CANARA COLLEGE

MANGALORE-575003

A Project Report On

“ART OF CAKES”

“ONLINE CAKE SHOPPING”

Submitted to the Mangalore University in Partial fulfilment of the requirement

for the award of the degree of

Bachelor of Computer Application

2021- 22

Submitted By

VARNA	**-** 191132722
H VIGNESH	**-** 191132729
KAVANA	**-** 191132730

Under the valuable Guidance of

Mrs. Rakshitha

DECLARATION

We hereby declare that this project work entitled **“ART OF CAKES - Online Cake Shopping”** has been prepared by us during the academic year 202122 under the guidance and supervision of Mrs. Rakshitha Lecturer in computer application and project guide, Canara College Mangalore in the partial fulfilment of the requirements for the award of Bachelor’s degree in the computer application from the University of Mangalore for the academic year 2021-22.

We also declare that this project is the outcome of our efforts and it has not been submitted to any other university for the award of any degree and diploma.

Team Members **Signature**

1. Varna - 191132722
2. H Vignesh - 191132729
3. Kavana - 191132730

ACKNOWLEDGEMENT

We present our project entitled **“ART OF CAKES -Online cake shopping”** with great pleasure and satisfaction we would like to thank the people who have supported us during the course of our project.

Working on the project was very interesting and enhanced our knowledge.

I express my sincere gratitude to our beloved principal, **Mrs. Premalatha V,** without whose permission I would not able to do my project and for taking keen interest for the student of BCA in providing useful guidelines and giving all the necessary facilities.

I extend thanks to our guide, **Mrs. Rakshitha**, for her able guidance and constant encouragement which help me in successfully completing our project.

Finally, I extend thanks to my parents and friends who were directly or indirectly involved in the completion of my project.

INDEX

SYNOPSIS

1.1 INTRODUCTION

Online Cake Order is a web-based application which enable users to order cakes through Online. The internet users are increasing rapidly, the company has introduced Online cake ordering system for getting the orders from users. This application not only improves user's experiences but also eases the workload on the employees of the cake shop.

It would be comfortable for the users to have an online cake ordering system. It would be free for users; they can select the cake they want and make payment for it. It will reduce the purchasing time for the users. This Cake Order Web application will help users in ordering the cakes. This system will surely enhance the image of the cake and user satisfaction will be more.

The user can order cake items through online. Before ordering cake, the user should add items to the cart. Items are added to the cart will be saved, so that the user can choose the other cakes as well. User can order cart items by entering checkout button. Even the system will ask delivery address or pick up date and time before making payment. After the payment, order details will be shown and confirmed.

1.1.1 Project Title

ART OF CAKES – ONLINE CAKE SHOPPING

1.1.2 Category

Web application and RDBMS.

1.1.3 Overview

The central concept of the application is to allow the user to shop virtually using the Internet and allow users to buy the items and articles of their desire from the store. The information pertaining to the products are stores on an RDBMS at the server side (store). The Server process the users and the items are shipped to the address submitted by them. The application was designed into two modules first is for the users who wish to buy the articles. Second is for the storekeepers who maintains and updates the information pertaining to the articles and those of the users. The end user of this product is a departmental store where the application is hosted on the web and the administrator maintains the database. The application which is deployed at the user database, the details of the items are brought forward from the database for the user view based on the selection through the menu and the database of all the products are updated at the end of each transaction. Data entry into the application can be done through various screens designed for various levels of users. Once the authorized personnel feed the relevant data into the system, several reports could be generated as per the security.

1.2 BACKGROUND

1.2.1 Brief Note on Existing System

The purpose of the "Cake Ordering System (COS)" is to have a system that handles the day-to-day sales operations in a cake shop and measures them very thoroughly. Users can spend less time ordering it over a call or physically going to shop for ordering. The shop Admin can receive multiple order simultaneously.

1.3 OBJECTIVES

The main aim for developing this project where user can purchase and order cakes online. The system is very useful for user can easily buy the cake from home through internet. The system reduces lot of work load for user as well as owner. By this system online cake shopping the cake is directly delivered to user address. The system functionality of products and orders is stored on server side in a web service. This project provides a lot of features to manage the products in very well manner. This project contains a lot of advance modules which makes the back-end system very powerful.

1.4 SCOPE

The "Art of Cakes" cake project is an MVC (Model-View-Controller) web-based application which includes user interaction. Our project is going to be a web portal. It is going to provide communication environment for users (user and admin). Every user has their own profiles and they can have access with given password to the system. The system will bring many advantages. For instance, the user and owner spend less money on traffic. And everyone benefits from these advantages.

1.5 STRUCTURE

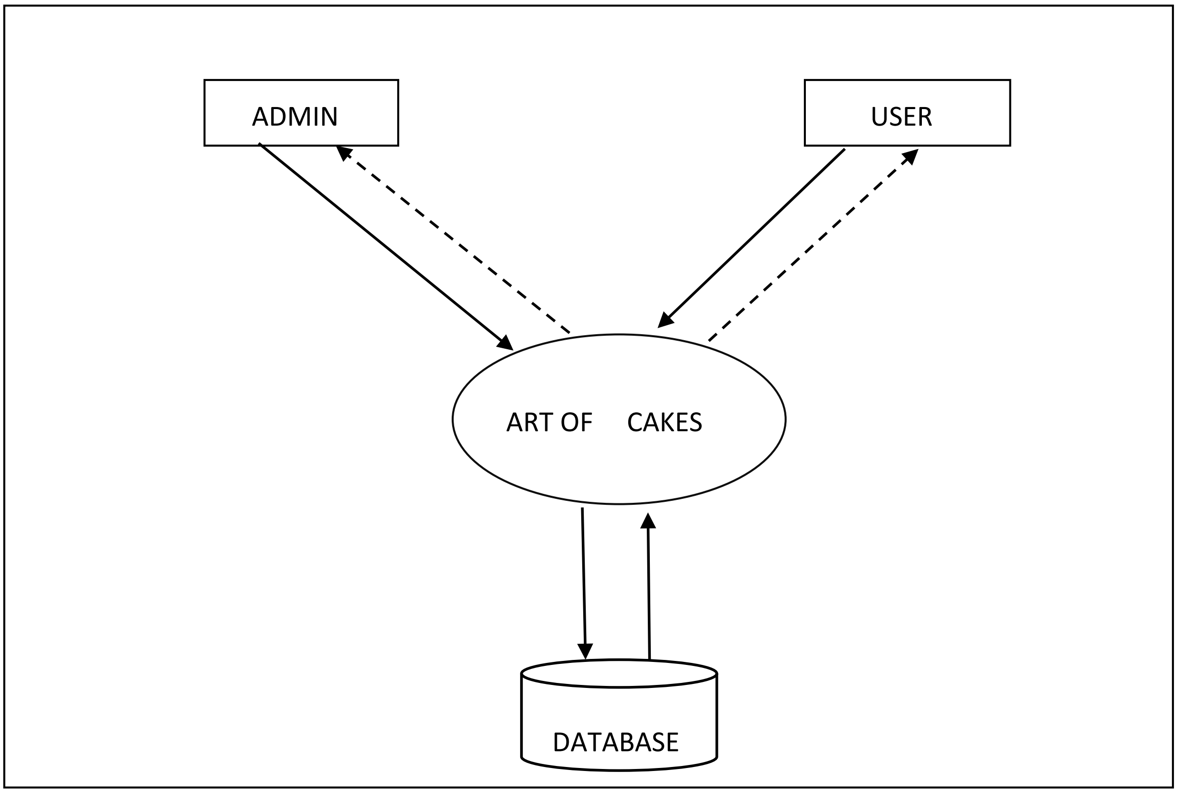
- Registration
- Login
- Home Page

- ✦ View Cakes
- ✦ Category
- ✦ Add to cart
- ✦ Place Order
- ✦ View Booking
- ✦ Profile

➢ Logout

1.6 SYSTEM ARCHITECTURE

Architecture design is the process of decomposing a large complex system into small subsystem. These subsystems are meant for providing some related services. The architecture design is basically a layout or a framework of the system for the subsystem control or communication.



1.7 END USER

Admin Role: Manipulates all the records. Provides authorization to different types of users.

An admin can add configurations to the system. Admin can add product or remove product from the products list.

User Role: User can order the cakes through online. Before ordering the cake, they can add
the items into and place the order. If they don't like they can also cancel, their order. User can send their reviews to the administrator.

1.8 SOFTWARE/HARDWARE USED FOR THE DEVELOPMENT

❖ Front-End Software Requirements for The Project *PHP:*

It is a general-purpose programming language originally designed for web development. PHP code may be executed with a command line interface (CLI), embedded into HTML code or used in combination with various web templates systems, web content management system and web frameworks.

JavaScript:

JavaScript is a programming language for the web. Its syntax is based on Java and C languages. Used in both the *front-end* and *back-end* of many platforms, JavaScript has become a standard. For every animated or interactive object, you see online, chances are JavaScript is involved.

CSS:

It is a simply designed language intended to simplify the process of making webpage presentable. CSS allows you to apply styles to webpage. More importantly CSS enables you to do this independent of the HTML that makes up each webpage. It provides powerful control over the presentation of an HTML document.

HTML:

It is used to webpage using Markup language. It is a combination of Hyper Text and Markup language. Hypertext defines link between webpages. Markup language is used to define the text document within tag which defines the structure of webpage.

- ✞ **Server Side:** PHP

- ✞ **Back-End Software Requirements for The Project**

  MYSQL:

  MYSQL use for both small and large applications. MYSQL is a database system used for developing web-based software applications.

  MYSQL is a relational database management system (RDBMS). MYSQL is a fast, reliable, and easy to use. MYSQL supports standard SQL (Structured Query Language).

- ✞ **Web Server:** XAMPP

- ✞ **OS:** Windows 7 or Above

1.9 SOFTWARE/HARDWARE REQUIREMENTS

Hardware Requirements

- Processor: 133-MHz Intel Pentium-class processor or higher
- RAM: 4GB RAM or above.
- Storage: 80 GB of Hard Drive

SOFTWARE REQUIREMENTS SPECIFICATIONS

2.1 INTRODUCTION

The introduction of the Software Requirement Specification (SRS) provides an overview of the entire SRS with purpose, scope, definitions, acronyms, abbreviations and overview of the SRS. The aim of this document is to gather and analyse and give an in-depth insight of the complete online cake shopping service system by defining the problem statement in detail.

The detailed requirements of “ART OF CAKES” system for customers are provided in this document.

2.1.1 Purpose

The purpose of this document is to provide the information about the software requirements of the "ART OF CAKES" system, which can be referred by the developers and will be basis for the validation of the final system. The objective of this system is to help the customers book online and get the cakes delivered easily.

2.1.2 Scope

The scope of the document is to provide sufficient detail on the functional and nonfunctional requirements of the item in order to see that all the functionalities are properly implemented. With the help of this application people can go through the application and avail various services like booking a cake, viewing cake details, availing offers, giving reviews and so on which will help people in getting their work done easily and in less time.

2.1.4 Overview

This system provides an easy solution for customers to buy the items without going to the shop and also for shop owner to sale the items. This proposed system can be used by any user and it does not require any educational level, experience or technical expertise in computer field but it will be of good use if user has the good knowledge of how to operate a computer.

2.2 OVERALL DESCRIPTION

2.2.1 Product Perspective

The proposed application contains easy graphical interfaces to all types of users. It contains simple databases which eliminates the complications of users. The product is a new, selfcontained product. The product provides authenticated access to the data which makes the system secure. It also provides

- Good and easy user interfaces which makes the system user friendly.
- Active workspaces for the users with many functions.

2.2.2 Product Functions

- The web application has a user-friendly interface and is easy to use.
- It can be accessed by admin, and customers at any time in any platform.

- Keeps the track on the details filled by the users.
- Reduces manual work and time consumption.
- Provides data security.
- Data loss and misuse of data is avoided.

2.2.3 User Characteristics

1. **User:** These are the users who have to access to item information enabling them to browse through the website but to make orders they must register themselves onto the site.
2. **Registered user**: They have all the advantage of the guest and in addition they have an account and can create a profile for free of cost. They can make payments to buy items. They can provide feedback and avail offers.
3. **Admin:** Admin is the core user. He/ She has complete access to the entire site. He/ She manages the site and the members. Admin keeps track of the orders. He/ She is responsible for updating the menu. Admin is responsible for adding the items and sorting them into the various categories.

2.2.4 General Constraints

- The system must provide a capacity for parallel operation and system design should not introduce scalability issues with regard to the number of surface computers, tablets or displays connected at any one time.
- Response time for loading the item should be analysed.
- The main constraint here would be the checking the genuineness of the user, which is not always possible. There can be security risks involved.
- The developed system should run under any platform (Unix, Linux, Mac, Windows etc.) that contains a web browser which supports PHP.

2.2.5 Assumptions

- The details related to the user, payment and notification has to be provided.
- Administrator is created in the system already.
- Roles and tasks are predefined.

2.3 SPECIFIC REQUIREMENTS

2.3.1 External Interface Requirements

The interface should be simple and easy to understand and use. It should also be an interactive interface. The system should prompt for the user and administrator to login to the application and for proper input criteria.

2.3.2 User Interfaces

The user interface for the software shall be compatible to any browser such as Internet Explorer, Mozilla or Google Chrome by which user can access to the system.

2.3.3 Hardware Interfaces

- Processor: 133-MHz Intel Pentium-class processor or higher
- RAM: 4GB and above
- Hard disk Utilization: 80GB and above
- Input Devices: Mouse, Keyboard
- Output Devices: Monitor, Printer

2.3.4 Software Interfaces

- Language: PHP 5.4
- Editor: Brackets
- Web Components: HTML
- Database (Backend): MYSQL
- Operating System: Windows XP or above
- Browser: Chrome, Mozilla Firefox or any other browsing application

2.3.5 Communication Interfaces

The system shall use the HTTP protocol for communication over the internet and for the intranet communication will be through TCP/IP protocol suite.

2.4 FUNCTIONAL REQUIREMENTS

- **Admin Module:**

  Admin should have complete knowledge of website. Admin has the following control over the websites. This module is designed with an intention to provide security to the admin. In this module the customer can order cake items through online. Before ordering cake, the customer should add items to the cart. Items added to the cart will be updated. It also helps customer to view the details of the cake. customized cake order. The customer can communicate with staff regarding customized order in the discussion panel. After the approval of quotation and customized design the customer can order for the cakes.

- LOGIN
- MANAGE PRODUCT
- VIEW ORDER DETAILS:
- VIEW CUSTOMER DETAILS:
- MANAGE CATEGORY
- MANAGE FEEDBACK

- **User Module:**

In this module the customer can order cake items through online. Before ordering cake, the customer should add items to the cart. Items added to the cart will be saved, so that customer can choose other cakes as well. Customer can order cart items by entering checkout button. Even the system will ask delivery address or pick up date and time before making payment. After the payment, order details will be shown and confirmed. This module helps customer to send customized cake order. The customer can communicate with staffs regarding customized order in the discussion panel. After the approval of quotation and customized design the customer can order for the cakes.

- **REGISTRATION**

If customer wants to buy the Items, then he/she must be registered, unregistered user can't go to the order cart.

- **LOGIN**

Customer logins to the system by entering valid user id and password for ordering a cake.

- **VIEW PRODUCT**

Customer can view cake items before login.

- **CHANGES TO CART**

Changes to cart means the customer after login or registration can make order or cancel order of the product from the ordered cart.

- **PLACE ORDER**

  Customer can place an order after adding to the cart.

- **PAYMENT**

  In this system we are dealing the mode of payment by the cash, credit card, debit card etc.

- **ORDER DETAILS**

  Customer can view their order details after placing order.

- **LOGOUT**

  After ordering or surfing for the product customer has to logout.

2.5 PERFORMANCE REQUIREMENTS

Performance is measured in terms of the output provided by the application. Only when the requirement specifications are properly given, it is possible to design a system, which will fit into required environment. It is very difficult to change the system once it has been designed and on the other hand designing a system, which does not cater to the requirements of the user, is of no use.

The requirement specification for any system can be broadly stated as below:

- The system should be able to interface with the existing system.
- The system should be accurate.
- You will be signed in within 25sec.
- Search results shown within 15 sec.
- Web support 200 customers logged at the same time.

2.6 DESIGN CONSTRAINTS

The system must be designed to allow web usability. That is, the system must be designed in such a way that will be easy to use and visible on most of the browsers.

12

2.7 SYSTEM ATTRIBUTES

The quality of the website is maintained in such a way so that it can be very user friendly to all the users of the website.

- **Reliability:** Good validation of user inputs will be done to avoid entering incorrect username and password.
- **Maintainability:** During maintenance stage, the SRS can be referred for the validation.
- **Portability:** This system can be run in any operating system and browser.
- **Compatibility:** This system will be compatible with almost all the web servers.
- **Flexibility:** The system keeps on updating the data according to the changes that takes place.

2.8 OTHER REQUIREMENTS

2.8.1 Safety Requirements:

- In case the user forgets or loses password, the repair functionality helps by choosing

"Forgot Password" option in the main login window.

- Consistency: Checking the fact that all clients must be attached to one server, so there is an appropriate control of the information.
- Authorization: Checking for the entity and provide features for them.

2.8.2 Security Requirements:

The proposed website is secured. It means the administrator is able to modify and delete the data. All others only have the rights to retrieve the information from the website that is concerned to them. When an error is detected an error message will be returned.

SYSTEM DESIGN

3.1 INTRODUCTION

System design is the architecture built on the basis of the system analysis. The purpose of the design phase is to plan a solution for the problem specified by the requirements. The system design describes the data to be input, calculated and stored. It is built on the required input and output design of the files. It aims to identify the module that should be present in the system the specification of this module and how they interact with each other to produce the desired result. The design activity is often divided into separate phase system design is a sometimes also called top level design. At the end of the system design all the major data structure file format and the major modules in the system and their specification are decided.

3.2 ASSUMPTIONS AND CONSTRAINTS

Project assumptions and constraints are identified at the beginning of the project. Throughout the project life cycle, they will be refined and re-analysed. Project assumptions and constraints are key to many processes in the PMBOK Guide. An essential aspect of your project, assumptions, and constraints are not managed like requirements or risks. However, documenting them helps to protect you from future aggravation. You should outline your project's assumptions and constraints in the project scope statement.

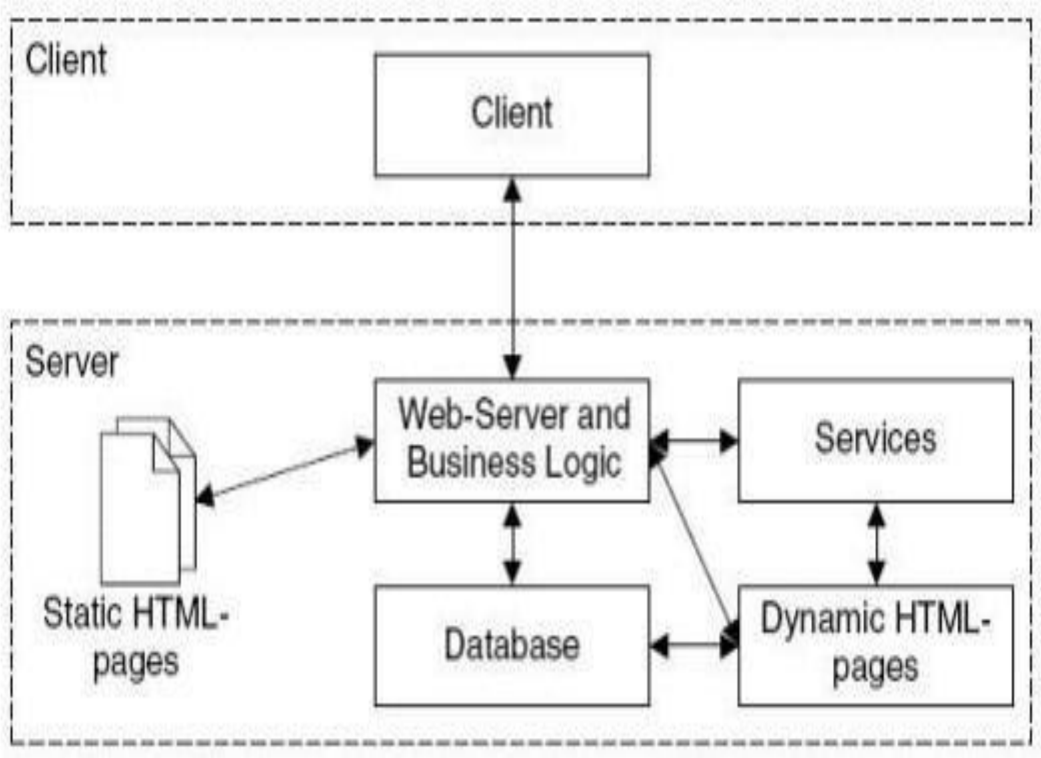
Project constraints are limitations imposed on the project, like the budget, schedule, or resources. The recognizes six project constraints: scope, quality, schedule, budget, resources, and risk. Out of these six, scope, schedule, and budget are known as the triple constraints

3.3 FUNCTIONAL DECOMPOSITION

Functional decomposition is a method of breaking down a problem into smaller and smaller functions. Each function is subdivided until it is manageable. Functional decomposition refers broadly to the process of resolving a functional relationship into its constituent parts in such a way that the original can be reconstructed from those parts by function components.

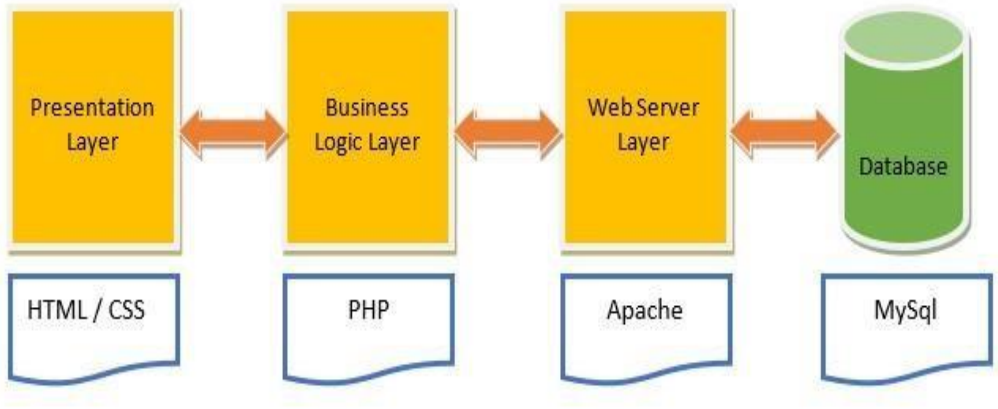
3.3.1 System Software Architecture

Software architecture refers to the fundamental structures of a software system and the discipline of creating such structures and systems. Each structure comprises software elements, relations among them, and properties of both elements and relations.


Client
Client
Server
Web-Server and Business Logic
Services
Static HTML-pages
Database
Dynamic HTML-pages

3.3.2 System Technical Architecture

Architecture (TA) is a form of IT architecture that is used to design computer systems. It involves the development of a technical blueprint with regard to the arrangement, interaction, and interdependence of all elements so that system-relevant requirements are met.



The system is made up of four layers:

- The **presentation layer:** includes user interfaces built using the HTML5 and CSS tools. The interface is designed using GUI components which are user-friendly.

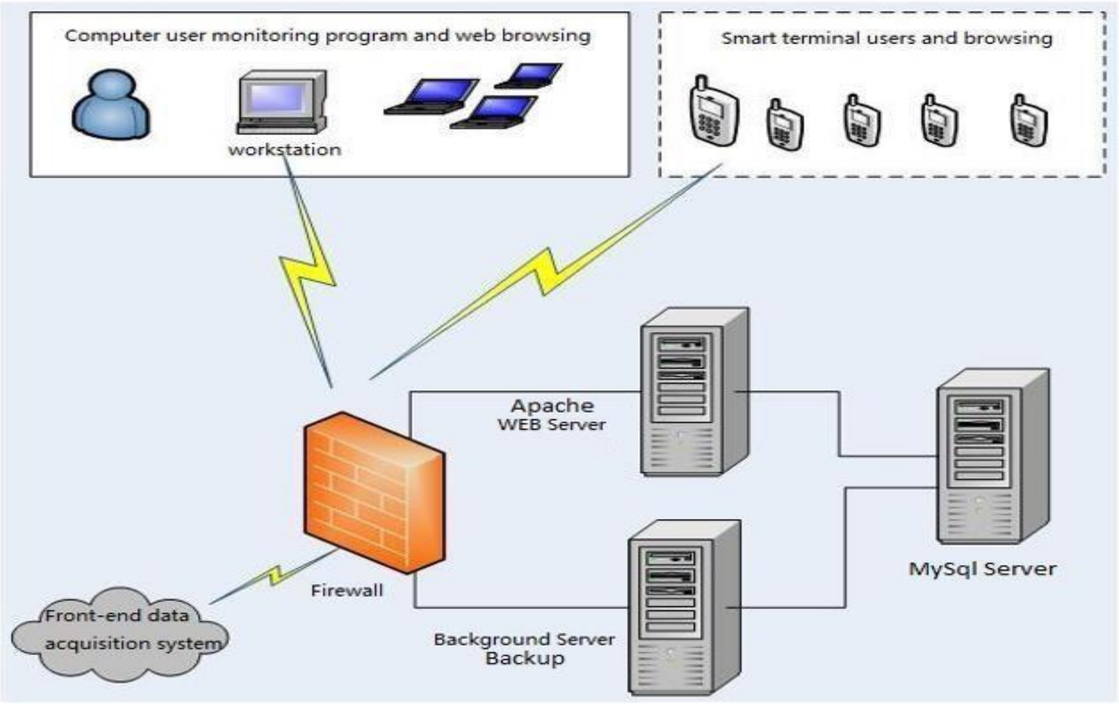
❾ **Business Logic layer:** is used for the retrieval of data obtained from user interfaces or acquired form databases. Activities for calculating, managing and making decisions and communicating with the front-end application and back-end database are managed by business logic written using PHP language are appearing here.

❾ **Web Server layer:** is to serve websites on the internet. It acts as a middleman between the server and client machines. It pulls content from the server on each user request and delivers it to the web. Apache Tomcat Web Server is used as a Web Server in the system.

❾ **Database layer:** is used to store data forwarded by the business layer and retrieve data on demand to the business layer. MySQL database software is used as a backend database of the system.

3.3.3 System Hardware Architecture

Hardware architecture is the representation of an engineered (or to be engineered) electronic or electromechanical hardware system, and the process and discipline for effectively implementing the designs for such a system.



Hardware Requirement	
Processor	**133-MHz Intel Pentium-class processor or higher**
Installed RAM	**8GB**
Display	**Standard Output Display**
Keyboard	**Standard Keyboard**
Mouse	**Standard Keyboard**

3.3.4 External Interfaces

Name of the Application: ART OF CAKES

Details of Interface: Admin Interface, User Interface

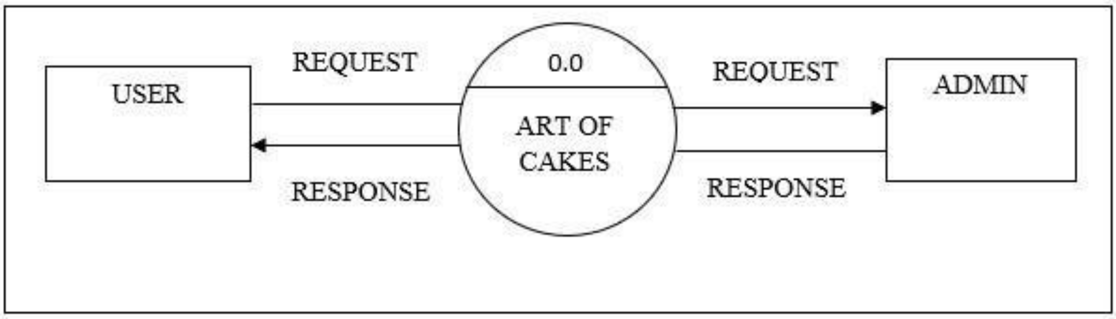
Type of Interface: Graphical User Interface, Menu driven Interface, Form based Interface

3.4 DESCRIPTION OF PROGRAMS

3.4.1 Context Flow Diagram (CFD)

Context Flow Diagram is a top level (also known as level 0) data flow datagram. It contains only one process node that generalizes the functions of the entire system in relationship to external entities. In context diagram the entire system is treated as single process and all its inputs, outputs, sinks, and sources are identified.

The above di


USER
REQUEST
RESPONSE
0.0
ART OF CAKES
REQUEST
RESPONSE
ADMIN

Admin manages all the entities of the Project.

3.4.2 Data Flow Diagrams

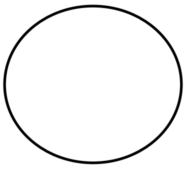
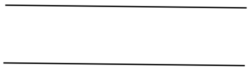
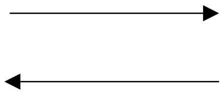
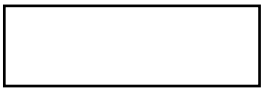
A Data Flow Diagram is a graphical representation of a system or a portion of the system. It consists of data flows, process, sources and sink and stores all the description through the use of easily understandable symbols.

- DFD is one of the most important modelling tools. It is used to model the system, components that interact with the system, uses the data and information flows in the system.
- DFD shows the information moves through the and how it is modified by a series of transformations. It is a graphical technique that depicts information moves from input or output.
- DFD is also known as bubble chart or Data Flow Graphs. DFD's may partition into a level that represents increasing information flows and functional details.

Rules Regarding DFD Construction:

- A process cannot have only outputs.
- A process cannot have only inputs.
- The inputs to a process must be sufficient to produce the outputs from the process.
- All data stores must be connected to at least one process.
- All data stores must be connected to a source or sink.
- A data flow can have only one direction of flow. Multiple data flows to and/or from the same process and data store must be shown by separate arrows.
- If the exact same data flows to two separate arrows, it should be represented by a forked arrow.

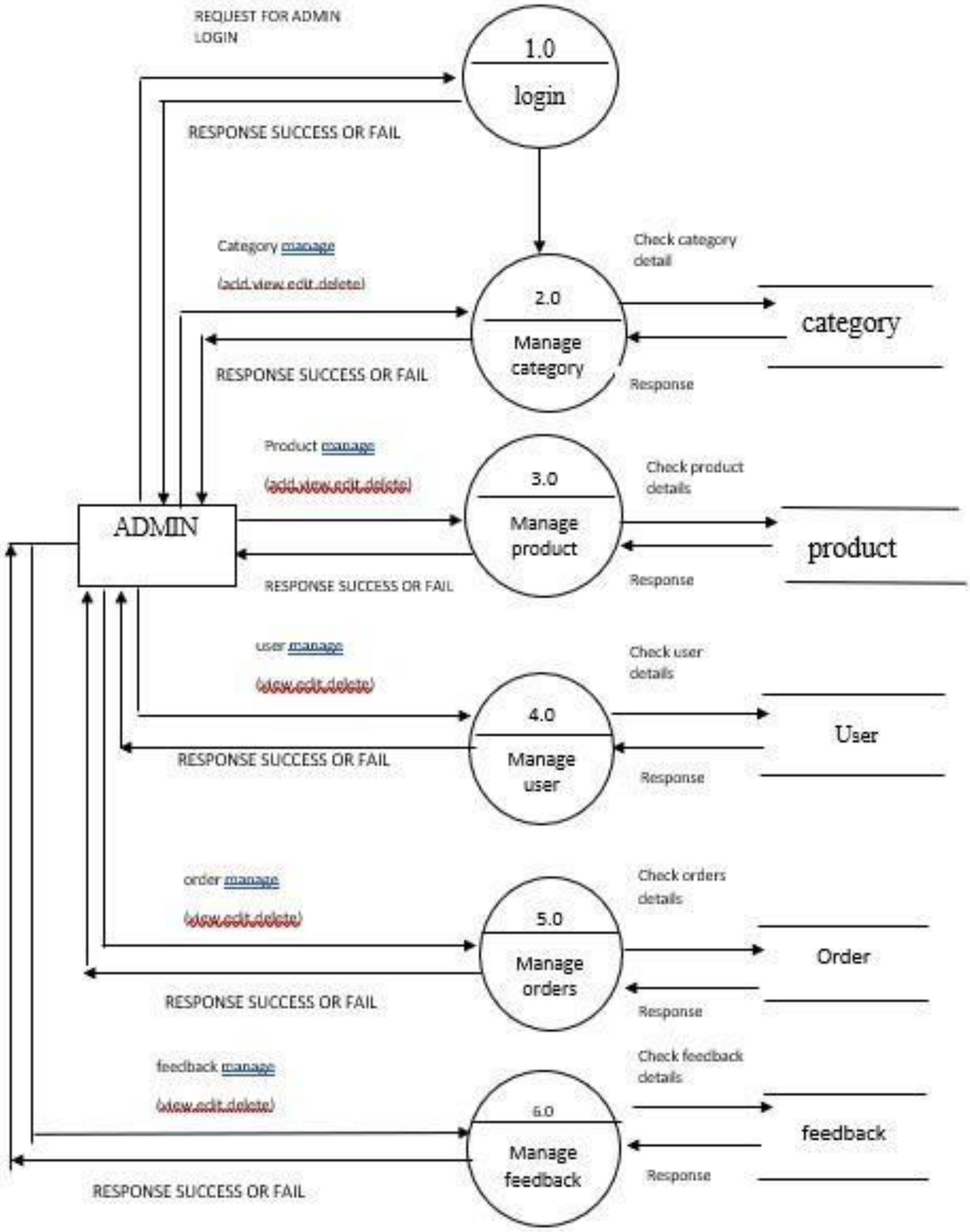
Name	Notation	Description

Process		A process transforms incoming data flow into outgoing data flow. The processes are shown by named circles.
Data store		Data stores are repositories of data in the system. They are sometimes also referred to as files.
Data flows		Data flows are pipelines through which packets of information flow. Label the arrows with the name of the data that moves through it.
External Entity		External entities are objects outside the system with which the system communicates. External Entities are sources and destinations of the system's inputs and outputs.

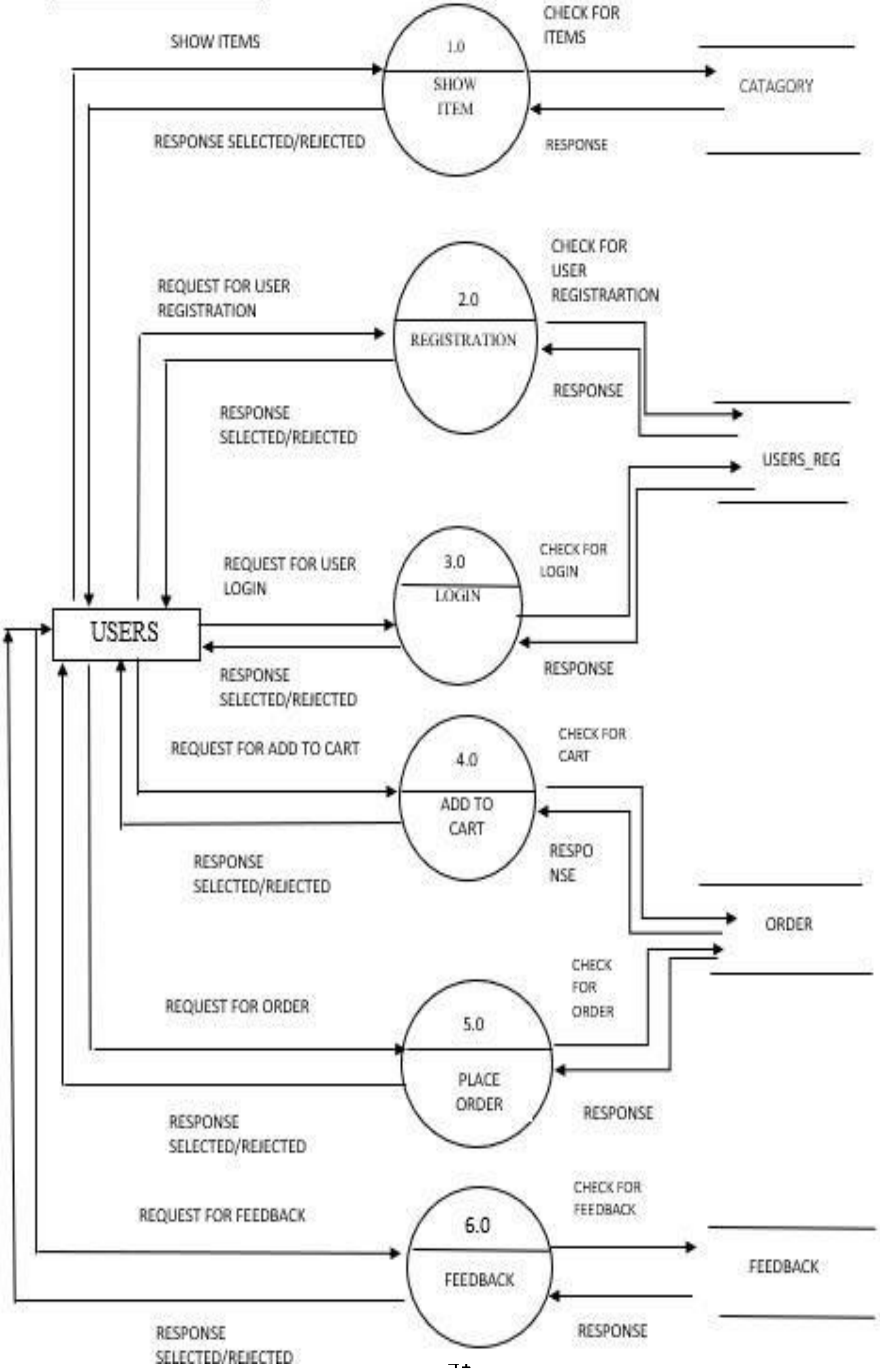
- Data can’t flow directly back into the process it has just left. All data flows must be named using a noun phrase.

A level 1 data flow diagram (DFD) is more detailed than a level 0 but not as detailed as a level 2 DFD. It breaks down the main processes into sub processes that can then be analysed and improved on a more intimate level. In 1-level DFD, context diagram is decomposed into multiple processes (system management, examination, evaluation and IA). In this level we highlight the main functions of the system and breakdown the high-level process of 0 level DFD into sub processes.

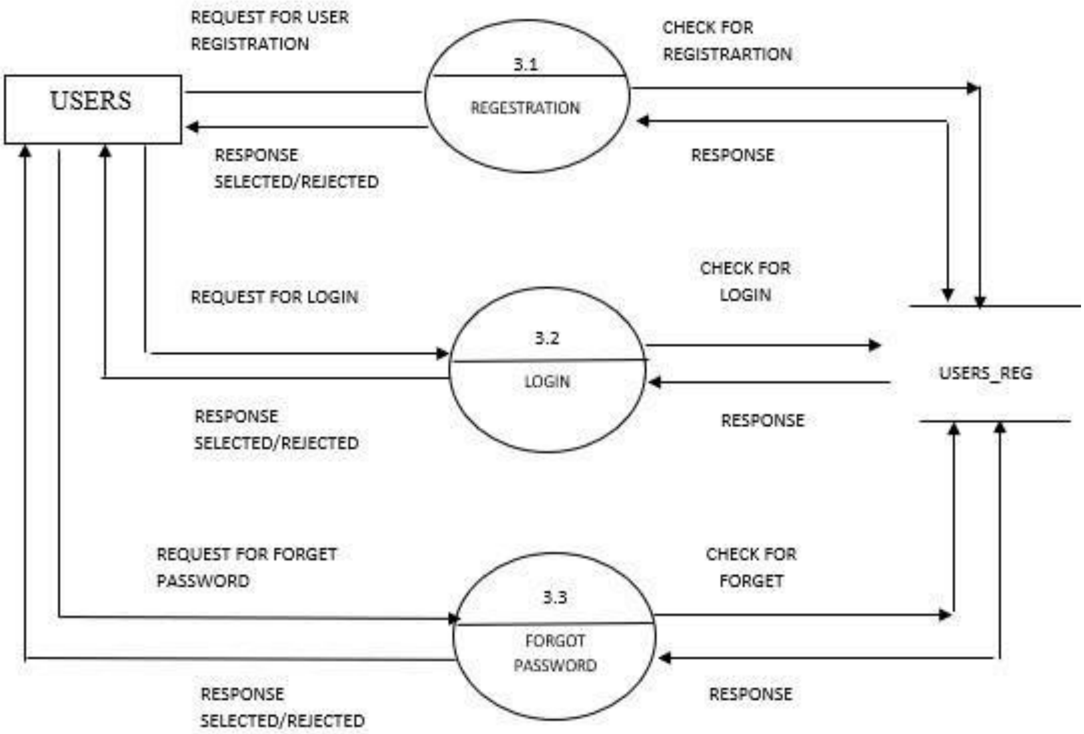
ADMIN SIDE DFD LEVEL



1ST LEVEL DFD for USER



2ND LEVEL DFD FOR USER



3.5 DESCRIPTION OF COMPONENTS

The following the entire component used in our application how the process for the input and output is specified as below.

ADMIN SIDE:

3.5.1 Functional Component 1: Registration

This component is to register of the admin

Input: admin_id, admin_username, admin_email, admin_password

Process: The system checks the validity of the admin. If the details provided by the admin are valid, the admin is registered and the details are stored in the admin table of the database.

Output: The admin in the dashboard is displayed on successful registration.

3.5.2 Functional Component 2: Login

Input: admin_email, admin_username, admin_password

The input for login component is admin id and password.

Process: The system checks for the correctness of the username and password.

Output: Admin will be taken to the home page on successful login.

3.5.3 Functional Component 3: Manage category

Input: category_id, category_name, category_image

Process: The system will manage to add, view, edit and delete the details of category.

Output: Admin successfully added a new category and details, viewed the category and its details, successfully edited the details of category, successfully deleted the category and its details.

3.5.4 Functional Component 4: Manage products

Input: product_id, product_name, product_category, product_description, product_price, product_image

Process: The system will manage to add, view, edit and delete the details of products.

Output: Admin successfully added a new product and details, viewed the product and its details, successfully edited the details of product, successfully deleted the product and its details.

3.5.5 Functional Component 5: Manage users

Input: user_id, users_name, user_email, user_password, user_address, user_mobile

Process: The system will manage to edit and delete the users and their details.

Output: Admin successfully edited the details of the users or admin successfully deleted the user and their details.

3.5.6 Functional Component 6: Manage orders

Input: order_id, user_id, delivery_date, payment_method, total_amount

Process: The system will manage to edit and delete the orders.

Output: Admin successfully edited the details of the order or admin successfully deleted the order.

3.5.7 Functional Component 7: Manage feedback

Input: id,

email, name, phone, feedback, suggestions **Process**: the system will collect the feedback from the user, by entering their details, if they are valid user the feedback will stored in a database.

Output: Admin can view the feedback.

USER SIDE:

3.5.8 Functional Component 8: Registration

Input: user_id, users_name, user_email, user_password, user_address, user_mobile

Process: The system checks the validity of the user. If the details provided by the user are valid, the user is registered and the details are stored in the user's registration table of the database.

Output: The user main page is displayed on successful registration.

3.5.9 Functional Component 9: Login

Input: user_name, user_password.

Process: The system checks for the correctness of the username and password.

Output: Entered Username and password will be checked for validity. If it is valid user will be directed to Homepage**.**

3.5.10 Functional Component 10: forgot password

Input: user_email, user_mobile.

Process: The system checks for the correctness of the user email and mobile number.

Output: Entered email and mobile number will be checked for validity. If it is valid user will be directed to setting new password page.

3.5.11 Functional Component 11: category

Input: category_name, category_image.

Process: The system lists out the image of the categories with its name.

Output: Clicking the any of the category will direct to the product page of that particular category and display the products with its details.

3.5.12 Functional Component 12: Add to cart

Input: product_name, product_price, product_image, product_description

Process: The system lists out the products names with image and details which have been selected by the user.

Output: Clicking to add to cart will direct to the cart page where the whole description of product take place.

3.5.13 Functional Component 13: Place order

Input: S. No, product_name, product_price, quantity, total_amount, delivery_date, payment _method

Process: The system checks for the entered details are valid, delivery date should be selected and choosing the any one of the payment methods.

Output: user successfully placed the order.

3.5.14 Functional Component 14: Feedback

Input: id, email, name, phone, feedback, suggestions

Process: the system will collect the feedback from the user, by entering their details, if they are valid user then the feedback will be stored in a database.

Output: user successfully given the feedback.

DETAILED DESIGN

5.1 INTRODUCTION

Detailed design is the second level of the design process. During detailed design, we specify how the module in the system interacts with each other and the internal logic of each of the modules specified during system design is decided, hence it is also called as logic design.

Detailed design essentially expands the system design and database design to contain a more detailed description of the processing logic and data structures so that the design is sufficiently complete for coding. Detailed design is the phase where the design is refined and plans, specifications and estimates are created. Detailed design will include outputs such as 2D and 3D models, P & ID's, cost build up estimates, procurement plans etc. This phase is where the full cost of the project is identified.

5.2 STRUCTURE OF THE SOFTWARE PACKAGE (structure chart)

The various functional components used are

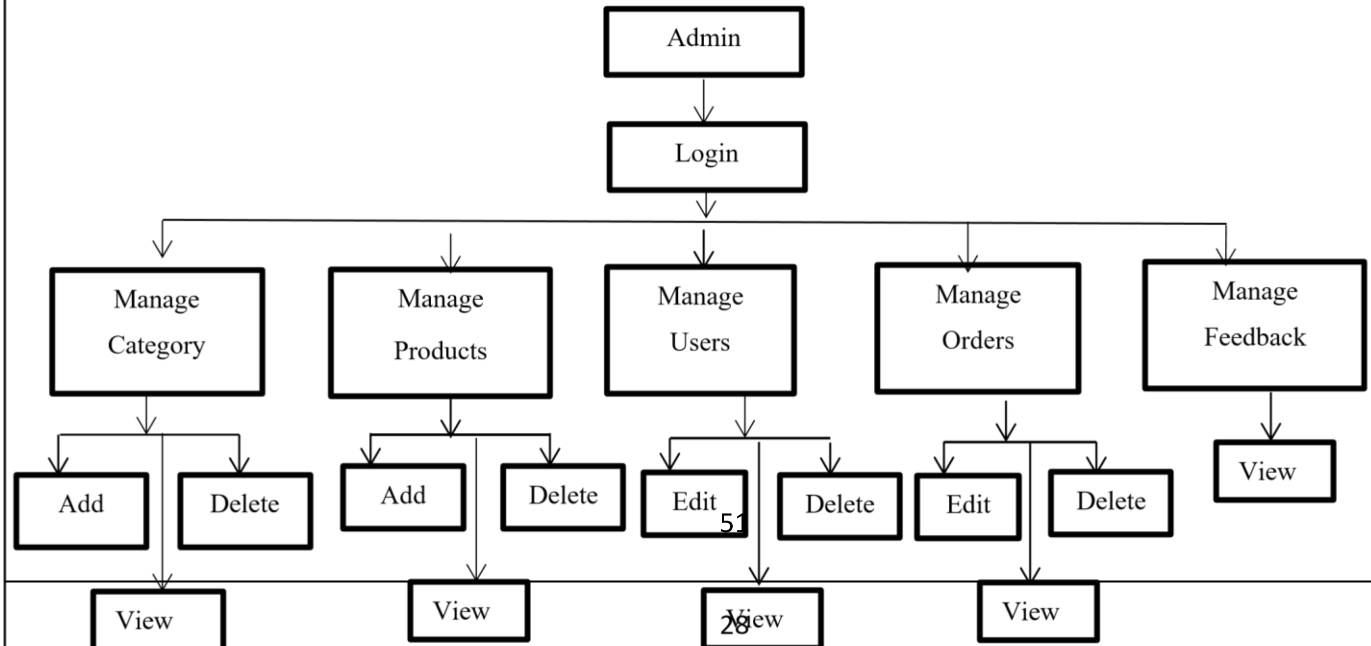
- Admin Module
- User Module

5.3 MODULAR DECOMPOSITION COMPONENTS

5.3.1 Admin Module:

The module is designed with an intention to allow the admin to use any of these options easily, so that he/she can enter and make changes to those details.

Structure chart:



LOGIN

Input: Entering the username and password.

Process: The system checks for the correctness of the username and password.

File I/O interface: Admin Table

Output: Entered Username and password will be checked for validity. If it is valid admin will be directed to Homepage.

MANAGE CATEGORY:

Input: Admin manages category by viewing information's such as category and its details

Process: The system will manage to add, view, edit and delete the details of category.

File I/O interface: Admin has the interfaces to view, add, edit, delete the category.

Output: Admin successfully added a new category and details, viewed the category and its details, successfully edited the details of category, successfully deleted the category and its details.

MANAGE PRODUCTS:

Input: Admin manages products by viewing information's such as product and its details

Process: The system will manage to add, view, edit and delete the details of products.

File I/O interface: Admin has the interfaces to view, add, edit, delete the Products.

Output: Admin successfully added a new product and details, viewed the product and its details, successfully edited the details of product, successfully deleted the product and its details.

MANAGE USERS:

Input: Admin manages users by viewing information's such as user id and its details.

Procedural details: The system will manage to view, edit and delete the details of Users.

File I/O interface: Admin has the interfaces to view, edit, delete the Users.

Output: Admin successfully viewed the user and its details, successfully edited the details of user, successfully deleted the user and its details.

MANAGE ORDERS:

Input: Admin manages orders by viewing information's such as order id and its details.

Procedural details : The system will manage to view, edit and delete the details of Orders.

File I/O interface: Admin has the interfaces to view, edit, delete the Orders.

Output: Admin successfully viewed the order and its details, successfully edited the details of order, successfully deleted the order and its details.

Manage feedback:

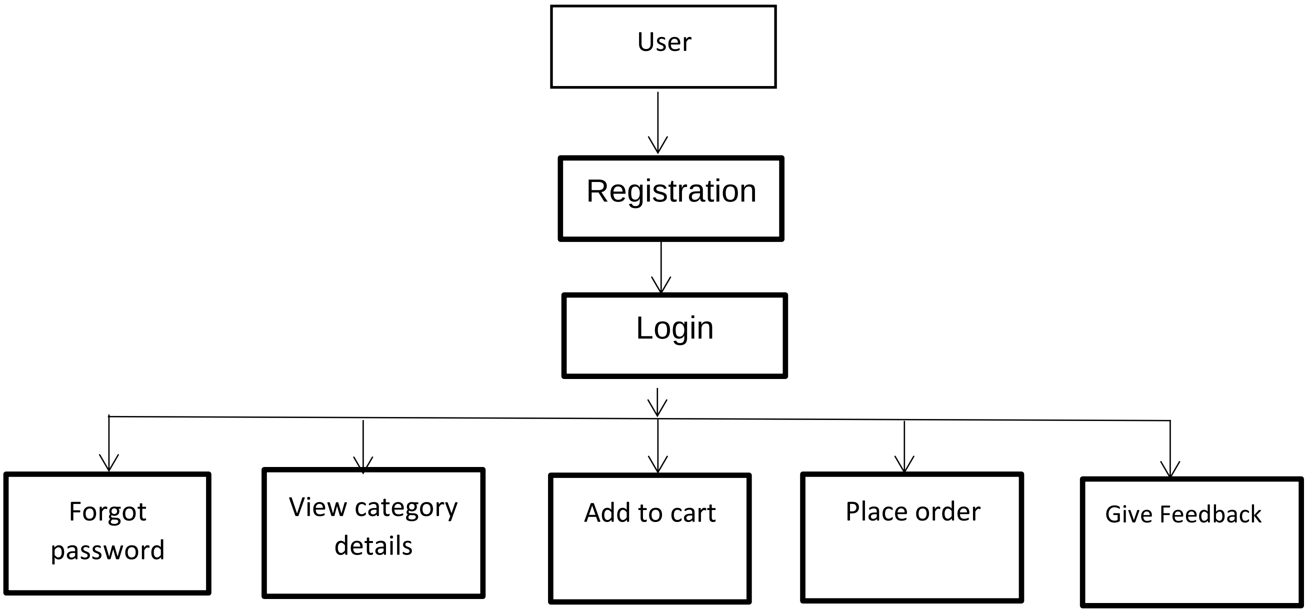
Input: Admin manages feedback by viewing information's such as username and its details.

Procedural details: The system will manage to view the details of Feedback.

File I/O interface: Admin has the interfaces to view the Feedback.

Output: Admin successfully viewed the Feedback and its details.

5.3.2 User module

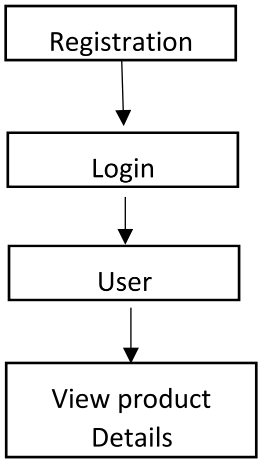


5.3.2 USER

Registration:

Input: New user must register to continue shopping by giving information such as username, number, email id, address and setting password.

Structure Chart:



File I/O Interfaces: User has interfaces to register.

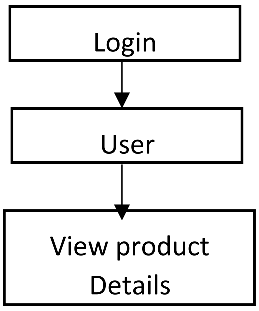
Output: Confirmation message will be displayed.

Function: Successful registration

Login:

Input: Registered customer can login to their account by providing username and password.

Structure Chart:



File I/O Interfaces: User has interfaces to Login.

Output: Confirmation message will be displayed and shopping proceeded.

Function: Registered users can order items.

Forgot password

Input: User can reset the password by providing email and phone number.

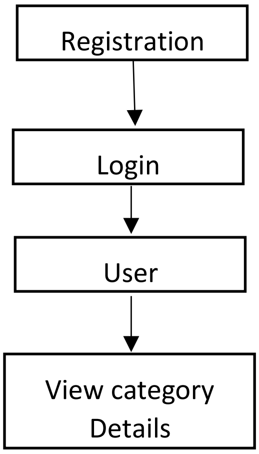
File I/O interface: User has interfaces to Forgot password.

Output: Confirmation message will be displayed and can login with new password.

View category details

Input: User can view category details.

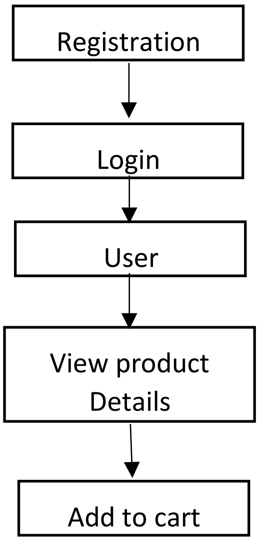
Structure Chart:



Add to cart

Input: User can add products to cart.

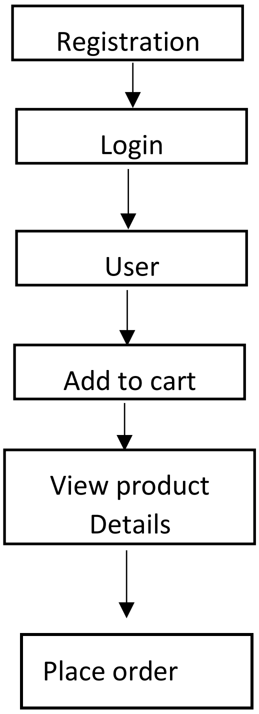
Structure Chart:



place order

Input: After adding to cart user can place order.

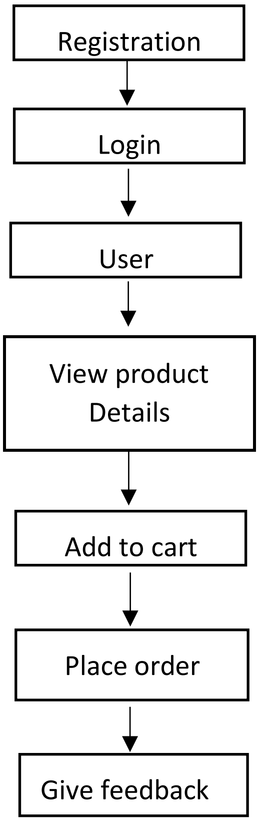
Structure Chart:



View feedback

Input: User can give feedback.

Structure Chart:



PROGRAM CODE LISTING

DATABASE CONNECTION

Config.php

```
<?php
$host = "localhost";
$config_username = "root";
$password = "";
$db = "artofcakes";
$conn = mysqli_connect($host, $config_username, $password, $db);
?>
```

ADMIN

Login.php

```
<div class="splash-container">
<div class="card ">
<div class="card-header text-center"><a href="#"><h2 class="text-primary">Art of
Cakes</h2></a><span class="splash-description">Please enter your user
information.</span></div>
<div class="card-body">
<form id="form" data-parsley-validate="" method="post" action="login_check.php"> <div
class="form-group">
<input class="form-control form-control-lg" type="text" name="admin_username"
dataparsley-trigger="change" required="" placeholder="Username" autocomplete="off">
</div>
<div class="form-group">
<input class="form-control form-control-lg" id="pass1" type="password" required=""
placeholder="Password" name="admin_password">
</div>
<button type="submit" class="btn btn-primary btn-lg btn-block">Sign in</button> </form>
</div>
<div class="card-footer bg-white p-0  ">
<div class="card-footer-item card-footer-item-bordered">
<a href="admin_signup.php" class="footer-link">Create An Account</a></div>
<div class="card-footer-item card-footer-item-bordered">
<a href="forgot.php" class="footer-link">Forgot Password</a>
</div>
</div>
</div>
</div> <?php
require_once('../config.php');
$admin_username = $_POST['admin_username'];
$admin_password = $_POST['admin_password'];
$select = "SELECT * FROM cake_shop_admin_registrations where admin_username =
'$admin_username' AND admin_password = '$admin_password'";
$query = mysqli_query($conn, $select);
$res = mysqli_fetch_assoc($query); if
(mysqli_num_rows($query) > 0) {
session_start();
$_SESSION['user_admin_id'] = $res['admin_id'];
$_SESSION['user_admin_username'] = $res['admin_username']; header("Location:
dashboard.php?login_success=1");
} else
{
header("Location: index.php?login_error=1");
}
?>
```

Dashboard.php

```
<body>
   <div class="dashboard-main-wrapper">
   <?php
include('includes/navbar.php');
   include('includes/sidebar.php');
   ?>
      <div class="dashboard-wrapper">
         <div class="container-fluid dashboard-content">
            <div class="row">
               <div class="col-xl-12 col-lg-12 col-md-12 col-sm-12 col-12">
                  <div class="page-header">
                     <h2 class="pageheader-title">Dashboard</h2>
                     <p class="pageheader-text">Proin placerat ante duiullam scelerisque a velit
ac porta, fusce sit amet vestibulum mi. Morbi lobortis pulvinar quam.</p>
                     <div class="page-breadcrumb">
                        <nav aria-label="breadcrumb">
                           <ol class="breadcrumb">
                              <li class="breadcrumb-item"><a href="dashboard.php"
class="breadcrumb-link">Dashboard</a></li>
                              <li class="breadcrumb-item active"
ariacurrent="page">Overview</li>
                           </ol>
                        </nav>
                     </div>
                  </div>
               </div>
            </div>
            <div class="row">
               <?php
               require_once('../config.php');
               ?>
               <div class="col-xl-3 col-lg-6 col-md-6 col-sm-12 col-12">
                  <div class="card bg-primary">
                     <div class="card-body">
                        <h3 class="card-title text-white">Users</h3>
                        <?php
                        $select_users = "SELECT * FROM cake_shop_users_registrations";
                        $query_users = mysqli_query($conn, $select_users);
                        $res_users = mysqli_num_rows($query_users);
                        ?>
                        <p class="card-text">Number of users in database = <?php echo
$res_users;?>.</p>
```

```
                <a href="view_users.php" class="btn btn-rounded btn-dark">View</a>
              </div>
            </div>
          </div>
          <div class="col-xl-3 col-lg-6 col-md-6 col-sm-12 col-12">
            <div class="card bg-brand">
              <div class="card-body">
                <h3 class="card-title">Category</h3>
                <?php
                $select_category = "SELECT * FROM cake_shop_category";
                $query_category = mysqli_query($conn, $select_category);
                $res_category = mysqli_num_rows($query_category);
                ?>
                <p class="card-text">Number of categories in database = <?php echo
$res_category;?>.</p>
                <a href="view_category.php" class="btn btn-rounded
btnlight">View</a>
              </div>
            </div>
          </div>
          <div class="col-xl-3 col-lg-6 col-md-6 col-sm-12 col-12">
            <div class="card bg-secondary">
              <div class="card-body">
                <h3 class="card-title text-white">Products</h3>
                <?php
               $select_product = "SELECT * FROM cake_shop_product";
                $query_product = mysqli_query($conn, $select_product);
                $res_product = mysqli_num_rows($query_product);
                ?>
                <p class="card-text">Number of products in database = <?php echo
$res_product;?>.</p>
                <a href="view_product.php" class="btn btn-rounded btn-info">View</a>
              </div>
            </div>
          </div>
          <div class="col-xl-3 col-lg-6 col-md-6 col-sm-12 col-12">
            <div class="card bg-success">
              <div class="card-body">
                <h3 class="card-title">Orders</h3>
                <?php
                $select_orders = "SELECT * FROM cake_shop_orders";
                $query_orders = mysqli_query($conn, $select_orders);
                $res_orders = mysqli_num_rows($query_orders);
                ?>
                <p class="card-text">Number of orders in database = <?php echo
$res_orders;?>.</p>
                <a href="view_orders.php" class="btn btn-rounded
btndanger">View</a>
              </div>
            </div>
          </div>
          <div class="col-xl-3 col-lg-6 col-md-6 col-sm-12 col-12">
            <div class="card bg-dark">
              <div class="card-body">
                <h3 class="card-title text-white">Feedback</h3>
<?php
                $select_feedback = "SELECT * FROM poll";
                $query_feedback = mysqli_query($conn, $select_feedback);
                $res_feedback = mysqli_num_rows($query_feedback);
                ?>
                <p class="card-text text-white">Number of feedbacks in database =
<?php echo $res_feedback;?>.</p>
                <a href="feedback.php" class="btn btn-rounded btn-light">View</a>
              </div>
            </div>
          </div>
        </div>
      </div>
     <?php
     include('includes/footer.php');
      ?>
    </div>
  </div>
  <script src="../js/jquery-3.3.1.min.js"></script>
<script src="../js/bootstrap.bundle.js"></script>
  <script src="../js/jquery.slimscroll.js"></script>
  <script src="../js/main-js.js"></script>
</body>
```

View_users.php

```
      <div class="row">
          <div class="col-xl-12 col-lg-12 col-md-12 col-sm-12 col-12">
            <div class="card">
              <h5 class="card-header">Users Table</h5>
              <div class="card-body">
                <div class="table-responsive">
                  <table class="table table-striped table-bordered first">
                    <thead>
```

```
                                            <tr>
                                                <th>S. No.</th>
                                                <th>Username</th>
                                                <th>Email</th>
                                                <th>Password</th>
                                                <th>Mobile</th>
                                                <th>Address</th>
                                                <th>Action</th>
                                            </tr>
                                        </thead>
                                        <tbody>
<?php
                                            require_once('../config.php');
                                            $select = "SELECT * FROM cake_shop_users_registrations";
                                            $query = mysqli_query($conn, $select);
                                            $i = 1;
                                            while ($res = mysqli_fetch_assoc($query)) {
                                            ?>
                                            <tr>
                                                <td><?php echo $i++;?></td>
                                                <td><?php echo $res['users_username'];?></td>
                                               <td><?php echo $res['users_email'];?></td>
                                               <td><?php echo $res['users_password'];?></td>
                                               <td><?php echo $res['users_mobile'];?></td>
                                                <td><?php echo $res['users_address'];?></td>
                        <td>
                                                    <button data-toggle="modal" data-target="#exampleModal"
class="btn btn-space btn-primary" onclick="edit_users(<?php echo
$res['users_id'];?>)">Edit</button>
                                                    <button onclick="delete_users(<?php echo
$res['users_id'];?>)" class="btn btn-space btn-secondary">DELETE</button>
                                                </td>
                                            </tr>
                                            <?php } ?>
                                        </tbody>
                                    </table>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
```

<u>Edit_users.php</u>

```
<?php
```

```
require_once('../config.php');
$users_username = $_POST['users_username'];
$users_email = $_POST['users_email'];
$users_password = $_POST['users_password'];
$users_mobile = $_POST['users_mobile'];
$users_address = $_POST['users_address'];
$hidden_id = $_POST['hidden_users'];
$update = "UPDATE cake_shop_users_registrations SET users_username =
'$users_username', users_email = '$users_email', users_password = '$users_password',
users_mobile = '$users_mobile', users_address = '$users_address' where users_id =
$hidden_id";
mysqli_query($conn, $update);
header('Location: view_users.php?edit_msg=1'); ?>
```

Delete_users.php

```
<?php
$users_id = $_GET['users_id']; require_once('../config.php');
$delete = "DELETE FROM cake_shop_users_registrations where users_id = $users_id";
mysqli_query($conn, $delete); header("Location: view_users.php");
?>
```

Add_category.php

```
        <div class="row">
              <div class="col-xl-12 col-lg-12 col-md-12 col-sm-12 col-12">
                 <div class="card">
                       <h5 class="card-header">Add category</h5>
                       <div class="card-body">
                          <form action="insert_category.php" id="form" method="post"
enctype="multipart/form-data">
                             <div class="form-group">
                                <label for="inputCategoryName">Category Name</label>
                                <input id="inputCategoryName" type="text"
name="category_name" required="" placeholder="Enter category name" autocomplete="off"
class="form-control">
                             </div>
                             <div class="custom-file mb-3">
                                <input type="file" class="custom-file-input" id="customFile"
name="category_image">
                                <label class="custom-file-label" for="customFile">Choose
Image</label>
                             </div>
                             <div class="row">
                                <div class="col-sm-6 pb-2 pb-sm-4 pb-lg-0 pr-0">
                                </div>
```

```
                                   <div class="col-sm-6 pl-0">
                                      <p class="text-right">
                                         <button type="submit" class="btn btn-space
btnprimary">Add</button>
                                         <button type="reset" class="btn btn-space
btnsecondary">Clear</button>
                                      </p>
                                   </div>
                                </div>
                             </form>
                          </div>
                       </div>
                 </div>
              </div>
```

View_category.php

```
         <div class="row">
               <div class="col-xl-12 col-lg-12 col-md-12 col-sm-12 col-12">
                  <div class="card">
                     <h5 class="card-header">Category Table</h5>
                     <div class="card-body">
                        <div class="table-responsive">
                           <table class="table table-striped table-bordered first">
<thead>
                                   <tr>
                                      <th>S. No.</th>
                                      <th>Name</th>
                                      <th>Image</th>
                                      <th>Action</th>
                                   </tr>
                                </thead>
                                <tbody>
<?php
                                   require_once('../config.php');
                                   $select = "SELECT * FROM cake_shop_category";
                                   $query = mysqli_query($conn, $select);
                                   $i = 1;
                                   while ($res = mysqli_fetch_assoc($query)) {
                                   ?>
                                   <tr>
                                      <td><?php echo $i++;?></td>
                                      <td><?php echo $res['category_name'];?></td>
                                      <td><img src="../upload/<?php echo
$res['category_image'];?>" height="100px" width="100px"></td>
```

```
                                          <td>
                                             <button data-toggle="modal" data-target="#exampleModal"
class="btn btn-space btn-primary" onclick="edit_cat(<?php echo
$res['category_id'];?>)">Edit</button>
                                             <button onclick="delete_cat(<?php echo
$res['category_id'];?>)" class="btn btn-space btn-secondary">DELETE</button>
                                          </td>
                                       </tr>
                                       <?php } ?>
                                    </tbody>
                                 </table>
                              </div>
                           </div>
                        </div>
                     </div>
                  </div>
```

Edit_category.php

```
<?php require_once('../config.php');
$category_name =
$_POST['category_name'];
$file_name = $_FILES['category_image']['name'];
$hidden_id = $_POST['hidden_category']; if
($file_name != "") {
   $f_name = Date('ymdhis');
   $file_array = explode('.', $file_name);
   $ext = $file_array[count($file_array) - 1];
   $new_file_name = $f_name.'.'.$ext;
   $destination = "../uploads/".$new_file_name;
$source = $_FILES['category_image']['tmp_name'];
move_uploaded_file($source, $destination);
   $update = "UPDATE cake_shop_category SET category_name = '$category_name', category_image
= '$new_file_name' where category_id = $hidden_id";
mysqli_query($conn, $update);     header('Location:
view_category.php?edit_msg=1');
}
elseif ($file_name == "") {
   $update = "UPDATE cake_shop_category SET category_name = '$category_name' where
category_id = $hidden_id";     mysqli_query($conn, $update);     header('Location:
view_category.php?edit_msg=1');
}
?>
```

Delete_category.php

```
<?php
$cat_id = $_GET['cat_id']; require_once('../config.php');
$delete = "DELETE FROM cake_shop_category where category_id = $cat_id";
mysqli_query($conn, $delete); header("Location: view_category.php");
?>
```

Add_product.php

```
<div class="row">
          <div class="col-xl-12 col-lg-12 col-md-12 col-sm-12 col-12">
            <div class="card">
                <h5 class="card-header">Add Product</h5>
                <div class="card-body">
                  <form action="insert_product.php" id="form" method="post"
enctype="multipart/form-data">
                    <div class="form-group">
                      <label for="inputProductName">Product Name</label>
                      <input id="inputProductName" type="text"
name="product_name" required="" placeholder="Enter product name" autocomplete="off"
class="form-control">
                    </div>
                    <div class="form-group">
                      <label   for="inputProductCategory">Categories</label>
<select           class="form-control"           id="inputProductCategory"
name="product_category">
                        <?php
                        require_once('../config.php');
                        $select = "SELECT * FROM cake_shop_category";
                        $query = mysqli_query($conn, $select);
                        while ($res = mysqli_fetch_assoc($query)) {
                        ?>
                        <option value="<?php echo $res['category_id'];?>"><?php
echo $res['category_name'];?></option>
                        <?php } ?>
                      </select>
                      <a href="add_category.php">Add category.</a>
                    </div>
                    <div class="form-group">
                      <label for="inputType">Select type</label>
                      <select class="form-control" id="inputType" required
name="product_type">
                        <!--<option selected>Choose category</option>-->
                        <option value="Veg">Veg</option>
                        <option value="Non-Veg">Non-Veg</option>
                      </select>
                    </div>

                    <div class="form-group">
                        <div class="mb-3">
                            <label for="inputStatus">Availability</label><br>
                            <label class="switch">
                                <input type="checkbox" name="product_status">
                                <span class="slider round"></span>
                            </label><br>
                            <small class="help-text">Green=Available, Red=Not
Available</small>
                        </div>
                    </div>

                    <div class="form-group">
                       <div class="mb-3">
                            <label for="inputStatus">Trending</label><br>
                            <label class="switch">
                                <input type="checkbox" name="product_trend">
                                <span class="slider round"></span>
                            </label><br>
                            <small class="help-text">Green=Trending,
Red=default</small>
                        </div>
                    </div>
                    <div class="form-group">
                        <label for="inputProductPrice">Price</label>
<input id="inputProductPrice" type="number" name="product_price" min="1"
required="" placeholder="Enter product price" autocomplete="off" class="form-
control currency-inputmask">
                    </div>
                    <div class="custom-file mb-3">
                        <input type="file" class="custom-file-input" id="customFile"
name="product_image[]" multiple="">
                        <label class="custom-file-label" for="customFile">Choose
Image</label>
                    </div>
                    <div class="form-group">
                        <label for="inputProductDescription">Description</label>
                        <textarea id="inputProductDescription"
name="product_description" required="" placeholder="Product description"
class="formcontrol"></textarea>
                    </div>
                    <div class="row">
```

```
                      <div class="col-sm-6 pb-2 pb-sm-4 pb-lg-0 pr-0">
                      </div>
                      <div class="col-sm-6 pl-0">
                        <p class="text-right">
                          <button type="submit" class="btn btn-space
btnprimary">Add</button>
                          <button type="reset" class="btn btn-space
btnsecondary">Clear</button>
                        </p>
                      </div>
                    </div>
                  </form>
                </div>
              </div>
          </div>
        </div>
```

View_product.php

```
<div class="row">
          <div class="col-xl-12 col-lg-12 col-md-12 col-sm-12 col-12">
            <div class="card">
              <h5 class="card-header">Product Table</h5>
              <div class="card-body">
                <div class="table-responsive">
                  <table class="table table-striped table-bordered first">
                    <thead>
                      <tr>
                        <th>S. No.</th>
                        <th>Name</th>
                        <th>Category</th>
                        <th>Price</th>
                        <th>Type</th>
                        <th>Status</th>
                        <th>Trending</th>
                        <th>Image</th>
                        <th>Description</th>
                        <th>Action</th>
                      </tr>
                    </thead>
                    <tbody>
<?php
require_once('../config.php');
                      $select = "SELECT cake_shop_product.*,
cake_shop_category.category_name FROM cake_shop_product JOIN cake_shop_category ON
cake_shop_product.product_category = cake_shop_category.category_id";
```

```
                      $query = mysqli_query($conn, $select);
                      $i = 1;
                      while ($res = mysqli_fetch_assoc($query)) {
                      ?>
                      <tr>
                        <td><?php echo $i++;?></td>
                        <td><?php echo $res['product_name'];?></td>
                        <td><?php echo $res['category_name'];?></td>
                        <td>Rs. <?php echo $res['product_price'];?></td>
                        <td><?php echo $res['product_type'];?></td>
                        <td>
                          <?php
                          if($res['product_status'] == '0')
                          {
                            echo '<span class="bg-success badge badge-pill">Available</span>';
                          }else
                          {
                            echo '<span class="bg-danger badge badgepill">Unavailable</span>';
                          }
                          ?>
                        </td>
                        <td>
                          <?php
if($res['product_trend'] == '1')
                          {
                            echo '<span class="bg-success badge badge-pill
textwhite">Yes</span>';
                          }else
                          {
                            echo '<span class="bg-dark badge badge-pill text-white">No</span>';
                          }
                          ?>
                        </td>
                        <td>
                          <?php
                          $file_array = explode(', ', $res['product_image']);
                          ?>
                          <div class="owl-carousel owl-theme" style="width: 100px;">
                          <?php                               for
($j=0; $j < count($file_array); $j++) {
                          ?>
                          <div class="item">
                          <img src="../upload/<?php echo $file_array[$j];?>" height="100px"
width="100px">
                          </div>
                          <?php
```

```
                                        }
                                        ?>
                                        </div>
                                    </td>
                                    <td><?php echo $res['product_description'];?></td>
                                    <td>
                                        <button data-toggle="modal" data-target="#exampleModal" class="btn
btn-space btn-primary" onclick="edit_prod(<?php echo $res['product_id'];?>)">Edit</button>
                                        <button onclick="delete_prod(<?php echo $res['product_id'];?>)"
class="btn btn-space btn-secondary">DELETE</button>
                                    </td>
                                </tr>
                                <?php } ?>
                            </tbody>
                        </table>
                    </div>
                </div>
            </div>
        </div>
</div>
```

Edit_product.php

```
<?php
require_once('../config.php');
$product_name = $_POST['product_name'];
$product_category = $_POST['product_category'];
$product_type = $_POST['product_type'];
$product_price = $_POST['product_price'];
$product_status = isset($_POST['product_status'])?'1':'0';
$product_trend = isset($_POST['product_trend'])?'0':'1';
$product_description = $_POST['product_description'];
$hidden_id = $_POST['hidden_product']; if
($_FILES['product_image']['name'][0] != "") {
for ($i=0; $i < count($_FILES['product_image']['name']); $i++) {
    $file_name = $_FILES['product_image']['name'][$i];
    $f_name = Date('ymdhis').$i;
    $file_array = explode('.', $file_name);
    $ext = $file_array[count($file_array) - 1];
    $new_file_name = $f_name.'.'.$ext;
    $source = $_FILES['product_image']['tmp_name'][$i];
$destination = "../upload/".$new_file_name;
move_uploaded_file($source, $destination);
    if ($i == count($_FILES['product_image']['name']) - 1) {
        $upload_file_name .= $f_name.'.'.$ext;
    } else {
        $upload_file_name .= $f_name.'.'.$ext.", ";
    }
}
$update = "UPDATE cake_shop_product set product_name = '$product_name',
product_category = '$product_category', product_type = '$product_type', product_price =
'$product_price', product_status = '$product_status', product_trend = '$product_trend',
product_description = '$product_description', product_image = '$upload_file_name'
where product_id = $hidden_id"; mysqli_query($conn, $update);
header("Location: view_product.php?edit_msg=2");
}
elseif ($_FILES['product_image']['name'][0] == "") {
    $update = "UPDATE cake_shop_product set product_name = '$product_name',
product_category = '$product_category', product_type = '$product_type', product_price =
'$product_price', product_status = '$product_status', product_trend = '$product_trend',
product_description = '$product_description' where product_id = $hidden_id";
mysqli_query($conn, $update);
    header("Location: view_product.php?edit_msg=2");
}
?>
```

Delete_product.php

```
<?php
$prod_id = $_GET['prod_id']; require_once('../config.php');
$delete = "DELETE FROM cake_shop_product where product_id = $prod_id";
mysqli_query($conn, $delete); header("Location: view_product.php"); ?>
```

View_orders.php

```
<div class="row">
                    <div class="col-xl-12 col-lg-12 col-md-12 col-sm-12 col-12">
                        <div class="card">
                            <h5 class="card-header">Orders Table</h5>
                            <div class="card-body">
                                <div class="table-responsive">
                                    <table class="table table-striped table-bordered first">
                                        <thead>
                                            <tr>
                                                <th>S. No.</th>
                                                <th>Orders id</th>
                                                <th>Users id</th>
                                                <th>Delivery date</th>
                                                <th>Payment method</th>
                                                <th>Total amount</th>
                                                <th>Action</th>
                                            </tr>
                                        </thead>
                    <tbody>
<?php
                      require_once('../config.php');
                      $select = "SELECT * FROM cake_shop_orders";
                      $query = mysqli_query($conn, $select);
                      $i = 1;
                      while ($res = mysqli_fetch_assoc($query)) {
                      ?>
                      <tr>
                        <td><?php echo $i++;?></td>
                        <td><?php echo $res['orders_id'];?></td>
                        <td><?php echo $res['users_id'];?></td>
                        <td><?php echo $res['delivery_date'];?></td>
                        <td><?php echo $res['payment_method'];?></td>
                        <td>Rs. <?php echo $res['total_amount'];?></td>
                        <td>
                          <button data-toggle="modal" data-target="#exampleModal"
class="btn btn-space btn-primary" onclick="edit_orders(<?php echo
$res['orders_id'];?>)">Edit</button>

                          <button onclick="confirm_orders(<?php echo
$res['orders_id'];?>)" class="btn btn-space btn-success">CONFIRM</button>

                          <button onclick="delete_orders(<?php echo
$res['orders_id'];?>)" class="btn btn-space btn-secondary">DELETE</button>
                        </td>
                      </tr>
                      <?php } ?>
                    </tbody>
                  </table>
                </div>
              </div>
            </div>
          </div>
        </div>
        <div class="row">
          <div class="col-xl-12 col-lg-12 col-md-12 col-sm-12 col-12">
            <div class="card">
              <h5 class="card-header">Orders Detail Table</h5>
              <div class="card-body">
                <div class="table-responsive">
                  <table class="table table-striped table-bordered first">
<thead>
                      <tr>
                       <th>S. No.</th>
```

```
                                    <th>Orders id</th>
                                    <th>Product name</th>
                                    <th>Quantity</th>
                                    <th>Action</th>
                                </tr>
                            </thead>
                            <tbody>
  <?php
                                require_once('../config.php');
                                $select = "SELECT * FROM cake_shop_orders_detail";
                                $query = mysqli_query($conn, $select);
                                $i = 1;
                                while ($res = mysqli_fetch_assoc($query)) {
                                ?>
                                <tr>
                                    <td><?php echo $i++;?></td>
                                    <td><?php echo $res['orders_id'];?></td>
                                    <td><?php echo $res['product_name'];?></td>
                                    <td><?php echo $res['quantity'];?></td>
                                    <td>
                                        <button data-toggle="modal" datatarget="#exampleModal1"
class="btn btn-space btn-primary" onclick="edit_orders_detail(<?php echo
$res['orders_detail_id'];?>)">Edit</button>
                                        <button onclick="delete_orders_detail(<?php echo
$res['orders_detail_id'];?>)" class="btn btn-space btn-secondary">DELETE</button>
                                    </td>
                                </tr>
                                <?php } ?>
                            </tbody>
                        </table>
                    </div>
                </div>
            </div>
        </div>
    </div>
```

Edit_orders.php

```
<?php
require_once('../config.php'); $users_id
= $_POST['users_id'];
$delivery_date = $_POST['delivery_date'];
$payment_method = $_POST['payment_method'];
$total_amount = $_POST['total_amount'];
$hidden_id = $_POST['hidden_orders'];
```

```
$update = "UPDATE cake_shop_orders SET users_id = '$users_id', delivery_date =
'$delivery_date', payment_method = '$payment_method', total_amount = '$total_amount'
where orders_id = $hidden_id";
mysqli_query($conn, $update);
header('Location: view_orders.php?edit_msg=1');
?>
```

Edit_orders_deetails.php

```
<?php
require_once('../config.php'); $orders_id
= $_POST['orders_id'];
$product_name = $_POST['product_name'];
$quantity = $_POST['quantity'];
$hidden_id = $_POST['hidden_orders_detail'];
$update = "UPDATE cake_shop_orders_detail SET orders_id = '$orders_id', product_name =
'$product_name', quantity = '$quantity' where orders_detail_id = $hidden_id";
mysqli_query($conn, $update);
header('Location: view_orders.php?edit_msg=2');
?>
```

Delete_orders.php

```
<?php
$orders_id = $_GET['orders_id'];
require_once('../config.php');

use PHPMailer\PHPMailer\PHPMailer; use
PHPMailer\PHPMailer\SMTP;
use PHPMailer\PHPMailer\Exception;

require 'PHPMailer\src\PHPMailer.php'; require
'PHPMailer\src\SMTP.php';
require 'PHPMailer\src\Exception.php';

$delete = "DELETE FROM cake_shop_orders where orders_id = $orders_id";
mysqli_query($conn, $delete);
$mail = new PHPMailer(true);
    try
{
        $mail->isSMTP();
        $mail->Host       = 'smtp.gmail.com';
        $mail->SMTPAuth   = true;
       $mail->Username   = 'vigneshdevadiga012@gmail.com';
        $mail->Password   = 'hyybzdebagryxmhj';
        $mail->SMTPSecure = PHPMailer::ENCRYPTION_SMTPS;
        $mail->Port       = 465;
        //Recipients
        $mail->setFrom('artofcakes@gmail.com', 'Admin');
        $mail->addAddress('vigneshdevadiga012@gmail.com');     //Add a recipient

        //Content
        $mail->isHTML(true);                                  //Set email format to HTML
        $mail->Subject = 'Your Order has been Cancelled';
        $email_template = "
        <h2>Hello<h2>
        <h3>Your order id: $orders_id has been cancelled </h3>
<h4>We hope to see you again soon. Click here to <a
href='http://localhost/artofcakes/login_users.php'>Login</a> our page.</h4>
        ";

        $mail->Body    = $email_template;

        $mail->send();

        echo "<script>
        alert('e-mailed sucessfully sent!!!');
        window.location.assign('view_orders.php')</script>";
    } catch (Exception $e) {
        echo "Message could not be sent. Mailer Error: {$mail->ErrorInfo}";
    }
?>
```

Delete_orders_details.php

```
<?php
$orders_detail_id = $_GET['orders_detail_id']; require_once('../config.php');
$delete = "DELETE FROM cake_shop_orders_detail where orders_detail_id =
$orders_detail_id"; mysqli_query($conn,
$delete); header("Location:
view_orders.php");
?>
```

Confirm_orders.php

```
<?php
require_once("../config.php"); $orders_id
= $_GET['orders_id'];

use PHPMailer\PHPMailer\PHPMailer; use
PHPMailer\PHPMailer\SMTP;
use PHPMailer\PHPMailer\Exception;
```

```
require 'PHPMailer\src\PHPMailer.php'; require
'PHPMailer\src\SMTP.php';
require 'PHPMailer\src\Exception.php';

$check_user = "SELECT * FROM cake_shop_orders WHERE orders_id=$orders_id ";
$query = mysqli_query($conn, $check_user); if($query)
{
   $mail = new PHPMailer(true);
    try
{
      $mail->isSMTP();
      $mail->Host       = 'smtp.gmail.com';
      $mail->SMTPAuth   = true;
      $mail->Username   = 'vigneshdevadiga012@gmail.com';
      $mail->Password   = 'hyybzdebagryxmhj';
      $mail->SMTPSecure = PHPMailer::ENCRYPTION_SMTPS;
      $mail->Port       = 465;
      //Recipients
      $mail->setFrom('artofcakes@gmail.com', 'Admin');
      $mail->addAddress('vigneshdevadiga012@gmail.com');     //Add a recipient

      //Content
      $mail->isHTML(true);                                  //Set email format to HTML
      $mail->Subject = 'Your Order has been Confirmed';
      $email_template = "
      <h2>Hello<h2>
      <h3>You are receiving this email because we received an order request from your
account. Your order has been confirmed and it will be delivered on the specified date. <h3>
<br/><br/>
      <a href='http://localhost/artofcakes/feedback_form.php'>Click Here</a>
      ";

      $mail->Body    = $email_template;

      $mail->send();

      $delete = "DELETE FROM cake_shop_orders where orders_id = $orders_id";
    mysqli_query($conn, $delete);

      echo "<script>
      alert('e-mailed sucessfully sent!!!');
      window.location.assign('view_orders.php')</script>";
   } catch (Exception $e) {
      echo "Message could not be sent. Mailer Error: {$mail->ErrorInfo}";
    }
} else{     echo
"<script>
        alert('Something Went Wrong!!!');
        window.location.assign('view_orders.php')</script>"; }
?>
```

Logout.php

```
<?php
session_start();
unset($_SESSION['user_admin_id']); unset($_SESSION['user_admin_username']);
header("Location: index.php");
?>
```

USER

Register.php

```
<form id="form" class="splash-container" data-parsley-validate="" method="post"
action="insert_users.php">
        <div class="card">
            <div class="card-header">
                <h3 class="mb-1">Registrations Form</h3>
                <p>Please enter your user information.</p>
            </div>
            <div class="card-body">
                <div class="form-group">
                    <input class="form-control form-control-lg" type="text"
name="users_username" data-parsley-trigger="change" required="" placeholder="Username"
pattern="^[a-zA-Z]+$" autocomplete="off">
                </div>
                <div class="form-group">
                    <input class="form-control form-control-lg" type="email" name="users_email"
data-parsley-trigger="change" required="" placeholder="E-mail" autocomplete="off">
                </div>
                <div class="form-group">
                    <input class="form-control form-control-lg" id="pass1" type="password"
required="" placeholder="Password" name="users_password">
                </div>
                <div class="form-group">
                    <input class="form-control form-control-lg" data-parsley-equalto="#pass1"
type="password" required="" placeholder="Confirm password">
                </div>
                <div class="form-group">
                    <input class="form-control form-control-lg" type="tel" name="users_mobile"
data-parsley-trigger="change" required="" placeholder="Mobile no." pattern="[0-9]{10}"
autocomplete="off">
                </div>
                <div class="form-group">
                    <input class="form-control form-control-lg" type="text" name="users_address"
data-parsley-trigger="change" required="" placeholder="Address" autocomplete="off">
                </div>
                <div class="form-group pt-2">
                    <button class="btn btn-block btn-primary" type="submit">Register</button>
                </div>
            </div>
            <div class="card-footer bg-white">
                <p>Already member? <a href="login_users.php" class="text-secondary">Login
Here.</a></p>
            </div>
        </div>
</form>
```

Login.php

```
<div class="splash-container">
    <div class="card ">
        <div class="card-header text-center"><a href="#"><h2 class="text-primary">Art of
Cakes</h2></a><span class="splash-description">Please enter your user
information.</span></div>
        <div class="card-body">
            <form id="form" data-parsley-validate="" method="post"
action="login_check_users.php">
                <div class="form-group">
                    <input class="form-control form-control-lg" type="text"
name="users_username" data-parsley-trigger="change" required="" placeholder="Username"
autocomplete="off">
                </div>
                <div class="form-group">
                   <input class="form-control form-control-lg" id="pass1" type="password"
required="" placeholder="Password" name="users_password">
                </div>
                <button type="submit" class="btn btn-primary btn-lg btn-block">Sign
in</button>
            </form>
        </div>
        <div class="card-footer bg-white p-0  ">
            <div class="card-footer-item card-footer-item-bordered">
                <a href="register.php" class="footer-link">Create An Account</a></div>
            <div class="card-footer-item card-footer-item-bordered">
                <a href="forgotpass.php" class="footer-link">Forgot Password</a>
            </div>
        </div>
    </div>
</div>
```

Index.php

```
<div class="dashboard-main-wrapper">
         <div class="dashboard-header">
            <nav class="navbar navbar-expand-lg bg-white fixed-top">
                <div class="logo">
                <img src="upload/logo.png" width="100px" style="cursor: pointer;" alt="" >
</div>
                <a class="navbar-brand" href="#">Art of Cakes</a>
                <button class="navbar-toggler" type="button" data-toggle="collapse"
datatarget="#navbarSupportedContent" aria-controls="navbarSupportedContent"
ariaexpanded="false" aria-label="Toggle navigation">
                    <span><i class="fas fa-bars mx-3"></i></span>
</button>
                <div class="collapse navbar-collapse " id="navbarSupportedContent">
                    <ul class="navbar-nav ml-auto navbar-right-top">
                        <li class="nav-item">
                            <a class="nav-link active" href="index.php">Home</a>
                        </li>
                        <li class="nav-item dropdown">
                            <a class="nav-link" href="#" id="navbarDropdownMenuLink1"
datatoggle="dropdown" aria-haspopup="true" aria-expanded="false">Category</a>
                            <div class="dropdown-menu dropdown-menu-right"
arialabelledby="navbarDropdownMenuLink1">
                            <?php
                            require_once('config.php');
                            $select = "SELECT * FROM cake_shop_category";
                            $query = mysqli_query($conn, $select);
                            while ($res = mysqli_fetch_assoc($query)) {
                            ?>
                                <a class="dropdown-item" href="shop.php?category=<?php echo
$res['category_id'];?>">
                                    <?php echo $res['category_name'];?>
                                </a>
                            <?php
                            }
                            ?>
                            </div>
            </li>
            <li class="nav-item">
              <a class="nav-link" href="cart.php"><i class="fas fa-shopping-cart"></i>
<span class="badge badge-pill badge-secondary"><?php echo $printCount;?></span></a>
            </li>
            <li class="nav-item">
              <a class="nav-link" href="about.php">About us</a>
            </li>
            <li class="nav-item">
              <a class="nav-link" href="contact.php">Contact</a>
            </li>
            <li class="nav-item dropdown nav-user">
              <a class="nav-link nav-user-img" href="#"
id="navbarDropdownMenuLink2" data-toggle="dropdown" aria-haspopup="true"
ariaexpanded="false"><img src="upload/default-image.jpg" alt="" class="user-avatar-md
rounded-circle"></a>
              <div class="dropdown-menu dropdown-menu-right nav-user-dropdown"
aria-labelledby="navbarDropdownMenuLink2">
                <div class="nav-user-info">
                  <h5 class="mb-0 text-white nav-user-name"><?php echo
$printUsername;?></h5>
                  <span class="status"></span><span class="ml-2">Available</span>
                </div>
                <a class="dropdown-item" href="account_users.php"><i class="fas
fauser mr-2"></i>Account</a>
                <a class="dropdown-item" href="login_users.php"><i class="fas fa-
signin-alt mr-2"></i>Login</a>
                <a class="dropdown-item" href="logout_users.php"><i class="fas
fapower-off mr-2"></i>Logout</a>
              </div>
            </li>
          </ul>
        </div>
      </nav>
    </div>
<div class="container-fluid dashboard-content">
        <div class="row">
          <div class="col-xl-12 col-lg-12 col-md-12 col-sm-12 col-12">
<div id="carouselExampleControls" class="carousel slide"
dataride="carousel">
              <div class="carousel-inner rounded">
                <div class="carousel-item active">
                  <img class="d-block w-100" src="upload/cake1.jpg" alt="First slide">
                  <div class="carousel-caption d-md-block pb-5">
                    <h3 class="text-white">GOOD FOOD, GOOD LIFE</h3>
                    <p>Cakes are special, every celebration ends with something sweet
like a cake and people remember it's all about the memories.</p>
                    <a href="about.php" class="btn btn-rounded btn-outlinelight">Read
More</a>
                  </div>
                </div>
                <div class="carousel-item">
                  <img class="d-block w-100" src="upload/cake2.jpg" alt="Second
slide">
                  <div class="carousel-caption d-md-block pb-5">
                    <h3 class="text-white">The cake we bake with love</h3>
                    <p>Take the shredded pieces of your life and bake a master cake out
of it.</p>
                    <a href="about.php" class="btn btn-rounded btn-outlinelight">Read
More</a>
                  </div>
                </div>
                <div class="carousel-item">
                  <img class="d-block w-100" src="upload/cake3.jpg" alt="Third
slide">
                  <div class="carousel-caption d-md-block pb-5">
                    <h3 class="text-white">Love at first bite</h3>
                    <p>You can’t be down when you are holding a cupcake.</p>
                    <a href="about.php" class="btn btn-rounded btn-outlinelight">Read
More</a>
                  </div>
                </div>
                <div class="carousel-item">
                  <img class="d-block w-100" src="upload/cake4.jpg" alt="Fourth
slide">
                  <div class="carousel-caption d-md-block pb-5">
                    <h3 class="text-white">A party without cake is really just a
meeting.</h3>
                    <p>Cakes are special, every celebration ends with something sweet
like a cake and people remember it's all about the memories.</p>
                    <a href="about.php" class="btn btn-rounded btn-outlinelight">Read
More</a>
                  </div>
                </div>
                <div class="carousel-item">
                  <img class="d-block w-100" src="upload/cake5.jpg" alt="Fifth slide">
                  <div class="carousel-caption d-md-block pb-5">
                    <h3 class="text-white">Eat a cake.</h3>
                    <p>Cakes are special, every celebration ends with something sweet
like a cake and people remember it's all about the memories.</p>
```

```
                    <a href="about.php" class="btn btn-rounded btn-outlinelight">Read
More</a>
                  </div>
                </div>
                <div class="carousel-item">
                  <img class="d-block w-100" src="upload/cake6.jpg" alt="Sixth slide">
                  <div class="carousel-caption d-md-block pb-5">
                    <h3 class="text-white">View our categories.</h3>
                    <p>Cakes are special, every celebration ends with something sweet
like a cake and people remember it's all about the memories.</p>
                    <a href="about.php" class="btn btn-rounded btn-outlinelight">Read
More</a>
                  </div>
                </div>
</div>
              <a class="carousel-control-prev" href="#carouselExampleControls"
role="button" data-slide="prev">
                  <span class="carousel-control-prev-icon" aria-hidden="true"></span>
                     <span class="sr-only">Previous</span>   </a>
              <a class="carousel-control-next" href="#carouselExampleControls"
role="button" data-slide="next">
                    <span class="carousel-control-next-icon" ariahidden="true"></span>
                  <span class="sr-only">Next</span>  </a>
            </div>
          </div>
        </div>

        <div class="row m-5">
          <div class="col-xl-12 col-lg-12 col-md-12 col-sm-12 col-12 text-center">
            <h1>Our Features</h1>
          </div>
          <div class="col-xl-4 col-lg-4 col-md-4 col-sm-12 col-12">
            <div class="card text-center p-3">
              <div class="card-body">
                <h1 class="card-title"><i class="fas fa-thumbs-up"></i></h1>
                <h3 class="card-title">Quality</h3>
                <p class="card-text">Our very first priority is the quality we never
compromised in the quality of our bakery products.</p>
              </div>
            </div>
          </div>
          <div class="col-xl-4 col-lg-4 col-md-4 col-sm-12 col-12">
            <div class="card text-center p-3">
              <div class="card-body">
                <h1 class="card-title"><i class="fas fa-birthday-cake"></i></h1>
                <h3 class="card-title">Fresh & natural</h3>
                <p class="card-text">Our every product is fresh and made with natural
ingredients we do not use the artificial food ingredient in our products.</p>
              </div>
            </div>
          </div>
          <div class="col-xl-4 col-lg-4 col-md-4 col-sm-12 col-12">
            <div class="card text-center p-3">
              <div class="card-body">
                <h1 class="card-title"><i class="fas fa-shipping-fast"></i></h1>
                <h3 class="card-title">Free delivery</h3>
                <p class="card-text">We provide free delivery to our customers. We
deliver in 1 hr from the time customer order the product.</p>
              </div>
            </div>
          </div>
        </div>

        <div class="row mx-5">
          <div class="col-xl-12 col-lg-12 col-md-12 col-sm-12 col-12 text-center">
<h1>Our Trending</h1>
          </div>
            <div class="row owl-carousel owl-theme trend-foods">
              <?php
                require_once('config.php');
                $tspecial_query = "SELECT * FROM cake_shop_product WHERE
product_trend='1'";
                $tspecial_query_run = mysqli_query($conn, $tspecial_query);

                if($tspecial_query_run)
                {
                  if(mysqli_num_rows($tspecial_query_run) > 0)
                  {
                    foreach($tspecial_query_run as $item)
                    {
                      ?>
                        <div class="item">
                          <div class="box-card-one">
                            <div class="box-card-body">
                              <div class="product-img img-box">
                                <?php
                                $file_array = explode(', ', $item['product_image']);
?>
                                <img src="upload/<?php echo $file_array[0]; ?>"
class="img-fluid" alt="item image">
```

```
                                        </div>
                                        <div class="box-card-content">
                                        <h3 class="box-card-title"></h3>
                                        <h3 class="box-card-title"></h3>
                                          <h3 class="box-card-title"><?=
$item['product_name']; ?></h3>
                                          <h4 class="box-card-price">
                                            Rs: <?= $item['product_price']; ?>

                                            <span class="veg-non-veg-tag">
                                              <i class="fa fa-circle <?= $item['product_type']
== 'Veg' ? 'veg-item':'non-veg-item'; ?>"></i>
                                            </span>
                                          </h4>
                                          <div class="divider"></div>
                                          <div class="row">
                                            <div class="col-md-6 col-6">
                                              <div class="product_btn">
                                                <?php if($item['product_status'] == '0') : ?>
                                                <button onclick="add_cart(<?php echo
$item['product_id'];?>)" class="btn btn-primary"><i class="fa fa-shopping-cart"></i> Add to
Cart</button>
                                                <?php else : ?>
                                                <button class="btn btn-primary bg-danger
disabled">Unavailable</button>
                                                <?php endif; ?>

                                              </div>
                                            </div>
                                            <div class="col-md-6 col-6">
                                              <div>
                                              <a href="single_product.php?product_id=<?php
echo $item['product_id'];?>" class="btn btn-outline-light">Details</a>
                                              </div>
                                            </div>
                                          </div>
                                        </div>
                                      </div>
                                    </div>
                                  </div>
                                <?php
                              }
                            }
                          }
                        ?>
```

```
                </div>
            </div>
        </div>

        <div class="row m-5">
            <div class="col-xl-12 col-lg-12 col-md-12 col-sm-12 col-12 text-center">
<h1>Our Categories</h1>
            </div>
            <div class="col-xl-12 col-lg-12 col-md-12 col-sm-12 col-12">
                <div class="owl-carousel owl-theme">
                    <?php
                    require_once('config.php');
                    $select = "SELECT * FROM cake_shop_category";
                    $query = mysqli_query($conn, $select);
                    while ($res = mysqli_fetch_assoc($query)) {
                    ?>
                    <div class="item">
                        <div class="card h-100">
                            <div class="card-body">
                                <h3 class="card-title"><?php echo $res['category_name'];?></h3>
                                <a href="shop.php?category=<?php echo
$res['category_id'];?>"><img class="card-img" src="upload/<?php echo
$res['category_image'];?>"></a>
                            </div>
                        </div>
                    </div>
                    <?php
                    }
                    ?>
                </div>
            </div>
        </div>

        <div class="row m-5 hero-image2 rounded">
            <div class="col-xl-12 col-lg-12 col-md-12 col-sm-12 col-12 p-3 hero-text">
                <h1 class="text-dark">Who We Are</h1>
                <p class="text-dark px-5">We are bakers, we bake the piece of joy. We
believe cake and baked goods are an expression of love.</p>
                <p class="text-dark px-5">We bake from scratch daily using traditional
methods and quality ingredients. There are some things in life you just can't fake, and dang
good cake? That's one of them. We use organic whole milk, cage-free eggs, loads of real
fruit, pure extracts, amazingly delicious chocolate, and lots and lots of real butter to create
simply delicious treats the old-fashioned way.</p>
                <a href="about.php" class="btn btn-rounded btn-success">Read More</a>
            </div>
        </div>

        <div class="row mx-5 hero-image rounded">
            <div class="col-xl-12 col-lg-12 col-md-12 col-sm-12 col-12 p-3 hero-text">
                <h1 class="text-white">Always happy to hear from you.</h1>
                <a href="contact.php" class="btn btn-rounded btn-brand">Contact Us</a>
            </div>
        </div>
    </div>
```

Single_product.php

```
<div class="row mx-5">
            <?php
            require_once('config.php');
            $product_id = $_GET['product_id'];
            $select = "SELECT * FROM cake_shop_product where product_id =
$product_id";
            $query = mysqli_query($conn, $select);
            $res = mysqli_fetch_assoc($query);
            ?>
            <div class="offset-xl-2 col-xl-8 col-lg-12 col-md-12 col-sm-12 col-12">
                <div class="row">
                    <div class="col-xl-6 col-lg-6 col-md-6 col-sm-12 col-12 pr-xl-0 pr-lg-0 pr-
md-0 m-b-30">
                        <div class="product-slider p-4">
                            <div id="carouselExampleIndicators" class="product-carousel carousel
slide" data-ride="carousel">
                                <?php
                                $file_array = explode(', ', $res['product_image']);
?>
                                <div class="carousel-inner">
                                    <?php
                                    for ($i = 0; $i < count($file_array); $i++) {
?>
                                    <div class="carousel-item <?php if($i == 0) {echo 'active';} ?>">
                                        <img class="d-block w-100" src="upload/<?php echo
$file_array[$i];?>" alt="slide<?php echo $i;?>">
                                    </div>
                                    <?php } ?>
                                </div>
                                <a class="carousel-control-prev"
href="#carouselExampleIndicators" role="button" data-slide="prev">
                                    <span class="carousel-control-prev-icon"
ariahidden="true"></span>
```

```
                                            <span class="sr-only">Previous</span>
                                        </a>
                                        <a class="carousel-control-next"
href="#carouselExampleIndicators" role="button" data-slide="next">
                                            <span class="carousel-control-next-icon"
ariahidden="true"></span>
                                            <span class="sr-only">Next</span>
                                        </a>
                                    </div>
                                </div>
</div>
                            <div class="col-xl-6 col-lg-6 col-md-6 col-sm-12 col-12 pl-xl-0 pl-lg-0
plmd-0 border-left m-b-30 d-flex">
                                <div class="product-details p-4">
                                    <div class="border-bottom pb-3 mb-3">
                                        <div class="veg-non-veg-tag">
                                            <i class="fa fa-circle <?php echo $res['product_type'] == 'Veg' ?
'veg-item':'non-veg-item'; ?>"></i>
                                        </div>
                                        <h2 class="mb-3"><?php echo $res['product_name'];?></h2>
                                        <h3 class="mb-0 text-primary">Rs. <?php echo
$res['product_price'];?></h3>
                                    </div>
                                    <div class="product-description">
                                        <h4 class="mb-1">Descriptions</h4>
                                        <p><?php echo $res['product_description'];?></p>
                                        <?php if($res['product_status'] == '0') : ?>
                                        <button onclick="add_cart(<?php echo $res['product_id'];?>)"
class="btn btn-primary btn-block btn-lg"><i class="fa fa-shopping-cart"></i> Add to
Cart</button>
                                        <?php else : ?>
                                            <button class="btn btn-primary bg-danger
disabled">Unavailable</button>
                                            <?php endif; ?>
                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
```

Cart.php

```
            <div class="row mx-5">
                    <div class="col-xl-12 col-lg-12 col-md-12 col-sm-12 col-12">
                        <div class="card">
                            <div class="card-body">
```

```
                <div class="table-responsive">
                  <table class="table table-bordered">
                    <thead>
                      <tr>
                        <th>S. No.</th>
                        <th>Product Name</th>
                        <th>Price</th>
                        <th>Quantity</th>
                        <th>Total</th>
                        <th>Action</th>
                      </tr>
                    </thead>
                    <form method="post" action="insert_orders.php">
                    <tbody>
<?php
                      if ($printCount == 0) {
                      ?>
                      <tr>
                        <td colspan="6" align="center">Your cart is empty!</td>
                      </tr>
                      <?php } else { ?>
                      <?php
                      $total_amount = 0;
require_once('config.php');
                      for ($i=0; $i < count($_SESSION['cart']); $i++) {
$select = "SELECT * FROM cake_shop_product where product_id =
{$_SESSION['cart'][$i]}";
                        $query = mysqli_query($conn, $select);
                        $j = $i;
                        while ($res = mysqli_fetch_assoc($query)) {
                        $total_amount = $total_amount + $res['product_price'];
                      ?>
                      <tr>
                        <td><?php echo ++$j;?></td>
                        <td><?php echo $res['product_name'];?><input type="hidden"
name="hidden_product_name[]" value="<?php echo $res['product_name'];?>"></td>
                        <td>Rs. <?php echo $res['product_price'];?><input
type="hidden" name="hidden_product_price[]" value="<?php echo
$res['product_price'];?>"></td>
                        <td><input class="form-control" type="number" min="1"
max="30" step="1" value="1" name="product_quantity[]"
onchange="prodTotal(this)"></td>
                        <td><span>Rs. <?php echo $res['product_price'] *
1;?></span><input type="hidden" name="hidden_product_total[]" value="<?php echo
$res['product_price'];?>"></td>
```

```
                                    <td align="center"><a
href="remove_product.php?val_i=<?php echo $i;?>"><i class="fas fa-
trashalt"></i></a></td>
                                 </tr>
                                 <?php } ?>
                                 <?php } ?>
                                 <?php } ?>
                                 <tr>
                                    <td colspan="4" align="right">Total Amount:</td>
                                    <td colspan="2" id="total_amount"><span>Rs. <?php if
($printCount == 0){echo 0;} else {echo $total_amount;}?></span><input type="hidden"
name="hidden_total_amount" value="<?php echo $total_amount;?>"></td>
                                 </tr>
                                 <tr>
                                    <td colspan="3">
                                       Delivery Date:<input class="form-control" type="date"
name="delivery_date" min="2022-08-27" max="2022-09-27" required="">
                                    </td>
                                    <td colspan="3">
                                       <label>
                                          <input type="radio" name="radio1" value="Cash"
required="">Cash
                                          <span class="select"></span>
                                       </label>
                                       <label>
                                          <input type="radio" name="radio1" value="Card"
required="">Card
                                          <span class="select" onclick="card()"></span>
</label>

                                    </td>
                                 </tr>
                                 <tr>
                                    <td colspan="6" align="right">
<button class="btn btn-warning"
onclick="clear_cart()">Clear</button>
                                    <button class="btn btn-primary"
type="submit">Checkout</button>
                                    </td>
                                 </tr>
                              </tbody>
                              </form>
                           </table>
                        </div>
                     </div>
```

```
                </div>
             </div>
        </div>
```

Clear_cart.php

```
<?php
session_start(); unset($_SESSION['cart']);
header("Location: cart.php");
?>
```

Fetch_cart.php

```
<?php
session_start(); $id
= $_GET['id']; if
(!in_array($id,
$_SESSION['cart']
)) {
   $_SESSION['cart'][] = $id;
}
echo json_encode($_SESSION['cart']);
?>
```

Insert_order.php

```
<?php
require_once('config.php');
session_start();
if (isset($_SESSION['cart']) && $_SESSION['cart'] !== null) {
if (isset($_SESSION['user_users_id']) && $_SESSION['user_users_username'] !== null) {
   $username = $_SESSION['user_users_username'];
   $users_id = $_SESSION['user_users_id'];
   $product_name = $_POST['hidden_product_name'];
   $price = $_POST['hidden_product_price'];
   $quantity = $_POST['product_quantity'];
   $total = $_POST['hidden_product_total'];
   $total_amount = $_POST['hidden_total_amount'];
   $delivery_date = $_POST['delivery_date'];
   $payment_method = $_POST['radio1'];

   if($payment_methond=="Card")
   {
      header("Location: onlinepay.php");
   }
   $insert_orders = "INSERT INTO cake_shop_orders (users_id, delivery_date,
payment_method, total_amount) values ('$users_id', '$delivery_date', '$payment_method',
'$total_amount')";
   mysqli_query($conn, $insert_orders);
$orders_id = mysqli_insert_id($conn);     for
($i=0; $i < count($product_name); $i++) {
       $insert_orders_detail = "INSERT INTO cake_shop_orders_detail (orders_id,
product_name, quantity) values ('$orders_id', '$product_name[$i]', '$quantity[$i]')";
mysqli_query($conn, $insert_orders_detail);
   }
   unset($_SESSION['cart']);
   header("Location: cart.php?order_success=1");
} else {     echo "<script>
alert('Please login!!!');
location.assign('login_users.php');
</script>";
} } else {     echo "<script>
alert('Please select a product!!!');
location.assign('cart.php');
   </script>";
}
?>
```

Insert_user.php

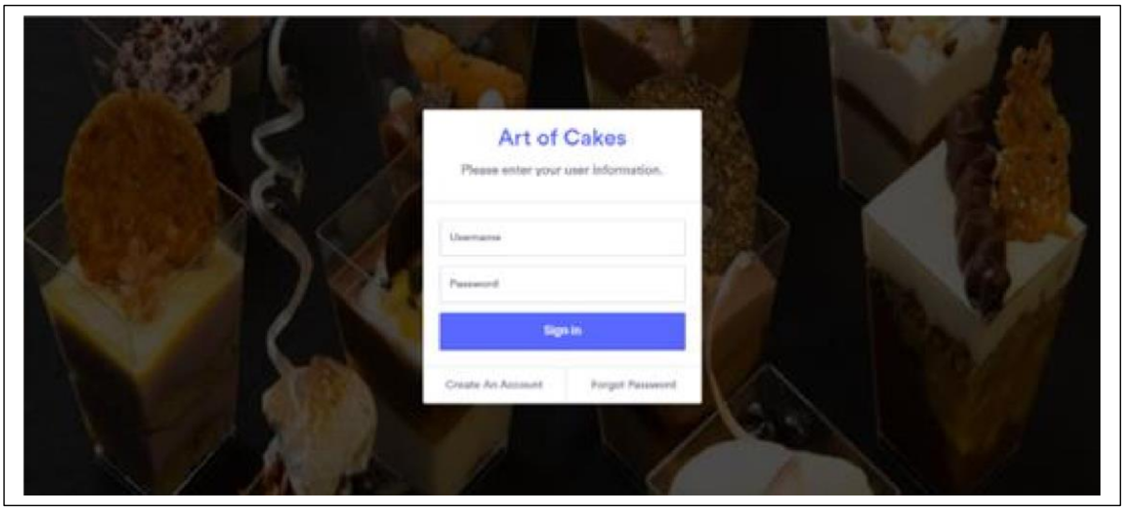
```
<?php
require_once('config.php');
$users_username = $_POST['users_username'];
$users_email = $_POST['users_email'];
$users_password = $_POST['users_password'];
$users_mobile = $_POST['users_mobile'];
$users_address = $_POST['users_address'];
$insert = "INSERT INTO cake_shop_users_registrations (users_username, users_email,
users_password, users_mobile, users_address) values ('$users_username', '$users_email',
'$users_password', '$users_mobile', '$users_address')";
$select = "SELECT * FROM cake_shop_users_registrations where users_username =
'$users_username'";
$query = mysqli_query($conn, $select);
$res = mysqli_fetch_assoc($query); if
(mysqli_num_rows($query) > 0) {
   header("Location: register.php?register_msg=1");
} else
{
   mysqli_query($conn, $insert);
header("Location: login_users.php");
}
?>
```

Logout.php

```
<?php
session_start();
unset($_SESSION['user_users_id']);
unset($_SESSION['user_users_username']);
header("Location: index.php?logout_success=1");
?>
```

USER INTERFACE

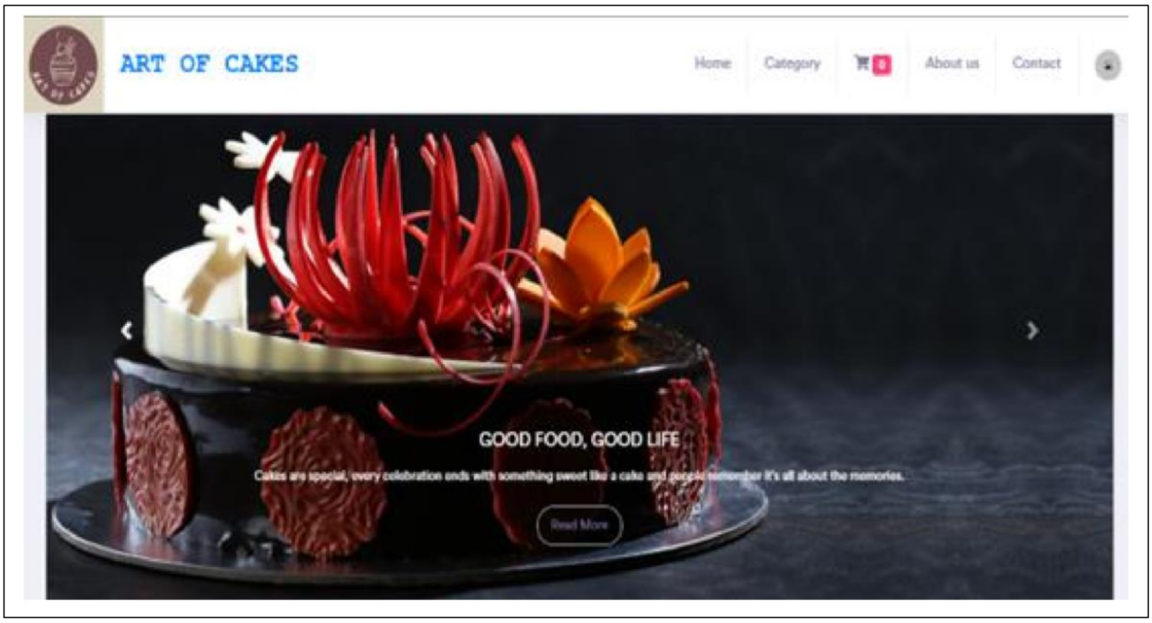
7.1 LOGIN:



Art of Cakes

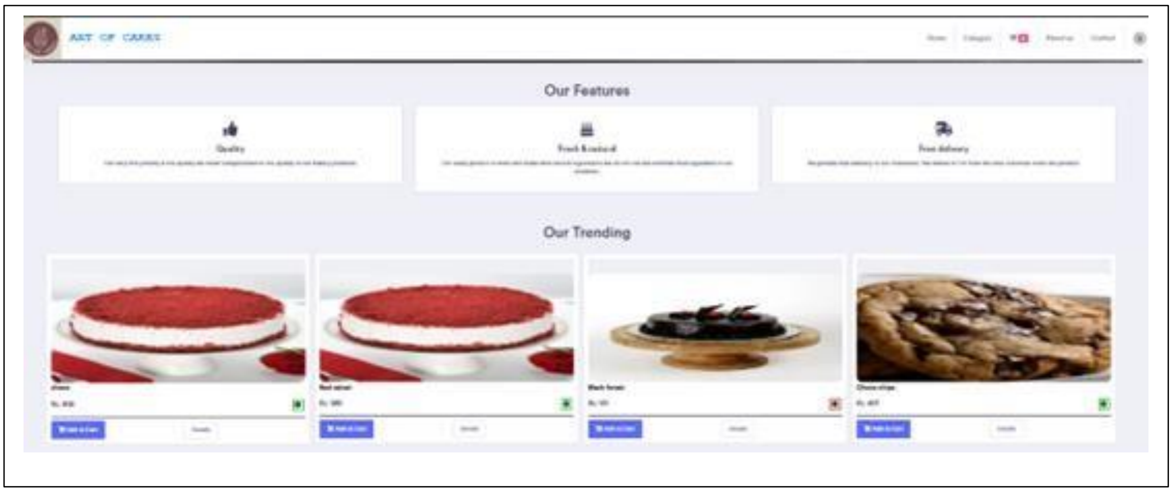
Please enter your user information.

7.2 MAIN SCREEN/HOMEPAGE:

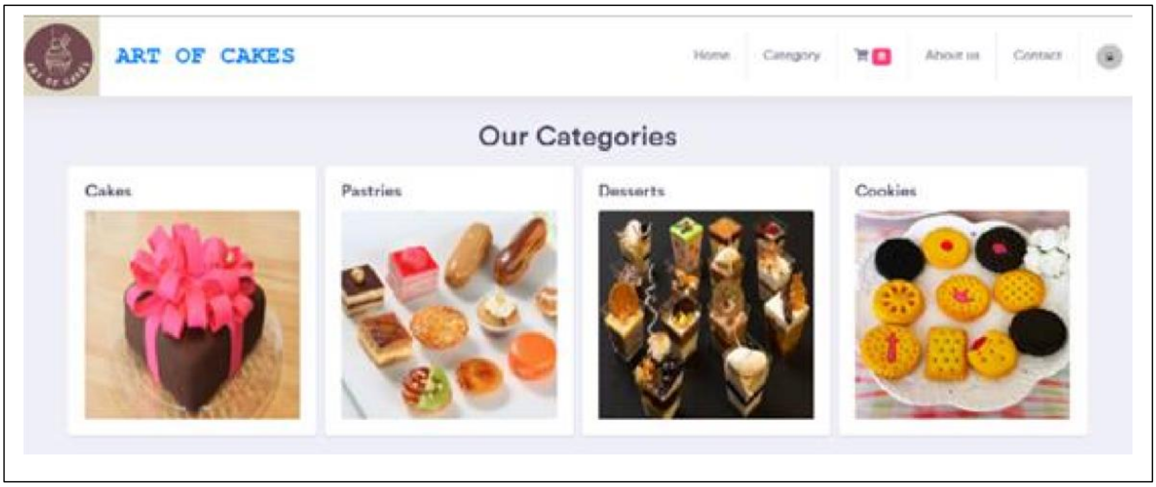


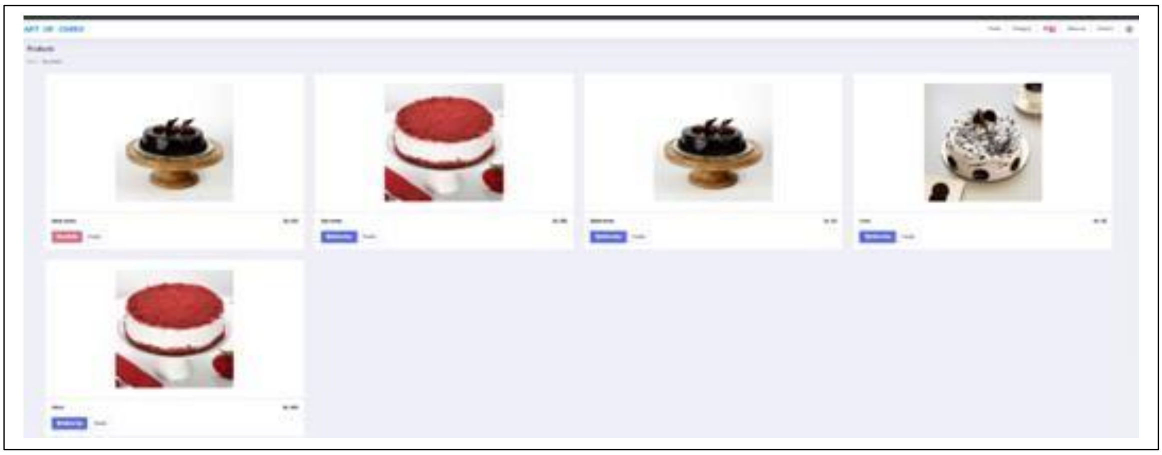
GOOD FOOD, GOOD LIFE

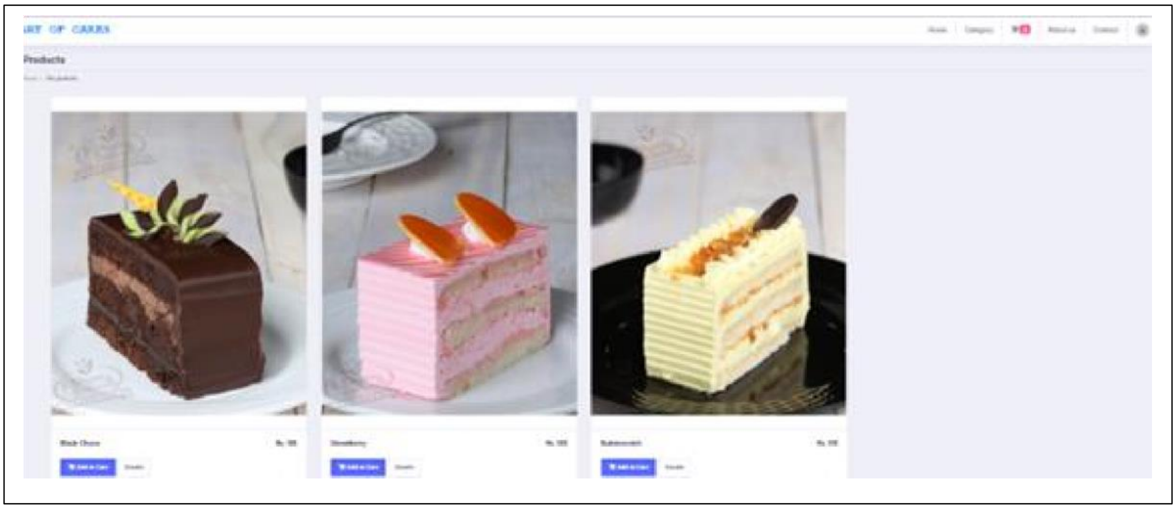
Cakes are special, every celebration ends with something sweet like a cake and people remember it's all about the memories.

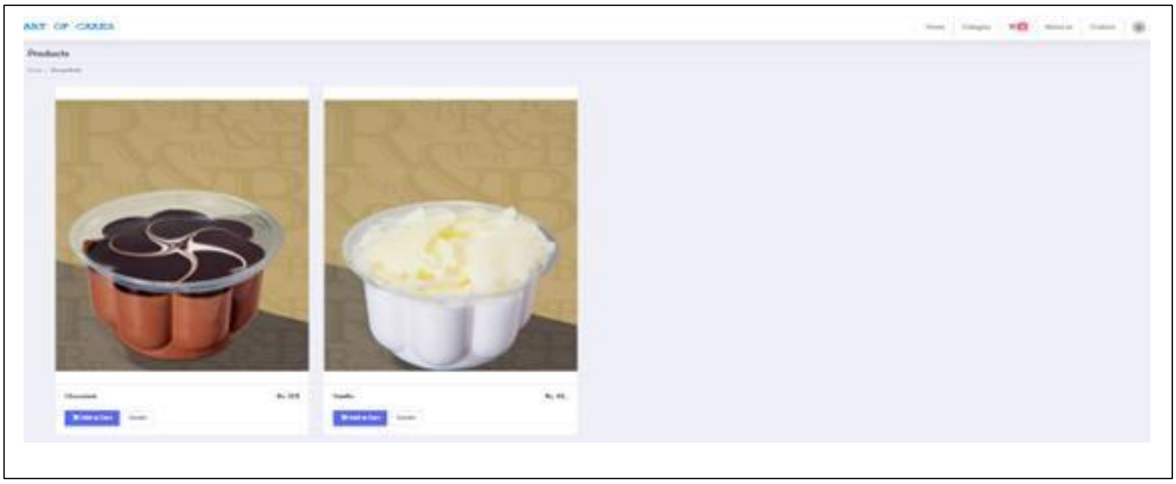

ART OF CAKES
Our Features
Quality
Fresh & natural
Free delivery
Our Trending
cheese
Red velvet
Black forest
Choco chips

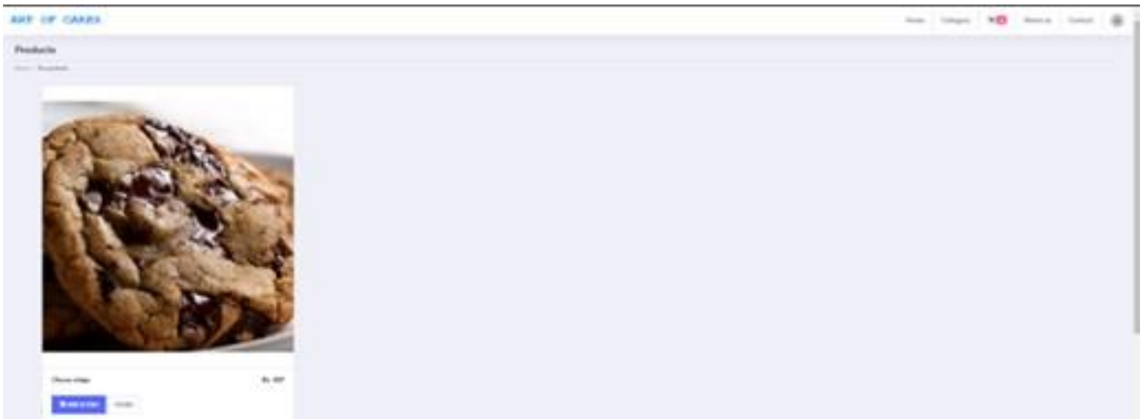
7.3 MENU:


ART OF CAKES
Home
Category
About us
Contact
Our Categories
Cakes
Pastries
Desserts
Cookies




ART OF CAKES
Home
Category
About us
Contact
Products
Black Choco
Strawberry
Butterscotch
Add to Cart
Details


ART OF CAKES
Home
Category
About us
Contact
Products
Chocolate
Vanilla
Add to Cart
Details



7.4 DATA STORE / RETRIEVE:

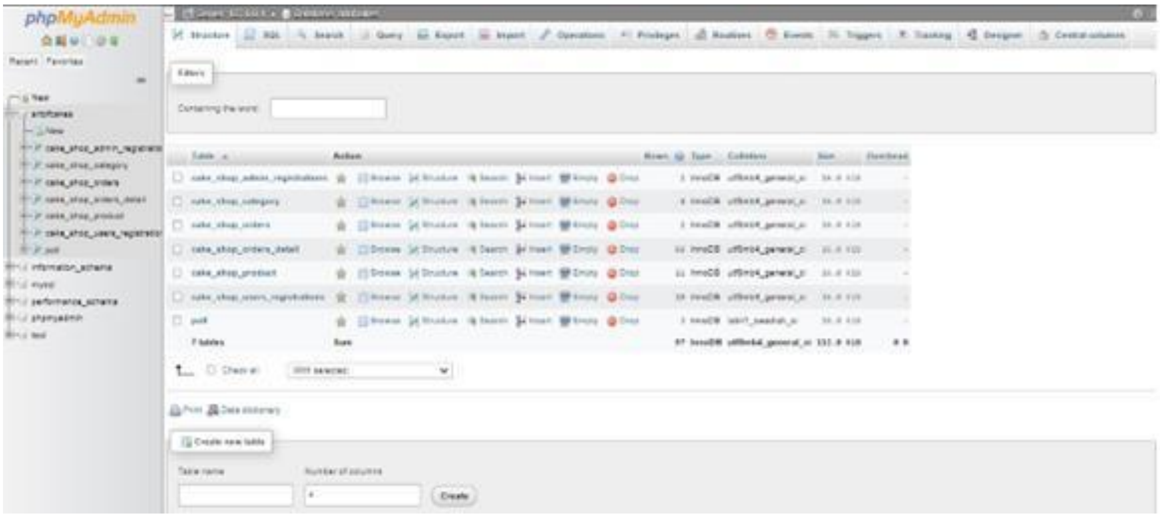


	Table	Action	Rows	Type	Collation	Size	Overhead
□	cake_shop_admin_registrations	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	utf8mb4_general_ci	[illegible]	-
□	cake_shop_category	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	utf8mb4_general_ci	[illegible]	-
□	cake_shop_orders	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	utf8mb4_general_ci	[illegible]	-
□	cake_shop_orders_detail	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	utf8mb4_general_ci	[illegible]	-
□	cake_shop_product	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	utf8mb4_general_ci	[illegible]	-
□	cake_shop_users_registrations	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	utf8mb4_general_ci	[illegible]	-
□	pull	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	[illegible]	[illegible]	-
	7 tables	Sum	[illegible]	InnoDB	[illegible]	[illegible]	[illegible]

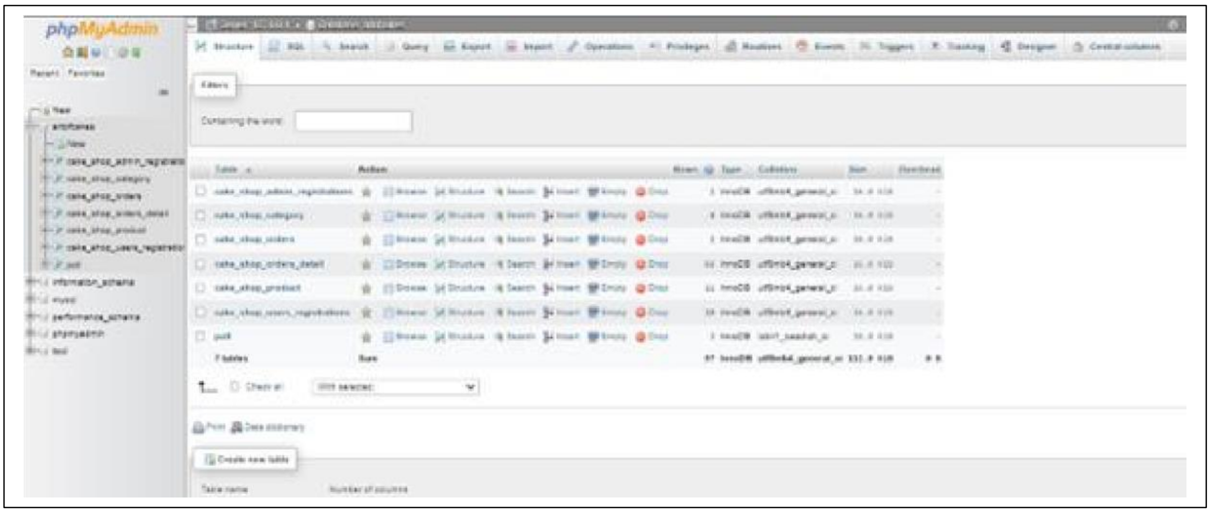
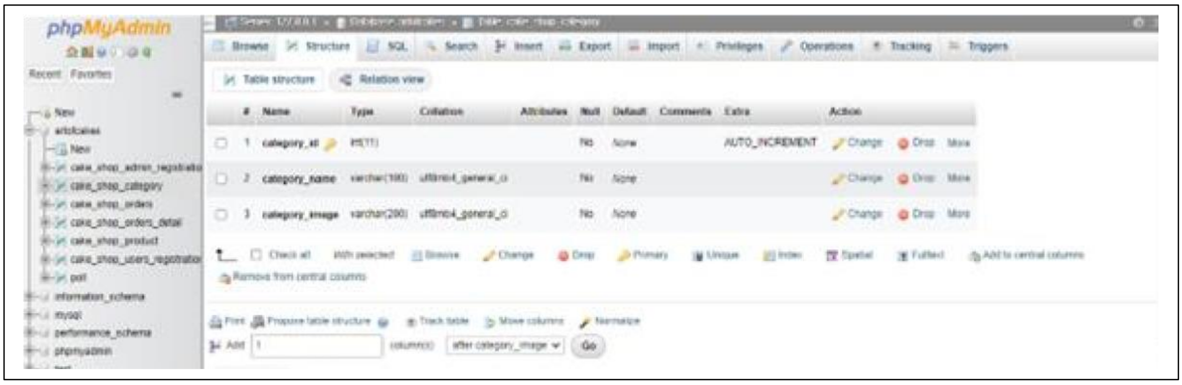
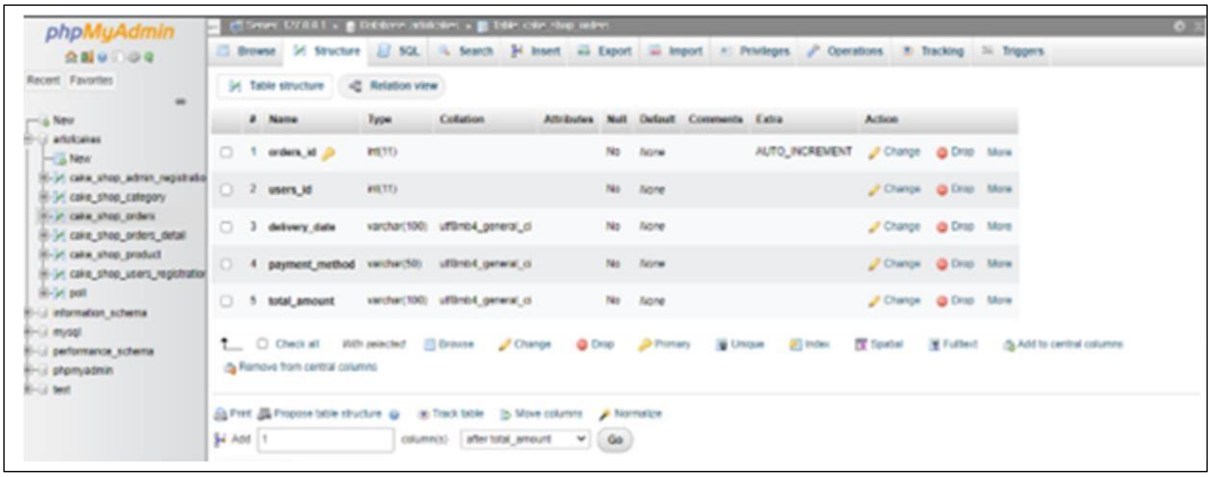


Table	Action	Rows	Type	Collation	Size	Overhead
cake_shop_admin_registrations	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_general_ci	16.0 KiB	-
cake_shop_category	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	utf8mb4_general_ci	16.0 KiB	-
cake_shop_orders	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_general_ci	16.0 KiB	-
cake_shop_orders_detail	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	utf8mb4_general_ci	16.0 KiB	-
cake_shop_product	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	utf8mb4_general_ci	16.0 KiB	-
cake_shop_users_registrations	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	utf8mb4_general_ci	16.0 KiB	-
poll	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16.0 KiB	-
7 tables	Sum	[illegible]	InnoDB	utf8mb4_general_ci	112.0 KiB	0 B



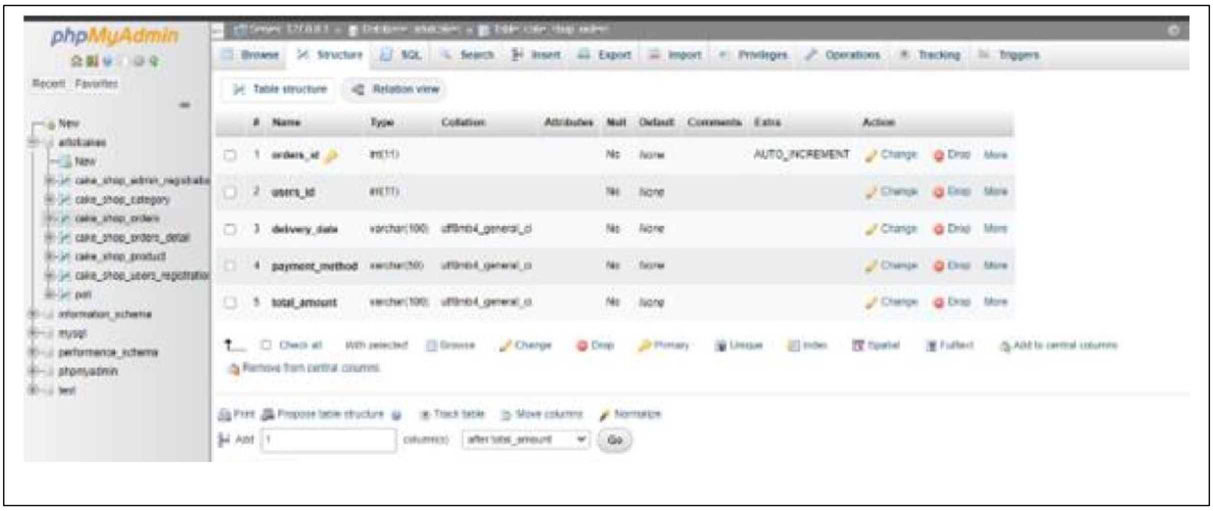
Server: 127.0.0.1 » Database: artofcakes » Table: cake_shop_category

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	category_id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	category_name	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
3	category_image	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More

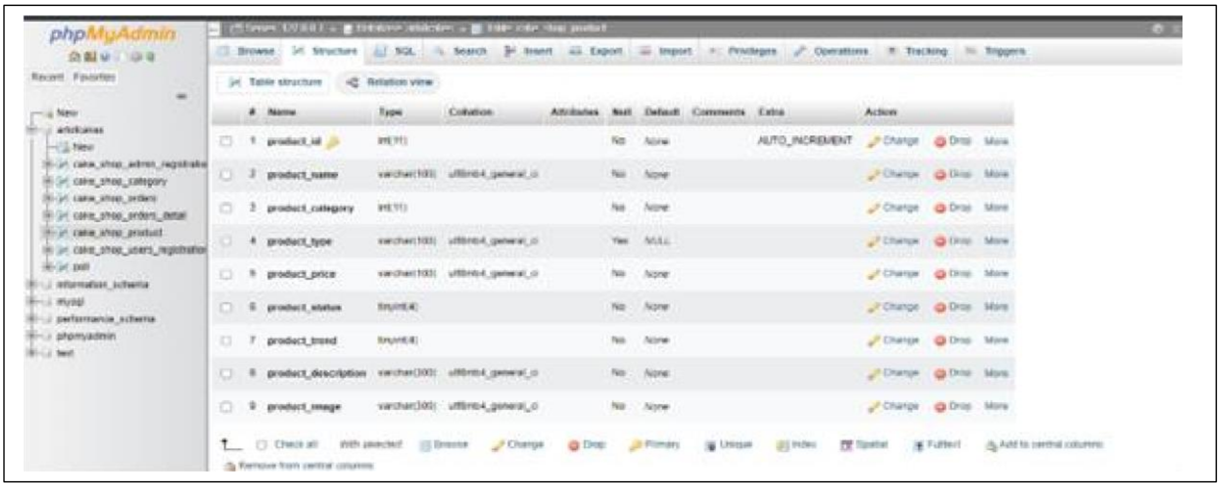


	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	orders_id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
□	2	users_id	int(11)			No	None			Change Drop More
□	3	delivery_date	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
□	4	payment_method	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
□	5	total_amount	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More

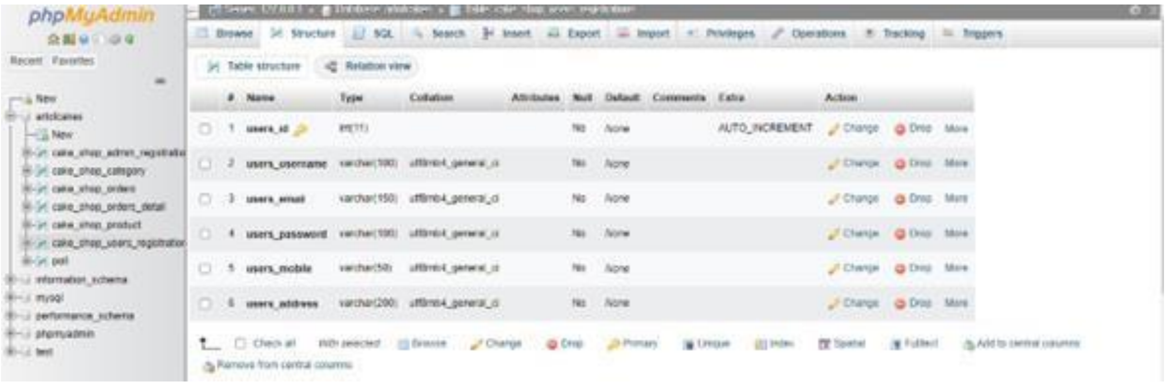
Add 1 column(s) after total_amount


	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	orders_id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
□	2	users_id	int(11)			No	None			Change Drop More
□	3	delivery_date	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
□	4	payment_method	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
□	5	total_amount	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More

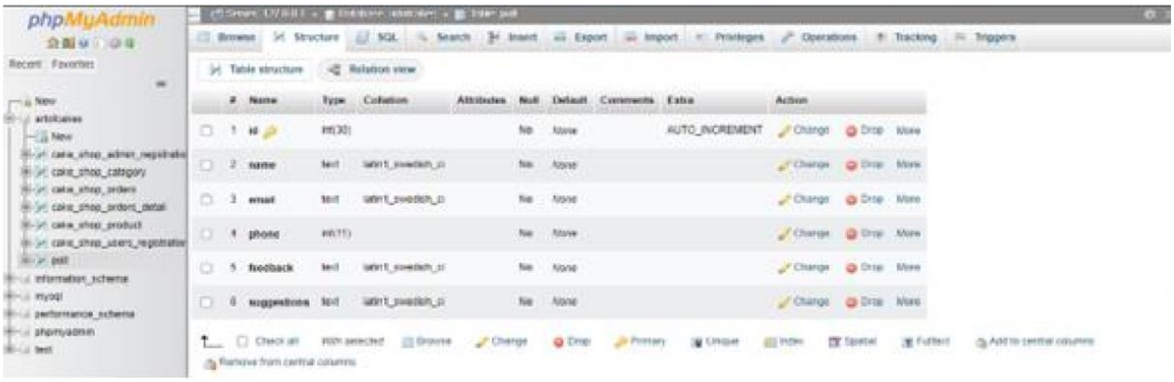
Add 1 column(s) after total_amount



	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	product_id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
□	2	product_name	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
□	3	product_category	int(11)			No	None			Change Drop More
□	4	product_type	varchar(100)	utf8mb4_general_ci		Yes	NULL			Change Drop More
□	5	product_price	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
□	6	product_status	tinyint(4)			No	None			Change Drop More
□	7	product_trend	tinyint(4)			No	None			Change Drop More
□	8	product_description	varchar(300)	utf8mb4_general_ci		No	None			Change Drop More
□	9	product_image	varchar(300)	utf8mb4_general_ci		No	None			Change Drop More

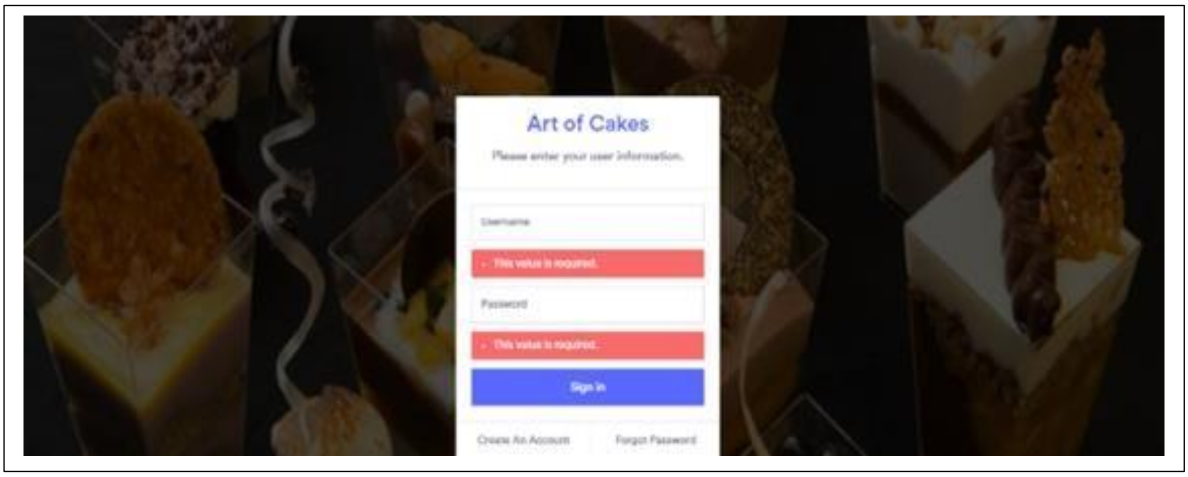


#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	users_id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	users_username	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
3	users_email	varchar(150)	utf8mb4_general_ci		No	None			Change Drop More
4	users_password	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
5	users_mobile	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
6	users_address	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(30)			No	None		AUTO_INCREMENT	Change Drop More
2	name	text	latin1_swedish_ci		No	None			Change Drop More
3	email	text	latin1_swedish_ci		No	None			Change Drop More
4	phone	int(11)			No	None			Change Drop More
5	feedback	text	latin1_swedish_ci		No	None			Change Drop More
6	suggestions	text	latin1_swedish_ci		No	None			Change Drop More

7.5 VALIDATION:



Art of Cakes

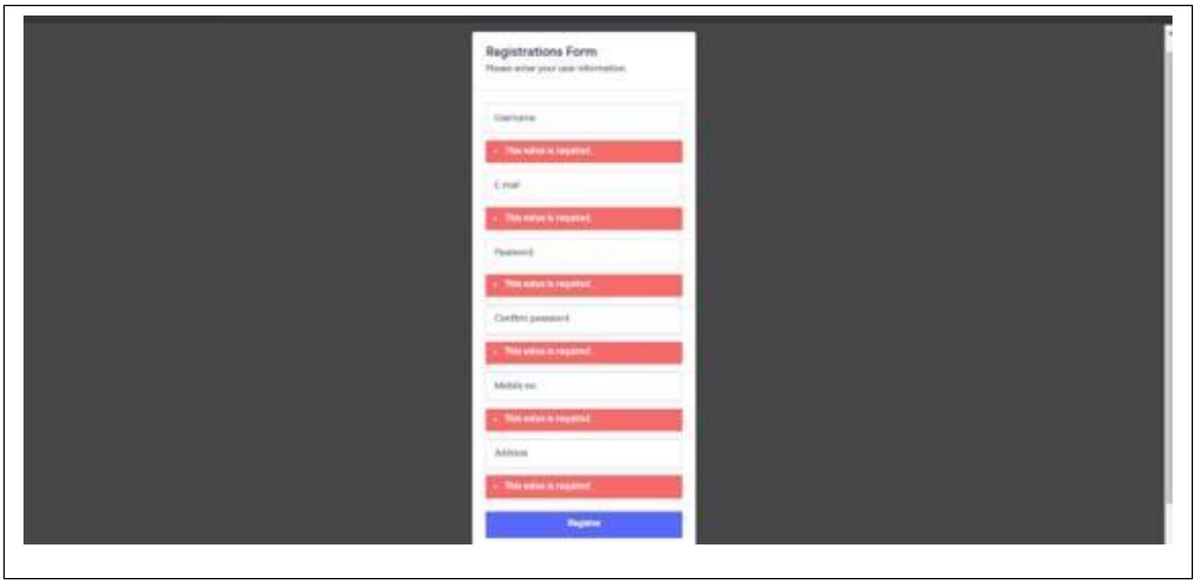
Please enter your user information.

Username

This value is required.

Password

This value is required.



Registrations Form

Please enter your user information.

Username

This value is required.

E mail

This value is required.

Password

This value is required.

Confirm password

This value is required.

Mobile no.

This value is required.

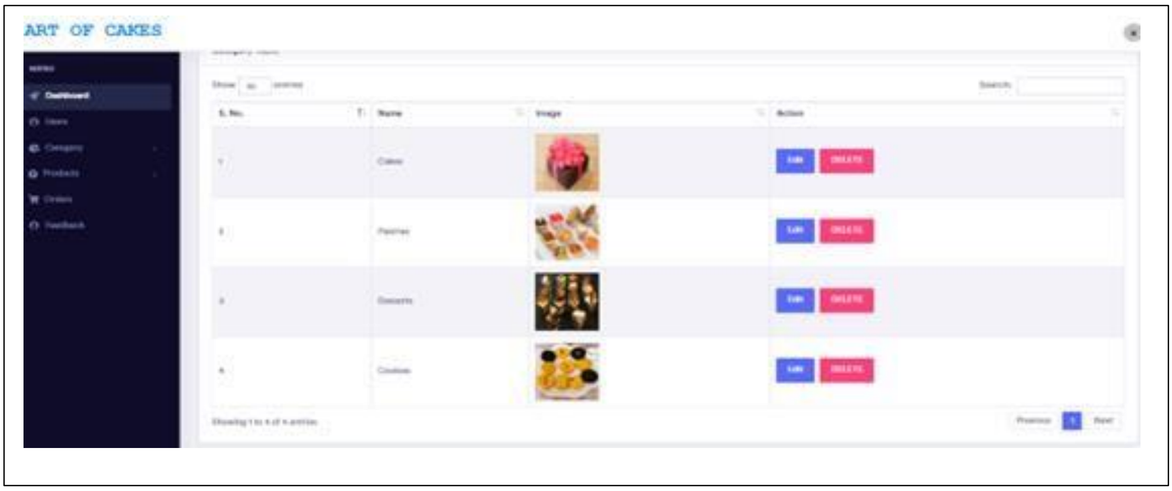
Address

This value is required.

7.6 VIEW:

ADMIN

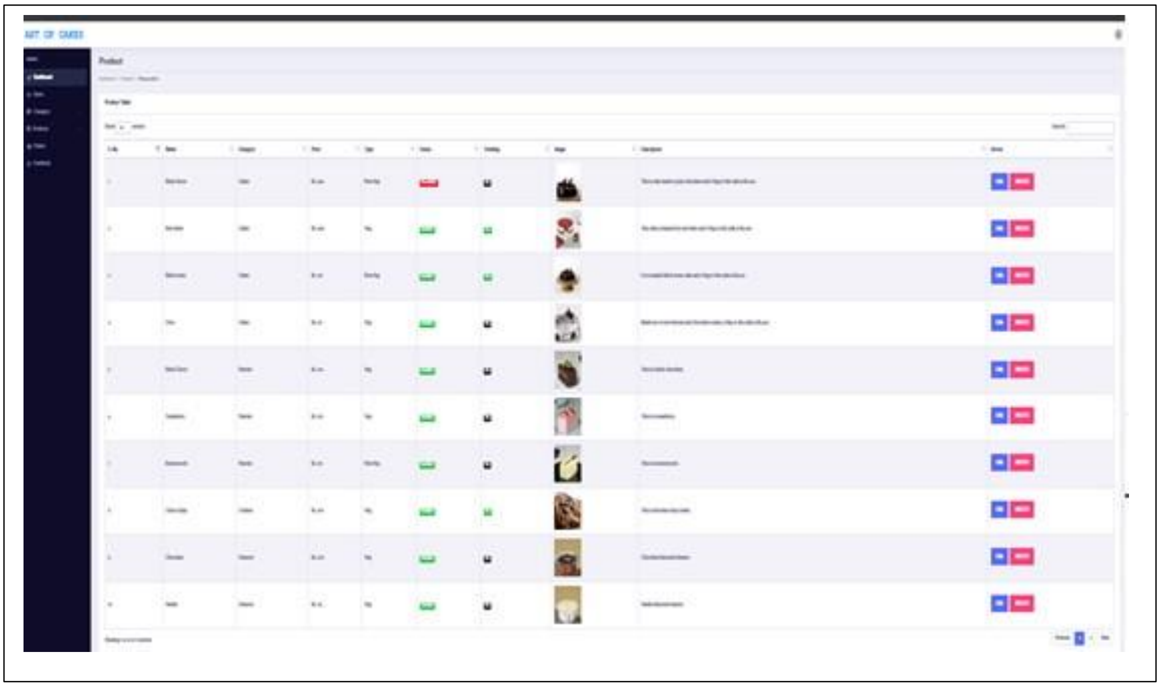
7.6.1 View category:



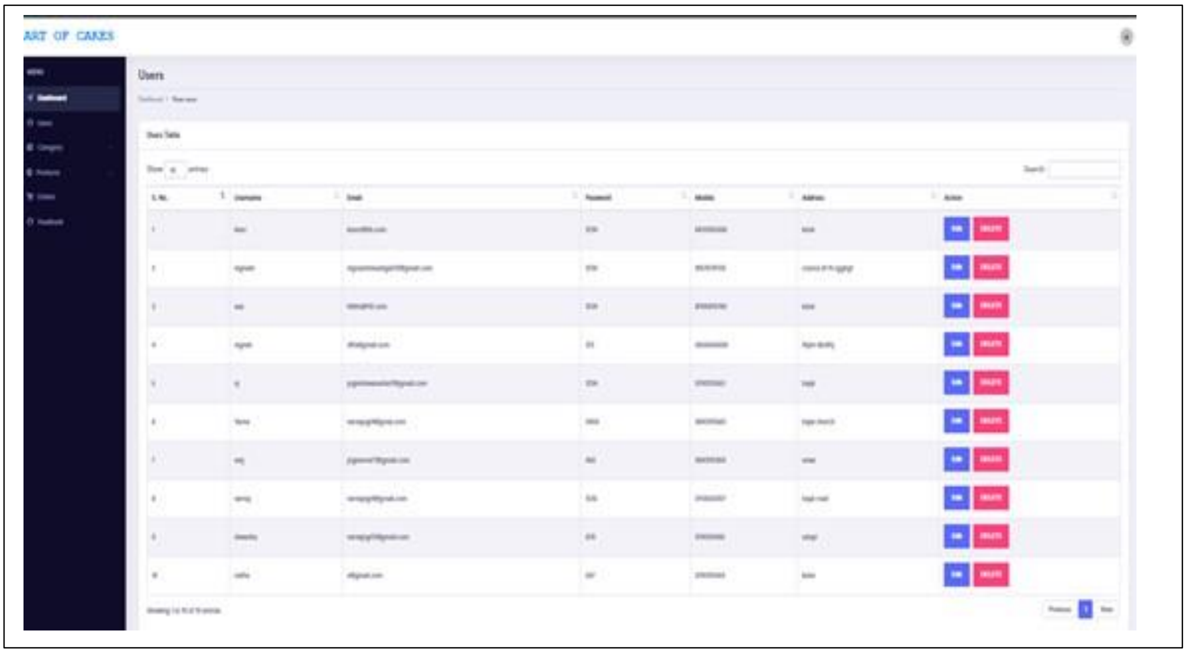
S. No.	Name	Image	Action
1	Cakes		Edit DELETE
2	Pastries		Edit DELETE
3	Desserts		Edit DELETE
4	Cookies		Edit DELETE

Showing 1 to 4 of 4 entries

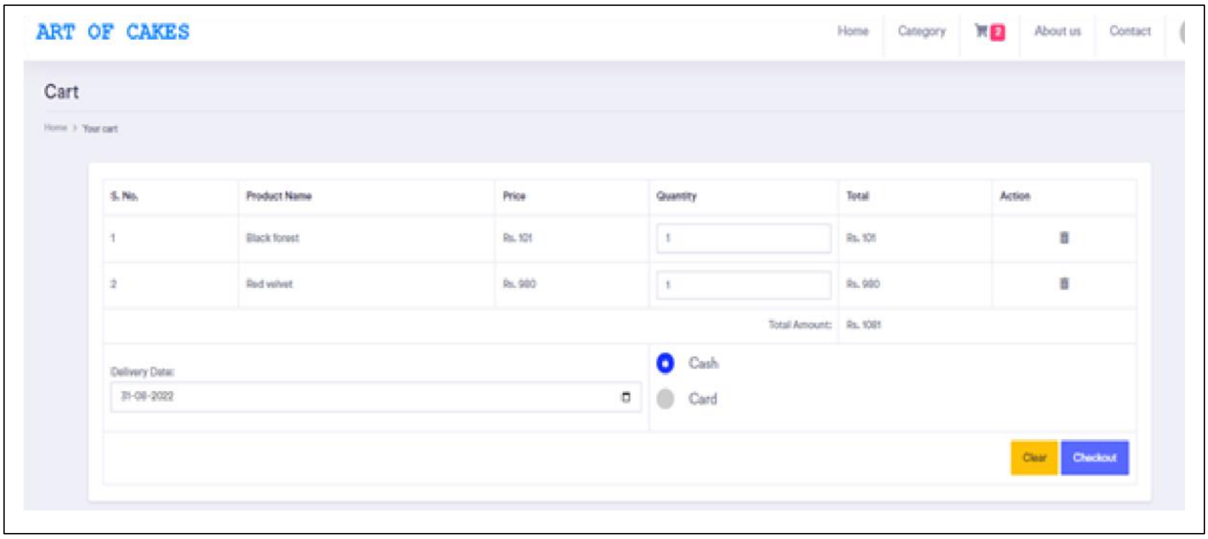
7.6.2 view product:



7.6.3 view users:



USER



Cart

Home > Your cart

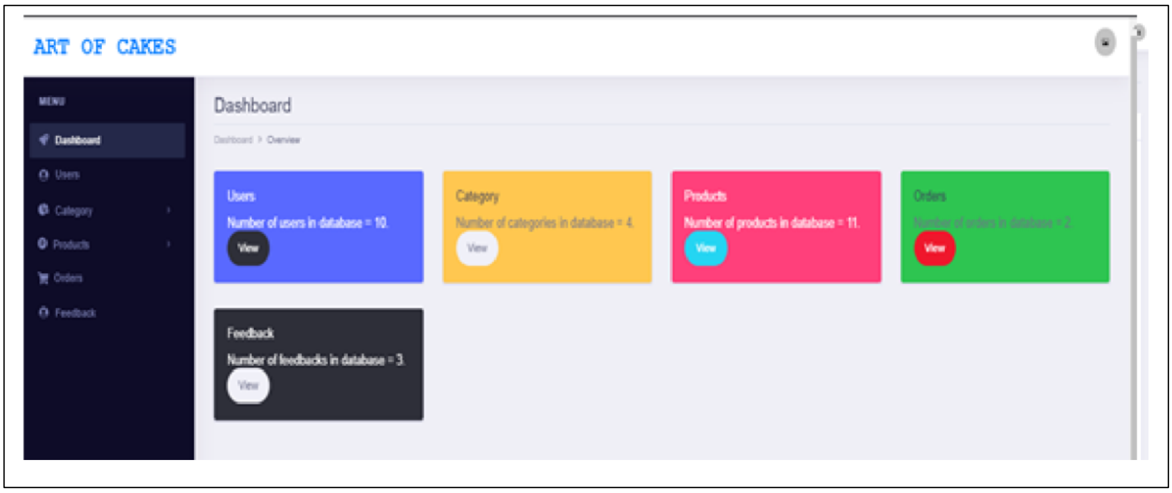
S. No.	Product Name	Price	Quantity	Total	Action
1	Black forest	Rs. 101	1	Rs. 101	🗑
2	Red velvet	Rs. 980	1	Rs. 980	🗑
			Total Amount:	Rs. 1081	

Delivery Date: 31-08-2022

● Cash

○ Card

7.7 ON SCREEN REPORTS



Users
Number of users in database = 10.
View

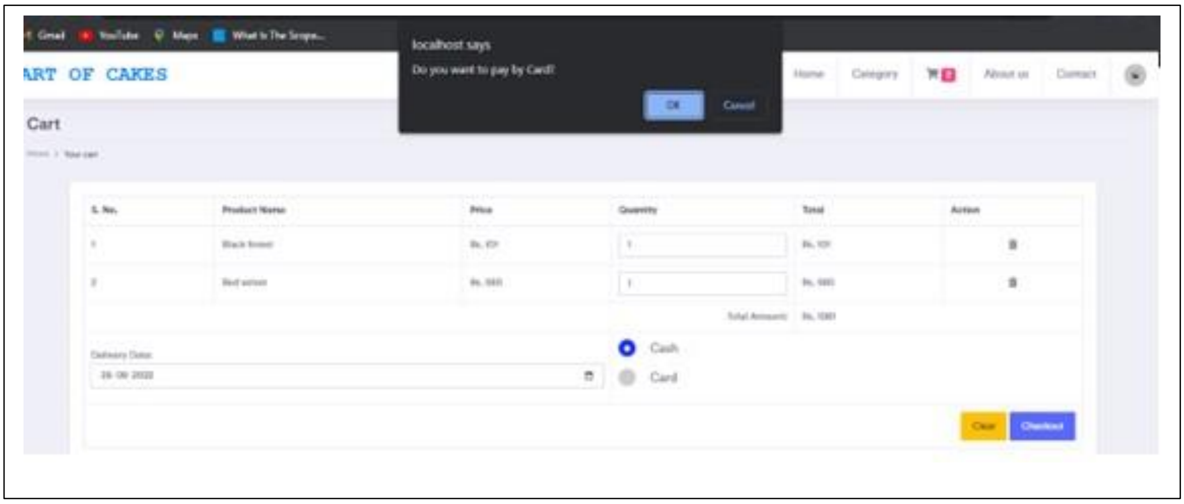
Category
Number of categories in database = 4.
View

Products
Number of products in database = 11.
View

Orders
Number of orders in database = 2.
View

Feedback
Number of feedbacks in database = 3.
View

7.8 ALERTS



localhost says

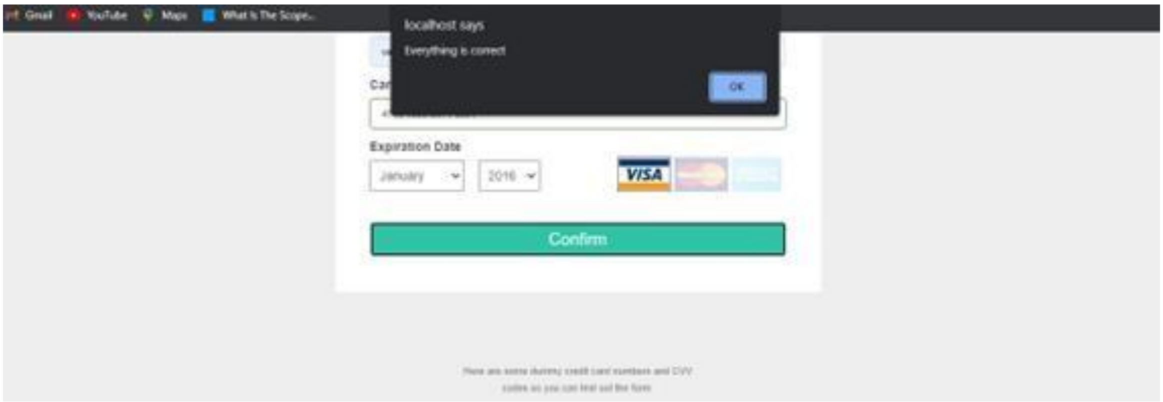
Do you want to pay by Card?

OK Cancel

S. No.	Product Name	Price	Quantity	Total	Action
1	Black forest	[illegible]	1	[illegible]	🗑
2	Red velvet	[illegible]	1	[illegible]	🗑
			Total Amount:	[illegible]	

Delivery Date: [illegible]

● Cash
○ Card



localhost says

Everything is correct

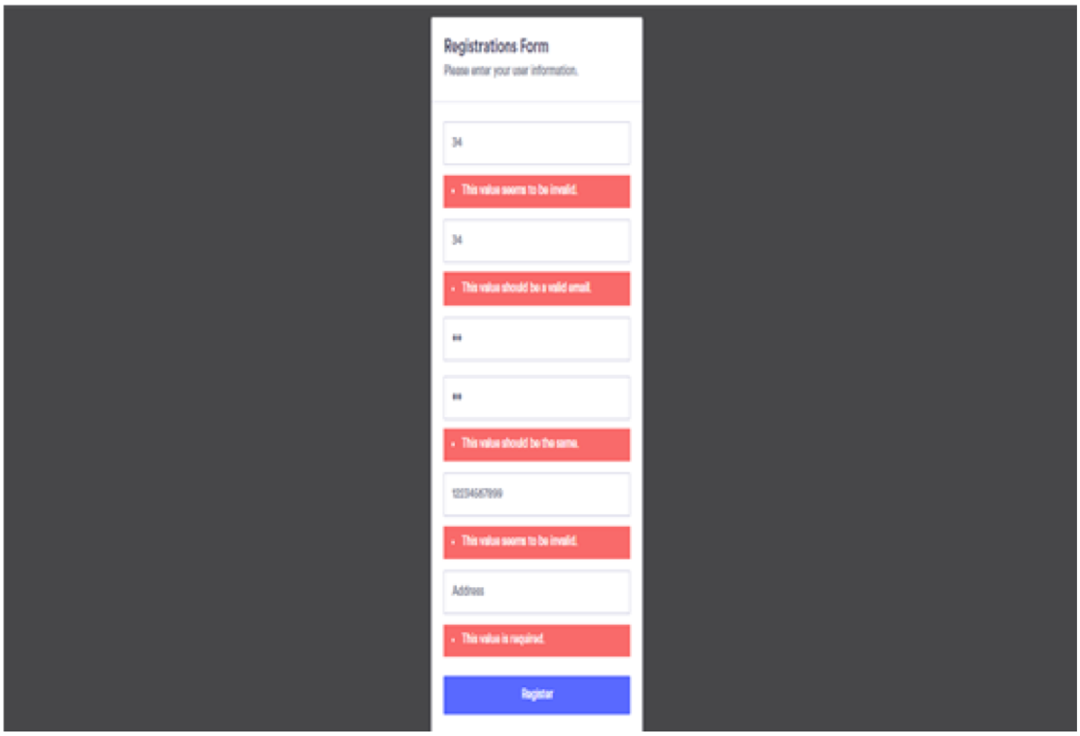
Expiration Date

January

2016

Here are some dummy credit card numbers and CVV codes so you can test out the form

7.9 ERROR MESSAGES


Registrations Form

Please enter your user information.

34

- This value seems to be invalid.

34

- This value should be a valid email.

••

••

- This value should be the same.

12234567899

- This value seems to be invalid.

Address

- This value is required.

TESTING

8.1 INTRODUCTION

Testing has been defined as the process of analysing a software item to detect the differences between existing and required conditions and to evaluate the features of the software item. Software testing is the process used to assess the quality of computer software. It involves operation of a system or application under controlled conditions and evaluating the results. The controlled conditions should include both normal and abnormal conditions. Testing should intentionally attempt to make things go wrong to determine if things happen when they should. It is oriented to 'detection'.

Software testing has three main purposes:

- The verification process confirms that the software meets its technical specifications. A "specification" is a description of a function in terms of a measurable output value given a specific input value under specific preconditions.
- The validation process confirms that the software meets the business requirements
- A defect is a variance between the expected and actual result. The defect's ultimate source may be traced to a fault introduced in the specification, design, or development phases. Not all the defects will necessarily result in failures.

There are two types of software testing:

- **Black box testing**-internal system design is not considered in this type of testing. test is based on requirements and functionality
- **White box testing**-this testing is based on knowledge of the internal logic of an application's code, also known as glass box testing. Internal software and code working should be known for this type of testing. Tests are based on coverage of code statements, branches, paths and conditions.

  A test case is a software testing document, which consists of event, action, input, output, expected result and actual result. Clinically defined a test case is an input and an expected result. This can be pragmatic as 'for condition x your derived result is y, where as other test cases described in more detail the input scenario and what results might be expected. It can occasionally be a series of steps but one with expected results or expected outcome. A test case should also contain a place for the actual result.

8.2 TEST REPORTS

Testing for valid user name:

TEST CASE	INPUT	TEST DESCRIPTION	OUTPUT
1	User name starts with number	Name cannot have number	Invalid name
2	User name is left blank	User name cannot be left blank	Please enter Required details

Testing for valid password:

TEST CASE	INPUT	TEST DESCRIPTION	OUTPUT
1	Password is left blank	Password cannot be blank	Please enter password
2	Invalid password entered	Enter valid password	Entered password does not matcg

Testing for valid email address:

TEST CASE	INPUT	TEST DESCRIPTION	OUTPUT
1	Email address is not in correct format	Email address should have correct format	Invalid email address
2	Email is left blank	Email cannot br blank	Please the enter email

Testing for data insertion:

TEST CASE	INPUT	TEST DESCRIPTION	OUTPUT
1	Mandatory fields left empty	Mandatory fields cannot be left empty	Please enter the required details

2	Duplicate entry	Duplicate entry is not allowed	Appropriate error message
3	Input without above faults	Valid input	Record inserted successfully

Testing for phone number

TEST CASE	INPUT	TEST DESCRIPTION	OUTPUT
1	Phone number with alphabets	Phone number cannot have alphabets	Please enter valid contact number

Testing for change password

TEST CASE	INPUT	TEST DESCRIPTION	OUTPUT
1	Mandatory fields left blank	Mandatory fields cannot be left blank	Please enter required details
2	Valid input	Valid input	Password changed successfully

Other cases

TEST CASE	INPUT	TEST DESCRIPTION	OUTPUT
1	Click on logout	Application will redirect to start page	Application closes and redirects to introduction page
2	Click on submit	Entering details	Record is submitted to the database
3	Click on submit	Not entering details	Appropriate error message is displayed
4	Click on drop down list	Selecting data	Data is displayed

8.2.1 Unit Testing

Unit tests are written from a programmer's perspective. They ensure that a particular method of a class successfully performs a set of specific tasks. Each test confirms that a method produces the expected output when given a known input. Each test examines an individual component that is new or has been modified. Unit tests focus on functionality and reliability. Unit testing is done in a test environment prior to system integration. If a defect is discovered at unit test, the severity of the defect will dictate whether or not it will be fixed before the module is approved.

8.2.2 Integrate Testing

Integration testing examines all the components and modules that are new, changed, affected by a change, or needed to form a complete system. Integration testing requires involvement of other systems and interfaces with other applications, including those owned by an outside vendor, external partners, or the customer it is the phase of software testing in which individual software modules are combined and tested as a group. it follows unit testing and precedes system testing. integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

Therefore, without adequate testing, there is a greater risk that an application will inadequately deliver what was expected by the admins or that the final product will have problems such that the admins will eventually abandon it out of frustration. In either case, time and money are lost and the credibility and reputation of both the developers and the software is damaged. More formal, rigorous testing will go far to reducing the risk that either of these scenarios occurs.

8.2.3 System Testing

Apart from testing the system to validate the functionality of software against the requirements. It is also necessary to test the non-functional aspect of the system, some examples of non-functional tools include tests to check the performance data security usability volume load and stress that we have used in a project to test the various module.

CONCLUSION:

The project work entitled “ART OF CAKES- Web Based Online cake shopping System” has been designed using JAVASCRIPT, CSS, HTML as front end and PHP, MYSQL as back end, were in many users friendly from have been added in order to make it a user interactive application. The system is developed in such a way that the user with basic knowledge of computer can work with it easily. The system has produced the report and fulfilled all the objectives.

LIMITATION:

- System is ineffective if there is no Internet Connection.
- If user required more kg cake, then they have to directly contact the admin.

SCOPE FOR ENHANCEMENT

- System can be deployed in either public or private servers.
- System can be upgraded so that it runs in different platform and with more functionality.

ABBREVATION AND ACRONYMS(LIST)

GUI		-Graphical User Interface
DBMS		-Database Management System
RDBMS		-Relational Database Management System
SRS		-Software Requirement Specification
DFD		-Data Flow Diagram
ERD		-Entity Relationship Diagram
API		-Application Programming Interface
ADMIN		-The Administrator
CFD		-Context Flow Diagram
PHP		-Hypertext Pre processor
SQL		-Structured Query Language
HTML		-Hyper Text Markup Language

BIBLIOGRAPHY

- Pankaj Jalote – An integrated approach to software engineering.
- Publication: Narosa publication house.
- Luke Welling, Luaura Thomson, PHP and MySQL Web Development, Developer's Library, Sams Publishing.
- Steven Holzner, PHP the Complete Reference, Mc Graw Hill, 2007

References

- www.Google.com
- www.wikipedia.com
- www.geeksforgeeks.com
- www.msdn.microsoft.com/liberies
- www.tutorialpoint.com
- www.stackoverflow.com
- www.youtube.com
- www.github.com