete my project with all accurate information. I wish to thank my parents for their personal support or attention who inspired/encouraged me to go my own way.

DECLARATION

I hereby declare that the project entitled**, ”30MinRecipes”** done at **Jai Hind College,** has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirements for the award of degree of **BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)** to be submitted as final semester project as part of our curriculum.

Name and Signature of the Student

TABLE OF CONTENTS

Contents

List of Figures

List of Tables

1 INTRODUCTION

1.1 Background

This project provides easy quick recipes along with the available ingredients the user wishes to use during cooking. In this website the user can select the various available recipes provided in the different modules given. This website is to get suggestions, methods to cook easy home cooked meals in less than 30 minutes. Target audience is working professionals, students, especially those who have moved away from their home towns and have very less time and knowledge of cooking. They end up with take away meals which are neither healthy nor cheap. This will be the attraction page for all the users which will pop up new and easy recipes according to their needs and will also help them create a menu plan for the week in a few easy steps! The main objective of this website is to create a platform where people can get ideas to prepare healthy meals in very less time. If they choose the option to select meals for the whole week, it would help them in preparing grocery shopping list in advance if they shop/order on weekends.

1.2 Objectives

30MinRecipes website provides us with the healthy food recipes. A full stack application made for those users who want suggestions for quick, simple and healthy meals. This 3-tier application is aimed to be popular with students or executives away from their home who would like to cook. We will provide a list of ingredients and allow the users to export to their shopping list. There will also be some personalization features like saving recipes as favorites.

- Registration and Login.
- Admin adding recipes categorically.
- Quick search bar.
- Recipe favorites.
- Creating a weekly menu as per the user’s choice.
- Creating menu for any one of the day in the week.
- Creating a pantry of ingredients if the user wishes to have them on hand to look upon which would help them in preparing grocery shopping list in advance if they shop/order on weekends.
- A newsletter subscription.
- Quick recipe options.
- Feedbacks from user’s.
- Learn about meal types.

1.3 Purpose, Scope, and Applicability

1.3.1 Purpose

The main purpose of this website is to create a platform where people can get ideas to prepare healthy meals in very less time. They choose the option to select meals for the entire week; it would help them in preparing grocery shopping list in advance if they shop/order on weekends, save the recipes for future needs. If user wants to learn how to cook as a starter or even those who are professionals can use this website to cater their needs.

1.3.2 Scope

This project has admin section and user functionality which plays a major role and a challenge in coding, it will be responsible for adding the recipes and categorizing them in each section appropriately with high resolution images, the backend server side API, json code along with php code to connect to the database retrieve the data according to the unique id, unique recipe id, if the user wants to save, each user will have a unique user id. The creation of the entire menu for the week and can also remove it as per their choice, creating a pantry of ingredients which the user can select specifically and add it or remove it later, method to make the recipe is given in an easy format, comments can be added.

1.3.3 Applicability

This project is beneficial for all those people who have no basic idea of cooking to those who are chefs or cooking professionals. Because my platform doesn’t require any knowledge, anyone can use it. This platform will create an interest to the executives or students who live far away from their home and want to prepare quick recipes which are healthy. The system will automatically come to know what the user’s interest is into. If the respective user has registered then all their activity will be stored in their history will which be easier for them to access later. This website will be an interesting socializing platform where users can read views by others and comments.

1.4 Achievements

The project has successfully met its objectives. The login system is working. Users can access the various functionalities such as saving the recipe, creating a menu, and adding ingredients to pantry and also add comments on each recipe. The admin can successfully add recipes and also view and delete them. The users can also mail the admin as well as subscribe. The users can also search the recipes and access them individually also read about each meal types.

1.5 Organization of reports

The website name is 30MinRecipes, this project is built for students or executives to have a platform to cook easy, healthy quick meals and share their thoughts.

The following chapters of this project consist of the technologies used, Requirement analysis, Designing, Implementation and testing phases of the project.

The first phase consists of the technologies that are required to build the project. Hence a survey was carried out and this project includes both client side and server side and a middleware. I am using html, css, JavaScript, jquery, bootstrap for my frontend and Apache PHP database technology for backend, Restful API JSON which will be the intermediate technology between the frontend and the server. The next phase is requirement analysis which defines the requirements specification of the project which includes both software and hardware requirements. It includes planning, scheduling, product description. The next phase is the Design phase where each module, align with its functionality is explained in detail. After this phase is the important phase called the implementation phase. The implementation phase is time consuming and complex to think of. It includes a design and how to program the functionalities or the modules mentioned above. The testing phase includes testing of the functionalities and implementation in every way. After all these phases are completed, the project is finally implemented, which is checked and concluded to have met its objectives successfully.

Users will be able to add the recipes to their favorite, create a recipe menu plan for an entire week, create a recipe plan for 2 or 3 days, add ingredients to their own pantry, and users can also comment on the recipes. Also, there will be a subscription service provided to the users, where once subscribed, users will receive emails on the newly added recipes on the tip of their fingers. There would be sections where recipes will be divided into categories according to their likes. This project also aims to raise interest in healthy meals to develop interest in cooking.

2 SURVEY OF TECHNOLOGIES

Hardware

- **Laptop/PC**

  The user can access this website through either a laptop or a PC.
- **Mobile Devices**

  This website can be viewed in any handheld mobile phone through the link of the website given.
- **Processor**

  Intel Processor i5
- **Memory**

  RAM - 4GB or More – ROM - 40GB or More

Software

- **Frontend**:
  - HTML: To create an attractive and dynamic website that is responsive.
  - JavaScript: To implement complex features on my website, animations, interactivity. Version used is 1.8.5.
  - AJAX: This will be used with json and Jquery for interacting with the server side. JSON data will be sent to the server using jquery and Ajax. Fetching he data from the database.
  - JQuery: This library will be used for accomplishing multiple lines of code by wrapping them into methods. Version used is 3.5.1
  - Responsive CSS using Bootstrap

- **Middleware:**
  - Restful API, JSON: The middleman which is responsible for interacting for data transfer and sending, receiving the request to and from the web server and the browser. This will be used with AJAX for retrieving the data. Version used is 1.0 since all json api will be backward compatible.

- **Backend**:
  - PHP on Apache: The backend server side programming that will be use is PHP which will serve files by fetching the data and sending it down to the html page via the middleware json along with an Ajax request. Xampp version used is 3.2.4.

- **Database**:
  - MySQL: Used to add, store, access, and process the data from the database. Queries will be fired from simple to complex functionality. The creation of tables, unique ID’s, etc.

3 REQUIREMENTS AND ANALYSIS

3.1 Problem Definition

3.1.1 Existing System

Currently on the internet there are several services or systems that provide free recipes. The problem with these systems is that they are scattered all over the internet. For a user to avail all these services, the user has to look for all these various websites to gain access to every other feature that every other site offers. Many times the users save recipes on social media apps which is not just a website for food, many other activities include in it, It might be the case that the users does not want to involve in the social media world.

3.1.2 Proposed System

This 3-tier application is aimed to be popular with students or executives away from their home who would like to cook. The admin section and user functionality plays a major role and a challenge in coding, it will be responsible for adding the recipes and categorizing them in each section appropriately with high resolution images, the backend server side API, json code along with php code to connect to the database retrieve the data according to the unique id, unique recipe id, if the user wants to save, each user will have a unique user id. Dynamic rendering of pages with the control on one page connecting to the database and displays the content as required by the user. The creation of the entire menu for the week and can also remove it as per their choice, creating a pantry of ingredients which the user can select specifically and add it or remove it later, method to make the recipe is given in an easy format, comments can be added. There is a GUI which throws many recipes for the user in advance, and if the user wants to Login he/she can or there’s an option to join the family of 30MinRecipes.

3.2 Requirement Specification

3.2.1 Functional Requirements

- Data collection from the search box in a separate table in the database to enhance user experience using analytics.
- It will display the recipes to the user via a query or a search box.
- The dynamic recipe page will be providing the recipe details in a very clear and specific format.
- Comments will help the user get to know more about how the recipe is.
- The user can add any number of ingredients to a list and create their own pantry, which can be editable.
- Subscription and joining the 30MinRecipes family.
- A major complex functionality is the creation of menu, which is pre-saving of recipes based on the selected day the user wants. Weekly recipes for any occasion, or choose from the recipes which are created for them.

3.2.2 Non-Functional Requirements

- **Software Quality Attributes**
  - Availability
    - This website will be on server side and is made available at all times so that the users don’t have to face any problem while using 30MinRecipes.
  - Correctness
    - Accurate services will be provided according to the user requirement. An admin login is provided who will maintain the website.
  - Maintainability
    - The system should keep track of all the user activities.
  - Usability
    - The system should satisfy a maximum number of users needs.
  - Cost
    - There is no installation cost since free and open source technologies are used. The only cost that might incur is the hardware cost such as server.

3.3 Planning and Scheduling

PERT CHART

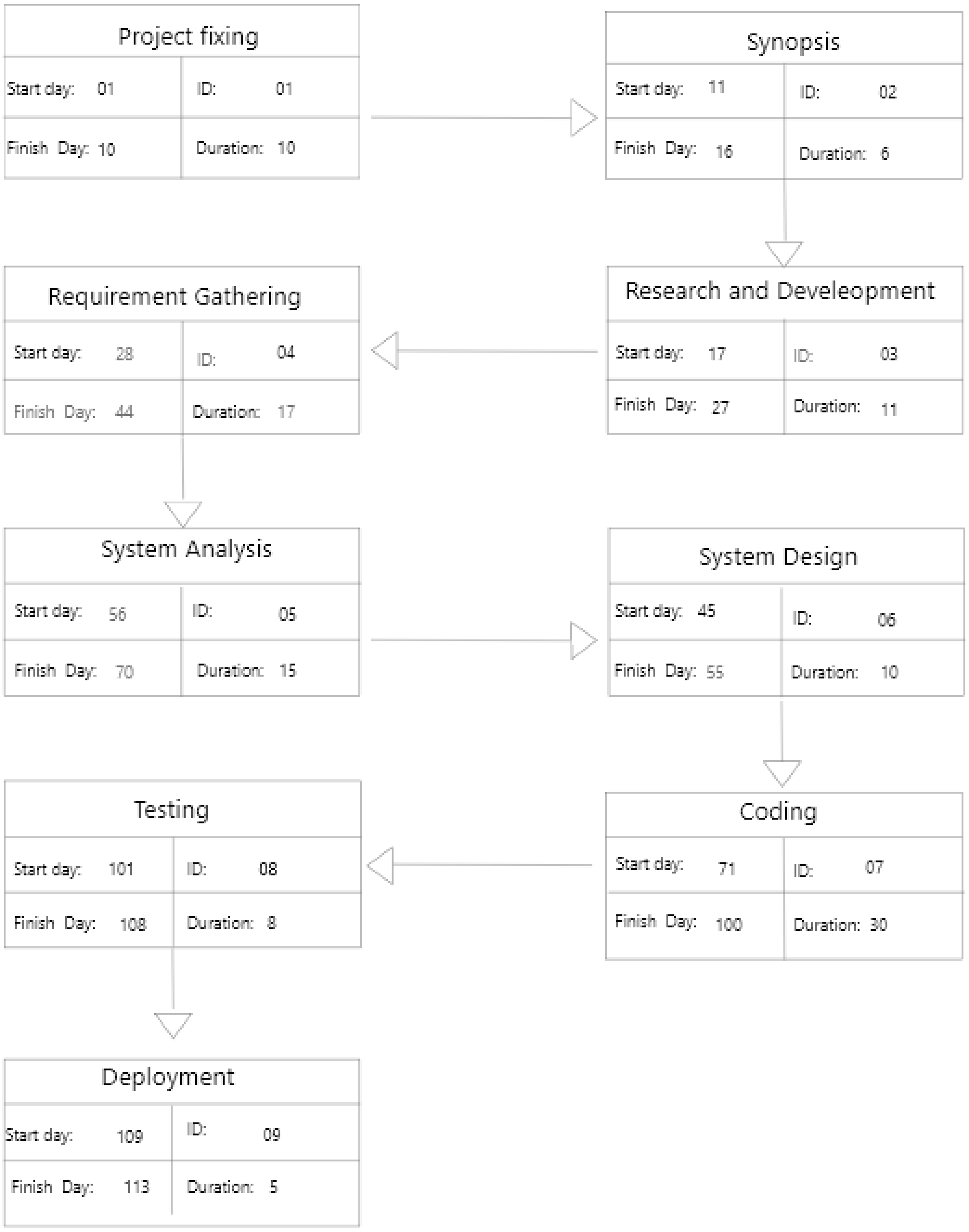


Figure 1: Pert Chart

GANTT CHART

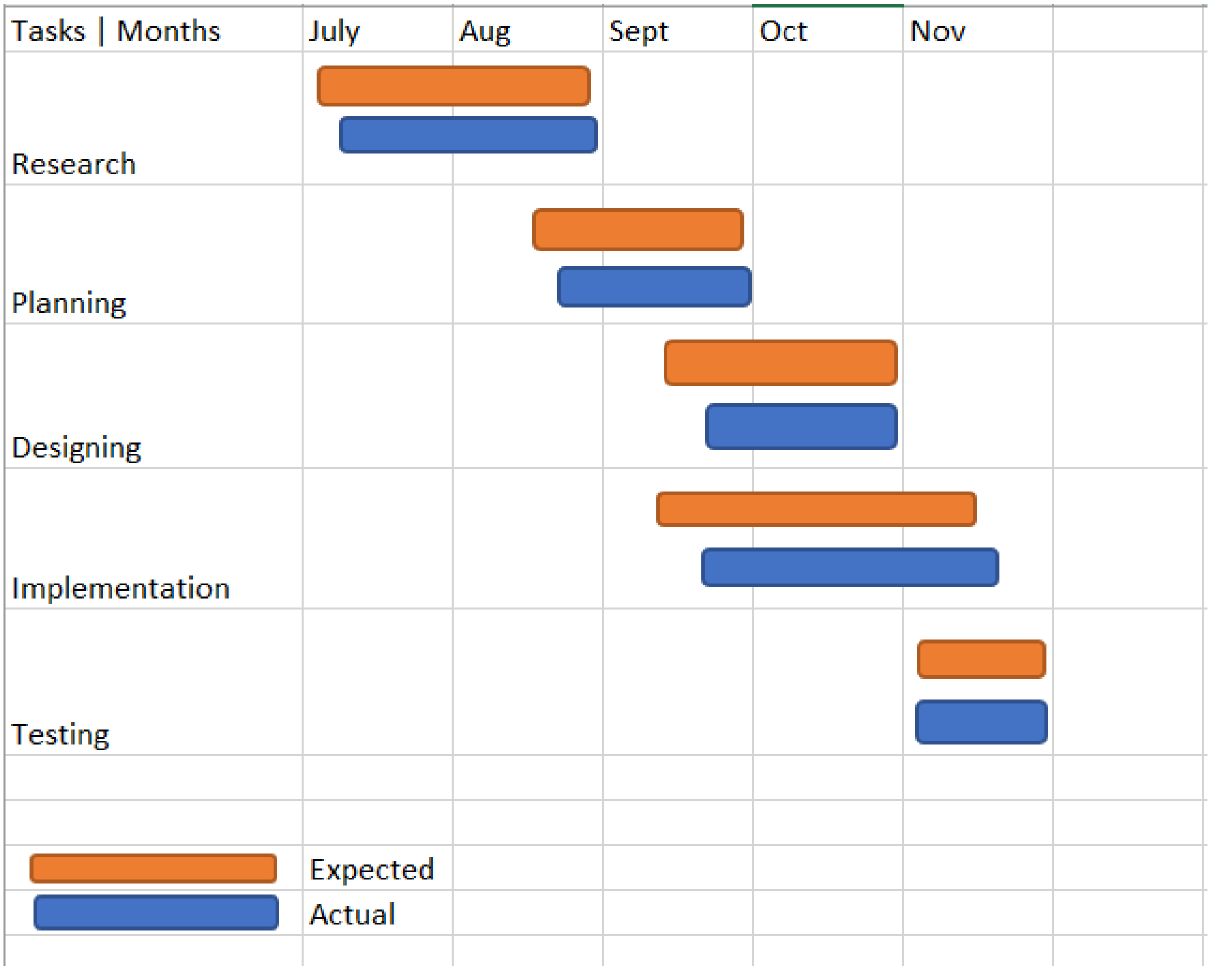


Figure 2: Gantt Chart

3.4 Software and Hardware Requirements

- **Software Requirements:**
  - MySQL Server 5.1
  - PHP on Apache
  - Restful API, JSON
  - AJAX
  - Jquery
  - JavaScript
- **Hardware Requirements:**
  - RAM -4GB or More.
  - ROM -4GB or More.
  - Intel Processor i5

3.5 Preliminary Product Description.

Login Module

This is the module where a registered user’s information will be retrieved from the database and checked.

Registration Module

If the user wants to access the functionalities or the features the website provides, they can register their information by following easy registration steps. Once the user is registered he/she can gain access to the core functionalities that this website provides.

Forgot Password

If the user has forgotten the password this module resets the password with full security via unique tokens by sending a mail to the users Gmail account.

Change Password

This module updates the user password if he/she wants to change it by validating the users Email ID.

Admin Login

The admin should log in, when the admin session is matching the session in the dashboard then only the admin gets redirected to the main dashboard.

Admin Module

The core admin functionality module plays an important part which is responsible for adding new recipes based on particular categories, view all the recipes that have been added in the database and can also update, delete them.

Admin Insert Recipe

The can insert recipes which will interlink to the main recipe pages and display the content there. The data which the admin can add are as follows:

- Title of the recipe.
- Description of the recipe.
- Adding the ingredients.
- Adding the steps.
- The number of servings it will provide.
- The amount of time it will take to prepare.
- The category the recipe belongs to:
    - Vegan
    - Vegetarian
    - Lacto veg
    - Ovo veg
- The image of the recipe.

All this data will be stored in the database.

Admin View/Delete Recipe

The admin can view and delete the recipes as well.

Save Recipes

After the user has registered, he/she can add the recipes they like to favorites. This will help them access the recipes quickly later in the future. Just by clicking on a save button the user can add the recipe.

Pantry Module

This core functionality is new which helps the user create their own pantry by choosing the ingredients they want. The recipe page will display all the ingredients which will be required to cook the recipe, the user can explicitly select any ingredient from the ingredients given or an option to select all the ingredients at once. When added, this will

Create a list for the user which they can access anytime they want, and if not they there is an option to delete or remove the pantry ingredients as well.

Menu Creation Module

One of the 2nd core complex functionality this website provides is the Menu creation. A menu creation is a feature that allows the user to create a list of recipes they would like to cook for the entire week by pre saving or storing them in a separate page which can be accessed in seconds rather than searching and thinking of what to cook the next day or any special occasion they want. An option will be provided to the user on the recipe page to create or add the recipe to a menu, further it will have a dropdown of the number of days, and the user can choose any day for which they want to cook this meal. That information will be stored in the user's unique id in the database. Later the user can also remove or delete the recipe so that they can add other recipes on the same day.

Subscription Module

This module is to become a member of the 30MinRecipes website.

Mail the Admin Module

If the users wish to add a recipe of their own they can do so by mailing the Admin and telling them the recipe they wish to add.

Read About Us Module

The users can also read about the Recipe Types provided by the website, what are the ingredients included the methods and servings and timings respectively.

3.6 Conceptual Modules

3.6.1 Event Table

Sr. No	**Event**	**Trigger**	**Source**	**Use Case**	**Response**	**Destination**
1.	User registration	Registration request	User	Entry of details into database	Homepage	User
2.	Login	Login request	User	Validate user details	Homepage	User
3.	Admin Login	Admin Login request	Admin	Validating the Admin	Admin Dashboard	Admin
4.	Forgot Password	Enter Valid Email	Admin/ User	Changing the password via Email Id	Change Password	Admin/User
5.	Create menu	User clicks on menu button	User	User adds menu recipes for the days stored in the db	Recipe page	User
6.	Pantry	Create pantry	User	User adds individual ingredients I the pantry	Recipe page	User
7.	favorites	Like button	User	Clicked on like button	Recipe page	User
8.	Admin insertion of recipes	Adding the details	Admin	Entry of the recipes in the db	Admin page	Admin

9.	subscription	Subscribe button	User	Added in the subscription db	Homepage	User
10.	Comments	Comment on post	User	Comments will be stored in the db	Recipe page	User
11.	Mail Admin	Mail	User	Mailed to the User	About Us	Admin
12.	Search Recipe	Search	User	Recipes will be displayed	Homepage	Recipe Page

Table 3.6.1: Event Table

ER Diagram

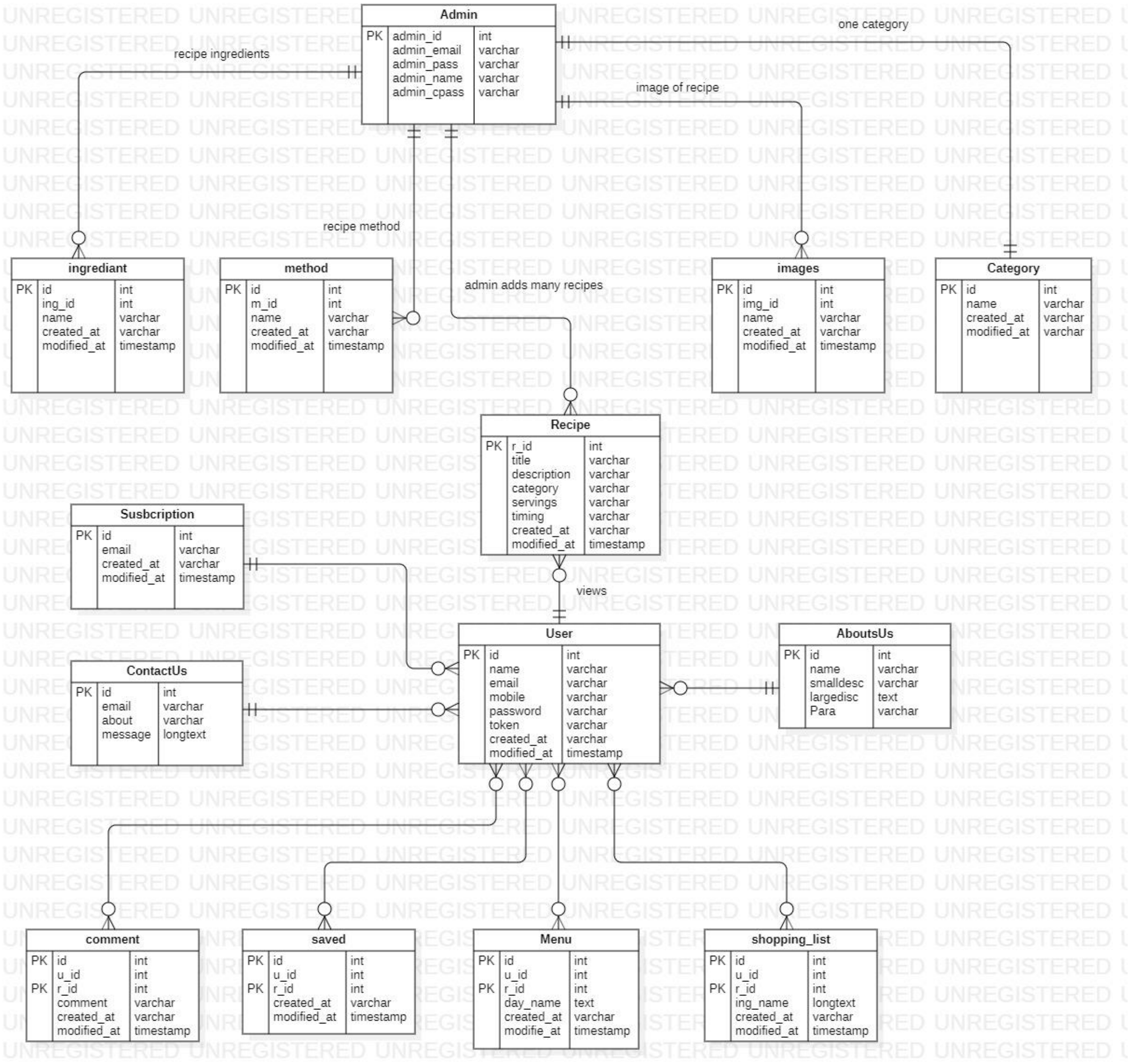


Figure.1 ER DIAGRAM

Use Case Diagram

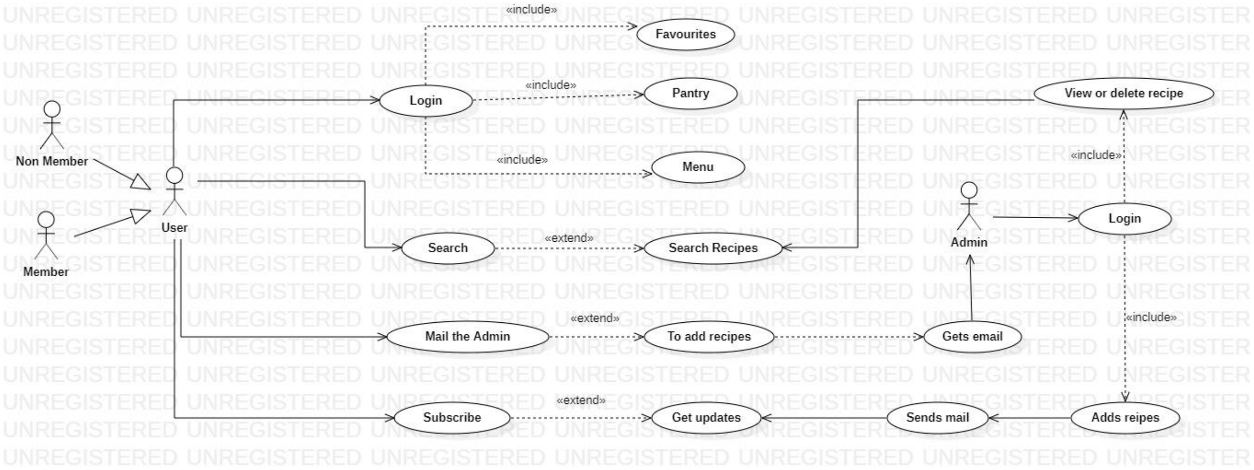


Figure.2 USE CASE DIAGRAM

Class Diagram

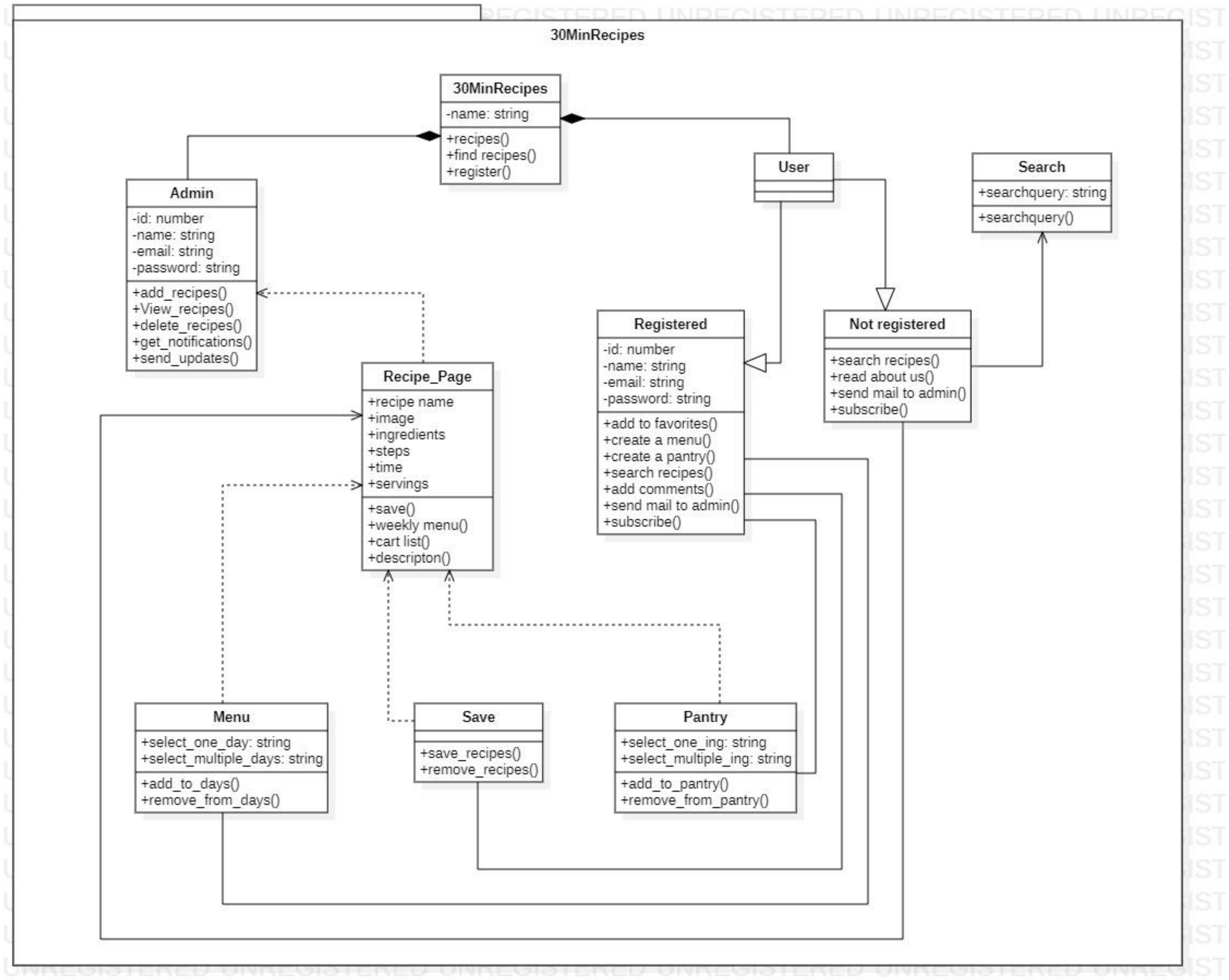


Figure.3 CLASS DIAGRAM

Object Diagram

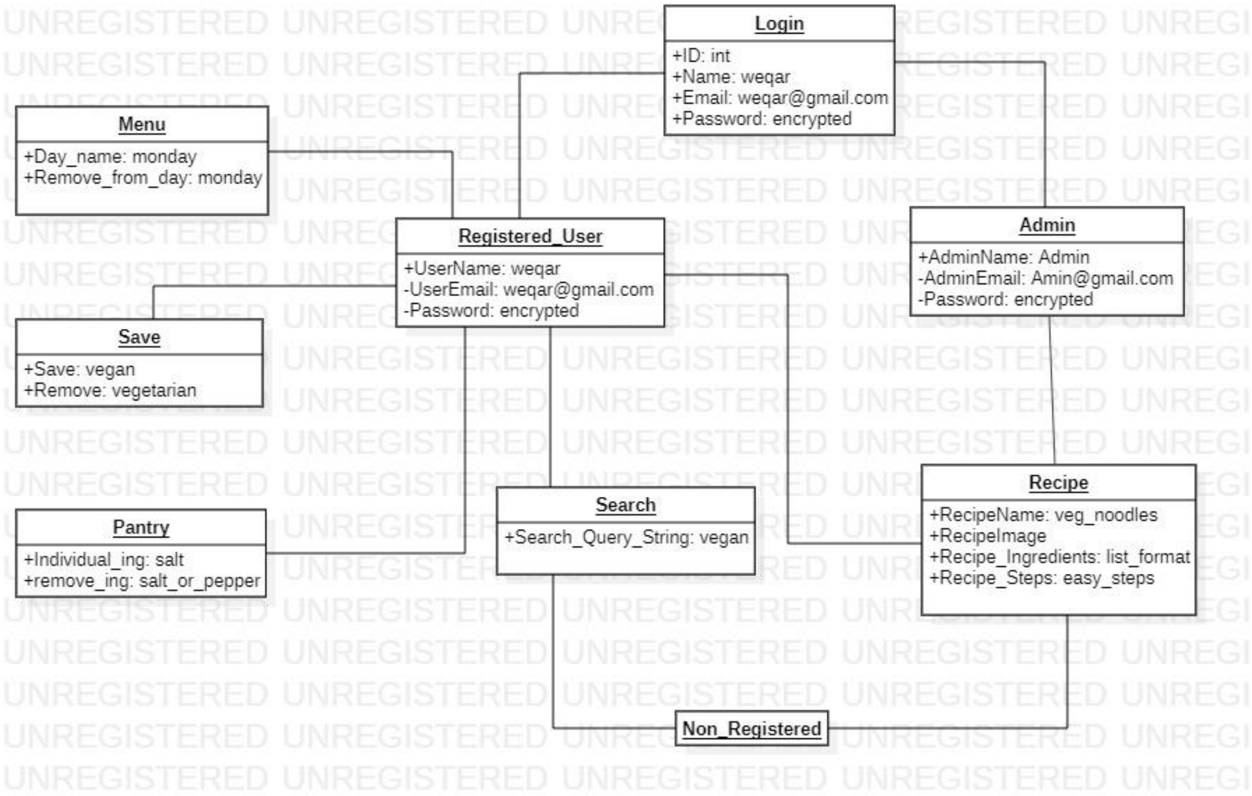


Figure.4 OBJECT DIAGRAM

Activity Diagram

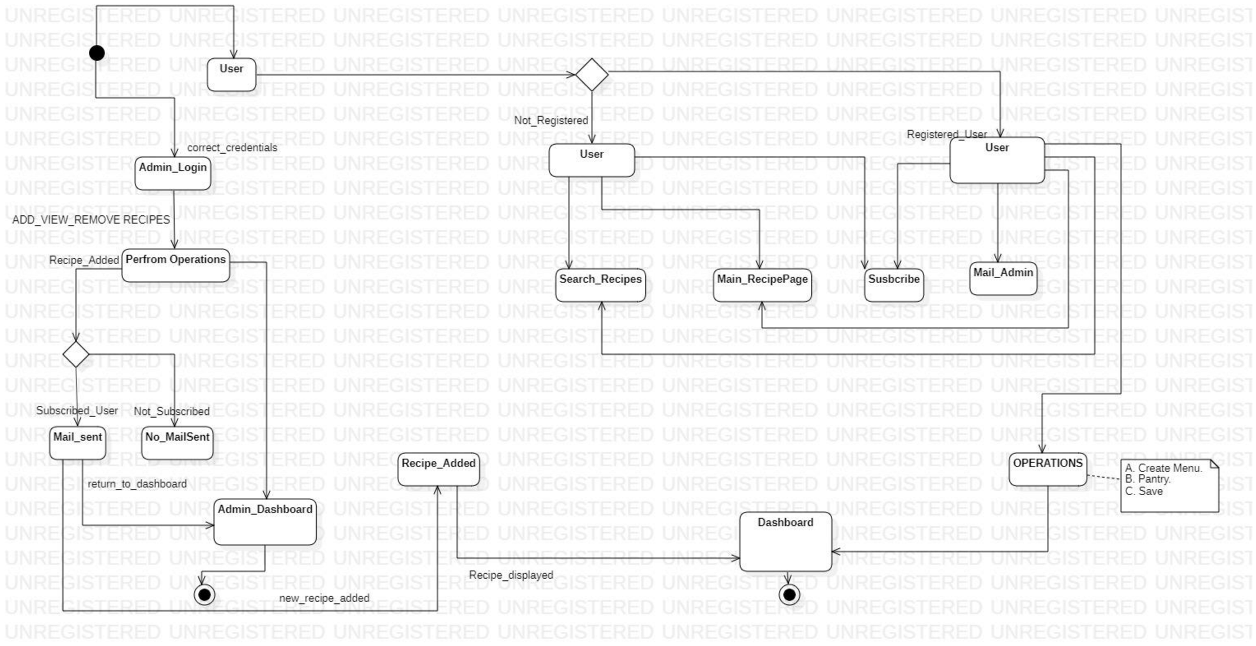


Figure.5 ACTIVITY DIAGRAM

Sequence Diagram

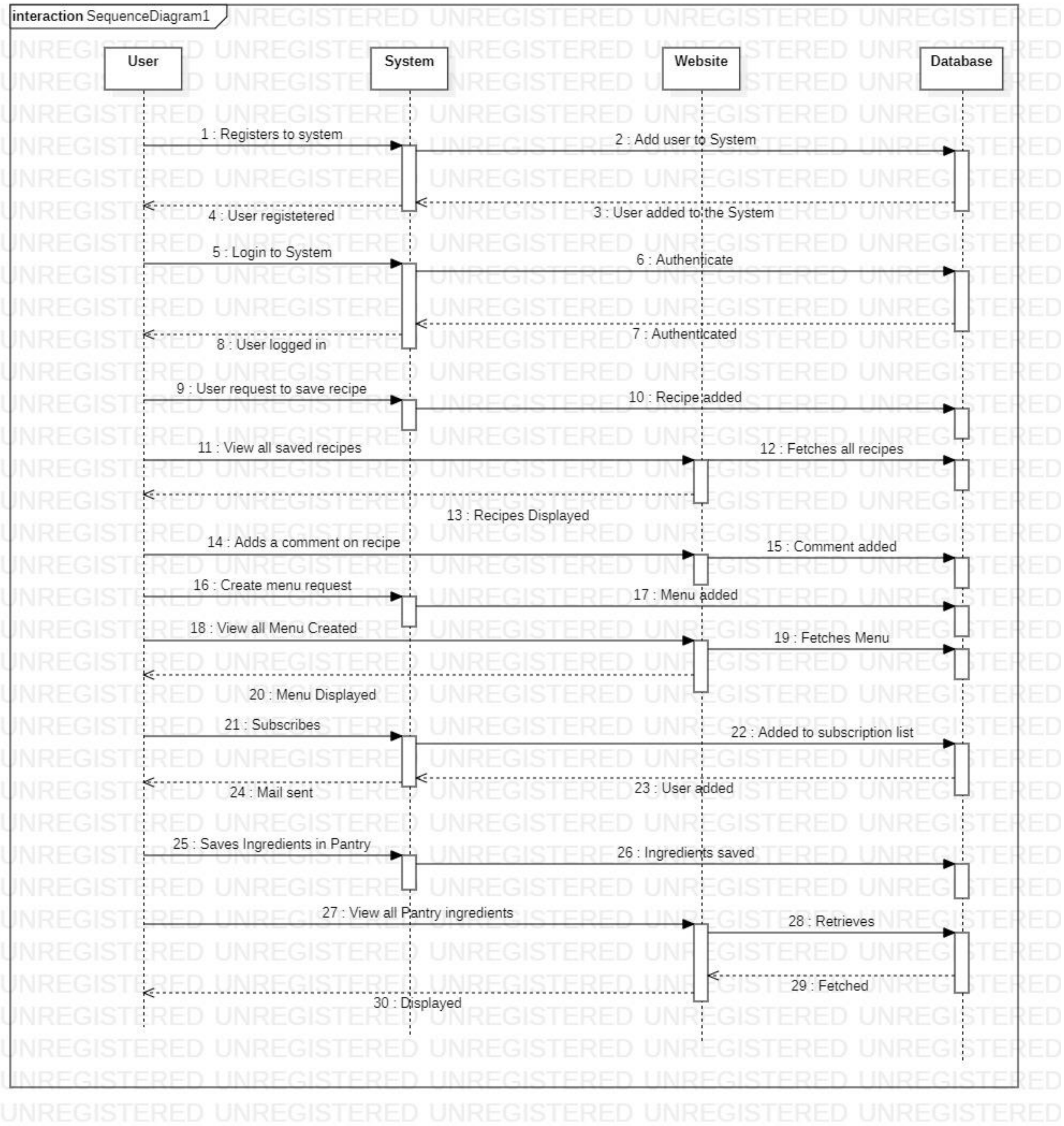


Figure.6 SEQUENCE DIAGRAM

State Diagram

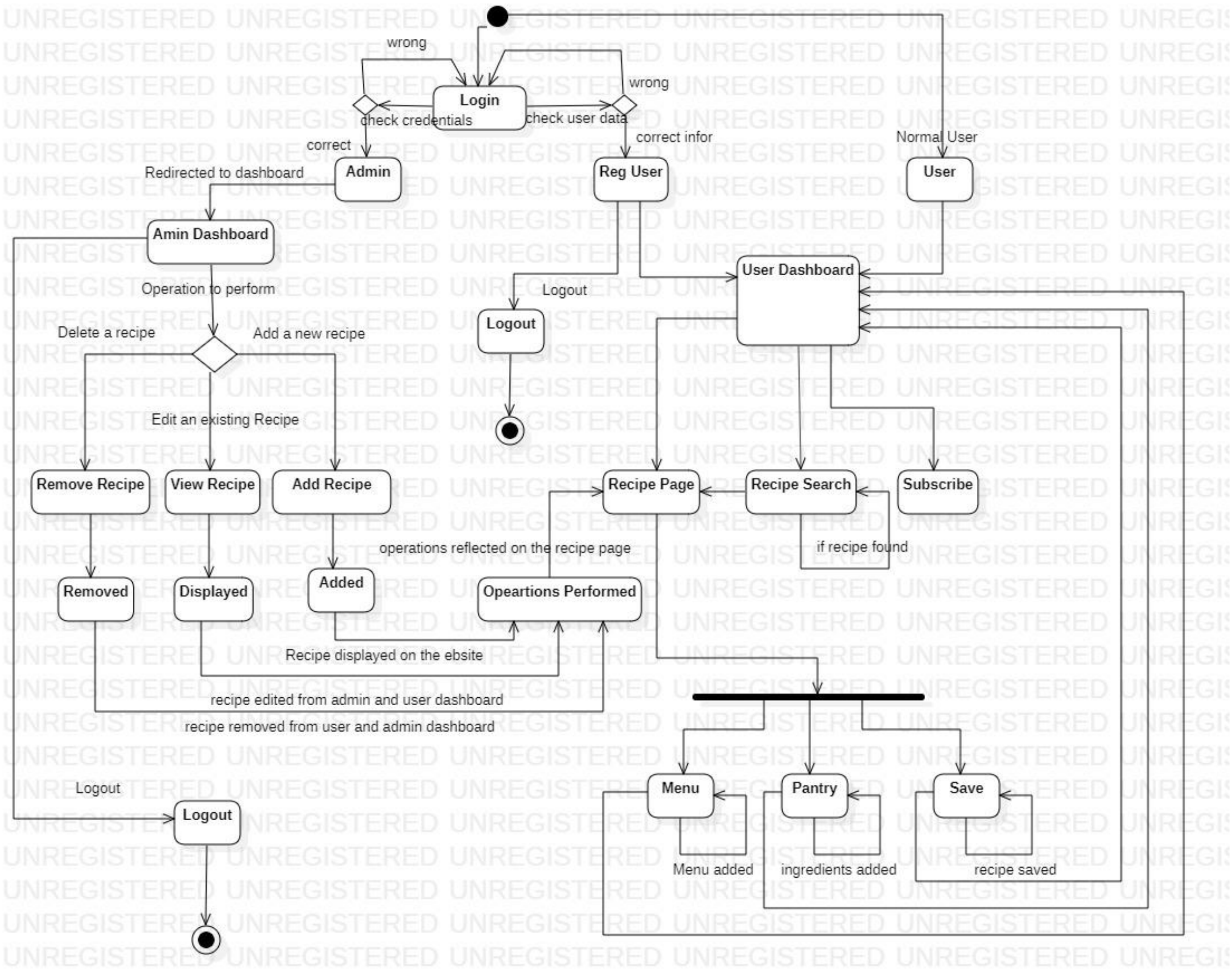


Figure.7 STATE DIAGRAM

Package Diagram

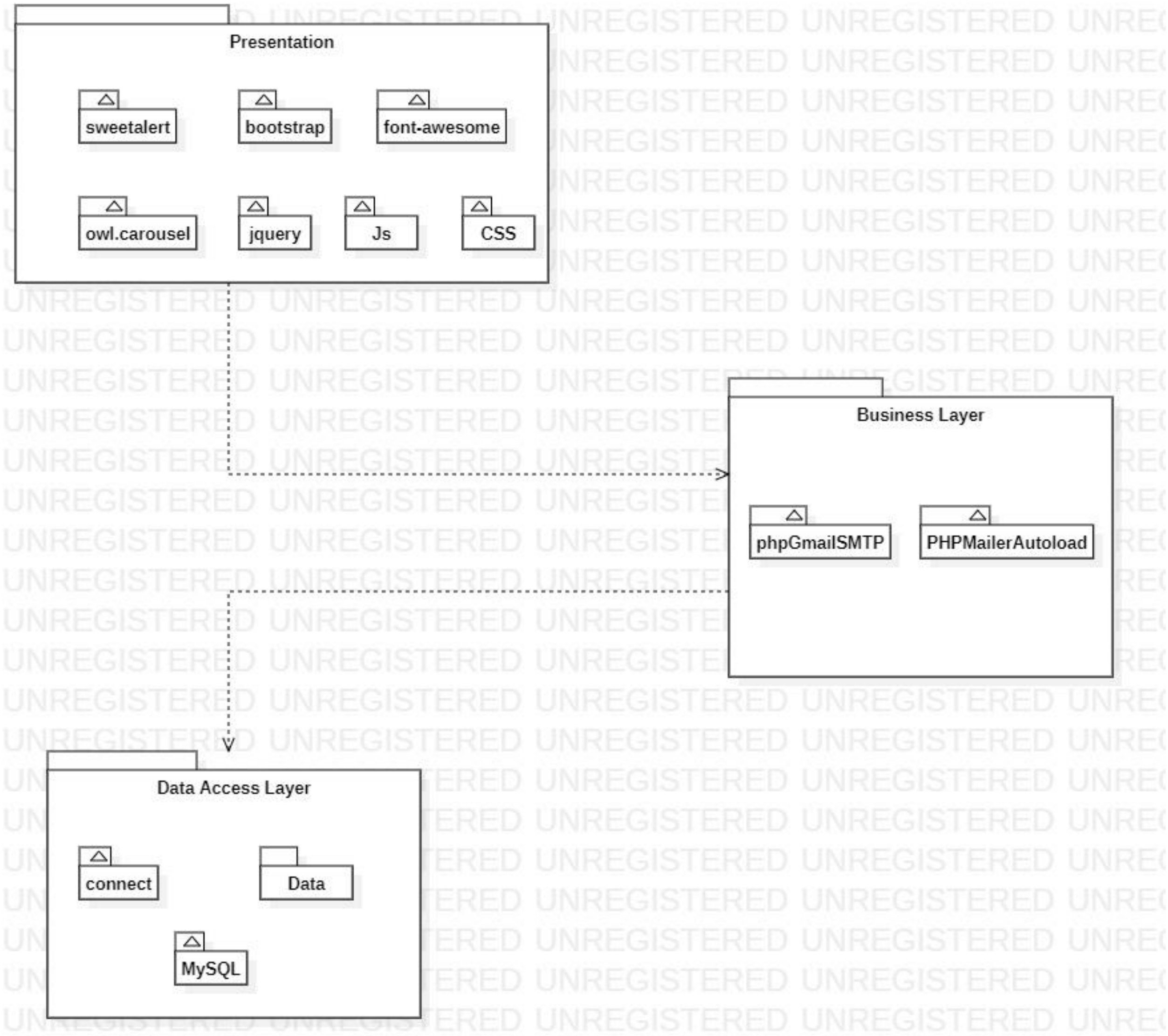


Figure.8 PACKAGE DIAGRAM

Deployment Diagrma

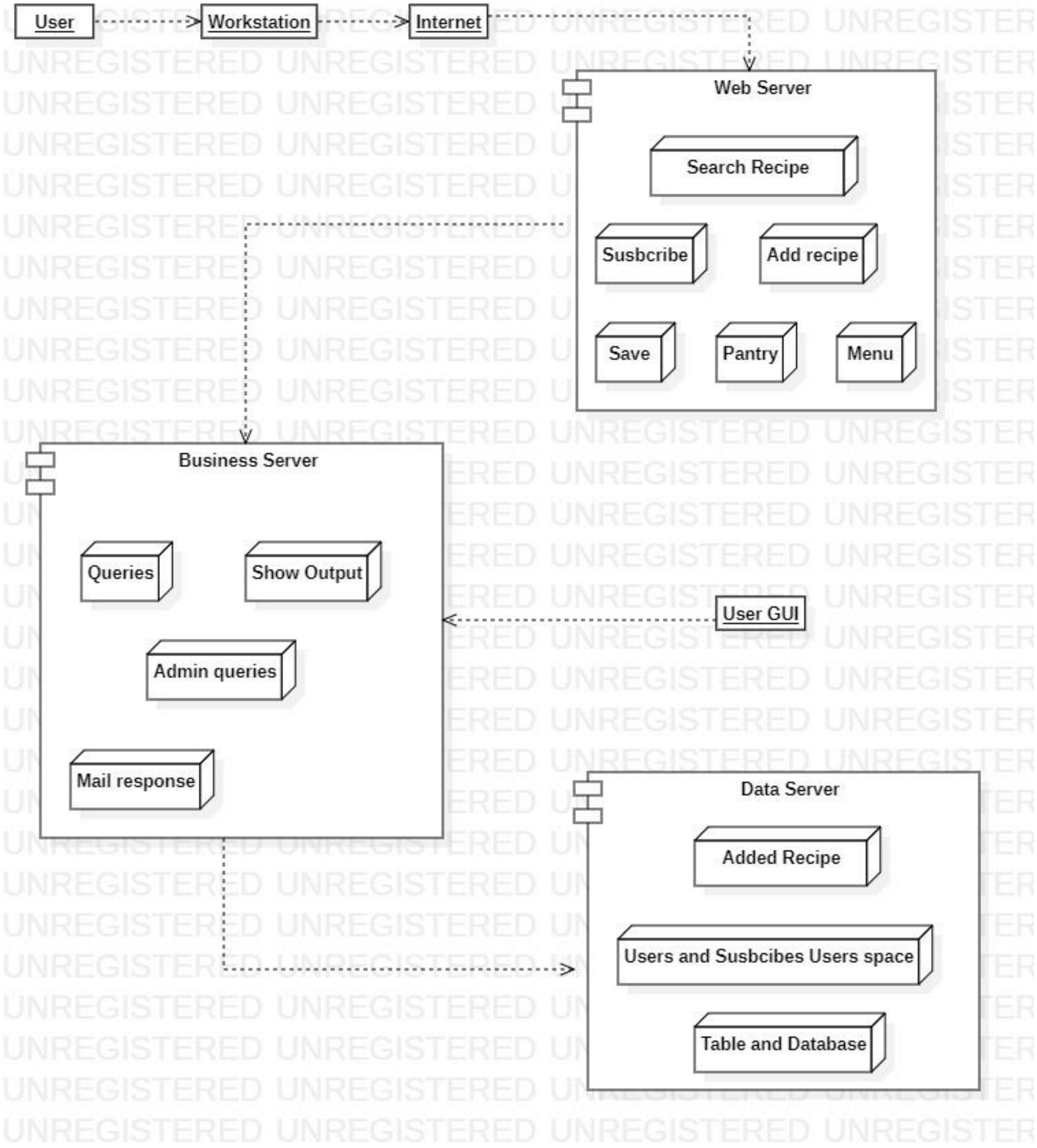


Figure.9 DEPLOYMENT DIAGRAM

Component Diagram

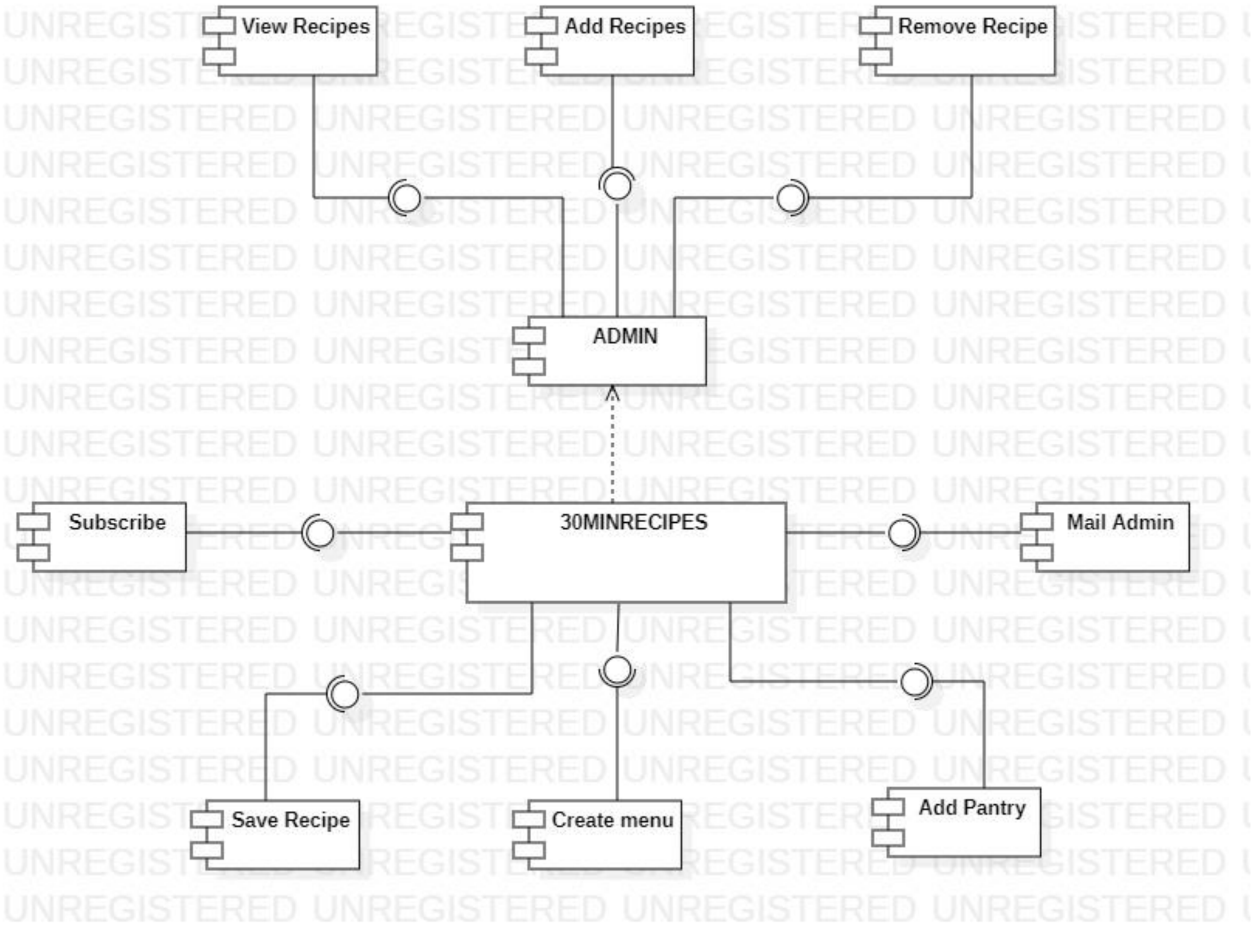


Figure.10 COMPONENT DIAGRAM

Data Flow Diagram Level 0

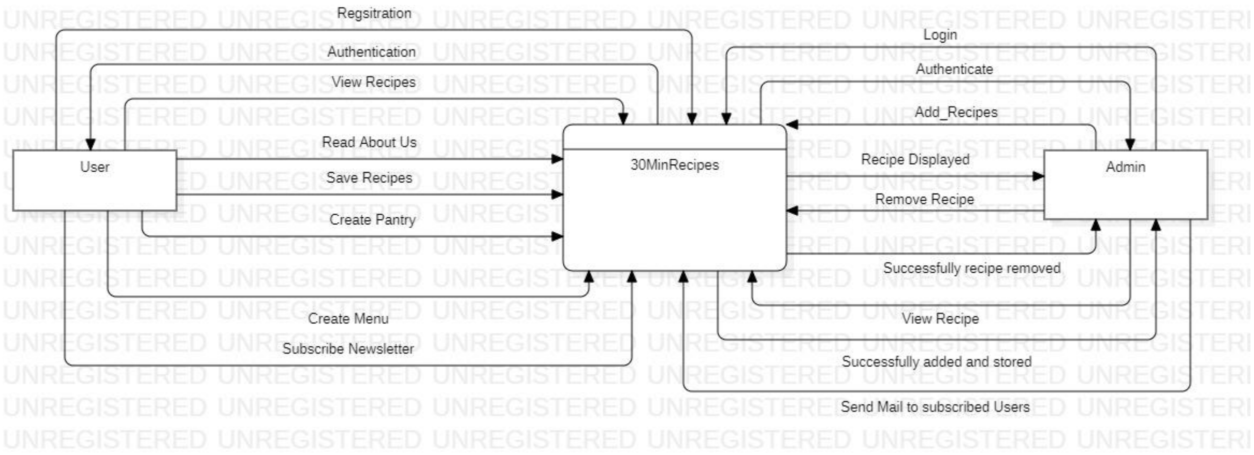


Figure.11 DATA FLOW DIAGRAM LEVEL 0

Data Flow Diagram Level 1

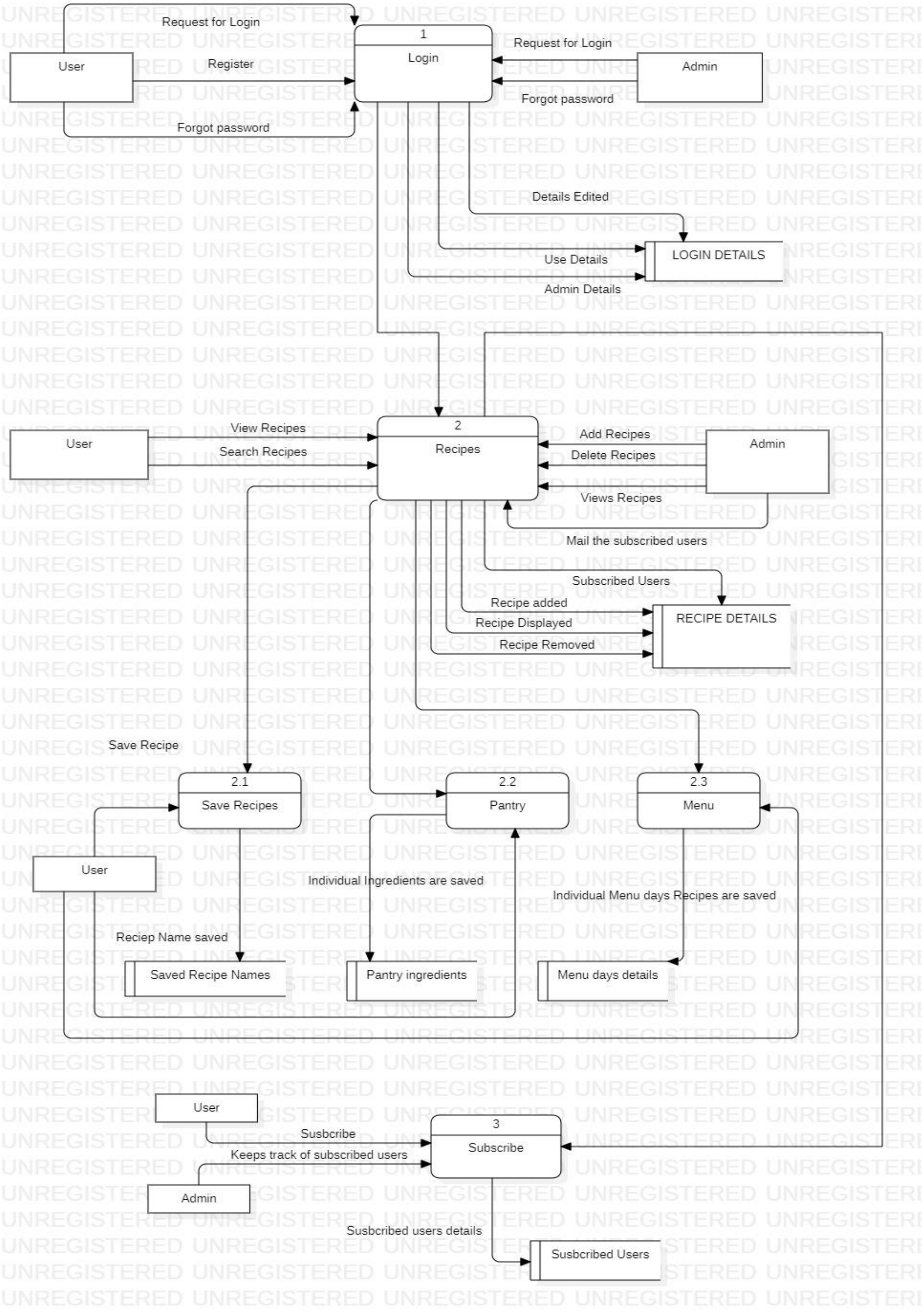


Figure.12 DATA FLOW DIAGRAM LEVEL 1

4 SYSTEM DESIGN

4.1 basic Module:

Login Module

This is the module where a registered user's information will be retrieved from the database and checked.

Registration Module

If the user wants to access the functionalities or the features the website provides, they can register their information by following easy registration steps. Once the user is registered he/she can gain access to the core functionalities that this website provides.

Forgot Password

If the user has forgotten the password this module resets the password with full security via unique tokens by sending a mail to the users Gmail account.

Change Password

This module updates the user password if he/she wants to change it by validating the users Email ID.

Admin Login

The admin should log in, when the admin session is matching the session in the dashboard then only the admin gets redirected to the main dashboard.

Admin Module

The core admin functionality module plays an important part which is responsible for adding new recipes based on particular categories, view all the recipes that have been added in the database and can also update, delete them.

Admin Insert Recipe

The can insert recipes which will interlink to the main recipe pages and display the content there. The data which the admin can add are as follows:

- Title of the recipe.
- Description of the recipe.
- Adding the ingredients.
- Adding the steps.
- The number of servings it will provide.
- The amount of time it will take to prepare.
- The category the recipe belongs to:
  - Vegan
  - Vegetarian
  - Lacto veg
  - Ovo veg
- The image of the recipe.

Admin View/Delete Recipe

The admin can View and delete the recipes as well.

Save Recipes

After the user has registered, he/she can add the recipes they like to favorites. This will help them access the recipes quickly later in the future. Just by clicking on a save button the user can add the recipe.

Pantry Module

This core functionality is new which helps the user create their own pantry by choosing the ingredients they want. The recipe page will display all the ingredients which will be required to cook the recipe, the user can explicitly select any ingredient from the ingredients given or an option to select all the ingredients at once. When added, this will create a list for the user which they can access anytime they want, and if not they there is an option to delete or remove the pantry ingredients as well.

Menu Creation Module

One of the 2nd core complex functionality this website provides is the Menu creation. A menu creation is a feature that allows the user to create a list of recipes they would like to cook for the entire week by pre saving or storing them in a separate page which can be accessed in seconds rather than searching and thinking of what to cook the next day or any special occasion they want. An option will be provided to the user on the recipe page to create or add the recipe to a menu, further it will have a dropdown of the number of days, and the user can choose any day for which they want to cook this meal. That information will be stored in the user's unique id in the database. Later the user can also remove or delete the recipe so that they can add other recipes on the same day.

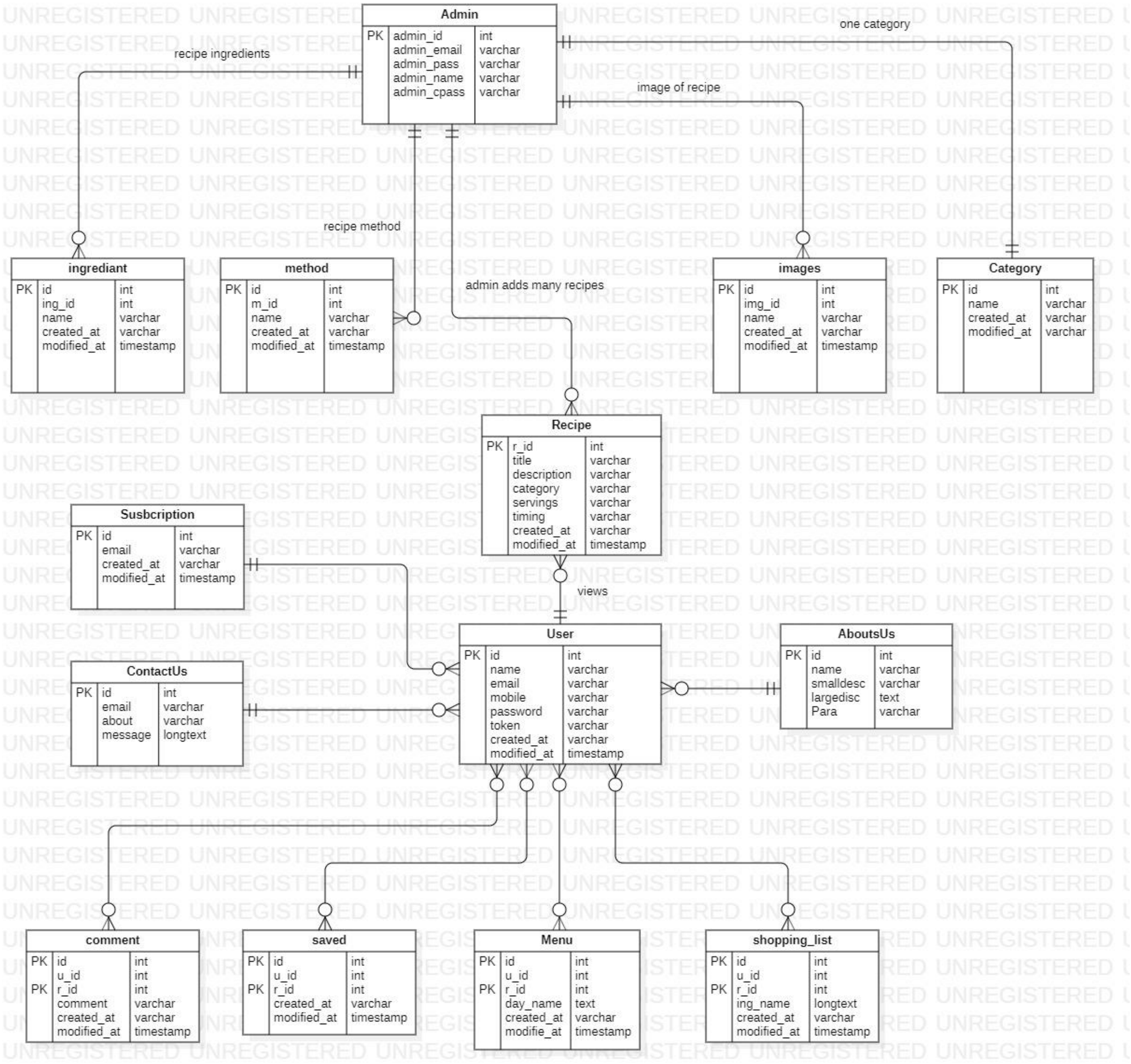
Subscription Module

This module is to become a member of the 30MinRecipes website.

About us Module

It contains detailed description of what the categories are about, the meals types, the ingredients each recipe contains, etc.

4.2 data Design



Admin		
PK	admin_id	int
	admin_email	varchar
	admin_pass	varchar
	admin_name	varchar
	admin_cpass	varchar

ingrediant		
PK	id	int
	ing_id	int
	name	varchar
	created_at	varchar
	modified_at	timestamp

method		
PK	id	int
	m_id	int
	name	varchar
	created_at	varchar
	modified_at	timestamp

images		
PK	id	int
	img_id	int
	name	varchar
	created_at	varchar
	modified_at	timestamp

Category		
PK	id	int
	name	varchar
	created_at	varchar
	modified_at	varchar

Recipe		
PK	r_id	int
	title	varchar
	description	varchar
	category	varchar
	servings	varchar
	timing	varchar
	created_at	varchar
	modified_at	timestamp

Susbcription		
PK	id	int
	email	varchar
	created_at	varchar
	modified_at	timestamp

ContactUs		
PK	id	int
	email	varchar
	about	varchar
	message	longtext

User		
PK	id	int
	name	varchar
	email	varchar
	mobile	varchar
	password	varchar
	token	varchar
	created_at	varchar
	modified_at	timestamp

AboutsUs		
PK	id	int
	name	varchar
	smalldesc	varchar
	largedisc	text
	Para	varchar

comment		
PK	id	int
	u_id	int
PK	r_id	int
	comment	varchar
	created_at	varchar
	modified_at	timestamp

saved		
PK	id	int
	u_id	int
PK	r_id	int
	created_at	varchar
	modified_at	timestamp

Menu		
PK	id	int
	u_id	int
PK	r_id	int
	day_name	text
	created_at	varchar
	modifie_at	timestamp

shopping_list		
PK	id	int
	u_id	int
PK	r_id	int
	ing_name	longtext
	created_at	varchar
	modified_at	timestamp

Figure 4.2.1 Schema Design

Data Dictionary

- Admin

Field	DataType	Size	Constraints
Amin_id	int	10	PRIMARY KEY
Admin_name	varchar	255	
Admin_email	Varchar	255	
Admin_pass	Varchar	255	
Admin_cpass	varchar	255	

- User

Field	DataType	Size	Constraints
id	Int	10	PRIMARY KEY
Name	Varchar	255	
Email	Varchar	255	
Mobile	Varchar	255	
Password	Varchar	255	
Created_at	Varchar	255	
Modified_at	timestamp	255	current_timestamp()

- Subscriptions

Field	DataType	Size	Constraints
id	int	10	PRIMARY KEY
email	Varchar	255	
Created_at	varchar	255	
Modified_at	timestamp		current_timestamp()

- Aboutus

Field	DataType	Size	Constraints
id	int	10	PRIMARY KEY
name	varchar	255	
smalldesc	varchar	255	
largedesc	text	255	
Para	varchar	255	

- Contactus

Field	DataType	Size	Constraints
id	int	10	PIMARY KEY
email	varchar	255	
about	varchar	255	
message	longtext	255	

- Category

Field	DataType	Size	Constraints
id	int	10	PRIMARY KEY
name	varchar	255	
Created_at	varchar	255	
Modified_at	timestamp		current_timestamp()

- Recipe

Field	DataType	Size	Constraints
R_id	int	10	PRIMARY KEY
Title	Varchar	255	
Description	Varchar	255	
category	Varchar	255	
servings	Varchar	255	
timing	Varchar	255	
Created_at	Varchar	255	
Modified_at	Timestamp		current_timestamp()

- Ingredient

Field	DataType	Size	Constraints
Id	Int	10	PRIMARY KEY
Ing_id	Int	255	
Name	Varchar	255	
Created_at	Varchar	255	
Modified_at	Timestamp		current_timestamp()

- Method

Field	DataType	Size	Constraints
Id	Int	10	PRIMARY KEY
M_id	Varchar	255	
Name	Varchar	255	
Create_at	Varchar	255	
Modified_at	Timestamp		current_timestamp()

- Images

Field	DataType	Size	Constraints
Id	Int	10	PRIMARY KEY
Img_id	Varchar	255	
Name	varchar	255	
Created_at	Varchar	255	
Modified_at	Timestamp		current_timestamp()

- Comments

Field	DataType	Size	Constraints
Id	Int	10	PRIMARY KEY
Img_id	varcha	255	
Name	Varchar	255	
Created_at	Varchar	255	
Modified_at	Timestamp		current_timestamp()

- Menu

Field	DataType	Size	Constraints
Id	Int	10	PRIMARY KEY
U_id	longtext	255	
r_id	Int	255	
day_name	Varchar	255	
Created_at	Varchar	255	
Modified_at	Timestamp		current_timestamp()

- Saved

Field	DataType	Size	Constraints
Id	Int	10	PRIMARY KEY
U_id	Longtext	255	
R_id	Int	255	
Created_at	Varchar	255	
Modified_at	timestamp		current_timestamp()

- Shopping_list

Field	DataType	Size	Constraints
Id	Int	10	PRIMAR KEY
U_id	Longtext	255	
R_id	Int	255	
Ing_name	Longtext	255	
Created_at	Varchar	255	
Modified_at	Timestamp		current_timestamp()

4.2.2 Data constraint and integrity

- Auto Increment Constraint

  Auto increment attribute when specified on a column with a numeric data types, generates numbers sequentially whenever a new row is added into the database. The Auto Increment is commonly used to generate primary keys. The defined data type on the auto increment should be large enough to accommodate many records. It is required because, to set the primary key because it will keep incrementing its value by 1. So, there will be no redundancy value in the ID column.

- Primary Key Constraint

  A primary key constraint defines a column or a combination of columns whose values uniquely identifies each row in a table. This column or the combination of columns is also known as primary key of the table. Uniquely identifies each row/record in a database table. The Id of student, teacher, class, etc, needed for uniquely identifying different rows in a table.

- Foreign Key Constraint

  A foreign key is a key used to link two tables together. This is sometimes also called as a referencing key. A foreign key is a column or a combination of columns whole values match a primary key in a different table. Uniquely identifies each row/record in a database table.

- Mandatory Constraint

  The constraint enforces a column to NOT accept NULL values.

- Unique Key Constraint

  A unique key is a set of one or more than one fields/columns of a table that uniquely identifies a record in a database table. You can say that it is little like primary key but it can accept only one null value and it cannot have duplicate values. Ensures that all values in a column are different.

4.3 User Interface Design.

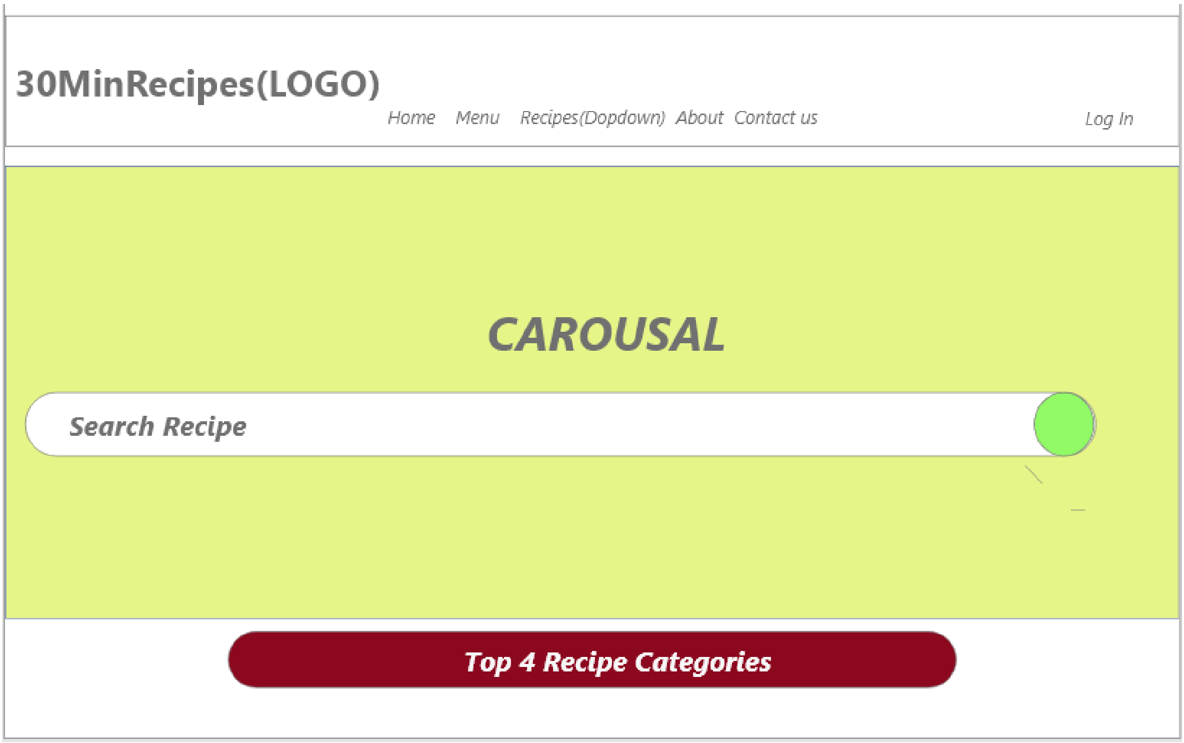


Figure: 4 Home Page

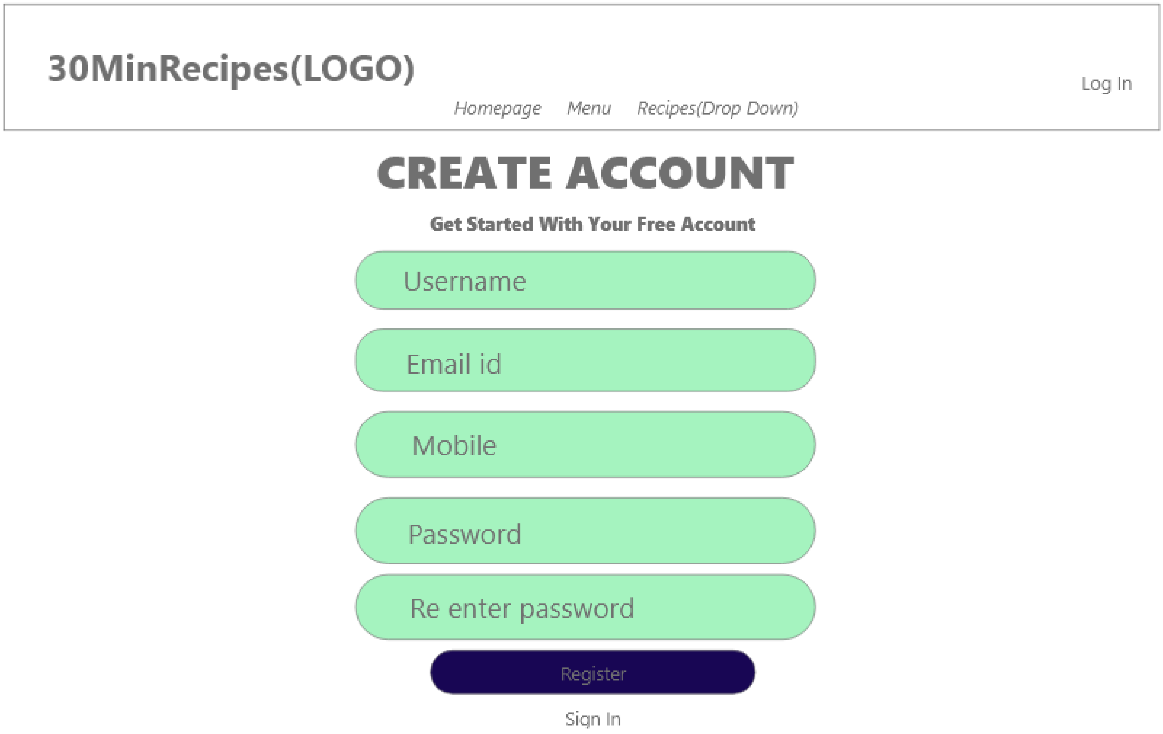


Figure: 5 Registration Page

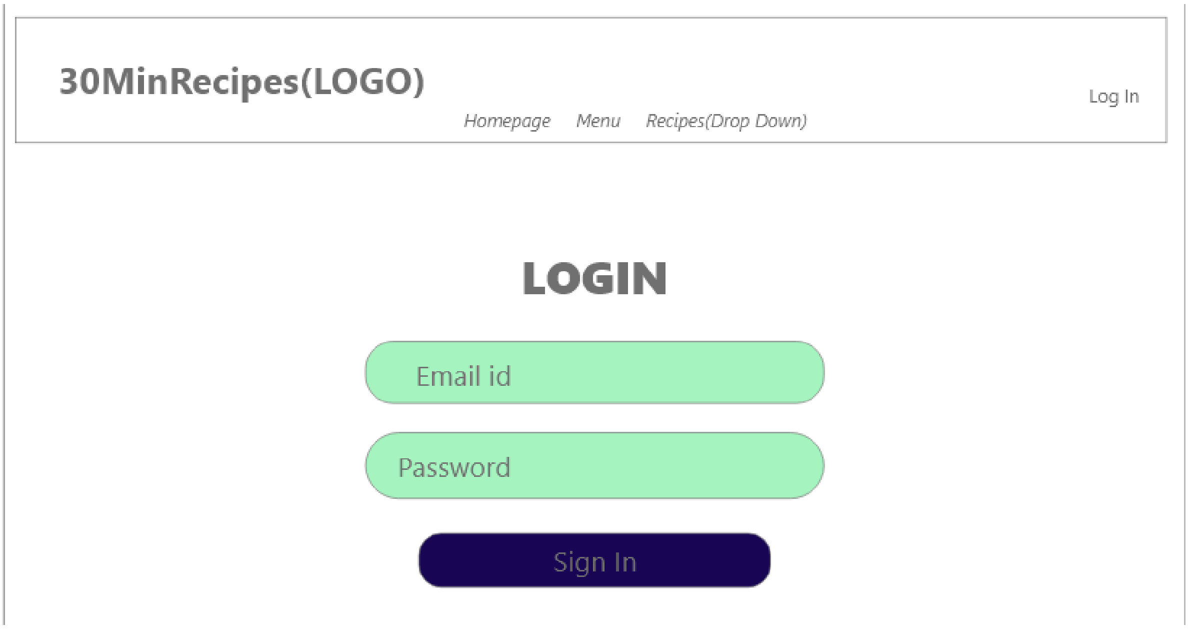


Figure: 6 Login Page

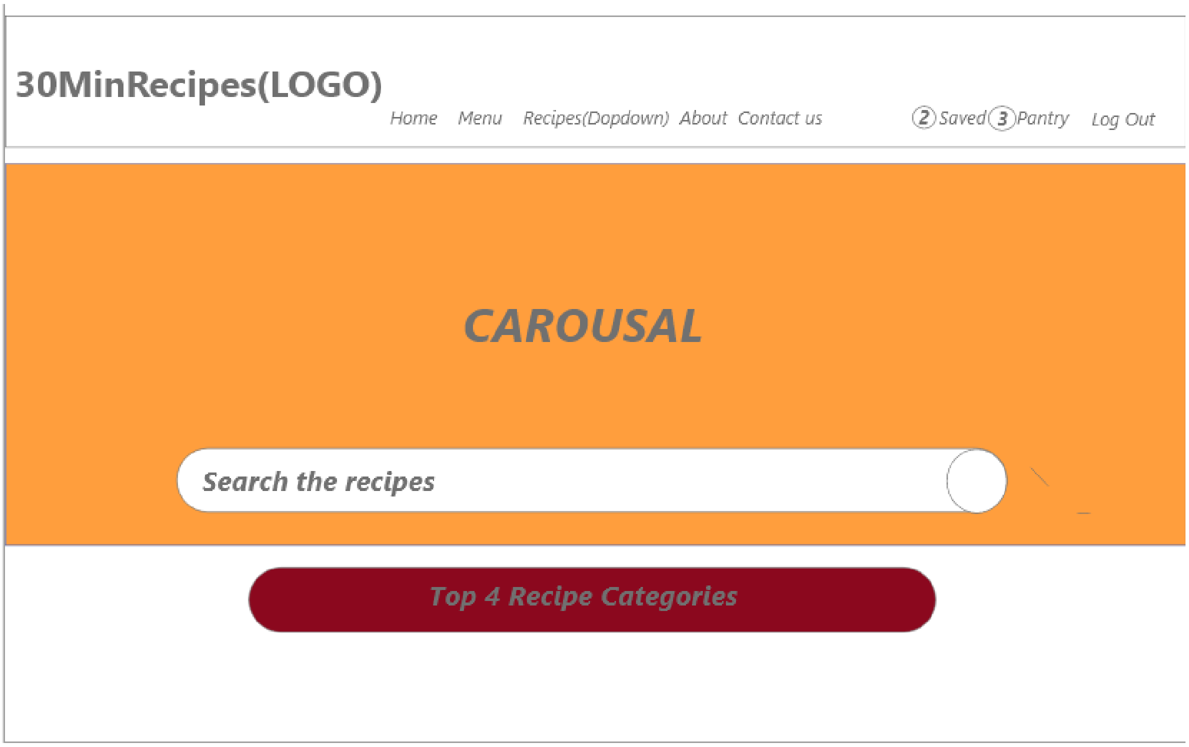


Figure: 7 User Home Page

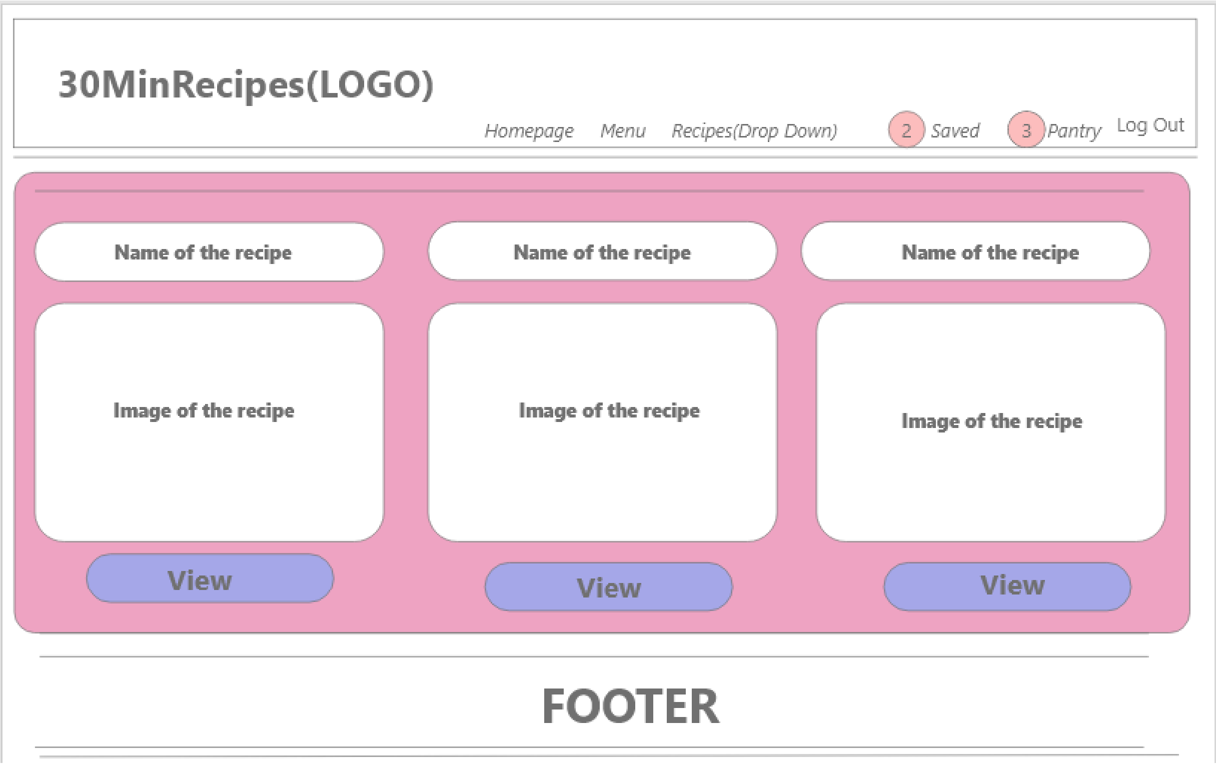


Figure: 8 Recipe Categories

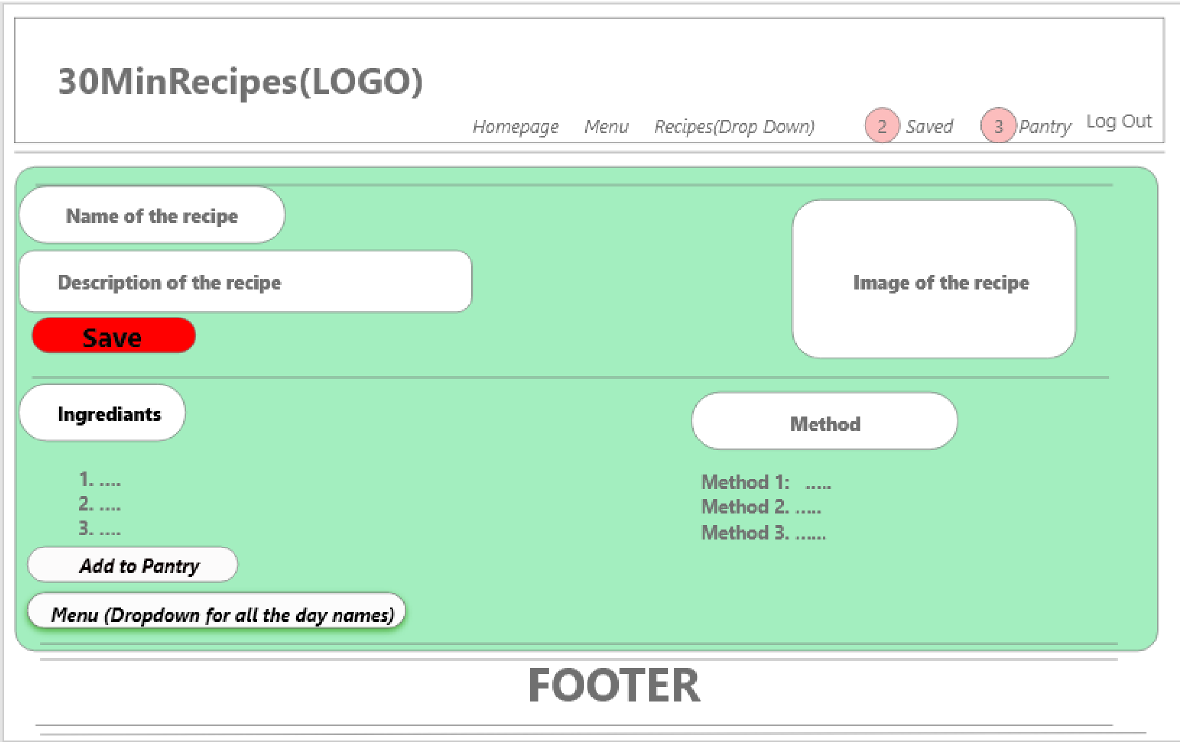


Figure: 9 Recipe Page

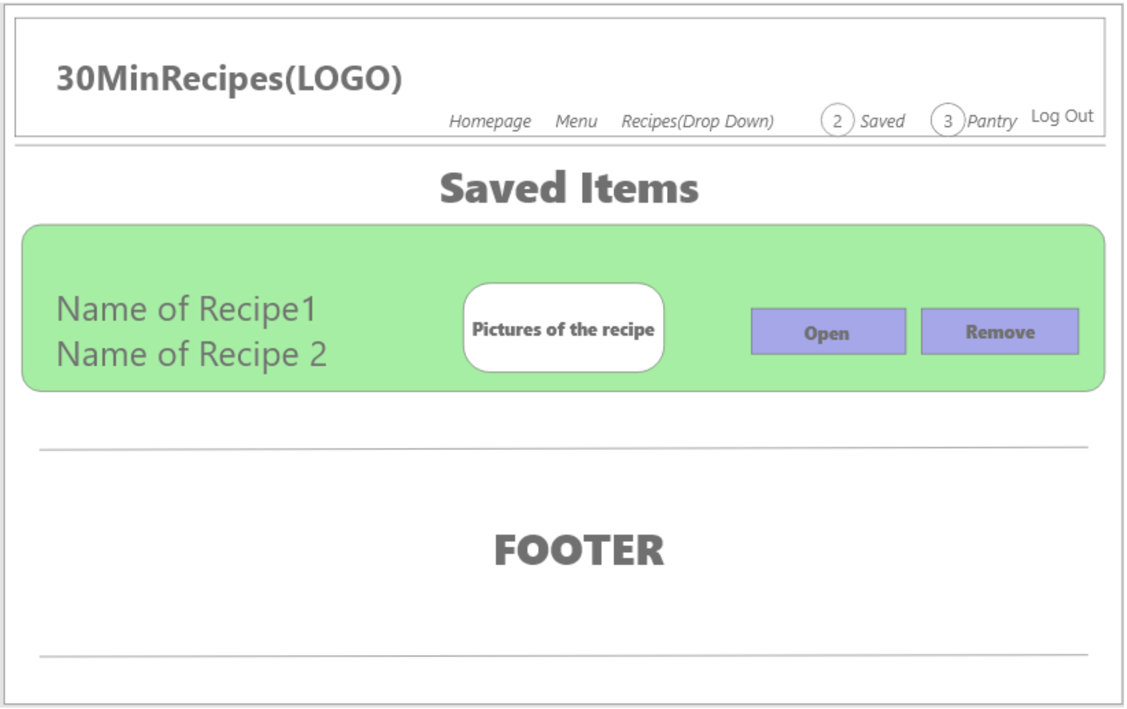


Figure: 10 Saved Recipe Page

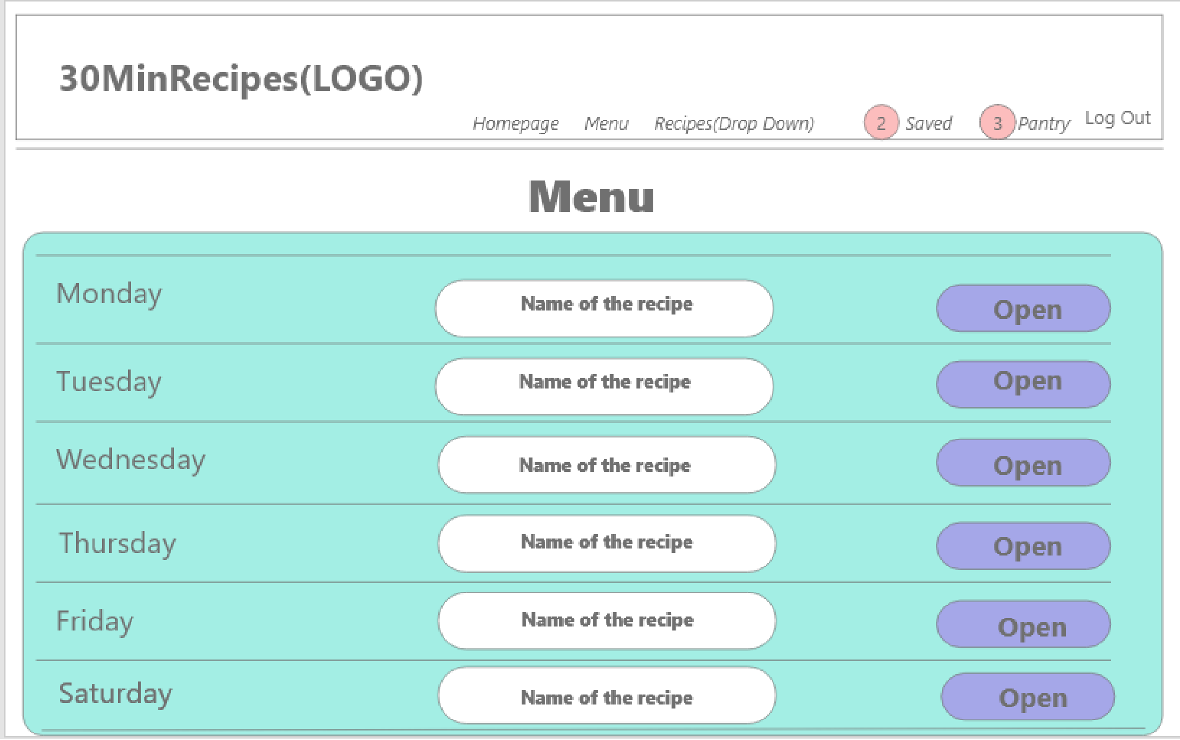


Figure: 11 Menu Page

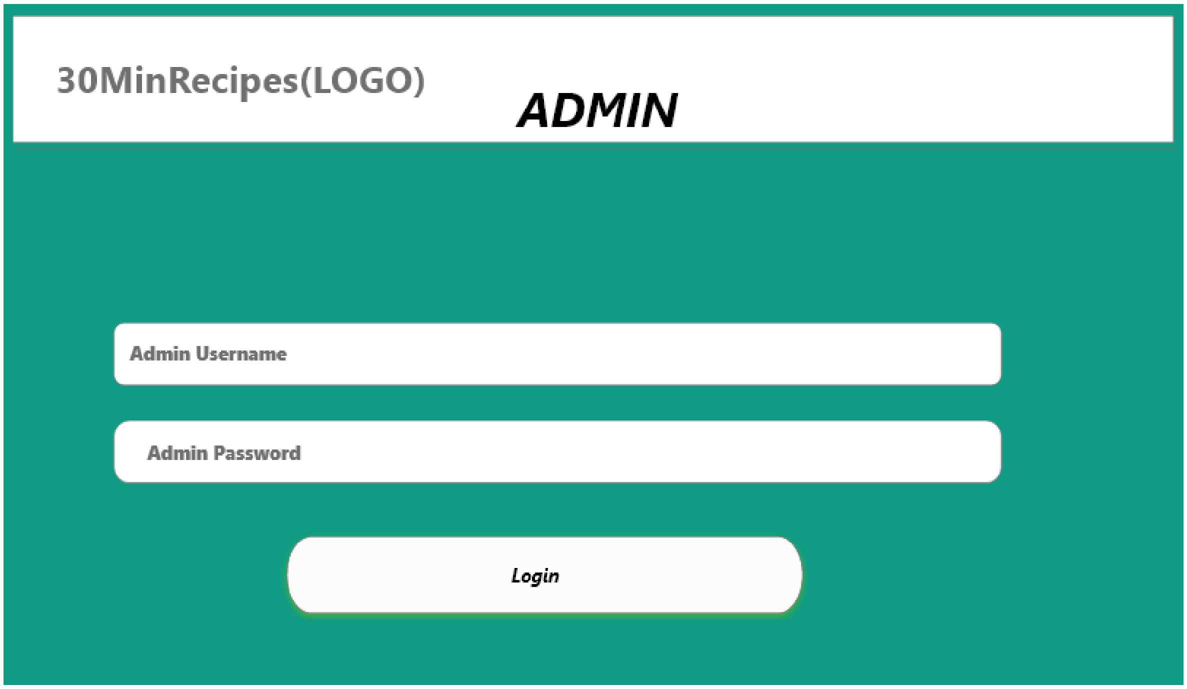


Figure: 12 Admin Login Page

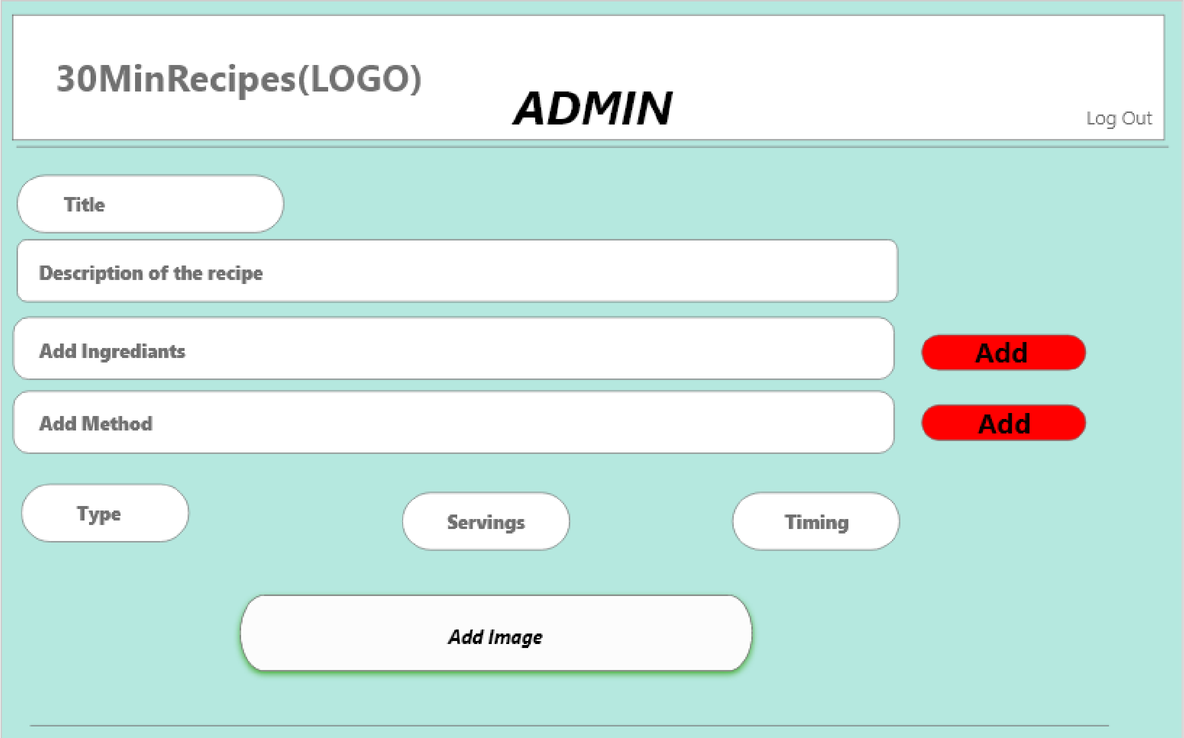


Figure: 13 Admin insert Recipe Page

4.4 Security Issues.

1. Password encryption:
    - We are using Password Encryption Algorithm technique because my project has all the data in the database. To make my platform more secure, the Hashing Technique is used to encrypt the Passwords which converts the normal text and creates a new password hash using one-way hashing algorithm. This maintains the security such that the hackers are unable to decrypt the passwords.
    - Another security feature that is dependent on the users to use is they can Change their passwords to make it stronger.
    - Each User is provided with a unique Token, this becomes helpful when the user has forgotten the password and wants to update it via an email link.

2. Authorization
    - System authorises only the admin to insert the recipe and does not authorise other users to do so. It checks whether the user is a normal user or an admin because the user can misuse the information and add unnecessary information.

3. Authentication:
    - This approach is used because every user has to authenticate him/her to operate and have access to their dashboard in the system. This is achieved because every user has a unique ID, username, and email.

4. Manage Recipes
    - Only the admin in the system is allowed to manage the recipes operations like adding a recipe, displaying the recipe, and deleting the recipe.

4.5 Test Case Design.

Test Condition	Input Selected	Expected Result	Actual Result
User/Admin Login	Email ID=”” Password=””	Username and Password Fields are empty	Username and Password Fields are empty
	Email ID=”user@gmail.com” Password=”1234”	Authenticates the User’s Gmail and Password and redirect to the Homepage.	Authenticates the User’s Gmail and Password and redirect to the Homepage.
Change Password	Email ID=”” Old Password=”” New Password=””	Enter your Correct Email ID’s and update your Password.	Enter your Correct Email ID’s and update your Password.
	Email ID=”user@gmail.com” Old Password=”1234” New Password=”user”	Your passwords are not Matching!	Your passwords are not Matching!

	Email ID=”user@gma.com” Old Password=”1234” New Password=”user”	Invalid Email Id Entered!	Invalid Email Id Entered!
	Email ID=”user@gmail.com” Old Password=”user1” New Password=”user1”	Successfully Updated Password!	Successfully Updated Password!
Forgot Password	Email ID=”” (A link will be sent to the particular Email ID based on the users Unique Token)	Enter your correct Email ID	Enter your correct Email ID
	New Password=””	Enter Your new Password	Enter Your new Password
	New Password=”user123”	Password Updated Successfully!	Password Updated Successfully!

Admin: Insert Recipe	A Form tag which includes the 1. Title=”” 2. Ingredients=”” 3. Methods/Step=”” 4. Description Recipe=”” 5. Category=”” 6. Servings=”” 7. Timings=”” 8. Image=””	Enter all the Fields Correctly!!	Enter all the Fields Correctly!!
	1. Title=”soup” 2. Ingredients=” • Salt • Pepper • Veggies” 3. Methods/Step=” • Add salt • Add pepper • Add veggies” 4. Description Recipe=”very easy and tasty” 5. Category=”Vegetarian” 6. Servings=”2” 7. Timings=”10mins” 8. Image=”image”	Recipe Added successfully!!	Recipe Added successfully!!

Admin: Delete Recipe	Delete Recipe?	YES/NO	YES/NO
	Delete Recipe?	YES	Recipe deleted!
Admin: View Recipe	View Recipe	Redirects to the Recipe Page.	Redirects to the Recipe Page.
User: Subscription	Email ID=””	Add Email ID	Add Email ID
	Email ID=”user@gmail.com”	Successfully Subscribed!	Successfully Subscribed!
User: Save Recipe	Save Recipe	Login First	Login First
	Save Recipe	Recipe Saved!	Recipe Saved!
User: Create Menu	Choose a Day 1. Monday 2. Tuesday…	Login First	Login First

	Choose a Day 1. Monday 2. Tuesday…	Select a Day. Recipe added. Menu Create	Select a Day. Recipe added. Menu Create
User: Create Pantry	Select Ingredients: • Salt • Pepper • Veggies	Login First	Login First
	Select Ingredients: • Salt • Pepper • Veggies	Ingredient Added in Pantry!	Ingredient Added in Pantry!

Figure: 14 Test Case Designs

5 IMPLEMENTATION AND TESTING

5.1 Implementation Approach

- Introduction

  - The implementation of 30MinRecipes starts with the Homepage that shows all the information and the attractive UI design. The first module implemented was the Login Page; you can also navigate to the Registration Page with just one click. Since I also have an Admin Module which is very important this had the task to Add Recipes. To implement this module firstly I made a database design and a table structure. For the Admin Module my database consists of 6 Main Tables which stores data of the Recipe added. After completing this Module the focus was given to the UI design where the Recipes are to be displayed, the user can view all the Recipes added by the Admin, they can also Save the Recipe, Create a Pantry of their own Ingredients, Create a Menu and also comment.

- Input and Output Design Implementation

    - To start with my Project Implementation Plan, firstly, I had to decide which language I will be coding my project in, what the database schema will look like, what dependencies are required, and what will be the frontend UI design. Once these Functionalities were decided I started with my Modules Implementation and I moved on to making the front-end design of my website application.

- Table Structures
  - This Activity Shows the Tables that are used in my Project, and their Data Types with their Size and Constrains.

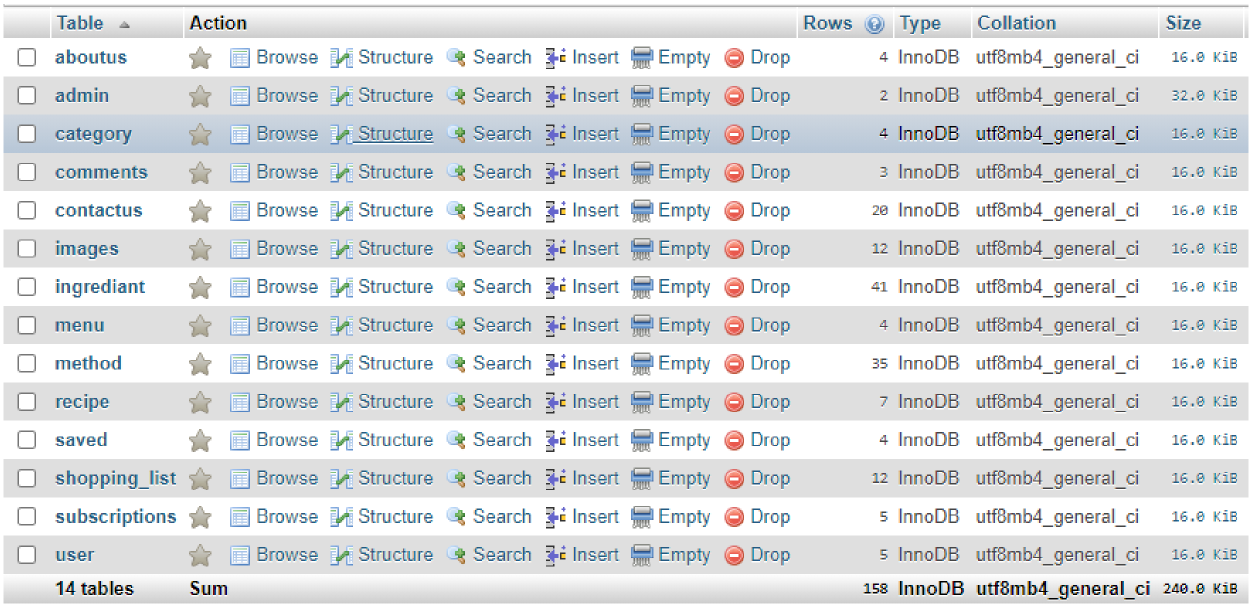
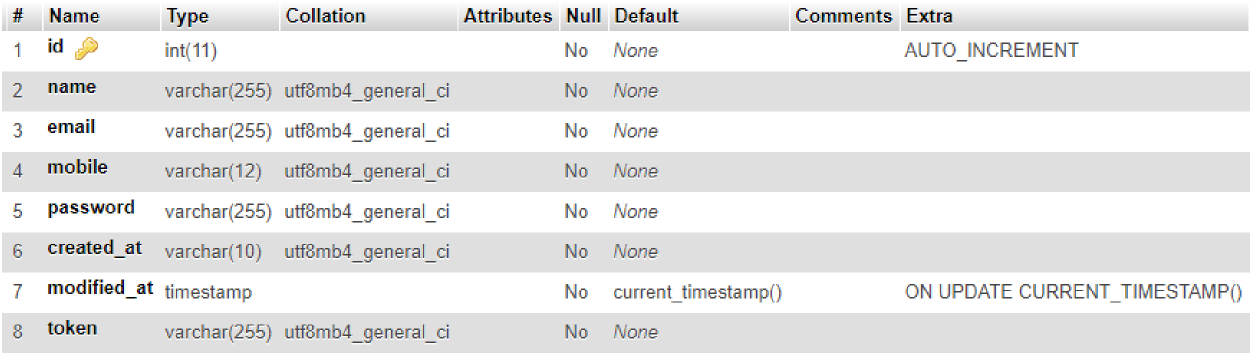


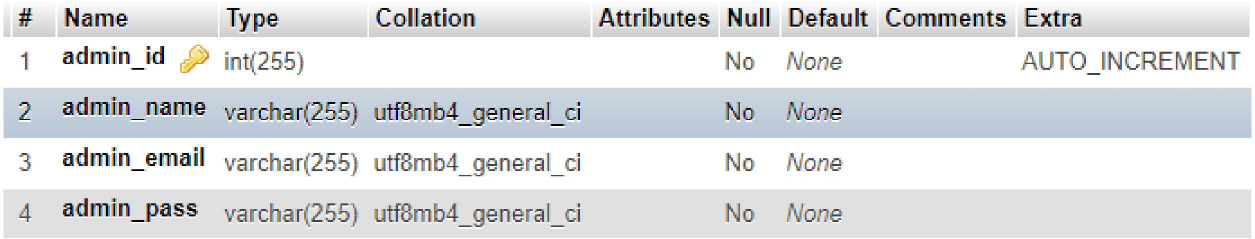
	Table ▲	Action	Rows	Type	Collation	Size
□	aboutus	☆ Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_general_ci	16.0 KiB
□	admin	☆ Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	32.0 KiB
□	category	☆ Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_general_ci	16.0 KiB
□	comments	☆ Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_general_ci	16.0 KiB
□	contactus	☆ Browse Structure Search Insert Empty Drop	20	InnoDB	utf8mb4_general_ci	16.0 KiB
□	images	☆ Browse Structure Search Insert Empty Drop	12	InnoDB	utf8mb4_general_ci	16.0 KiB
□	ingrediant	☆ Browse Structure Search Insert Empty Drop	41	InnoDB	utf8mb4_general_ci	16.0 KiB
□	menu	☆ Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_general_ci	16.0 KiB
□	method	☆ Browse Structure Search Insert Empty Drop	35	InnoDB	utf8mb4_general_ci	16.0 KiB
□	recipe	☆ Browse Structure Search Insert Empty Drop	7	InnoDB	utf8mb4_general_ci	16.0 KiB
□	saved	☆ Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_general_ci	16.0 KiB
□	shopping_list	☆ Browse Structure Search Insert Empty Drop	12	InnoDB	utf8mb4_general_ci	16.0 KiB
□	subscriptions	☆ Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_general_ci	16.0 KiB
□	user	☆ Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_general_ci	16.0 KiB
	14 tables	Sum	158	InnoDB	utf8mb4_general_ci	240.0 KiB

- **User Table:**



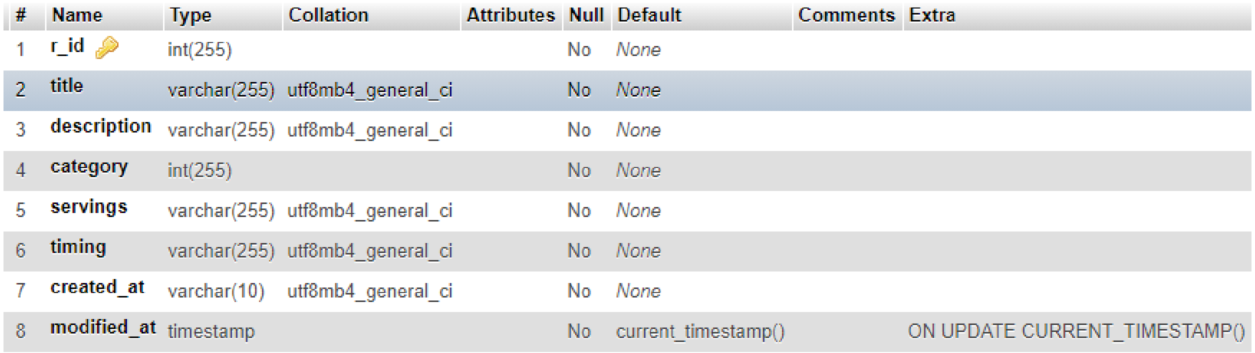
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	*None*		AUTO_INCREMENT
2	name	varchar(255)	utf8mb4_general_ci		No	*None*		
3	email	varchar(255)	utf8mb4_general_ci		No	*None*		
4	mobile	varchar(12)	utf8mb4_general_ci		No	*None*		
5	password	varchar(255)	utf8mb4_general_ci		No	*None*		
6	created_at	varchar(10)	utf8mb4_general_ci		No	*None*		
7	modified_at	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()
8	token	varchar(255)	utf8mb4_general_ci		No	*None*		

- ### Admin Table



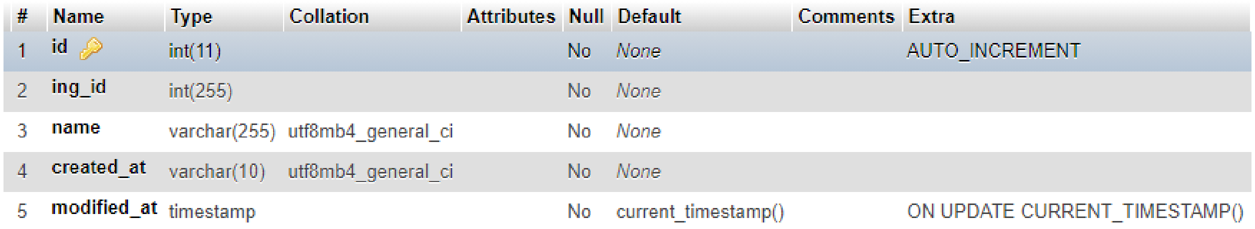
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**admin_id** 🔑	int(255)			No	*None*		AUTO_INCREMENT
2	**admin_name**	varchar(255)	utf8mb4_general_ci		No	*None*		
3	**admin_email**	varchar(255)	utf8mb4_general_ci		No	*None*		
4	**admin_pass**	varchar(255)	utf8mb4_general_ci		No	*None*		

- ### Recipe Table



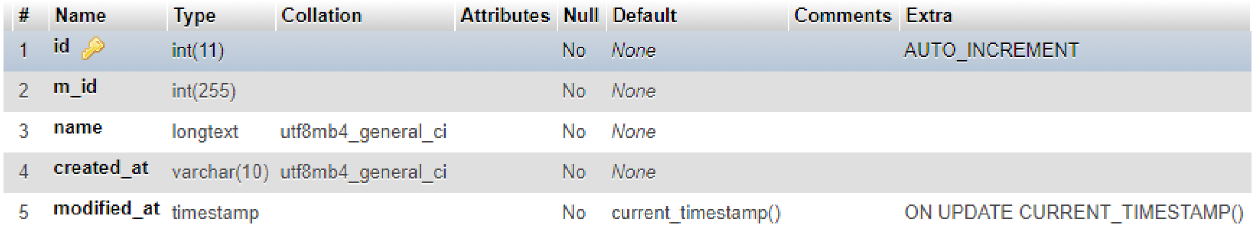
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**r_id** 🔑	int(255)			No	*None*		
2	**title**	varchar(255)	utf8mb4_general_ci		No	*None*		
3	**description**	varchar(255)	utf8mb4_general_ci		No	*None*		
4	**category**	int(255)			No	*None*		
5	**servings**	varchar(255)	utf8mb4_general_ci		No	*None*		
6	**timing**	varchar(255)	utf8mb4_general_ci		No	*None*		
7	**created_at**	varchar(10)	utf8mb4_general_ci		No	*None*		
8	**modified_at**	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()

- ### Ingredients Table



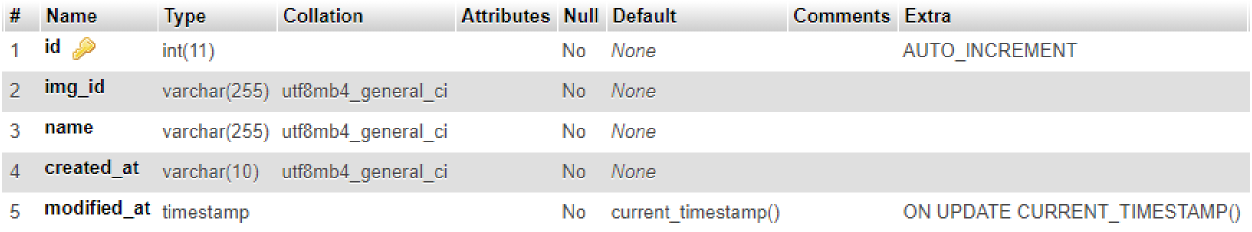
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**id** 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	**ing_id**	int(255)			No	*None*		
3	**name**	varchar(255)	utf8mb4_general_ci		No	*None*		
4	**created_at**	varchar(10)	utf8mb4_general_ci		No	*None*		
5	**modified_at**	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()

○ Method Table



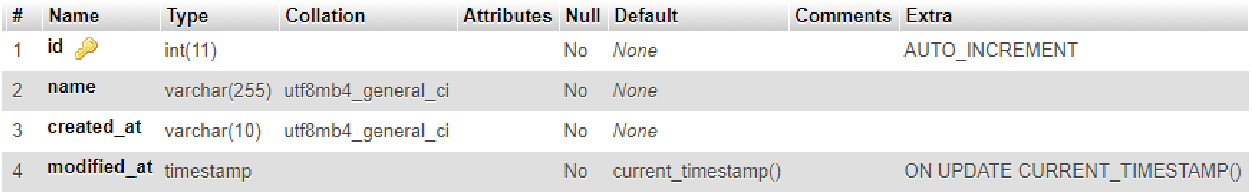
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	m_id	int(255)			No	*None*		
3	name	longtext	utf8mb4_general_ci		No	*None*		
4	created_at	varchar(10)	utf8mb4_general_ci		No	*None*		
5	modified_at	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()

○ Images Tables



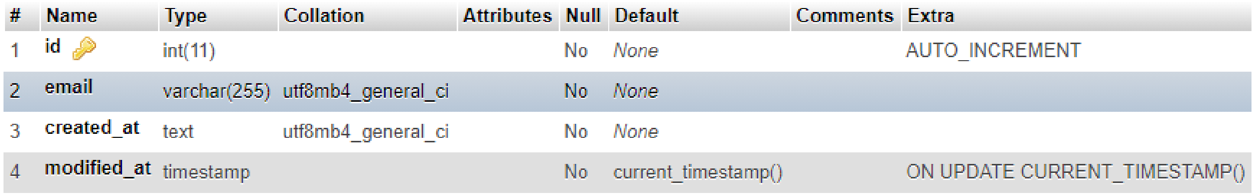
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	img_id	varchar(255)	utf8mb4_general_ci		No	*None*		
3	name	varchar(255)	utf8mb4_general_ci		No	*None*		
4	created_at	varchar(10)	utf8mb4_general_ci		No	*None*		
5	modified_at	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()

○ Category Table



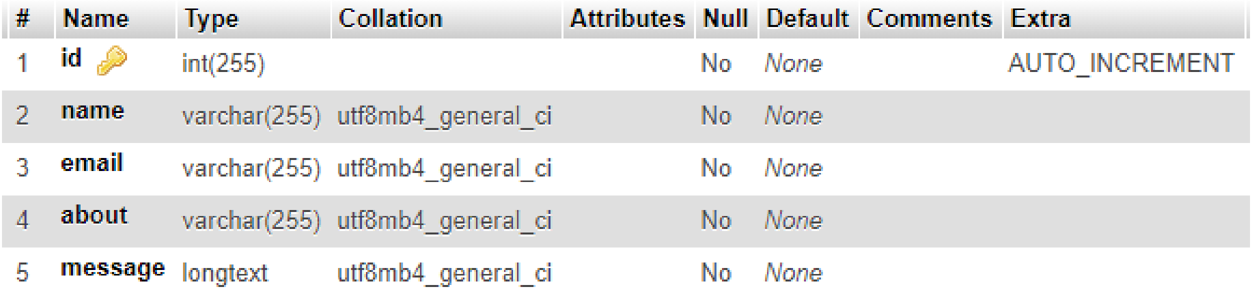
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	name	varchar(255)	utf8mb4_general_ci		No	*None*		
3	created_at	varchar(10)	utf8mb4_general_ci		No	*None*		
4	modified_at	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()

○ Subscriptions Table



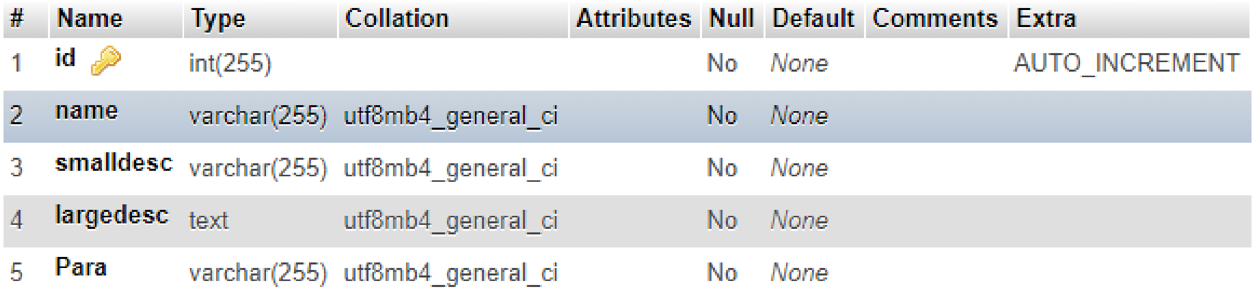
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	email	varchar(255)	utf8mb4_general_ci		No	*None*		
3	created_at	text	utf8mb4_general_ci		No	*None*		
4	modified_at	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()

- ### Contactus Table



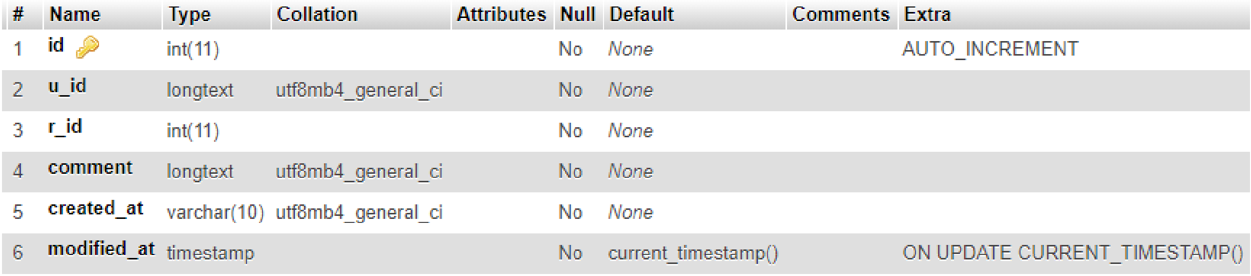
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🔑	int(255)			No	*None*		AUTO_INCREMENT
2	name	varchar(255)	utf8mb4_general_ci		No	*None*		
3	email	varchar(255)	utf8mb4_general_ci		No	*None*		
4	about	varchar(255)	utf8mb4_general_ci		No	*None*		
5	message	longtext	utf8mb4_general_ci		No	*None*		

- ### Aboutus Table



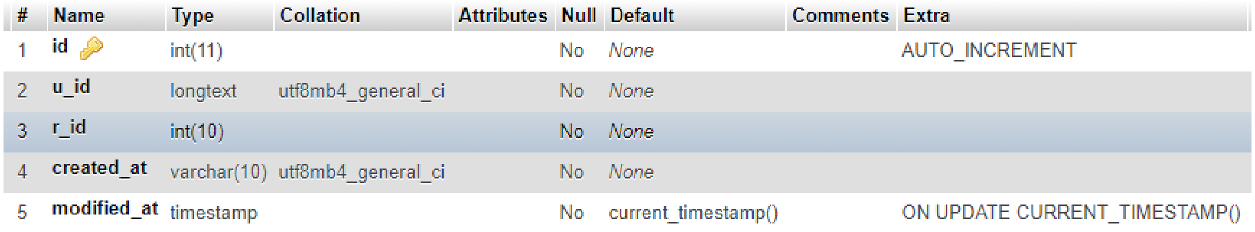
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🔑	int(255)			No	*None*		AUTO_INCREMENT
2	name	varchar(255)	utf8mb4_general_ci		No	*None*		
3	smalldesc	varchar(255)	utf8mb4_general_ci		No	*None*		
4	largedesc	text	utf8mb4_general_ci		No	*None*		
5	Para	varchar(255)	utf8mb4_general_ci		No	*None*		

- ### Comments Table



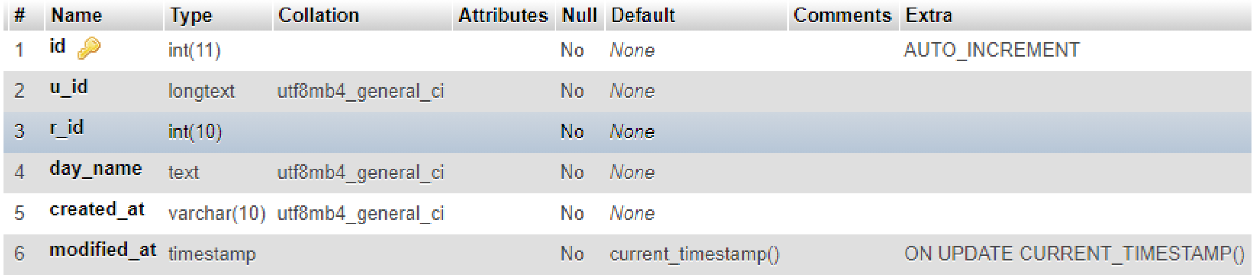
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	u_id	longtext	utf8mb4_general_ci		No	*None*		
3	r_id	int(11)			No	*None*		
4	comment	longtext	utf8mb4_general_ci		No	*None*		
5	created_at	varchar(10)	utf8mb4_general_ci		No	*None*		
6	modified_at	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()

- ### Saved Table



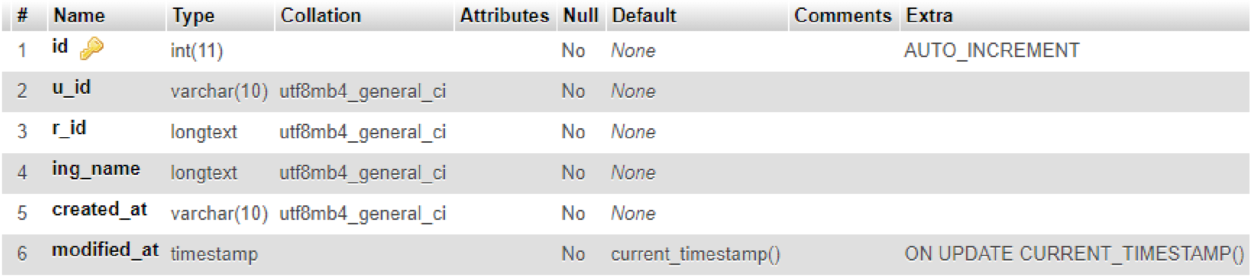
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	u_id	longtext	utf8mb4_general_ci		No	*None*		
3	r_id	int(10)			No	*None*		
4	created_at	varchar(10)	utf8mb4_general_ci		No	*None*		
5	modified_at	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()

- ### Menu Table



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	u_id	longtext	utf8mb4_general_ci		No	*None*		
3	r_id	int(10)			No	*None*		
4	day_name	text	utf8mb4_general_ci		No	*None*		
5	created_at	varchar(10)	utf8mb4_general_ci		No	*None*		
6	modified_at	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()

- ### Shopping_list Table



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	u_id	varchar(10)	utf8mb4_general_ci		No	*None*		
3	r_id	longtext	utf8mb4_general_ci		No	*None*		
4	ing_name	longtext	utf8mb4_general_ci		No	*None*		
5	created_at	varchar(10)	utf8mb4_general_ci		No	*None*		
6	modified_at	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()

- Code Module

  - This Activity shows that what Technologies I am using or my Front-End and Back-End and which Database. So, I am using HTML, CSS, JavaScript for Front-End and JSON for Back-End. I am using MySQL as my Database. In my Project, the User can Login or Register and then the Sessions will be created or Checked to authenticate the user and then the Main Functionalities of my project like Save, Pantry, Menu will work. At the User end a Web server will reside, the user will use a workstation and an internet connection to connect to my website then he/ she will authenticate/authorize themselves and then in the middle Business Server will fetch the Data requests and calls, and at the End tier Data Server, All the user data, saved, pantry, menu will be stored.

- System Implementation

  - My project consists of multiple modules. Firstly, the user can choose to login to the website or subscribe to it, If the user doesn't wish to do both they can still be able to view the recipes and make them at home! The admin has the privileges to add the recipes; the added recipes will be dynamically reflected in the categories in the main homepage of the user to view them. The admin also has the privilege to remove or directly view the recipe from the admin's dashboard. The user will be able to view healthy and wide categories of recipes, and also read about them to gain more knowledge. If the user wishes to add a recipe of their own they have to flexibility to mail the admin by inserting important details of the recipe and the user can add it to the website. A core function of this website is that the users can also Search for the recipes so that they do not have to scroll and search manually for it. The main core functionalities such as saving, creating a pantry, and creating a menu will be functional only when the user is has registered to the website. When the user has logged in to the system, they will be redirected to their homepage and they can now save the recipe, create a pantry or a menu and also comment on the recipes they like. Whatever activity the user has performed such as the number of recipes saved, the pantry added, and the menus created will be stored and the user can view them anytime and also edit them according to their choice. The users can logout of the system.

- Project Summary

  - When the user opens this website they are provided with an attractive and user friendly interface. The user can view the multiple categories of recipe provided and also read about them in a separate section if they have any doubts. The user can also mail the admin and ask them to add a recipe of their choice or what they want. They can also subscribe to the website. The core admin who is responsible for adding the recipes and also viewing or deleting them, first have to Login to the system and will be redirected to the admin dashboard. The admin can insert the recipes which will be dynamically reflected in the main user side and simultaneously the user can view them. The user can register to the system and use the features provided by the website. After logging in to the system to the user can view the recipe and also add them to favorites or save them to open it later, the user can also create a pantry of their own by selecting the ingredients they want to save and later access them by just a single click, this will help them for grocery shopping or to give a quick view of the ingredients. The use can also create a Menu, a menu is a feature in which the user can select any day of the week and add the recipe to that day. This feature will help the user to quickly prepare meals for the day they have saved the recipe on and not waste their time searching and thinking what to make that day. The user can also comment on the recipe and give their views and suggestions which will be visible to all the users who visit the website.

5.2 Coding Details and Code Efficiency

- **30MinRecipes Inserting_recipe.php**

```
       <script>
          $(document).ready(function(){

                  var ing_wrapper = $("#ingrediants_field");
                  var add_ing = $("#add_ingrediant");
                  var x = 1;
                  $(add_ing).click(function(e){
                      e.preventDefault();
                          $(ing_wrapper).append(

                     '<div class="form-row">'+
'<div class="col-md-10 form-group text-center">'+
 '<input type="text" class="form control"
placeholder="Ingrediants" name="ingrediants[]" required/>'+
                      '</div>'+
'<div class="col-md-2 form-group text-center">'+
'<button type="button" class="remove_ingrediant btn btn-
danger">Remove</button>'+
                    '</div>'+
                '</div>'
              );
           x++;
                  });

          $(ing_wrapper).on("click",".remove_ingrediant",
          function(e){
                      e.preventDefault();
          $(this).parent('div').parent('div').remove();
```

```
                      x--;
                  });

                  var steps_wrapper = $("#steps_field");
                  var add_step  = $("#add_step");
                  var y = 1;
                  $(add_step).click(function(e){
                   e.preventDefault();
                    $(steps_wrapper).append(
                     '<div class="form-row">'+
              '<div class="col-md-10 text-center form-group">'+
              '<input type="text" class="form-control"
          placeholder="Write step" name="steps[]" required/>'+
                        '</div>'+
            '<div class="col-md-2 form-group text-center">'+
           '<button type="button" class="remove_step btn btn-
          danger">Remove</button>'+
                            '</div>'+
                         '</div>'
                      );
                          y++;
                  });

$(steps_wrapper).on("click",".remove_step", function(e){
e.preventDefault();
$(this).parent('div').parent('div').remove();
                      y--;
                  });
        $("#addRecipe").on('submit', function(e){
              e.preventDefault();
                      $.ajax({
                          url : 'api/recipe.php',
                          type : 'POST',
                          data : new FormData(this),
                          contentType : false,
                          cache : false,
                          processData : false,
                          success : function(data){
                              swal({
                                      title: "Recipe Added!",
                        text: "New recipe added successfully!",
                                      icon: "success",
                                      button: "OK!",

                                  }).then((ok)=>{
                                      if(ok){
                             window.location ='allrecipes.php';
                                      }
                                  });
                          }
                      });
                  });

           });
<script>
```

- **30MinRecipes Recipe Page**

```
          <script>
            function getRecipe(){
                  $.ajax({
                      url : '../resources/api/recipe.php',
                      type : 'POST',
                      dataType : 'json',
                      contentType : 'application/json',
                      data : JSON.stringify({
                          call : 3,
                      }),
                      success : function(data){
                          console.log(data);
                          var info = data[0][0];
                          var image = data[1][0];
                          var ingredients = data[2];
                          var method = data[3];

                          var user = data[4];
                          var comArea = data[6];

                          var getRecipe = '';
                          var ingredientList = '';
                          var methodList = '';
                          var coms = '';
                          if(comArea.length>0){
                              $.each(comArea, function(i,d)
                              {
                                  coms+=
           '<div class="p-3" style="background-color:#ecf0f1;
          border-radius:10px">'+
                      '<h6>'+d.name+'</h6>'+
                       '<p>&nbsp&nbsp'+d.comment+'</p>'+
                                  '</div><br>';
                              });
                          }
                          else{
                              coms=
'<div class="p-3 text-center" style="background-color:#ecf0f1;
border-radius:10px">'+
 '<p>No comments available</p>'+
                     '</div>';
               }

      $.each(ingredients, function(i,d){
             i++;
              ingredientList+=
 '<input type="checkbox" id="child_chkbox"
name="ingrediantsList" value="'+d.name+'">&nbsp&nbsp'+
'<label style="font-size:18px" class="py-
1">'+d.name+'</label><br>';
               });

               $.each(method, function(i,d){
                   i++;
 methodList+='<p style="font-size:18px"><b>Step
'+i+'.</b><br>'+d.name+'</p><hr>';
               });

               getRecipe+=
                   '<div class="row py-3">'+
                       '<div class="col-md-12">'+
  '<form id="ingForm" enctype="multipart/form-data">'+
   '<div class="form-row py-4">'+
   '<div class="form-group col-md-7 px-4">'+
'<h1>'+info.title+'</h1><br>'+
'<h4>Description</h4>'+
 '<p style="font-size:18px">'+info.description+'</p>'+
'<p style="font-size:18px"><b>Servings :</b>
'+info.servings+' &nbsp &nbsp<b>Timing :</b>
'+info.timing+'</p>'+
      '<button class="btn btn-rimary btn-outline-primary"
     type="button" onclick="saveRecipe()">Save Recipe<i
     class="fa  fa-shopping-cart"></i></button>'+
         '</div>'+



     '<div class="form-group col-md-5 py-4 text-center">'+
     '<img src="../uploads/'+image.name+'" height="300"
width="400" style="border-radius:10px;"> '+
        '</div>'+
     '</div>’
 '<div class="form-row py-2">'+
'<div class="form-group col-md-5 px-4">'+
 '<h3>Ingredients</h3><br>'+

     ingredientList+
     '<input type="checkbox" id="parent"
     onclick="checkall()">&nbsp&nbsp'+
 '<label style="font-size:18px" class="py-1">'+

     '<strong>Select all</strong>'+
      '<i class="fa fa-check"></i>'+
     '</label><br><br>'+
     '<button class="btn btn-uccess btn-outline-primary"
     type="button" onclick="addIngrediants()">Add to
     Pantry'+
      ' <i class="fa fa-shopping-cart"></i>'+

     '</button><br><br>'+
      '<div class="input-group mb-3">'+
       '<select class="custom-select" id="day">'+
      '<option value="Monday" selected>Monday</option>'+
      '<option value="Tuesday">Tuesday</option>'+
      '<option value="Wednesday">Wednesday</option>'+
      '<option value="Thursday">Thursday</option>'+
      '<option value="Friday">Friday</option>'+
      '<option value="Saturday">Saturday</option>'+
      '<option value="Sunday">Sunday</option>'+

     '</select>'+

      '<div class="input-group-append">'+
      '<button class="btn btn-success" type="button"
     onclick="addToMenu()" for="day">Add to Menu <i
     class="fa fa-book"></i></button>'+
      '</div>'+
     '</div>'+
     '</div>'+
      '<div class="form-group col-md-7 px-3">'+

     '<h3>Method</h3><br>'+
             methodList+
          '</div>'+
     '</div>'+
      '</form>'+
     '</div>'+
     '</div>';

       $("#getRecipe").html(getRecipe);
     $("#commentSection").html(coms);
     $("#ccount").html(comArea.length);
     }
             });
         }
```

```
</script>
```

30MinRecipes Saving Recipe

```
<script>
     function saveRecipe(){
        $.ajax({
            url : '../resources/api/recipe.php',
            type : 'POST',
            dataType : 'json',
            contentType : 'application/json',
            data : JSON.stringify({
                call : 6
            }),
            success : function(data){
                if(data==0){
                    swal({
                        title: "Login required!",
      text: "You must have to login to save recipe!",
                        icon: "info",
                        button: "OK!",
                    });
                }
                else if(data==3){
                    swal({
                        title: "Already saved!",
                     text: "Recipe is already saved!",
                        icon: "warning",
                        button: "OK!",
                    });
                }

                else{
                    getRecipe();      //optional
                    getNavLinks();
                        swal({
                            title: "Saved!",
                            text: "Recipe saved!",
                            icon: "success",
                            button: "OK!",
                        });
                }
            }
        });
    }
</script>
```

- **30MinRecipes Menu Creation**

```
<script>
     function addToMenu(){

        var day = $("#day").val
            $.ajax({
                url : '../resources/api/menu.php',
                type : 'POST',
                dataType : 'json',
                contentType : 'application/json',
                data : JSON.stringify({
                    day : day,
                    call : 1
                }),
                success : function(data){
                    if(data==0){
                        swal({
                            title: "Login required!",
  text: "You must have to login to add item in menu!",
                            icon: "info",
                            button: "OK!",
                        });
                    }
                    else if(data==2){
                        swal({
title: 'Already recipe selected for '+day+'. Do you
want to replace it by this one?',
                            icon: "warning",
                            buttons: ['No', 'Yes'],
                            dangerMode: true,
                            })
                            .then((done) => {
                            if (done) {
                                $.ajax({
                 url : '../resources/api/menu.php',
                 type : 'POST',
                 dataType : 'json',
                 contentType : 'application/json',
                 data : JSON.stringify({
                       day : day,
                       call : 2
                     }),
              success : function(data){
                   if(data==1){

getRecipe();
   swal({
        title: "Recipe replaced!",
        text: "Recipe is replaced in Menu for "+day,
        icon: "success",
        button: "OK!",
       });
     }
         else{
             alert("error!");
            }
           }
         });
      }
else {
       swal("Think again and decide!");
                            }
                        });
                    }
                    else{
                        getRecipe();
                        getNavLinks();
                        swal({
                         title: "Added to Menu!",
            text: "Recipe is added in Menu for "+day,
                            icon: "success",
                            button: "OK!",
                        });
                    }
                }
        });
    }
</script>
```

- **30MinRecipes Pantry**

```
<script>
     function addIngrediants(){

        var ingrediantsArray = [];

$.each($("input[name='ingrediantsList']:checked"),
function(){
            ingrediantsArray.push($(this).val());
        });
        if(ingrediantsArray==''){
            swal({
                title: "No ingrediant selected!",
                text: "At least one ingrediant is
required!",
                icon: "warning",
                button: "OK!",
            });
        }
        else{
            $.ajax({
                url : '../resources/api/recipe.php',
                type : 'POST',
                dataType : 'json',
                contentType : 'application/json',
                data : JSON.stringify({
               ingrediantsArray : ingrediantsArray,
                    call : 5
                }),
                success : function(data){
                    if(data==0){
                        swal({
```

```
                                      title: "Login required!",
text: "You must have to login to save ingrediants list!",
                                      icon: "info",
                                      button: "OK!",
                                  });
                              }
                              else if(data==2){
                                  swal({
                            title: "Some error occured!",
                            text: "There is some issue!",
                                      icon: "error",
                                      button: "OK!",
                                  });
                              }
                              else{
                                  getRecipe();
                                  getNavLinks();
                                  swal({
                                      title: "Added to cart!",
              text: "Your shopping list is added in cart!",
                                      icon: "success",
                                      button: "OK!",
                                  });
                              }
                          }
                  });
                  }
              }
          </script>
```

5.2.1 Code Efficiency

HTML is a user friendly and provides a very attractive UI design. The backend is data is passed by using Ajax and JQuery with JON objects to the database and getting the responses which doesn’t require refreshing the entire page. PHP and Apache MySQL is used as the database and it is opened only when the user has logged in to the system.

5.3 Testing Approach

I have tested my project with Unit Testing, Integration Testing, System Testing, Black Box Testing, White Box Testing, Regression Testing, etc. Regression Testing means when we do some changes after testing and then test those changes only which are made after testing is done that's called as Regression Testing. I have tested my each module separate that comes under Unit Testing and I have also Integrated my one module with another then test it that comes under integration Testing.

5.3.1 Unit Testing

Unit Testing is done by testing each module as a separate function. As an example the recipe page has multiple modules to it. So each module had to be tested e.g. Testing if the save button works, if adding comment works, is creating a menu working and when adding ingredients of the recipe is it being added individually in the database.

5.3.2 Integration Testing

Integration Testing means testing more than one Module at a time. Taking the same example, the comment module had to be checked if it worked together e.g. if saving and commenting on a particular recipe, reflected changes in the same recipe document in the database.

5.3.3 System Testing

System testing is conducted on a complete integrated system to evaluate the system's compliance with its specified requirements. System testing takes, as its input, all of the integrated components that have passed integration testing. The application has successfully passed the system testing as per the objectives defined and has been tested on multiple devices covering different browser platforms.

5.4 Modification and Improvements

After testing and handling all errors, exceptions and bugs, errors were solved by troubleshooting, exceptions were handled by try catch block at the correct places avoiding redundancy of code from index files. Bugs can never be removed completely but most of them are solved and remaining is under development.

6 RESULTS AND DISCUSSION

6.1 User Documentation

When the user visits this website, he/she are provided with the categories of recipes along with subscription. The user can view the various types of recipes and can also read about the meal types the system has provided along with its benefits. The users can also mail the admin asking them for any query they like such as adding a recipe of their choice. For saving, creating a pantry and creating a menu the user has to log in to the system first.

After logging into the system the user is redirected to the user's profile. The user can view multiple functionalities provided such as the number of recipes saved, the menu created for the days and also the ingredients that has been saved.

The admin is responsible for adding the recipes and also viewing and deleting them. When the admin adds a recipe it is dynamically reflected in the main page of the recipes according to the categories. Users can add comments on them. The user can use the search bar to search for other recipes, which will redirect them to their page which will only consist of the recipe that has been searched. The users can then logout of the system.

7 CONCLUSIONS AND FUTURE WORK

7.1 Conclusion

I have done my project according to the objectives of my project. I have Firstly defined my all the modules and then one by one I have completed them. So, In My Project Mainly I am giving the best Interface to users to read recipes and the admin adds easy quick recipes for Knowledgeable or Non-Knowledgeable Users. And Even I am providing the Facility to Users that they can save recipes, create menus, add ingredients to pantry.

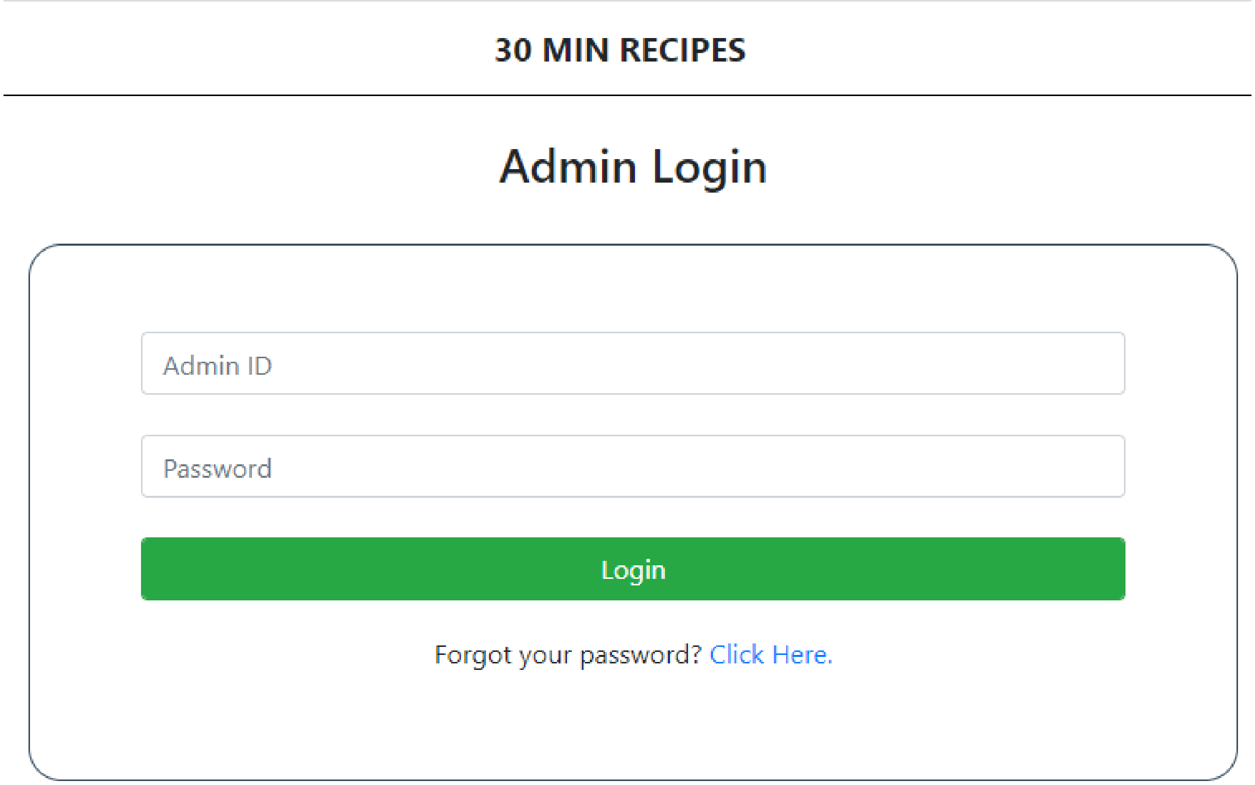
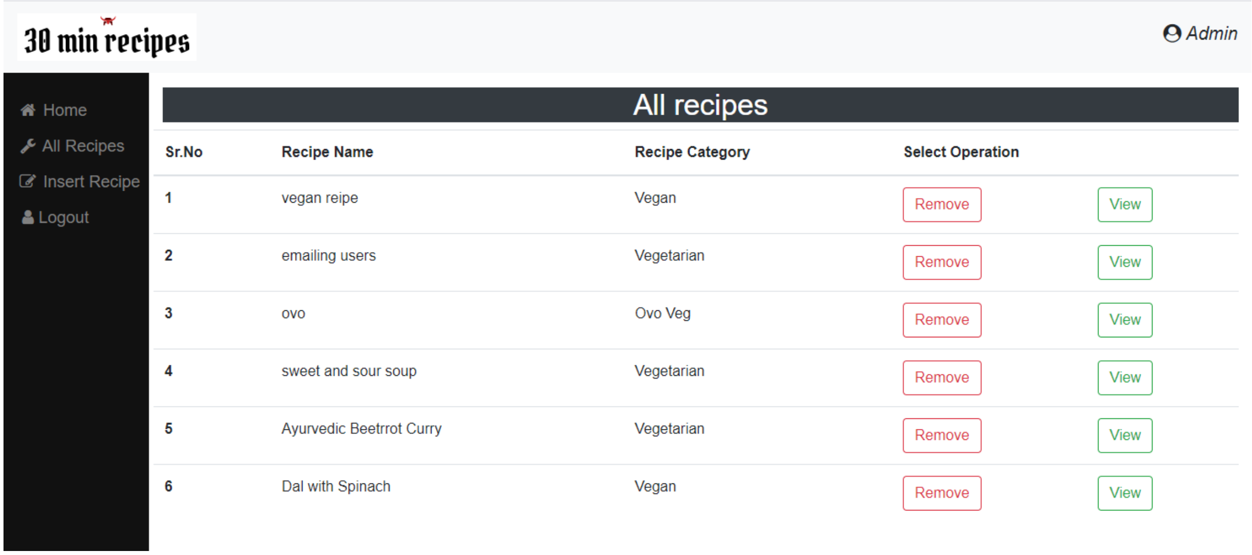


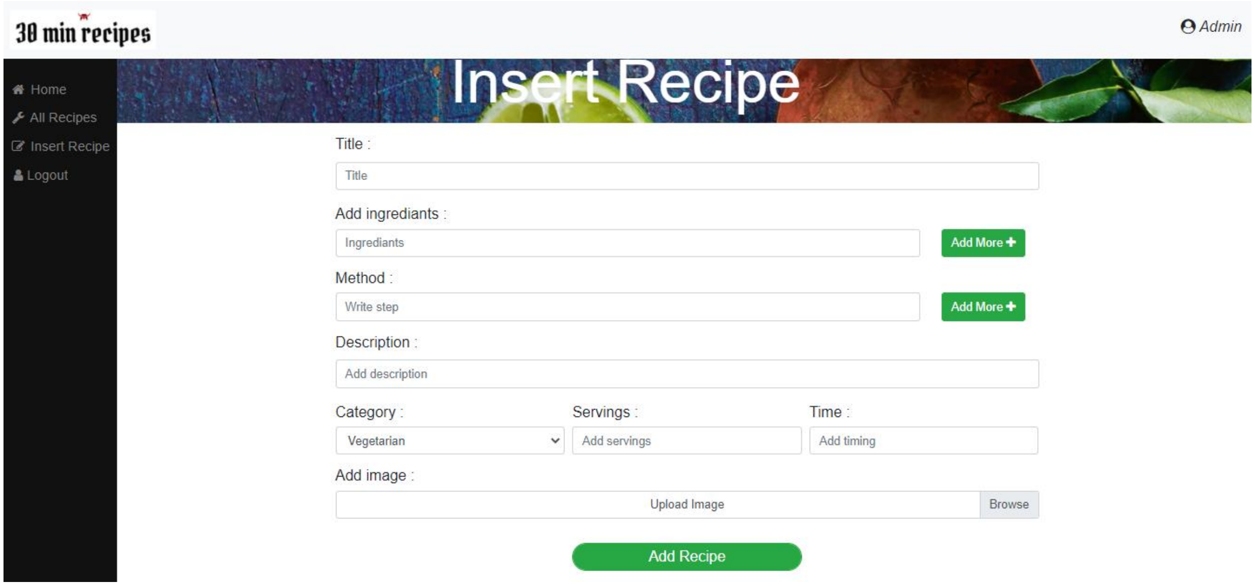
Figure: 15 Admin Login



All recipes

Sr.No	Recipe Name	Recipe Category	Select Operation	
1	vegan reipe	Vegan	Remove	View
2	emailing users	Vegetarian	Remove	View
3	ovo	Ovo Veg	Remove	View
4	sweet and sour soup	Vegetarian	Remove	View
5	Ayurvedic Beetrot Curry	Vegetarian	Remove	View
6	Dal with Spinach	Vegan	Remove	View

Figure: 16 All Recipes Page



Insert Recipe

Title : Title

Add ingrediants : Ingrediants

Method : Write step

Description : Add description

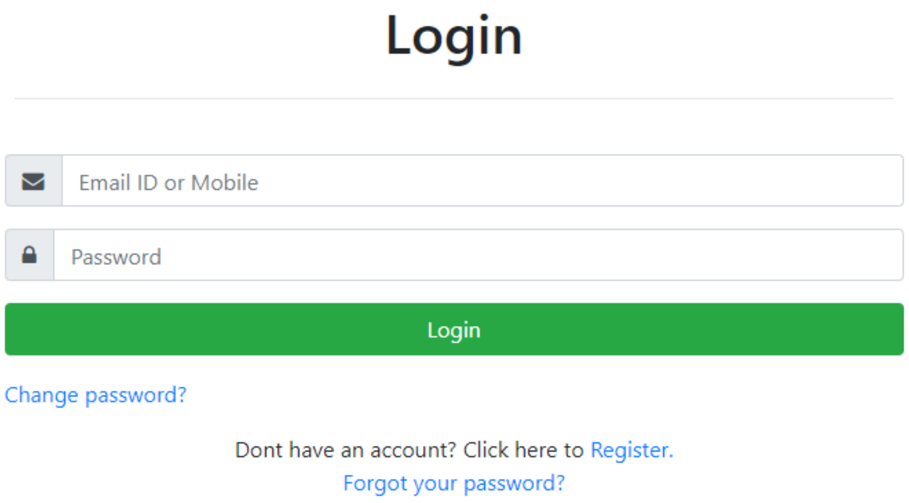
Category : Vegetarian

Servings : Add servings

Time : Add timing

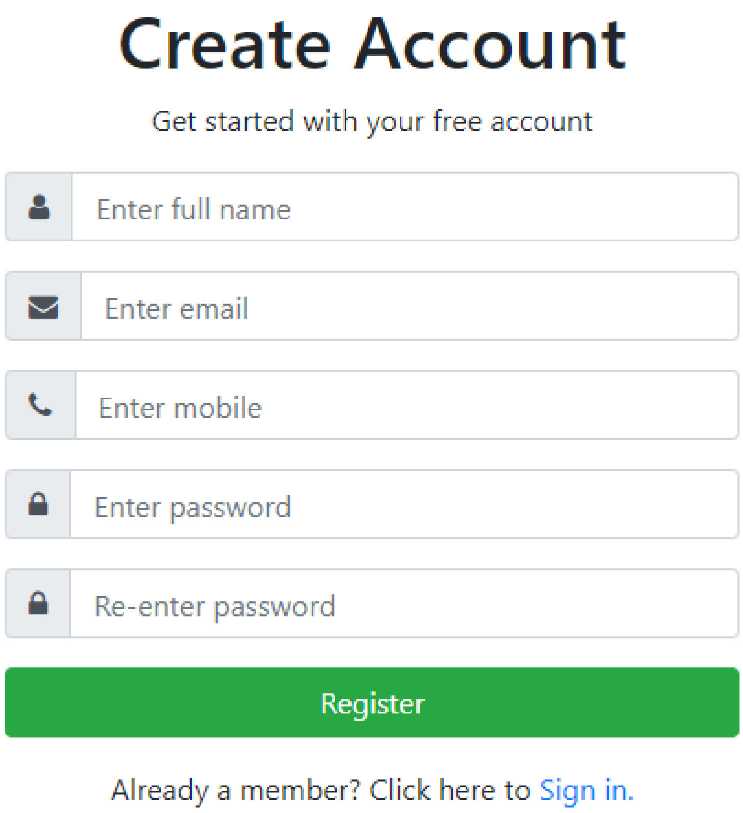
Add image : Upload Image

Figure: 17 Add Recipe Page


Login

Dont have an account? Click here to Register.
Forgot your password?

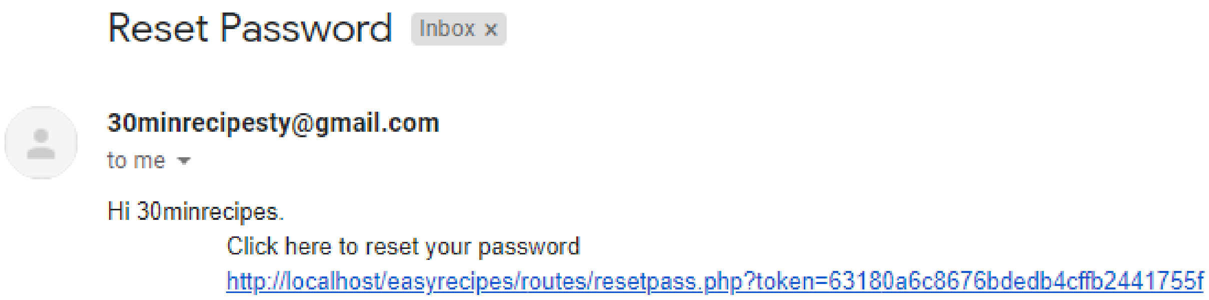
Figure: 18 Login Page


Create Account

Get started with your free account

Already a member? Click here to Sign in.

Figure: 19 Registration Page



Reset Password

30minrecipesty@gmail.com

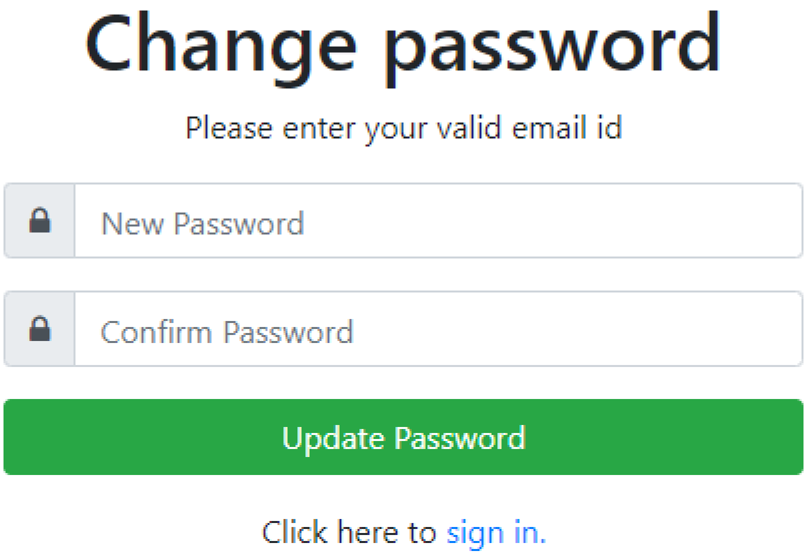
to me

Hi 30minrecipes.

Click here to reset your password

http://localhost/easyrecipes/routes/resetpass.php?token=63180a6c8676bdedb4cffb2441755f

Figure: 20 Forgot Password



Change password

Please enter your valid email id

Click here to sign in.

Figure: 21 Change Password

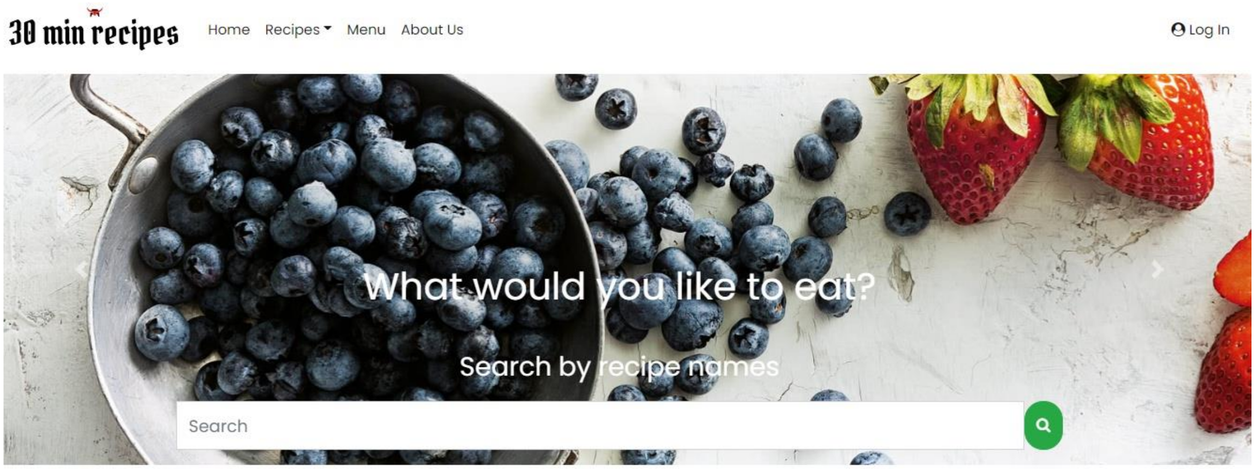


Figure: 22 Homepage

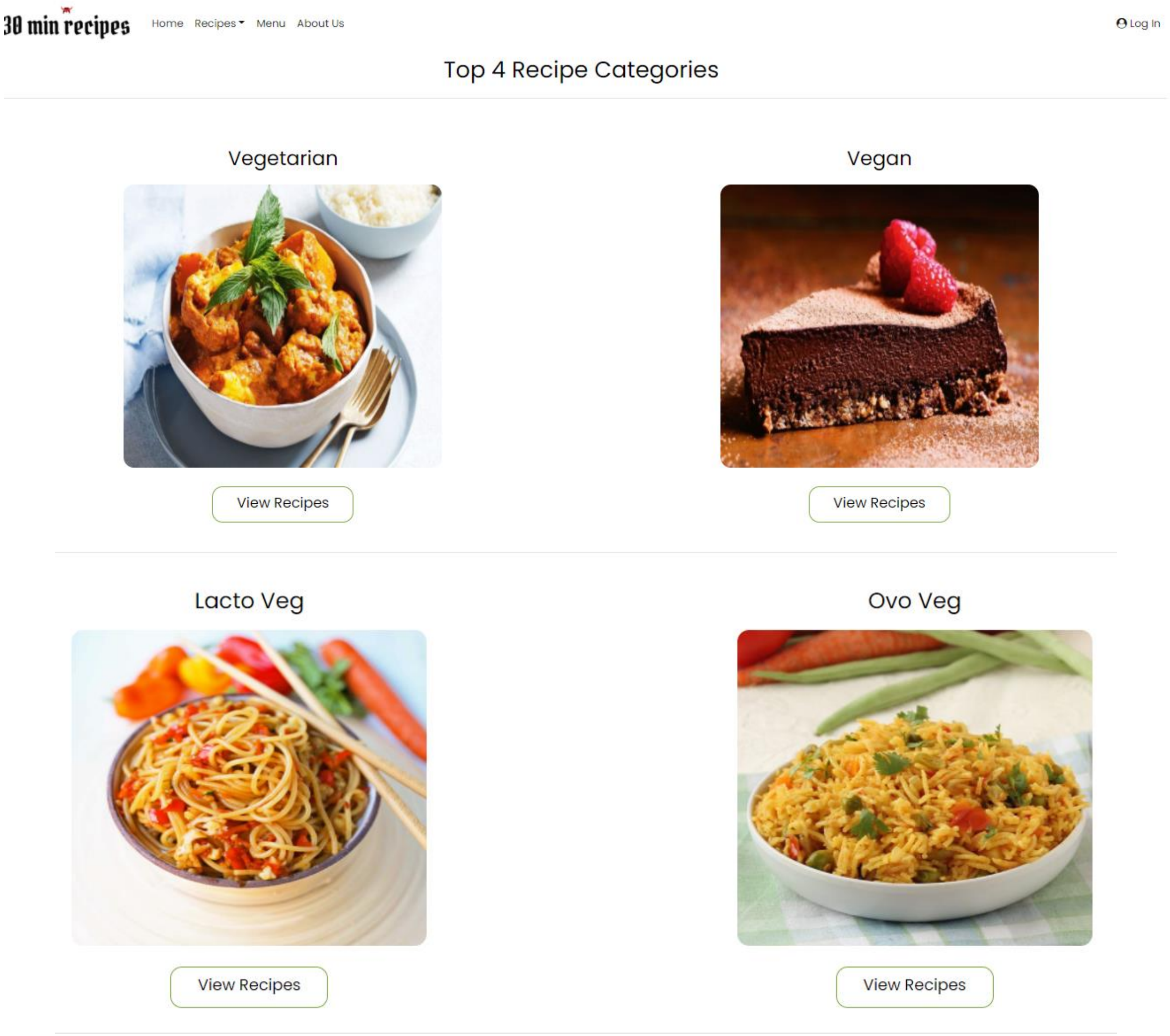


Figure: 23 Recipe Categories

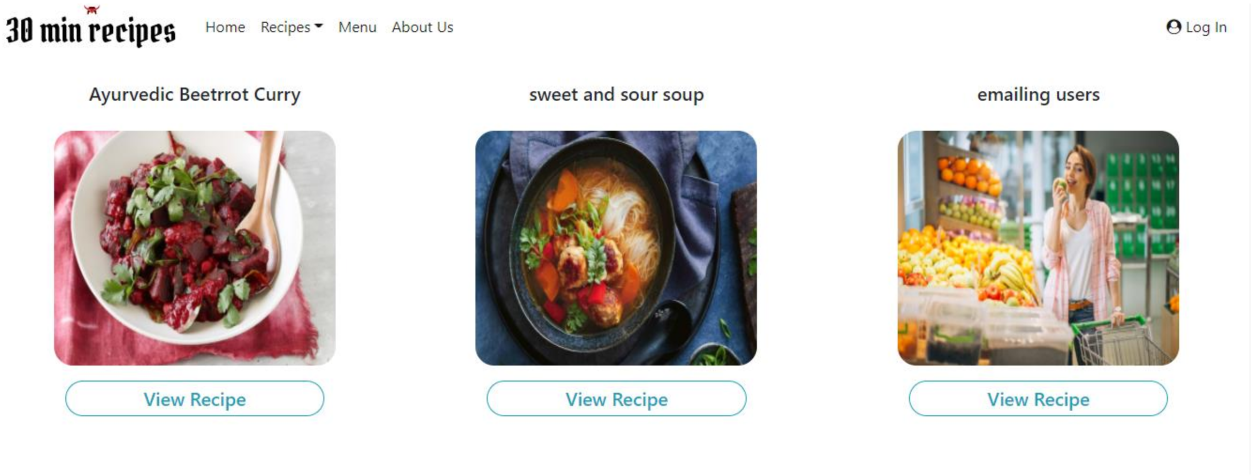
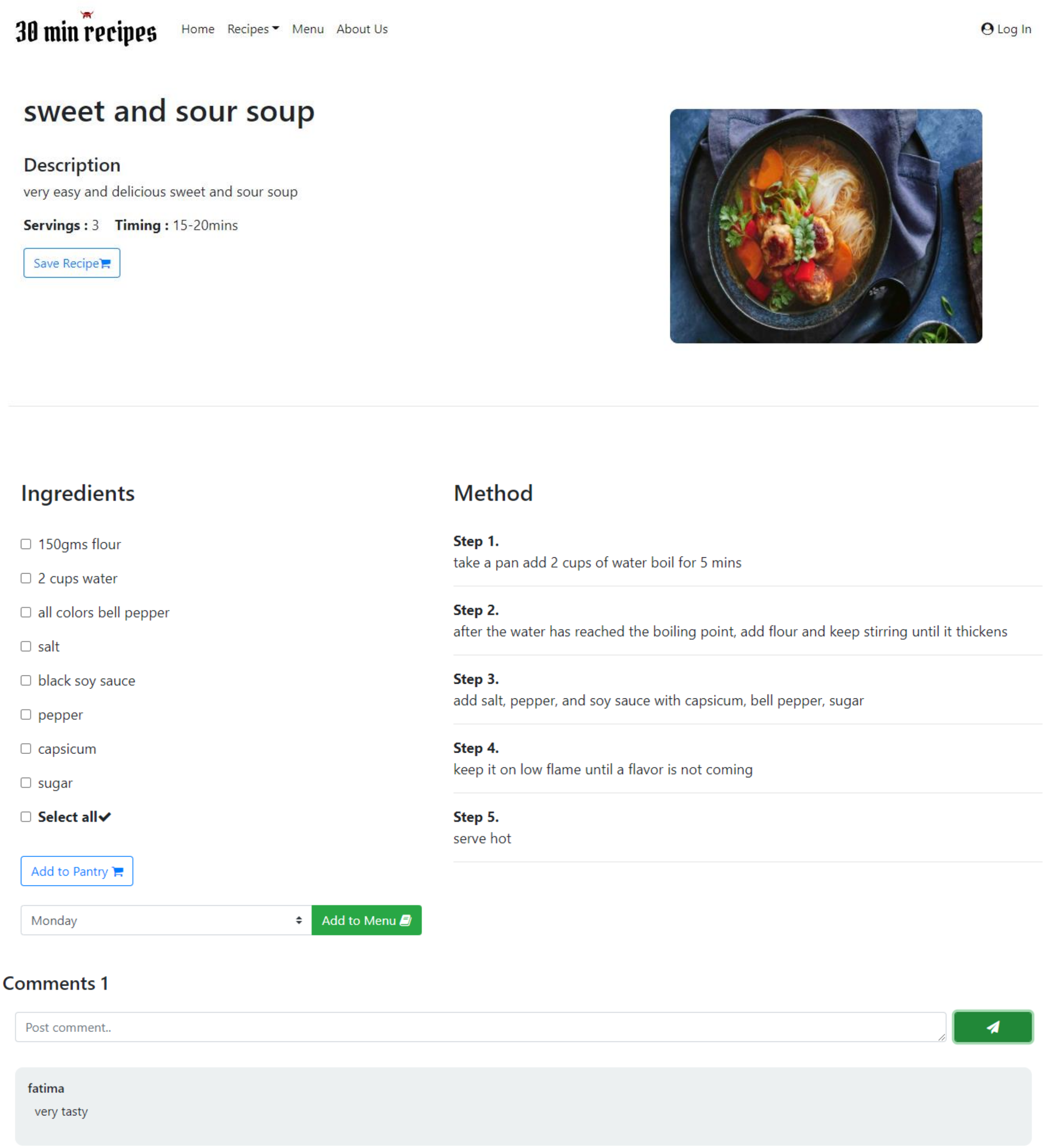


Figure: 24 Categories



sweet and sour soup

Description

very easy and delicious sweet and sour soup

Servings : 3 **Timing :** 15-20mins

Ingredients

- □ 150gms flour
- □ 2 cups water
- □ all colors bell pepper
- □ salt
- □ black soy sauce
- □ pepper
- □ capsicum
- □ sugar
- □ **Select all✓**

Method

Step 1.
take a pan add 2 cups of water boil for 5 mins

Step 2.
after the water has reached the boiling point, add flour and keep stirring until it thickens

Step 3.
add salt, pepper, and soy sauce with capsicum, bell pepper, sugar

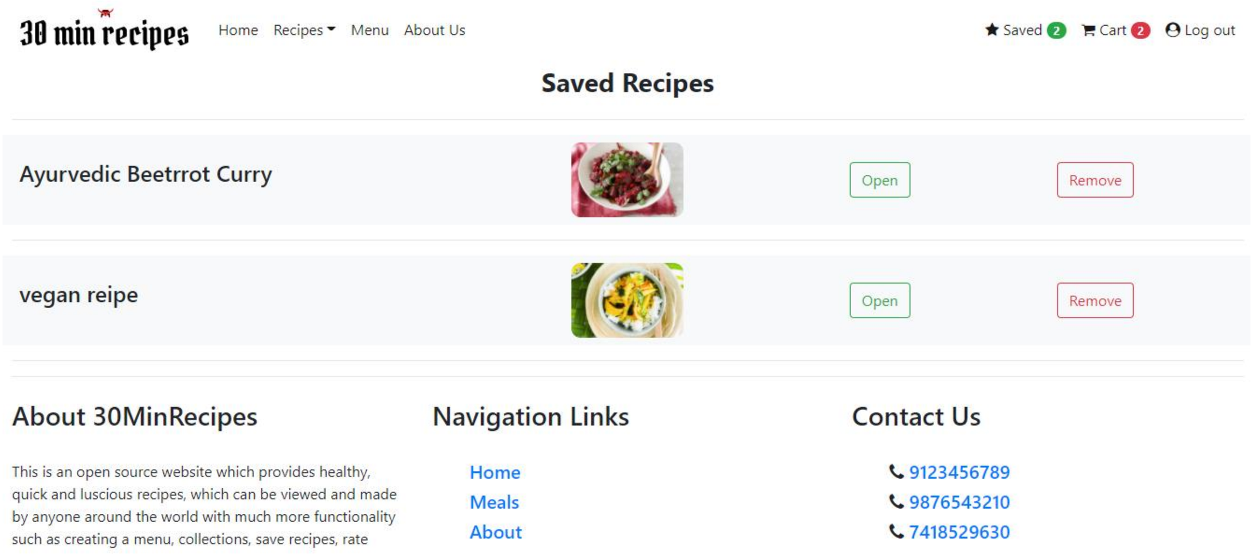
Step 4.
keep it on low flame until a flavor is not coming

Step 5.
serve hot

Comments 1

fatima
very tasty

Figure: 25 Recipe Page



Saved Recipes

Ayurvedic Beetrrot Curry

vegan reipe

About 30MinRecipes

This is an open source website which provides healthy, quick and luscious recipes, which can be viewed and made by anyone around the world with much more functionality such as creating a menu, collections, save recipes, rate

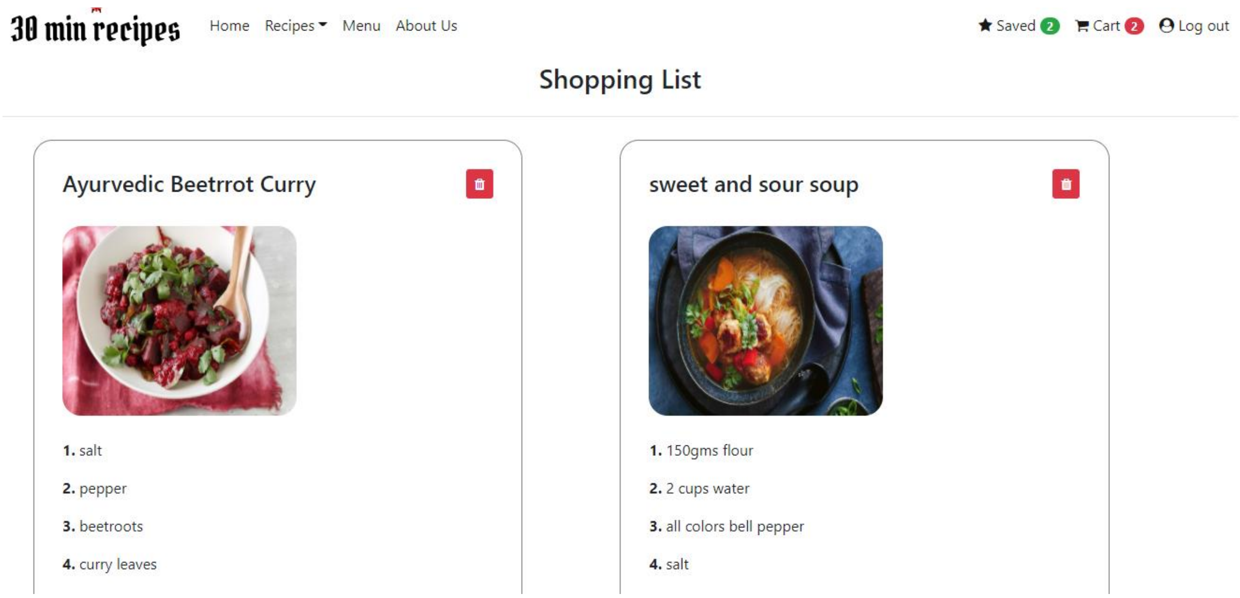
Navigation Links

- Home
- Meals
- About

Contact Us

- ☎ 9123456789
- ☎ 9876543210
- ☎ 7418529630

Figure: 26 Saved items



Shopping List

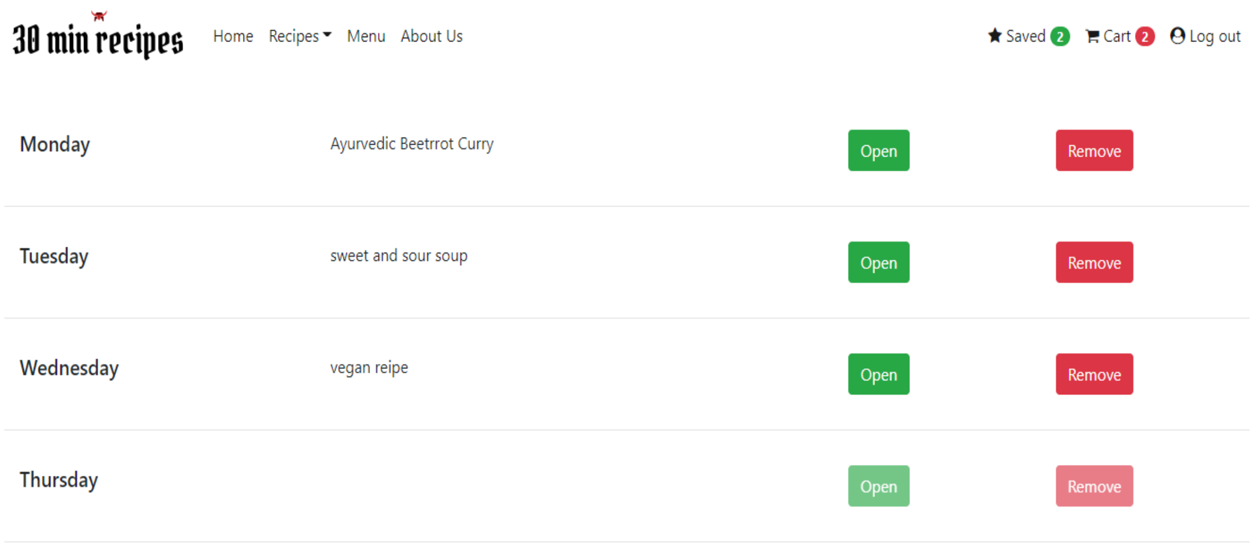
Ayurvedic Beetrrot Curry

1. salt
2. pepper
3. beetroots
4. curry leaves

sweet and sour soup

1. 150gms flour
2. 2 cups water
3. all colors bell pepper
4. salt

Figure: 27 Pantry



Monday	Ayurvedic Beetrrot Curry	Open	Remove
Tuesday	sweet and sour soup	Open	Remove
Wednesday	vegan reipe	Open	Remove
Thursday		Open	Remove

Figure: 28 Menu Page

7.2 Limitations of the system

- Users cannot add recipes, it is only limited to the admin.
- Another Limitation is that there is no way to assure that the users won’t mail the admin and engage in acts of piracy.
- The users can only subscribe and no further updates will be given as of now.
- The admin cannot edit the recipes; either he has to delete the entire recipe.

7.3 Future Scope of the project

- First Future Enhancement will be to add videos in the recipe category section of the web application.
- Another Future Enhancement will be that the users can add recipes of their own choice instead of having to mail the admin.
- The admin can update the recipe if there’s a mistake instead of deleting.
- Another Future Enhancement is that all the users who have subscribed to the website will get updates whenever a new recipe will be added by the admin.

REFERENCES

- Thapa Technical. (2018, June 12). *JSON in One Video in Hindi 2018* [Video]. Youtube. https://www.youtube.com/watch?v=bMqevPKAPD4
- Thapa Technical. (2018a, March 2). *PHP TUTORIAL IN ONE VIDEO IN HINDI 2019* [Video]. Youtube. https://www.youtube.com/watch?v=3024Kc6SDJ0
- dcode. (2018, December 6). *Sending JSON with AJAX/XMLHttpRequest - JavaScript Tutoril* [Video]. Youtube. https://www.youtube.com/watch?v=mNrJDGfQGz0
- Yahoo Baba. (2020, April 15). *Read JSON Data with jQuery Tutorial in Hindi / Urdu* [Video]. Youtube. https://www.youtube.com/watch?v=l2Fjf7BzeWE
- (FreeLogoDesign: Logo Maker - Create Your Own Logo, It's Free!, n.d.)
- *W3Schools Online Web Tutorials*. (n.d.). Https://Www.W3schools.Com/. https://www.w3schools.com/
- *Stack Overflow - Where Developers Learn, Share, & Build Careers*. (n.d.). Stack Overflow. https://stackoverflow.com/
- *GitHub: Where the world builds software*. (n.d.). GitHub. https://github.com/
- *Online Courses - Learn Anything, On Your Schedule*. (n.d.). Udemy. https://www.udemy.com/
- *Online Courses - Learn Anything, On Your Schedule*. (n.d.-b). Udemy. https://www.udemy.com/course/php-for-complete-beginners-includes-msql-object-oriented/
- *JavaScript*. (n.d.). Bootsrap. https://getbootstrap.com/docs/4.0/getting-started/javascript/

- *Vegefoods*. (2020, August 6). Colorlib. https://colorlib.com/wp/template/vegefoods/
- *Online Courses - Learn Anything, On Your Schedule*. (n.d.-c). Udemy. https://www.udemy.com/course/php-for-beginners-how-to-build-an-ecommerce-store/
- *Online Courses - Learn Anything, On Your Schedule*. (n.d.-d). Udemy. https://www.udemy.com/course/complete-json-ajax-course/