Government Women's Polytechnic College, Thrissur

[Approved by All India Council for Technical Education]

NBA Accredited Diploma in Computer Engineering(2019-2022)

Nedupuzha, Thrissur, Kerala, INDIA – 680 007

webs:gwpctsr.ac.in

Ph:487-2449182, Email: wpcnedupuzha@gmail.com

Project

Report On

ONLINE GROCERY SHOPPING WEB APPLICATION - SHOPFRESH

By

T. S. PARVATHY(REG.NO: 19138360)

Semester 6, Computer Engineering

2021 – 22

Government Women's Polytechnic College, Thrissur

[Approved by All India Council for TechnicalEducation]
NBA Accredited Diploma in Computer Engineering
(2019-2022)
Nedupuzha, Thrissur, Kerala, INDIA – 680 007
webs: gwpctsr.ac.in
Ph:487-2449182, Email: wpcnedupuzha@gmail.com

CERTIFICATE

This is to certify that the Project report titled **ONLINE GROCERY SHOPPING WEB APPLICATION – SHOPFRESH** has been completed by **T. S. PARVATHY (Register No: 19138360)** of Semester VI, in partial fulfilment of the requirement for the award of Diploma in Computer Engineering under the Directorate of Technical Education, Kerala State during the academic year 2020- 21.

Ms Subi Abdulla P M

Project Guide

HOD Computer Engineering

Internal Examiner

External Examiner

Nedupuzha, Thrissur

Date:

ACKNOWLEDGEMENT

I would like to take this opportunity to express my gratitude to all the people who have in various ways, helped in the successful completion of my project report on the Online Grocery Shopping Web Application - Shopfresh. This work is the outcome of the guidance provided by experienced and dedicated persons of Govt. Women's Polytechnic College, Nedupuzha.

First of all, I would like to convey my hearty gratitude to my **Project Guide Ms Subi Abdulla P M**, for giving me a constant source of inspiration and helping me in the completion of my Project report, personally correcting my work and providing encouragement throughout the work. I am deeply obliged to the **Head of the Department** who guided me in the successful completion of the work.

I would like to thank the technical staff and all other faculty members the of Computer Engineering Department for their valuable and timely suggestions which led to the successful completion of the work.

I would also like to thank my parents and my classmates for their encouragement and support through every phase of this report. Above all, I thank God almighty for providing us with an opportunity to present my work to the best of my capabilities.

Thanks to all once again……

T S Parvathy

ABSTRACT

An online Grocery Store permits a customer to submit online orders for items and/or services from a store that serves online customers. The online Store system presents an online display of all items they want to sell. This web application helps customers to choose their daily needs and add products to their shopping carts. The customers can then pay for an item which was already placed in the cart. They should provide their delivery address and contact details for home delivery purposes. The items available in the online shop can be viewed separately based on their category. Specific items can also be searched from a given list of items can be searched to avoid inconvenience.

For designing an online grocery store, we create a website using PHP with MySQL. This application can add grocery items by their name, quantity, status (available, not available), and date. We can view, delete and update those items. There will be a category filtering feature where we can view the grocery items according to the dates or categories and also search for available items.

Online grocery shopping application, just like any other application has their share of pros and cons. Some of the advantages include Time-saving, travel expense saving and Price comparison. When we shop online, we can save the time which would have been spent on dressing up and moving out of our homes on our two-wheeler or four-wheeler vehicles. This way, we also save petrol expenses which are becoming a rising problem for today's generation. Another area where our application benefits are that we can compare the price ranges of different shops in the comfort of our own homes. Elder members of our society who have trouble travelling from one place to another also benefit from our application.

Some disadvantages include inaccurate delivery or delivery of damaged items, increased prices and the inability to view items personally before buying them.

But again, due to the recent spread of the pandemic, people were forced to rely on Online shopping systems rather than their offline counterparts which drastically improved their efficiency and ease of access thereby providing better online Shopping experiences for everyone.

INDEX

CHAPTER 1

INTRODUCTION

Our project titled **"ONLINE GROCERY SHOPPING WEB APPLICATION - SHOPFRESH"** is an efficient application which manages online grocery shopping. In this application, the users can buy the required grocery items from an online shop by first adding them to a cart and then buying the products one by one in either online or offline(cash on delivery) mode. The items will be delivered to the user's doorstep if they provide the correct delivery address details.

Our application opens with a welcome page which shows the overall functionalities involved in our application.

Some of the major Modules are listed below:

- **User Registration:** First-time users should enter through this option to register themselves first before Logging in to the application.
- **User Login:** Users who have already registered can directly enter through this option and enter their registered email id and password.
- **Admin Login:** This option can only be accessed by the admin as the Admin must enter the pre-set Admin's ID, Name and password.
- **About Us:** Both the Admin and the Users can click this option to know more about the Shop and the details of the project creators.
- **Contact Us:** The users can submit their reviews and queries about this application which will be visible to the Admin.
- **Gallery:** This section shows a photo gallery of our shop.

On a basic level, this project can be divided into two modules, i.e. User Module and Admin Module.

1. **User Module**: This is the interface that users see when they enter this module and this is where online shopping takes place. Upon opening this web page, the users can either select from a particular category of items or from the main Shop where all the items are visible.

   Some of the major Sub-modules under the User Module are:

   - **Shop/ Sidebar shop**: This is the area where the users can view all the available Items and select their required items by clicking on the 'Add to Cart

option. The users can then buy a single item from their 'My cart' option by clicking on the 'Buy now' option. After that, the user is directed to another page which allows them to select the number of items required (edit Quantity) and the total price is displayed according to the quantity selected. Then another window opens which allows the users to edit their Delivery/Billing Address after which the mode of Payment can be selected. Finally, the user is redirected to the final success page which displays the placed orders.

- **My Cart:** The items selected in the Shop module are added to the My Cart module. Inside this module, a search option is also provided to facilitate instant item searching. Items no longer required can be removed from the cart by clicking on the 'remove from cart' option.
- **View Placed orders:** The users can view their own placed orders which can also be cancelled at any time.

2. **Admin Module**: This is the interface that the admin can see when they enter this module.

   There are mainly three sub-modules which come under the Admin Module

   - **Manage Items:** In this module, the admin can Add, Update and Delete Items.
   - **View Placed Orders:** In this module, the Admin can view the user's booked orders.
   - **View User Reviews:** In this module, the admin can view the reviews posted by the users.

CHAPTER 2

REQUIREMENT ANALYSIS

Project development must concentrate on software engineering principles. So this project has undergone the various stages of software engineering. Thus it had gone through the requirement gathering phase. So the requirements gathered are as follows:

Requirement analysis:

After gathering all requirements, the main purpose of requirement analysis is clearly to understand the user requirements and detect inconsistencies and incompleteness. There are three main types of problems in the requirements that the analyst need to identify and resolve, which are: ambiguity, inconsistency, and incompleteness.

After all, these SRS documents are prepared. SRS is a software requirement specification document. It is useful in various contexts. They are statements of user needs, contract documents, reference documents and definitions for implementation.

Functional requirements:

ADMIN

1. **Introduction**

   The Admin can enter the Admin module through the Admin Login Link. The Admin then enters the Admin Login page where they have to enter a pre-defined ID, Admin name and Password. After this step, the login page is redirected to the Admin Home page. There are three actions that the Admin can perform i.e. 1. Manage Items, 2. View Placed orders and 3. View User Reviews.

2. **Manage Items**

   In this module, the Admin can add new items, Update existing Items and also delete an existing item.

3. **View Placed Orders**

   In this module, the Admin can view the orders placed by the logged-in users.

4. **View User Reviews**

   In this module, The Admin can view the reviews posted by the Users.

USER

1. **Introduction**

   The users can enter the web application in two ways i.e. User Registration and User Login. New Users may register first before login in. Already registered users may enter their Email ID and Password to enter the User's Home page.

2. **Shop**

   This is the area where the users can choose their required items from a list of available items and these selected items will be automatically added to the cart.

3. **My Cart**

   This is the module where the items selected on the Shop page are added. Items can be bought from this module**.**

4. **Placed Orders**

   After successfully booking an order, the user can view their Placed orders in this module. They can also cancel the booked order anytime.

5. **User Reviews**

   Users can also post reviews about the shop and/or its services through this module which can be viewed by the admin.

Non-Functional Requirements:

It includes:

- Reliability issues
- Performance issues
- Human-computer interface issues
- Interface with other external systems, security, maintainability, etc. After requirement analysis, the SRS document is prepared.

SRS DOCUMENTATION

A **software requirements specification (SRS)** is a description of a software system to be developed. It is modelled after **business requirements specification (CONOPS)**, also known as a stakeholder requirements specification (STRS). The software requirements specification lays out functional and non-functional requirements, and it may include a set of use cases that describe user interactions that the software must provide to the user for perfect interaction.

Software requirements specification establishes the basis for an agreement between customers and contractors or suppliers on how the software product should function (in a market-driven project, these roles may be played by the marketing and development divisions). Software requirements specification is a rigorous assessment of requirements before the more specific system design stages, and its goal is to reduce later redesign. It should also provide a realistic basis for estimating product costs, risks, and schedules. Used appropriately, software requirements specifications can help prevent software project failure.

The software requirements specification document lists sufficient requirements for the project development. To derive the requirements, the developer needs to have a clear and thorough understanding of the products under development. This is achieved through detailed and continuous communications with the project team and customer throughout the software development process.

1. **Introduction**

   The project entitled "ONLINE GROCERY SHOPPING WEB APPLICATION - SHOPFRESH" objective is to provide a system that manages the Buying and selling of grocery items online. There are mainly two types of users who access this system that is., Admin and the Customers. Both types of users consume less amount of time when they work through an automated system. The system will take care of all the processes in a quick manner. Data storing is easier too.

   a. **Purpose**

      The main purpose of this project is to buy/ sell Grocery Items online where the items can be ordered online with online/offline payment.

   b. **Intended Audience**

      The project is mainly used by Customers and Admin. Customers use this site to buy grocery items online while Admin uses this site to add new items, view user orders and view user reviews.

c. Project scope

To buy grocery items online. Make sure that the web application is simple and easy to use.

2. Overall Description

The Web Application "SHOPFRESH" is based on ordering grocery items online which is managed by an Admin. As soon as a customer registers and logs in to this website they can order grocery items online which will be delivered to their doorstep. Both online and offline payment modes are available. All data will be stored in the database.

a. Product perspective

To ensure a hassle-free online grocery shopping experience by providing reliable and true item details.

b. Product features

- Login features

  Customers should enter valid usernames and passwords to access their profiles.

c. Design and Implementation Constraints

The insert, Update and Delete operations are available. SQL command for the above queries has been successfully executed. Implement the database using the centralized database management system.

d. Software and Hardware Requirements

- Software requirements
  - Front - end: PHP
  - Back – end: MySQL
- Hardware requirements
  - Windows 7,8,10
  - Core i3 and above
  - 2GB RAM ad above

e. User Documentation

User document gives customers the information they need to use the product. They are primarily teaching materials that include some technical explanation.

3. System features

a. Module Description (Admin and User) "ONLINE GROCERY SHOPPING WEB APPLICATION – SHOPFRESH" is an efficient application for carrying out online shopping. There are two types of users, i.e., ADMIN and USER(customers). ADMIN can add new items, Update existing items, delete existing items, view booked user orders and view user reviews.

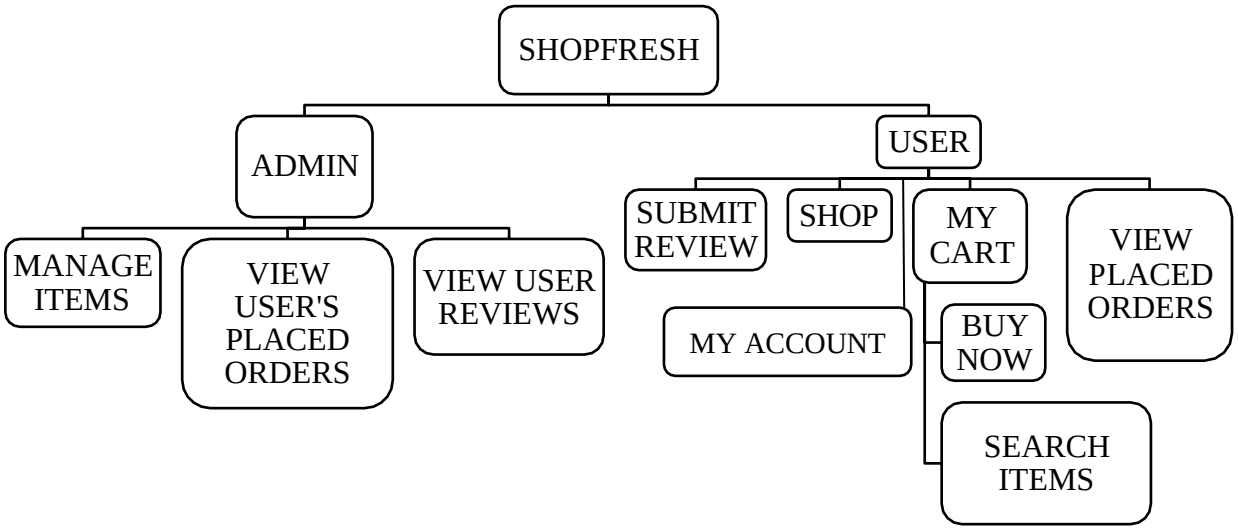
CHAPTER – 3

SOFTWARE DESIGN

After requirement analysis, the next phase undergone was software designing. Designing activities are done through the following types of graphical methods.

Structure Chart

A Structure Chart (SC) in software engineering and organizational theory is a chart that shows the breakdown of a system to its lowest manageable levels. They are used in structured programming to arrange program modules into a tree. Each module is represented by a box, which contains the module's name.

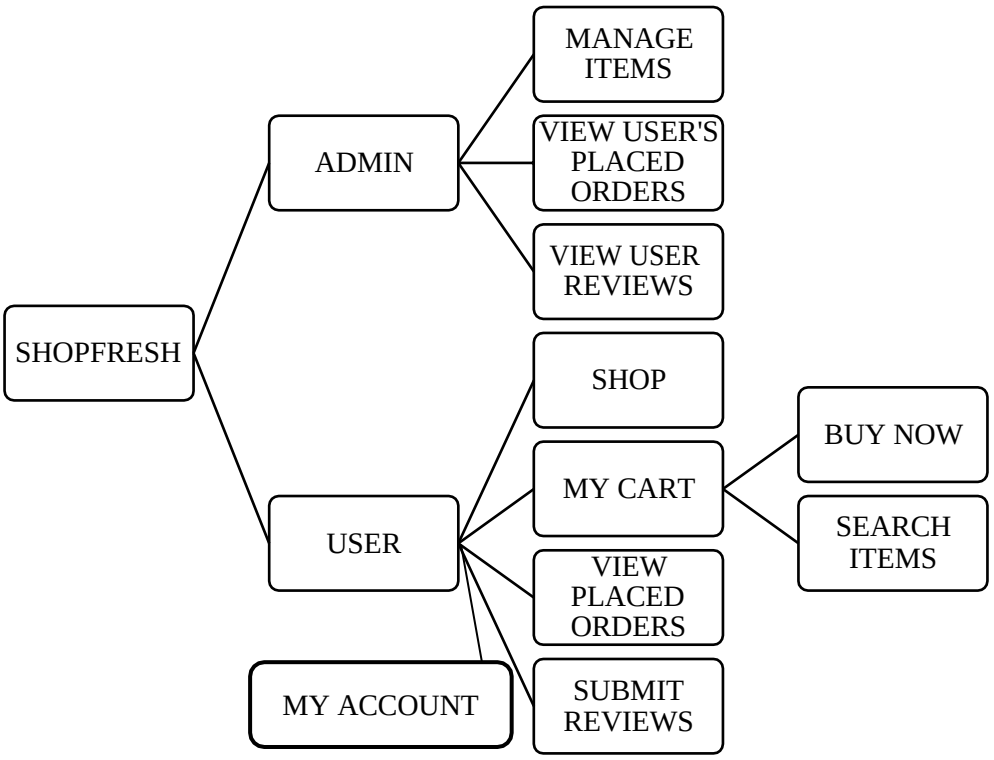


UML Diagram

A UML diagram is a diagram based on the UML (Unified Modeling Language) to visually represent a system along with its main actors, roles, actions, artefacts or classes, to better understand, alter, maintain, or document information about the system.

a. Use Case Diagram

A use case diagram is a dynamic or behaviour diagram in UML. Use case diagrams to model the functionality of a system using actors and use cases. Use cases are a set of actions, services, and functions that the system needs to perform.

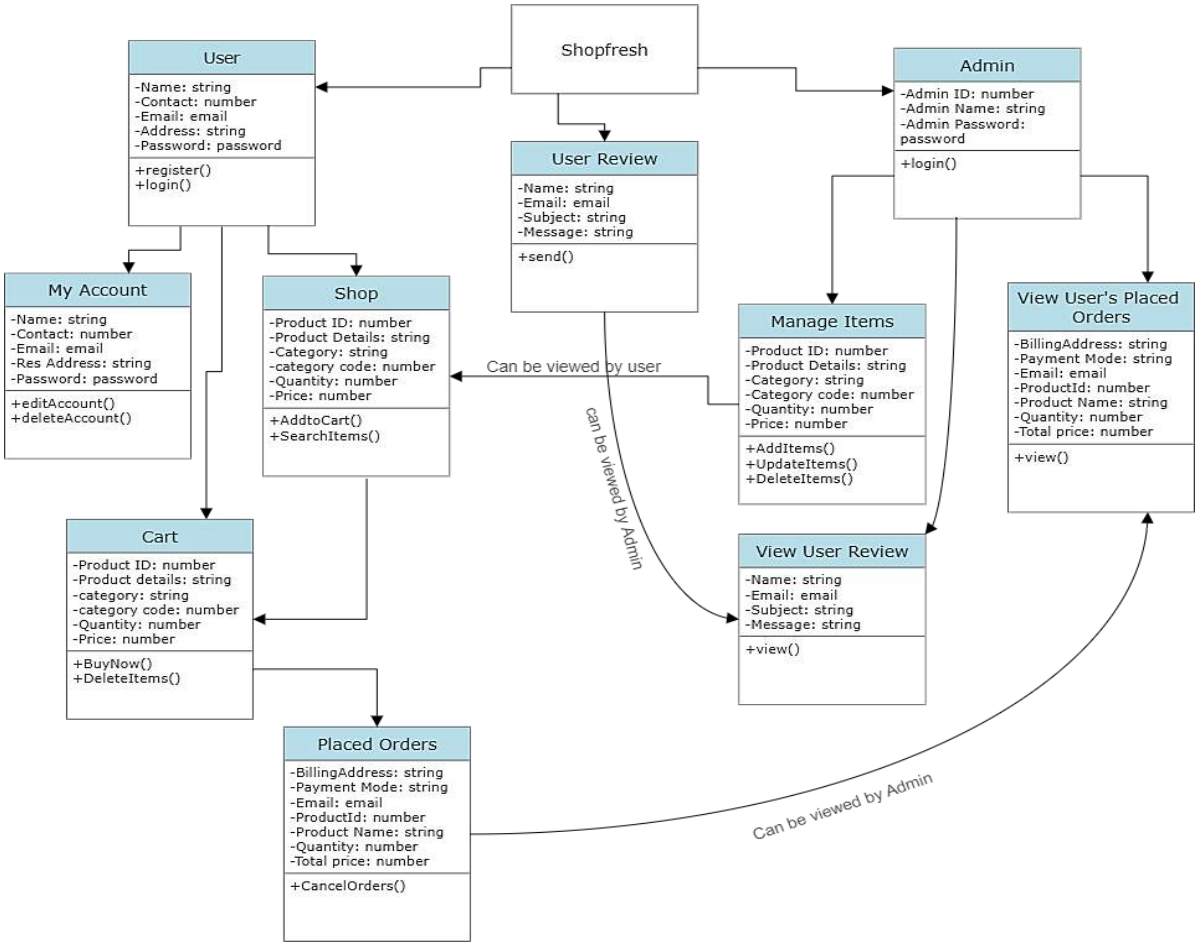


b. Class Diagram

The class diagram is static. It represents the static view of an application. The class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

A class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modelling of object-oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.

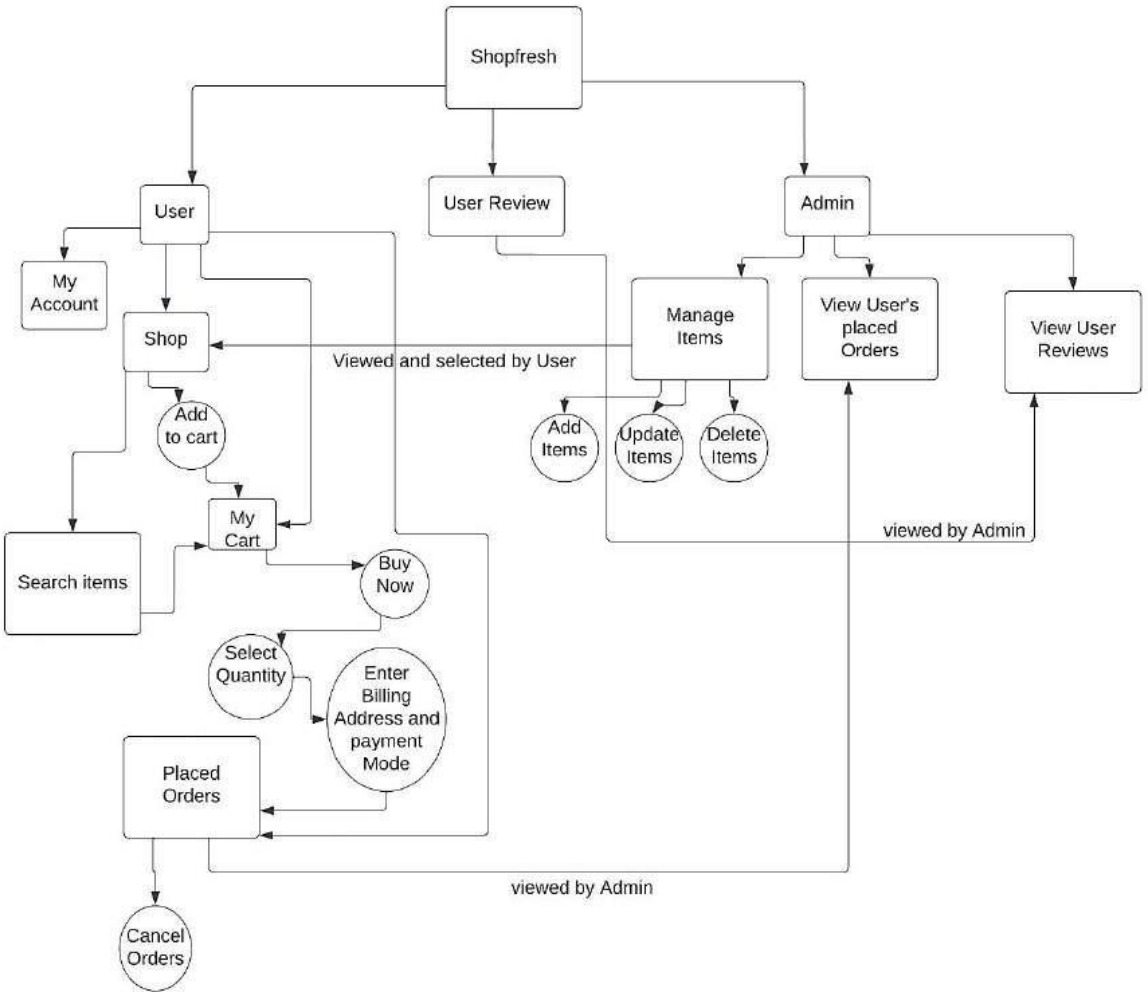
The class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.



c. DFD Diagram

Also known as DFD, Data flow diagrams are used to graphically represent the flow of data in a business information system. DFD describes the processes that are involved in a system to transfer data from the input to the file storage and reports generation.

Data flow diagrams can be divided into logical and physical. The logical data flow diagram describes the flow of data through a system to perform certain functions of a business. The physical data flow diagram describes the implementation of the logical data flow.


Shopfresh
User
User Review
Admin
My Account
Shop
Manage Items
View User's placed Orders
View User Reviews
Viewed and selected by User
Add to cart
Add Items
Update Items
Delete Items
My Cart
viewed by Admin
Search items
Buy Now
Select Quantity
Enter Billing Address and payment Mode
Placed Orders
viewed by Admin
Cancel Orders

CHAPTER – 4

IMPLEMENTATION

Front - End

The front end used for this project is PHP(PHP Hypertext preprocessor). PHP is a widely-used, open-source scripting language which is executed on a server. It is free and easy to download.

PHP Supported Technologies

- PHPStorm
- Zend Studio
- NuSphere PhpED
- Netbeans
- Cloud 9
- Aptana Studio
- CodeLobster
- Sublime Text
- PHPDebugbar
- Eclipse
- Code Lobster
- Selenium Test
- Scriptcase

New features in version 7.4

- Arrow functions' support
- Typed properties' support
- Preloading
- Covariant returns & contravariant parameters
- Weak References
- Coalescing assign operator
- A spread operator in array expression
- A new custom object serialization mechanism
- Reflection for references
- Support for throwing exceptions from__toString()

Hardware and Software required for running NetBeans 8.2

PHP requires at least Windows 2008/Vista. Either 32-Bit or 64-bit (AKA X86 or X64. PHP does not run on Windows RT/WOA/ARM). As of PHP 7.2.0, Windows 2008 and Vista are no longer supported.

PHP requires the Visual C runtime (CRT). Many applications require that so it may already be installed.

Supporting Operating Systems

- Linux
- MAC OSX
- WINDOWS

Hardware Configurations

Requirement	**Minimum**	**Recommended**
PHP Version	5.6.0	*Latest* 5.6 or 7.0 Release
PHP Memory Limit	64MB	128MB**
PHP Database Extension	PDO	PDO
PHP Extensions	Curl with SSL GD2 Image Library JSON Support XML	Curl with SSL GD2 Image Library JSON Support **MBString** **Iconv**
MySQL Version	5.2.0	5.5.x
Ioncube Loaders	5.0.21 or later for PHP 5.6 6.0.2 or later for PHP 7.0	The latest 5.x version for PHP 5.6 The latest 6.x version for PHP 7

PHP is a fast and feature-rich open-source scripting language used to develop Web Applications or Internet / Intranet Applications.

MySQL is a powerful open-source database server built based on a relational database management system (RDBMS) and is capable of handling a large concurrent database

connection.

When combined, talented PHP and MySQL developers can build very powerful and scalable Web / Internet / Intranet Applications.
PHP and MySQL are referred to as development tools.

PHP and MySQL are Open Sources, meaning that they are free development tools, and there is a large community of dedicated volunteer programmers who contribute to making improvements and are continuously adding features to it. The development tools and database servers that require licensing costs have limited programming resources compared to open source development tools, which have an enormous and fast-growing dedicated and knowledgeable community that extends around the world.

There has been disagreement about which tool is better. Naturally, the developer who is more familiar with one tool over the other will stand behind the tool that he or she has experience with.

With our experience, we have found that PHP and MySQL are the best development tools. When developed correctly, applications can be built with clean and simple usability, complex functionality, speed, power and scalability.

Back – End

The back end used for the project is MYSQL. MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by Oracle Corporation.

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one-to-one, one-to-many, unique, required or optional, and "pointers" between different tables. The database enforces these rules, so that with a well-designed database, your application never sees inconsistent, duplicate, orphan, out-of-date, or missing data.

The SQL part of "MySQL" stands for "Structured Query Language". SQL is the most common standardized language used to access databases. Depending on your programming environment, you might enter SQL directly (for example, to generate reports), embed SQL statements into code written in another language, or use a language-specific API that hides the SQL syntax.

SQL is defined by the ANSI/ISO SQL Standard. The SQL standard has been evolving since 1986 and several versions exist. In this manual, "SQL-92" refers to the standard released in 1992, "SQL:1999" refers to the standard released in 1999, and "SQL:2003" refers to the current version of the standard. We use the phrase "the SQL standard" to mean

the current version of the SQL Standard at any time.

MySQL Server can run comfortably on a desktop or laptop, alongside your other applications, web servers, and so on, requiring little or no attention. If you dedicate an entire machine to MySQL, you can adjust the settings to take advantage of all the memory, CPU power, and I/O capacity available. MySQL can also scale up to clusters of machines, networked together.

MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Although under constant development, MySQL Server today offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL The server highly suited for accessing databases on the Internet.

SQL supports many data types: signed/unsigned integers 1, 2, 3, 4, and 8 bytes long, FLOAT, DOUBLE, CHAR, VARCHAR, BINARY, VARBINARY, TEXT, BLOB, DATE, TIME, DATETIME, TIMESTAMP, YEAR, SET, ENUM, and OpenGIS spatial types.

Clients can connect to MySQL Server using several protocols. It has a privilege and password system that is very flexible and secure, and that enables host-based verification. Password security by encryption of all password traffic when you connect to a server is also available.

CHAPTER – 5

RESULT

The project was entitled “Online Grocery Shopping Web Application - Shopfresh” after going through a series of phases like requirement analysis, design phase, coding and testing. The various limitations of the current system were diagnosed and are improved in this work. After considering various feasible solutions, the most feasible one was selected for design by considering time and efficiency as constraints. The performance of the system was evaluated to determine whether the system achieved the results that were expected and whether the predicted benefits of the system were being realized. The results are obtained in a timely and constrained manner.

Since each process is implemented as simple modules, the system is liable to further modification and also provides easy maintenance. The project contains a lot of improvement in it. But the overall look and feel of the project give a rough picture of an automated version of existing systems. The performance of the system was evaluated to determine whether the system achieved the results that were expected and whether the predicted benefits of the system were being realized.

CHAPTER – 6

CONCLUSION

The Project titled “Online grocery Shopping Application - Shopfresh” was completed successfully. Each module in this project was successfully run. The project entitled “Admission management for College” was made operational after going through a series of phases like requirement analysis, design phase, coding and testing. The various limitations of the current system were diagnosed and are improved in this project. After considering various feasible solutions, the most feasible one was selected for design by considering time and efficiency as constraints. The performance of the system was evaluated to determine whether the system achieved the results that were expected and whether the predicted benefits of the system were realized.

The results are obtained in a timely and constrained manner. Since each process is implemented as simple modules, the system is liable to further modification and also provides easy maintenance. The project contains a lot of scope for improvement in it. But the overall look and feel of the project give a rough picture of an automated version of the existing system. The screenshots of basic frames are included in the appendix.

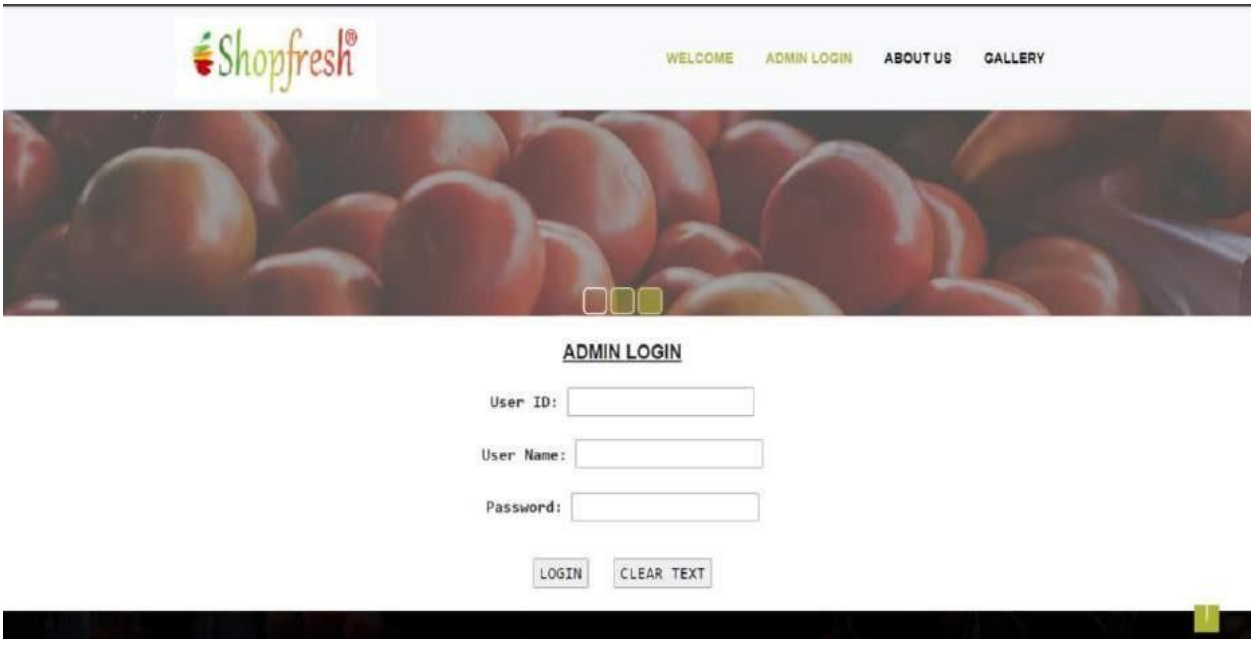
APPENDIX

1. SCREENSHOTSOTS

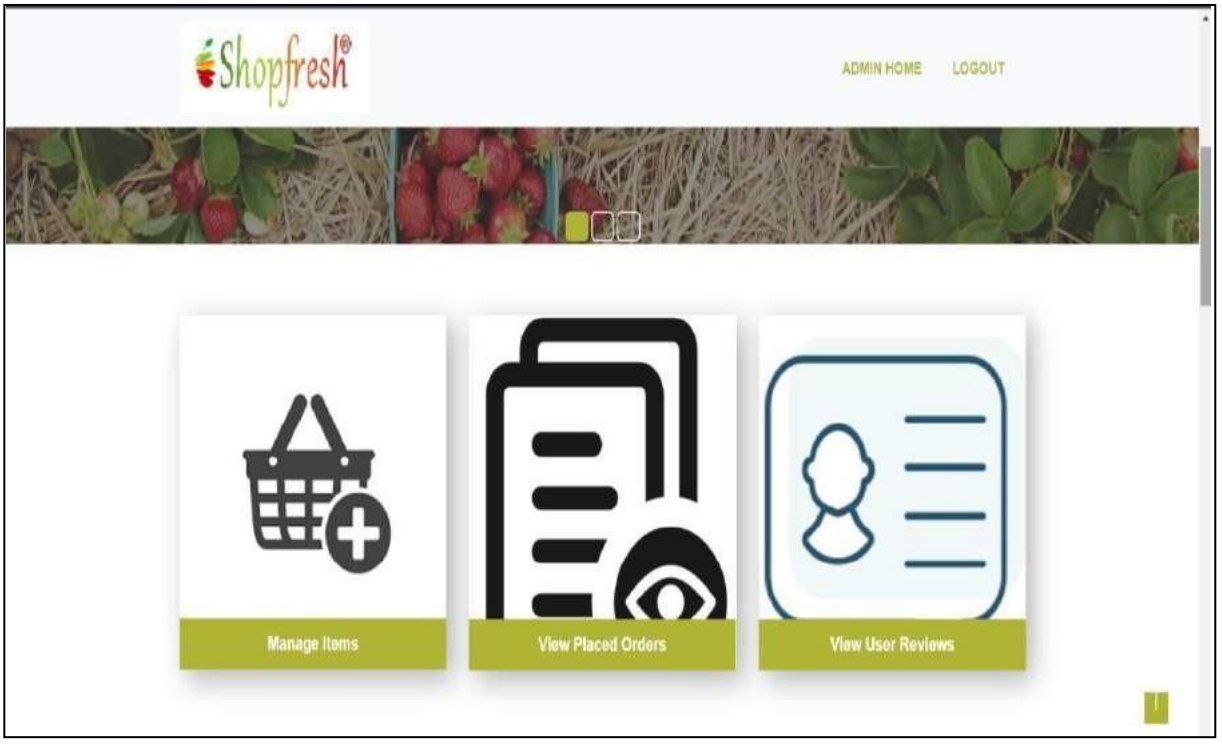
1.1 Home Page

1.2 Admin

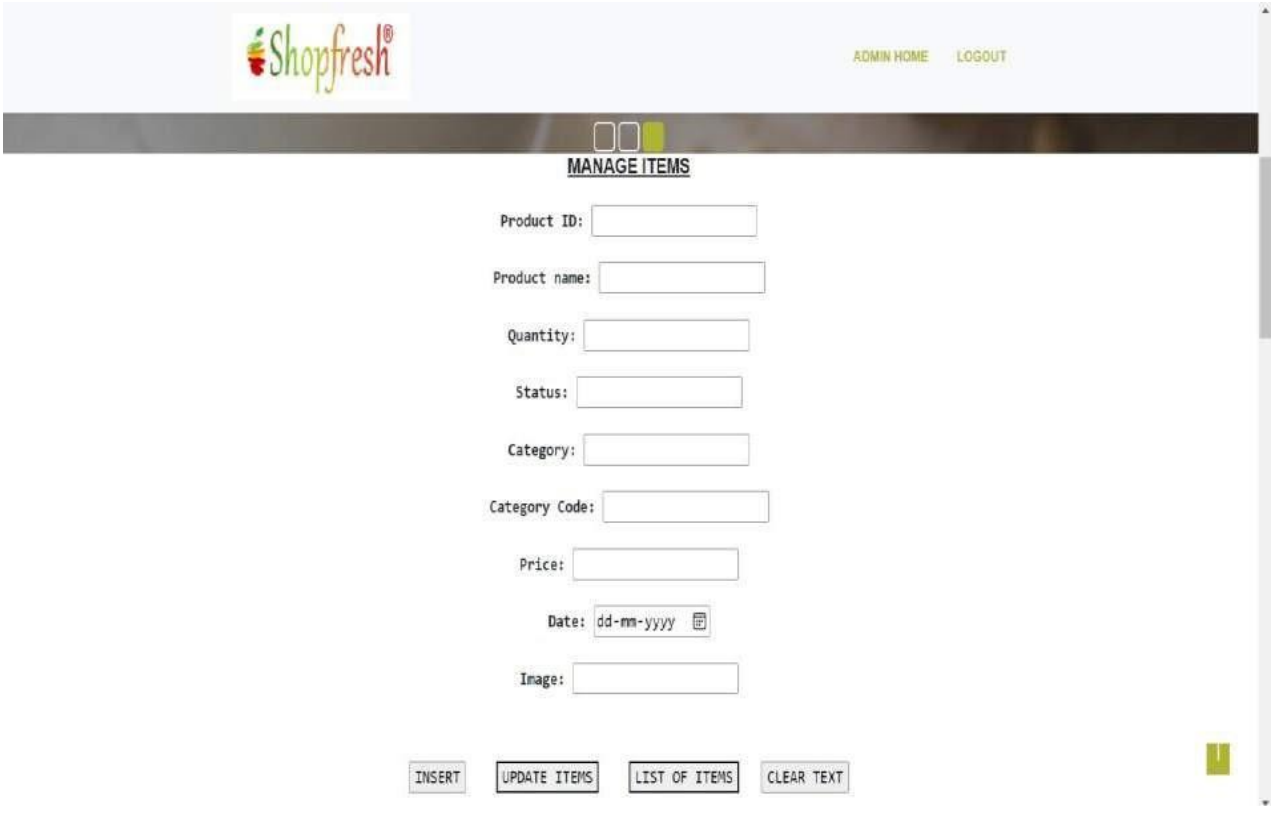
1.2.1 Admin Login



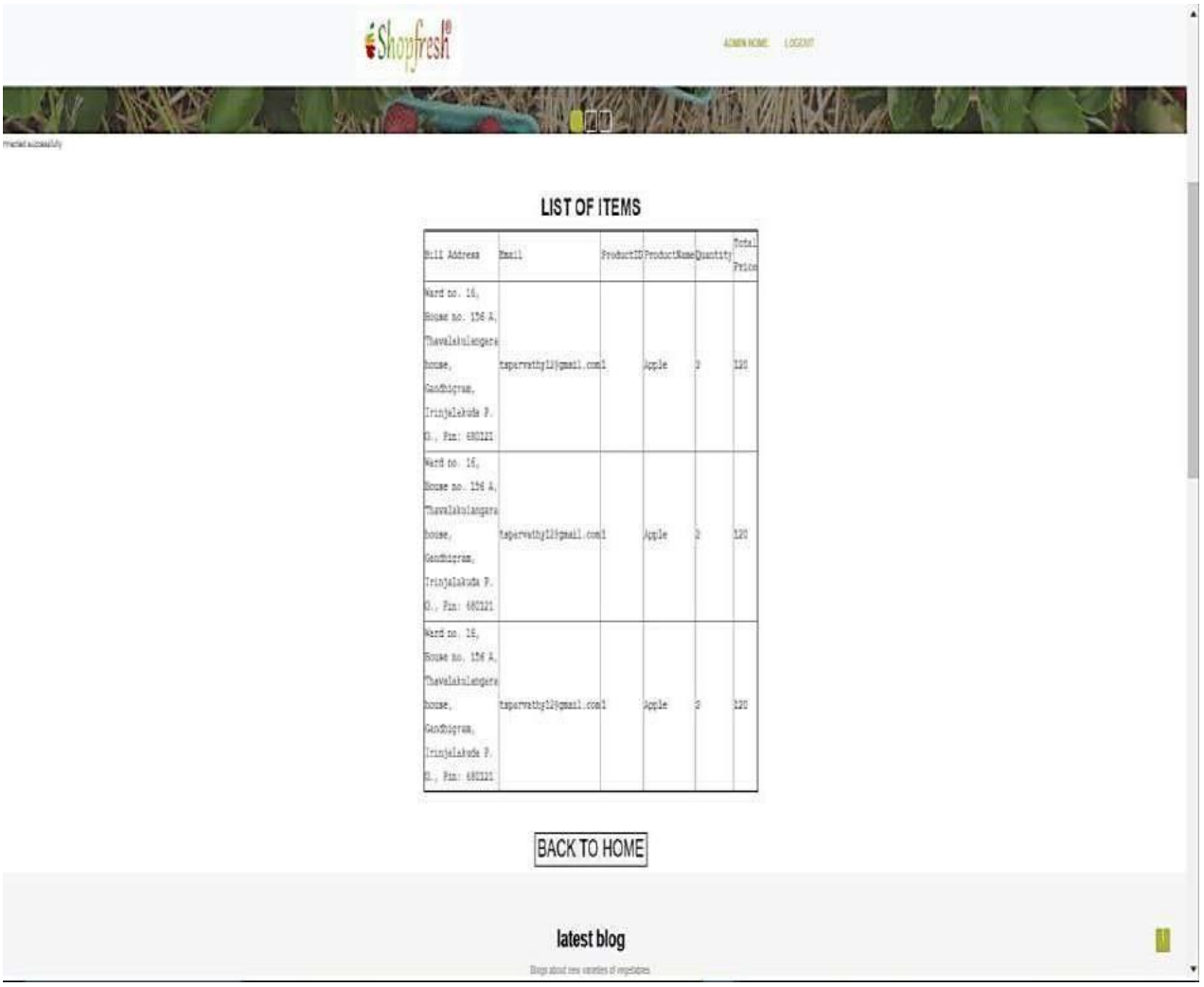
1.2.2 Admin Home Page



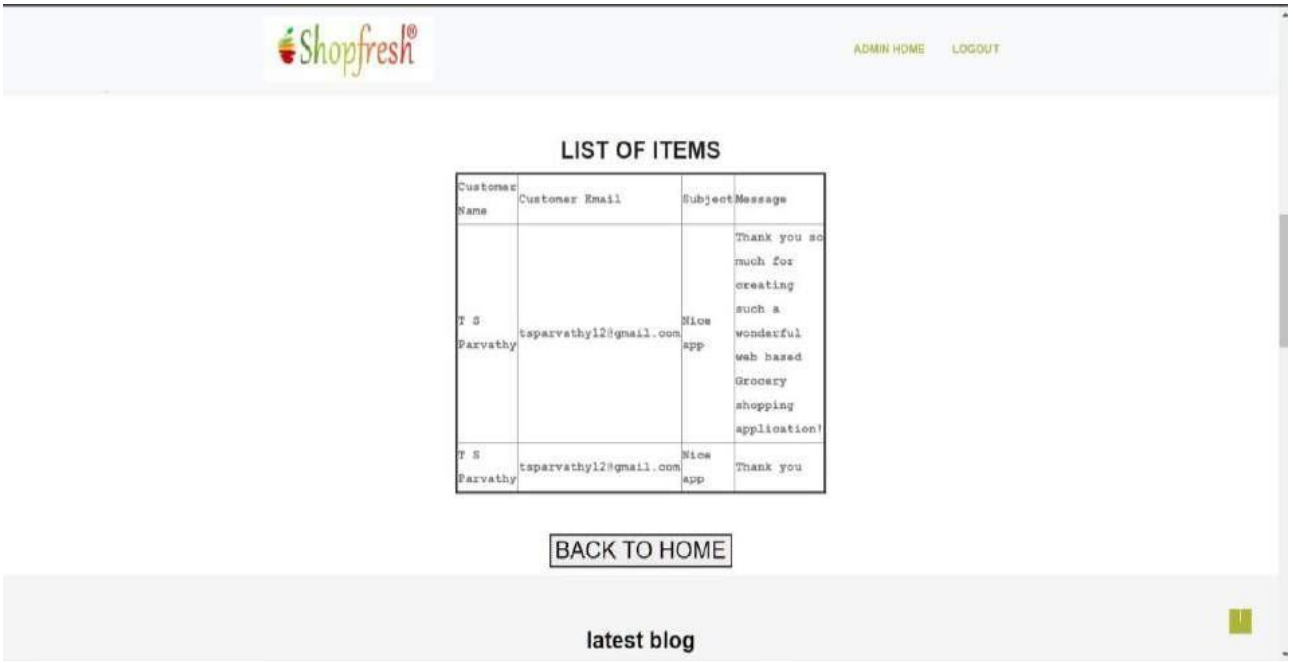
1.. 1.2.3 Admin – Manage Items



2.. 1.2.4 Admin – View User's placed orders

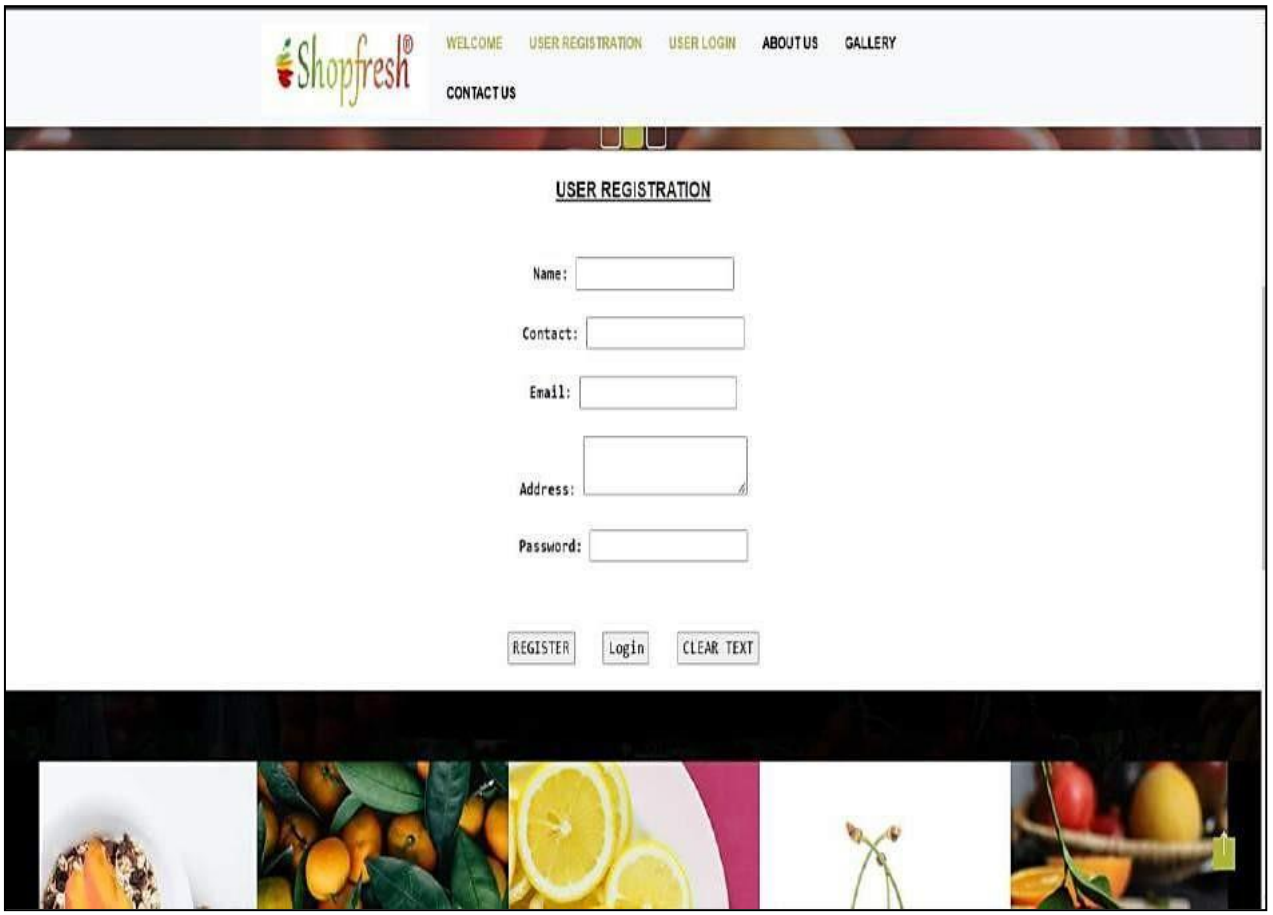


3.. 1.2.5 Admin – View User Reviews

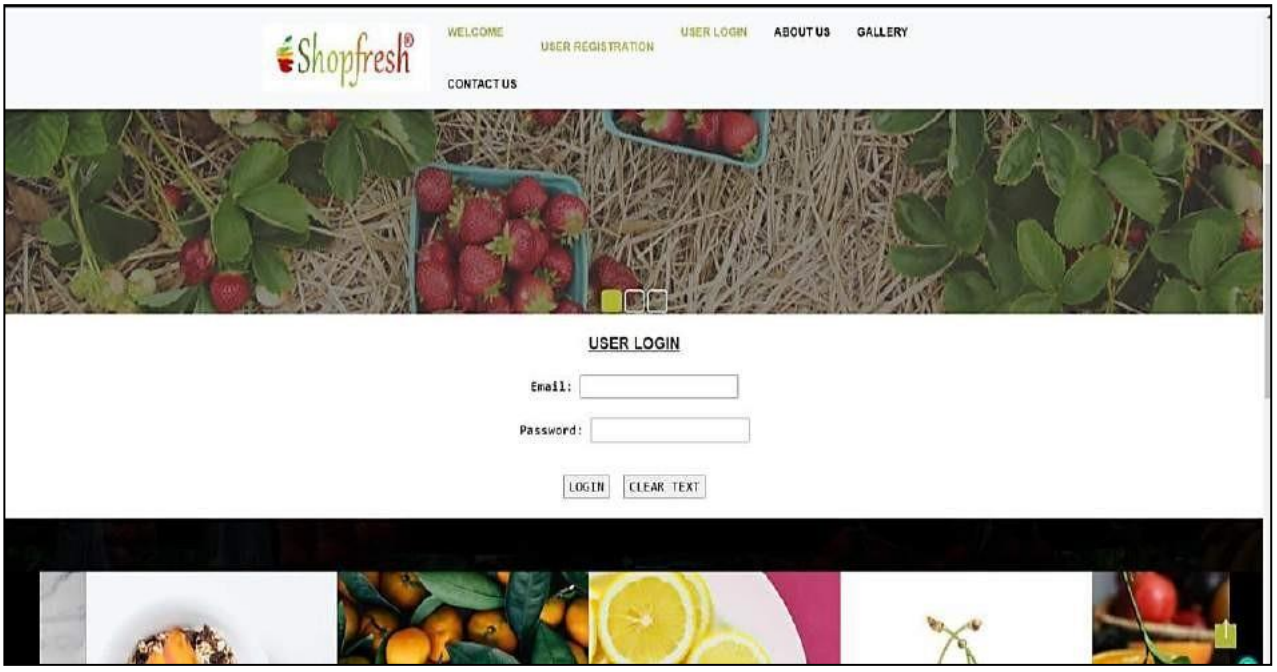


1. 1.3 User

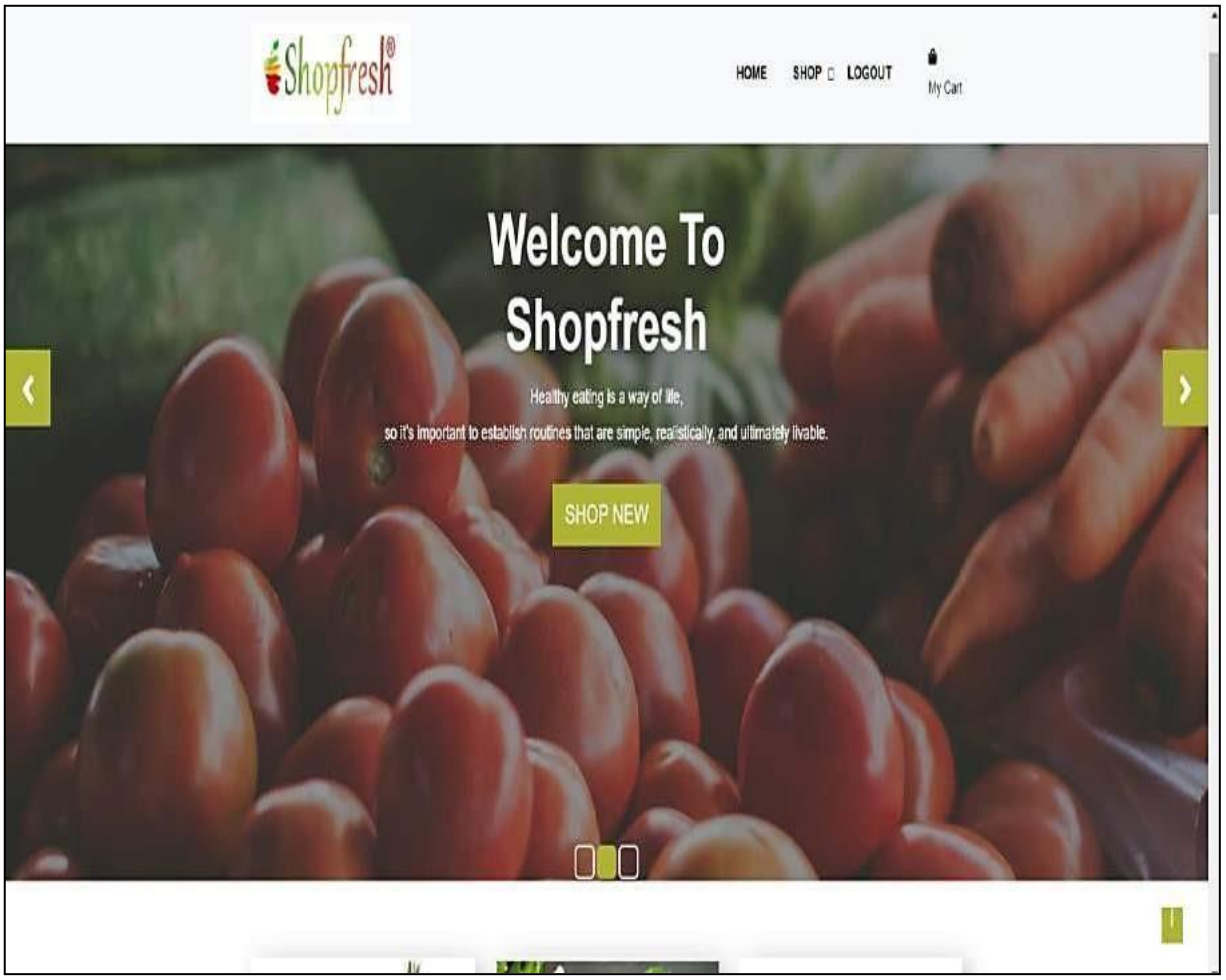
1.3.1 User Registration



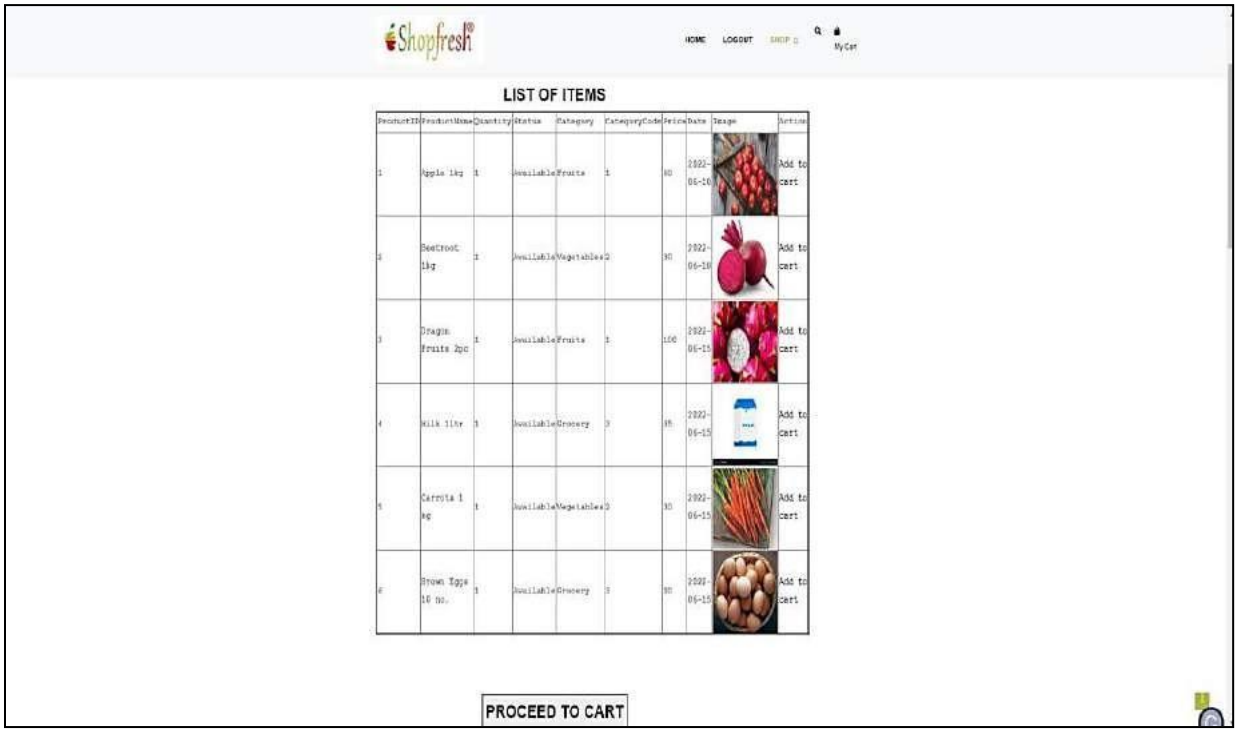
1.. 1.3.2 User Login



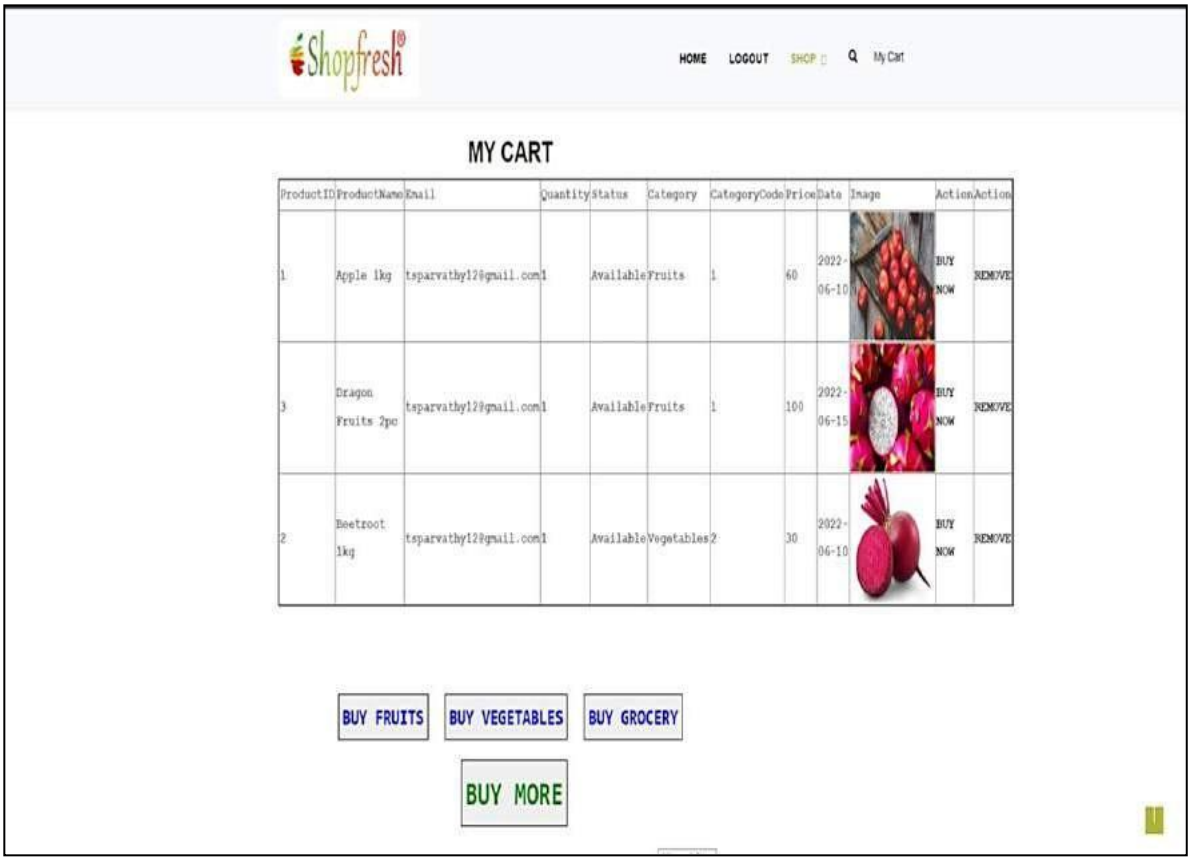
1.. 1.3.3 User Homepage



1.. 1.3.4 Shop

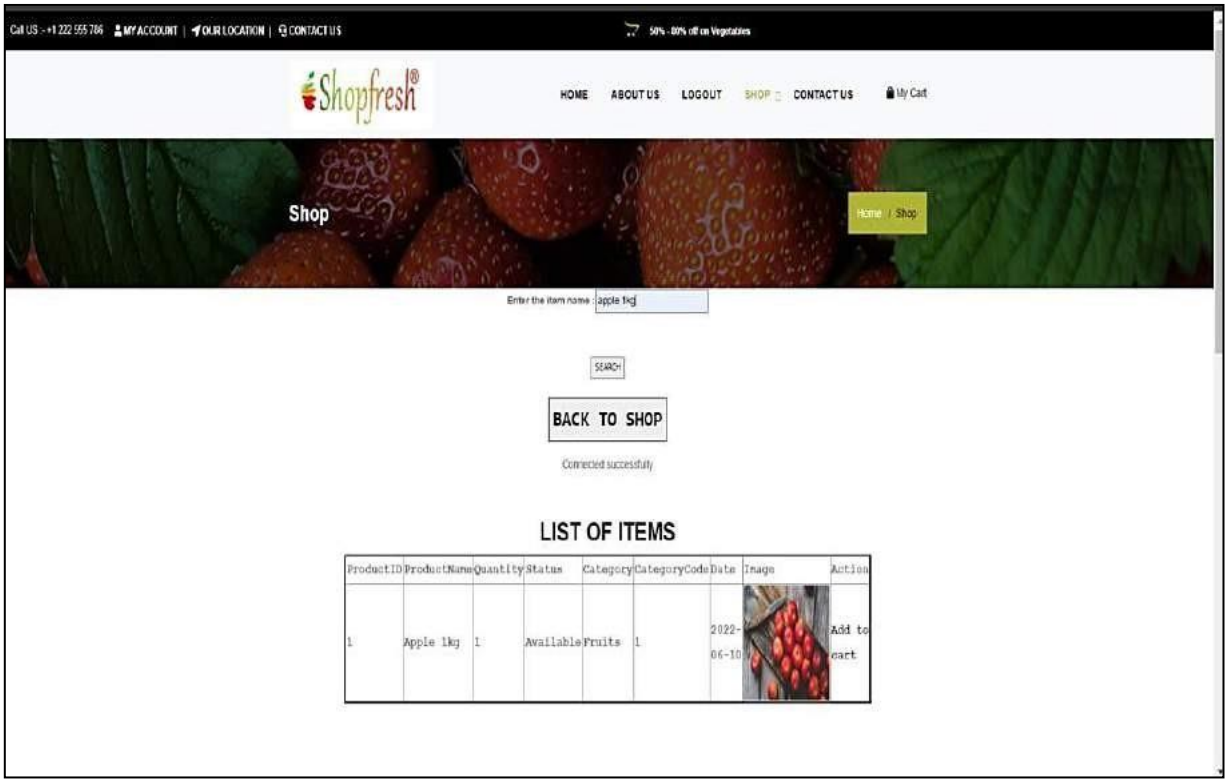


1.. 1.3.5 My Cart

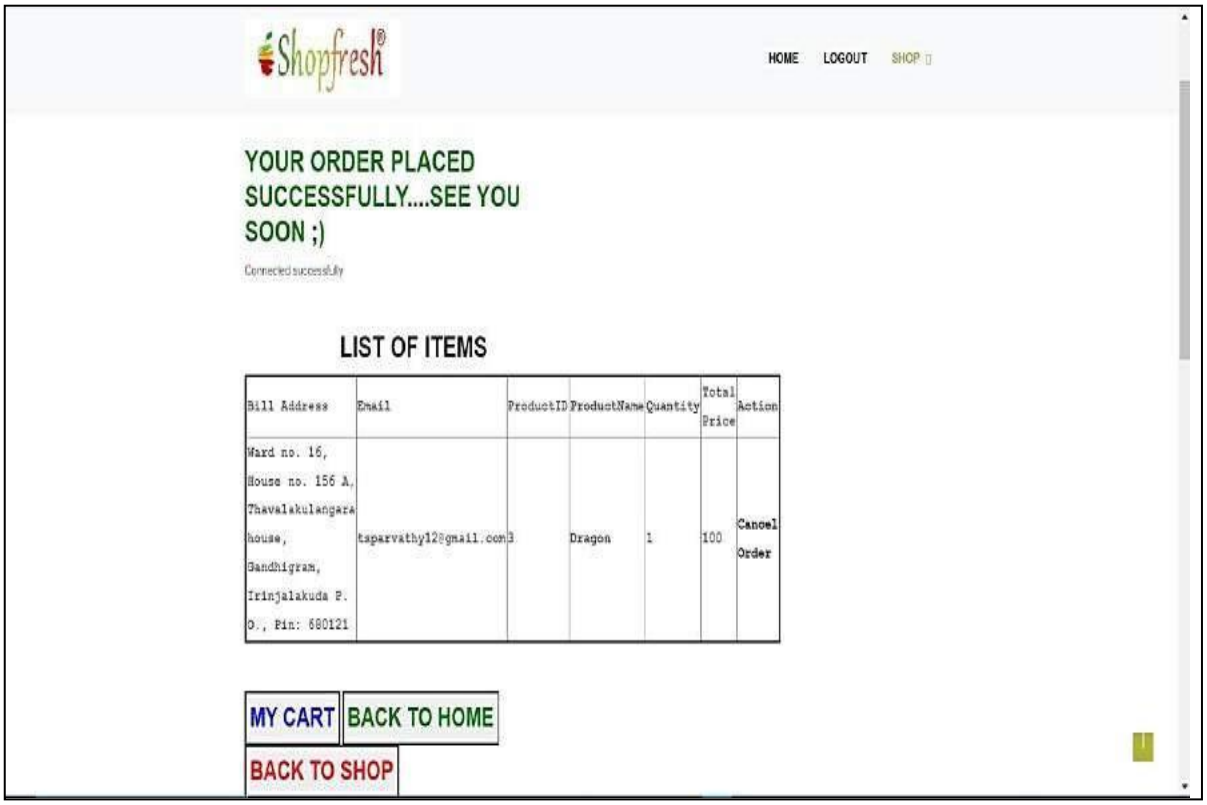


2..

1.. 1.3.6 Search Items



1.3.7 View Placed Order



2. CODE

2.1 Database Connection

```
<?php
  $link=mysqli_connect("localhost","root","","grocery");
  if(mysqli_connect_error())
  {
    echo "failed to connect MySQL: ".mysqli_connect_error();
    exit();
  }
  else
  {
    echo "Connected successfully";
  }
?>
```

2.2

2.3 User - Shop

```
<!-- Display Table -->

     <?php

include 'connect.php';

$query="select * from manageitems";
$res=mysqli_query($link,$query);

?>
<br>
<br>
<br>
<h2 align="center" style="font-weight:bold; font-size:220%">LIST OF ITEMS</h2>
<table border="3" width="22%" align="center" style="font-family:Courier New, Courier,
monospace; font-size:120%; font-weight:bold;">
<tr style="font-weight:bold;">
```

```
<td>ProductID</td>
<td>ProductName</td>
<td>Quantity</td>
<td>Status</td>
<td>Category</td>
<td>CategoryCode</td>
<td>Price</td>
<td>Date</td>
<td>Image</td>

<td>Action</td>
</tr>

<?php
while($row=mysqli_fetch_array($res))
{

  $pdid=$row['ProductID'];
  $pdn=$row['ProductName'];
  $quantity=$row['Quantity'];
  $stat=$row['Status'];
  $cat=$row['Category'];
  $cc=$row['CategoryCode'];
  $pri=$row['Price'];
  $d=$row['Date'];
  $img=$row['Image'];

$path="uploads/".$img;

?>
<tr>
<td><?php echo $pdid; ?></td>
<td><?php echo $pdn; ?></td>
```

```
<td><?php echo $quantity; ?></td>
<td><?php echo $stat; ?></td>
<td><?php echo $cat; ?></td>
<td><?php echo $cc; ?></td>
<td><?php echo $pri; ?></td>
<td><?php echo $d; ?></td>
<td><img src="<?php echo $path; ?>" height="150" width="150"></td>

<td><a href="add.php?ProductID=<?php echo $pdid;?>&&ProductName=<?php echo
$pdn; ?>&&Email=<?php echo $logged; ?>&&Quantity=<?php echo
$quantity;?>&&Status=<?php echo $stat;?>&&Category=<?php echo
$cat;?>&&CategoryCode=<?php echo $cc;?>&&Price=<?php echo
$pri;?>&&Date=<?php echo $d;?>&&Image=<?php echo $img;?>">Add to
cart</a></td>
</tr>
<?php
}
?>
</table>

<br><br>
<!-- close display table -->
<br>
<br>
```

2.4 User – My Cart

```
<!-- Display Table -->

    <?php

include 'connect.php';

$query="select * from cart where Email='$logged'";
```

```
$res=mysqli_query($link,$query);

?>



<br>
<br>
<br>
<h2 align="center" style="font-weight:bold; font-size:220%">MY CART</h2>
<table border="3" width="22%" align="center" style="font-family:Courier New, Courier,
monospace; font-size:120%; font-weight:bold;">
<tr style="font-weight:bold;">
<td>ProductID</td>
<td>ProductName</td>
<td>Email</td>
<td>Quantity</td>
<td>Status</td>
<td>Category</td>
<td>CategoryCode</td>
<td>Price</td>
<td>Date</td>
<td>Image</td>

<td>Action</td>
<td>Action</td>
</tr>

<?php
while($row=mysqli_fetch_array($res))
{

  $pdid=$row['ProductID'];
  $pdn=$row['ProductName'];
```

```
    $email=$row['Email'];
  $quantity=$row['Quantity'];
  $stat=$row['Status'];
  $cat=$row['Category'];
  $cc=$row['CategoryCode'];
  $pri=$row['Price'];
  $d=$row['Date'];
  $img=$row['Image'];

$path="uploads/".$img;

?>
<tr>
<td><?php echo $pdid; ?></td>
<td><?php echo $pdn; ?></td>
<td><?php echo $email; ?></td>
<td><?php echo $quantity; ?></td>

<!--<td><?php echo $quantity; ?></td>-->
<td><?php echo $stat; ?></td>
<td><?php echo $cat; ?></td>
<td><?php echo $cc; ?></td>
<td><?php echo $pri; ?></td>
<td><?php echo $d; ?></td>
<td><img src="<?php echo $path; ?>" height="150" width="150"></td>
</form>
<!--<?php $qty=$_GET['q']; ?>-->
<td><a href="confirmbooking.php?ProductID=<?php echo
$pdid;?>&&ProductName=<?php $pdn; ?>&&Email=<?php echo $email;
?>&&Quantity=<?php echo $quantity; ?>&&Price=<?php echo $pri; ?>">BUY
NOW</a></td>

<td><a href="removecart.php?ProductID=<?php echo $pdid;?>">REMOVE</a></td>
```

```
</tr> <!--<input type="submit" value="BUY NOW"/>-->
<?php
}
?>
</table><br><br>
<pre>
<center>

<button onclick="window.location.href = 'fruits.php';"><b><align="center" style="font-
weight:bold; font-size:200%; color:blue">BUY FRUITS</b></button>    <button
onclick="window.location.href = 'veg.php';"><b><align="center" style="font-
weight:bold; font-size:200%; color:blue">BUY VEGETABLES</b></button>    <button
onclick="window.location.href = 'grocery.php';"><b><align="center" style="font-
weight:bold; font-size:200%; color:blue">BUY GROCERY</b></button>

   <button onclick="window.location.href = 'shop.php';"><b><align="center"
style="font-weight:bold; font-size:300%; color:green">BUY MORE</b></button>
```

2.5 Admin – Manage Items

2.5.1 Insert Items

```
<?php
include 'connect.php';
  $pdid=$_POST['pid'];
  $pdn=$_POST['pname'];
  $quantity=$_POST['qty'];
  $stat=$_POST['status'];
  $cat=$_POST['category'];
  $cc=$_POST['catcode'];
     $pri=$_POST['price'];
  $d=$_POST['date'];
  $img=$_POST['image'];

  $query="insert into
```

```
manageitems(ProductID,ProductName,Quantity,Status,Category,CategoryCode,Price,Dat
e,Image) values('$pdid','$pdn','$quantity','$stat','$cat','$cc','$pri','$d','$img')";
 echo $query;
  $res=mysqli_query($link,$query);

  if($res)
  {
    //echo "Data Inserted Successfully";
       header('location:items.php');
  }
  else
  {
?>
    <script language="javascript">alert("Data not inserted")</script>
<?php
  }
?>
```

2.5.2 Update items

```
<?php
 include 'connect.php';
$pdid=$_GET['ProductID'];
$query="select * from manageitems where ProductID='$pdid'";
$res=mysqli_query($link,$query);
$row=mysqli_fetch_assoc($res);
$name=$row['ProductName'];
$quan=$row['Quantity'];
$stat=$row['Status'];
$cat=$row['Category'];
$cc=$row['CategoryCode'];
$pri=$row['Price'];
$date=$row['Date'];
$img=$row['Image'];
```

```
//echo $name;
?>
<!-- Register -->
                    <h2 align="center"><u><b>MANAGE ITEMS</b></u><br><br>
                    <form action="" method="post">
<pre>
<strong>Product ID: <input type="number" id="p" name="pid" required value=<?php
echo $pdid; ?>><br>
Product name: <input type="text" id="n" name="pname" required value=<?php echo
$name; ?>><br>
Quantity: <input type="number" id="q" name="qty" required value=<?php echo $quan;
?>><br>
Status: <input type="text" id="s" name="status" required value=<?php echo $stat;
?>><br>
Category: <input type="text" id="c" name="category" required value=<?php echo $cat;
?>><br>
Category Code: <input type="number" id="cc" name="catcode" required value=<?php
echo $cc; ?>><br>
Price: <input type="number" id="pr" name="price" required value=<?php echo $pri;
?>><br>
Date: <input type="date" id="d" name="date" required value=<?php echo $date; ?>><br>
Image: <input type="text" id="i" name="image" required value=<?php echo $img;
?>><br>
</strong>
</pre>
                        <pre><input type="submit" value="UPDATE"/> <button
onclick="window.location.href = 'items.php';">LIST OF ITEMS</button> <input
type="reset" value="CLEAR TEXT"></pre>
                    </form></h2>
            <!-- Close Register -->


<?php
if($_POST)
```

```
{
    include 'connect.php';

    $pdid=$_POST['pid'];
    $name=$_POST['pname'];
    $quan=$_POST['qty'];
    $stat=$_POST['status'];
    $cat=$_POST['category'];
    $cc=$_POST['catcode'];
    $pri=$_POST['price'];
    $date=$_POST['date'];
    $img=$_POST['image'];

    $query="update manageitems set ProductID='$pdid', ProductName='$name',
Quantity='$quan', Status='$stat', Category='$cat', CategoryCode='$cc', Price='$pri',
Date='$date', Image='$img' where ProductID='$pdid'";
    $res=mysqli_query($link,$query);
    if($res)
    {
    ?>
    <script language="javascript">alert('updated
successfully');window.location.replace('items.php');</script>

    <?php
    }
    else
    {
    ?>

    <script language="javascript">alert('Updation
failed');window.location.replace('items.php');</script>

    <?php
```

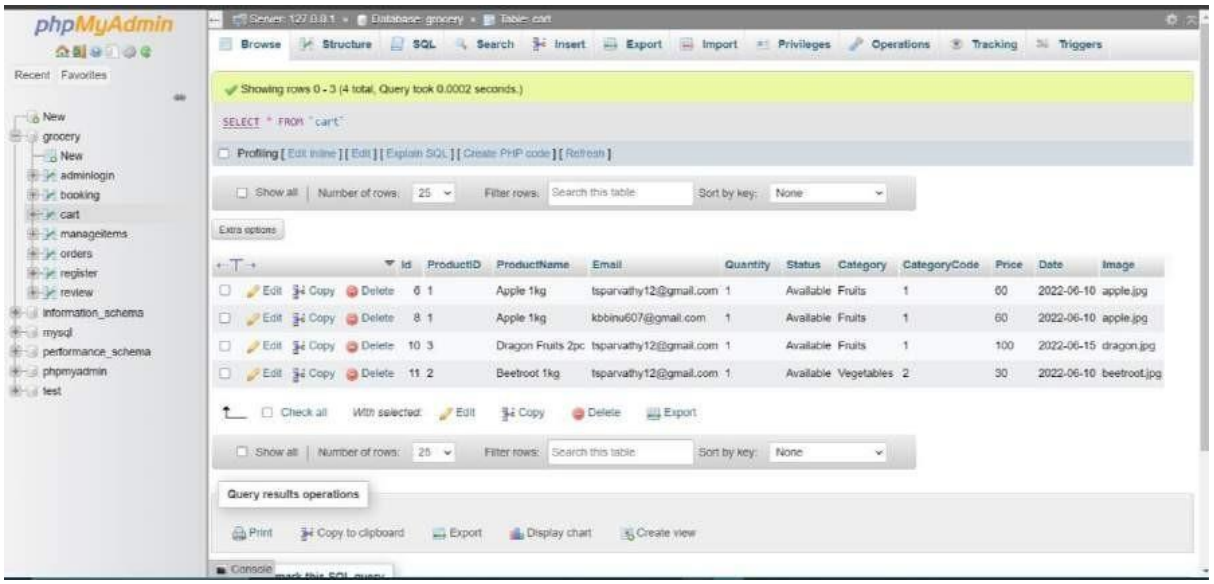
```
    }
}
    ?>
```

2.5.3 Delete Items

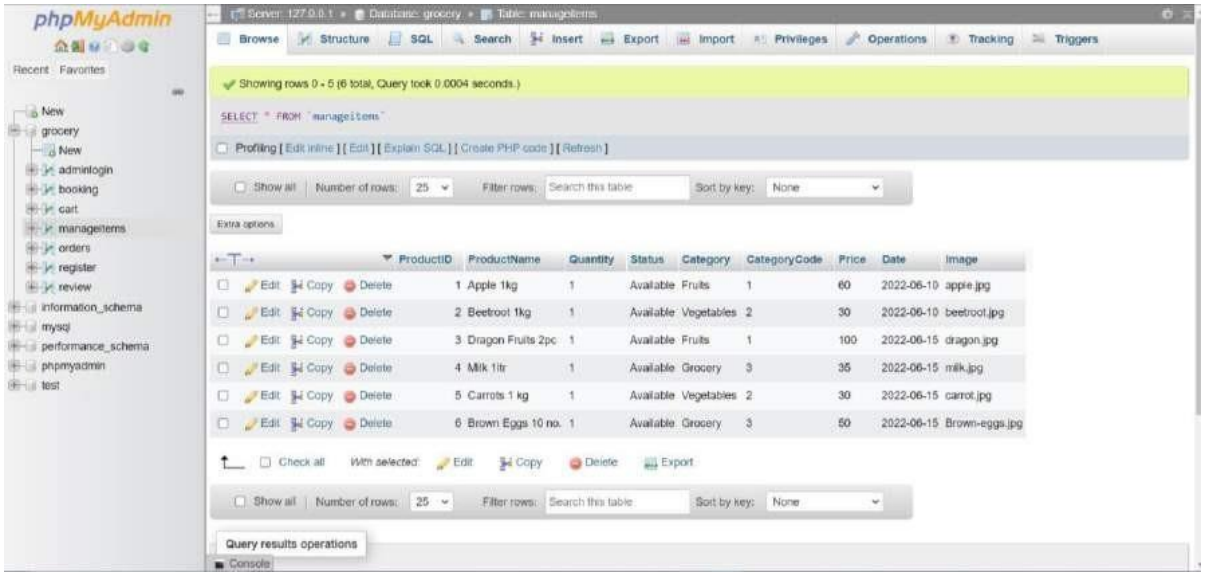
```
<?php
include 'connect.php';
$pdid=$_GET['ProductID'];
$query="delete from manageitems where ProductID='$pdid'";
$res=mysqli_query($link,$query);
if($res)
{
?>
<script language="javascript">alert('deleted
successfully');window.location.replace('items.php');</script>

<?php
}
else
{
?>

<script language="javascript">alert('deletion
failed');window.location.replace('items.php');</script>

<?php
}
?>
```

3. Table

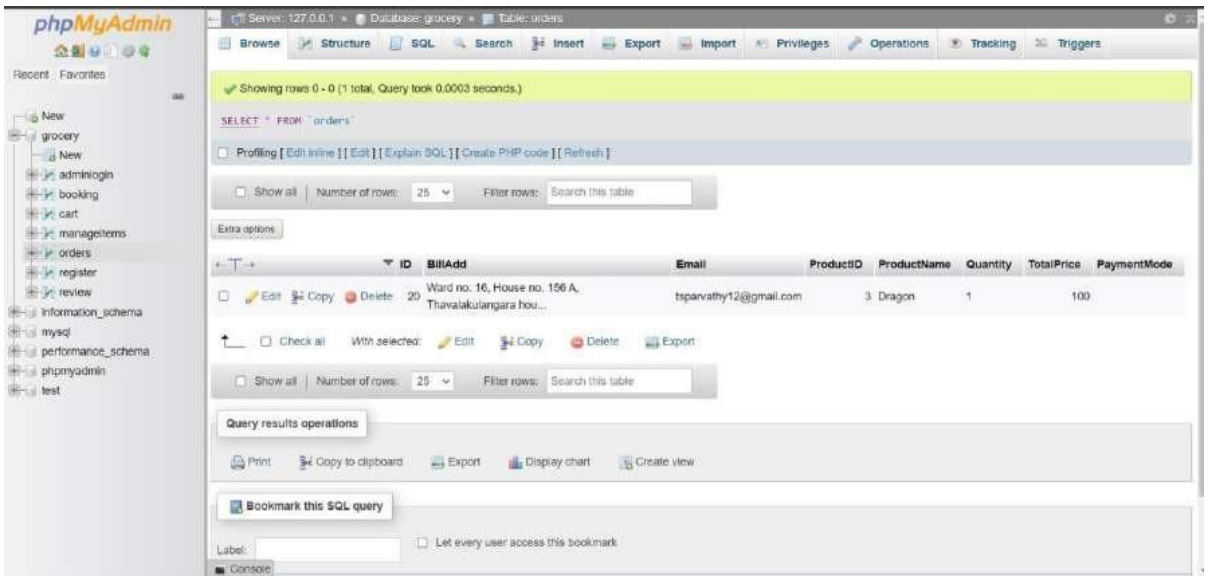
3.1 User – My Cart



3.2 Admin – Manage Items



3.3 User – Placed Orders



3.4 User – Reviews