1. INTRODUCTION

We introduce a simple but efficient and user-friendly website where users can have a comfortable time buying our products. This project is based on Online Grocery Store where users can buy products for their daily needs. This is fully dynamic project which is connected to the database. In this chapter, we will be looking into the modules of our project and the objective which our project wants to accomplish.

MODULE DESCRIPTION

ADMIN MODULE:

In this admin can add, update and delete categories, users and products according to the requirements. As this is fully dynamic project whatever the admin add here it will show to the user automatically. Admin can also view orders received, reviews and feedback from the users. All the admin pages are secured and cannot open without an admin login.

USER MODULE:

Users can sign up and login into the website to buy the products they need. All the content shown on the user's side is dynamic like categories, products and product details show are fetch from the database. Even the categories that are shown in the side menu are also dynamic, therefore it s connected to the database.

Users can see the product details and add their desired products to the cart. There are various payment options that one can use for making payment. After making the payment the user will get the order details and order number. User can also see the order details from the order history. The users can also search for their desired products by various search options available on the site. Users can also change their passwords whenever needed.

REGISTRATION/LOGIN MODULE:

Customer wants to buy the product then he/she must be registered, unregistered user can't go to the shopping cart. Customer logins to the system by entering valid user id and password for the shopping.

CART MODULE:

The customer after login or registration can make order or cancel order of the product from the shopping cart.

PAYMENT MODULE:

In this system we are dealing the mode of payment by Cash and UPI code. We will extend this to credit card, debit card etc. in the future.

OBJECTIVE

The objective of this project is to help the consumer to Save more to make their ordering more convenient and easier. For the customer, it can minimize the workload and effort of roaming around the grocery store. They can search the grocery items that they're looking for.

GENERAL OBJECTIVE:

The objective of this system is to save time and effort for the consumer. Save time and effort in terms of driving a car or commuting on a jeepney. In online grocery shopping you can just sit down relax and search for the product while in physical market you have to stand in line at the checkout counter and wait to load all your groceries packed. The proponents proposed an online grocery shopping system to lessen difficulty to the customer.

SPECIFIC OBJECTIVES:

- To develop an Online Grocery Shopping System
- To make the transaction easier and faster
- The branch manager can see all the orders to be process
- To display all the updated information
- Customers will choose their products and the corresponding grocery items will automatically load into their shopping cart.

2. LITERATURE SURVEY

This chapter presents a review of the articles and business reports related to consumers' grocery shopping decision making process, in both offline and online retail channels. The intent was to acquire a general overview of grocery shopping, in what pertains to this dissertation and subsequent research questions, and as such the focus relies mostly on the decisional phase and influencing pre-decisional phase of the grocery shopper decision making process. Based on the outcome of the literature review performed, a conceptual framework that guided the design and performance of the empirical studies, aiming at providing answers to the proposed research questions, is also presented.

A consumer purchase is typically a response to a problem or need, and once a consumer realizes this, he or she undergoes a series of steps until his or her need is satisfied. This is a reflection of the consumer decision making process, which main stages are generally defined in the Consumer Behaviour and Marketing literature as:

- Problem identification
- Information search
- Evaluation of alternatives
- Purchasing decision
- Post-purchase behaviour

Online grocery is expanding fast from its small base. Its market value has increased from 2016 to 2018, according to business insider intelligence. Online grocery shopping – "Analysing the paradigm shift in Consumer buying behaviour " The groceries segment of India is one of the major components of the Indian retail market of which the groceries holds a whopping share of 60% as food is the basic requirement of all the people irrespective of their class. Online grocery shopping is one of the mega-trends which involves sale and purchase of groceries over the internet. The concept though conceived in the west is gaining lot of popularity in the emerging markets of India.

Since the industry is growing at a tremendous pace it has attracted huge investments by venture capitalists and angel investors. The market has seen lot of emerging players of which Big Basket, Zopnow, Aaramshop and Local Banya leads from the front. These start-ups have been very successful because they currently are present in different cities and commanding leading sales. Although online grocery shopping is yet to catch up with the residents of many cities, yet all these e-commerce companies have started operating in many metro cities with an aim to change the way people do their grocery, veggie and staples shopping.

As several studies indicate, grocery shoppers have begun to find ways to spend less and reduce risk in the mist of the current economic and financial crisis. They have learned new tactics to save money on supermarket purchases and manage their household pantry, while shopping trips have also become more careful and focused.

Consumers' grocery shopping routine now regularly includes strategic and tactical features like clarifying wants versus needs, delaying gratification, lowering quality requirements, frequent channel, store and brand switching, an intense use of coupons, loyalty cards, shopping lists and other promotional offers, stockpiling and increasing purchase of private label products, among others. Furthermore, consumers are becoming increasingly less loyal to national brands and also less likely to engage in impulse buying or new product trial, as the new aim for grocery shopping is household gratification while maintaining quality but minimizing expenditure.

As the world continue grappling with the COVID-19 pandemic, online purchases for fresh food is gradually becoming the norm across the world. As such, food producers must be able to adapt accordingly to take advantage of the emerging market. However, the majority of consumers are still concerned about freshness and food waste. Unlike in a brick-and-mortar store where shoppers can visibly check the freshness of their produce, this is more difficult with home delivery.

Thus, brands must try and opt for packaging that can keep food safe and fresh during transit and displays its freshness to re-assure customers. Moreover, to meet sustainability goals, fresh food brands need to balance the use of more sustainable, recyclable materials, with packaging that continues to extend shelf life and avoid food waste.

One of the problems identified from the consumer's perspective is that most of the things mentioned in the online food menus are often not available. Instead, they act as click baits designed to entice online users to continue interacting with their platform and marketing content. In rare cases, some click bait links often forward online users to pages that require them to make payments, register, or even fill in their payment details. Consequently, a significant communication gap exists between consumers and restaurants while shopping on phone and online. While numerous studies examine the purchase intention of food among online shoppers, few highlight the inherent challenges experienced by consumers as they go about their day.

While it is crucial to investigate both perspectives, more studies need to be conducted on the customer ones. This is because most online businesses often find it difficult to deal with customers, but note that this is usually because they do not see things from the buyers' point of view. The authors, however, refutes the popular phrase that "customer is always right" and notes that even when they are completely wrong, they can always win. For example, customers can criticize a business online or even refuse to pay their bills. As such, failing to grasp a customer's perspective can result in a meltdown with them which is always bad business.

3. SYSTEM ANALYSIS

EXISTING SYSTEM & DRAWBACKS

The current system for shopping is to visit the shop manually and from the available grocery item choose the item user want and buying the item by payment of the price of the item. Grocery stores also offer non-perishable foods that are packaged in bottles, boxes, and cans; some also have bakeries, butchers, delis, and fresh produce. Large grocery stores that stock significant amounts of non-food grocery items, such as clothing and household items are called supermarkets. Some large supermarkets also include a pharmacy, and user service, redemption, and electronics sections.

Some grocery stores (especially large ones) form the centre piece of a larger complex that includes other facilities, such as gas stations, which will often operate under the store's name. Some groceries specialize in the foods of a certain nationality or culture. These stores are known as ethnic markets and may also serve as gathering places for immigrants. In many cases, the wide range of grocery items carried by larger supermarkets has reduced the need for such specialty stores the variety and availability of food is no longer restricted by the diversity of locally grown food or the limitations of the local growing season.

DRAWBACKS OF EXISTING SYSTEM:

- User must go to shop and select grocery items.
- It is difficult to identify the required grocery item.
- Description of the grocery item limited.
- It is a time-consuming process
- Not in reach of distant users.
- It is less user-friendly.

PROPOSE SYSTEM

In the proposed system user need not go to the shop for buying the grocery items. He can order the grocery item the wishes to buy through the application in his website. The shop owner will be admin of the system. Shop owner can appoint moderators who will help owner in managing the users and grocery item orders. The system also recommends a home delivery system for the purchased grocery items.

ADVANTAGES OF PROPOSE SYSTEM:

- Save on transportation costs. Back when I was working away from home, I simply scheduled my time to swing by the grocery store on the way home from work.

- Save time. A 28-mile round trip, plus whatever time it takes to shop can easily burn up an hour or more of my day.

- Stay organized.

SCOPE OF THE PROPOSED SYSTEM:

The Online Grocery market is to increase the profit margin of business owners and conveniently grant access to products available in the supermarket to the potential customers. The online grocery market will be made up various parts which include:

- Product catalogue: An organization of the products will be made for clear navigation of the site which will mean it will be browse able by all.

- Product attributes: Customers need to know what product they need before logging on to make a purchase. Attributes are generally accepted as a must: product name, category, description, price, and photo / image of the product.

- Check out: The e Commerce checkout needs to provide a way to take payment, append relevant taxes (if necessary), compute shipping and handling costs, provide a sub-total of the amount due, collect billing and shipping information from the customer.

- Security is vital: The website will be encrypted with HTTPS, at a minimum to ensure losses are not made.

ANALYSIS

Friends of Earth believed that e-grocery method is the best way to promote industry’s brand image. However, Fox and Karpiak and Kotler stated conversely; brand image of company will help to promote online grocery business. They also believed that industry’s business experience and its infrastructure will be added advantage for promotion. One more factor is people ability to spend money for long duration household goods.

Pachauri Moesha stated that competition in e-grocery retail sector remains the same until company stay in game. If any child company’s online business started down falling then parental company may not let it to happen for longer duration because they are inter-related in means of profit sharing. Without the demand, services of online business company will lead to business failure. Retention of existing customer is one of the main aspects of business. To gain customer loyalty, companies should provide reward to regular shopper. Customers can also gain loyalty points for their spending and it will convert into reward coupons or vouchers. Brick and mortar stores utilise their existing business infrastructure to click and mortar stores to save on costs.

Online shopping and home delivery service is one of the critical aspects of e-grocery business. E-grocery business should have to understand the variable cost affecting structures and service concepts in order to turn online grocery business into profitable one. The different strategies were discussed. Analysing and adopting best strategy for e-grocery business is mast important. This literature provided insight of e-groceries. In a service context, as with product marketing, getting and keeping customers is of utmost importance. With service, in particular, someone within the organization will interact with customers; hence they should be viewed as customer relationships not just customers. Customer relationships have to be built and a useful way of viewing these developments as a life cycle. The marketing objective will change as the customer proceeds through the cycle. Understanding service marketing is becoming increasingly important for all organizations as more firms seek to differentiate their product or services through their service offering. Interaction and customer relationship concepts, two key areas of service marketing are set to be of even greater importance to all firms in the future.

On one hand, many customers are interested to do online shopping for their household goods, but yet they are not interest to abandon shopping in-store in traditional way. Most of the people still believed that, online shopping is too expensive. On the other hand, many more industries also made their effort to build online stores by providing this extra service to the customers. Organisations are continuously striving hard to build different strategic approaches for fulfilment of orders in profitable way. However, online unit business will add extra expenditure to grocers with or without some nominal charges. The profit margin is comparably very low in e-grocery to traditional one.

SYSTEM ANALYSIS

The main components of this system are registration, login, browsing of items, ordering and view history and order details. The users have to register themselves then only they will be able to buy goods. The registered members have to login first in order to purchase the goods. After that, the users (consumers) need to select the required goods and add it to the cart. At checkout, they have to pay the amount. The order details and history of ordered items can be viewed as well by the consumers.

ADVANTAGES:

The major goals of this system are:

- Much of the time can be saved while shopping for groceries over the internet. Customers don’t have to wait in a long queue.
- They don’t have to carry the load of groceries as they are delivered by us to their doorstep.
- It is a good option for people who are old and physically weak.
- The overspending can be prevented.
- Online store is suitable for busy people who don’t have time to visit the local store.
- Online grocery shopping is available 24 hours a day and 7 days a week and 365 days in a year.
- Customers can keep track of the number of grocery items purchased by looking at the shopping cart which helps in purchasing grocery within the budget.
- Customers are able to compare the price of grocery items of different brands with ease.
- Fuel costs can be reduced.
- The consumers are also allowed to modify their carts individually. They can add/remove grocery item.
- It is user- friendly and easy to use for both consumers and vendors.

The developed online store will enable user to shop grocery goods among wide varieties through secured payment option. It will enhance the user experience & promote hassle-free shopping environment.

After going through e-commerce processes and real market trends, we aim at developing an online grocery store having simple and easy to use interface and secured transaction. It will provide convenience to consumer.

FEASIBILITY STUDY

As the name suggest, feasibility study is a study to reveal whether a project is feasible or not. A feasibility study is an analysis that considers all of a project's relevant factors—including economic, technical, legal, and scheduling considerations—to ascertain the likelihood of completing the project successfully.

Whether a project is feasible or not can depend on several factors, including the project's cost and return on investment, meaning whether the project generated enough revenue or sales from consumers.

The types of feasibility namely:

- Technical feasibility
- Financial feasibility
- Time feasibility
- Operational feasibility

TECHNICAL FEASIBILITY:

The project is quite feasible technically as it can be implemented using the support and features provided by the programming languages and handy software tools which are easily available to user. Also, with the technical support of the books available, internet resources and internal staff the technical obstacles that are expected/unexpected could be resolved without much delay. Also, there is no special hardware involved in the system. Thus, the overall project is technically feasible.

FINANCIAL FEASIBILITY:

Since no special hardware is required for the system the direct or indirect cost required for the development and the deployment of the project is reduced. As our project consists of a webpage and a mobile application it does not require any financial help. We will only need a web domain and nothing else. Hence, our project is financially very feasible.

TIME FEASIBILITY:

As the requirements of the project are not large and the objective of project is well defined, well understood among the team members and the schedule for the project is initially marked out, the project can be satisfactorily completed within the expected timeline. However, some issues may arise due to lot of contents but, they can be solved with proper planning and team efforts. Thus, the project is timely feasible.

OPERATIONAL FEASIBILITY:

As our project consist of a webpage and a mobile application having a very Simple Graphical User Interface. So to use it a person need not to be a highly technical person, even a common person can use it very easily .The person using our web site or application does not need to know any kind of programming language and also does not need to have technical knowledge .people belonging to any age group can use our website and application without any issues .Overall ,our project is very much operationally feasible the buyer age profile in which, the online market for grocery have most buyer from 25-35 age and the least from below 18 age

4. SOFTWARE REQUIREMENT SPECIFICATION

INTRODUCTION

The production of the requirements stage of the software development process is Software Requirements Specifications (SRS) (also called a requirements document). This report lays a foundation for software engineering activities and is constructing when entire requirements are elicited and analysed. SRS is a formal report, which acts as a representation of software that enables the customers to review whether it (SRS) is according to their requirements. Also, it comprises user requirements for a system as well as detailed specifications of the system requirements.

The SRS is a specification for a specific software product, program, or set of applications that perform particular functions in a specific environment. It serves several goals depending on who is writing it. First, the SRS could be written by the client of a system. Second, the SRS could be written by a developer of the system. The two methods create entirely various situations and establish different purposes for the document altogether. The first case, SRS, is used to define the needs and expectation of the users. The second case, SRS, is written for various purposes and serves as a contract document between customer and developer.

OVERVIEW OF SYSTEM

The Online Shopping system application enables vendors to set up online shops, customers to browse through the shops, and a system administrator to approve and reject requests for new shops and maintain lists of shop categories. Also, the developer is designing an online shopping site to manage the items in the shop and also help customers to purchase them online without visiting the shop physically. The online shopping system will use the internet as the sole method for selling goods to its consumers.

FUNCTIONAL & NON-FUNCTIONAL REQUIREMENT

Various functional modules can be implemented by the system. it provides a requirement overview of the system.

- ➢ Master Maintenance: - Is the system that maintains the detail of the products and their hierarchy attributes (size, weight, cost etc) The two main components of master maintenance are as follows.
- ➢ Transaction: -Transaction is a payment method in which the transfer of money of buying products. This process is secure and password protected. Three steps involved in the online transaction are Registration, Placing an order, and, Payment.
- ➢ Order Report: -List of the products that can be buy by the customer.
- ➢ Delivery Report: -List of the products that can be delivered to the customer.
- ➢ Registration: -Customer wants to buy the product then he/she must be registered, unregistered user can't go to the shopping cart.
- ➢ Login: -Customer logins to the system by entering valid user id and password for the shopping.
- ➢ Changes to Cart: -Changes to cart means the customer after login or registration can make order or cancel order of the product from the shopping cart.
- ➢ Payment: - In this system we are dealing the mode of payment by Cash. We will extend this to credit card, debit card etc. in the future.
- ➢ Logout: - After ordering or surfing for the product customer has to logout.

Here are some basic types of non-functional requirements that should make it to the website specification document of all ecommerce businesses.

- ➢ Usability: No matter the size of your business, you want your website to be intuitive and easy-to-use. It takes about 0.05 seconds for users to figure out if your website is worth their time and attention. So, you'll definitely want to work on your homepage design, calls-to-action, and easy checkout to get past those milliseconds of doom. Website usability is also defined by

  - o how easily a user can achieve their goal in a single page visit;
  - o how quickly they perform the tasks in the store;
  - o how memorable and intuitive the design is;
  - o number and time of errors users make.

➢ Security: Security is paramount while dealing with monetary transactions and sensitive data. A simple SSL certification and data privacy policy will instil trust into your website and convert the customers into your brand advocates. It is also about different admin roles allowing you to control who can create, see, copy, change, or delete information. Depending on your business location, security also means complying with the customer data protection rules.There are many factors at play when it comes to security; specifying this non-functional requirement means taking the first step to ecommerce fraud prevention.

➢ Performance: If your goal is increasing your website traffic, performance should be the priority NFR in your specification document. This NFR is often found in briefs from large enterprises or websites with legacy architecture: they want their e-stores to load fast no matter the number of integrations and sales seasons. Set up the speed benchmark, a maximum number of SKUs to be added, or any other performance indicator suitable to your business. Don't include third-party system delivery time, though; your developers can't do much if a certain business operation depends on an API call to another database.

➢ Maintainability: It's widely known that the tricky part of planning a business budget is accounting for the operational costs of business maintenance. Striving to make the website maintainable from the initial development phase means cutting the time and cost to identify and resolve the system faults in the future. As saddening as it may seem, there's no escape from the future issues and you can see many cues on how to maintain an ecommerce website. But your task is to make the system easy-to-maintain right from its launch.

➢ Scalability: If you're looking into the future-proof solution, scalability should be your take. This requirement defines how the website can grow and expand its functionality without affecting its performance. You should be able to add more memory, servers, or disc space to complete more transactions on your website. On the server side, you might want to add localization features in case you plan to enter new markets and sell products internationally. Overall, this NFR accounts for painless business expansion and has both hardware and software implications.

DOMAIN REQUIREMENTS

Domain requirements are expectations related to a particular type of software, purpose or industry vertical. Domain requirements can be functional or non-functional. The common factor for domain requirements is that they meet established standards or widely accepted feature sets for that category of software project.

Domain requirements typically arise in military, medical and financial industry sectors, among others. One example of a domain requirement is for software in medical equipment:

Software can be functional and usable but not acceptable for production because it fails to meet domain requirements.

SYSTEM REQUIREMENTS

- Processor: 1.9 gigahertz (GHz) x86- or x64-bit dual core processor with SSE2 instruction set
- Memory: 2-GB RAM
- Display: Super VGA with a resolution of 1024 x 768
- Bandwidth greater than 50 KBps (400 kbps)
- Latency under 150 ms

5. SYSTEM DESIGN

INTRODUCTION

Systems design is the process of defining elements of a system like modules, architecture, components and their interfaces and data for a system based on the specified requirements. It is the process of defining, developing and designing systems which satisfies the specific needs and requirements of a business or organization.

UML DIAGRAMS

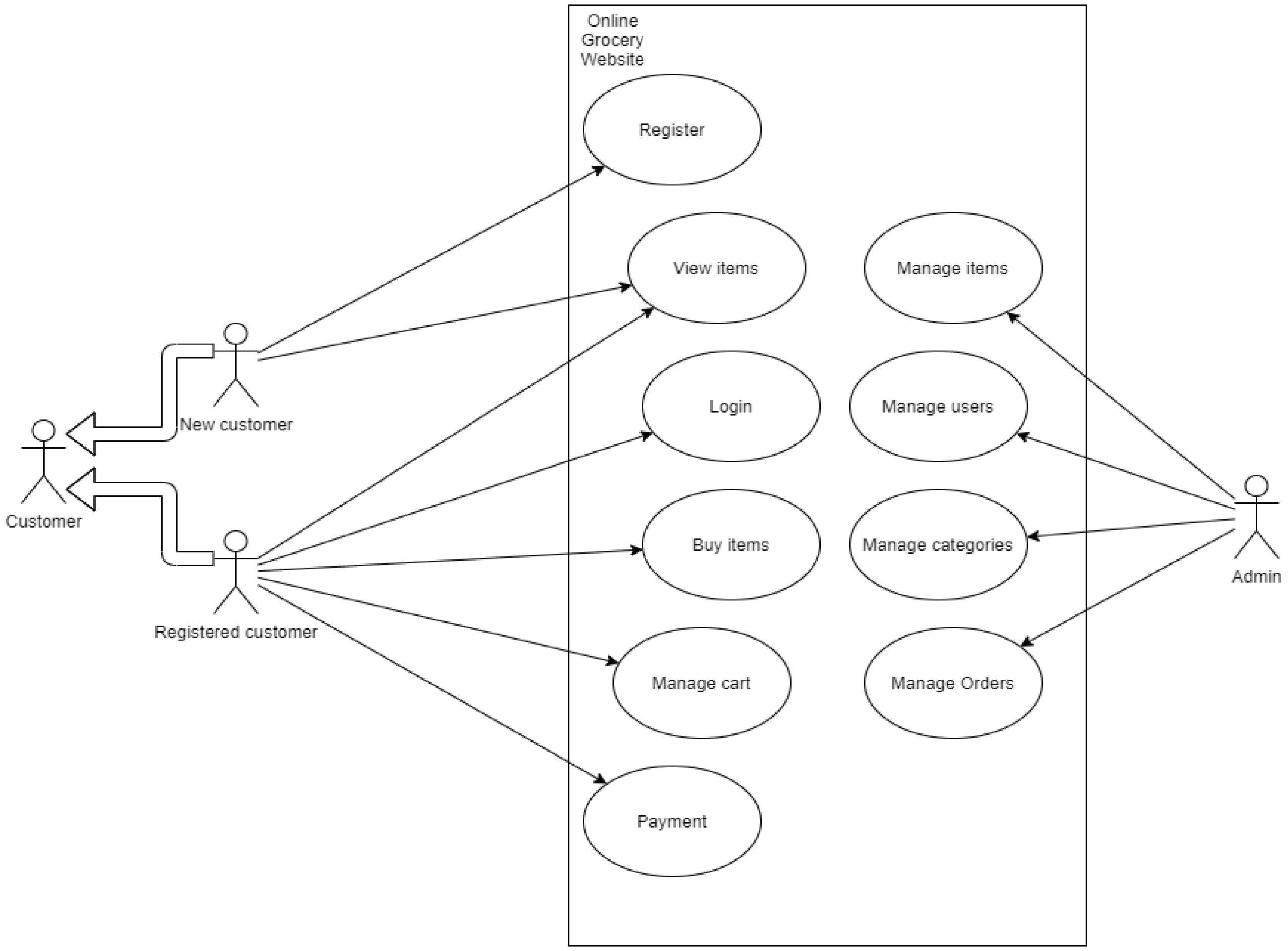
UML (Unified Modelling Language) is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems.

- UML stands for **Unified Modelling Language**.
- UML is different from the other common programming languages such as C++, Java, COBOL, etc.
- UML is a pictorial language used to make software blueprints.
- UML can be described as a general-purpose visual modelling language to visualize, specify, construct, and document software system.
- Although UML is generally used to model software systems, it is not limited within this boundary. It is also used to model non-software systems as well. For example, the process flow in a manufacturing unit, etc.

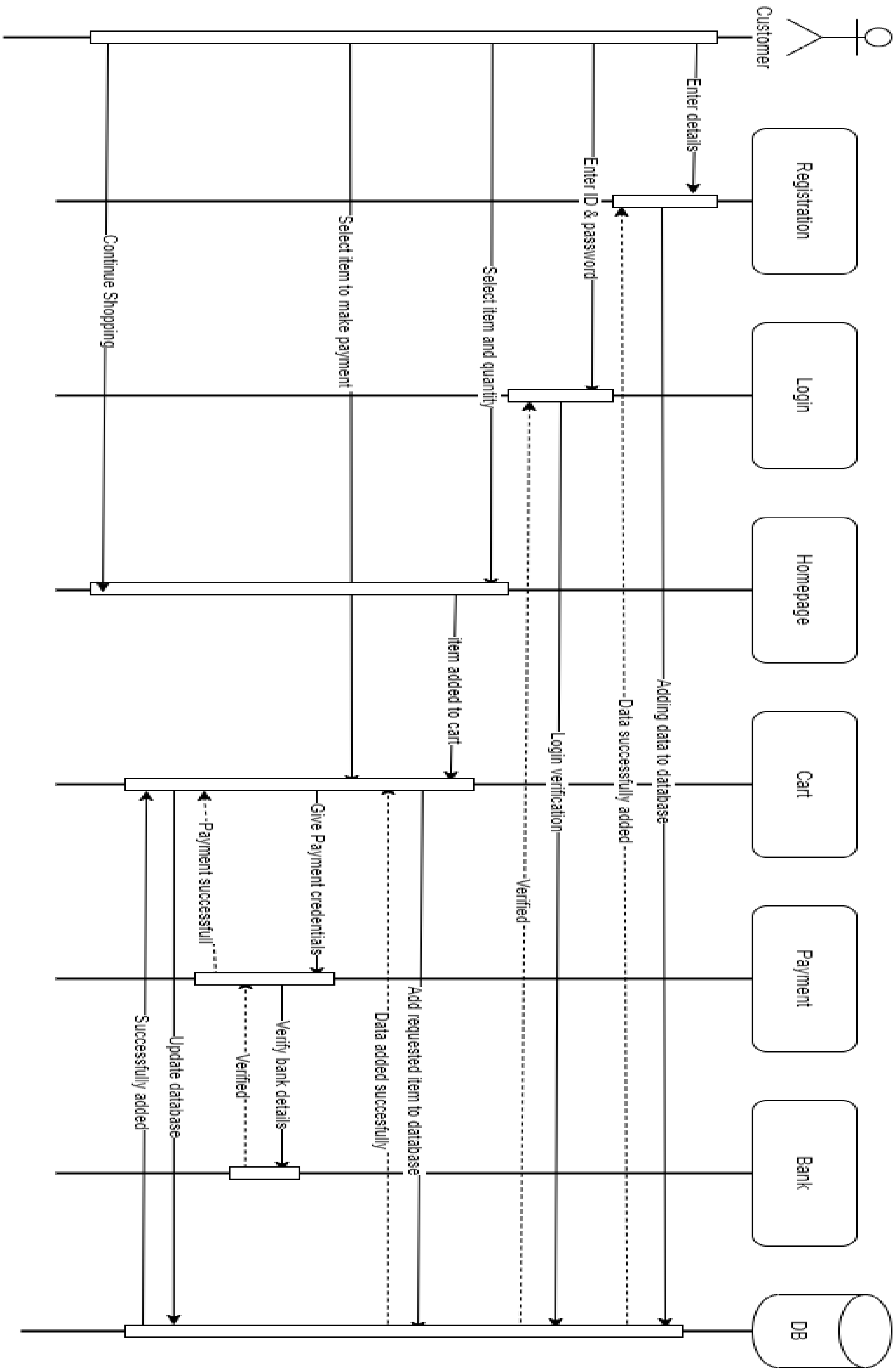
GOALS OF UML:

- *A picture is worth a thousand words*, this idiom absolutely fits describing UML. Object-oriented concepts were introduced much earlier than UML. At that point of time, there were no standard methodologies to organize and consolidate the object-oriented development. It was then that UML came into picture.
- The most important goal is to define some general-purpose modelling language, which all modelers can use and it also needs to be made simple to understand and use.
- UML diagrams are not only made for developers but also for business users, common people, and anybody interested to understand the system. Thus, it must be clear that UML is not a development method rather it accompanies with processes to make it a successful system.
- In conclusion, the goal of UML can be defined as a simple modelling mechanism to model all possible practical systems in today’s complex environment.

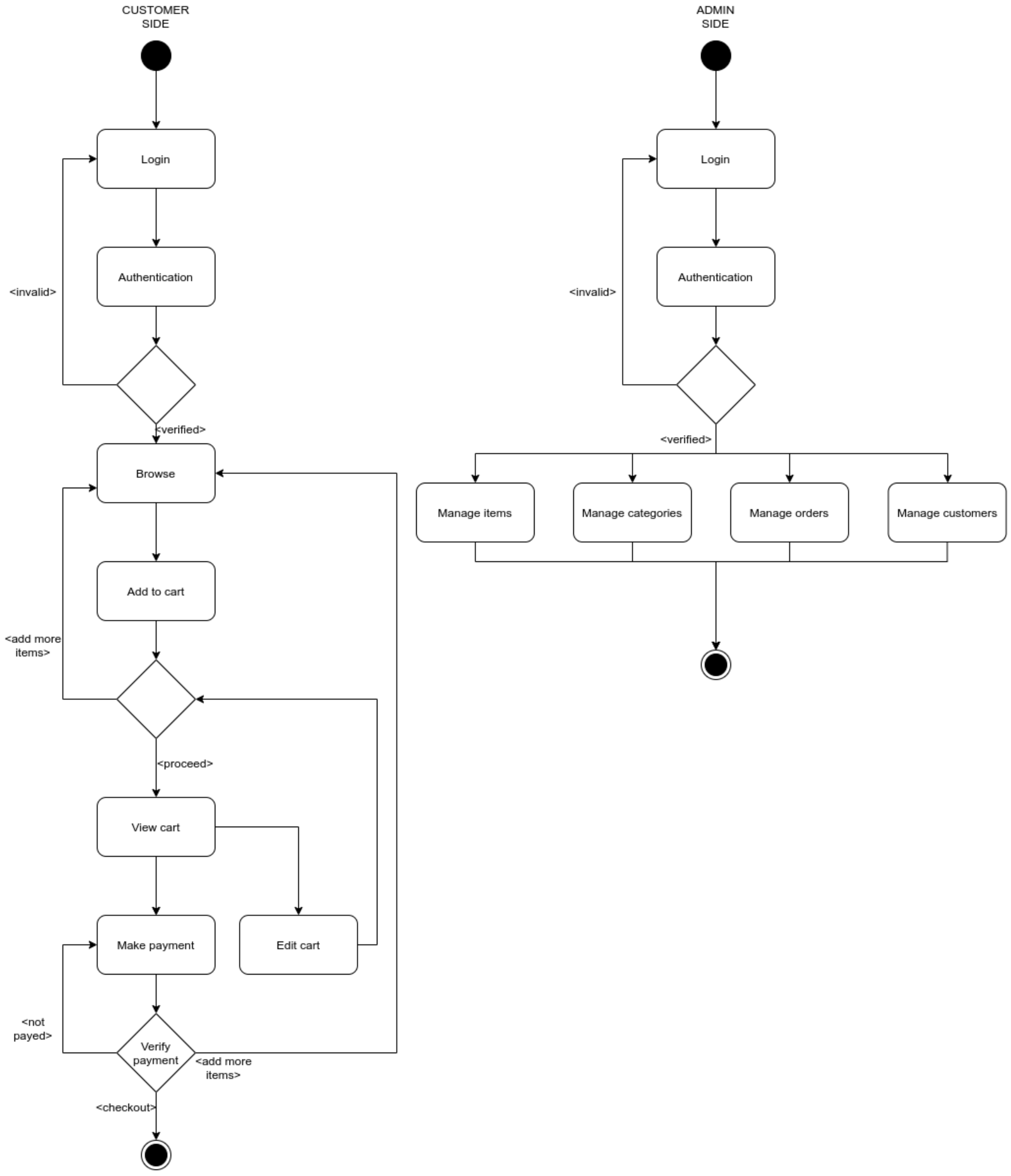
USECASE DIAGRAM



SEQUENCE DIAGRAM

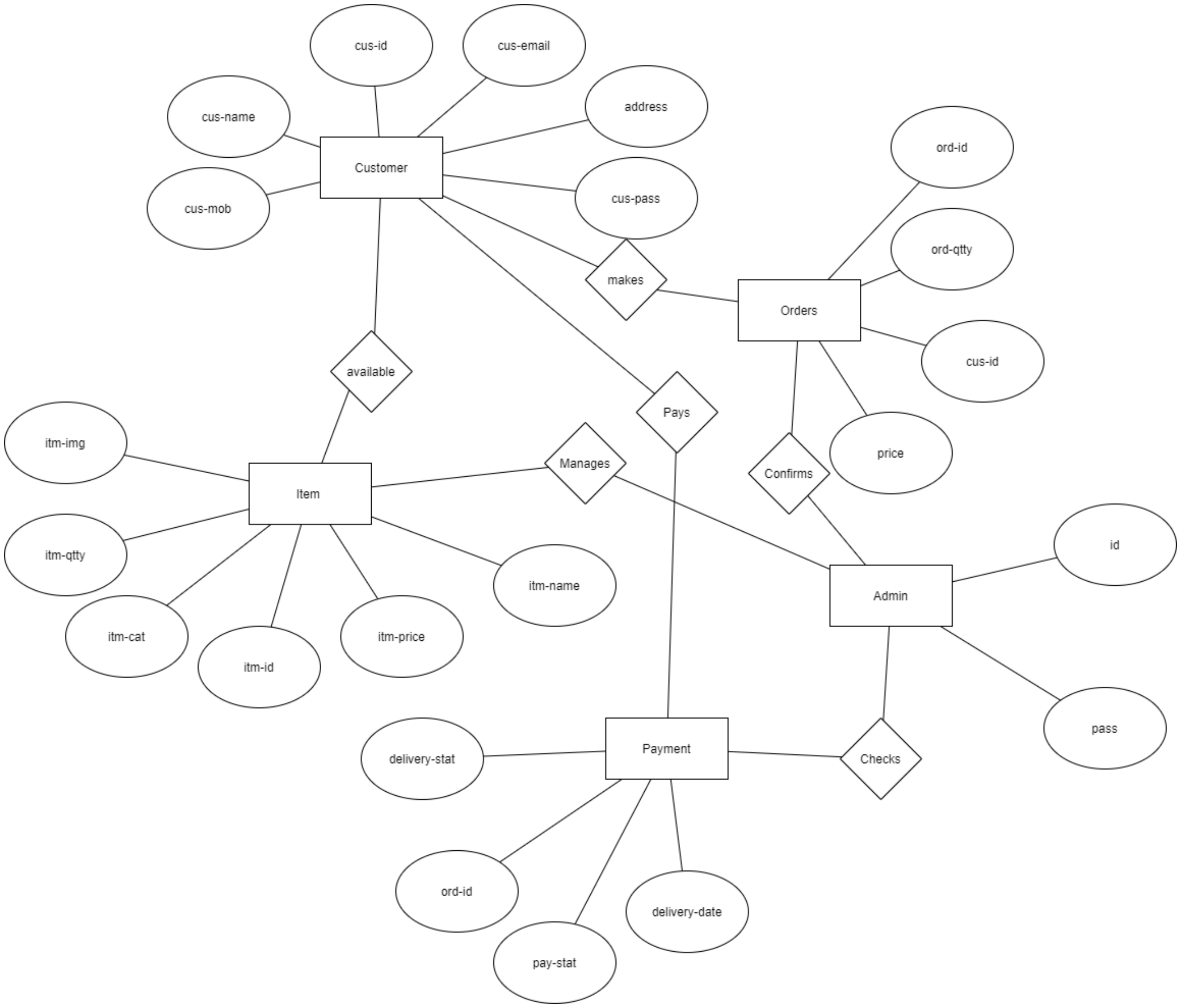


ACTIVITY DIAGRAM



6. DATABASE DESIGN

ER DIAGRAM


cus-id
cus-email
cus-name
address
Customer
cus-mob
cus-pass
ord-id
ord-qtty
makes
Orders
cus-id
available
Pays
price
itm-img
Manages
Confirms
Item
id
itm-qtty
itm-name
Admin
itm-cat
itm-price
itm-id
pass
delivery-stat
Payment
Checks
ord-id
delivery-date
pay-stat

TABLES

COLUMN-NAME	DATA-TYPE	PRIMARY KEY
Cus-id	nvarchar (50)	1
Cus-name	nvarchar (50)	-
Cus-pass	nvarchar (50)	-
Cus-mob	char (10)	-
Cus-email	nvarchar (50)	-
Cus-address	nvarchar (MAX)	-

COLUMN-NAME	DATA-TYPE	PRIMARY KEY
Ord-id	nvarchar (50)	1
Ord-qtty	nvarchar (50)	-
Cus-id	nvarchar (50)	-
price	int	-

COLUMN-NAME	DATA-TYPE	PRIMARY KEY
Ord-id	nvarchar (50)	1
Pay-stat	nvarchar (50)	-
Delivery-date	nvarchar (50)	-
Delivery-stat	nvarchar (50)	-

COLUMN-NAME	DATA-TYPE	PRIMARY KEY
Itm-img	nvarchar (MAX)	-
Itm-qtty	nvarchar (50)	-
Itm cat	nvarchar (50)	-
Itm-id	nvarchar (50)	1
Itm-name	nvarchar (50)	-
Itm-price	int	-

COLUMN-NAME	DATA-TYPE	PRIMARY KEY
Id	nvarchar (50)	1
pass	nvarchar (50)	-

NORMALIZATION

Normalization is the process of minimizing **redundancy** from a relation or set of relations. Redundancy in relation may cause insertion, deletion and updation anomalies. So, it helps to minimize the redundancy in relations. **Normal forms** are used to eliminate or reduce redundancy in database tables.

FIRST NORMAL FORM (1NF):

- A relation will be 1NF if it contains an atomic value.
- It states that an attribute of a table cannot hold multiple values. It must hold only single-valued attribute.
- First normal form disallows the multi-valued attribute, composite attribute, and their combinations.

Customer table:

Cus-id	**Cus-name**	**Cus-mobile**
14	John	7272826385, 9064738238
20	Harry	8574783832
12	Sam	7390372389, 8589830302

The decomposition of the Customer table into 1NF has been shown below:

Cus-id	**Cus-name**	**Cus-mobile**
14	John	7272826385
14	John	9064738238
20	Harry	8574783832
12	Sam	7390372389
12	Sam	8589830302

SECOND NORMAL FORM (2NF):

- In the 2NF, relational must be in 1NF.
- In the second normal form, all non-key attributes are fully functional dependent on the primary key

Let's assume, our website stores data of our customer details and purchased item. A customer can purchase more than one item.

Customer table:

Cus-id	Itm-name	Cus-name
25	Egg	John
25	Milk	John
47	Butter	Sam
83	Oil	Harry
83	Apple	Harry

In the given table, non-prime attribute cus-name is dependent on cus-id which is a proper subset of a candidate key. That's why it violates the rule for 2NF.

To convert the given table into 2NF, we decompose it into two tables:

Customer details table:

Cus-id	Cus-name
25	John
47	Sam
83	Harry

Item name table:

Cus-id	Itm-name
25	Egg
25	Milk
47	Butter
83	Oil
83	Apple

THIRD NORMAL FORM (3NF):

- A relation will be in 3NF if it is in 2NF and not contain any transitive partial dependency.
- 3NF is used to reduce the data duplication. It is also used to achieve the data integrity.
- If there is no transitive dependency for non-prime attributes, then the relation must be in third normal form.
- A relation is in third normal form if it holds at least one of the following conditions for every non-trivial function dependency X → Y.X is a super key.Y is a prime attribute, i.e., each element of Y is part of some candidate key.

Customer details table:

Cus-id	**C**us-name	**C**us-pin	**C**us-state	**C**us-city
222	Harry	201010	UP	Noida
333	Stephan	02228	US	Boston
444	Lan	60007	US	Chicago
555	Katharine	06389	UK	Norwich
666	John	462007	MP	Bhopal

Super keys are: {**Cus**-id}, {**Cus**-id, **C**us-name}, {**Cus**-id, **C**us-name, **C**us-pin} .... Candidate key is {**Cus**-id}. All attributes except **C**us-id are non-prime.

Here, **C**us-state & **C**us-city dependent on **C**us-pin and **C**us-pin dependent on **C**us-id. The non-prime attributes (**C**us-state, **C**us-city) transitively dependent on super key **C**us-id It violates the rule of third normal form.

That's why we need to move the **C**us-city and **C**us-state to the new **C**ustomer-address table, with **C**us-pin as a Primary key.

Customer table:

Cus-id	**C**us-name	**C**us-pin
222	Harry	201010
333	Stephan	02228
444	Lan	60007
555	Katharine	06389
666	John	462007

Customer-address table:

Cus-pin	**C**us-state	**C**us-city
201010	UP	Noida
02228	US	Boston
60007	US	Chicago
06389	UK	Norwich
462007	MP	Bhopal

7.IMPLEMENTATION

SIGNUP/LOGIN C#:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data.SqlClient;
using System.Data;
using System.Drawing;
public partial class signui : System.Web.UI.Page{
    protected void Page_Load(object sender, EventArgs e){
           if(Request.QueryString["check"]== "no"){
                  Response.Write("<script> alert('Not signed in yet'); </script>");
           }
           Panel2.Visible = true;
           Panel1.Visible = false;
           Panel4.Visible = false;
           TextBox1.BorderColor = System.Drawing.ColorTranslator.FromHtml("#808080");
           TextBox2.BorderColor = System.Drawing.ColorTranslator.FromHtml("#808080");
    }
    protected void Button2_Click(object sender, EventArgs e){
           Panel2.Visible = true;
           Panel1.Visible = false;
    }
    protected void Button3_Click(object sender, EventArgs e){
           Panel2.Visible = false;
           Panel1.Visible = true;
    }
    protected void Button1_Click(object sender, EventArgs e){
           SqlConnection con = new SqlConnection(@"Data Source=LAPTOP-BJQ4MMQG; Initial
           Catalog=mdas; Integrated Security=True;");
           SqlDataAdapter sda = new SqlDataAdapter("select * from logup where
           email='"+TextBox1.Text+"' and password='"+TextBox2.Text+"'",con);
           DataTable dt = new DataTable();
           sda.Fill(dt);
           if (TextBox1.Text == "admin" & TextBox2.Text == "admin"){
                  Session["username"] = "admin";
                  Response.Redirect("adminpage.aspx");
           }
           if (dt.Rows.Count == 1){
                  Session["username"] = TextBox1.Text;
                  Response.Redirect("cart.aspx?cartclear=y");
           }
           else{
                  Panel4.Visible = true;
                  TextBox1.BorderColor = Color.Red;
                  TextBox2.BorderColor = Color.Red;
           }
    }
    protected void Button4_Click(object sender, EventArgs e){
           string query = "select count(*) from logup where replace(email,' ','')='" +
           TextBox4.Text.Trim() + "'";
           SqlConnection con = new SqlConnection("Data Source=LAPTOP-BJQ4MMQG; Initial
           Catalog=mdas; Integrated Security=True;");
           SqlCommand cmd1 = new SqlCommand(query, con);
```

```
        cmd1.Parameters.AddWithValue("@email", TextBox4.Text.Trim());
        con.Open();
        int count = Convert.ToInt32(cmd1.ExecuteScalar());
        if (count > 0){
            Response.Write("<script>alert('E-mail ID already exists.');</script>");
        }
        else{
            SqlCommand cmd2 = new SqlCommand("insert into logup" + "(name, age, gender,
            mobile, email, address, password) values(@name, @age, @gender, @mobile,
            @email, @address, @password)", con);
            cmd2.Parameters.AddWithValue("@name", TextBox8.Text);
            cmd2.Parameters.AddWithValue("@age", TextBox9.Text);
            cmd2.Parameters.AddWithValue("@gender", DropDownList1.SelectedItem.Value);
            cmd2.Parameters.AddWithValue("@mobile", TextBox3.Text);
            cmd2.Parameters.AddWithValue("@email", TextBox4.Text);
            cmd2.Parameters.AddWithValue("@address", TextBox7.Text);
            cmd2.Parameters.AddWithValue("@password", TextBox5.Text);
            cmd2.ExecuteNonQuery();
            Response.Redirect("logui.aspx");
        }
    }
    protected void LinkButton1_Click(object sender, EventArgs e){
        Panel2.Visible = false;
        Panel1.Visible = true;
    }
}
```

CART C#:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data.SqlClient;
using System.Data;
public partial class cart : System.Web.UI.Page{
    protected void Page_Load(object sender, EventArgs e){
           if (Request.QueryString["cartclear"] != null) {
                  Session["buyitems"] = null;
                  Response.Redirect("products.aspx");
           }
           if (!IsPostBack){
                  if (Session["buyitems"] == null){
                  Button3.Enabled = false;
           }
           else{
                  Button3.Enabled = true;
           }
           Session["addproduct"] = "false";
           DataTable dt = new DataTable();
           DataRow dr;
           dt.Columns.Add("slno");
           dt.Columns.Add("pid");
           dt.Columns.Add("pname");
           dt.Columns.Add("pqty");
           dt.Columns.Add("pprice");
           dt.Columns.Add("tprice");
           if (Request.QueryString["id"] != null){
                  if (Session["buyitems"] == null){
                         dr = dt.NewRow();
                         SqlConnection conn = new SqlConnection("Data Source=LAPTOP-BJQ4MMQG;
                         Initial Catalog=mdas; Integrated Security=True;");
                         SqlDataAdapter da = new SqlDataAdapter("select * from prodadd where
                         pid=" + Request.QueryString["id"], conn);
                         DataSet ds = new DataSet();
                         da.Fill(ds);
                         dr["slno"] = 1;
                         dr["pid"] = ds.Tables[0].Rows[0]["pid"].ToString();
                         dr["pname"] = ds.Tables[0].Rows[0]["pname"].ToString();
                         dr["pprice"] = ds.Tables[0].Rows[0]["pprice"].ToString();
                         dr["pqty"] = Request.QueryString["qty"];
                         decimal price =Convert.ToDecimal(ds.Tables [0].Rows [0] ["pprice"]
                         .ToString());
                         decimal quantity = Convert .ToDecimal (Request.QueryString ["qty"]
                         .ToString());
                         decimal totalprice = price * quantity;
                         dr["tprice"] = totalprice;
                         dt.Rows.Add(dr);
                         GridView1.DataSource = dt;
                         GridView1.DataBind();
                         Session["buyitems"] = dt;
                         Button3.Enabled = true;
                         GridView1.FooterRow.Cells[4].Text = "total amount";
                         GridView1.FooterRow.Cells[5].Text = grandtotal().ToString();
```

```
                    Response.Redirect("cart.aspx");
                }
                else{
                    dt = (DataTable)Session["buyitems"];
                    int sr;
                    sr = dt.Rows.Count;
                    dr = dt.NewRow();
                    SqlConnection scon = new SqlConnection("Data Source=LAPTOP-BJQ4MMQG;
                    Initial Catalog=mdas; Integrated Security=True;");
                    SqlDataAdapter da = new SqlDataAdapter("select * from prodadd where
                    pid=" + Request.QueryString["id"], scon);
                    DataSet ds = new DataSet();
                    da.Fill(ds);
                    dr["slno"] = sr + 1;
                    dr["pid"] = ds.Tables[0].Rows[0]["pid"].ToString();
                    dr["pname"] = ds.Tables[0].Rows[0]["pname"].ToString();
                    dr["pprice"] = ds.Tables[0].Rows[0]["pprice"].ToString();
                    dr["pqty"] = Request.QueryString["qty"];
                    decimal price =Convert .ToDecimal (ds.Tables[0] .Rows[0]["pprice"]
                    .ToString());
                    decimal quantity = Convert .ToDecimal(Request .QueryString["qty"]
                    .ToString());
                    decimal totalprice = price * quantity;
                    dr["tprice"] = totalprice;
                    dt.Rows.Add(dr);
                    GridView1.DataSource = dt;
                    GridView1.DataBind();
                    Session["buyitems"] = dt;
                    Button3.Enabled = true;
                    GridView1.FooterRow.Cells[4].Text = "total amount";
                    GridView1.FooterRow.Cells[5].Text = grandtotal().ToString();
                    Response.Redirect("cart.aspx");
                }
            }
            else{
                dt = (DataTable)Session["buyitems"];
                GridView1.DataSource = dt;
                GridView1.DataBind();
                if (GridView1.Rows.Count > 0){
                    GridView1.FooterRow.Cells[4].Text = "Total amount";
                    GridView1.FooterRow.Cells[5].Text = grandtotal().ToString();
                }
            }
        }
        string orderdate = DateTime.Now.ToShortDateString();
        Session["orderdate"] = orderdate;
        orderid();
    }
    public void orderid(){
        String alpha = "abcdefghijklmnopqrstuvwxyz0123456789";
        Random r = new Random();
        char[] myarray = new char[5];
        for (int i = 0; i < 5; i++){
            myarray[i] = alpha[(int)(35 * r.NextDouble())];
        }
        String orderid;
        orderid = "Order_Id:" + DateTime.Now.Hour .ToString() +DateTime.Now.Second
        .ToString() + DateTime.Now.Day .ToString() +DateTime.Now.Month .ToString() +
        DateTime.Now.Year .ToString() + new string(myarray) +DateTime.Now.Minute
        .ToString() + DateTime.Now.Second .ToString();
        Session["orderid"] = orderid;
    }
    public decimal grandtotal(){
        DataTable dt = new DataTable();
        dt = (DataTable)Session["buyitems"];
        int nrow = dt.Rows.Count;
        int i = 0;
        decimal totalprice = 0;
        while (i < nrow){
            totalprice = totalprice + Convert.ToDecimal(dt.Rows[i]["tprice"].ToString());
            i = i + 1;
        }
        return totalprice;
    }
    protected void GridView1_RowDeleting(object sender, GridViewDeleteEventArgs e){
        DataTable dt = new DataTable();
        dt = (DataTable)Session["buyitems"];
        for (int i = 0; i <= dt.Rows.Count - 1; i++){
            int sr;
            int sr1;
            string qdata;
            string dtdata;
            sr = Convert.ToInt32(dt.Rows[i]["slno"].ToString());
            TableCell cell = GridView1.Rows[e.RowIndex].Cells[0];
            qdata = cell.Text;
            dtdata = sr.ToString();
            sr1 = Convert.ToInt32(qdata);
            if (sr == sr1){
                dt.Rows[i].Delete();
                dt.AcceptChanges();
                break;
            }
        }
        for (int i = 1; i <= dt.Rows.Count; i++){
            dt.Rows[i - 1]["slno"] = i;
            dt.AcceptChanges();
        }
        Session["buyitems"] = dt;
        Response.Redirect("cart.aspx");
    }
    protected void Button3_Click(object sender, EventArgs e){
        if (Session["username"] == null){
            Response.Write("<script> alert('Please sign in to buy items'); </script>");
        }
        else{
            if (GridView1.Rows.Count.ToString() == "0"){
                Response.Write("<script> alert('your cart is empty.you cannot place an
                order'); </script>");
            }
            else{
                DataTable dt;
                dt = (DataTable)Session["buyitems"];
                int i;
                for (i = 0; i <= dt.Rows.Count - 1; i++){
                    SqlConnection scon = new SqlConnection("Data Source=LAPTOP-
                    BJQ4MMQG; Initial Catalog=mdas; Integrated Security=True;");
                    scon.Open();
                    SqlCommand cmd = new SqlCommand("insert into [order] (
                    oid,slno,pid,pname,price,qty,odate,cid) values('" +
                    Session["orderid"] + "','" + dt.Rows[i]["slno"] + "','" +
                    dt.Rows[i]["pid"] + "','" + dt.Rows[i]["pname"] + "','" +
                    dt.Rows[i]["tprice"] + "','" + dt.Rows[i]["pqty"] + "','" +
                    Session["orderdate"] + "','" + Session["username"] + "')",
                    scon);
```

```
                                cmd.ExecuteNonQuery();
                                scon.Close();
                         }
                         i = 0;
                         SqlConnection con = new SqlConnection("Data Source=LAPTOP-BJQ4MMQG;
                         Initial Catalog=mdas; Integrated Security=True;");
                         con.Open();
                         SqlCommand ccmd = new SqlCommand("UPDATE prodadd SET pqtty1 = pqtty1-
                         '" + dt.Rows[i]["pqty"] + "'where pname='" + dt.Rows[i]["pname"] +
                         "'", con);
                         SqlCommand c=new SqlCommand("insert into prodadd (pqtty1) values('0')
                         where pqtty1<0",con);
                         ccmd.ExecuteNonQuery();
                         con.Close();
                         dt.Clear();
                         Response.Redirect("products.aspx?order=done");
                  }
           }
    }
}
```

PRODUCT C#

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Reflection;
public partial class products : System.Web.UI.Page{
    protected void Page_Load(object sender, EventArgs e){
           if(Request.QueryString["order"] != null){
                  Response.Write("<script> alert('Your order has been successfully
                  placed.');</script>");
           }
           if (!Page.IsPostBack){
                  Session["addproduct"] = "false";
                  Label8.Text = Request.QueryString["cat"];
                  if (Request.QueryString["cat"] != null){
                         SqlConnection con = new SqlConnection(@"Data Source=LAPTOP-BJQ4MMQG;
                         Initial Catalog=mdas; Integrated Security=True;");
                         SqlDataAdapter sda = new SqlDataAdapter("select * from prodadd
                         where(pcat like '%" + Label8.Text + "%')", con);
                         DataTable dt = new DataTable();
                         sda.Fill(dt);
                         DataList1.DataSourceID = null;
                         DataList1.DataSource = dt;
                         DataList1.DataBind();
                         ImageButton3.Visible = true;
                         if (DataList1.Items.Count == 0){
                                Response.Write("<script> alert('Selected Category is currently
                                empty '); </script>");
                         }
                  }
                  else{
                         Label8.Text = "ALL";
                         ImageButton3.Visible = false;
                  }
           }
    }
    protected void Timer1_Tick(object sender, EventArgs e){
           Random ran = new Random();
           int i = ran.Next(1, 11);
           Image2.ImageUrl = "~/images/slides/" + i.ToString() + ".jpg";
    }
    protected void DataList1_ItemCommand(object source, DataListCommandEventArgs e){
            Session["addproduct"] = "true";
            if (e.CommandName == "addtocart"){
                  DropDownList list = (DropDownList)(e.Item.FindControl("DropDownList1"));
                  Response.Redirect("cart.aspx?id=" + e.CommandArgument.ToString() + "&qty=" +
                  list.SelectedItem.ToString());
            }
    }
    protected void ImageButton2_Click(object sender, ImageClickEventArgs e){
           SqlConnection con = new SqlConnection(@"Data Source=LAPTOP-BJQ4MMQG; Initial
           Catalog=mdas; Integrated Security=True;");
           SqlDataAdapter sda = new SqlDataAdapter("select * from prodadd where(pname like '%"
           + TextBox2.Text + "%' ) or (pcat like '%" + TextBox2.Text + "%' )", con);
           DataTable dt = new DataTable();
```

```
               sda.Fill(dt);
               DataList1.DataSourceID = null;
               DataList1.DataSource = dt;
               DataList1.DataBind();
               if (DataList1.Items.Count == 0) {
                      Response.Write("<script> alert('no products'); </script>");
               }
        }
        protected void ImageButton3_Click(object sender, ImageClickEventArgs e){
               Response.Redirect("products.aspx");
        }
        protected void ImageButton4_Click(object sender, ImageClickEventArgs e){
               Response.Redirect("Products.aspx?cat=FRUITS");
        }
        protected void ImageButton5_Click(object sender, ImageClickEventArgs e){
               Response.Redirect("Products.aspx?cat=VEGETABLES");
        }
        protected void ImageButton6_Click(object sender, ImageClickEventArgs e){
               Response.Redirect("Products.aspx?cat=BEVERAGES");
        }
        protected void ImageButton7_Click(object sender, ImageClickEventArgs e){
               Response.Redirect("Products.aspx?cat=DIARY");
        }
        protected void ImageButton8_Click(object sender, ImageClickEventArgs e){
               Response.Redirect("Products.aspx?cat=SPICES");
        }
        protected void ImageButton9_Click(object sender, ImageClickEventArgs e){
               Response.Redirect("Products.aspx?cat=DRY FRUITS");
        }
    }
```

USERMASTER C#

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
public partial class usermaster : System.Web.UI.MasterPage{
    protected void Page_Load(object sender, EventArgs e){
           if (Session["username"] != null){
                  Button1.Visible = true;
                  Label1.Text = "Logged in as " + Session["username"].ToString();
           }
           else{
                  Label1.Text = "Guest User";
                  Button1.Visible = false;
           }
    }
    protected void Button1_Click(object sender, EventArgs e){
           Session.Abandon();
           Response.Redirect("login.aspx");
    }
    protected void ImageButton1_Click(object sender, ImageClickEventArgs e){
           Response.Redirect("Default.aspx");
    }
    protected void ImageButton1_Click1(object sender, ImageClickEventArgs e){
           Response.Redirect("cart.aspx");
    }

}
```

USER REMOVE ASPX

```
<%@ Page Title="" Language="C#" MasterPageFile="~/adminmaster.master" AutoEventWireup="true"
CodeFile="usrrem.aspx.cs" Inherits="usrrem" %>
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="Server"></asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="Server">
 <table class="style1">
  <tr>
   <td>
    <asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False"
    DataSourceID="SqlDataSource1" DataKeyNames="email" BorderColor="White" BorderStyle="Ridge"
    BorderWidth="5px" Width="891px" AllowSorting="True" HorizontalAlign="Right"
    CellPadding="4" ForeColor="#333333" GridLines="None">
     <AlternatingRowStyle BackColor="White" />
     <Columns>
      <asp:BoundField DataField="name" HeaderText="NAME" SortExpression="name">
       <HeaderStyle HorizontalAlign="Center" VerticalAlign="Middle" />
       <ItemStyle HorizontalAlign="Left" VerticalAlign="Middle" />
      </asp:BoundField>
      <asp:BoundField DataField="gender" HeaderText="GENDER" SortExpression="gender">
       <HeaderStyle HorizontalAlign="Center" VerticalAlign="Middle" />
       <ItemStyle HorizontalAlign="Left" VerticalAlign="Middle" />
      </asp:BoundField>
      <asp:BoundField DataField="mobile" HeaderText="MOBILE" SortExpression="mobile">
       <HeaderStyle HorizontalAlign="Center" VerticalAlign="Middle" />
       <ItemStyle HorizontalAlign="Left" VerticalAlign="Middle" />
      </asp:BoundField>
      <asp:BoundField DataField="email" HeaderText="E-MAIL" SortExpression="email">
       <HeaderStyle HorizontalAlign="Center" VerticalAlign="Middle" />
       <ItemStyle HorizontalAlign="Left" VerticalAlign="Middle" />
      </asp:BoundField>
      <asp:BoundField DataField="address" HeaderText="ADDRESS" SortExpression="address">
       <HeaderStyle HorizontalAlign="Center" VerticalAlign="Middle" />
       <ItemStyle HorizontalAlign="Left" VerticalAlign="Middle" />
      </asp:BoundField>
      <asp:BoundField DataField="password" HeaderText="PASSWORD" SortExpression="password">
       <HeaderStyle HorizontalAlign="Center" VerticalAlign="Middle" />
       <ItemStyle HorizontalAlign="Left" VerticalAlign="Middle" />
      </asp:BoundField>
      <asp:CommandField ButtonType="Image" ShowDeleteButton="True" DeleteImageUrl ="~/images/
       delete-b.png" DeleteText="">
       <ControlStyle Height="20px" Width="20px" />
      </asp:CommandField>
     </Columns>
     <EditRowStyle BackColor="#2461BF" />
     <FooterStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White" />
     <HeaderStyle BackColor="#507CD1" ForeColor="White" BorderStyle="None" Font-Size="X-Large"
      Font-Bold="True" />
     <PagerStyle BackColor="#2461BF" ForeColor="White" HorizontalAlign="Center" />
     <RowStyle BackColor="#EFF3FB" />
     <SelectedRowStyle BackColor="#D1DDF1" Font-Bold="True" ForeColor="#333333" />
     <SortedAscendingCellStyle BackColor="#F5F7FB" />
     <SortedAscendingHeaderStyle BackColor="#6D95E1" />
     <SortedDescendingCellStyle BackColor="#E9EBEF" />
     <SortedDescendingHeaderStyle BackColor="#4870BE" />
    </asp:GridView>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<%$
     ConnectionStrings:mdasConnectionString %>" SelectCommand="SELECT * FROM [logup]"
     DeleteCommand="DELETE From [logup] WHERE [email] = @email">
    <DeleteParameters>
     <asp:Parameter Name="pid" Type="String" />
```

```
          </DeleteParameters>
        </asp:SqlDataSource>
      </td>
    </tr>
  </table>
</asp:Content>
```

SIGN UP ASPX

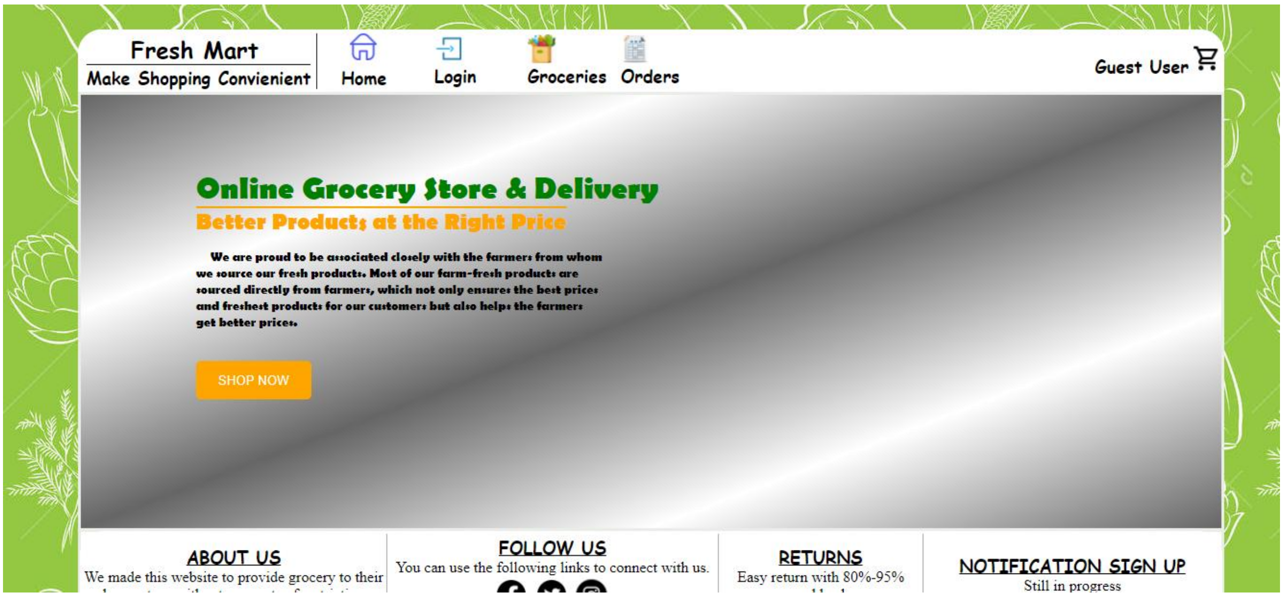
```
<%@ Page Title="" Language="C#" MasterPageFile="~/usermaster.master" AutoEventWireup="true"
CodeFile="signui.aspx.cs" Inherits="signui" %>
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="Server">
 <style type="text/css">
  .style11{
   height: 23px;
  }
  .style12{
   font-size: x-small;
  }
  .style13{
   height: 39px;
  }
  .signui{
   background-color:white;
  }
  .signui:hover{
   color:white;
   background-color:#94c842;
  }
  .style14{
   text-align: right;
   font-size: large;
   font-family: cursive;
  }
  .style15{
   width: 22%;
  }
  .txt{
   border-radius:3px;
   border:1px solid #808080;
  }
  .style16{
   height: 23px;
   width: 574px;
  }
  .style17{
   width: 574px;
 </style>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="Server">
 <div style="font-family: cursive">
  <table class="style1">
   <tr>
    <td style="text-align: right" class="style13">
     <asp:Button ID="Button2" type="button" class="signui" runat="server" Text="Log In"
     OnClick="Button2_Click" CausesValidation="false" Height="35px" Width="150px" style="border-
     radius:20px" BorderStyle="Groove" />
    </td>
    <td class="style13">
     <asp:Button ID="Button3"  runat="server" Text="Sign Up" OnClick="Button3_Click"
     CausesValidation="false" Height="35px" Width="150px" style="border-radius:20px;"
     BorderStyle="Groove" class="signui"/>
    </td>
   </tr>
   <tr>
    <td colspan="2">
     <asp:Panel ID="Panel2" runat="server">
      <div id="in">
```

```
<div>
 <table align="center" class="style1" style="width: 30%">
  <tr>
   <td colspan="3" >
    <asp:Panel ID="Panel4" runat="server">
     <table bgcolor="#FFCDCA" class="style1">
      <tr>
       <td>
        <asp:Image ID="Image1" runat="server" ImageUrl="~/images/cancel-b.png" />
       </td>
       <td>
        Sorry, we couldn&#39;t find an account with that username and password.Please
        try again.
       </td>
      </tr>
     </table>
    </asp:Panel>
   </td>
  </tr>
  <tr>
   <td class="style10">
    <table class="style1" style="padding: 5px; border-style: groove; border-color:
    #C0C0C0; background-color: #FFFFFF; border-radius: 8px;">
     <tr>
      <td colspan="2" style="border-width: 1px; border-color: #666666; text-align: center;
      border-bottom-style: solid;">
       <asp:Label ID="Label4" runat="server" BorderStyle="None" Font-Bold="True" Font-
        Size="XX-Large" Text="Login" style="text-align: center">
       </asp:Label>
      </td>
     </tr>
     <tr>
      <td class="style7" colspan="2" style="text-align: left; font-size: medium"></td>
     </tr>
     <tr>
      <td class="style7" colspan="2" style="text-align: left; font-size: medium">
       E-Mail ID:<br />
       <asp:TextBox ID="TextBox1" runat="server" BorderColor="Gray" class="txt"
       Height="40px" Width="97%"></asp:TextBox>
      </td>
     </tr>
     <tr>
      <td class="style7" colspan="2" style="text-align: left; font-size: medium">
       Password:<br />
       <asp:TextBox ID="TextBox2" runat="server" Height="40px" TextMode="Password"
       Width="97%" class="txt"></asp:TextBox>
       <asp:HyperLink ID="HyperLink3" runat="server" NavigateUrl="~/frgtnpass.aspx"
       style="text-align: right; font-size: small">Forgot Password?</asp:HyperLink>
      </td>
     </tr>
     <tr>
      <td class="style14" colspan="2">&nbsp;<br /></td>
     </tr>
     <tr>
      <td class="style15">
       &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
       <asp:Button ID="Button1" runat="server" Height="37px" OnClick="Button1_Click"
       Style="font-weight: 700;text-align: center;border-radius:10px" Text="Log In"
       Width="99px" BackColor="#00CC00" BorderStyle="None" ForeColor="White"  />
      </td>
      <td width="24%">
       <asp:LinkButton ID="LinkButton1" runat="server" onclick="LinkButton1_Click"
```

```
           style="font-size: small">No account yet?Start here</asp:LinkButton>
          </td>
         </tr>
        </table>
       </td>
      </tr>
     </table>
    </div>
   </div>
  </asp:Panel>
 </td>
</tr>
<tr>
 <td colspan="2">
  <asp:Panel ID="Panel1" runat="server">
   <div>
    <div class="content" style="text-align: center;">
     <center>
      <table>
       <tr>
        <td class="style17">
         <table class="style1" style="padding: 5px; border-style: groove; border-color:#C0C0C0
         ; background-color: #FFFFFF;border-radius:8px;">
          <tr>
           <td style="border-width: 1px; border-color:#666666;text-align: left; border-bottom-
           style: solid;"colspan="3">
            <asp:Label ID="Label5" runat="server" BorderStyle="None" Font-Bold="True" Font-
            Size="XX-Large" Text="Sign-Up"></asp:Label>
            <br />
            <asp:Label ID="Label14" runat="server" Text="Its quick and easy."style=" color:
            #666666"></asp:Label>
           </td>
          </tr>
          <tr>
           <td colspan="3" style="text-align: left;"> &nbsp;</td>
          </tr>
          <tr>
            <td style="vertical-align: bottom"width="58%" align="left">
             Full Name<br />
             <asp:TextBox ID="TextBox8" runat="server" Height="30px" placeholder="Ex:Bruce
             Wayne" Style="text-align: left;border:none;border-radius:3px;border:1px solid
             #808080;" Width="97%" BackColor="#EAEDE8" ForeColor="Black"
             ontextchanged="TextBox8_TextChanged" ></asp:TextBox>
            </td>
            <td style="vertical-align: bottom" width="20%" align="left">
             Age<br />
             <asp:TextBox ID="TextBox9" runat="server" Height="30px" placeholder="10-99"
             Style="text-align: left;border:none;border-radius:3px;border:1px solid #808080;"
             Width="97%" BackColor="#EAEDE8" ForeColor="Black"></asp:TextBox>
            </td>
            <td style="vertical-align: bottom" width="20%" align="left">
             Gender<br />
             <asp:DropDownList ID="DropDownList1" runat="server"  Style="text-align:
             left;border:none;border-radius:3px;border:1px solid #808080;" Width="97%"
             Height="30px" BackColor="#EAEDE8">
              <asp:ListItem>Male</asp:ListItem>
              <asp:ListItem>Female</asp:ListItem>
              <asp:ListItem>Others</asp:ListItem>
             </asp:DropDownList>
            </td>
           </tr>
           <tr>
            <td colspan="3">
             <asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"
             ControlToValidate="TextBox8" CssClass="style18" ErrorMessage="*Name cannot be
             empty" ForeColor="Red" style="font-size: small"></asp:RequiredFieldValidator>
             <asp:RequiredFieldValidator ID="RequiredFieldValidator2" runat="server"
             ControlToValidate="TextBox9" CssClass="style18" ErrorMessage="*Age field cannot
             be empty" ForeColor="Red" style="font-size: small"></asp:RequiredFieldValidator>
             <asp:RegularExpressionValidator ID="RegularExpressionValidator2" runat="server"
             ControlToValidate="TextBox9" CssClass="style18" ErrorMessage="*Invalid age"
             ForeColor="Red" ValidationExpression="[0-9]{2}" style="font-size:
             small"></asp:RegularExpressionValidator>
            </td>
           </tr>
           <tr>
            <td colspan="3">
             Mobile &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp; &nbsp;&nbsp;&nbsp;
             &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
             &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;  &nbsp;&nbsp;
             &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;E-Mail
             <br />
             <asp:TextBox ID="TextBox3" runat="server" Height="30px" placeholder="10 digits"
             style="border:none;border-radius:3px;border:1px solid #808080;" Width="48%"
             BackColor="#EAEDE8" ForeColor="Black"></asp:TextBox>
             <asp:TextBox ID="TextBox4" runat="server" Height="30px" placeholder=
             "Ex:myname123@gmail.com" style="border:none;border-radius:3px;border:1px solid
             #808080;" Width="48%" BackColor="#EAEDE8" ForeColor="Black"></asp:TextBox>
             <br />
            </td>
           </tr>
           <tr>
            <td colspan="3">
             <asp:RequiredFieldValidator ID="RequiredFieldValidator4" runat="server"
             ControlToValidate="TextBox3" CssClass="style18" ErrorMessage="*Mobile field
             cannot be empty" ForeColor="Red" style="font-size: small">
             </asp:RequiredFieldValidator>
             <asp:RegularExpressionValidator ID="RegularExpressionValidator1" runat="server"
             ControlToValidate="TextBox3" CssClass="style18" ErrorMessage="*Invalid number"
             ForeColor="Red" style="font-size: small" ValidationExpression="[0-9] {10} " >
             </asp:RegularExpressionValidator>&nbsp;
             <asp:RequiredFieldValidator ID="RequiredFieldValidator5" runat="server"
             ControlToValidate ="TextBox4" CssClass="style18" ErrorMessage="*Email cannot be
             empty" ForeColor="Red" style="font-size: small"></asp:RequiredFieldValidator>
            </td>
           </tr>
           <tr>
            <td colspan="3">
             Residential Address<br />
             <asp:TextBox ID="TextBox7" runat="server" Height="45px" placeholder = "#, Street,
             Area,Region,City,Pincode" style="border:none;border-radius:3px;border:1px solid
             #808080;" TextMode="MultiLine" Width="97%" BackColor="#EAEDE8"ForeColor="Black" >
             </asp:TextBox>
             <br />
             <asp:RequiredFieldValidator ID="RequiredFieldValidator6" runat="server" = "
             TextBox7" CssClass="style18" ErrorMessage="*This field cannot be empty" Fore
             Color = "Red" style="font-size:small"></asp:RequiredFieldValidator>
            </td>
           </tr>
          <tr>
           <td colspan="3">
            Password<br />
            <asp:TextBox ID="TextBox5" runat="server" Height="30px" placeholder="Enter
            Password" style="border:none;border-radius:3px;border:1px solid #808080;"
```

```
               TextMode="Password" Width="48%" BackColor="#EAEDE8"
               ForeColor="Black"></asp:TextBox>
               <asp:TextBox ID="TextBox6" runat="server" Height="30px" placeholder="Confirm
               Password" style="border:none;border-radius:3px;border:1px solid #808080;"
               TextMode="Password" Width="48%" BackColor="#EAEDE8" ForeColor= "Black" >
               </asp:TextBox>
              </td>
             </tr>
             <tr>
              <td colspan="3">
               <asp:RequiredFieldValidator ID="RequiredFieldValidator7" runat="server"
               ControlToValidate="TextBox5" CssClass="style18" ErrorMessage="*This field cannot
               be empty" ForeColor="Red" style="font-size:small"></asp:RequiredFieldValidator>
               <asp:CompareValidator ID="CompareValidator1" runat="server"
               ControlToCompare="TextBox5" ControlToValidate="TextBox6" CssClass="style18"
               ErrorMessage="*Password doesn't match" ForeColor="Red" style="font-size:
               small"></asp:CompareValidator>
              </td>
             </tr>
             <tr>
              <td colspan="3" style="text-align: left; color: #666666">
               <span class="style12">
                By clicking Sign Up,you agree to our Terms.Learn how we collect ,use and share
                your data in our Data Policy
               </span>
              </td>
             </tr>
             <tr>
              <td colspan="3">&nbsp;</td>
             </tr>
             <tr>
              <td colspan="3">
               <center>
                <asp:Button ID="Button4" runat="server" BorderStyle="none" Height="35px"
                OnClick="Button4_Click" Style="font-weight: 700; text-align: center;border-
                radius:10px; font-family: cursive;" Text="Sign Up" Width="175px"
                BackColor="#00CC00" ForeColor="White" />
               </center>
              </td>
             </tr>
            </table>
           </td>
          </tr>
         </table>
        </center>
       </div>
      </div>
     </asp:Panel>
    </td>
   </tr>
  </table>
 </div>
</asp:Content>
```

8.SCREENSHOTS



Online Grocery Store & Delivery

Better Products at the Right Price

We are proud to be associated closely with the farmers from whom we source our fresh products. Most of our farm-fresh products are sourced directly from farmers, which not only ensures the best prices and freshest products for our customers but also helps the farmers get better prices.

ABOUT US

We made this website to provide grocery to their

FOLLOW US

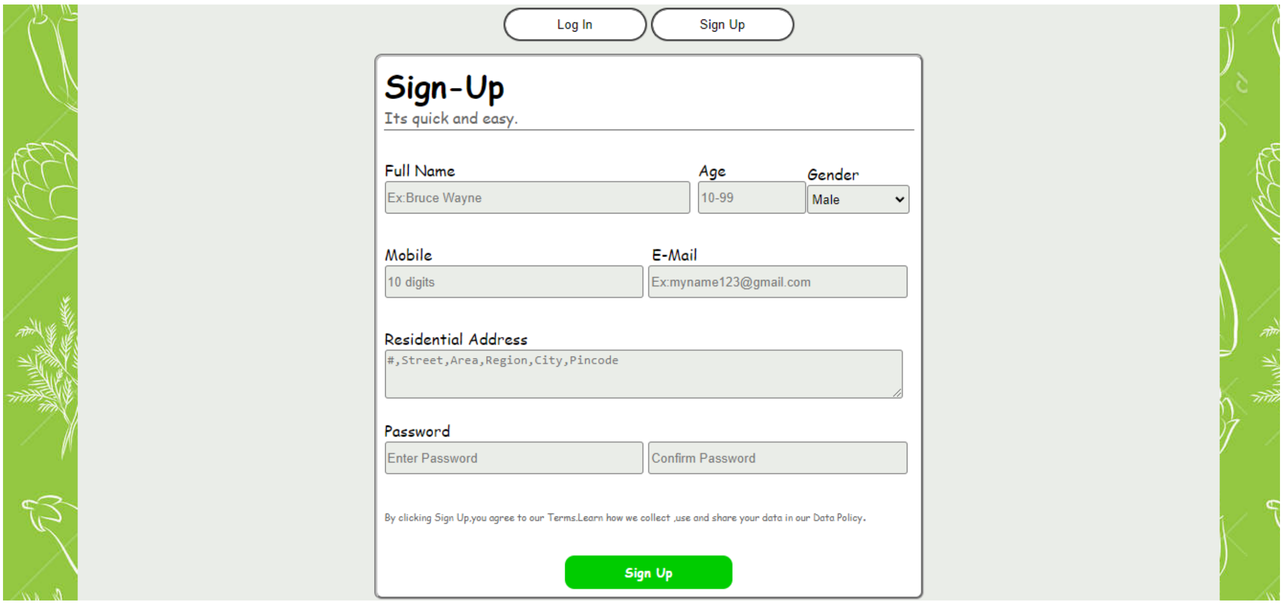
You can use the following links to connect with us.

RETURNS

Easy return with 80%-95%

NOTIFICATION SIGN UP

Still in progress



Sign-Up

Its quick and easy.

Full Name: Ex:Bruce Wayne

Age: 10-99

Gender: Male

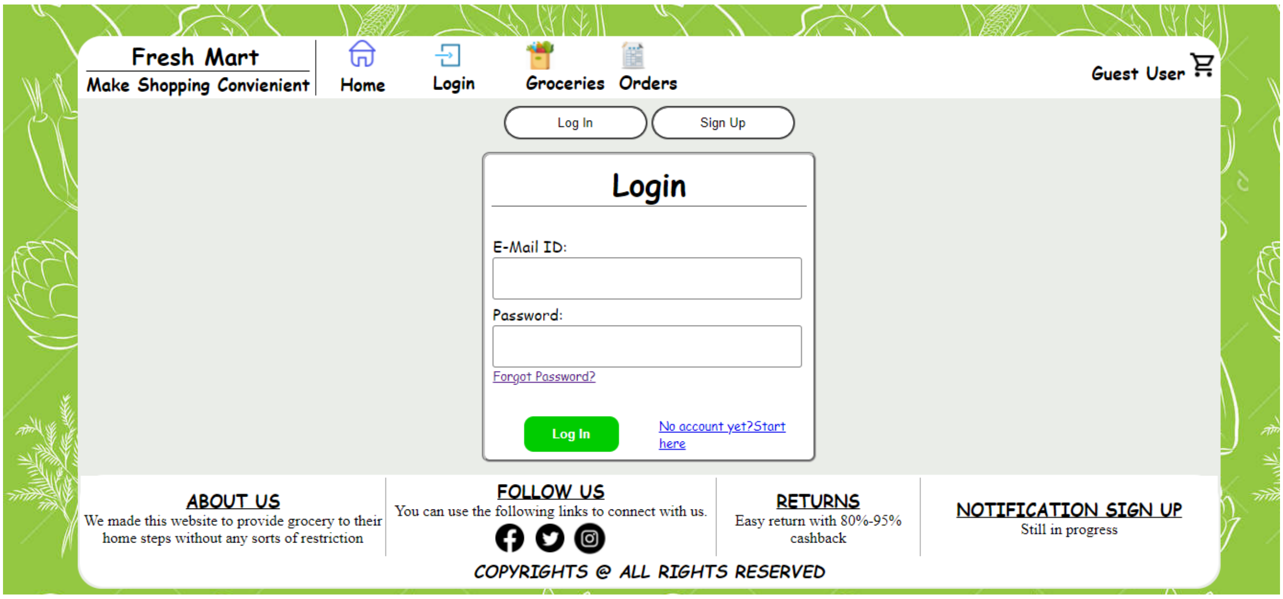
Mobile: 10 digits

E-Mail: Ex:myname123@gmail.com

Residential Address: #,Street,Area,Region,City,Pincode

Password: Enter Password / Confirm Password

By clicking Sign Up,you agree to our Terms.Learn how we collect ,use and share your data in our Data Policy.



Login

E-Mail ID:

Password:

Forgot Password?

No account yet?Start here

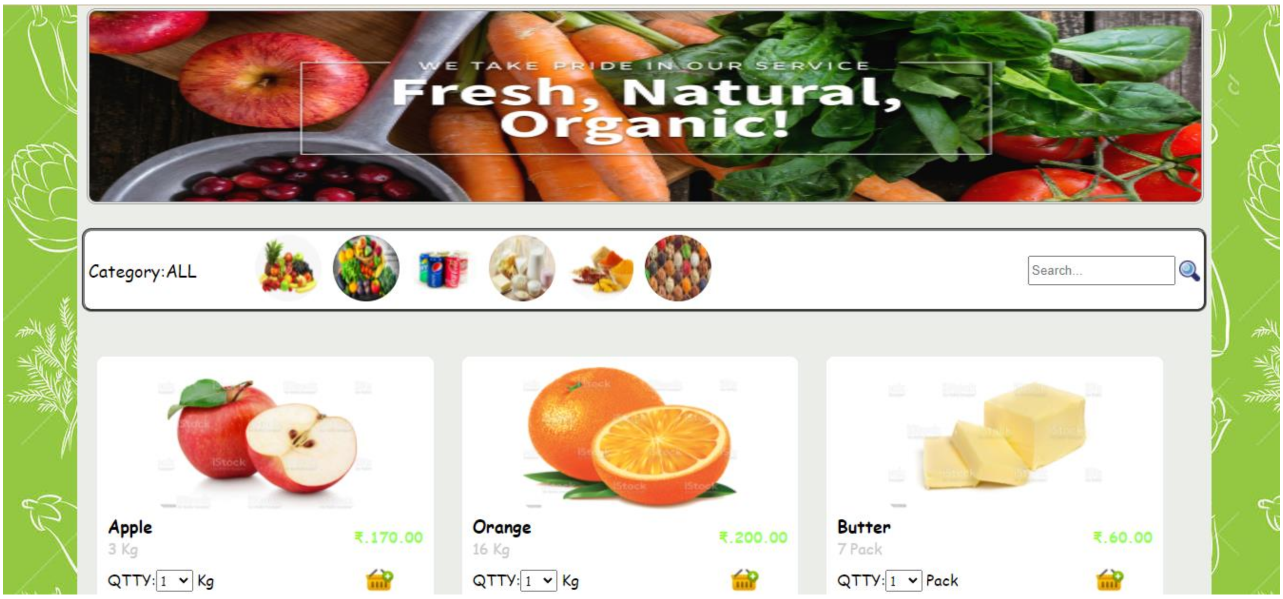
ABOUT US
We made this website to provide grocery to their home steps without any sorts of restriction

FOLLOW US
You can use the following links to connect with us.

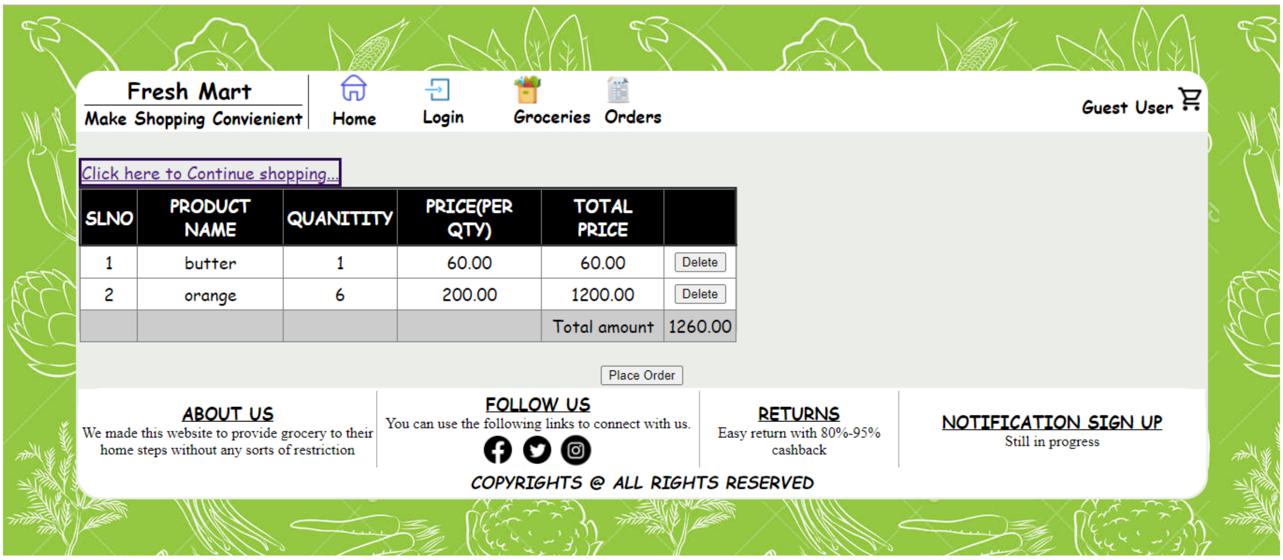
RETURNS
Easy return with 80%-95% cashback

NOTIFICATION SIGN UP
Still in progress

COPYRIGHTS @ ALL RIGHTS RESERVED



Apple 3 Kg ₹.170.00 QTTY: 1 Kg	Orange 16 Kg ₹.200.00 QTTY: 1 Kg	Butter 7 Pack ₹.60.00 QTTY: 1 Pack



SLNO	PRODUCT NAME	QUANTITY	PRICE(PER QTY)	TOTAL PRICE	
1	butter	1	60.00	60.00	Delete
2	orange	6	200.00	1200.00	Delete
				Total amount	1260.00

ABOUT US

We made this website to provide grocery to their home steps without any sorts of restriction

FOLLOW US

You can use the following links to connect with us.

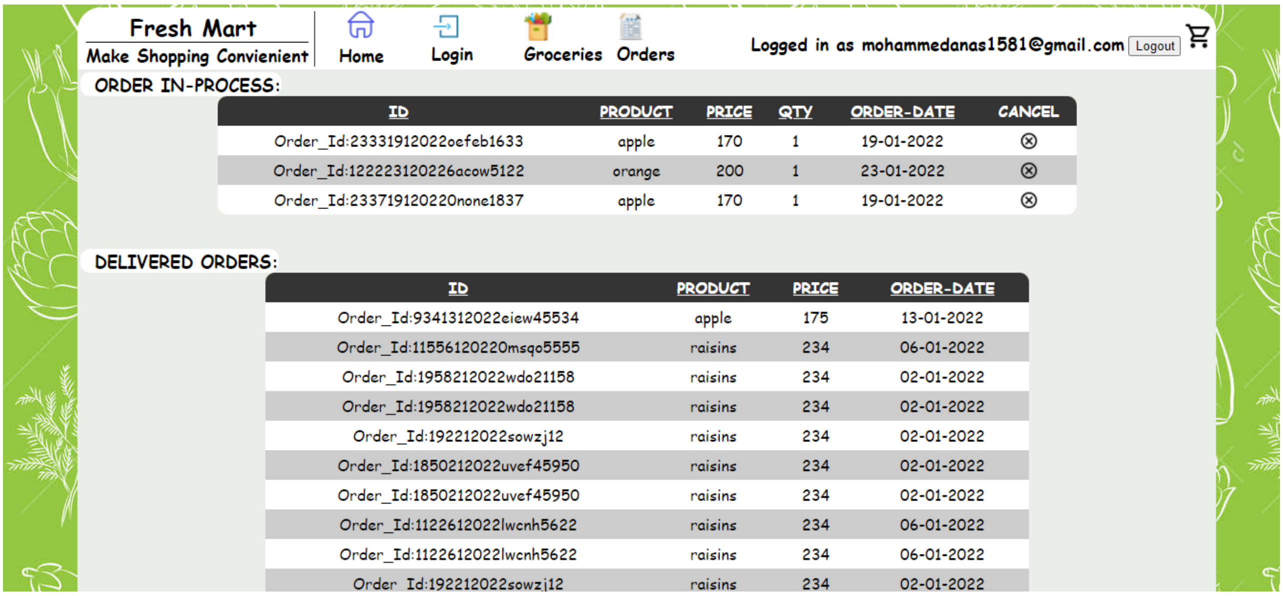
RETURNS

Easy return with 80%-95% cashback

NOTIFICATION SIGN UP

Still in progress

COPYRIGHTS @ ALL RIGHTS RESERVED



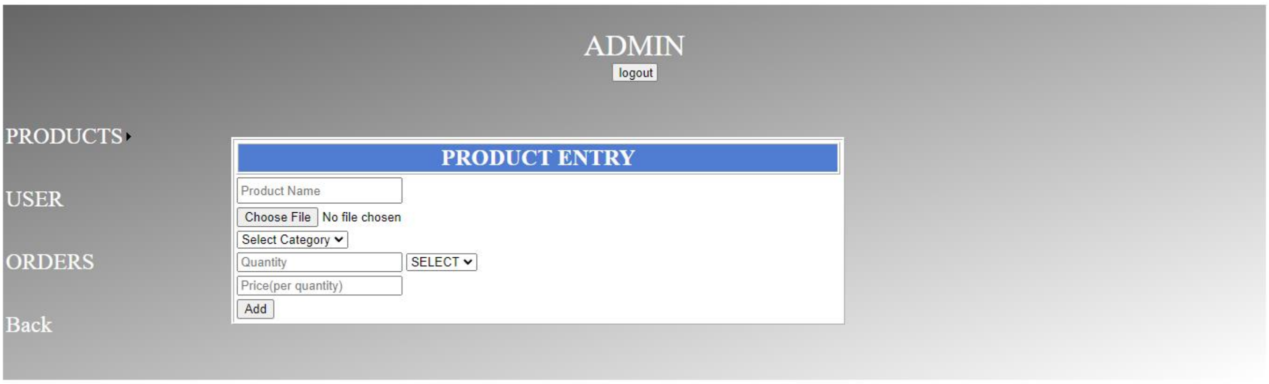
Logged in as mohammedanas1581@gmail.com

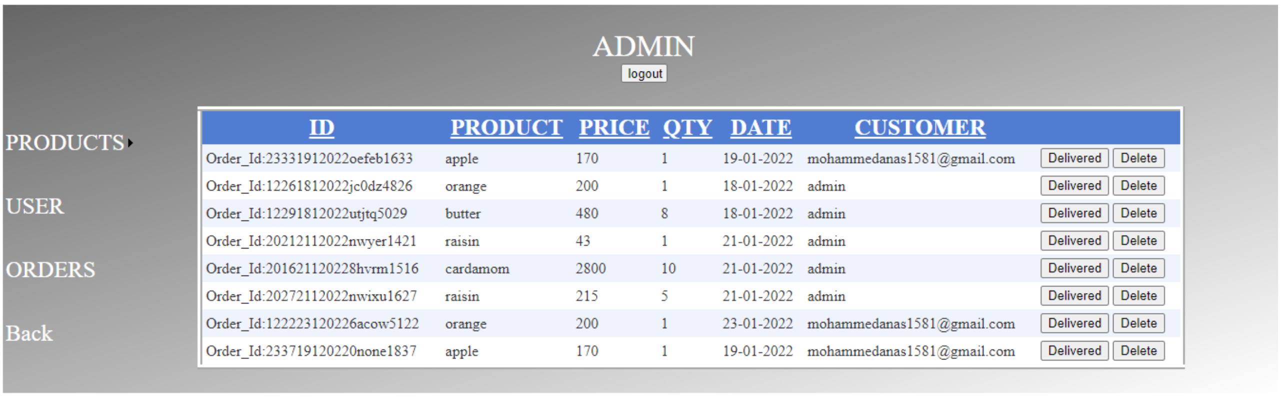
ORDER IN-PROCESS:

ID	PRODUCT	PRICE	QTY	ORDER-DATE	CANCEL
Order_Id:23331912022oefeb1633	apple	170	1	19-01-2022	⊗
Order_Id:12222312022 6acow5122	orange	200	1	23-01-2022	⊗
Order_Id:233719120220none1837	apple	170	1	19-01-2022	⊗

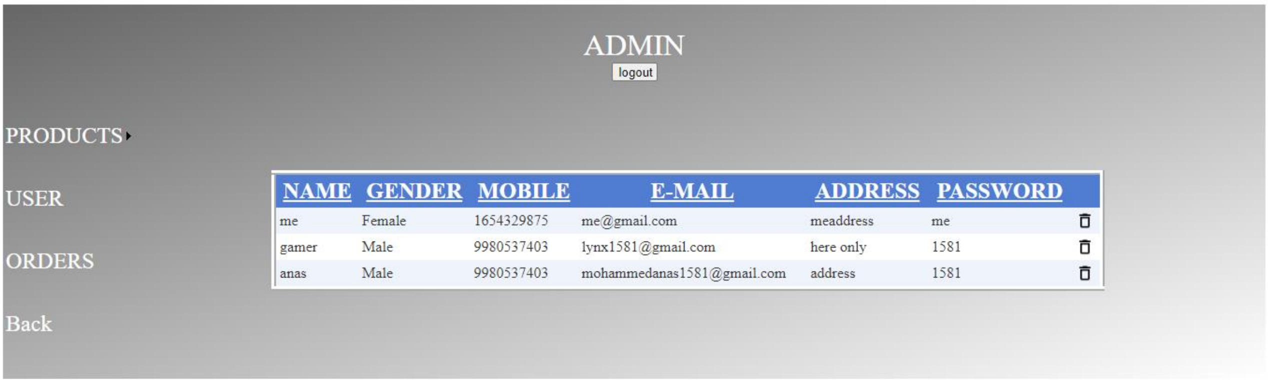
DELIVERED ORDERS:

ID	PRODUCT	PRICE	ORDER-DATE
Order_Id:9341312022eiew45534	apple	175	13-01-2022
Order_Id:11556120220msqo5555	raisins	234	06-01-2022
Order_Id:1958212022wdo21158	raisins	234	02-01-2022
Order_Id:1958212022wdo21158	raisins	234	02-01-2022
Order_Id:192212022sowzj12	raisins	234	02-01-2022
Order_Id:1850212022uvef45950	raisins	234	02-01-2022
Order_Id:1850212022uvef45950	raisins	234	02-01-2022
Order_Id:1122612022lwcnh5622	raisins	234	06-01-2022
Order_Id:1122612022lwcnh5622	raisins	234	06-01-2022
Order Id:192212022sowzj12	raisins	234	02-01-2022

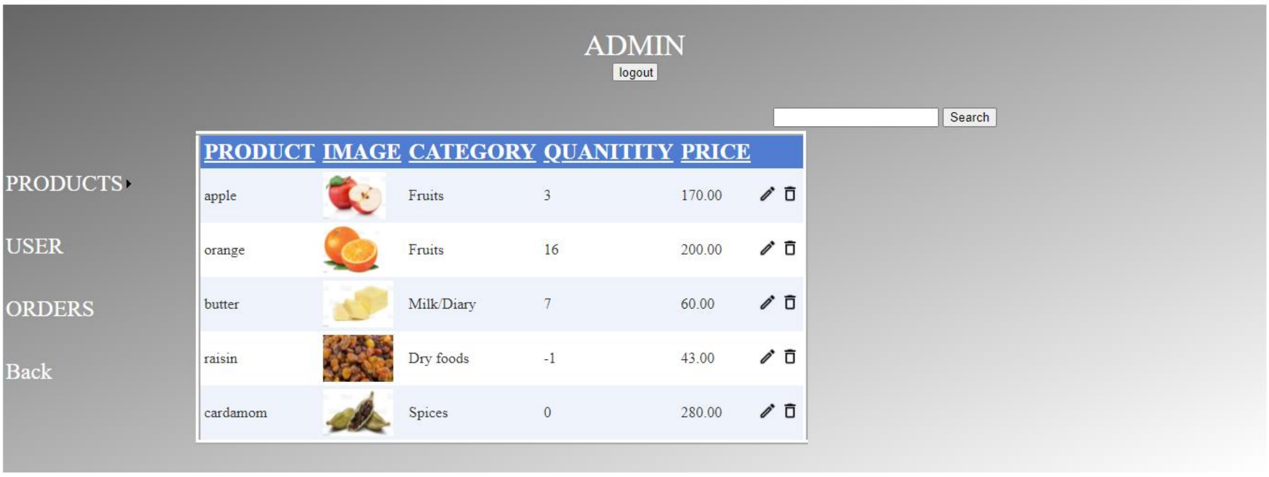




ID	PRODUCT	PRICE	QTY	DATE	CUSTOMER	
Order_Id:23331912022oefeb1633	apple	170	1	19-01-2022	mohammedanas1581@gmail.com	Delivered Delete
Order_Id:12261812022jc0dz4826	orange	200	1	18-01-2022	admin	Delivered Delete
Order_Id:12291812022utjtq5029	butter	480	8	18-01-2022	admin	Delivered Delete
Order_Id:20212112022nwyer1421	raisin	43	1	21-01-2022	admin	Delivered Delete
Order_Id:201621120228hvrm1516	cardamom	2800	10	21-01-2022	admin	Delivered Delete
Order_Id:20272112022nwixu1627	raisin	215	5	21-01-2022	admin	Delivered Delete
Order_Id:122223120226acow5122	orange	200	1	23-01-2022	mohammedanas1581@gmail.com	Delivered Delete
Order_Id:233719120220none1837	apple	170	1	19-01-2022	mohammedanas1581@gmail.com	Delivered Delete



NAME	GENDER	MOBILE	E-MAIL	ADDRESS	PASSWORD	
me	Female	1654329875	me@gmail.com	meaddress	me	🗑
gamer	Male	9980537403	lynx1581@gmail.com	here only	1581	🗑
anas	Male	9980537403	mohammedanas1581@gmail.com	address	1581	🗑



ADMIN

PRODUCT	IMAGE	CATEGORY	QUANITITY	PRICE
apple		Fruits	3	170.00
orange		Fruits	16	200.00
butter		Milk/Diary	7	60.00
raisin		Dry foods	-1	43.00
cardamom		Spices	0	280.00

9.TESTING

INTRODUCTION

Testing is the process of executing a program with the aim of finding errors. To make our software perform well it should be error-free. If testing is done successfully it will remove all the errors from the software.

PRINCIPLES OF TESTING

i. All the test should meet the customer requirements

ii. To make our software testing should be performed by a third party

iii. Exhaustive testing is not possible. As we need the optimal amount of testing based on the risk assessment of the application.

iv. All the test to be conducted should be planned before implementing it

v. It follows the Pareto rule (80/20 rule) which states that 80% of errors come from 20% of program components.

vi. Start testing with small parts and extend it to large parts.

TYPES OF TESTING

1. Unit Testing

It focuses on the smallest unit of software design. In this, we test an individual unit or group of interrelated units. It is often done by the programmer by using sample input and observing its corresponding outputs.

2. Integration Testing

The objective is to take unit tested components and build a program structure that has been dictated by design. Integration testing is testing in which a group of components is combined to produce output.

Integration testing is of four types: (i) Top-down (ii) Bottom-up (iii) Sandwich (iv) Big-Bang

3. Regression Testing

Every time a new module is added leads to changes in the program. This type of testing makes sure that the whole component works properly even after adding components to the complete program.

4. Smoke Testing

This test is done to make sure that software under testing is ready or stable for further testing
It is called a smoke test as the testing an initial pass is done to check if it did not catch the fire or smoke in the initial switch on.

5. Alpha Testing

This is a type of validation testing. It is a type of acceptance testing which is done before the product is released to customers. It is typically done by QA people.

6. Beta Testing

The beta test is conducted at one or more customer sites by the end-user of the software. This version is released for a limited number of users for testing in a real-time environment

7. System Testing

This software is tested such that it works fine for the different operating systems. It is covered under the black box testing technique. In this, we just focus on the required input and output without focusing on internal working.
In this, we have security testing, recovery testing, stress testing, and performance testing

8. Stress Testing

In this, we give unfavorable conditions to the system and check how they perform in those conditions.

9. Performance Testing

It is designed to test the run-time performance of software within the context of an integrated system. It is used to test the speed and effectiveness of the program. It is also called load testing. In it we check, what is the performance of the system in the given load.

10. Object-Oriented Testing

This testing is a combination of various testing techniques that help to verify and validate object-oriented software. This testing is done in the following manner:

- Testing of Requirements,
- Design and Analysis of Testing,
- Testing of Code,
- Integration testing,
- System testing,
- User Testing.

TEST CASES

SIGN IN PAGE

SLNO	Action	Validation	Expected result	Outcome
1	Sign in Button	Correct username Correct password	Session Start Redirect Product page	True
2		Correct username Incorrect password	Error message panel	True
3		Incorrect username Incorrect password	Error message panel	True
4		Incorrect username Correct password	Error message panel	True
5		Username as ‘admin’ Password as ‘admin’	Admin session starts Redirect admin page	True

SIGN UP PAGE

SLNO	Action	Validation	Expected result	Outcome
1	Sign Up Button	New email Correct format Click signup	Data insert Redirect login	True
2		Existing email Correct format Click signup	Error message ‘email exists’	True
3		New email Incorrect format Click signup	Input validation error	True
4		Existing email Incorrect format Click signup	Error message ‘email exists’	True

MASTER PAGE

SLNO	Action	Validation	Expected result	Outcome
1	Page Load	Logged in	User name visible Logout button visible	True
2		Not logged in	User name as ‘Guest’ Logout button invisible	True
3	Logout Button		Session abandon Redirect login	

HISTORY PAGE

SLNO	Action	Validation	Expected result	Outcome
1	Page Load	Logged in	Load history page	True
2		Not logged in	Redirect login Error message ‘login first’	True

COMMENT PAGE

SLNO	Action	Validation	Expected result	Outcome
1	Comment Button	Logged in	Data insert Refresh page Comment display	True
2		Not logged in	Error message ‘login first’	True

CART PAGE

SLNO	Action	Validation	Expected result	Outcome
1	Page Load	Empty cart	Message ‘cart is empty’ Order button disabled	True
2		Item present in cart	Order button enabled	True
3	Order Button	Not logged in	Error message ‘login first’	True
4		Logged in Empty cart	Error message ‘cart is empty’	True
5		Logged in Item present in cart	Order places Order Data insert Products data update Redirect products	True
6	Delete Button		Item removed from page	True

10.CONCLUSION

ON A FINAL NOTE:

- This website allows people to make online shopping simple and with privacy.
- We offer simpler and user-friendly website.
- The required specification to run this project is very minimum
- Design used in this website is simple and easy to understand.
- We follow a strict relationship between data in the database.
- Database are simplified to its limit.
- This isn’t the end, there will be future versions to make them more comprehensive.

11.FUTURE ENHANCEMET

FUTURE PROSPECTS:

- Tracking your order through GPS
- Providing strong client identity and authentication
- Make more user friendly for physically challenged.
- AI powered recommendation of products
- Link this website with the online banking facility
- Enhancement from customers recommendation

12.BIBLIOGRAPHY

We were able successfully complete this project only due to ease of information access from the internet. Internet truly connects us together. Following are links listed out which made it possible for our team to complete the project in give time:

- https://www.google.com/
- https://www.youtube.com/playlist?list=PL4HegTSNb5KEbe-1sZ7ONlTbCRub3AGN5
- https://www.youtube.com/
- https://fonts.google.com/
- https://stackoverflow.com/
- https://www.codeproject.com/