OUTLINE FOR THE PROJECT IN MCA I SEMESTER II

ONLINE FOOD ORDERING SYSTEM

BY

SANKET GOVIND KHARDEKAR

BSc. COMPUTER SCIENCE

UNDER THE GUIDENCE OF

Dr. Rajesh Kashyap

LLB, MCA, Ph.D. (Computer Science)

Head Of Department, Associate Professor, C.E.O.

Zeal’s

& RESEARCH

S.NO. 39, NARHE, PUNE-41

YEAR-2020

**MCA-I**

**INDEX**

**CHAPTER 1: INTRODUCTION**

1.1 Client Description

1.2 Existing System and Need for System

1.3 Scope of Work

1.4 Operating Environment – Hardware and Software

1.5 Detail Description of Technology Used

**CHAPTER 2: PROPOSED SYSTEM**

2.1 Proposed System

2.2 Objectives of System

2.3 User Requirements

**CHAPTER 3: ANALYSIS & DESIGN**

3.1 Object Diagram

3.2 Class Diagram

3.3 Use Case Diagrams and Use case description

3.4 Sequence Diagram

3.5 Activity Diagram

3.8 Data Flow Diagram (DFD)

**CHAPTER 4: DATABASE SPECIFICATIONS**

4.1 Entity Relationship Diagram (ERD)

4.2 Database Design (Normalized database)

4.3 Data Dictionary

**CHAPTER 5: USER INTERFACE SPECIFICATIONS**

5.1 Menu Tree

5.2 Input Screens

5.3 Reports

5.4 Test Cases

**CHAPTER 6: USER MANUAL**

6.1 User manual

**Limitations of the system**

**Proposed Enhancements**

**Conclusion**

**Bibliography**

**CHAPTER 1: INTRODUCTION**

**1.1 Client Description**

The client “Mr. Bajirao” is the owner of cafe named “Friends Corner” providing food facilities to everyone who walks into the cafe whilst during opening and closing hours on a regular basis throughout the year. Providing specifically mentioned food items according to their menu. The Primary Information of System was collected through direct interviews of cafe owner. The owner told us about the food items which he sells offline along with their rates. He told about the delivery process that he can follow.

He also suggests us some points.

**1.2 Existing System and Need for System**

**Existing System:**

Existing system is manual everything we have to do manually.

1. Platform Used: Records written in books.
2. Database Used: Books and Register.
3. Features:
   1. In pre-existing system customer goes to café and gives order and consumes the food item.
   2. The café owner has limited no of customers.

**Need for System:**

1. Maintaining order details and information becomes easy
2. As the system is online therefore there are no limit to the customer.
3. Suffering and searching facility is provided which will make searching of food items faster for the customer.
4. Cash on Delivery Facility is provided.

**1.3 Scope of Work**

1. On First page “Friends Corner Cafe” related Information will be available.
2. Searching and surfing through given food items facility will be provided.
3. Multiple given food items can be selected from the list and will be added to the cart.
4. Customer and Admin will only have login facility.
5. Bill will be generated according to given food item being selected as per their rates.
6. Customer and Admin will have the history of records of orders.

**1.4 Operating Environment – Hardware and Software**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Client-Side | Development-Side | Server-Side |
| Software | Windows or any operating system, Chrome, Firefox, etc. | Operating system, IDE and XAMMP | XAMMP |
| Hardware | At least 500MB RAM. | 4GB RAM | 8GB RAM  1GB web hosting space |

**1.5 Detailed Description of Technology Used**

**HTML-5**

**HTML5** is a language that defines the properties and behaviors of [web page](https://en.wikipedia.org/wiki/Web_page) [content](https://en.wikipedia.org/wiki/Web_content) by implementing a [markup](https://en.wikipedia.org/wiki/Markup_language)-based [pattern](https://en.wikipedia.org/wiki/Software_design_pattern) to it.

Some of the elements which we will be using:

Tags used are:

* <html>- element defines the root of HTML and XHTML documents.
* <div>- (short for division) tag is generic container for flow content, which has no default rendering or meaning.
* <h1> to <h6>- heading tags are used to define HTML headings. There are six levels of headings, from <h1> (most important) to <h6> (least important).
* <style>-tag is used to define style rules at a page-level (i.e. within the document).
* <link>-tag defines a link between the current document and an external resource.
* <button>- tag creates a clickable button in an [HTML form](https://www.tutorialrepublic.com/html-tutorial/html-forms.php).
* <body>-element represents the main content of the document. It typically wraps around all of the content that will be displayed on screen, such as headings, paragraphs, hyperlinks, images, forms, tables, lists, videos and so on.
* <form>- tag defines an HTML form that contains interactive controls which enable a user to submit information to a web server.
* <option>- element represents an option in a dropdown list defined by the [<select>](https://www.tutorialrepublic.com/html-reference/html-select-tag.php) element. A dropdown list must contain at least one <option> element.
* <select>- element defines a selection list within a [<form>](https://www.tutorialrepublic.com/html-reference/html-form-tag.php).
* <li>- (short for list item) defines an individual list item within a list. Each list item usually rendered with a bullet (in unordered lists, defined by the [<ul>](https://www.tutorialrepublic.com/html-reference/html-ul-tag.php) tag) or a number or letter (in the case of ordered lists, defined by the [<ol>](https://www.tutorialrepublic.com/html-reference/html-ol-tag.php) tag).
* <ul> -(short for unordered list) element defines an unordered list of items. Each list item is defined by a [<li>](https://www.tutorialrepublic.com/html-reference/html-li-tag.php) element.
* <ol>- (short for ordered list) tag defines an ordered list of items. Each list item is defined by a [<li>](https://www.tutorialrepublic.com/html-reference/html-li-tag.php) tag.
* <table>- tag is used to represents data in a grid-like fashion (in rows and columns). <th>,<tr>,<td> are elements used in table.
* <img>- (short for image) tag defines an image in an HTML document. Images are not directly inserted into the document; they are linked to the HTML pages.

**BOOTSTRAP-4**

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation and other interface components.

## **Bootstrap 4 CDN**

If you don't want to download and host Bootstrap 4 yourself, you can include it from a CDN (Content Delivery Network).

MaxCDN provides CDN support for Bootstrap's CSS and JavaScript. You must also include jQuery:

MaxCDN:

<!-- Latest compiled and minified CSS -->  
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css">  
  
<!-- jQuery library -->  
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>  
  
<!-- Popper JS -->  
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>  
  
<!-- Latest compiled JavaScript -->  
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.min.js"></script>

Bootstrap 4 Classes:

Classes is the important element in bootstrap.

All we have to include it in div tag. Eg:<div class=”.navbar”>

Some of the the classes used are: .active, .navbar, .bg-secondary, .bg-primary.etc.

Some of the elements used for are:

* Grid System-Bootstrap's grid system is built with flexbox and allows up to 12 columns across the page.

If you do not want to use all 12 columns individually, you can group the columns together to create wider columns.

Grid classes:.col-,.col-sm-,.col-md-,.col-lg-,.col-xl-

* Color-Bootstrap 4 has some contextual classes that can be used to provide "meaning through colors".The classes for text colors are: .text-muted, .text-primary, .text-success, .text-info, .text-warning, .text-danger, .text-secondary, .text-white, .text-dark, .text-body (default body color/often black) and .text-light
* Table- A basic Bootstrap 4 table has a light padding and horizontal dividers.

The .table class adds basic styling to a table.

* Image-Tag used for inserting image:<img src="paris.jpg" class="float-left">
* Buttons: Different types of button are-

Basic,Primary,Secondary, Success, Info, Warning, Danger, Dark Light .

* navigation bar -is a navigation header that is placed at the top of the page.
* Bootstrap provides two types of form layouts:

Stacked (full-width) form.

Inline form.

**CSS 3**

Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a markup language. CSS3 is a latest standard of CSS earlier versions (CSS2).

CSS can be implemented using 3 types: -

Inline CSS: -

E.g.- <p style=”color:sienna;margin-left:20px”>This is example</p> .

External CSS: -

In this type CSS file is external and is included in html file in <head> tag <link rel=”spreadsheet” type=”text/css” href=”mystyle.css”>.

Internal CSS: -

In this type CSS is written between <head> tag and <style> tag.

Some of the properties used in this are: -

|  |  |
| --- | --- |
| Property | Description |
| background | Defines a variety of background properties within one declaration. |
| background-attachment | Specify whether the background image is fixed in the viewport or scrolls. |
| background-clip | Specifies the painting area of the background. |
| background-color | Defines an elements background color |
| background-image | Defines an elements background image |
| border | Sets the width, style and color for all four sides of elements border. |
| border-radius | Defines the shape of the border corners of an element. |
| color | Specify the color of the text of an element. |
| opacity | Specifies the transparency of an element. |
| font-family | Defines a list of fonts for element. |
| font-stretch | Select a normal, condensed or expanded face from a font. |

**PHP 7**

**PHP** is a [general-purpose programming language](https://en.wikipedia.org/wiki/General-purpose_programming_language) originally designed for [web development](https://en.wikipedia.org/wiki/Web_development). It was originally created by [Rasmus Lerdorf](https://en.wikipedia.org/wiki/Rasmus_Lerdorf) in 1994, the PHP [reference implementation](https://en.wikipedia.org/wiki/Reference_implementation) is now produced by The PHP Group. PHP originally stood for Personal Home Page, but it now stands for the [recursive initialism](https://en.wikipedia.org/wiki/Recursive_initialism) PHP: Hypertext Pre-processor.

* PHP 7 is much faster than the previous popular stable release (PHP 5.6)
* PHP 7 has improved Error Handling
* PHP 7 supports stricter Type Declarations for function arguments
* PHP 7 supports new operators (like the spaceship operator: <=>)

In this project PHP7 is used only for server side, and used to process the data and connect to database.

**MYSQL 8.0**

MySQL is an [open-source](https://en.wikipedia.org/wiki/Open-source_software) [relational database management system](https://en.wikipedia.org/wiki/Relational_database_management_system) (RDBMS). Its name is a combination of "My", the name of co-founder [Michael Widenius](https://en.wikipedia.org/wiki/Michael_Widenius)'s daughter, and "[SQL](https://en.wikipedia.org/wiki/SQL)", the abbreviation for [Structured Query Language](https://en.wikipedia.org/wiki/Structured_Query_Language).

In this project mysql is used to store data.

Data is stored using connection of php and mysql.

<?php  
$servername= "localhost";  
$username= "username";  
$password= "password";   
$conn= new mysql($servername,$username,$password);?>

This is the code for connection of php with database.

In database data is inserted deleted and select query is used

**JavaScript**

JavaScript is the programming language of HTML and the Web.

JavaScript is easy to learn.

This tutorial will teach you JavaScript from basic to advanced.

In HTML, JavaScript code must be inserted between <script> and </script> tags.

JavaScript can "display" data in different ways:

* Writing into an HTML element, using innerHTML.
* Writing into the HTML output using document.write().

JavaScript provides a way to validate form's data on the client's computer before sending it to the web server. Form validation generally performs two functions.

* **Basic Validation** − First of all, the form must be checked to make sure all the mandatory fields are filled in. It would require just a loop through each field in the form and check for data.
* **Data Format Validation** − Secondly, the data that is entered must be checked for correct form and value. Your code must include appropriate logic to test correctness of data.

**Input Events**

[onblur - When a user leaves an input field](https://www.w3schools.com/js/tryit.asp?filename=tryjs_events_onblur).

[onchange - When a user changes the content of an input field](https://www.w3schools.com/js/tryit.asp?filename=tryjs_events_onchange).

[onchange - When a user selects a dropdown value](https://www.w3schools.com/js/tryit.asp?filename=tryjs_events_dropdown).

[onfocus - When an input field gets focus](https://www.w3schools.com/js/tryit.asp?filename=tryjs_events_onfocus).

[onselect - When input text is selected](https://www.w3schools.com/js/tryit.asp?filename=tryjs_events_onselect).

onsubmit - When a user clicks the submit button.

[onreset - When a user clicks the reset button](https://www.w3schools.com/js/tryit.asp?filename=tryjs_events_onreset).

**Click Events**

[onclick - When button is clicked](https://www.w3schools.com/js/tryit.asp?filename=tryjs_events_onclick).

[ondblclick - When a text is double-clicked](https://www.w3schools.com/js/tryit.asp?filename=tryjs_events_ondblclick)

**Load Events**

[onload - When the page has been loaded](https://www.w3schools.com/js/tryit.asp?filename=tryjs_events_body_onload).

[onload - When an image has been loaded](https://www.w3schools.com/js/tryit.asp?filename=tryjs_events_img_onload).

**CHAPTER 2: PROPOSED SYSTEM**

**2.1 PROPOSED SYSTEM:**

This system will provide bunch of benefits from various point of views. As this online system of "Friends Corner" allows the end users to surf through the food item list of their choice from the menu list, and order food items online, and allows user to register to the system.

The payment can be made through COD and the food items will be delivery depending upon the customer’s choice and convenience. The selection food items made by the customers will be available online to the owner of cafe or to the person handling the system.

As soon as the owner receives the order, then he will check if the food item is available or not or the delivery boy is available or not if everything is ok then later person the delivery persons is assigned with the address of customer else the order will be cancelled and appropriate message will be given to the customer.

Multiple food items can be ordered online at a time and it will be added to the cart. The system will be taking orders only when cafe is open. The system will be more secure as each time when the users tries to login, he should provide credentials to the system. The system will maintain the information of customers and about their orders in database

With this system the work load of the cafe owner and the customer is reduced or in some situations the work is abolished.

One of the various benefits of this is system from customer point of view is that the customer saves his/her time and can get fresh food items at given time.

As the bill is generated online customer as well as owner will see bill within few seconds of selection of food items and both can view their order list.

This system is more flexible for customer as well owner as it is user-friendly, less time consuming and helpful.

**2.2 OBJECTIVE OF PROJECT**

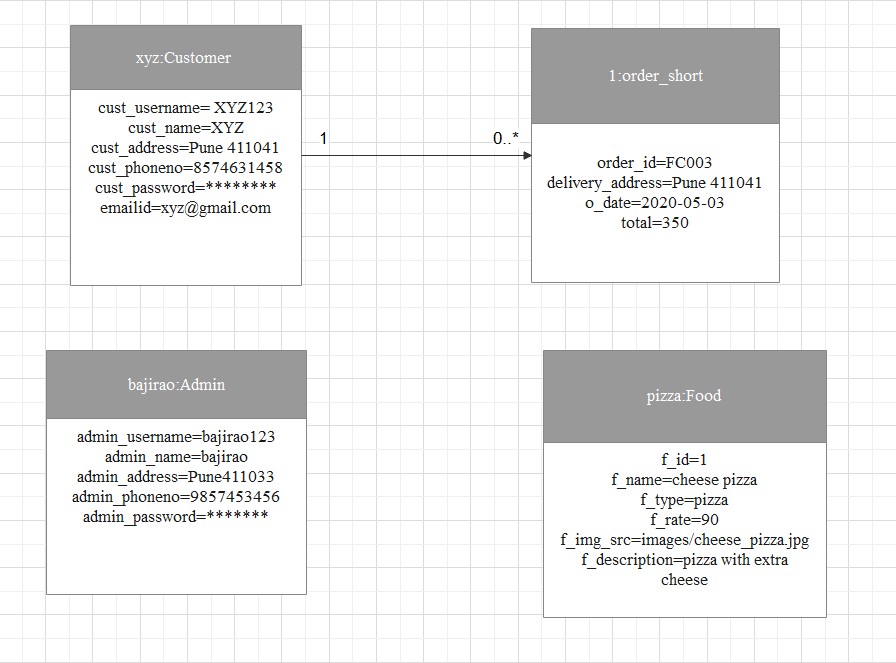
1. For Customer Surfing or searching through food items facility is provided.
2. To provide facility of ordering food items online.
3. To provide facility of selection of multiple food items at a time which will be added to the cart.
4. To provide Registration and login facility.
5. After ordering food items, the bill will be generated.
6. To maintain record of orders.

**2.3 User Requirements:**

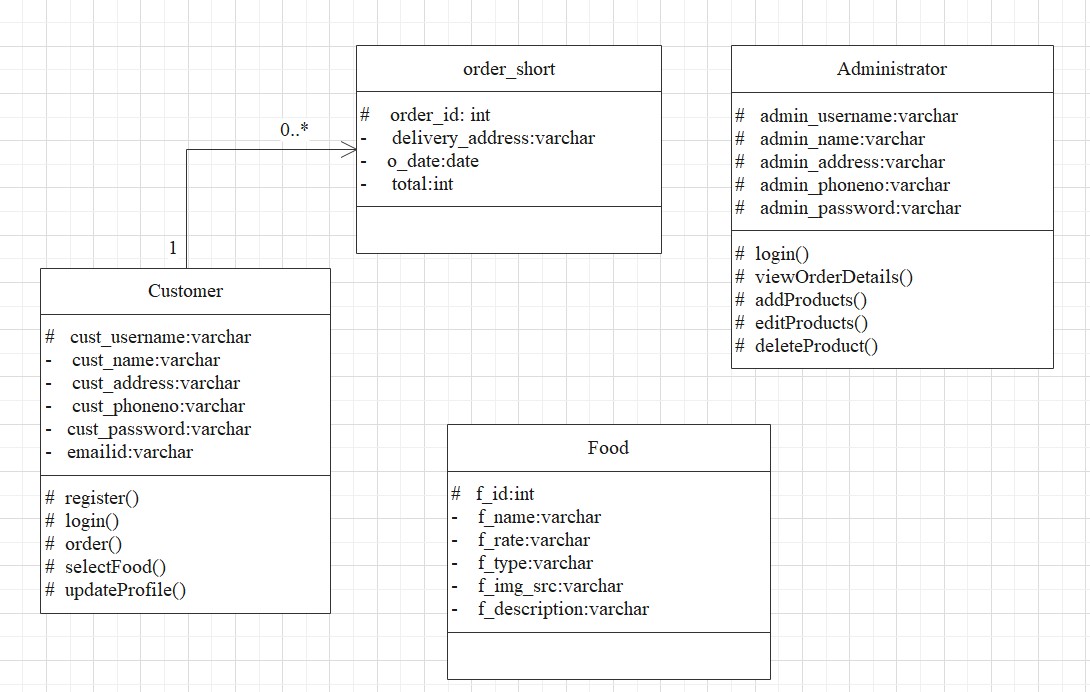
1. User Request that the order will be accepted and delivered whenever the delivery guy is available.
2. The User requested to differentiate between food types
3. User Requested that the order will be accepted and delivered whenever the offline shop is opened.
4. User requested that every food item will be charged extra 10Rs than offline rates.
5. User is Interested in COD (Cash on delivery) facility.

**CHAPTER 3: ANALYSIS & DESIGN**

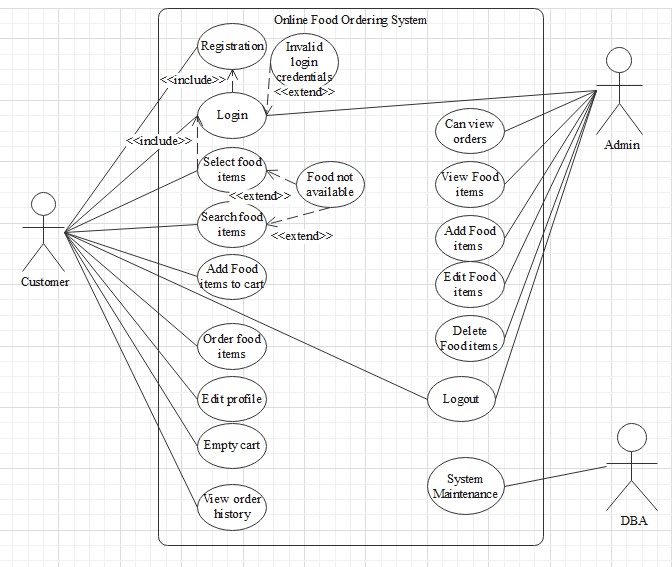
**3.1 Object Diagram:**

****

**3.2 Class Diagram:**

****

**3.3 Usecase Diagram and Usecase Description:**

****

**USER:**

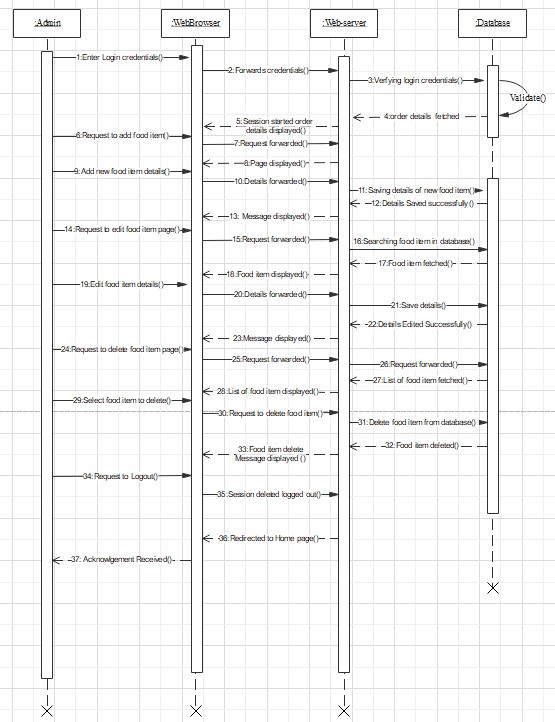
* New registration: The user must be registered in the system.
* Login: To order user must be logged in.
* Select Food Item: User can select multiple food items at a time and add it to the cart and order it.
* Search Food Item: User can search a particular food item.
* View /update login details: Users can view their details and update his details if required.
* View Order: User can view his orders history.

**ADMIN:**

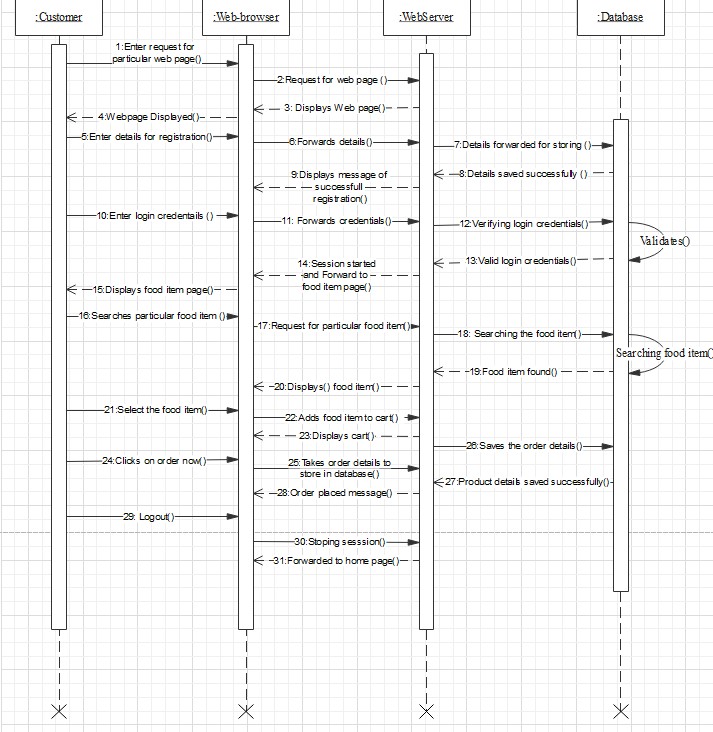
* Login: Admin must be logged in.
* View orders: Admin can view the orders of the customer.
* View food item: Admin can view food items he is selling.
* Add food item: Admin can food item to sell.
* Edit food item: Admin can edit a particular or multiple food items.
* Delete food item: Admin can delete multiple food items at a time.
* Profile: Admin can also edit his details and password.

**3.4 Sequence Diagram:**

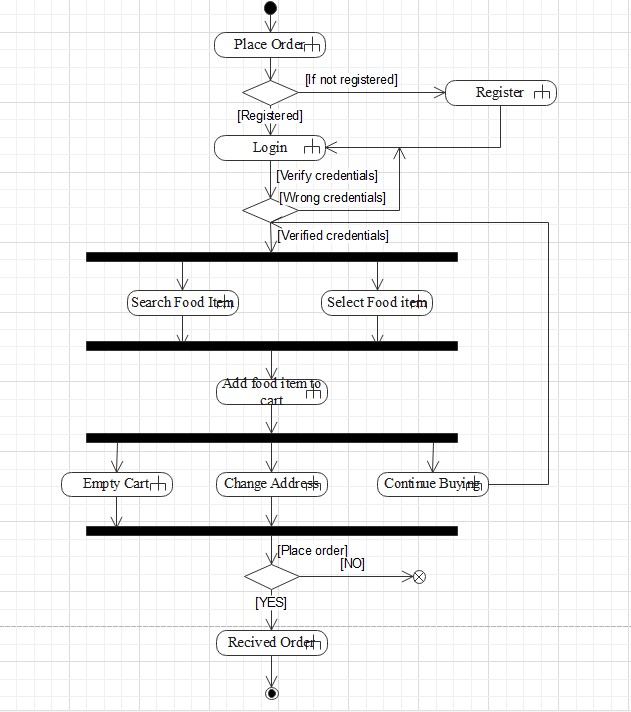
* Admin Side:



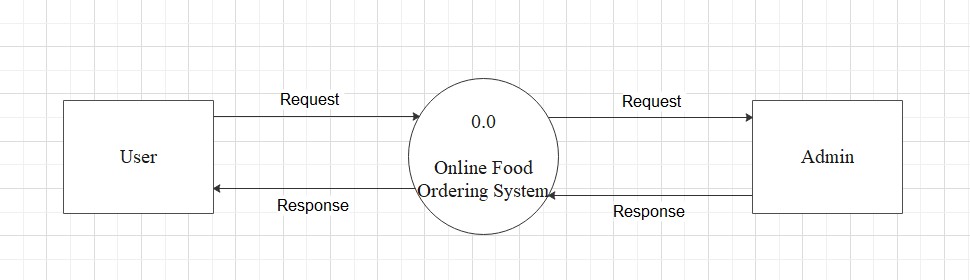
* Customer Side:



**3.5 Activity Diagram:**

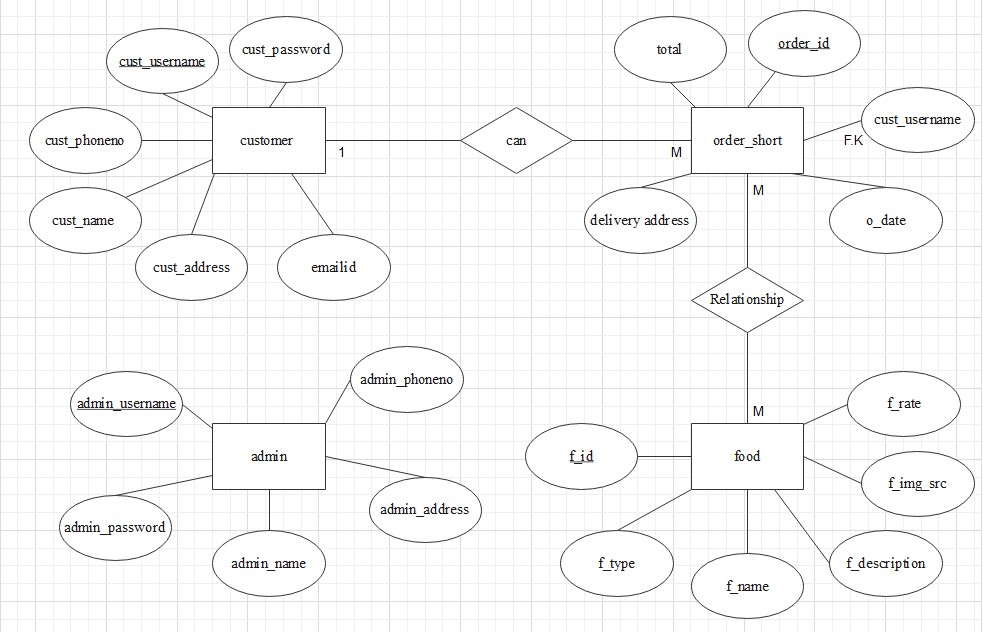


**3.6 Data Flow Diagram (DFD):**



**CHAPTER 4: DATBASE SPECIFICATIONS:**

4.1) **Entity Relationship Diagram (ERD):**



**4.2) Database Design (Normalized Database):**

1. TABLES:

* **admin**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table Name** | admin | | | |
| **Description of Table** | This includes Information of Admin | | | |
| **Primary or Candidate Key of Table** | admin\_username | | | |
| **Foreign Keys of Table and Association of Tables** | - | | | |
| * Sr.no | Field Name | Datatype | Width | Constraint |
| 1 | admin\_username | varchar | 30 | Primary Key |
| 2 | admin\_name | varchar | 50 | NOT NULL |
| 3 | admin\_address | varchar | 80 | NOT NULL |
| 4 | admin\_phoneno | varchar | 10 | NOT NULL |
| 5 | admin\_password | varchar | 20 | NOT NULL |

* **Customer**

|  |  |
| --- | --- |
| **Table Name** | customer |
| **Description of Table** | This includes Information of Customer |
| **Primary or Candidate Key of Table** | cust\_username |
| **Foreign Keys of Table and Association of Tables** | - |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr.no | Field Name | Datatype | Width | Constraint |
| 1 | cust\_username | varchar | 30 | Primary Key |
| 2 | cust\_name | varchar | 50 | NOT NULL |
| 3 | emailid | varchar | 30 | NOT NULL |
| 4 | cust\_address | varchar | 80 | NOT NULL |
| 5 | cust\_phoneno | varchar | 10 | NOT NULL |
| 6 | cust\_password | varchar | 30 | NOT NULL |

* food

|  |  |
| --- | --- |
| **Table Name** | food |
| **Description of Table** | This includes Information of food the Admin is giving. |
| **Primary or Candidate Key of Table** | f\_id |
| **Foreign Keys of Table and Association of Tables** | - |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr.no | Field Name | Datatype | Width | Constraint |
| 1 | f\_id | int | 11 | Primary Key |
| 2 | f\_name | varchar | 35 | NOT NULL |
| 3 | f\_rate | int | 11 | NOT NULL |
| 4 | f\_type | varchar | 30 | NOT NULL |
| 5 | f\_img\_src | varchar | 35 | NOT NULL |
| 6 | f\_description | varchar | 100 | NOT NULL |

* order\_short

|  |  |
| --- | --- |
| **Table Name** | admin |
| **Description of Table** | This includes Information of Orders given by customer |
| **Primary or Candidate Key of Table** | order\_id |
| **Foreign Keys of Table and Association of Tables** | cust\_username from customer table. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr.no | Field Name | Datatype | Width | Constraint |
| 1 | order\_id | int | 11 | Primary Key |
| 2 | delivery\_address | varchar | 80 | NOT NULL |
| 3 | o\_date | date |  | NOT NULL |
| 4 | total | int | 11 | NOT NULL |
| 5 | cust\_username | Varchar | 30 | Foreign Key |

* order\_details

|  |  |
| --- | --- |
| **Table Name** | order\_details |
| **Description of Table** | This includes Information of Orders details and order with their respective Foods. |
| **Primary or Candidate Key of Table** | - |
| **Foreign Keys of Table and Association of Tables** | order\_id from order\_short table.  f\_id from food table. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr.no | Field Name | Datatype | Width | Constraint |
| 1 | order\_id | int | 11 | Foreign key |
| 2 | f\_id | int | 11 | Foreign key |
| 3 | quantity | int | 11 | NOT NULL |
| 4 | amt | int | 11 | NOT NULL |

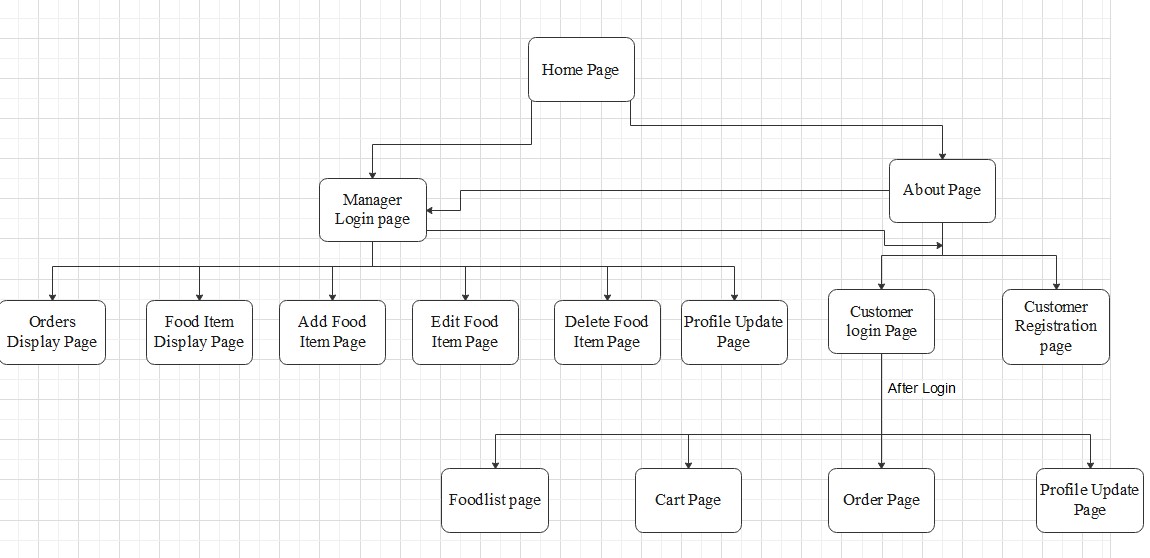
**4.3)Data Dictionary:-**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr.no | Field Name | Datatype | Width | Constraint | Description |
| 1 | admin\_address | varchar | 80 | NOT NULL | No Reference |
| 2 | admin\_name | varchar | 50 | NOT NULL | No Reference |
| 3 | admin\_password | varchar | 20 | NOT NULL | No Reference |
| 4 | admin\_phoneno | varchar | 10 | NOT NULL | No Reference |
| 5 | admin\_username | varchar | 30 | Primary Key | Primary key in admin table.  No Reference |
| 6 | amt | int | 11 | NOT NULL | No Reference |
| 7 | cust\_address | varchar | 80 | NOT NULL | No Reference |
| 8 | cust\_name | varchar | 50 | NOT NULL | No Reference |
| 9 | cust\_password | varchar | 30 | NOT NULL | No Reference |
| 10 | cust\_phoneno | varchar | 10 | NOT NULL | No Reference |
| 11 | cust\_username | varchar | 30 | Primary Key | Primary key in customer table.  Reference to order\_short table. |
| 12 | delivery\_address | varchar | 80 | NOT NULL | No Reference |
| 13 | emailid | varchar | 30 | NOT NULL | No Reference |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr.no | Field Name | Datatype | Width | Constraint | Description |
| 14 | f\_description | varchar | 100 | NOT NULL | No Reference |
| 15 | f\_id | int | 11 | Primary Key | Primary key in food table.  Reference to order\_details table. |
| 16 | f\_img\_src | varchar | 35 | NOT NULL | No Reference |
| 17 | f\_name | varchar | 35 | NOT NULL | No Reference |
| 18 | f\_type | varchar | 30 | NOT NULL | No Reference |
| 19 | f\_rate | int | 11 | NOT NULL | No Reference |
| 20 | order\_id | int | 11 | Primary Key | Primary key in order\_short table. Reference to order\_details table. |
| 21 | o\_date | date |  | NOT NULL | No Reference |
| 22 | quantity | int | 11 | NOT NULL | No Reference |
| 23 | total | int | 11 | NOT NULL | No Reference |

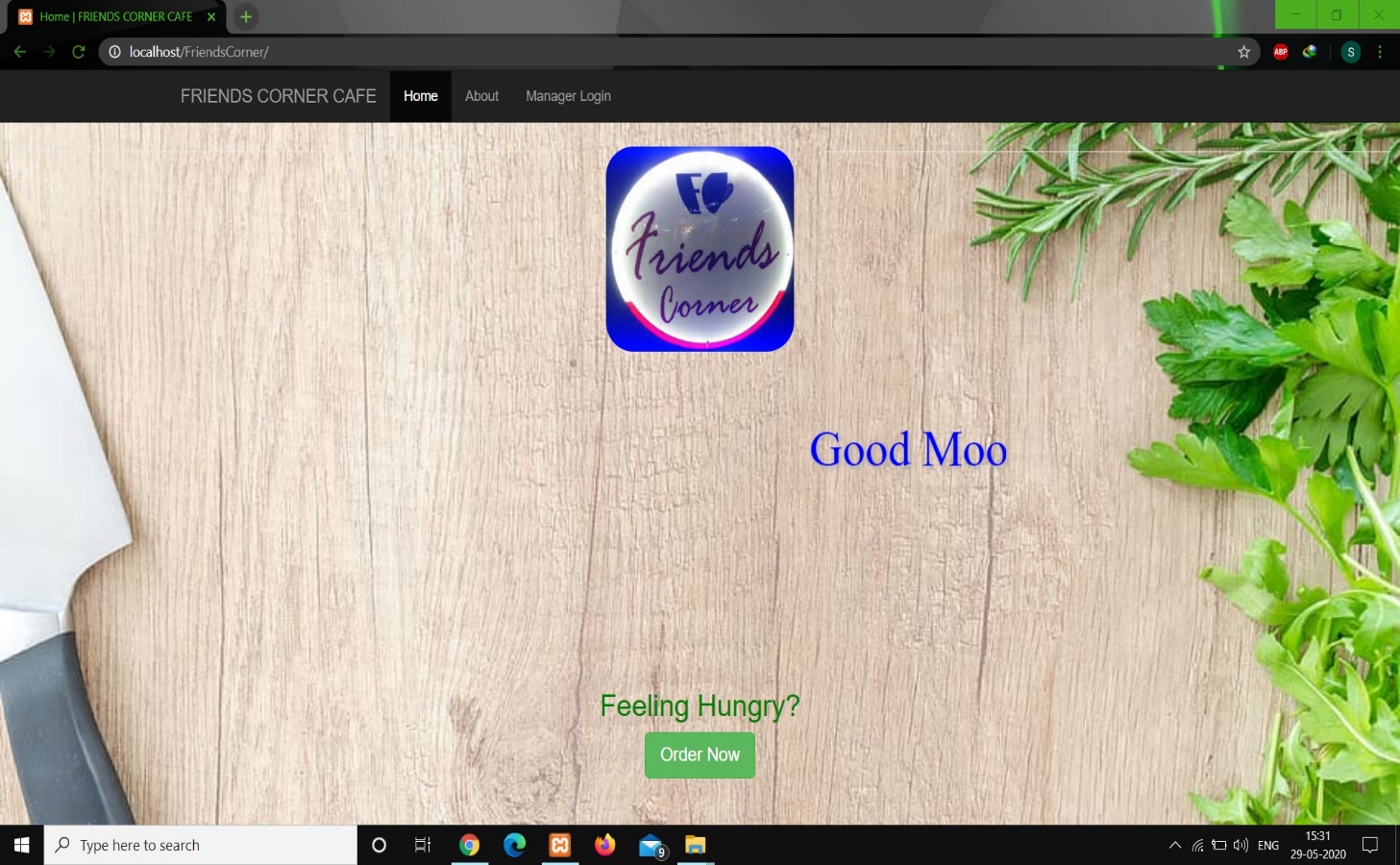
**CHAPTER 5 USER INTERFACE SPECIFICATIONS:**

**5.1 Menu Tree:**

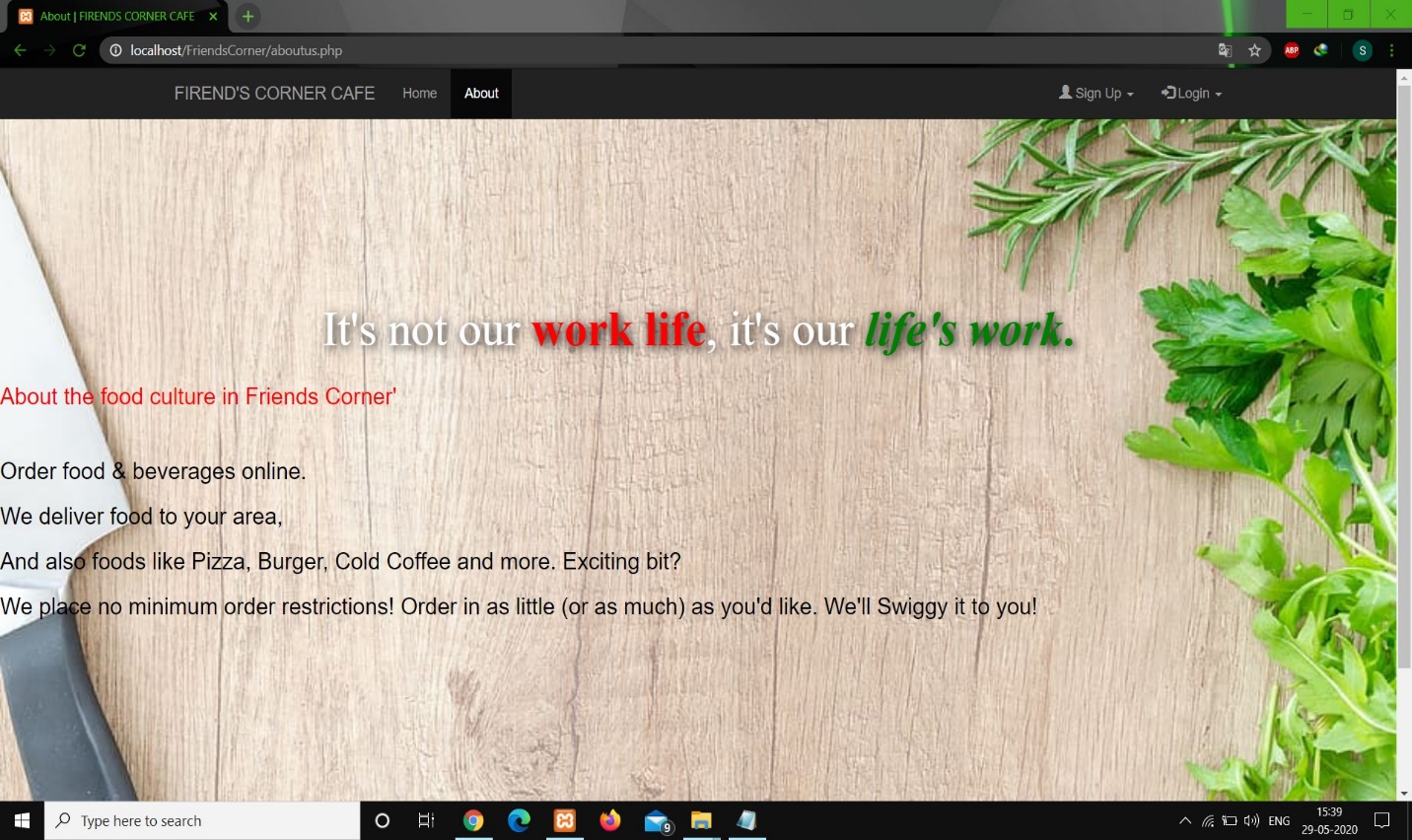


**5.2 Input Screens:**

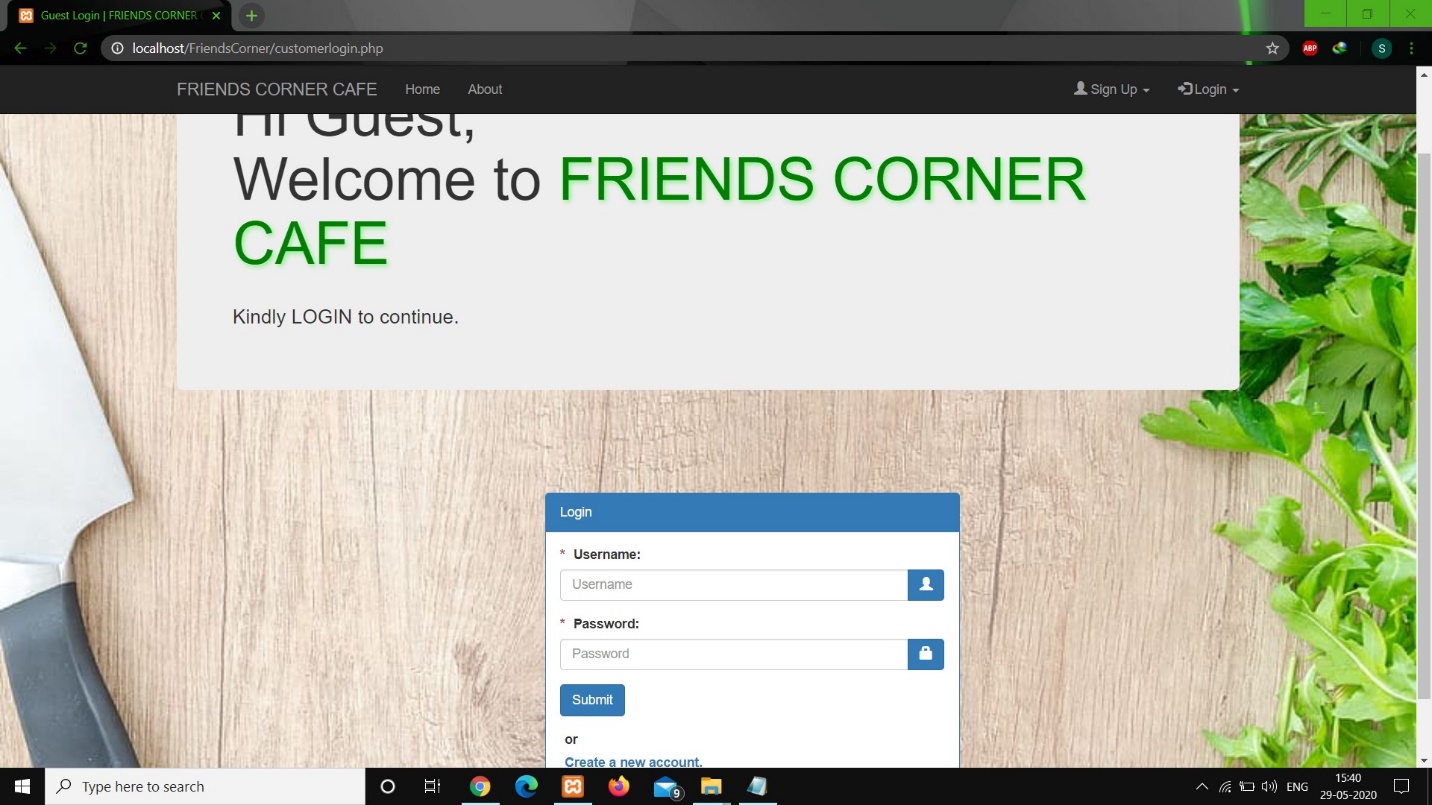
1. Home Page:



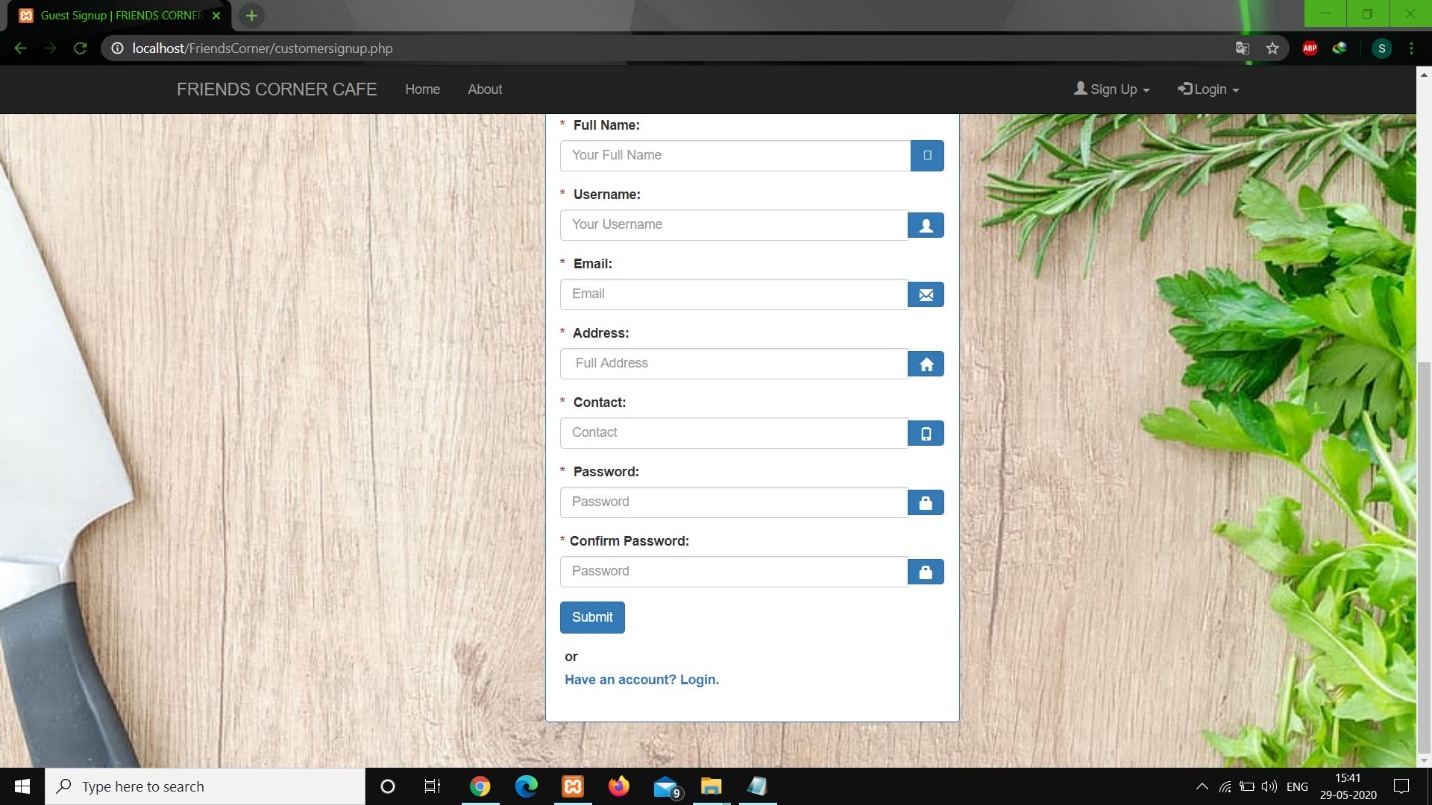
1. About Page:



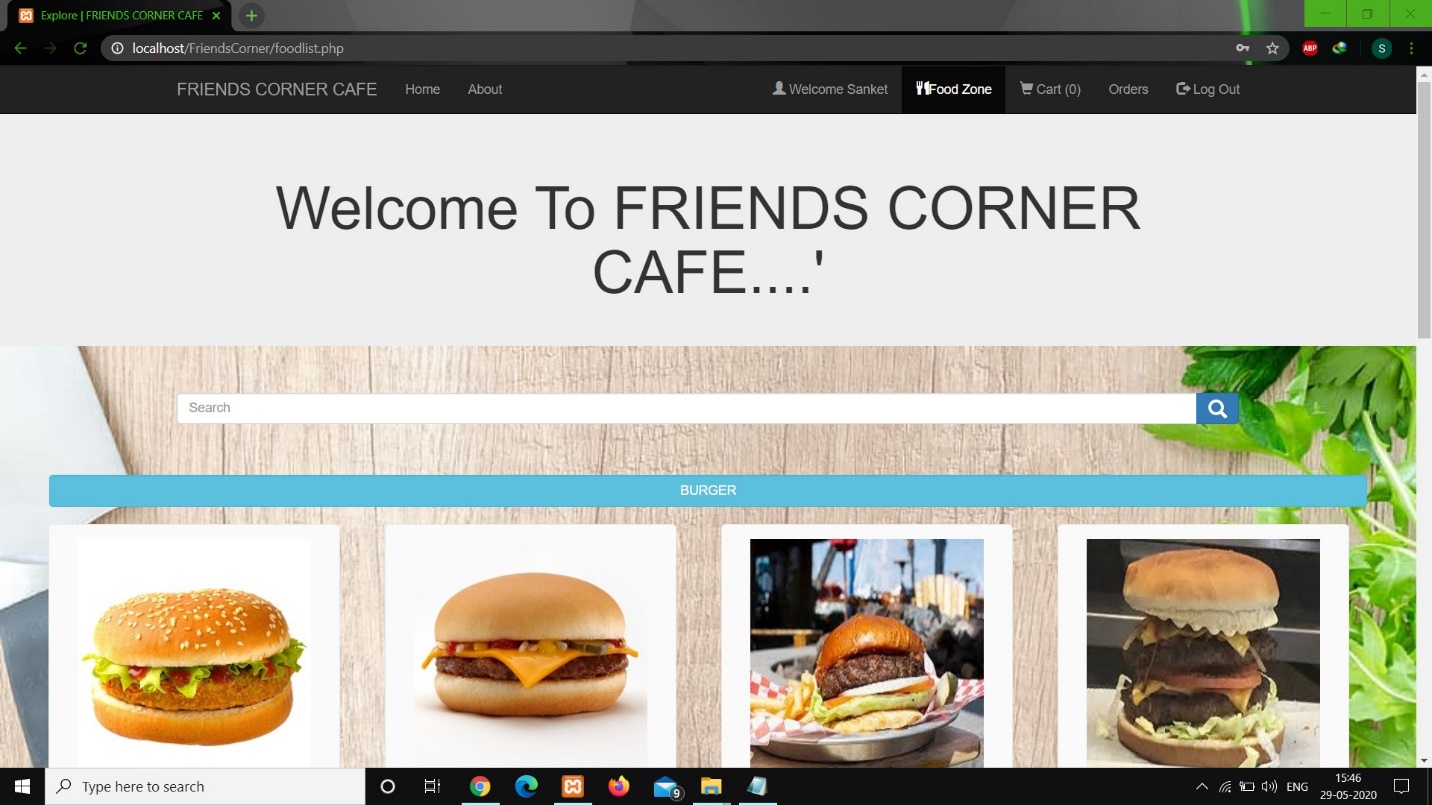
1. Customer Login Page:



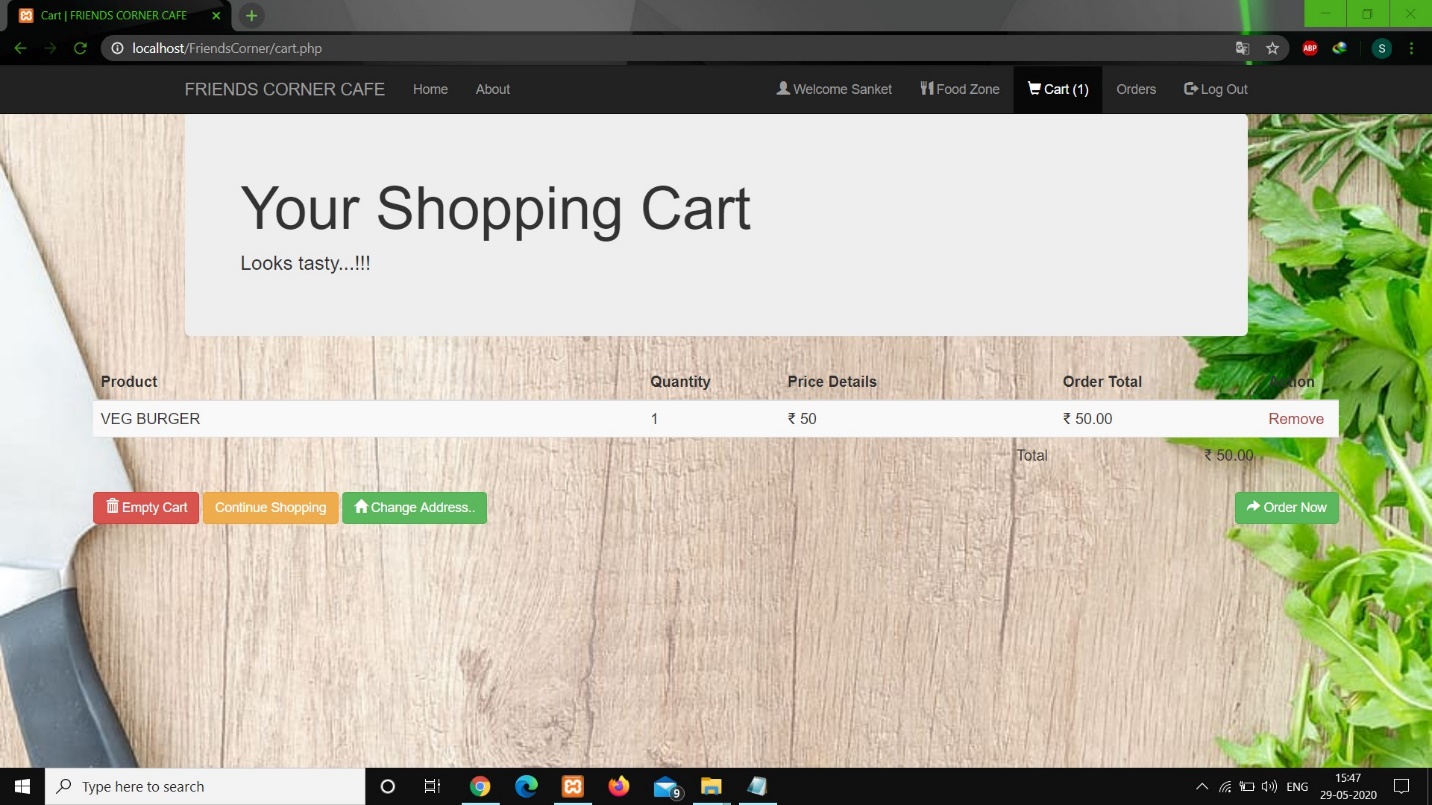
1. Customer Signup:



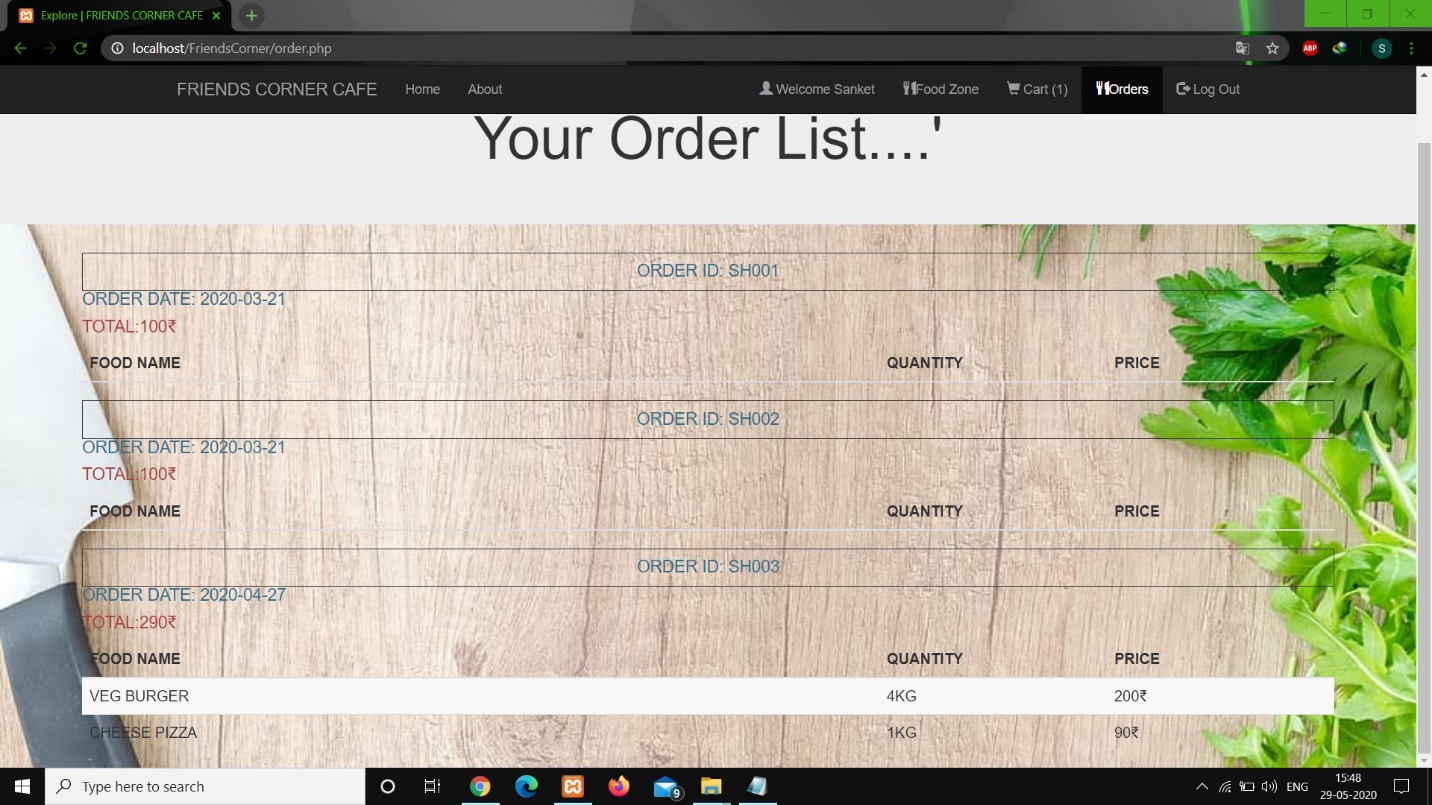
1. Main Food List Page:



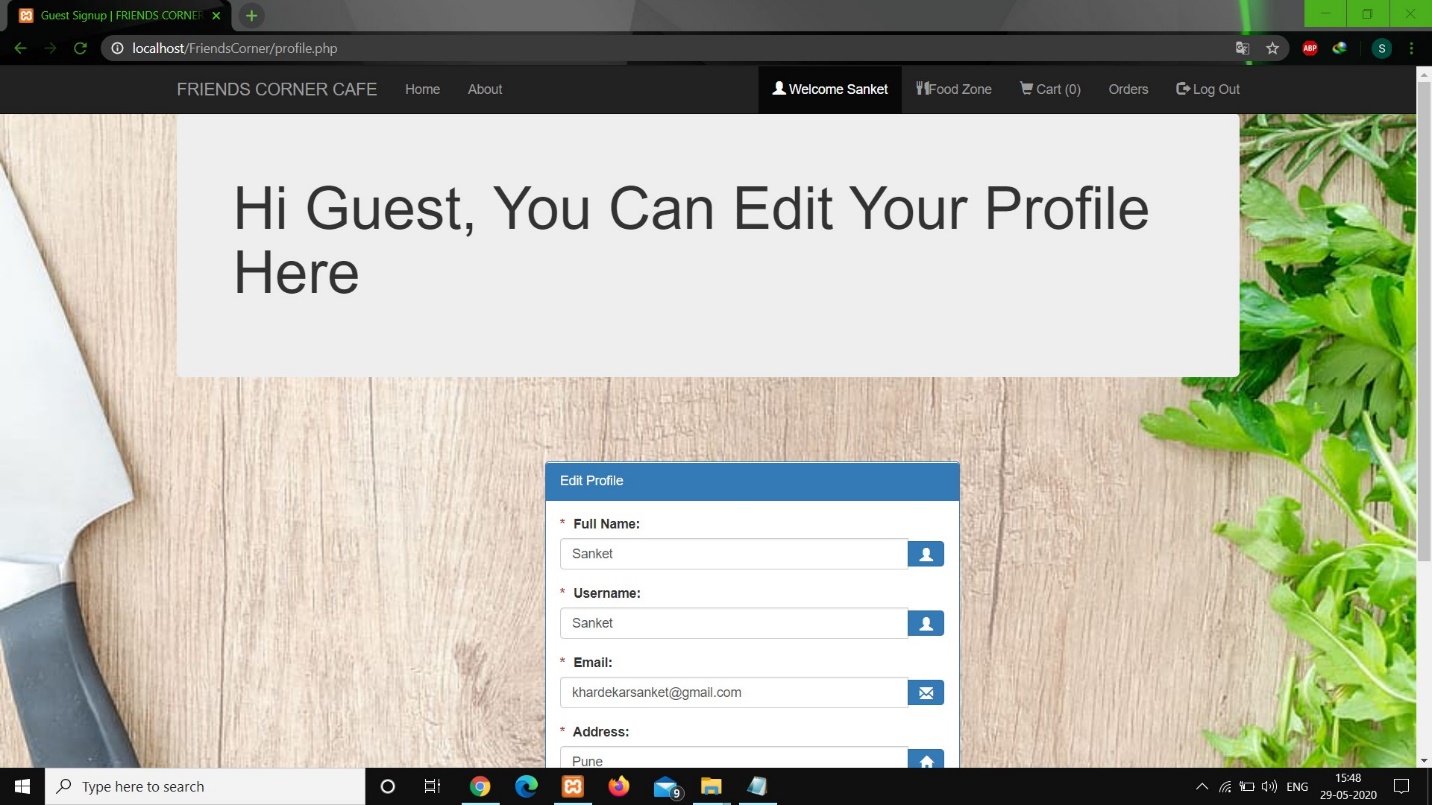
1. Cart Page:



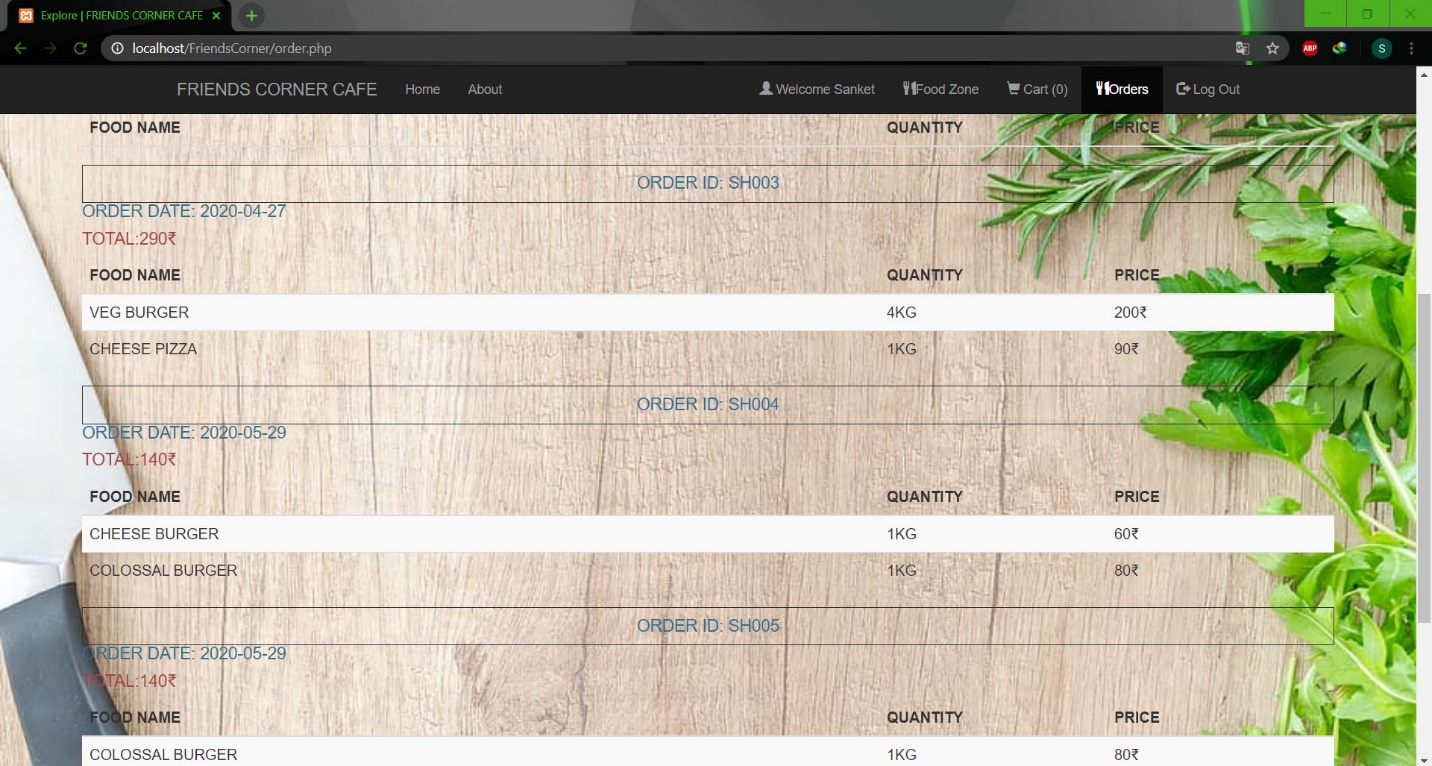
1. Order Page:



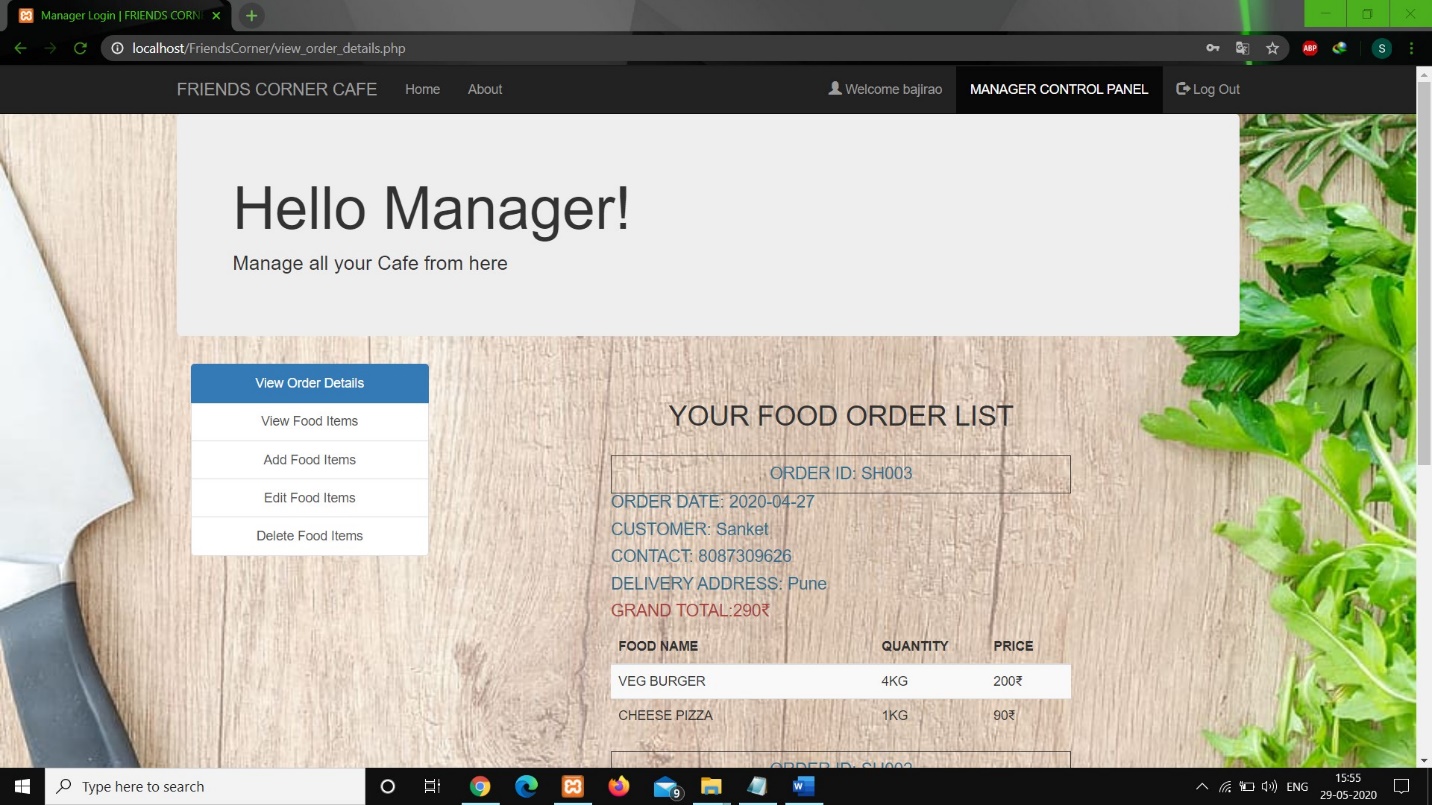
1. Profile Update Page:



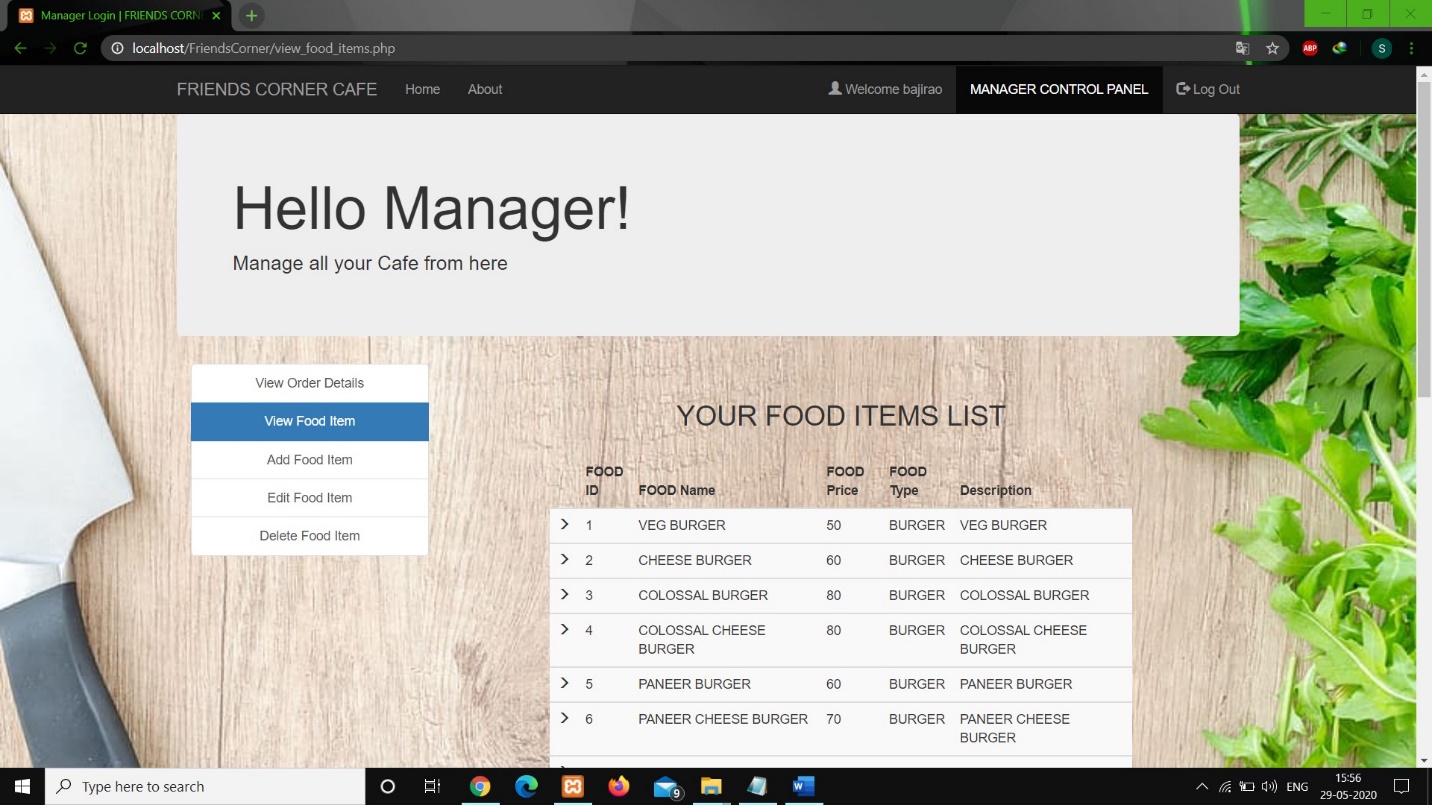
1. Manager Login Page:



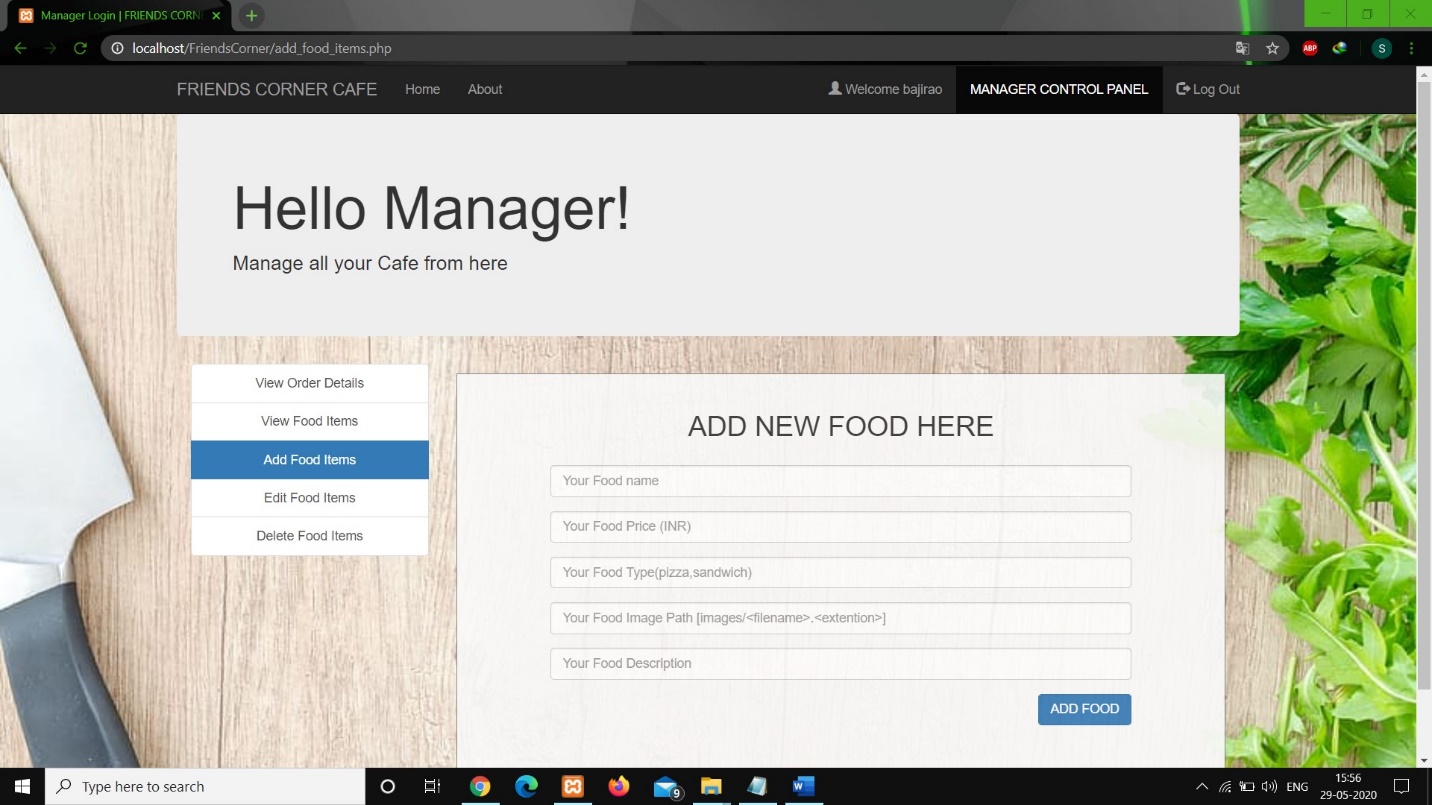
1. Admin Side Orders Page:



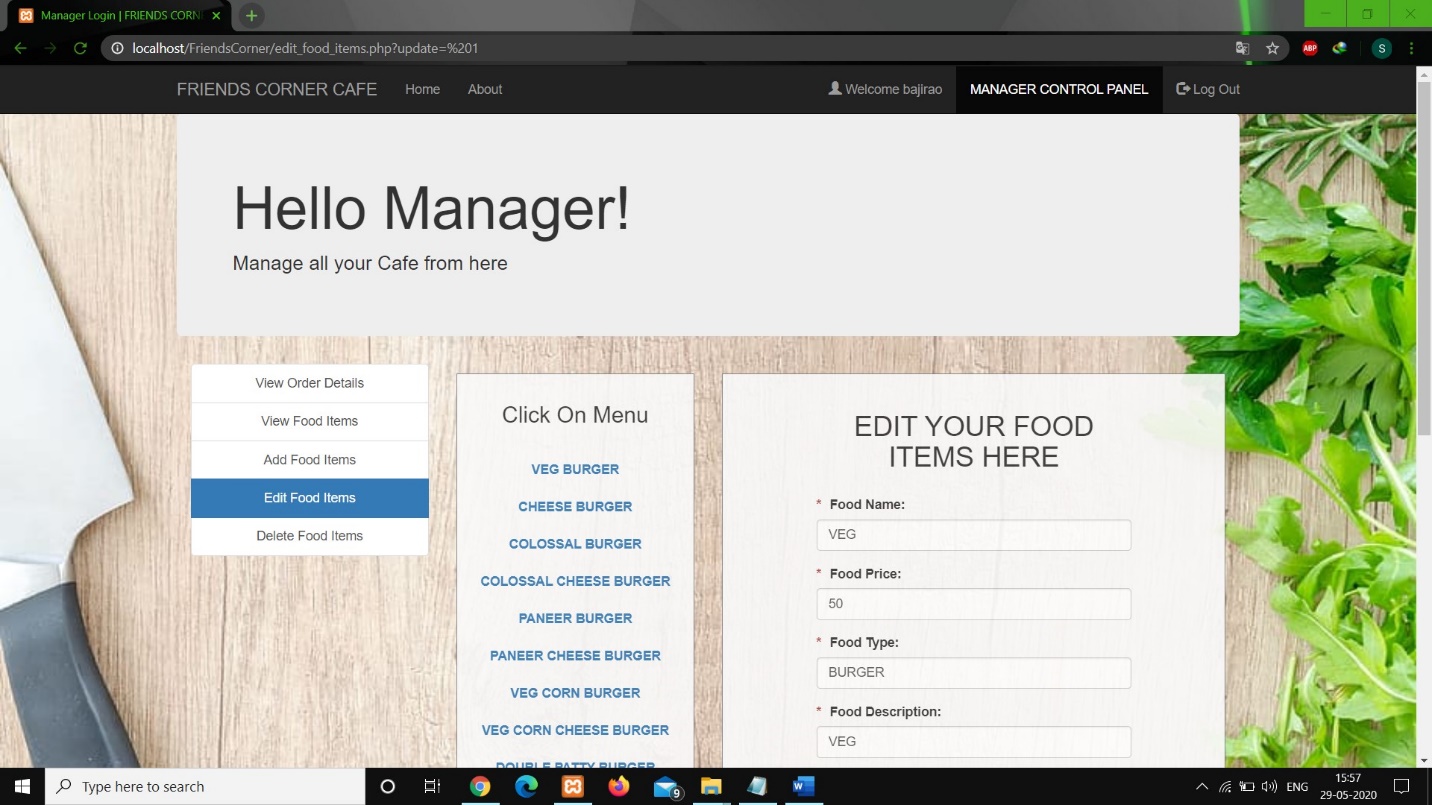
1. Admin Side View Food Item Page:



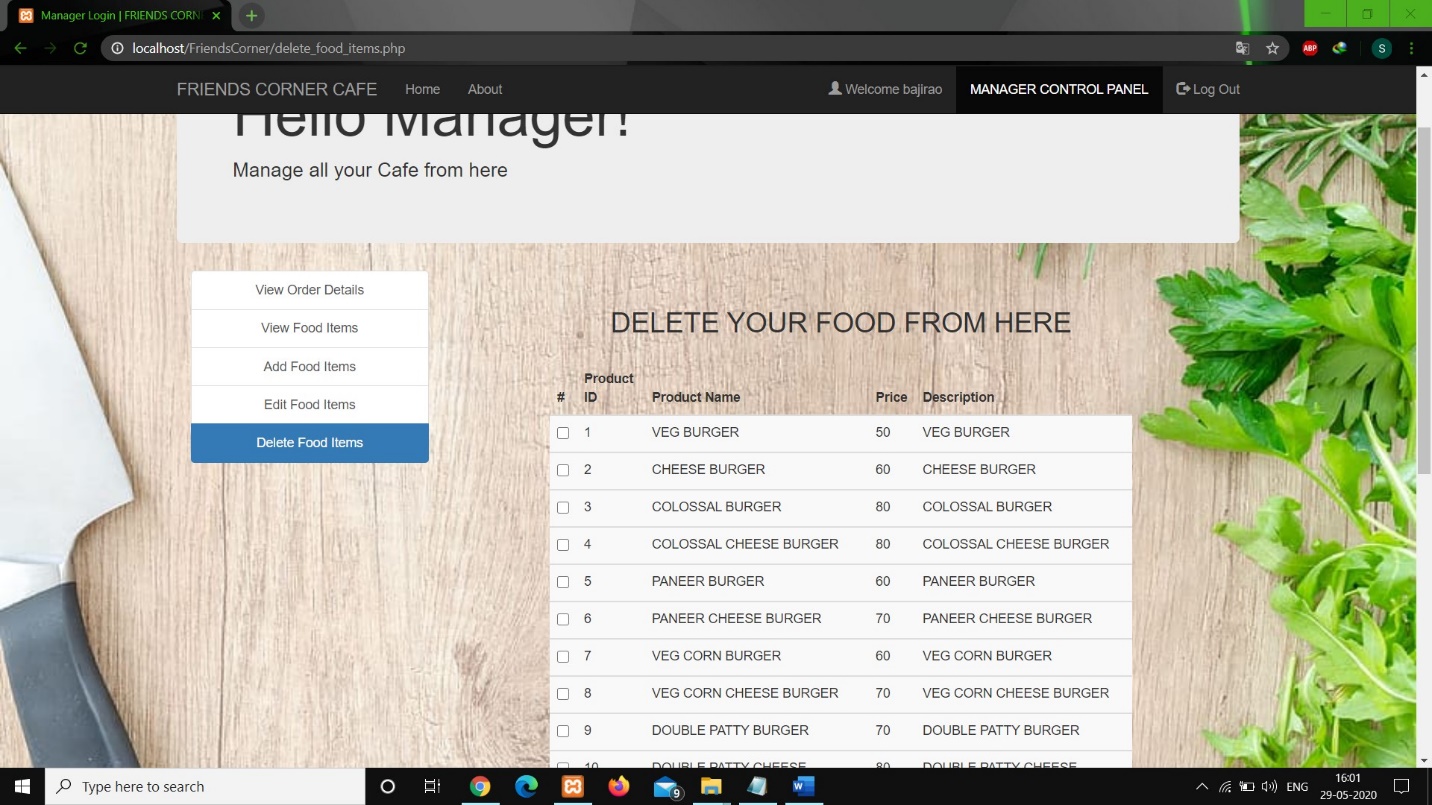
1. Admin Side Add Food Item Page:



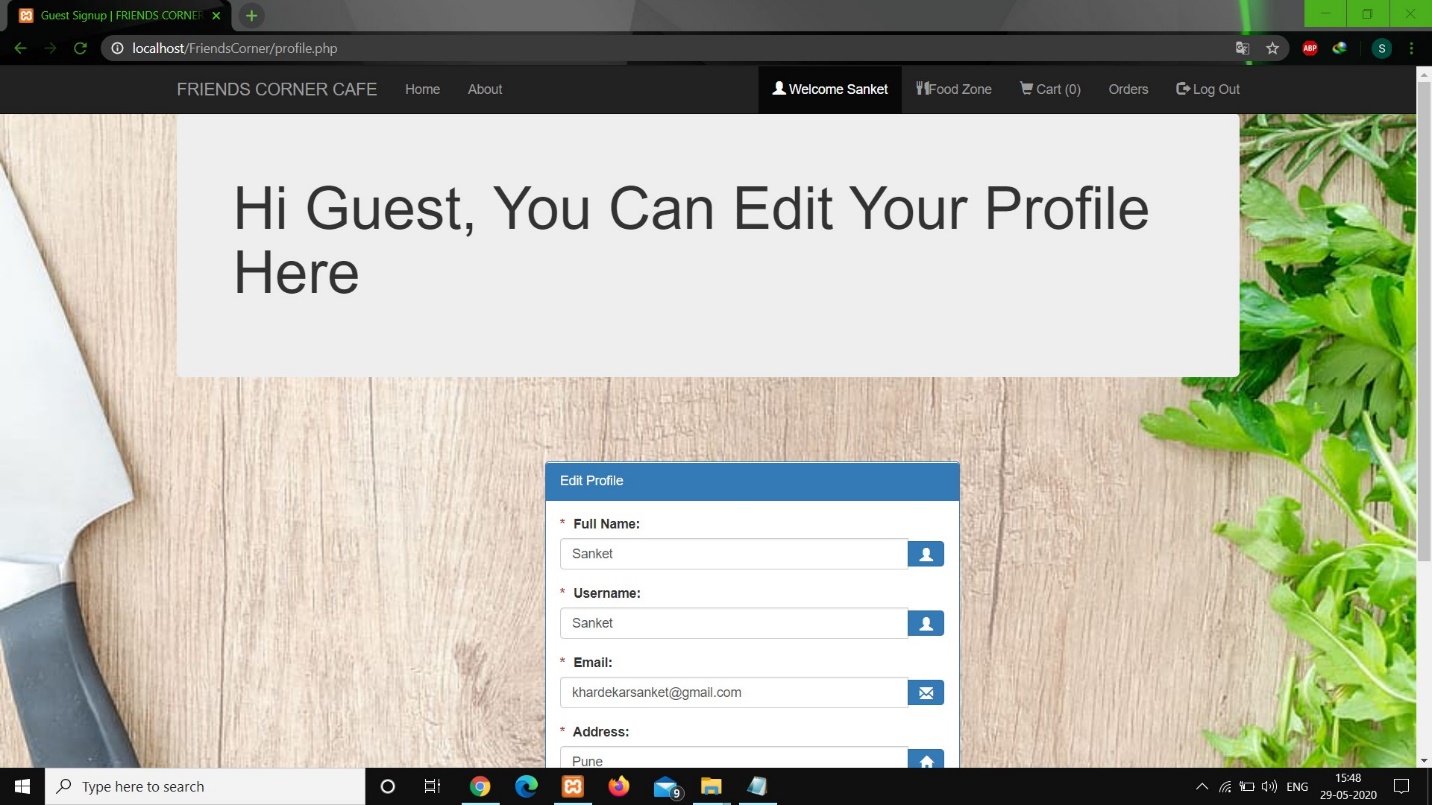
1. Admin Side Edit Food Item:



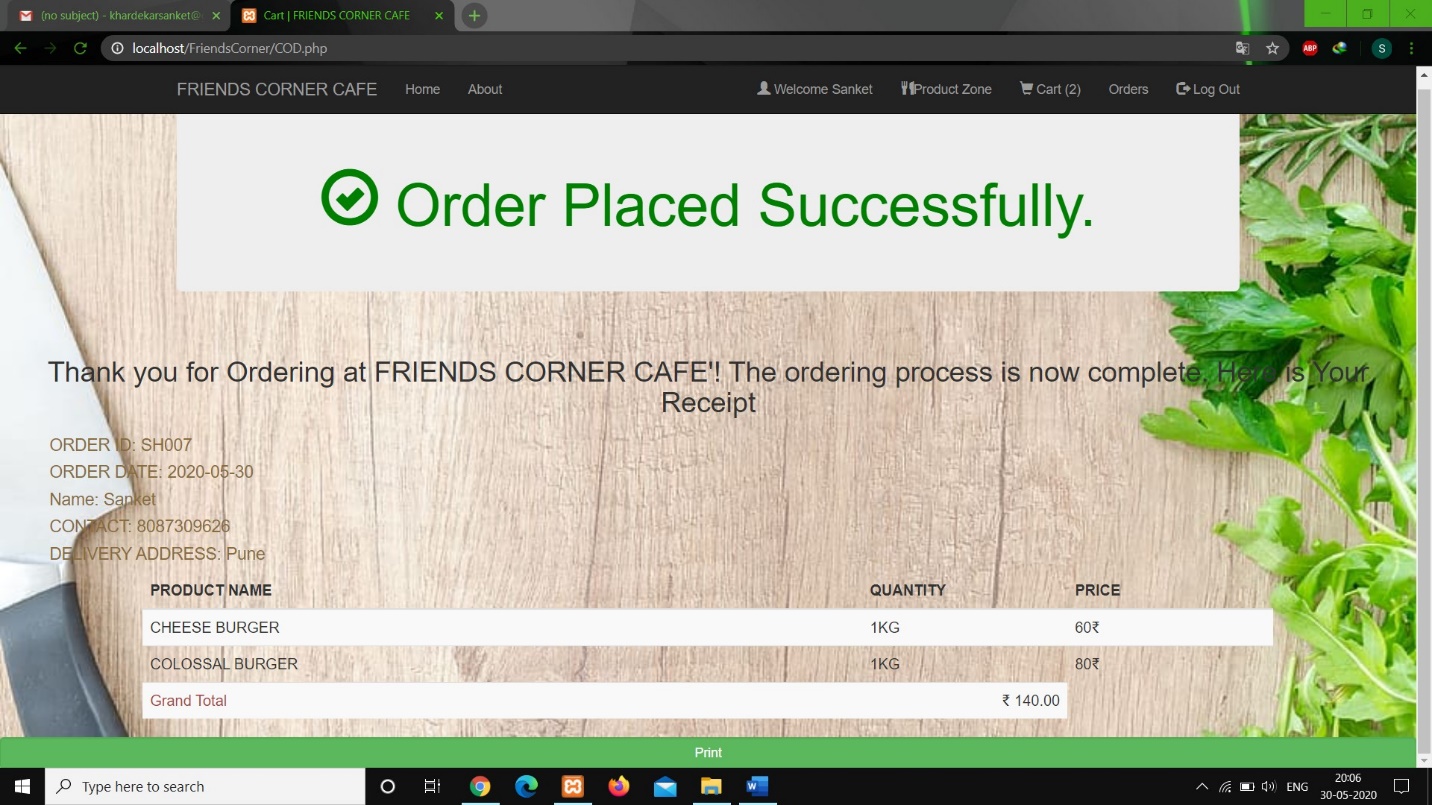
1. Admin Side Delete Food Item Page:

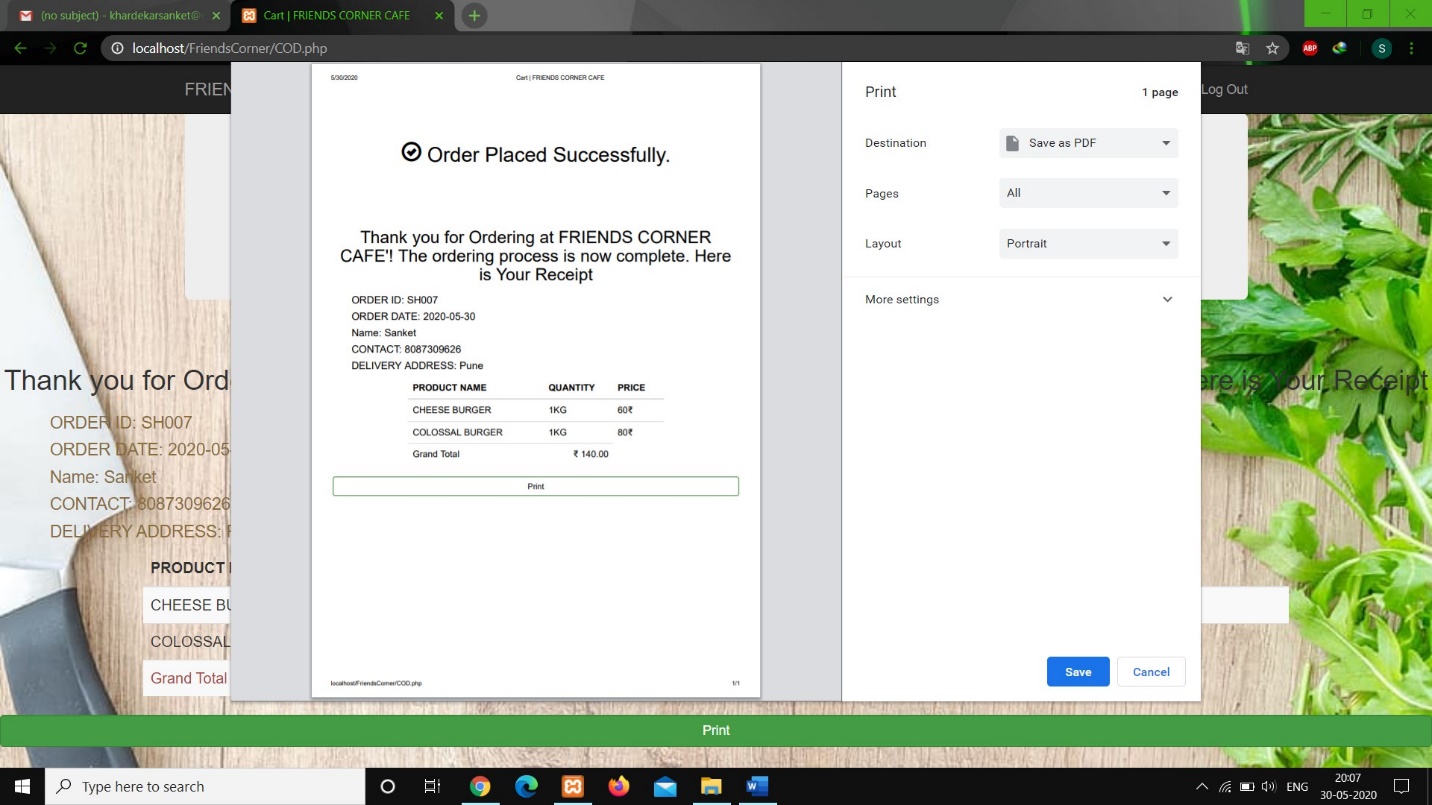


1. Admin Profile Update Page:



**5.3 Report:**

****

****

**5.4 Test Cases:**

1. Login for Both Admin and Customer:

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case | Test Data | Expected Output | Actual Output |
| Username and password not provided | Username: \_\_\_\_\_\_  Password: \_\_\_\_\_\_ | Please fill out fields. | Please fill out fields. |
| Username provided and password not provided. | Username: Rahul  Password: \_\_\_\_\_\_ | Please fill out field. | Please fill out field. |
| Username not provided and password provided. | Username: \_\_\_\_\_  Password: \*\*\*\*\*\*\*\* | Please fill out field. | Please fill out field. |
| Enter invalid username and password. | Username: Rahul  Password: \*\*\*\*\*\* | Username or Password is invalid | Username or Password is invalid |
| Enter valid username and password. | Username: Rahul  Password: \*\*\*\*\*\*\* | Login successful opens food list page | Login successful opens food list page |

1. Registration:

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case | Test Data | Expected Output | Actual Output |
| All fields Empty | - | Please fill out fields. | Please fill out fields. |
| Any single or multiple fields empty | - | Please fill out this field/fields. | Please fill out this field/fields. |
| Entering number in Full name field | Full name: 1 | No output displayed.  E.g.:  Full name: \_\_\_\_\_\_ | No output displayed.  E.g.:  Full name: \_\_\_\_\_\_ |
| Pressing Space in Username, Contact, Password, Confirm Password field | - | No Space displayed. | No Space displayed. |
| Email Validation | Email id: rahul | Please include @ and ‘.’ in emailed. | Please include @ and ‘.’ in emailed. |
| Entering character in contact field | Contact: d678987654 | No output displayed.  E.g.:  Contact: | No output displayed.  E.g.:  Contact: |
| Not entering 10-digit number in contact. | Contact: 8976546 | Alert: Mobile Number must have size 10 Please Check it!! | Alert: Mobile Number must have size 10 Please Check it!! |
| Password and confirm password different. | Password: \*\*\*\*\*\*\*\*\*  Confirm Password: \*\*\*\*\*\*\* | Alert:  Please Enter Same Password. | Alert:  Please Enter Same Password. |
| Password having less than 8 characters | Password: \*\* | Alert: Password Must Contain minimum 8 Character. | Alert: Password Must Contain minimum 8 Character. |
| Username having less than 10 characters | Username: Rahul | Alert: Username must have 10 character. | Alert: Username must have 10 character. |

Note: Above Similar Test Cases will be applied for Profile updating for both Customer and Admin.

**CHAPTER 6: USER MANUAL:**

**User Manual:**

User Module:

User can visit number of pages through one navigation bar.

Various facilities are provided to user from Registration to Selection to Logout.

Registration:

1. Various details are needed to be filled for registration.

Login:

1. User has to provide proper credentials to be logged in
2. If proper credentials are not given then error message is given.

Food Item List:

1. Once you have logged in you can select, search food item and add in cart.
2. Once you have selected food item you can place order and receipt will be generated as payment method is only COD.

Log out:

1. Once user has finished with ordering user can logout by clicking on logout button.

Admin Module:

Admin can visit number of pages through one navigation bar.

Login:

1. Admin has to provide proper credentials to be logged in.
2. If proper credentials are not inputted then error message is given.

Admin Dashboard:

1. Admin can update, add, delete food items as well as can view order details like user.

Log out:

1. Once user have finished with ordering user can logout by clicking on logout button.

**Limitation of the system:**

The current system has certain limitations listed below:

1) COD (Cash on delivery) facility is the only facility available for payment.

2) Weekly or Monthly report generation for admin is not available.

3) Admin has to accept all the order as he has to facility to cancel the order.

4) User can’t be notified through any of the notification till order is placed.

5) Customer can’t add the same food item to the cart twice.

**Proposed Enhancement:**

System can be made more effective by adding more features like:

1) More payment gateways can be provided to the customer as per customer’s needs.

2) Admin can view the report of sale monthly or weekly.

3) Accept or Decline button will be provided to Admin so that he can accept the order or decline it.

4) User can be notified by SMS or through any notification about their order status.

5) Customer can register through social media handle.

**Conclusion:**

The main objective of this project was to provide facility to customer of ordering the food items online. We had made the user-friendly by adding many features into it.

At the end it is concluded given stress on following points:

1. System provides facility to customer of online ordering.
2. System is very much authenticated by providing registration and login.
3. System has facility of selecting multiple food items at a time and prevents from adding duplicate food item in the cart.
4. System will provide order list to customer as well as admin.
5. Proper receipt will be generated and can be given to the customer.

This system has made us to work on different aspects like from requirement gathering to testing.

**Bibliography:**

1. [www.tutorialspoint.com](http://www.tutorialspoint.com)
2. [www.wikipedia.com](http://www.wikipedia.com)

1. [www.w3schools.com/php/php\_mysql\_intro.asp](http://www.w3schools.com/php/php_mysql_intro.asp)
2. <https://www.phptpoint.com/>
3. <https://javascript.info/>