ONLINE FOOD ORDERING SYSTEM

A PROJECT REPORT

Submitted by

Sighakolli susmitha(19BCB0056)

Chundi Indu(19BCB0002)

Simbothula Varun Kumar(19BCI0050)

CSE3002 Internet and Web Programming (EPJ)

Project Guide

DR.MANJULA V

Associate Professor Grade 1

School of Information Technology and Engineering

COMPUTER SCIENCE AND ENGINEERING

IN

School of Computer Science and Engineering

Winter Semester 2020-2021

TABLE OF CONTENTS

ABSTRACT:

An Online Food Ordering System simplifies the food ordering process. The proposed system aims to create a website that can be used in the food delivery industry. The system simplifies the process of food ordering. This system allows restaurants and hotels to increase their scope of business by reducing labor costs. This system allows customers to browse through the online menu and place an order with a few clicks and Customer can choose more than one item to make an order and can view order details before logging off. The modern food industries allow to quickly and easily delivery on customer place. Restaurant employees then use these orders through an easy delivery on customer place.

LIST OF FIGURES:

The Following images are present in The document of online Food Ordering system:

LIST OF ABBREVATIONS:

The Following Abbrevations are used in our Project report. Below are the abbrevations used along with their full forms:

- O_ID : Order ID
- R_ID : Restaurant ID
- F_ID : Food ID

- M_ID : Manager ID

LIST OF TABLES:

The following tables are present in the database of our project.

1) INTRODUCTION:

1.1) SYSTEM OVERVIEW :

The website is about online food ordering system in which customers can place order online easily and admin(restaurant management) will receive orders online.

This website benefits both customers and restaurant management. Because customer can avoid waiting in line in hotel for hours to get food. In the proposed system user can place order easily. Restaurant management can handle multiple orders online. It is less expensive and more convenient. Data can be stored for long time as system is computerized.

1.2) OBJECTIVES:

The objectives of the proposed system (online Food Ordering System) are:

- To develop a system that satisfies customers.
- To design a system that can take a huge amount of orders at a time.
- To minimize the time of ordering.
- To computerize the system to save data for a long time.

1.3) APPLICATIONS:

The Benefits of the proposed system (online Food Ordering System) are:

- Makes the ordering process easier.
- Efficient customer and order management.
- Easy to Monitor your expenses incurred in real-time.
- Free and cheap marketing.
- Easy to store customers data.
- increased convenience of mobile ordering.
- Helps to Stay ahead of the competition.

1.4) LIMITATIONS :

- Deliverymen put themselves in danger because they always wait outside hotel.
- Increased Expenses
- Revenue conflicts between the restaurants and delivery providers.
- Not taking proper care of health by eating outside food

2) SYSTEM ANALYSIS

2.1) EXISTING SYSTEM :

The existing Food Ordering System uses waiters to take orders from customers. During peak hours, customers may be too many to be served by waiters. The quality of the service may drop thus causing dissatisfaction of customers. However, if there are too many waiters are hired, it may be a waste of resources during a non-peak hour. This system is not comfortable for both customers and restaurant management.

2.2) PROPOSED SYSTEM :

The new online Food Ordering System consists of a website that allows customers to choose their desired food via the menu shown on the screen. The choices made will be sent to restaurant management. The system allows customers to see order details after placing an order.

2.2.1) Benefits of Proposed System :

The Benefits of the proposed system are:

- To develop a system that satisfies customers.
- To design a system that can take a huge amount of orders at a time.
- To minimize the time of ordering.
- To computerize the system to save data for a long time.

3) REQUIREMENT SPECIFICATION:

3.1) HARDWARE REQUIREMENTS :

For the Effective Implementation of the new system. The following hardware Requirements needed for the project are:

- 1GB RAM and above
- 40GB HD
- Printer
- Scanner
- Keyboard
- Intel Pentium
- Mouse

3.2) SOFTWARE REQUIREMENTS :

For the effective implementation of the new system, the following software has to be installed on the computer:

3.2.1) HTML:

HTML (Hypertext Markup Language) is the standard markup language for documents designed to be displayed in a web browser. We have designed the web pages in our Online Food Ordering using HTML-5.

3.2.2) CSS:

We will be using CSS3 in our Online Food ordering system project to customize various aspects of web pages like font size, style, colour, etc in order to make our web pages attractive.

3.2.3) Javascript:

We have used Java script in our Online Food ordering system.

3.2.4) Bootstrap:

We have used Bootstrap in Our Online Food ordering system project to make website Responsive.

3.2.5) MySql:

We have used MySQL as the relational database management system in our project in order to manage the database.

3.2.6) PHP:

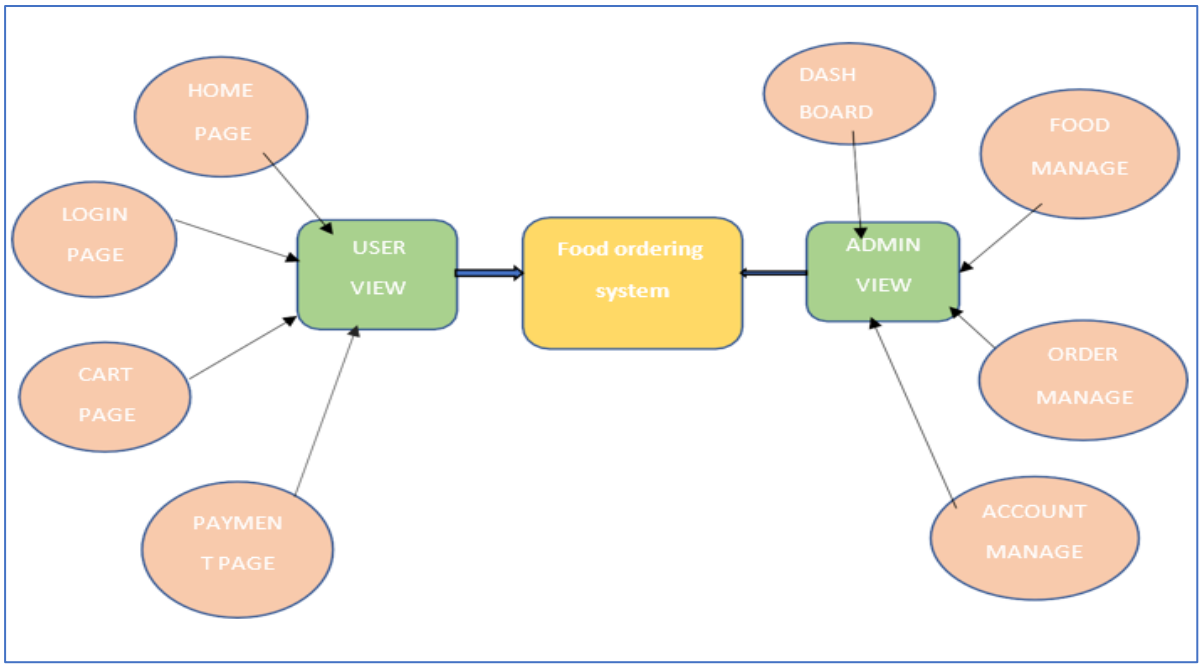
We will be using XAMPP for running the local server that we used for testing PHP scripts and to check the functionality of other parts of our online food ordering system.

4) SYSTEM DESIGN SPECIFICATION:

4.1) SYSTEM ARCHITECTURE:

Our Project can be represented in Repository model Because a central database/Repository has been used to store/read/edit/remove food items.

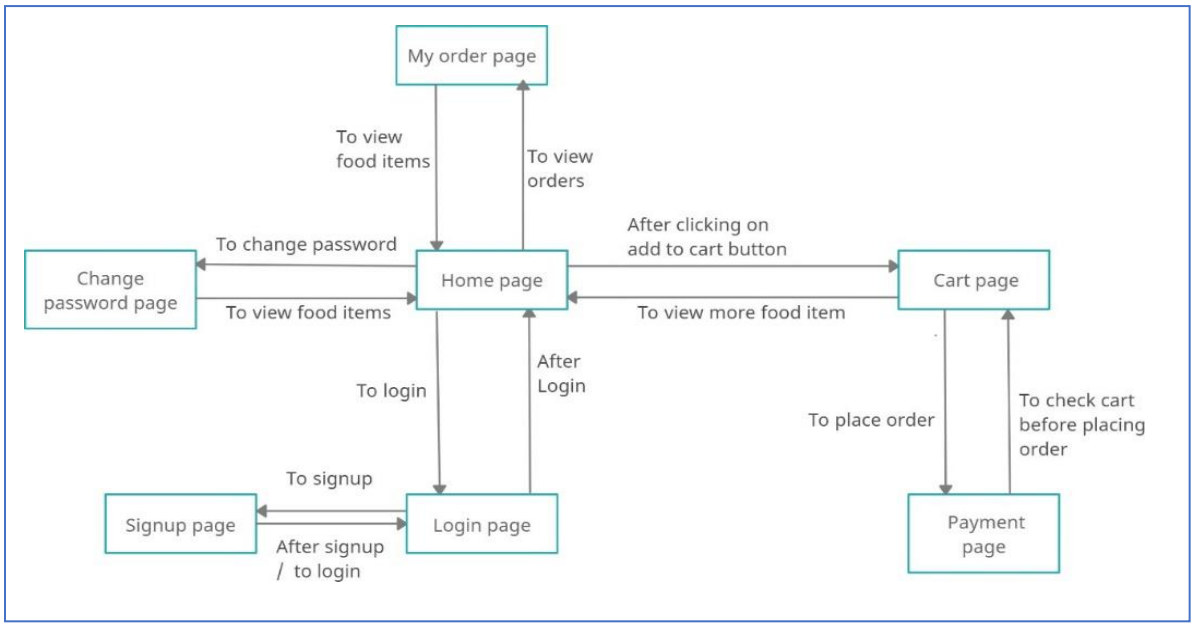
The flow of system architecture at the user end begins with the user interaction via opening the Homepage. The user can create account using signup page . user can login to their account using the login page and connects to the local host/ online server. After logging in user can check the food items and place an order by seeing the E-menu card. Order can be placed after adding products to cart page/ Generation of unique order id then the user is followed by payment method. The payment is facilitated by a payment page. After successful payment an Conformation Status is generated.



4.2) Detailed Design:

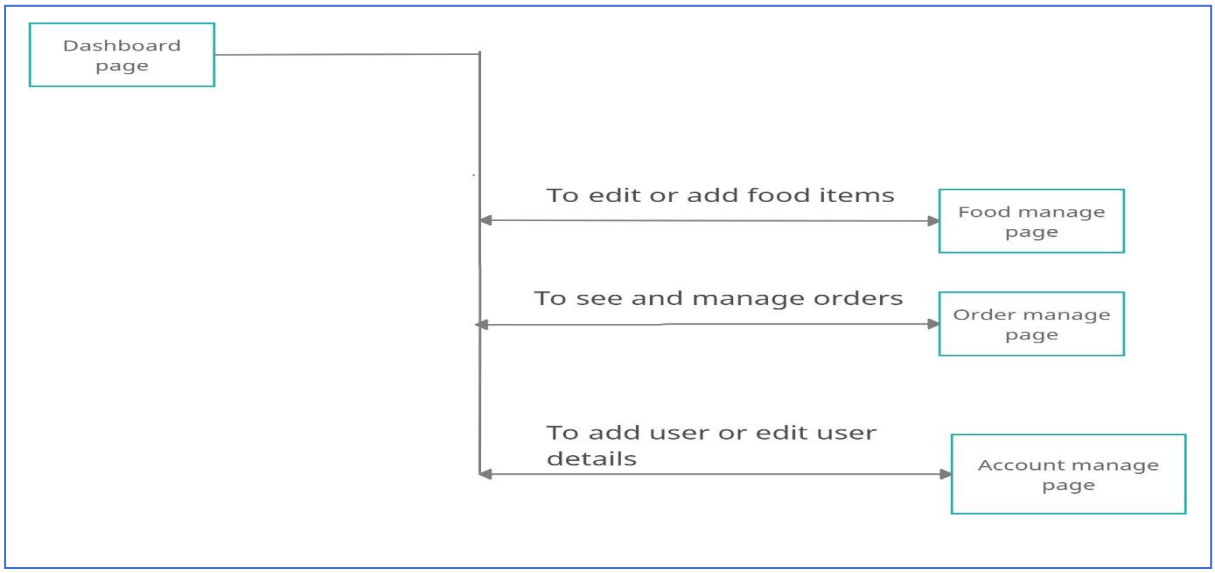
4.2.1) Detailed Design For User View:

User can select the food items from the menu displayed and can add them to cart and can proceed to checkout with available payment methods



4.2.2) DETAILED DESIGN FOR ADMIN VIEW:

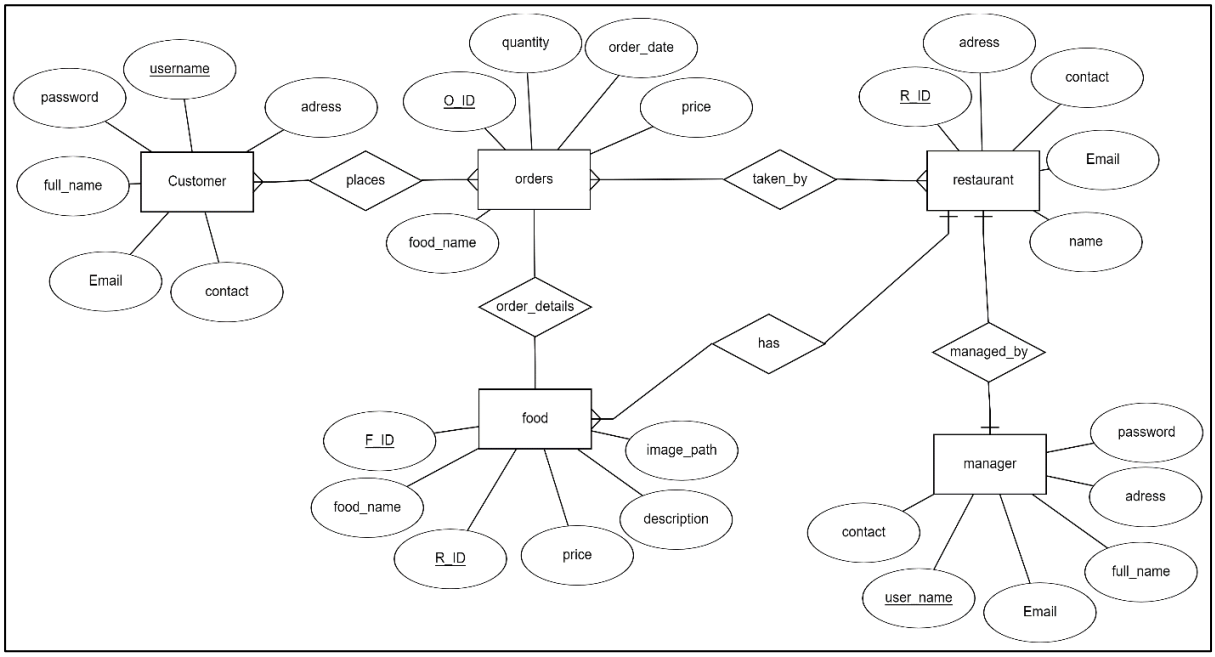
Admin can perform all the actions performed by use, In addition to those admin also have some extra rights over website as shown Below:



4.3) DATABASE DESIGN

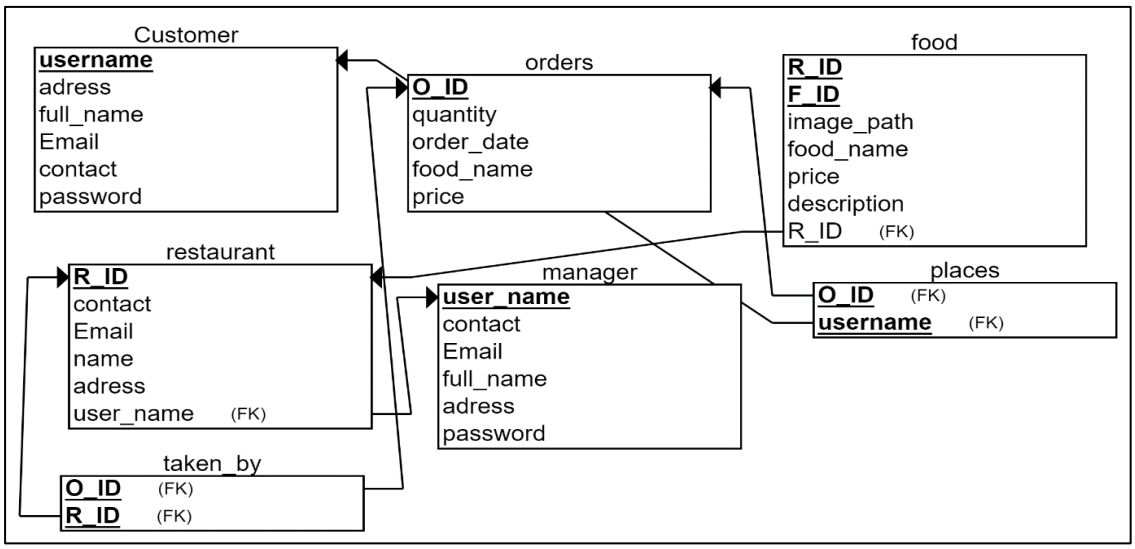
4.3.1) ER-Diagram:

Online food ordering system is a web based platform designed to cater the orders of customer. This ER Diagram will provide you with the basic layout and entities for every table in the development of online food ordering system.



4.3.2) Relational Schema:

The Relational Schema of Our Online food ordering system is shown below with total 7 tables:



4.3.3) Tables in Database:

4.3.3.1) Customer Table:

If new customer account is created, customer details get inserted Into customer Table:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	username	varchar(30)	utf8_general_ci		No	None			Change Drop More
2	fullname	varchar(30)	utf8_general_ci		No	None			Change Drop More
3	email	varchar(30)	utf8_general_ci		No	None			Change Drop More
4	contact	varchar(30)	utf8_general_ci		No	None			Change Drop More
5	address	varchar(50)	utf8_general_ci		No	None			Change Drop More
6	password	varchar(30)	utf8_general_ci		No	None			Change Drop More

4.3.3.2) Food Table:

If new food item is created, food details get inserted Into food Table:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	F_ID	int(30)			No	None		AUTO_INCREMENT	Change Drop More
2	name	varchar(30)	utf8_general_ci		No	None			Change Drop More
3	price	int(30)			No	None			Change Drop More
4	description	varchar(200)	utf8_general_ci		No	None			Change Drop More
5	R_ID	int(30)			No	None			Change Drop More
6	images_path	varchar(200)	utf8_general_ci		No	None			Change Drop More
7	options	varchar(10)	utf8_general_ci		No	ENABLE			Change Drop More

4.3.3.3) Manager Table:

If new manager account is created, manager details get inserted Into manager Table:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	username	varchar(30)	utf8_general_ci		No	None			Change Drop More
2	fullname	varchar(30)	utf8_general_ci		No	None			Change Drop More
3	email	varchar(30)	utf8_general_ci		No	None			Change Drop More
4	contact	varchar(30)	utf8_general_ci		No	None			Change Drop More
5	address	varchar(50)	utf8_general_ci		No	None			Change Drop More
6	password	varchar(30)	utf8_general_ci		No	None			Change Drop More

4.3.3.4) Orders Table:

If new order is placed by customer, Order details get inserted Into Orders Table:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	order_ID	int(30)			No	None		AUTO_INCREMENT	Change Drop More
2	F_ID	int(30)			No	None			Change Drop More
3	foodname	varchar(30)	utf8_general_ci		No	None			Change Drop More
4	price	int(30)			No	None			Change Drop More
5	quantity	int(30)			No	None			Change Drop More
6	order_date	date			No	None			Change Drop More
7	username	varchar(30)	utf8_general_ci		No	None			Change Drop More
8	R_ID	int(30)			No	None			Change Drop More

4.3.3.5) Restaurants Table:

If new restaurant is added by manager, Restaurant details get inserted Into restaurant Table:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	R_ID	int(30)			No	None		AUTO_INCREMENT	Change Drop More
2	name	varchar(30)	utf8_general_ci		No	None			Change Drop More
3	email	varchar(30)	utf8_general_ci		No	None			Change Drop More
4	contact	varchar(30)	utf8_general_ci		No	None			Change Drop More
5	address	varchar(50)	utf8_general_ci		No	None			Change Drop More
6	M_ID	varchar(30)	utf8_general_ci		No	None			Change Drop More

4.3.3.6) Contact Table:

If customer fills contact form, form details get inserted into contact Table:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Name	varchar(250)	latin1_swedish_ci		No	None			Change Drop More
2	Email	varchar(250)	latin1_swedish_ci		No	None			Change Drop More
3	Mobile	varchar(250)	latin1_swedish_ci		No	None			Change Drop More
4	Subject	varchar(250)	latin1_swedish_ci		No	None			Change Drop More
5	Message	varchar(250)	latin1_swedish_ci		No	None			Change Drop More

4.3.4) Primary Keys:

The Primary Keys of each table are mentioned Below:

- Customer Table– username
- Orders Table– O_ID
- Restaurant Table – R_ID
- Food Table– (R_ID, F_ID)
- Manager Table- user_name

4.3.5) RELATIONSHIPS:

The Relationships that are present between tables in Online food ordering system database are shown below:

- Customer ->places -> orders
- Restaurant ->has ->food
- Restaurant ->managed_by ->manager
- Orders-> taken_By ->restaurant
- Order-> order_details -> food

5) System Implementation:

5.1) Modules Description:

There are 2 modules in our Online Food Ordering system Project. They are:

- User View
- Admin View

5.1.1) User Module:

User can perform various actions on website like creation of account, logging into account, placing order etc. The action that can be performed by Users are mentioned below:

- User can Sign up.
- User can enter username for new registration
- User can create new password

- User can re -enter new password for confirmation
- User can login.
- User can enter a username
- User can enter created password
- System validates new password and username
- User can easily view the menu many number of times
- User can choose the quantity of food items
- User can add multiple food items to cart.
- User can delete the extra items from the cart that user doesn't want
- users have options to check out the food item list selected before proceeding for payment.
- Users can choose various payment portal method

5.1.2) Admin Module:

Admin can manage all the website in admin dashboard. The actions that can be performed in admin view are:

- Admin can login
- Admin can see number of food items present
- Admin can add different food items to the website
- Admin can delete the food items.
- Admin can see the order details placed by the Customers
- Admin maintains whole information of order details given by the customers

- Admin provide description to food items present in the website.

6) Conclusion and Future Enhancements:

6.1) Conclusion:

The Online Food Ordering System has helped make food ordering easier than the previously existing methods. The user can easily view the menu many number of times. He/she wishes and chooses what he/she wants to eat.

The system asks the user step by step which item he/she wants, which type of crust and size they want. It also allows changing the quantity of the item added and deleting extra items that the user doesn't want.

The user can also reorder his/her previous order. All this results in a better consumer experience, improving the efficiency of the restaurant which in turn gives them the chance to have more profits.

It overcomes many of the limitations the previous methods used to have. It also saves paper and eliminates human errors. The attractive interface makes the ordering process easier and the system itself easier to understand.

6.2) Future Enhancements:

There are many more features that can be added to the system like making it to accept input via mouse instead of keyboard. It may also store all previous orders so that the user can choose from all previous orders instead of just one. It may add the orders into a queue and give the user the approximate time it will take for the order to be prepared.

An option of customizing the item with extra toppings can also be added. Also, this whole system can be made two ways by dividing it into two modules: User and Store. The user section will contain all the features discussed above but the store section will provide the store a platform to manage their orders, scheduling them to improve the efficiency even more.

An option to put out offers and schemes which will be informed to the customer via notifications. With this two-way system many more features can be added like an option of payment through the app directly resulting in cashless

transactions. The system's implementation would significantly improve the whole experience and save time and money and also provide a better interface.

7) Appendices:

7.1) Appendix 1- Sample source code:

7.1.1) login_u.php (login page for user):

```
<?php
session_start();
$error='';

if (isset($_POST['submit'])) {
if (empty($_POST['username']) || empty($_POST['password'])) {
$error = "Username or Password is invalid";
}
else
{
// Define $username and $password
$username=$_POST['username'];
$password=$_POST['password'];
// Establishing Connection with Server by passing server_name, user_id and password as a parameter
require 'connection.php';
$conn = Connect();

// SQL query to fetch information of registerd users and finds user match.
$query = "SELECT username, password FROM CUSTOMER WHERE username=? AND password=? LI
MIT 1";

// To protect MySQL injection for Security purpose
$stmt = $conn->prepare($query);
$stmt -> bind_param("ss", $username, $password);
$stmt -> execute();
$stmt -> bind_result($username, $password);
$stmt -> store_result();

if ($stmt->fetch())
{
   $_SESSION['login_user2']=$username; // Initializing Session
   header("location: foodlist.php"); // Redirecting To Other Page
} else {
$error = "Username or Password is invalid";
}
mysqli_close($conn); // Closing Connection
}
}
```

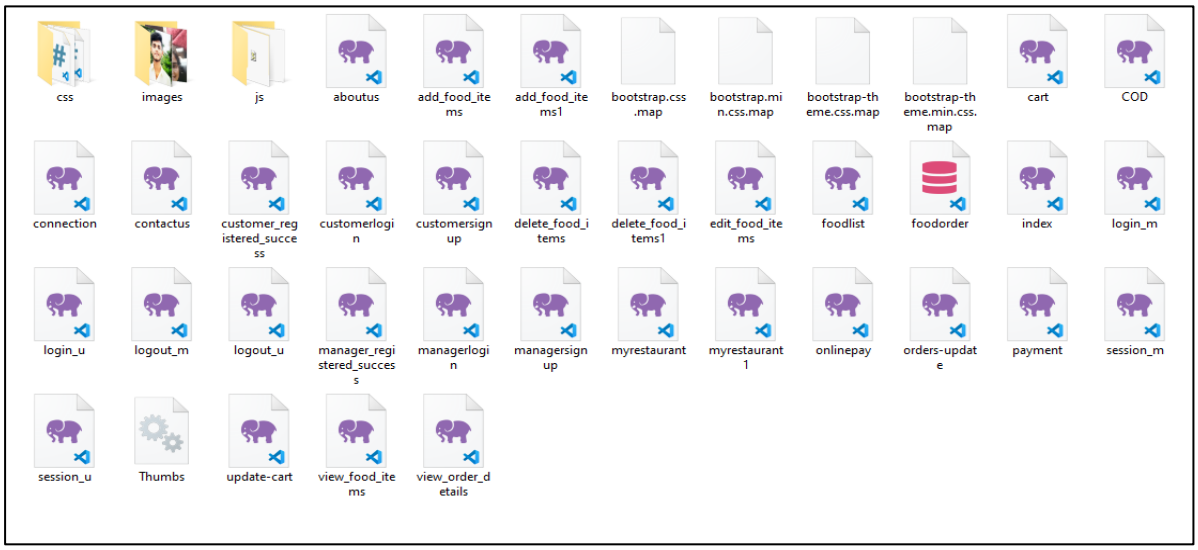
```php
?>
```

7.1.2)update_cart.php (updating cart page in user module):

```php
<?php

if(!isset($_SESSION))
  {
    session_start();
  }

include 'connection.php';
$conn = Connect();

$F_ID = $_GET['id'];
$action = $_GET['action'];




$sql = "SELECT quantity FROM food WHERE F_id = ".$F_ID;

$result = mysqli_query($conn, $sql);


if($result){

  if($obj = mysqli_fetch_assoc($result)) {

    switch($action) {

      case "add":
      if($_SESSION['cart'][$F_ID]+1 <= $row["quantity"])
        $_SESSION['cart'][$F_ID]++;
      break;

      case "remove":
      $_SESSION['cart'][$F_ID]--;
      if($_SESSION['cart'][$F_ID] == 0)
        unset($_SESSION['cart'][$F_ID]);
        break;



    }
  }
```

```
}


header("location:cart.php");

?>
```

7.1.3) Codes Used for Implementation of Website:



7.1.4) Link for Source Code:

Google Drive link for the source code of online Food Ordering system is given Below.

Link:

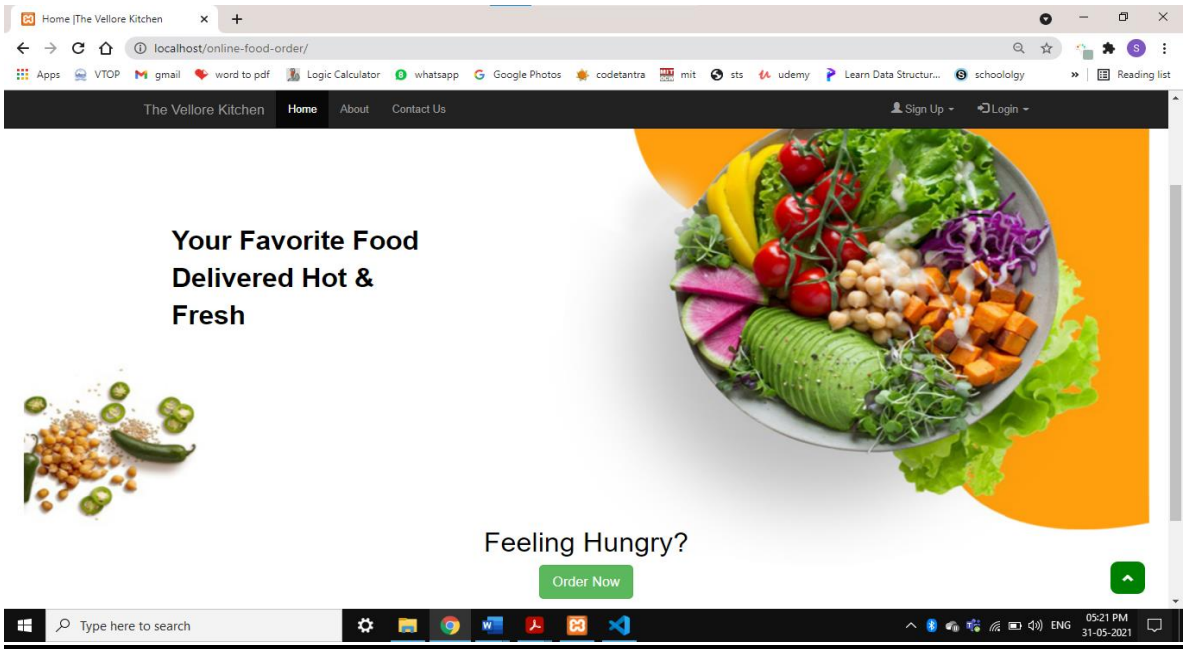
https://drive.google.com/drive/folders/1cnXQY8ts3Kbrc7TFWHCmTtCHGbOMmX2I?usp=sharing

7.2) Appendix 2- Screenshots:

7.2.1) User View:

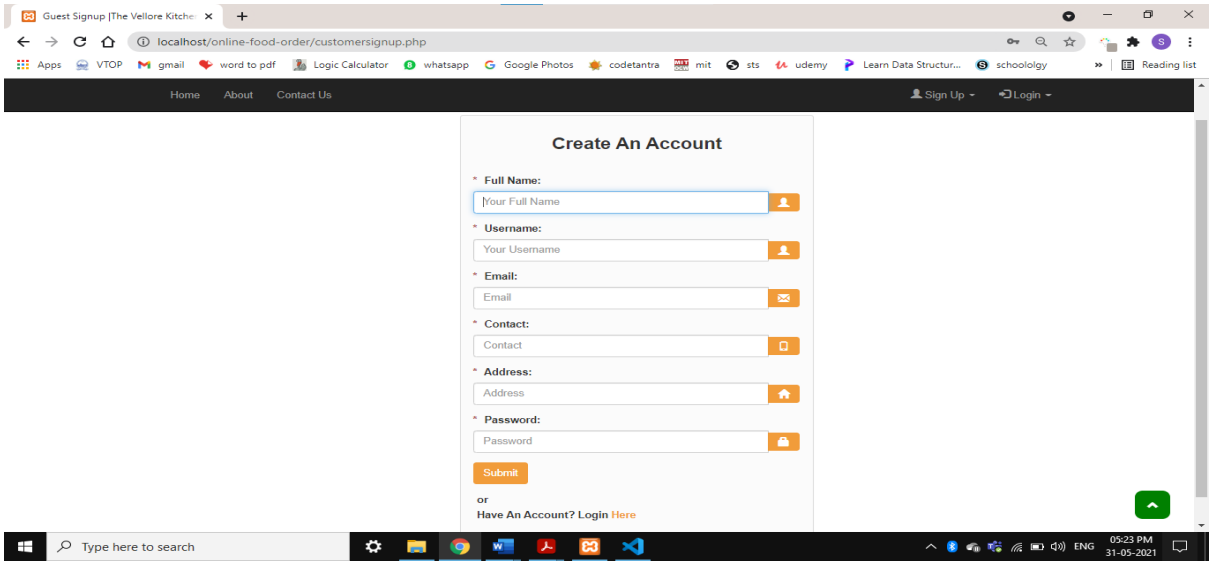
7.2.1.1) Home page:

This is attractive home page for users. It has beautiful navigation bar and options to signup and login.

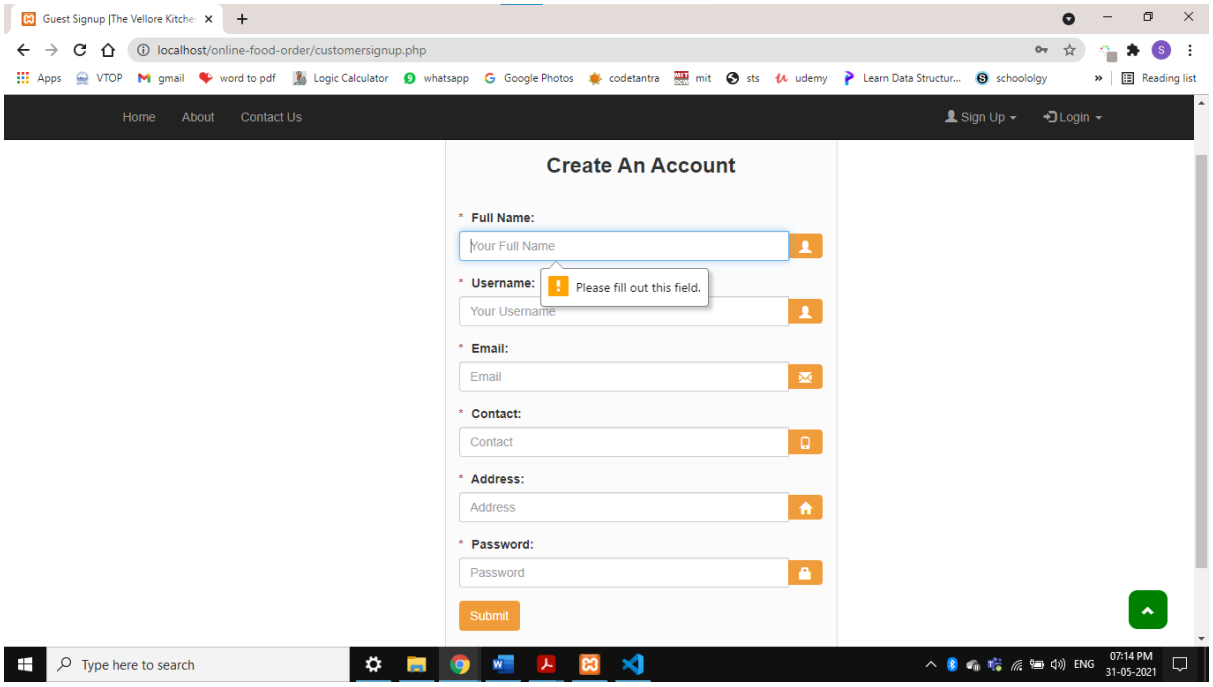


7.2.1.2) User Signup Page:

This page allows user to create new account by filling out the details like name, phone number etc.



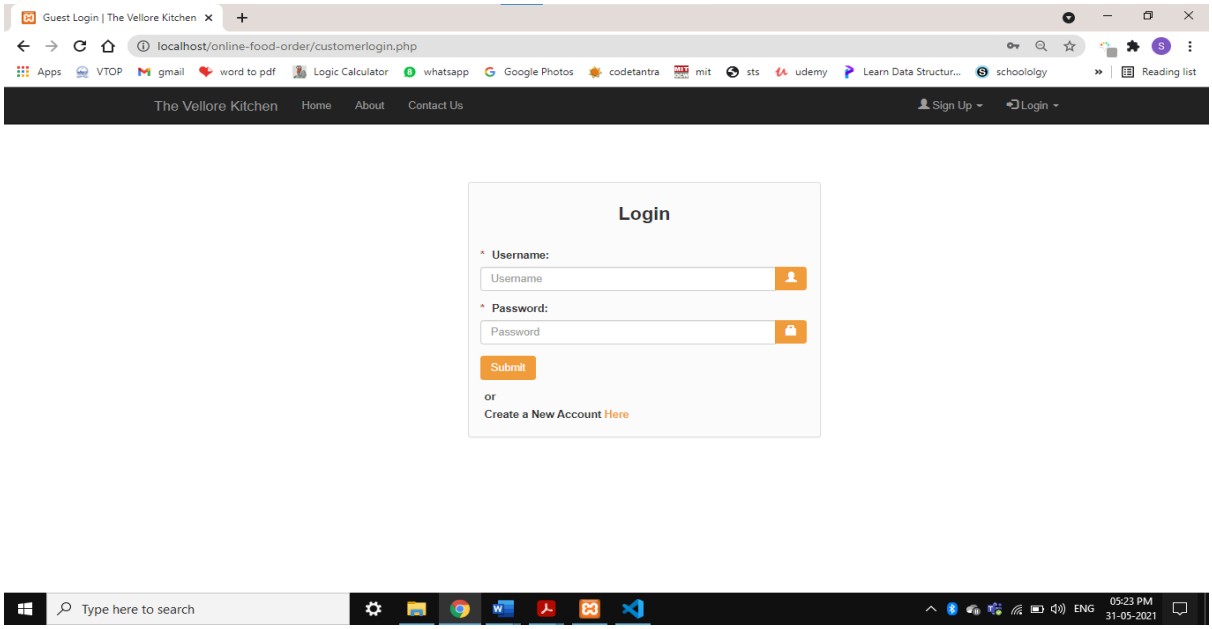
Validating The Form:



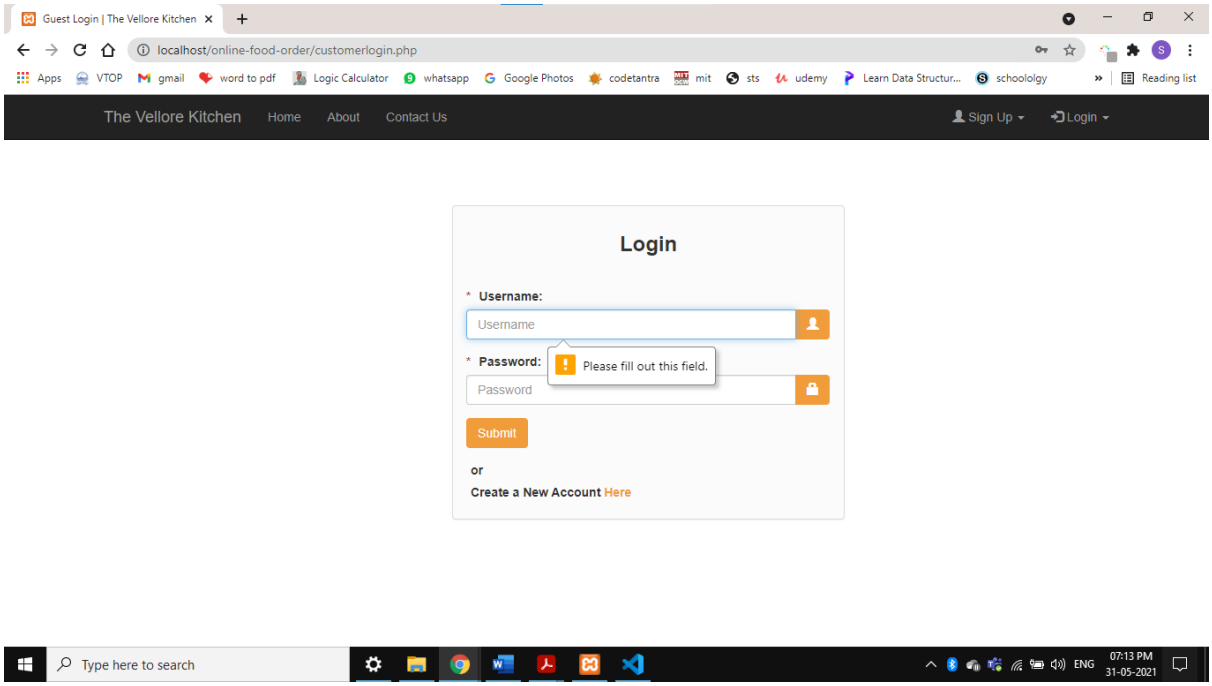
7.2.1.3) User Login Page:

This page allows user to login into their account with their username and password.

There is also validation in this form

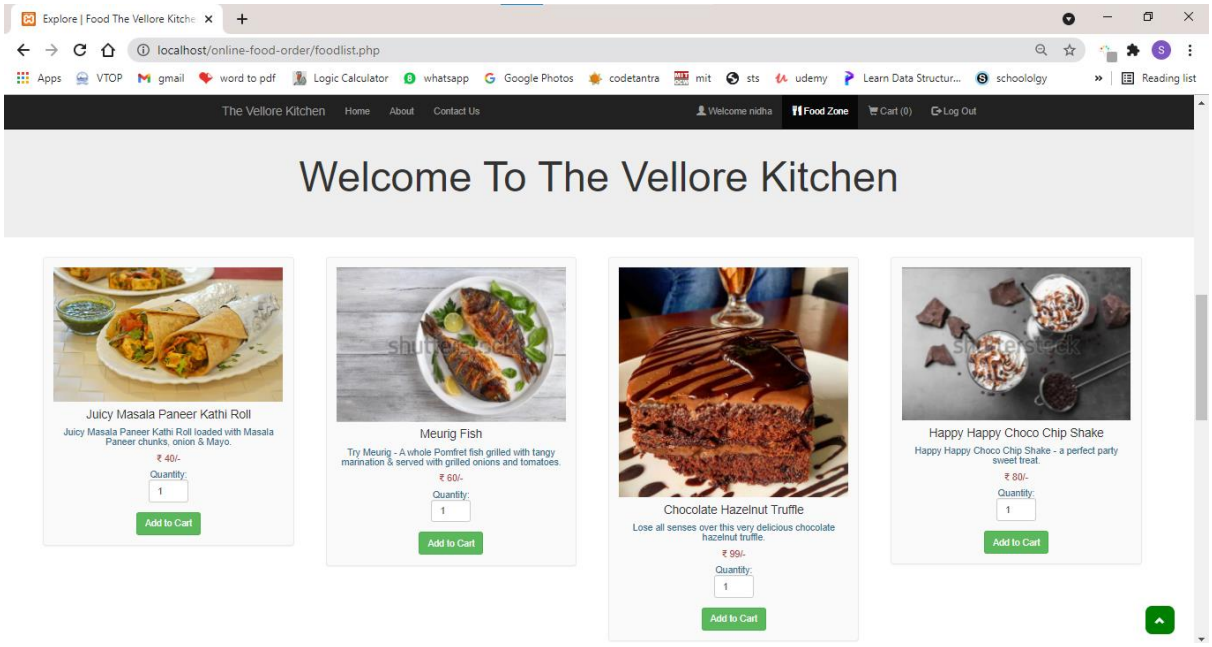


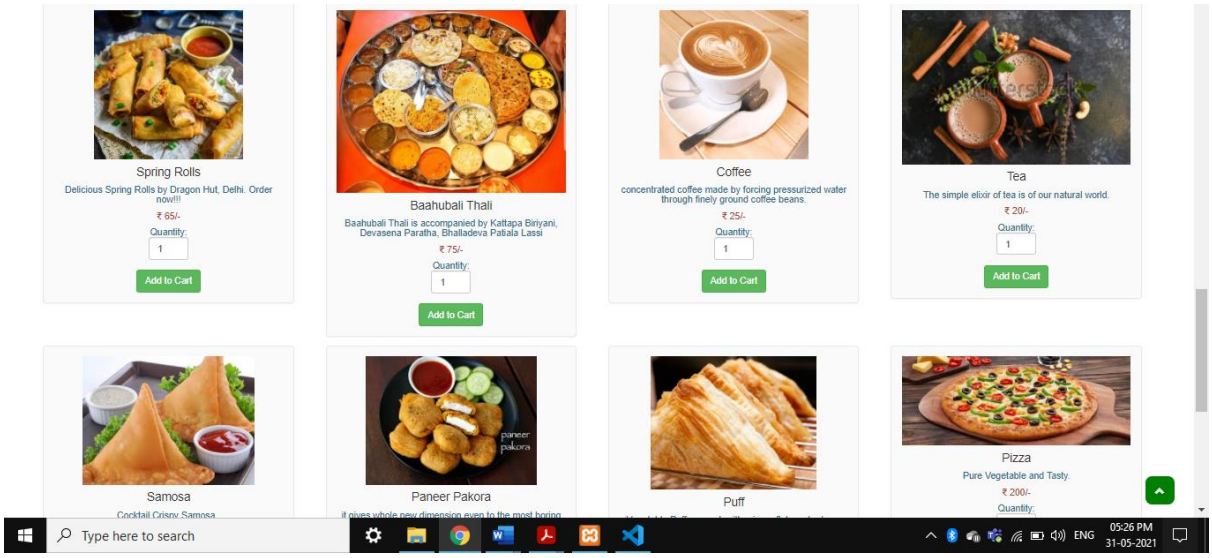
Validating The form:



7.2.1.4) Food Zone Page:

This page has all the food items. Users after logging into their account can see all the food items and can choose quantity and add food items to cart.

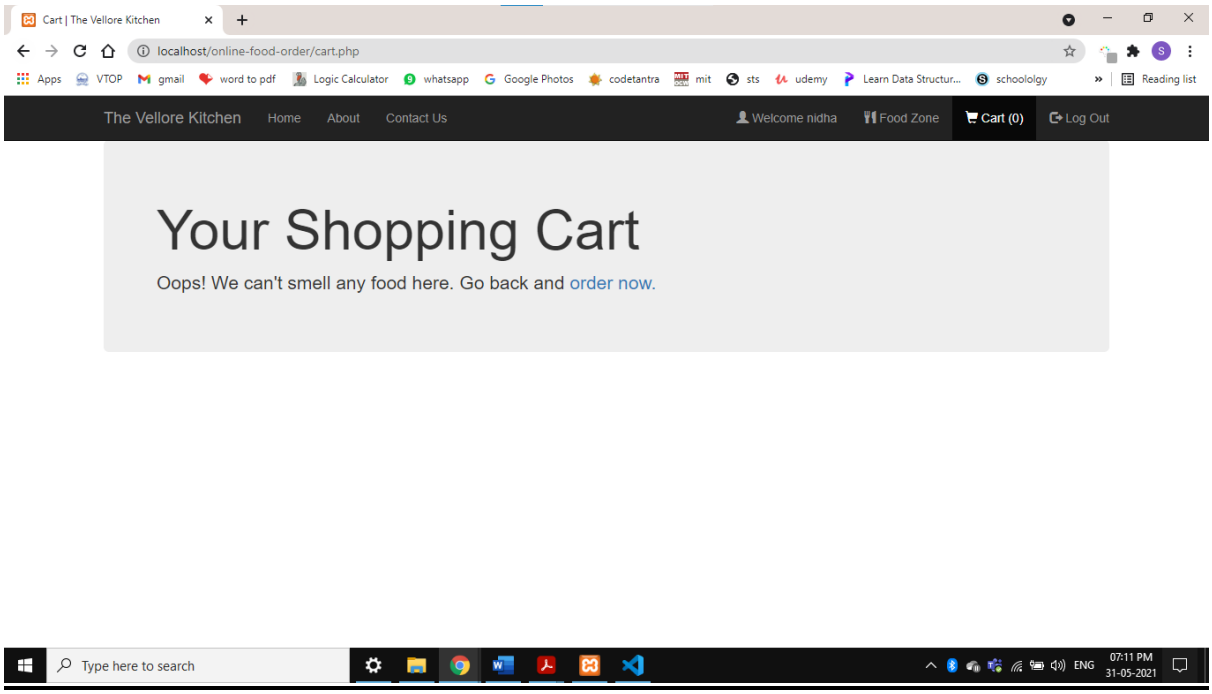




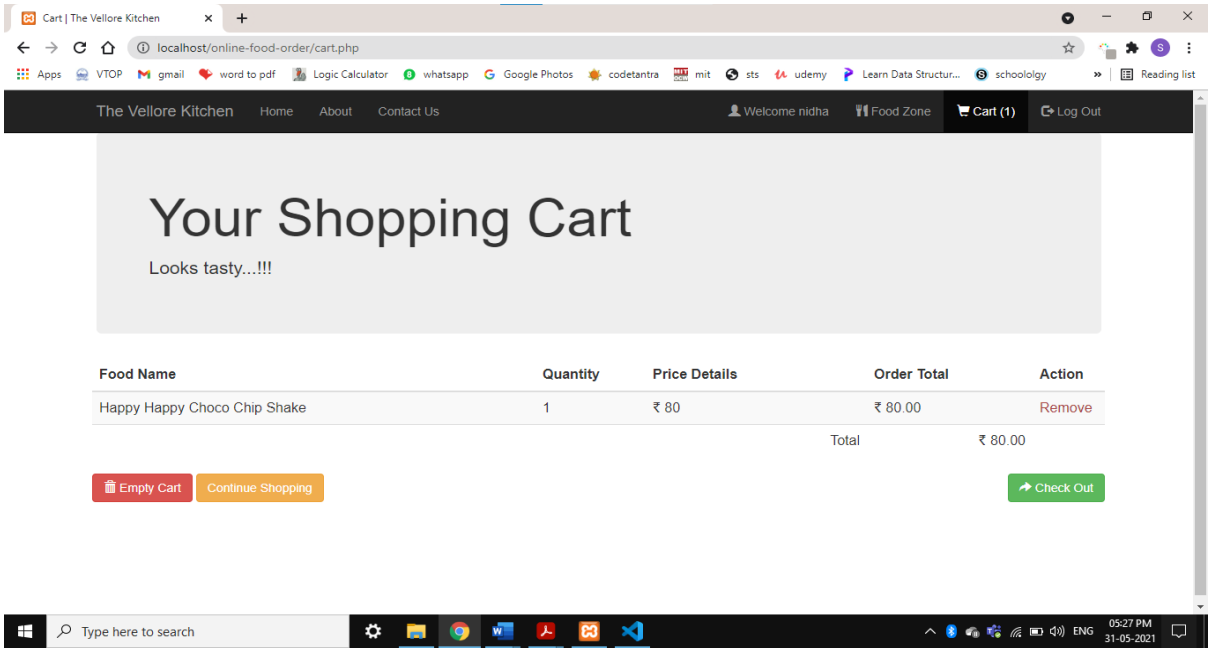
7.2.1.5) Cart Page:

This page shows all the items that are added to cart by user. User can also delete the food items from cart. It also has link to food zone page to brose for more food items.

When cart is empty:

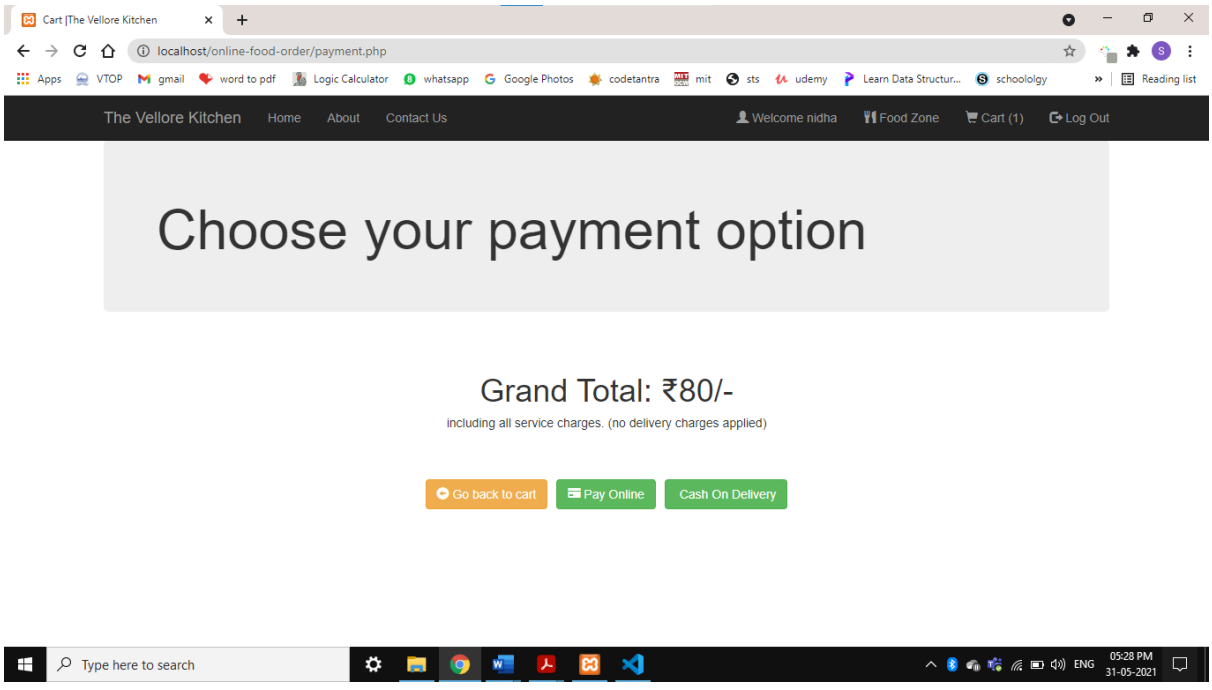


After adding Food items to cart:

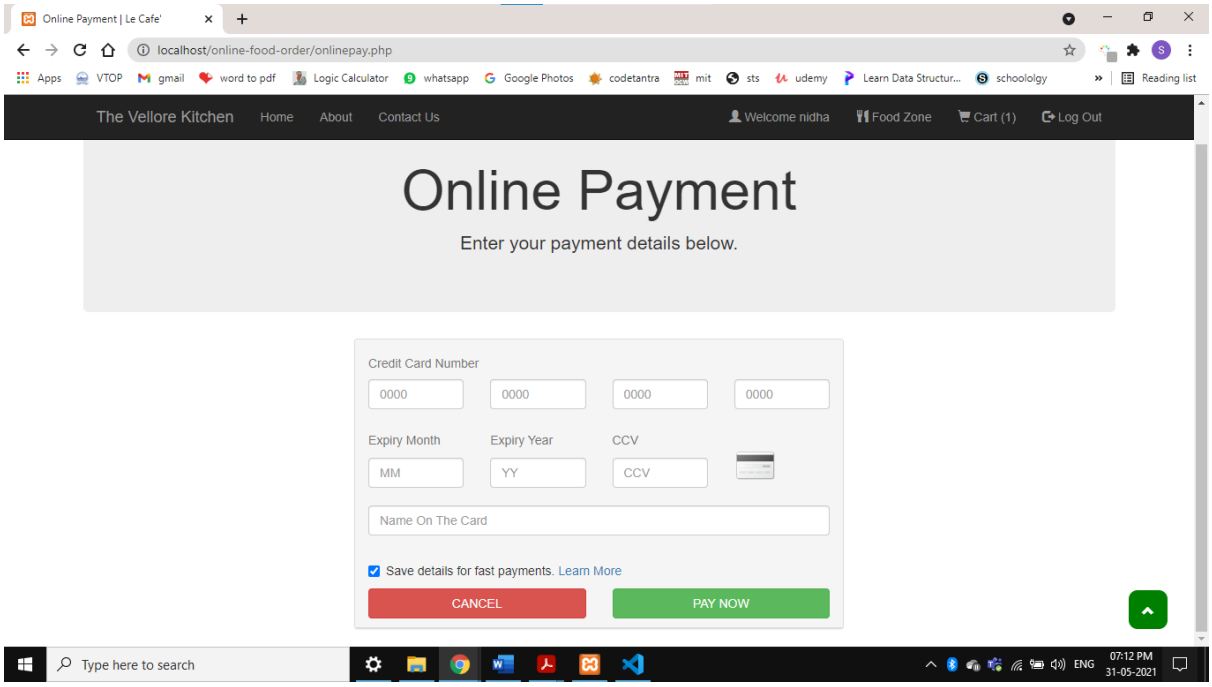


7.2.1.6) Payment page:

This page shows all the various payment modes available. User can choose one among those payment methods to proceed to place order.

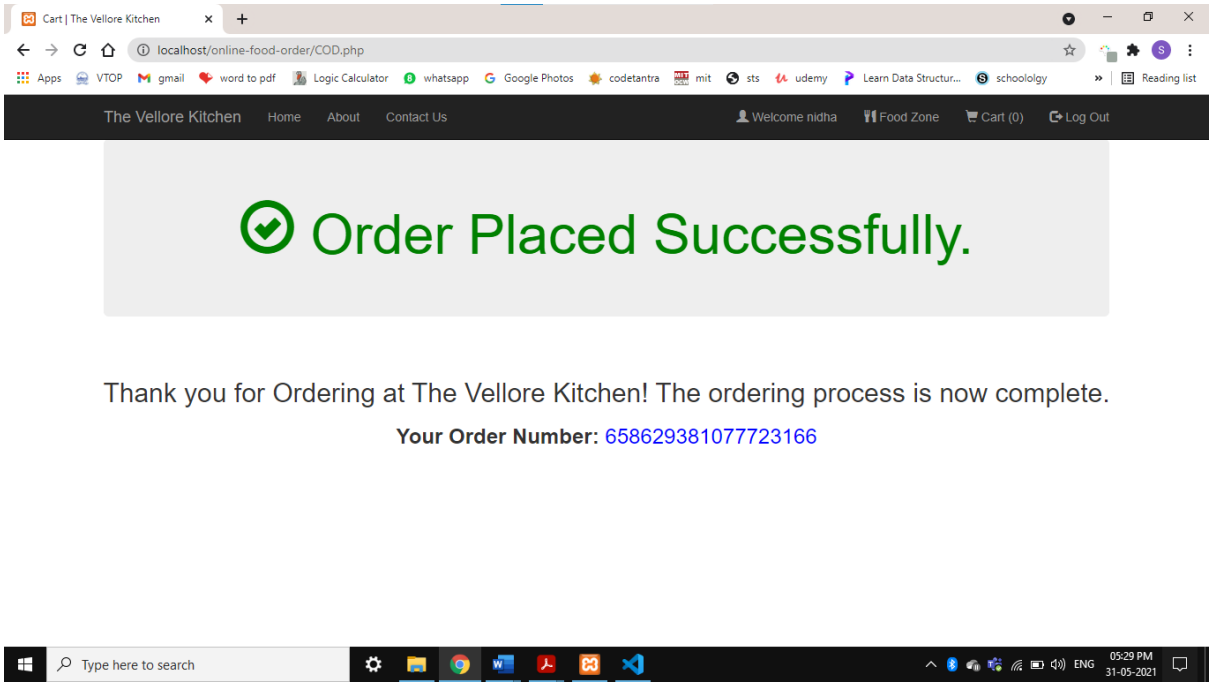


After choosing Online Payment Mode:



7.2.1.7) Order Placed Page:

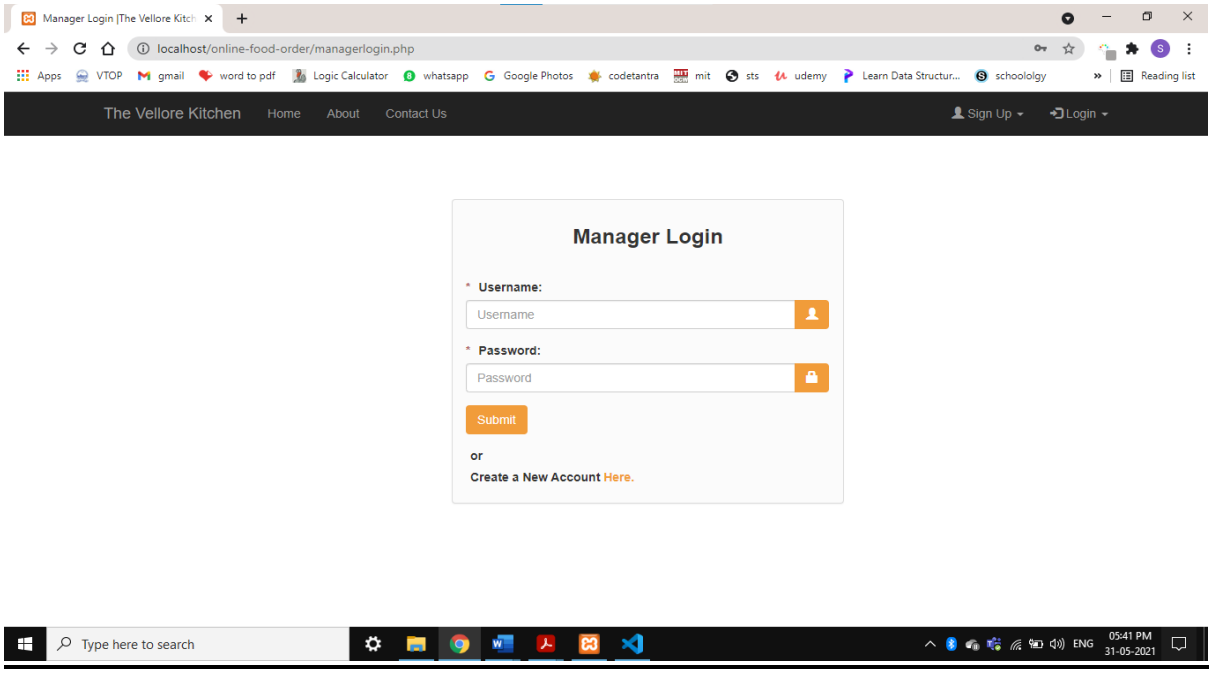
This page appears after placing order successfully. It shows order number of placed order.



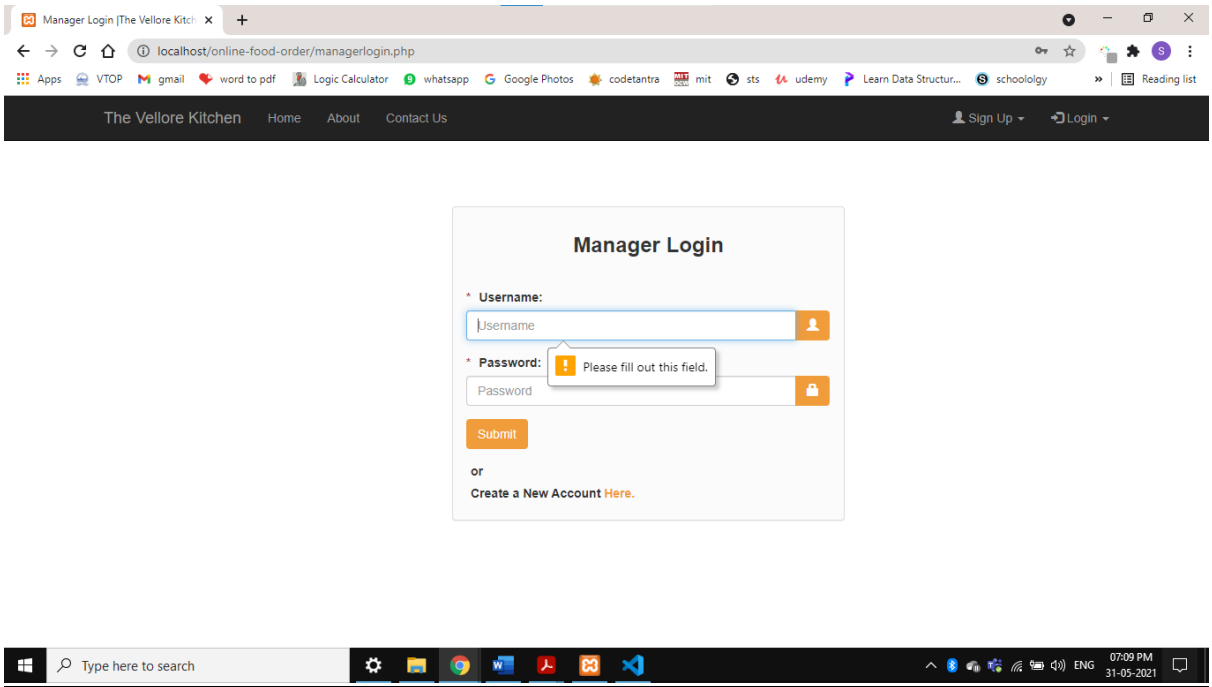
7.2) Admin View:

7.2.1) Login Page:

This page allows admin to login to admins account.

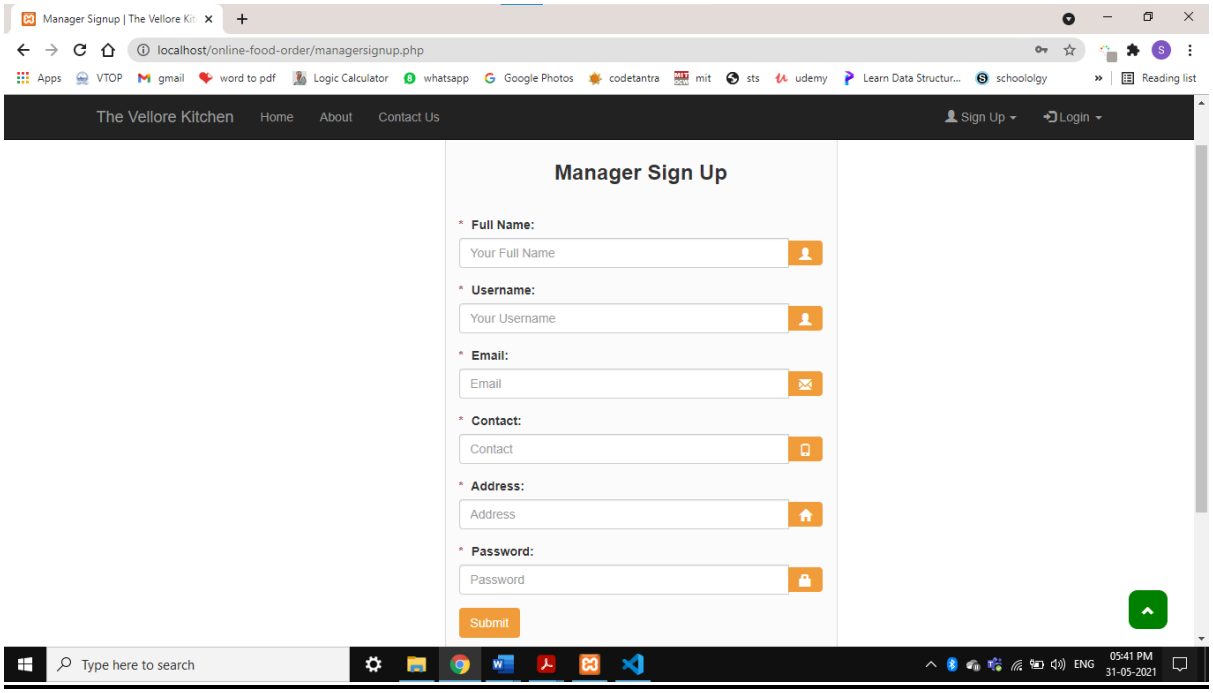


Validating The Form:

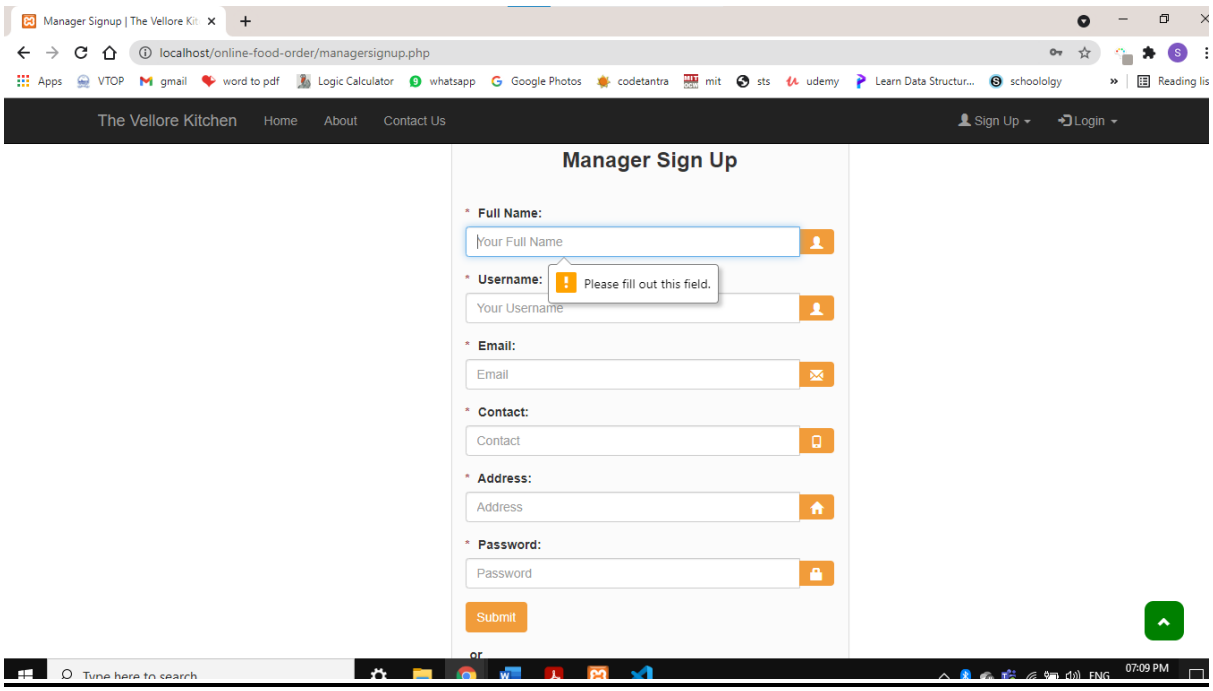


7.2.2) Signup page:

This page allows admin to create new account by filling the details that are given in the following form:

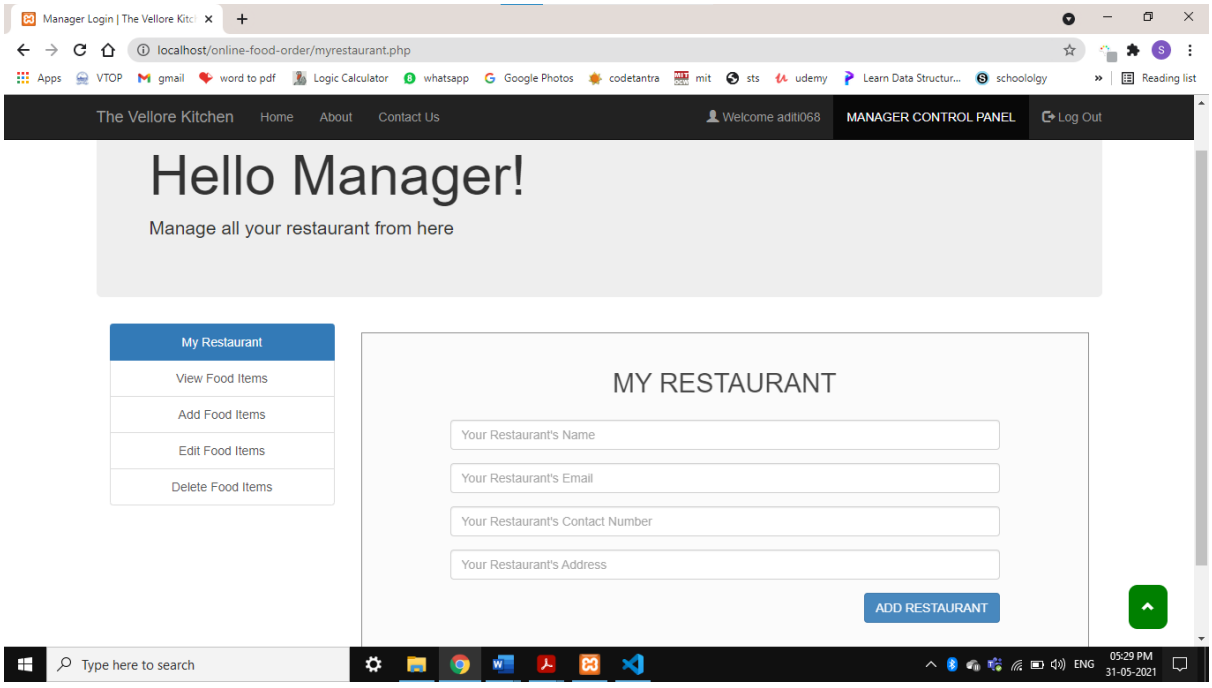


Validating The Form:



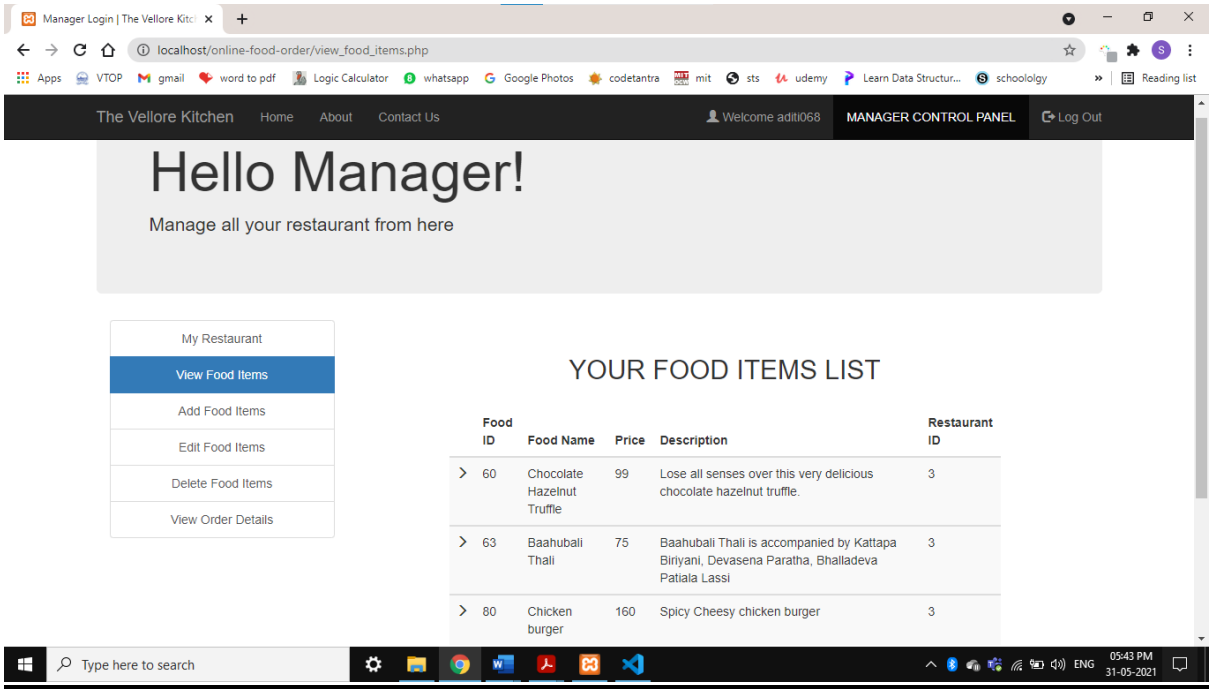
7.2.3) My Restaurant Page:

This page allows admin to add details of his/her restaurant details.



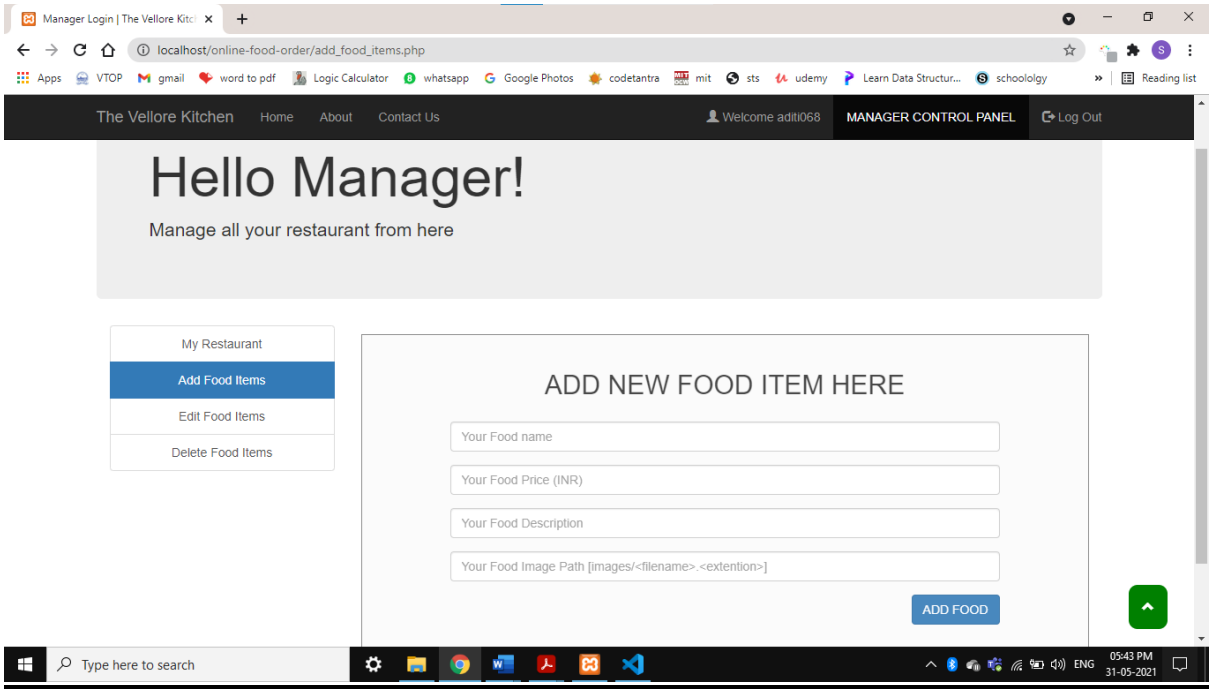
7.2.4) View Food Items Page:

This page allows admin to see all the food items that are uploaded by admin.



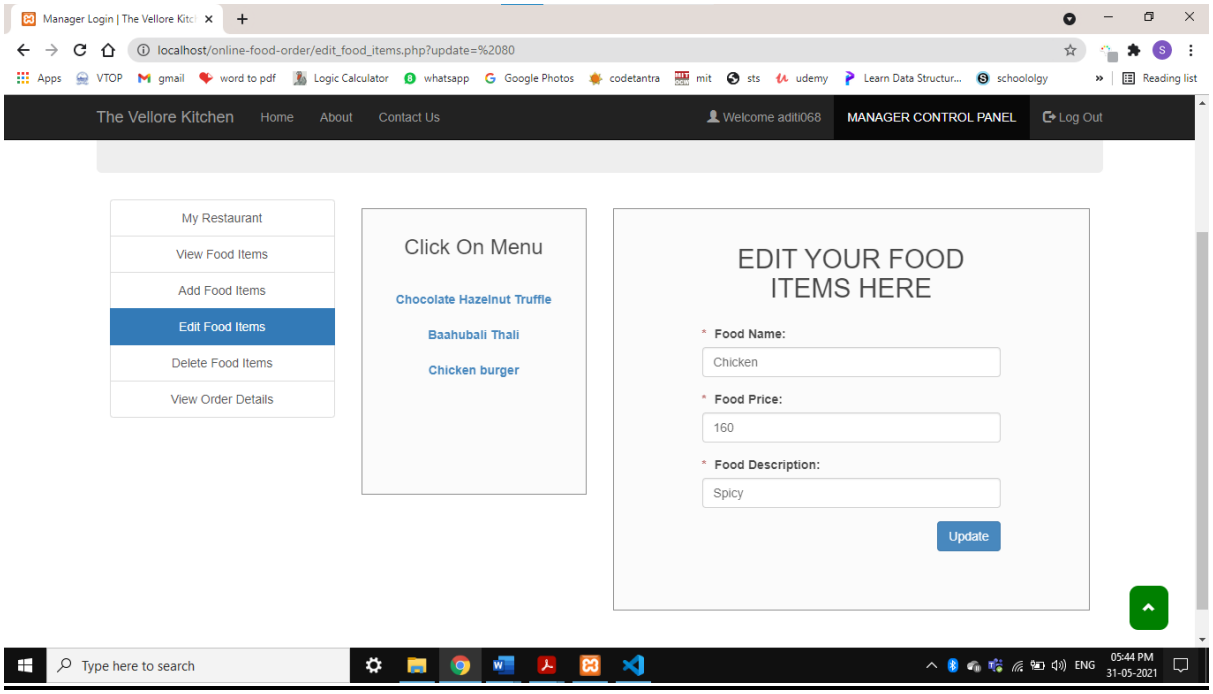
7.2.5) Add Food Items Page:

This page allows admin to add food items the website. Admin can also mention price and description to food items.



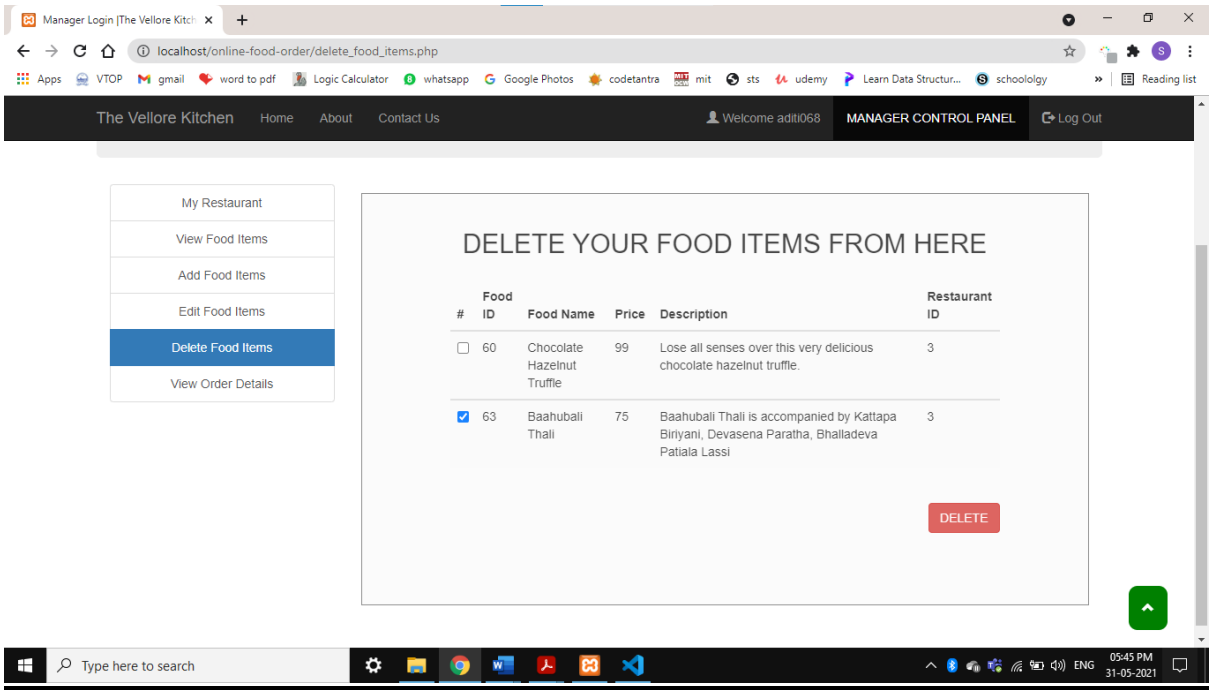
7.2.6) Edit Food Items Page:

This page allows admin to edit details of food items.



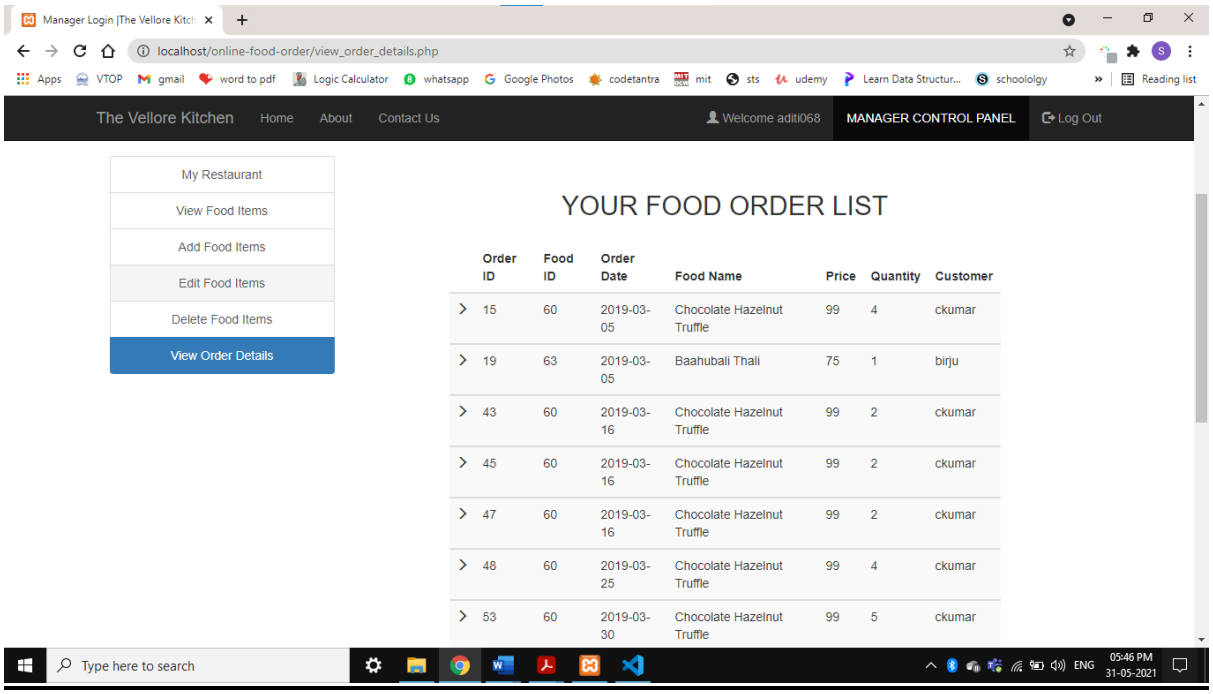
7.2.7) Delete Food Items Page:

This page allows admin to delete food items in website.



7.2.8) View Orders Page:

This page allows admin to see all the orders placed by customers.



8) References:

8.1) List of Journals:

The following journals are referenced while doing this project:

[1] Purname, J., & Wibowo, A.Y. (2007). Wireless Application for Ordering Management System in a Restaurant.

[2] Sharma, I. (2007, June 22). How Online Food Ordering System helps yourun more efficiently.

[3] Anderson, R.G. (1978), "Data processing and Management Information Systems" McDonald and Evans Ltd.

[4] TechCrunch.(2015,May 07) "A Secular Shift To Online Food Ordering".

[5] "Restaurant Food Delivery heating up". Columbian.com. Retrieved January 10, 2016.

8.2) List of Websites:

The following websites are referenced for the project

[1] https://www.swiggy.com/

[2] https://www.zomato.com/

[3] https://pizzaonline.dominos.co.in/

8.3) List of Books:

The following books are referenced for the project

[1] TechCrunch.(2015,May 07) "A Secular Shift To Online Food Ordering".

[2] "Restaurant Food Delivery heating up". Columbian.com. Retrieved January 10, 2016