AT YOUR DOOR

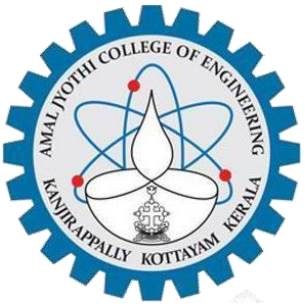
Project Report Submitted By

ANCY ALEXANDER

Reg. No: AJC20MCA-2016

In Partial fulfillment for the Award of the Degree Of

MASTER OF COMPUTER APPLICATIONS (2 Year)
(MCA)
APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY



AMAL JYOTHI COLLEGE OF ENGINEERING

KANJIRAPPALLY

[Affiliated to APJ Abdul Kalam Technological University, Kerala. Approved by AICTE, Accredited by NAAC with 'A' grade. Koovappally, Kanjirappally, Kottayam, Kerala – 686518]

2021-2022

DEPARTMENT OF COMPUTER APPLICATIONS

AMAL JYOTHI COLLEGE OF ENGINEERING

KANJIRAPPALLY

CERTIFICATE

This is to certify that the Project report, "**AT YOUR DOOR"** is the bonafide workof **ANCY ALEXANDER (Reg.No:AJC20MCA-2016)** in partial fulfillmentof the requirements for the award of the Degree of Master of Computer Applications under APJ Abdul Kalam Technological Universityduring the year 2021-2022.

Ms. Grace Joseph
Internal Guide

Rev.Fr.Dr.Rubin Thottupurathu Jose
Coordinator

Rev.Fr.Dr.Rubin Thottupurathu Jose

Head of the Department

DECLARATION

I hereby declare that the project report **"AT YOUR DOOR"** is a bonafided work done at Amal Jyothi College of Engineering, towards the partial fulfilment of the requirements for the award of the Degree of Master of Computer Applications (MCA) from APJ Abdul Kalam Technological University, during the academic year 2021-2022.

Date: **ANCY ALEXANDER**

KANJIRAPPALLY **Reg. No: AJC20MCA-2016**

ACKNOWLEDGEMENT

First and foremost, I thank God almighty for his eternal love and protection throughout the project. I take this opportunity to express my gratitude to all who helped me in completing this project successfully. It has been said that gratitude is the memory of the heart. I wish to express my sincere gratitude to our manager **Rev. Fr. Dr. Mathew Paikatt** and Principal **Dr. Lillykutty Jacob** for providing good faculty for guidance.

I owe a great depth of gratitude towards our Head of the Department **Rev.Fr.Dr.Rubin Thottupurathu Jose** for helping us. I extend my whole hearted thanks to the project coordinators **Rev.Fr.Dr.Rubin Thottupurathu Jose** and **Ms**. **Nimmy Francis** for their valuable suggestions and for overwhelming concern and guidance from the beginning to the end of the project. I would also like to express sincere gratitude to my guide**, Ms.Grace Joseph** for his inspiration and helping hand.

I thank our beloved teachers for their cooperation and suggestions that helped me throughout the project. I express my thanks to all my friends and classmates for their interest, dedication, and encouragement shown towards the project. I convey my hearty thanks to my family for the moral support, suggestions, and encouragement to make this venture a success.

ANCY ALEXANDER

ABSTRACT

Here we are introducing an application that works within a centralized network. This project presents a review on the software program “Online grocery store” as should be used in an Online shopping system like Flipkart and Amazon but for a local service, which is used to buy grocery items like fresh vegetables, fresh fruits, seeds and daily essentials, etc. Users can buy products from the website and can make payment from the website itself. Dealers handle daily essentials products.

Here the admin will be adding a seller who has all access to the application. He will be providing the ordered products to the delivery boy, and he will deliver it to the users. Users can verify their items and can buy from delivery boys. They can raise complaints if there are any issues in the delivery or for the product. They can also send feedback to the admin about the application or services. Products on the website will be categorized by its type and it can be rated as it’s purity. Some products like most selling items, useful items, etc. will be categorized differently. You can also keep products in the cart to buy those items after some time.

The main features of this projects are grocery suggestion where we can add things without searching, next feature is to set ‘Delivery extension date’ and can also add alarm notification. In the case of emergency, we can set emergency tab which provide fast delivery of products. A budget amount can be added, and it also provide online delivery of seeds.

CONTENT

List of Abbreviation

IDE -	Integrated Development Environment
HTML -	Hyper Text Markup Language.
CSS -	Cascading Style Sheet
SQL -	Structured Query Language
UML -	Unified Modeling Language.
PHP -	Hyper Text Preprocessor

CHAPTER 1

INTRODUCTION

1.1 PROJECT OVERVIEW

Whether you're stuck at home, too busy to shop, or want to save time, it's convenient to shop online for groceries**.** If you're looking for farm-fresh fruits and vegetables, world-class breads, top quality pulses and food grains, dairy products, and eggs, At your Door has you covered everything you need at a great price. In past of time today everyone is squeezed for time ,people want to easily shopping without going to shop. This site gives all the information about the grocery shop provides better service for customers .it provides the facility to the customer who wants to buy grocery products in short time.it provides full details about the grocery products and related information like cost, weight and best before date.

In the formation of stronger relationships with customers and delivering of a high level of service and support, which in turn improves organization sales and its goodwill. This website keeps the data in a centralized way which is available to all the users at the same time. It manages historical data in database.

New thing that included in this project are grocery suggestion where we can add things without searching, next feature is to set 'Delivery extension date' and can also add alarm notification. In the case of emergency, we can set emergency tab which provide fast delivery of products. A budget amount can be added, and it also provide online delivery of seeds.

1.2 PROJECT SPECIFICATION

On our website we introduce online shopping for the grocery items from our locality. You can buy products from the website and can also pay money online itself . Queries will be solved from here. Users feedback will be taken so improving the website will be helpful. Grocery sellers are selling products directly to dealer, so the intermediate charges will be avoided, which will help the users to buy products with low cost. Products will be delivered to your house within a time. This application has all these features included: buying groceries like items, barter , renting of reusable products and ordering food without waiting.

The system includes 5 modules. They are:

1. **Admin Module**

   Admin must have a login into this system. He has the overall control of the system. Admin can add or update service details, manage user data etc. Admin can View all the registered users and also manage all his data.Admin add dealers and he can manage dealer

2. **Customer Module**

   Customer can register and they can order products and also view also information about his/her products. And view their order.

3. **Dealer Module**

   Admin can add dealers and the dealer add products and view all the information about customers. dealer can accept the user approvel/rejection.

4. **Seed Providing**

   This module is for adding seeds and uploading images and description about seeds

5. **Delivery boy**

   In this module delivery boy has tp view the customers details he can confirm order,update order details. The First and second modules has been presentedin the current semester and the rest of the modules are extended to be completed on the upcoming semester

CHAPTER 2

SYSTEM STUDY

2.1 INTRODUCTION

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information to recommend improvements on the system. It is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minute's detail and analyzed. The system analyst plays the role of the interrogator and dwells deep into the working of the present system. The system is viewed as a whole and the input to the system are identified. The outputs from the organizations are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and decisional variables, analyzing and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action.

A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is loop that ends as soon as the user is satisfied with proposal.

Preliminary study is the process of gathering and interpreting facts, using the information for further studies on the system. Preliminary study is problem solving activity that requires intensive communication between the system users and system developers. It does various feasibility studies. In these studies, a rough figure of the system activities can be obtained, from which the decision about the strategies to be followed for effective system study and analysis can be taken.

2.2 EXISTING SYSTEM

In our existing system, there is no application for buying things online as a local service. We have to visit every shops for a variety of things and sometime we will never satisfies with the collections in the shop, may be the exact product will be not in the shop means we have to visit another shop, which is really a waste of energy and time. If we buys a thing from shop also we don't have a platform or any details of shopkeeper to inform some complaints on their products.

2.3 DRAWBACKS OF EXISTING SYSTEM

- No proper online management of system
- Human effort is needed.
- It is difficult to maintain important information in books.
- More manual hours need to generate required reports.

2.4 PROPOSED SYSTEM

The proposed system is defined to meets all the disadvantages of the existing system. It is necessary to have a system that is more user friendly and user attractive for growth of online grocery shop; on such consideration the system is proposed. In our proposed system there is admin who can view all the customers. It allows customers to make their service booking and do their transactions by using online payment method .Users of this proposed system are admin and customer. The software application which avoids more manual hours that need to spend in record keeping and generating reports. This application keeps the data in a centralized way which is available to all the users simultaneously. It is very easy to manage historical data in database. No specific training is required for the distributors to use this application. They can easily use the tool that decreases manual hours spending for normal things and hence increases the performance. It is very easy to record the information of online sales and purchases in the databases.

2.5 ADVANTAGES OF PROPOSED SYSTEM

The system is very simple in design and to implement. The system requires very low system resources, and the system will work in almost all configurations. It has got following features:

- **Better security: -**

For data to remain secure measures must be taken to prevent unauthorized access. Security means that data are protected from various forms of destruction. The system security problem can be divided into four related issues: security, integrity, privacy and confidentiality. Username and password requirement to sign in ensures security. It will also provide data security as we are using the secured databases for maintaining the documents.

- **Ensure data accuracy: -**

The proposed system eliminates the manual errors while entering the details of the users during the registration.

- **Better service: -**

The product will avoid the burden of hard copy storage. We can also conserve the time and human resources for doing the same task. The data can be maintained for longer period with no loss of data.

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 FEASIBILITY STUDY

Feasibility study is made to see if the project on completion will serve the purpose of the organization for the amount of work, effort and the time that spend on it. Feasibility study lets the developer foresee the future of the project and the usefulness. A feasibility study of a system proposal is according to its workability, which is the impact on the organization, ability to meet their user needs and effective use of resources. Thus, when a new application is proposed it normally goes through a feasibility study before it is approved for development.

The document provides the feasibility of the project that is being designed and lists various areas that were considered very carefully during the feasibility study of this project such as Technical, Economic and Operational feasibilities. The following are its features: -

3.1.1 Economical Feasibility

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

The proposed system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it give an indication of the system is economically possible for development.

The cost of project, DREAMS was divided according to the system used, its development cost and cost for hosting the project. According to all the calculations the project was developed in a low cost. As it is completely developed using open source software.

3.1.2 Technical Feasibility

The system must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input, output, programs and procedures. Having identified an outline system, the investigation must go on to suggest the type of equipment, required method developing the system, of running the system once it has been designed.

Technical issues raised during the investigation are:

- Does the existing technology sufficient for the suggested one?
- Can the system expand if developed?

The project should be developed such that the necessary functions and performance are achieved within the constraints. The project requires High Resolution Scanning device and utilizes Cryptographic techniques. Through the technology may become obsolete after some period of time, due to the fact that newer version of same software supports older versions, the system may still be used. So there are minimal constraints involved with this project. The system has been developed using PHP in front end and MySQL in server in back end, the project is technically feasible for development. The system has been developed using PHP in front end and MySQL in server in back end, the project is technically feasible for development. The System used was also of good performance of Processor Intel i3 core; RAM 4GB and, Hard disk 1TB

3.1.3 Behavioral Feasibility

The proposed system includes the following questions:

- Is there sufficient support for the users?
- Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible

3.1 SYSTEM SPECIFICATION

3.1.4 Hardware Specification

Processor - Intel core i3

RAM - 4 GB

Hard disk - 1 TB

3.1.5 Software Specification

Front End - HTML, CSS

Backend - MYSQL

Client on PC - Windows 7 and above.

Technologies used - JS, HTML5, AJAX, J Query, PHP, CSS

3.2 SOFTWARE DESCRIPTION

3.2.1 PHP

PHP is a server side scripting language designed for web development but also used as a general purpose programming language. PHP is now installed on more than 244 million websites and 2.1 million web servers. Originally created by Rasmus Ledorf in 1995, the reference implementation of PHP is now produced by the PHP group. While PHP originally stood for personal Home page ,it now stands for PHP:HypertextPreprocessor,

a recursive acronym.PHP code is interpreted by a web server with a PHP processor module which generates the resulting web page.PHP commands can be embedded directly into a HTML source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone incompatible with the GNU General Public License (GPL) due to restrictions on the usage of the term PHP.PHP can be deployed on most web servers and also as a standalone shell on almost every operating system and platform, free of charge.

3.2.2 MySQL

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by Oracle Corporation. The MySQL Web site provides the latest information about MySQL software.

- **MySQL is a database management system.**

  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.

- **MySQL databases are relational.**

  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, "SQL92" 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 software is Open Source.**

  Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. The MySQL software uses the GPL (GNU General Public License), to define what you may and may not do with the software in different situations. If you feel uncomfortable with the GPL or need to embed MySQL code into a commercial application, you can buy a commercially licensed version from us. See the MySQL Licensing Overview for more information.

- **The MySQL Database Server is very fast, reliable, scalable, and easy to use.**

  If that is what you are looking for, you should give it a try. 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 Server works in client/server or embedded systems.**

  The MySQL Database Software is a client/server system that consists of a multi-threaded SQL server that supports different backends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs). We also provide MySQL Server as an embedded multi-threaded library that you can link into your application to get a smaller, faster, easier-to-manage standalone product.

CHAPTER 4

SYSTEM DESIGN

4.1 INTRODUCTION

Design is the first step into the development phase for any engineered product or system. Design is a creative process. A good design is the key to effective system. The term “design” is defined as “the process of applying various techniques and principles for the purpose of defining a process or a system in sufficient detail to permit its physical realization”. It may be defined as a process of applying various techniques and principles for the purpose of defining a device, a process or a system in sufficient detail to permit its physical realization. Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm that is used. The system design develops the architectural detail required to build a system or product. As in the case of any systematic approach, this software too has undergone the best possible design phase fine tuning all efficiency, performance and accuracy levels. The design phase is a transition from a user-oriented document to a document to the programmers or database personnel. System design goes through two phases of development: Logical and Physical Design.

4.2 UML DIAGRAM

UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems. UML was created by the Object Management Group (OMG) and UML 1.0 specification draft was proposed to the OMG in January 1997.

UML stands for **Unified Modeling 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 modeling 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. UML is not a programming language but tools can be used to generate code in various languages using UML diagrams. UML has a direct relation with object oriented analysis and design. After some standardization, UML has

become an OMG standard. All the elements, relationships are used to make a complete UML diagram and the diagram represents a system. The visual effect of the UML diagram is the most important part of the entire process. All the other elements are used to make it complete. UML includes the following nine diagrams.

- Class diagram
- Object diagram
- Use case diagram
- Sequence diagram
- Collaboration diagram
- Activity diagram
- Statechart diagram
- Deployment diagram
- Component diagram

4.2.1 USE CASE DIAGRAM

A use case diagram is a graphic depiction of the interactions among the elements of a system. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system" refers to something being developed or operated, such as a mail-order product sales and service Web site. Use case diagrams are employed in UML (Unified Modeling Language), a standard notation for the modeling of real-world objects and systems.

System objectives can include planning overall requirements, validating a hardware design, testing and debugging a software product under development, creating an online help reference, or performing a consumer-service-oriented task. For example, use cases in a product sales environment would include item ordering, catalog updating, payment processing, and customer relations. A use case diagram contains four components.

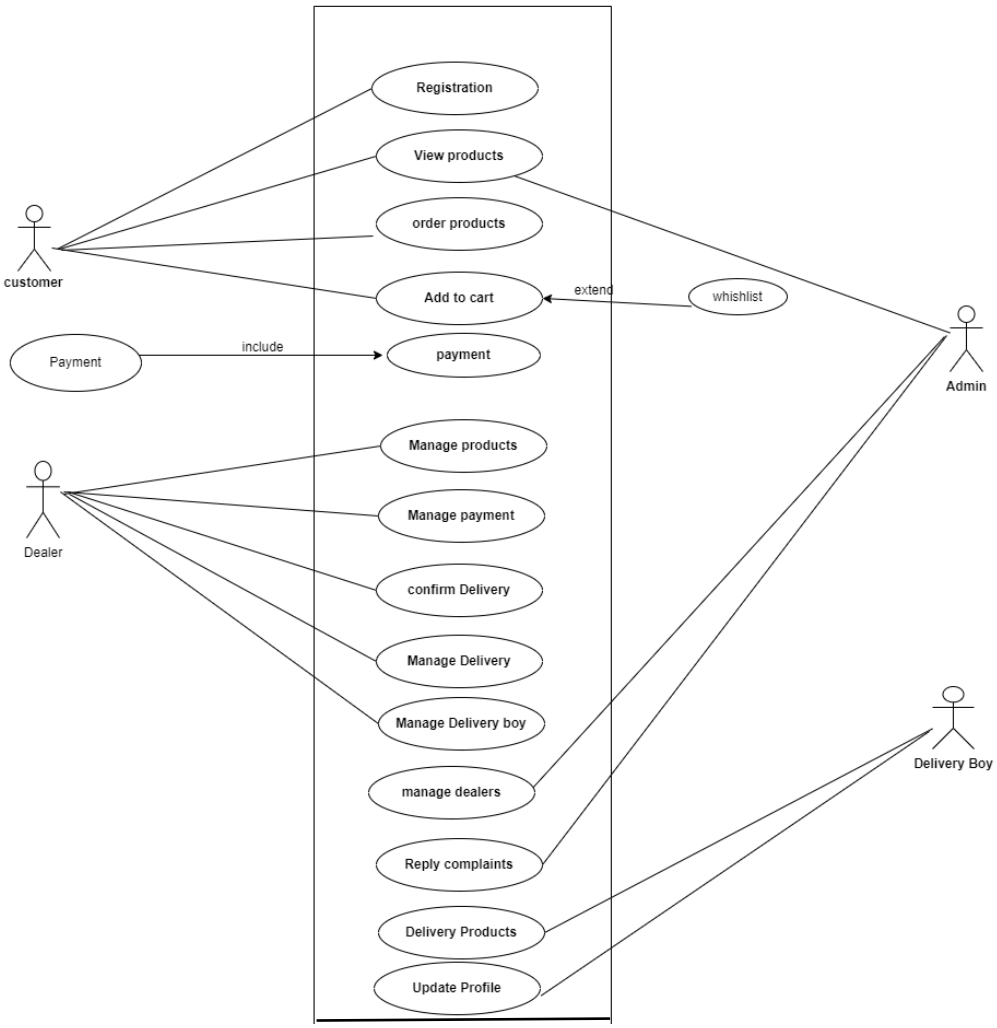
- The boundary, which defines the system of interest in relation to the world around it.
- The actors, usually individuals involved with the system defined according to their roles.

- The use cases, which are the specific roles are played by the actors within and around the system.
- The relationships between and among the actors and the use cases.

Use case diagrams are drawn to capture the functional requirements of a system. After identifying the above items, we have to use the following guidelines to draw an efficient use case diagram

- The name of a use case is very important. The name should be chosen in such a way so that it can identify the functionalities performed.
- Give a suitable name for actors.
- Show relationships and dependencies clearly in the diagram.
- Do not try to include all types of relationships, as the main purpose of the diagram is to identify the requirements.
- Use notes whenever required to clarify some important points.

Use case Daigram



4.2.2 SEQUENCE DIAGRAM

A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place. We can also use the terms event diagrams or event scenarios to refer to a sequence diagram. Sequence diagrams describe how and in what order the objects in a system function. These diagrams are widely used by businessmen and software developers to document and understand requirements for new and existing systems.

Sequence Diagram Notations –

i. **Actors –** An actor in a UML diagram represents a type of role where it interacts with the system and its objects. It is important to note here that an actor is always outside the scope of the system we aim to model using the UML diagram. We use actors to depict various roles including human users and other external subjects. We represent an actor in a UML diagram using a stick person notation. We can have multiple actors in a sequence diagram.

ii. **Lifelines –** A lifeline is a named element which depicts an individual participant in a sequence diagram. So basically each instance in a sequence diagram is represented by a lifeline. Lifeline elements are located at the top in a sequence diagram

iii. **Messages –** Communication between objects is depicted using messages. The messages appear in a sequential order on the lifeline. We represent messages using arrows. Lifelines and messages form the core of a sequence diagram.

Messages can be broadly classified into the following categories:

- Synchronous messages
- Asynchronous Messages
- Create message
- Delete Message

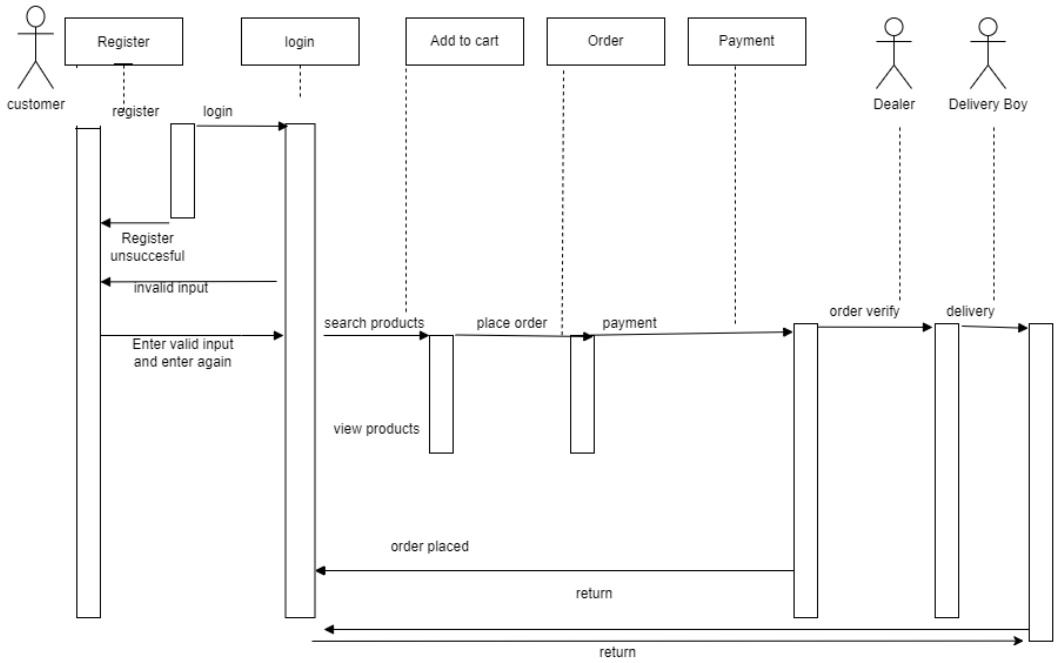
- Self-Message
- Reply Message
- Found Message
- Lost Message

iv. **Guards –** To model conditions we use guards in UML. They are used when we need to restrict the flow of messages on the pretext of a condition being met. Guards play an important role in letting software developers know the constraints attached to a system or a particular process.

Uses of sequence diagrams –

- Used to model and visualize the logic behind a sophisticated function, operation or procedure.
- They are also used to show details of UML use case diagrams.
- Used to understand the detailed functionality of current or future systems.
- Visualize how messages and tasks move between objects or components in a system.

Sequence Diagram



4.2.3 CLASS DIAGRAM

A Class Diagram in is a static structure that gives an overview of a software system by displaying classes, attributes, operations, and their relationships between each other. This Diagram includes the class name, attributes, and operation in separate designated compartments. Class Diagram helps construct the code for the software application development.

Class Diagram defines the types of objects in the system and the different types of relationships that exist among them. It gives a high-level view of an application. This modeling method can run with almost all Object-Oriented Methods. A class can refer to another class. A class can have its objects or may inherit from other classes.

Elements of A UML class diagram

Essential elements of UML class diagram are:

1. Class Name
2. Attributes
3. Operations

Class Name

The name of the class is only needed in the graphical representation of the class. It appears in the topmost compartment. A class is the blueprint of an object which can share the same relationships, attributes, operations, & semantics. The class is rendered as a rectangle, including its name, attributes, and operations in sperate compartments.

1. A class name should always start with a capital letter.
2. A class name should always be in the center of the first compartment.
3. A class name should always be written in **bold** format.
4. UML abstract class name should be written in italics format.

Attributes:

An attribute is named property of a class which describes the object being modeled. In the class diagram, this component is placed just below the name-compartment.

- The attributes are generally written along with the visibility factor.

- Public, private, protected and package are the four visibilities which are denoted by +, -, #, or ~ signs respectively.
- Visibility describes the accessibility of an attribute of a class.
- Attributes must have a meaningful name that describes the use of it in a class.

Relationships

There are mainly three kinds of relationships in UML:

1. Dependencies
2. Generalizations
3. Associations

Dependency

A dependency means the relation between two or more classes in which a change in one may force changes in the other. However, it will always create a weaker relationship. Dependency indicates that one class depends on another.

Generalization

A generalization helps to connect a subclass to its superclass. A sub-class is inherited from its superclass. Generalization relationship can't be used to model interface implementation. Class diagram allows inheriting from multiple superclasses.

In this example, the class Student is generalized from Person Class.

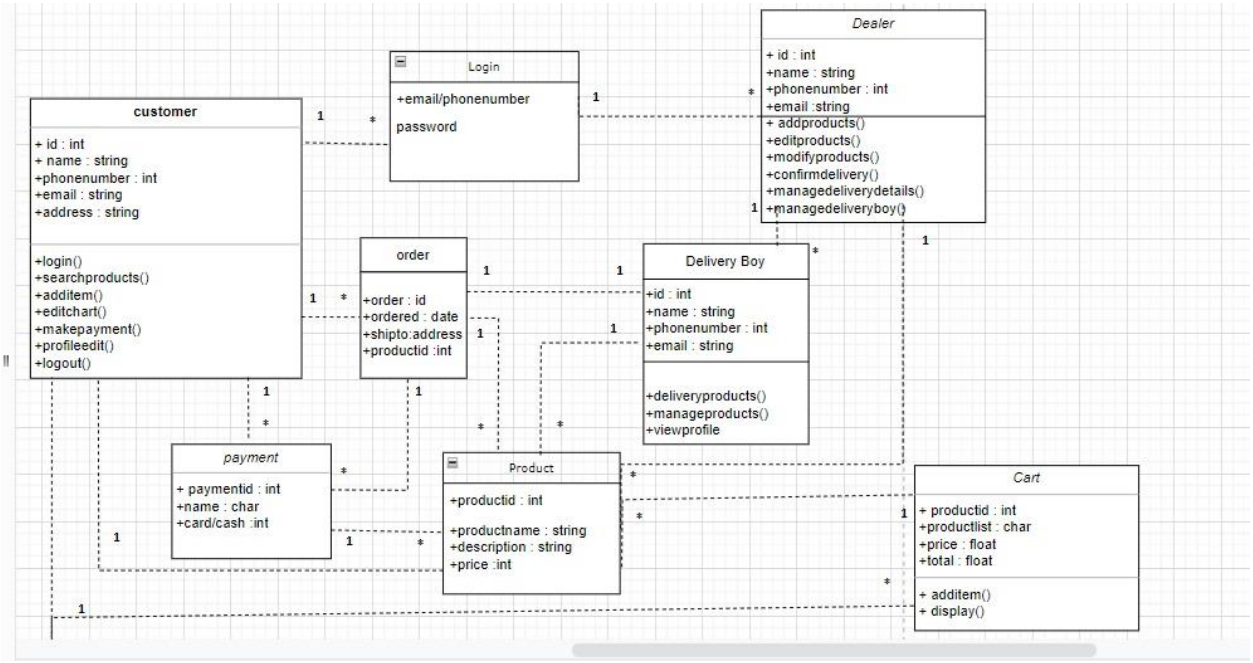
Association:

This kind of relationship represents static relationships between classes A and B. For example; an employee works for an organization.

Here are some rules for Association:

- Association is mostly verb or a verb phrase or noun or noun phrase.
- It should be named to indicate the role played by the class attached at the end of the association path.
- Mandatory for reflexive associations

CLASS DIAGRAM



4.2.4 OBJECT DIAGRAM

Object diagrams are dependent on the class diagram as they are derived from the class diagram. It represents an instance of a class diagram. The objects help in portraying a static view of an object-oriented system at a specific instant.

Both the object and class diagram are similar to some extent; the only difference is that the class diagram provides an abstract view of a system. It helps in visualizing a particular functionality of a system

Purpose of Object Diagram

The object diagram holds the same purpose as that of a class diagram. The class diagram provides an abstract view which comprises of classes and their relationships, whereas the object diagram represents an instance at a particular point of time.

The object diagram is actually similar to the concrete (actual) system behavior. The main purpose is to depict a static view of a system.

- It is used to perform forward and reverse engineering.
- It is used to understand object behavior and their relationships practically.
- It is used to get a static view of a system.
- It is used to represent an instance of a system.

Attributes

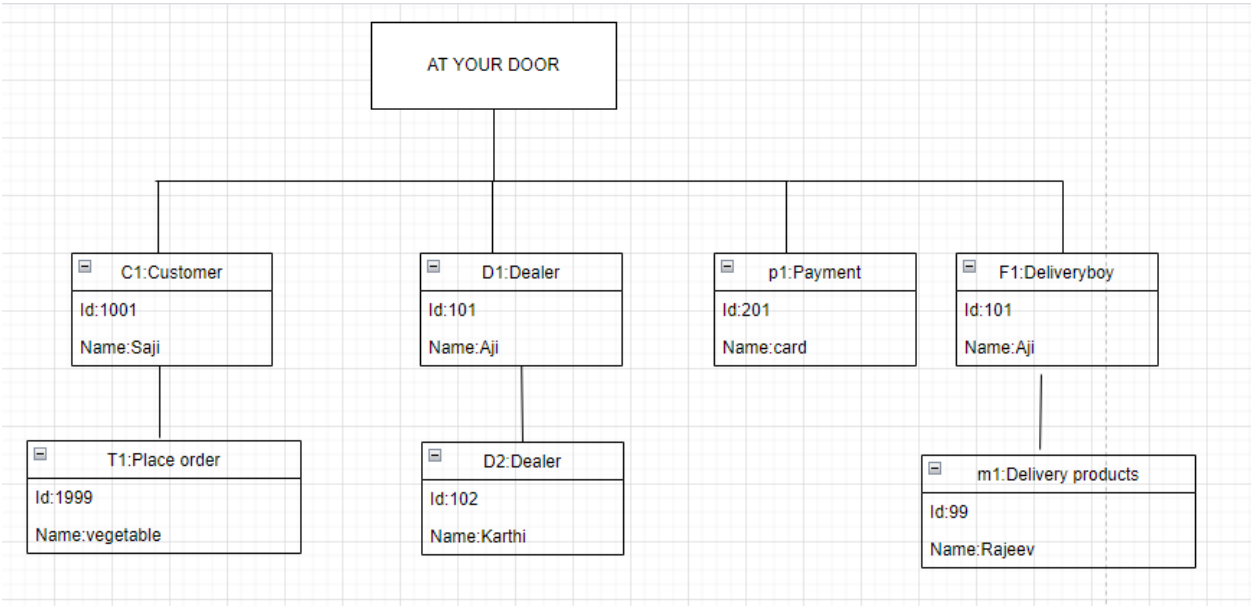
1. All the objects present in the system should be examined before start drawing the object diagram.
2. Before creating the object diagram, the relation between the objects must be acknowledged.
3. The association relationship among the entities must be cleared already.
4. To represent the functionality of an object, a proper meaningful name should be assigned.
5. The objects are to be examined to understand its functionality.

Object Names:Every object is actually symbolized like a rectangle,that offers the name from the object and its class underlined as well as divided with a colon.

Object Attributes: Similar to classes, you are able to list object attributes inside a separate compartment. However, unlike classes, object attributes should have values assigned for them.

Links:Links tend to be instances associated with associations. You can draw a link while using the lines utilized in class diagrams.

Object Diagram



4.2.5 ACTIVITY DIAGRAM:

The unified modeling language includes several subsets of diagrams, including structure diagrams, interaction diagrams, and behavior diagrams. Activity diagrams, along with use case and state machine diagrams, are considered behavior diagrams because they describe what must happen in the system being modeled.

Stakeholders have many issues to manage, so it's important to communicate with clarity and brevity. Activity diagrams help people on the business and development sides of an organization come together to understand the same process and behavior. You'll use a set of specialized symbols—including those used for starting, ending, merging, or receiving steps in the flow—to make an activity diagram, which we'll cover in more depth within this activity diagram guide.

Benefits of activity diagrams

Activity diagrams present a number of benefits to users. Consider creating an activity diagram to:

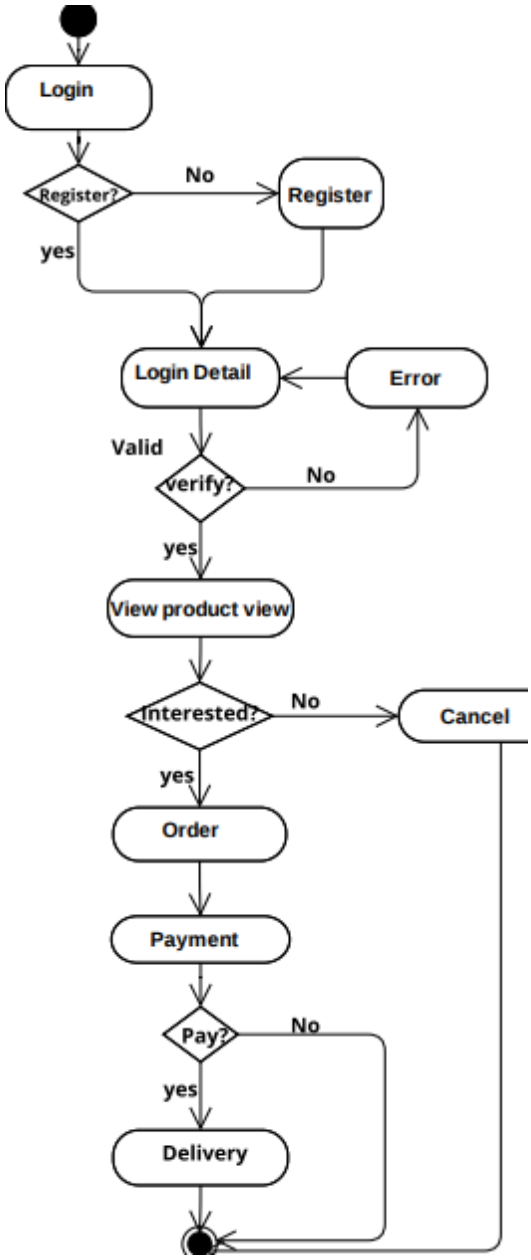
- Demonstrate the logic of an algorithm.
- Describe the steps performed in a UML use case.
- Illustrate a business process or workflow between users and the system.
- Simplify and improve any process by clarifying complicated use cases.
- Model software architecture elements, such as method, function, and operation.

Basic components of an activity diagram

Before you begin making an activity diagram, you should first understand its makeup. Some of the most common components of an activity diagram include:

- **Action:** A step in the activity wherein the users or software perform a given task. In Lucidchart, actions are symbolized with round-edged rectangles.
- **Decision node:** A conditional branch in the flow that is represented by a diamond. It includes a single input and two or more outputs.
- **Control flows:** Another name for the connectors that show the flow between steps in the diagram.
- **Start node:** Symbolizes the beginning of the activity. The start node is represented by a black circle.
- **End node:** Represents the final step in the activity. The end node is represented by an outlined black circle.

Activity Diagram

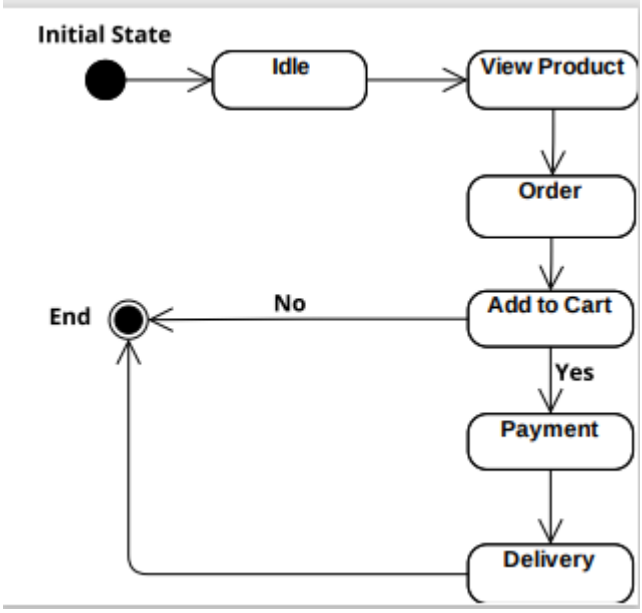


4.2.6 State Diagram

A state machine is any device that stores the status of an object at a given time and can change status or cause other actions based on the input it receives. States refer to the different combinations of information that an object can hold, not how the object behaves. In order to understand the different states of an object, you might want to

visualize all of the possible states and show how an object gets to each state, and you can do so with a UML state diagram.

Each state diagram typically begins with a dark circle that indicates the initial state and ends with a bordered circle that denotes the final state. However, despite having clear start and end points, state diagrams are not necessarily the best tool for capturing an overall progression of events. Rather, they illustrate specific kinds of behavior—in particular, shifts from one state to another**.**



4.2.7 COMPONENT DIAGRAM

The purpose of a component diagram is to show the relationship between different components in a system. For the purpose of UML 2.0, the term "component" refers to a module of classes that represent independent systems or subsystems with the ability to interface with the rest of the system.

There exists a whole development approach that revolves around components: component-based development (CBD). In this approach, component diagrams allow the planner to identify the different components so the whole system does what it's supposed to do.

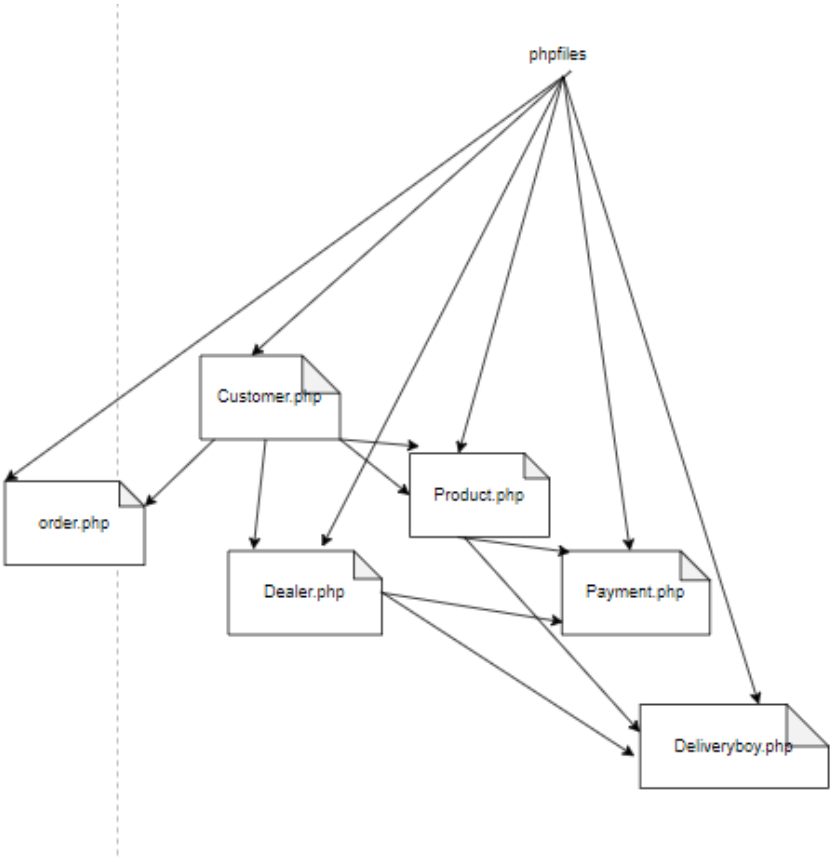
More commonly, in an OO programming approach, the component diagram allows a senior developer to group classes together based on common purpose so that the developer and others can look at a software development project at a high level.

Benefits of component diagrams

Though component diagrams may seem complex at first glance, they are invaluable when it comes to building your system. Component diagrams can help your team:

- Imagine the system's physical structure.
- Pay attention to the system's components and how they relate.
- Emphasize the service behavior as it relates to the interface.

COMPONENT DIAGRAM



4.2.7 DEPLOYMENT DIAGRAM:

In the context of the Unified Modeling Language (UML), a deployment diagram falls under the structural diagramming family because it describes an aspect of the system itself. In this case, the deployment diagram describes the physical deployment of information generated by the software program on hardware components. The information that the software generates is called an artifact. This shouldn't be confused with the use of the term in other modeling approaches like BPMN.

Deployment diagrams are made up of several UML shapes. The three-dimensional boxes, known as nodes, represent the basic software or hardware elements, or nodes,

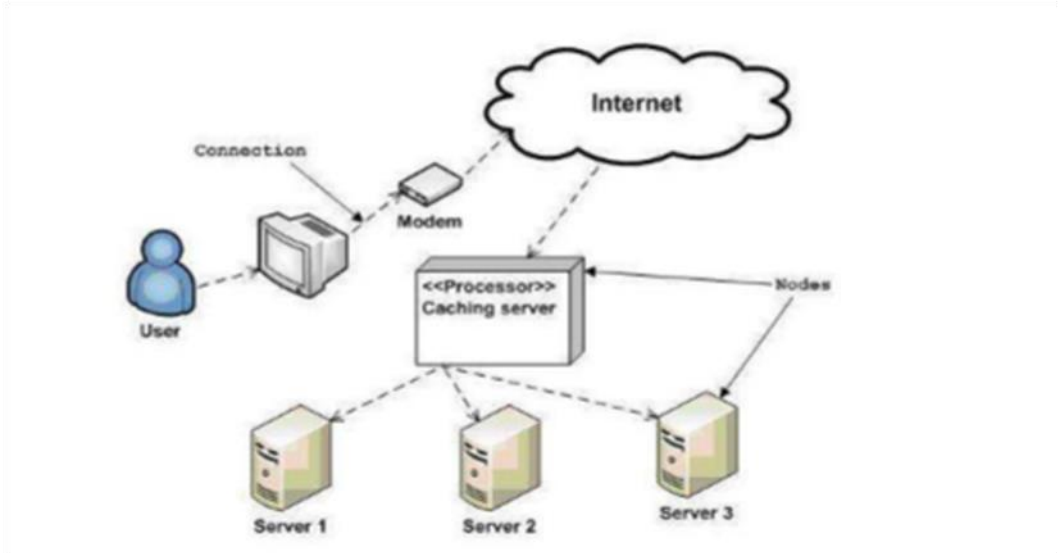
in the system. Lines from node to node indicate relationships, and the smaller shapes contained within the boxes represent the software artifacts that are deployed.

Deployment diagram applications

Deployment diagrams have several valuable applications. You can use them to:

- Show which software elements are deployed by which hardware elements.
- Illustrate the runtime processing for hardware.
- Provide a view of the hardware system's topology.

DEPLOYMENT DIAGRAM



4.5 USER INTERFACE DESIGN

4.5.1-INPUT DESIGN

Form Name : User Registration

REGISTRATION

FIRST NAME

LAST NAME

ENTER EMAIL

ENTER NUMBER

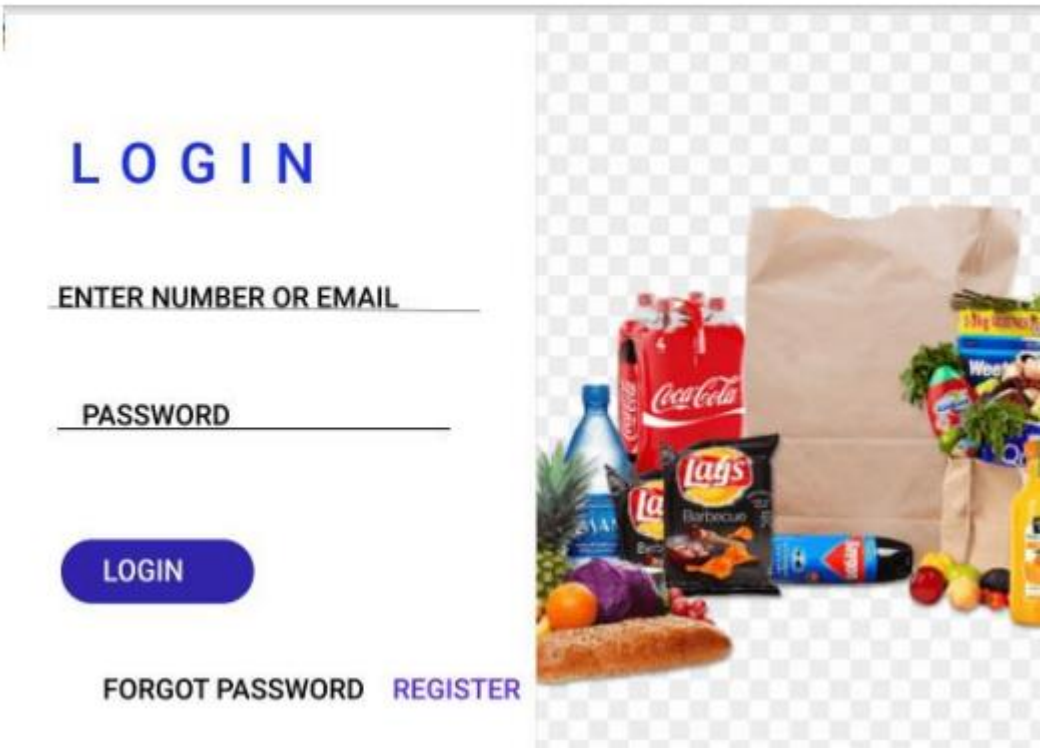
PASSWORD

CONFIRM PASSWORD

REGISTER

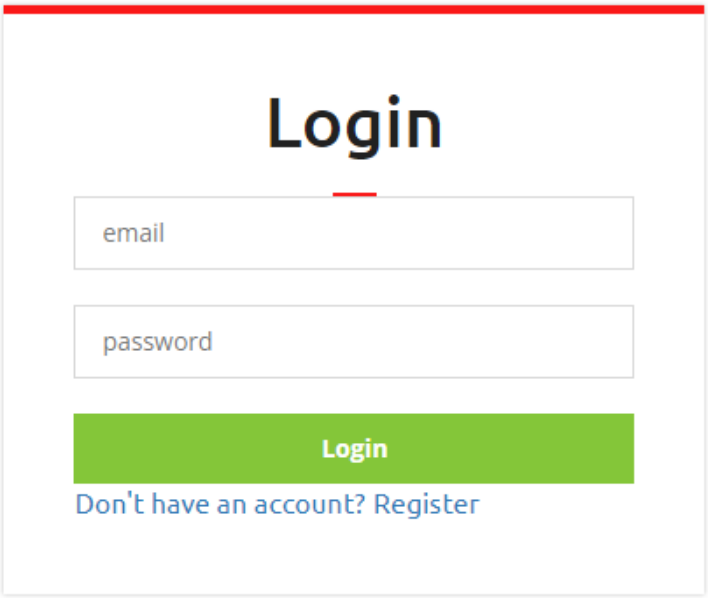
ALREADY REGISTERED SIGN IN

Form Name : User Login

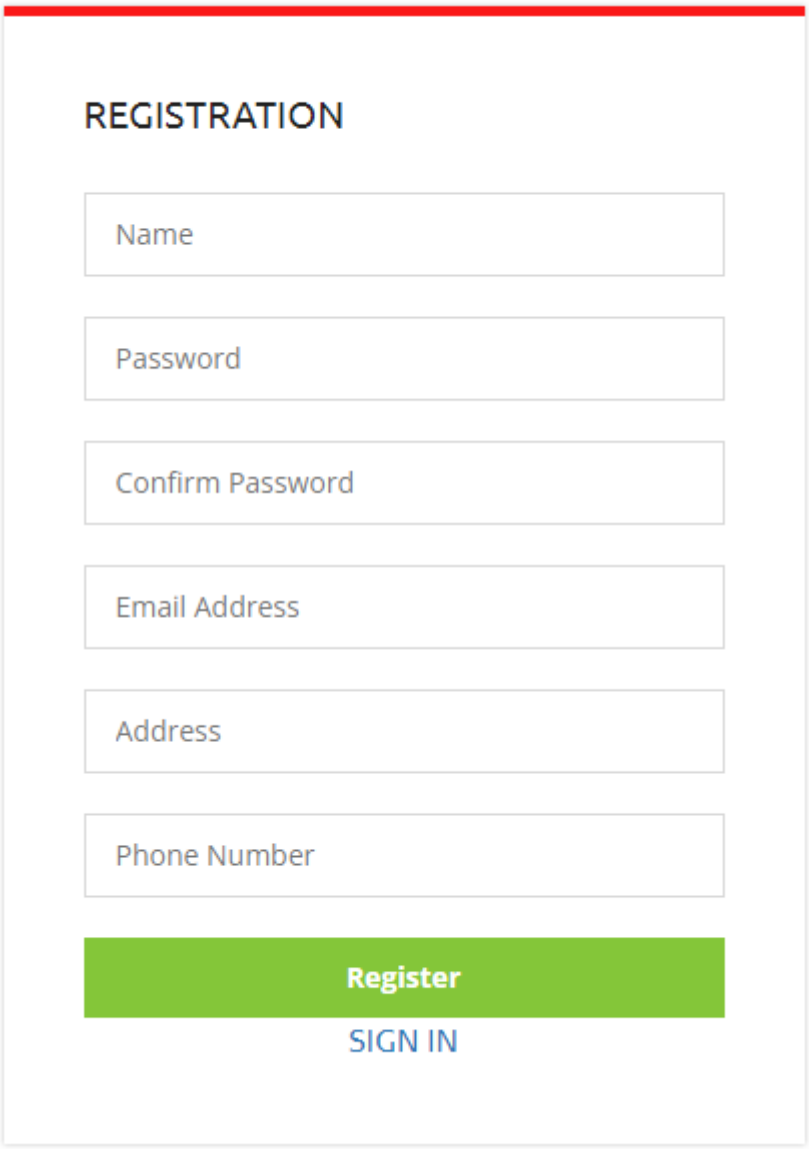


4.5.2 OUTPUT DESIGN

User Login



User Registration



4.6. DATABASE DESIGN

A database is an organized mechanism that has the capability of storing information through which a user can retrieve stored information in an effective and efficient manner. The data is the purpose of any database and must be protected.

The database design is a two level process. In the first step, user requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called Information Level Design and it is taken independent of any individual DBMS.

In the second step, this Information level design is transferred into a design for the specific DBMS that will be used to implement the system in question. This step is called Physical Level Design, concerned with the characteristics of the specific DBMS that will be used. A database design runs parallel with the system design. The organization of the data in the database is aimed to achieve the following two major objectives.

- Data Integrity
- Data independence

4.6.1 Relational Database Management System (RDBMS)

A relational model represents the database as a collection of relations. Each relation resembles a table of values or file of records. In formal relational model terminology, a row is called a tuple, a column header is called an attribute and the table is called a relation. A relational database consists of a collection of tables, each of which is assigned a unique name. A row in a tale represents a set of related values.

Relations, Domains & Attributes

A table is a relation. The rows in a table are called tuples. A tuple is an ordered set of n elements. Columns are referred to as attributes. Relationships have been set between every table in the database. This ensures both Referential and Entity Relationship Integrity. A domain D is a set of atomic values. A common method of specifying a domain is to specify a data type from which the data values forming the domain are drawn. It is also useful to specify a name for the domain to help in interpreting its values.

Every value in a relation is atomic, that is not decomposable.

Relationships

- Table relationships are established using Key. The two main keys of prime importance are Primary Key & Foreign Key. Entity Integrity and Referential Integrity Relationships can be established with these keys.
- Entity Integrity enforces that no Primary Key can have null values.
- Referential Integrity enforces that no Primary Key can have null values.
- Referential Integrity for each distinct Foreign Key value, there must exist a matching Primary Key value in the same domain. Other key are Super Key and Candidate Keys.

4.6.2 Normalization

Data are grouped together in the simplest way so that later changes can be made with minimum impact on data structures. Normalization is formal process of data structures in manners that eliminates redundancy and promotes integrity. Normalization is a technique of separating redundant fields and breaking up a large table into a smaller one. It is also used to avoid insertion, deletion, and updating anomalies. Normal form in data modelling use two concepts, keys and relationships. A key uniquely identifies a row in a table. There are two types of keys, primary key and foreign key. A primary key is an element or a combination of elements in a table whose purpose is to identify records from the same table. A foreign key is a column in a table that uniquely identifies record from a different table. All the tables have been normalized up to the third normal form.

As the name implies, it denotes putting things in the normal form. The application developer via normalization tries to achieve a sensible organization of data into proper tables and columns and where names can be easily correlated to the data by the user. Normalization eliminates repeating groups at data and thereby avoids data redundancy which proves to be a great burden on the computer resources. These include:

- ✓ Normalize the data.
- ✓ Choose proper names for the tables and columns.
- ✓ Choose the proper name for the data.

First Normal Form

The First Normal Form states that the domain of an attribute must include only atomic values and that the value of any attribute in a tuple must be a single value from the domain of that attribute. In other words 1NF disallows “relations within relations” or “relations as attribute values within tuples”. The only attribute values permitted by 1NF are single atomic or indivisible values. The first step is to put the data into First Normal Form. This can be donor by moving data into separate tables where the data is of similar type in each table. Each table is given a Primary Key or Foreign Key as per requirement of the project. In this we form new relations for each non-atomic attribute or nested relation. This eliminated repeating groups of data. A relation is said to be in first normal form if only if it satisfies the constraints that contain the primary key only.

Second Normal Form

According to Second Normal Form, for relations where primary key contains multiple attributes, no non-key attribute should be functionally dependent on a part of the primary key. In this we decompose and setup a new relation for each partial key with its dependent attributes. Make sure to keep a relation with the original primary key and any attributes that are fully functionally dependent on it. This step helps in taking out data that is only dependent on a part of the key. A relation is said to be in second normal form if and only if it satisfies all the first normal form conditions for the primary key and every non-primary key attributes of the relation is fully dependent on its primary key alone.

Third Normal Form

According to Third Normal Form, Relation should not have a non-key attribute functionally determined by another non-key attribute or by a set of non-key attributes. That is, there should be no transitive dependency on the primary key. In this we decompose and set up relation that includes the non-key attributes that functionally determines other non-key attributes. This step is taken to get rid of anything that does not depend entirely on the Primary Key. A relation is said to be in third normal form if only if it is in second normal form and more over the non key attributes of the relation should not be depend on other non-key attribute.

TABLE DESIGN

Table Name : tbl_login

Description:for login

Primary key :login_id

Field	Datatype	Constraints
Logn_id	int	Primary key
username	varchar(50)	NOT NULL
password	varchar(50)	NOT NULL
role	int	NOT NULL
status	Varchar(5)	NOT NULL

Table Name :tbl_registration

Description:For Store Registration Details

Primary Key :Reg_id

Foreign Key : login_id refers from tbl_login

Field	Datatype	Constraints
Reg_id	int	Primary key
First name	varchar(100)	NOT NULL
Last name	Varchar(30)	NOT NULL
email	Varchar(60)	NOT NULL
Phn_no	int	NOT NULL
Login_id	int	Foreign key

Table Name :tbl_product

Description : For Product Details

Primary Key :Product_id

Foreign key :category_id refers from tbl_ category

Field	Datatype	Constraints
Product_id	int	Primary key
Product_name	varchar(100)	NOT NULL
description	Varchar(100)	NOT NULL
type	Varchar(20)	NOT NULL
Image	Varchar(60)	NOT NULL
avalibility	Varchar(80)	NOT NULL
price	Varchar(50)	NOT NULL

Table name :tbl_delivery_request
Description : For delivery request

Primary Key: Delivery_id

Foreign key : login_id refers from tbl_login,product_id refers from tbl_product,

Payment_id refers from tbl_payment,Dboy_id refers from tbl_Dboy

Field	Datatype	Constraints
Delivery_id	int	Primary key
Product_id	int	Foreign key
Login_id	int	Foreign key
Payment_id	int	Foreign key
Dboy_id	int	Foreign key

Table Name : tbl_payment
Primary Key :Payment _id

Description : for payment

Field	Datatype	Constraints
Payment_id	int	Primary key
name	varchar(100)	NOT NULL
date	date	NOT NULL
payment	Varchar(60)	NOT NULL
service	Varchar(30)	NOT NULL

Table Name :tbl_deliveryboy
Primary Key : Dboy_id

Foreign key : login_id refers from tbl_login

Field	Datatype	Constraints
Dboy_id	int	Primary key
name	Varchar(20)	NOT NULL
mobile	Varchar(20)	NOT NULL
email	Varchar(20)	NOT NULL
Address	Varchar(60)	NOT NULL
Login_id	int	Foreign key

1) **Table Name : tbl_category**
   Primary key : category_id

Field	Datatype	Constraints
Category_id	int	Primary Key
Category name	Varchar(20)	NOT NULL
image	Varchar(20)	NOT NULL
status	varchar(10)	NOT NULL

2) **Table Name : tbl_dealer**
   Primary Key : dealer_id

   Foreign Key : login_id refers from tbl_login

Field	Datatype	Constraints
dealer_id	int	Primary key
dealer name	varchar(100)	NOT NULL
email	Varchar(30)	NOT NULL
place	Varchar(60)	NOT NULL
Phn_no	number	NOT NULL
Login_id	int	Foreign Key

Table Name :**tbl_cart**
Description:For Adding products

Primary Key :cart_id

Foreign Key: login_id refers from tbl_login,product_id refers from tbl_product

Field	Datatype	Constraints
Cart_id	int	Primary key
Product_id	int	Foreign key
Login_id	int	Foreign key

Table Name : tbl_order

Description: for ordering products

Primary key : order_id

Foreign key : product-id refers from tbl_product

Field	Datatype	constraints
Order_id	int	Primary key
Product_id	int	Foreign key

Table Name : tbl_feedback

Description : for feedback

Primary Key : Feedback_id

Foreign key :login_id refers from tbl_login

Field	Datatype	Constraints
Feedback_id	int	Primary key
Login_id	varchar(100)	Foreign key
subject	Varchar(50)	NOT NULL
review	Varchar(40)	NOT NULL
rating	Varchar(40	NOT NULL
type	Varchar(40)	NOT NULL
reply	Varchar(40)	NOT NULL

CHAPTER 5

SYSTEM TESTING

5.1 INTRODUCTION

Software Testing is the process of executing software in a controlled manner, in order to answer the question - Does the software behave as specified? Software testing is often used in association with the terms verification and validation. Validation is the checking or testing of items, includes software, for conformance and consistency with an associated specification. Software testing is just one kind of verification, which also uses techniques such as reviews, analysis, inspections, and walkthroughs. Validation is the process of checking that what has been specified is what the user actually wanted.

Other activities which are often associated with software testing are static analysis and dynamic analysis. Static analysis investigates the source code of software, looking for problems and gathering metrics without actually executing the code. Dynamic analysis looks at the behavior of software while it is executing, to provide information such as execution traces, timing profiles, and test coverage information.

Testing is a set of activity that can be planned in advanced and conducted systematically. Testing begins at the module level and work towards the integration of entire computers based system. Nothing is complete without testing, as it vital success of the system testing objectives, there are several rules that can serve as testing objectives. They are:

Testing is a process of executing a program with the intent of finding an error.

- A good test case is one that has high possibility of finding an undiscovered error.
- A successful test is one that uncovers an undiscovered error.

If a testing is conducted successfully according to the objectives as stated above, it would uncover errors in the software. Also testing demonstrate that the software function appear to be working according to the specification, that performance requirement appear to have been met.

There are three ways to test program.

- For correctness
- For implementation efficiency
- For computational complexity

Test for correctness are supposed to verify that a program does exactly what it was designed to do. This is much more difficult than it may at first appear, especially for large programs.

5.2 TEST PLAN

A test plan implies a series of desired course of action to be followed in accomplishing various testing methods. The Test Plan acts as a blue print for the action that is to be followed. The software engineers create a computer program, its documentation and related data structures. The software developers is always responsible for testing the individual units of the programs, ensuring that each performs the function for which it was designed. There is an independent test group (ITG) which is to remove the inherent problems associated with letting the builder to test the thing that has been built. The specific objectives of testing should be stated in measurable terms. So that the mean time to failure, the cost to find and fix the defects, remaining defect density or frequency of occurrence and test work-hours per regression test all should be stated within the test plan.

The levels of testing include:

- ❖ Unit testing
- ❖ Integration Testing
- ❖ Data validation Testing
- ❖ Output Testing

5.2.1 Unit Testing

Unit testing focuses verification effort on the smallest unit of software design – the software component or module. Using the component level design description as a guide, important control paths are tested to uncover errors within the boundary of the module. The relative complexity of tests and uncovered scope established for unit testing. The unit testing is white-box oriented, and step can be conducted in parallel for multiple components. The

modular interface is tested to ensure that information properly flows into and out of the program unit under test. The local data structure is examined to ensure that data stored temporarily maintains its integrity during all steps in an algorithm's execution. Boundary conditions are tested to ensure that all statements in a module have been executed at least once. Finally, all error handling paths are tested.

Tests of data flow across a module interface are required before any other test is initiated. If data do not enter and exit properly, all other tests are moot. Selective testing of execution paths is an essential task during the unit test. Good design dictates that error conditions be anticipated and error handling paths set up to reroute or cleanly terminate processing when an error does occur. Boundary testing is the last task of unit testing step. Software often fails at its boundaries.

Unit testing was done in Sell-Soft System by treating each module as separate entity and testing each one of them with a wide spectrum of test inputs. Some flaws in the internal logic of the modules were found and were rectified. After coding each module is tested and run individually. All unnecessary code where removed and ensured that all modules are working, and gives the expected result.

5.2.2 Integration Testing

Integration testing is systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit tested components and build a program structure that has been dictated by design. The entire program is tested as whole. Correction is difficult because isolation of causes is complicated by vast expanse of entire program. Once these errors are corrected, new ones appear and the process continues in a seemingly endless loop. After performing unit testing in the System all the modules were integrated to test for any inconsistencies in the interfaces. Moreover differences in program structures were removed and a unique program structure was evolved.

5.2.3 Validation Testing or System Testing

This is the final step in testing. In this the entire system was tested as a whole with all forms, code, modules and class modules. This form of testing is popularly known as Black Box testing or System tests.

Black Box testing method focuses on the functional requirements of the software. That is, Black Box testing enables the software engineer to derive sets of input conditions thatwill fully exercise all functional requirements for a program.

Black Box testing attempts to find errors in the following categories; incorrect or missing functions, interface errors, errors in data structures or external data access, performance errors and initialization errors and termination errors.

5.2.4 Output Testing or User Acceptance Testing

The system considered is tested for user acceptance; here it should satisfy the firm's need. The software should keep in touch with perspective system; user at the time of developing and making changes whenever required. This done with respect to the following points:

- Input Screen Designs,
- Output Screen Designs,

The above testing is done taking various kinds of test data. Preparation of test data plays a vital role in the system testing. After preparing the test data, the system under study is tested using that test data. While testing the system by which test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

CHAPTER 6

IMPLEMENTATION

6.1 INTRODUCTION

Implementation is the stage of the project where the theoretical design is turned into a working system. It can be considered to be the most crucial stage in achieving a successful new system gaining the users confidence that the new system will work and will be effective and accurate. It is primarily concerned with user training and documentation. Conversion usually takes place about the same time the user is being trained or later. Implementation simply means convening a new system design into operation, which is the process of converting a new revised system design into an operational one.

At this stage the main work load, the greatest upheaval and the major impact on the existing system shifts to the user department. If the implementation is not carefully planned or controlled, it can create chaos and confusion.

Implementation includes all those activities that take place to convert from the existing system to the new system. The new system may be a totally new, replacing an existing manual or automated system or it may be a modification to an existing system. Proper implementation is essential to provide a reliable system to meet organization requirements. The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from the old system to the new system. The system can be implemented only after through testing is done and if it is found to be working according to the specifications. The system personnel check the feasibility of the system. The more complex the system being implemented, the more involved will be the system analysis and design effort required to implement the three main aspects: education and training, system testing and changeover.

The implementation state involves the following tasks:

- Careful planning.
- Investigation of system and constraints.
- Design of methods to achieve the changeover.

6.2 IMPLEMENTATION PROCEDURES

Implementation of software refers to the final installation of the package in its real environment, to the satisfaction of the intended uses and the operation of the system. In many organizations someone who will not be operating it, will commission the software development project. In the initial stage people doubt about the software but we have to ensure that the resistance does not build up, as one has to make sure that:

- The active user must be aware of the benefits of using the new system.
- Their confidence in the software is built up.
- Proper guidance is imparted to the user so that he is comfortable in using the application.

Before going ahead and viewing the system, the user must know that for viewing the result, the server program should be running in the server. If the server object is not up running on the server, the actual process won't take place.

6.2.1 User Training

User training is designed to prepare the user for testing and converting the system. To achieve the objective and benefits expected from computer based system, it is essential for the people who will be involved to be confident of theirrole in the new system. As system becomes more complex, the need for training is more important. By user training the user comes to know how to enter data, respond to error messages, interrogate the database and call up routine that will produce reports and perform other necessary functions.

6.2.2 Training on the Application Software

After providing the necessary basic training on computer awareness the user will have to be trained on the new application software. This will give the underlying philosophy of the use of the new system such as the screen flow, screen design type of help on the screen, type of errors while entering the data, the corresponding validation check at each entry and the ways to correct the date entered. It should then cover information needed by the specific user/ group to use the system or part of the system while imparting the training of the program on the application. This training may be different across different user groups and across different levels of hierarchy

6.2.3 System Maintenance

Maintenance is the enigma of system development. The maintenance phase of the software cycle is the time in which a software product performs useful work. After a system is successfully implemented, it should be maintained in a proper manner. System maintenance is an important aspect in the software development life cycle. The need for system maintenance is for it to make adaptable to the changes in the system environment. Software maintenance is of course, far more than "Finding Mistakes".

CHAPTER 7

CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSION

The proposed system will provide a new marketing culture despite of the traditional system. The system has been developed with much care and free of errors at the same time it is efficient and less time consuming. The purpose of this project was to develop a web application for purchasing item from a shop, The application which can be implemented to any nearby shops or market or branded shops who were selling various kinds of product which were accessible by the user on a single touch.

CHAPTER 8

BIBLIOGRAPHY

REFERENCES:

- Gary B. Shelly, Harry J. Rosenblatt, "*System Analysis and Design*", 2009.
- Roger S Pressman, "*Software Engineering*", 1994.
- PankajJalote, "So*ftware engineering*: a precise approach", 2006.
- James lee and Brent ware Addison, "Open source web development with LAMP", 2003
- IEEE Std 1016 Recommended Practice for Software Design Descriptions.

WEBSITES:

- www.w3schools.com
- www.jquery.com
- http://homepages.dcc.ufmg.br/~rodolfo/es-1-03/IEEE-Std-830-1998.pdf
- www.agilemodeling.com/artifacts/useCaseDiagram.html

CHAPTER 9

APPENDIX

9.1 Sample Code

LOGIN.PHP

```
<?php
session_start();
include('db.php');
if(isset($_POST['submit']))
{
  @$email=$_POST['email'];
  @$pass=$_POST['password'];

$query="SELECT * FROM `login_tbl` WHERE email='$email' and `password`='$pass'";
$result=mysqli_query($conn,$query) or die("error".mysqli_error());
$rowcount=mysqli_num_rows($result);
if($rowcount>0)
{
 while($rows=mysqli_fetch_assoc($result)){
  if($rows['role']==0)
  {
          $_SESSION['login_admin'] = $rows['login_id'];
   header("location:admin/admin.php");
  }
  else if($rows['role']==3){
          $_SESSION['login_admin'] = $rows['login_id'];
          header("location:home.php");
  }
  else if($rows['role']==1){
          $_SESSION['login_admin'] = $rows['login_id'];
          header("location:dealer/dealer1.php");
  }
  else {
          echo "<script>alert('Username and password are incorrect')</script>";
```

```
            echo "<script>window.location='index.php'</script>";
            }
}
 }
}
?>

<!DOCTYPE html>
<html>
<head>
<title>Grocery Store a Ecommerce Online Shopping Category Flat Bootstrap Responsive
Website Template | Sign In & Sign Up :: w3layouts</title>
<!-- for-mobile-apps -->
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<meta name="keywords" content="Grocery Store Responsive web template, Bootstrap Web
Templates, Flat Web Templates, Android Compatible web template,
Smartphone Compatible web template, free webdesigns for Nokia, Samsung, LG,
SonyEricsson, Motorola web design" />
<script type="application/x-javascript"> addEventListener("load", function() {
setTimeout(hideURLbar, 0); }, false);
            function hideURLbar(){ window.scrollTo(0,1); } </script>
<!-- //for-mobile-apps -->
<link href="css/bootstrap.css" rel="stylesheet" type="text/css" media="all" />
<link href="css/style.css" rel="stylesheet" type="text/css" media="all" />
<!-- font-awesome icons -->
<link href="css/font-awesome.css" rel="stylesheet" type="text/css" media="all" />
<!-- //font-awesome icons -->
<!-- js -->
<script src="js/jquery-1.11.1.min.js"></script>
<!-- //js -->
<link
href='//fonts.googleapis.com/css?family=Ubuntu:400,300,300italic,400italic,500,500italic,70
0,700italic' rel='stylesheet' type='text/css'>
<link
href='//fonts.googleapis.com/css?family=Open+Sans:400,300,300italic,400italic,600,600itali
c,700,700italic,800,800italic' rel='stylesheet' type='text/css'>
```

```
<!-- start-smoth-scrolling -->
<script type="text/javascript" src="js/move-top.js"></script>
<script type="text/javascript" src="js/easing.js"></script>
<script type="text/javascript">
  jQuery(document).ready(function($) {
        $(".scroll").click(function(event){
              event.preventDefault();
              $('html,body').animate({scrollTop:$(this.hash).offset().top},1000);
        });
  });
</script>
<!-- start-smoth-scrolling -->
</head>

<body>

<!-- products-breadcrumb -->
  <div class="products-breadcrumb">
        <div class="container">
              <ul>
                    <li><i class="fa fa-home" aria-hidden="true"></i><a
href="index.php">Home</a><span>|</span></li>
                    <li>Sign In </li>
              </ul>
        </div>
  </div>
<!-- //products-breadcrumb -->
<!-- banner -->
  <div class="banner">

        <div class="w3l_banner_nav_right">
<!-- login -->
        <div class="w3_login">

              <div class="w3_login_module">
                    <div class="module form-module">
                     <div class="form">
```

```
<form  method="POST"  name="myForm"
onsubmit="return validateForm()">
<h3>Login</h3>
<input type="text" name="email"
placeholder="email" required=" ">
<input type="password" name="password"
placeholder="password" required=" ">

<input type="submit" name="submit"
value="Login" >
<a href="">Don't have an account?</a>
<a href="register.php">Register</a>
</form>
<script>
function validateForm()
{
var y=document.myForm.password.value;
var x=document.myForm.email.value;
var atposition=x.indexOf("@");
var dotposition=x.lastIndexOf(".");
if (atposition<1 || dotposition<atposition+2 ||
dotposition+2>=x.length){
alert("Please enter a valid e-mail address" );
return false;
}
else if(y.length<6){
alert("Password must be at least 6 characters long.");
return false;
}
else
{
alert(' Login
Successfully');

}
}
```

```
                                        </script>
                                </div>
                                <div class="form">

                                <form  method="POST"  name="myForm" onsubmit="return
validateForm()">
                                        <h3>Login</h3>
                                        <input type="text" name="email"
placeholder="email" required=" ">
                                        <input type="password" name="password"
placeholder="password" required=" ">

                                        <input type="submit" name="submit"
value="Login" >
                                        <a href="">Don't have an account?</a>
                                        <a href="register.php">Register</a>
                                </form>
                                </div>

                        </div>
                </div>
                <script>
                        $('.toggle').click(function(){
                          // Switches the Icon
                          $(this).children('i').toggleClass('fa-pencil');
                          // Switches the forms
                          $('.form').animate({
                                height: "toggle",
                                'padding-top': 'toggle',
                                'padding-bottom': 'toggle',
```

```
                                        opacity: "toggle"
                                    }, "slow");
                                });
                        </script>
                </div>
<!-- //login -->
                </div>
                <div class="clearfix"></div>
    </div>

</script>

</body>

</html>
```

REGISTER.PHP

```
<?php
include 'db.php';

if(isset($_POST['submit'])){
    $name=$_POST['Name'];
    $email=$_POST['Email'];
    $pass=$_POST['Password'];
    $phn=$_POST['Phone'];
    $add=$_POST['Address'];
    $sel="SELECT * FROM `login_tbl` WHERE `email`='$email'";
    $query1=mysqli_query($conn,$sel);
    $num=mysqli_num_rows($query1);
    if($num>0)
    {
            echo '<script>alert("Mail already in use..!");</script>';
            echo "<script>window.location='register.php'</script>";
    }
    else{
```

```
  $sq = mysqli_query($conn,"INSERT INTO `login_tbl`(`email`, `password`, `role`, `status`)
VALUES ('$email','$pass',3,1)");
   $rid = mysqli_insert_id($conn);
  $sql = mysqli_query($conn,"INSERT INTO `register`(`uname`, `phonenumber`,
`login_id`,`address`) VALUES ('$name','$phn','$rid','$add')");
header("Location:login.php");
  }
}

?>
<!DOCTYPE html>
<html>
<head>
<title>Grocery Store a Ecommerce Online Shopping Category Flat Bootstrap Responsive
Website Template | Sign In & Sign Up :: w3layouts</title>
<!-- for-mobile-apps -->
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<meta name="keywords" content="Grocery Store Responsive web template, Bootstrap Web
Templates, Flat Web Templates, Android Compatible web template,
Smartphone Compatible web template, free webdesigns for Nokia, Samsung, LG,
SonyEricsson, Motorola web design" />
<script type="application/x-javascript"> addEventListener("load", function() {
setTimeout(hideURLbar, 0); }, false);
        function hideURLbar(){ window.scrollTo(0,1); } </script>
<!-- //for-mobile-apps -->
<link href="css/bootstrap.css" rel="stylesheet" type="text/css" media="all" />
<link href="css/style.css" rel="stylesheet" type="text/css" media="all" />
<!-- font-awesome icons -->
<link href="css/font-awesome.css" rel="stylesheet" type="text/css" media="all" />
<!-- //font-awesome icons -->
<!-- js -->
<script src="js/jquery-1.11.1.min.js"></script>
<!-- //js -->
<link
href='//fonts.googleapis.com/css?family=Ubuntu:400,300,300italic,400italic,500,500italic,70
0,700italic' rel='stylesheet' type='text/css'>
```

```
<link
href='//fonts.googleapis.com/css?family=Open+Sans:400,300,300italic,400italic,600,600itali
c,700,700italic,800,800italic' rel='stylesheet' type='text/css'>
<!-- start-smoth-scrolling -->
<script type="text/javascript" src="js/move-top.js"></script>
<script type="text/javascript" src="js/easing.js"></script>
<script type="text/javascript">
  jQuery(document).ready(function($) {
        $(".scroll").click(function(event){
              event.preventDefault();
              $('html,body').animate({scrollTop:$(this.hash).offset().top},1000);
        });
  });
</script>
<!-- start-smoth-scrolling -->
</head>

<body>

<!-- products-breadcrumb -->
  <div class="products-breadcrumb">
        <div class="container">
              <ul>
                    <li><i class="fa fa-home" aria-hidden="true"></i><a
href="index.php">Home</a><span>|</span></li>
                    <li>Sign up</li>
              </ul>
        </div>
  </div>
<!-- //products-breadcrumb -->
<!-- banner -->
  <!--  -->
<!-- login -->
        <div class="w3_login">

              <div class="w3_login_module">
                    <div class="module form-module">
```

```
                                <div class="form">
                                        <h2>REGISTRATION</h2>

                                        <script>
        function registration()
          {

            var name= document.getElementById("Name").value;
            var email= document.getElementById("Email").value;
            var number= document.getElementById("Phone").value;
            var address=document.getElementById("Address").value;
            var Password= document.getElementById("Password").value;
            var CPassword= document.getElementById("CPassword").value;


            var pwd_expression = /^(?=.*?[A-Z])(?=.*?[a-z])(?=.*?[0-9])(?=.*?[#?!@$%^&*-
])/;
            var letters = /^[A-Za-z]+$/;
            var filter = /^([a-zA-Z0-9_\.\-])+\@(([a-zA-Z0-9\-])+\.)+([a-zA-Z0-9]{2,4})+$/;
            var phoneno =  /^[\+]?[(]?[0-9]{3}[)]?[-\s\.]?[0-9]{3}[-\s\.]?[0-9]{4,6}$/im;

            if(name==")
            {

            }
            else if(!letters.test(name))
            {
              alert('Name field required only alphabet characters');
                                        return false;
            }

            else if(email==")
            {
              alert('Please enter your user email id');
                                        return false;
            }
            else if(address==")
```

```
            {
               alert('Please enter your Address');
                                    return false;
            }
            else if (!filter.test(email))
            {
               alert('Enter the e-mail format correctly');
                                    return false;
            }
         else if(Phone=='')
            {
               alert('Please enter your number');
                                    return false;
            }
            else if (!validatePhoneNumber(number))
            {
               alert('Enter the phone format correctly');
                                    return false;
            }



            else if(Password=='')
            {
               alert('Please enter Password');
                                    return false;
            }
            else if(CPassword=='')
            {
               alert('Enter Confirm Password');
                                    return false;
            }
            else if(!pwd_expression.test(Password))
            {
               alert ('Upper case, Lower case, Special character and Numeric letter are required
   in Password filed');
                                    return false;
            }
```

```
            else if(Password != CPassword )
            {
               alert ('Password not Matched');
                                    return false;
            }
            else if(document.getElementById("Password").value.length < 6)
            {
               alert ('Password minimum length is 6');
                                    return false;
            }
            else if(document.getElementById("CPassword").value.length > 12)
            {
               alert ('Password max length is 12');
                                    return false;
            }
            else
            {
                 return true;

            }
          }

                    function validatePhoneNumber(input_str) {
   var re = /^[\+]?[(]?[0-9]{3}[)]?[-\s\.]?[0-9]{3}[-\s\.]?[0-9]{4,6}$/im;

   return re.test(input_str);
}

function validateForm(event) {
   var phone = document.getElementById('myform_phone').value;
   if (!validatePhoneNumber(phone)) {
      document.getElementById('phone_error').classList.remove('hidden');
   } else {
      document.getElementById('phone_error').classList.add('hidden');
      alert("validation success")
   }
   event.preventDefault();
```

```
}
        </script>

                                </div>
                                <div class="form">

                                <h2>REGISTRATION</h2>
                                        <form method="POST" onsubmit="return registration()">
                                                <input type="text" name="Name" id="Name"
placeholder="Name" required=" ">
                                                <input type="password" name="Password"
id="Password" placeholder="Password" required=" ">
                                                <input type="password" name="CPassword"
id="CPassword" placeholder="Confirm Password" required=" ">

                                                <input type="email" name="Email" id="Email"
placeholder="Email Address" required=" ">
                                                <input type="text" name="Address" id="Address"
placeholder="Address" required=" ">
                                                <input type="text" name="Phone" id="Phone"
placeholder="Phone Number" required=" ">
                                                <input type="submit" value="Register"
name="submit" onClick="registration()">

                                                <div><center><a href="login.php">SIGN
IN</a></center></div>
                                        </form>
                                </div>
                                <div class="form">


                        </div>
                        <script>
                                $('.toggle').click(function(){
                                  // Switches the Icon
                                  $(this).children('i').toggleClass('fa-pencil');
                                  // Switches the forms
```

```
                $('.form').animate({
                        height: "toggle",
                        'padding-top': 'toggle',
                        'padding-bottom': 'toggle',
                        opacity: "toggle"
                 }, "slow");
                });
            </script>
        </div>
<!-- //login -->
        </div>
        <div class="clearfix"></div>
   </div>
<!-- //banner -->

</body>
</html>
```

HOME.php

```
<?php
include 'db.php';
session_start();
error_reporting(0);
$uid = $_SESSION['login_admin'];
if($_SESSION['login_admin']==""){
  header('location:login.php');
}
$sql = mysqli_query($conn,"SELECT * from register where login_id='$uid'");
while($row = mysqli_fetch_array($sql)){
  $name = $row['uname'];
}

?>
<!DOCTYPE html>
<html>
<head>
```

```
<title>@Your door</title>
<!-- for-mobile-apps -->
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<meta name="keywords" content="Grocery Store Responsive web template, Bootstrap Web
Templates, Flat Web Templates, Android Compatible web template,
Smartphone Compatible web template, free webdesigns for Nokia, Samsung, LG,
SonyEricsson, Motorola web design" />
<script type="application/x-javascript"> addEventListener("load", function() {
setTimeout(hideURLbar, 0); }, false);
        function hideURLbar(){ window.scrollTo(0,1); } </script>
<!-- //for-mobile-apps -->
<link href="css/bootstrap.css" rel="stylesheet" type="text/css" media="all" />
<link href="css/style.css" rel="stylesheet" type="text/css" media="all" />
<!-- font-awesome icons -->
<link href="css/font-awesome.css" rel="stylesheet" type="text/css" media="all" />
<!-- //font-awesome icons -->
<link href="https://maxcdn.bootstrapcdn.com/font-awesome/4.2.0/css/font-
awesome.min.css" rel="stylesheet">
<!-- js -->
<script src="js/jquery-1.11.1.min.js"></script>
<!-- //js -->
<link
href='//fonts.googleapis.com/css?family=Ubuntu:400,300,300italic,400italic,500,500italic,70
0,700italic' rel='stylesheet' type='text/css'>
<link
href='//fonts.googleapis.com/css?family=Open+Sans:400,300,300italic,400italic,600,600itali
c,700,700italic,800,800italic' rel='stylesheet' type='text/css'>
<!-- start-smoth-scrolling -->
<script type="text/javascript" src="js/move-top.js"></script>
<script type="text/javascript" src="js/easing.js"></script>
<script type="text/javascript">
  jQuery(document).ready(function($) {
        $(".scroll").click(function(event){
                event.preventDefault();
                $('html,body').animate({scrollTop:$(this.hash).offset().top},1000);
        });
```

```
  });
</script>
<!-- start-smoth-scrolling -->
</head>

<body>
<!-- header -->
  <div class="agileits_header">
        <div class="w3l_offers">
            <a href="products.php">Today's special Offers !</a>
        </div>
        <div class="w3l_search">
            <form action="#" method="post">
                <input type="text" name="Product" value="Search a product..."
onfocus="this.value = ';" onblur="if (this.value == '') {this.value = 'Search a product...';}"
required="">
                <input type="submit" value=" ">
            </form>
        </div>
        <div class="product_list_header">
            <form action="#" method="post" class="last">
        <fieldset>
          <input type="hidden" name="cmd" value="_cart" />
          <input type="hidden" name="display" value="1" />
         <a href="carts.php"> <input type="submit" name="submit" value="View your
cart" class="button" /></a>
        </fieldset>
      </form>
        </div>
        <div class="w3l_header_right1">
            <ul>

            </ul>
        </div>


        <div class="w3l_header_right1" style="margin-top: 6px;">
```

```
                    <h2><a href="logout.php">Logout</a></h2>

            </div>
            <div class="w3l_header_right1">
            <div class="form-floating">

            <div class="w3l_header_right1">
  <select class="form-select" id="floatingSelect" aria-label="Floating label select
example"style=" color: #fff;background-color: #84C639; height: 47px; font-size: 1.74rem;
margin: 6px;text-align: center;font-weight: 600;">
  <option selected><?php echo $name; ?></option>
   <option value="1">My profile</option>
   <option value="2">My orders</option>
  </select>
            </div>
</div>
            </div>

            <div class="clearfix"> </div>
  </div>
<!-- script-for sticky-nav -->
  <script>
  $(document).ready(function() {
            var navoffeset=$(".agileits_header").offset().top;
            $(window).scroll(function(){
                    var scrollpos=$(window).scrollTop();
                    if(scrollpos >=navoffeset){
                            $(".agileits_header").addClass("fixed");
                    }else{
                            $(".agileits_header").removeClass("fixed");
                    }
            });

  });
  </script>
<!-- //script-for sticky-nav -->
  <div class="logo_products">
```

```
            <div class="container">
                <div class="w3ls_logo_products_left">
                    <h1><a href="index.php"><span>@Your</span>Door</a></h1>
                </div>
                <div class="w3ls_logo_products_left1">
                    <ul class="special_items">
                        <li><a href="events.php">Events</a><i>/</i></li>
                        <li><a href="about.php">About Us</a><i>/</i></li>
                        <li><a href="products.php">Best Deals</a><i>/</i></li>
                        <li><a href="services.php">Services</a></li>
                    </ul>
                </div>
                <div class="w3ls_logo_products_left1">
                    <ul class="phone_email">
                        <li><i class="fa fa-phone" aria-
hidden="true"></i>(+91)7000998867</li>
                        <li><i class="fa fa-envelope-o" aria-
hidden="true"></i><a href="mailto:door@grocery.com">door@grocery.com</a></li>
                    </ul>
                </div>
                <div class="clearfix"> </div>
            </div>
        </div>
<!-- //header -->
<!-- banner -->
    <div class="banner">
            <div class="w3l_banner_nav_left">
                <nav class="navbar nav_bottom">
                 <!-- Brand and toggle get grouped for better mobile display -->
                  <div class="navbar-header nav_2">
                        <button type="button" class="navbar-toggle collapsed navbar-
toggle1" data-toggle="collapse" data-target="#bs-megadropdown-tabs">
                            <span class="sr-only">Toggle navigation</span>
                            <span class="icon-bar"></span>
                            <span class="icon-bar"></span>
                            <span class="icon-bar"></span>
                        </button>
```

```
                    </div>
                    <!-- Collect the nav links, forms, and other content for toggling -->
                        <div class="collapse navbar-collapse" id="bs-megadropdown-
tabs">
                            <ul class="nav navbar-nav nav_1">
                                <li><a href="products.php">Branded
Foods</a></li>
                                <li><a
href="household.php">Households</a></li>
                                <li><a href="vegetables.php">Vegetables and
Fruits</a></li>
                                <li><a href="kitchen.php">Kitchen</a></li>
                                <li><a href="drinks.php">Beverages</a></li>

                                <li><a href="pet.php">Pet Food</a></li>
                                <li><a href="frozen.php">Frozen</a></li>
                                <li><a href="bread.php">Bread &
Bakery</a></li>
                            </ul>
                        </div><!-- /.navbar-collapse -->
                    </nav>
                </div>
                <div class="w3l_banner_nav_right">
                    <section class="slider">
                        <div class="flexslider">
                            <ul class="slides">
                                <li>
                                    <div
class="w3l_banner_nav_right_banner">
                                        <h3>Make your
<span>food</span> with Spicy.</h3>
                                        <div class="more">
                                            <a href="products.php"
class="button--saqui button--round-l button--text-thick" data-text="Shop now">Shop
now</a>
                                        </div>
                                    </div>
```

```
                                        </li>
                                        <li>
                                                <div
class="w3l_banner_nav_right_banner1">
                                                        <h3>Make your
<span>food</span> with Spicy.</h3>
                                                        <div class="more">
                                                                <a href="products.php"
class="button--saqui button--round-l button--text-thick" data-text="Shop now">Shop
now</a>
                                                        </div>
                                                </div>
                                        </li>
                                        <li>
                                                <div
class="w3l_banner_nav_right_banner2">
                                                        <h3>upto <i>50%</i> off.</h3>
                                                        <div class="more">
                                                                <a href="products.php"
class="button--saqui button--round-l button--text-thick" data-text="Shop now">Shop
now</a>
                                                        </div>
                                                </div>
                                        </li>
                                </ul>
                        </div>
                </section>
                <!-- flexSlider -->
                        <link rel="stylesheet" href="css/flexslider.css" type="text/css"
media="screen" property="" />
                        <script defer src="js/jquery.flexslider.js"></script>
                        <script type="text/javascript">
                        $(window).load(function(){
                          $('.flexslider').flexslider({
                                animation: "slide",
                                start: function(slider){
                                  $('body').removeClass('loading');
```

```
                                    }
                                });
                            });
                        </script>
                        <!-- //flexSlider -->
                </div>
                <div class="clearfix"></div>
    </div>
<!-- banner -->
    <div class="banner_bottom">
                        <div class="wthree_banner_bottom_left_grid_sub">
                        </div>
                        <div class="wthree_banner_bottom_left_grid_sub1">
                                <div class="col-md-4 wthree_banner_bottom_left">
                                        <div class="wthree_banner_bottom_left_grid">
                                                <img src="images/4.jpg" alt=" " class="img-
responsive" />
                                                <div
class="wthree_banner_bottom_left_grid_pos">
                                                        <h4>Discount Offer
<span>25%</span></h4>
                                                </div>
                                        </div>
                                </div>
                                <div class="col-md-4 wthree_banner_bottom_left">
                                        <div class="wthree_banner_bottom_left_grid">
                                                <img src="images/5.jpg" alt=" " class="img-
responsive" />
                                                <div class="wthree_banner_btm_pos">
                                                        <h3>introducing <span>best store</span>
for <i>groceries</i></h3>
                                                </div>
                                        </div>
                                </div>
                                <div class="col-md-4 wthree_banner_bottom_left">
                                        <div class="wthree_banner_bottom_left_grid">
```

```
                                        <img src="images/6.jpg" alt=" " class="img-
responsive" />
                                        <div class="wthree_banner_btm_pos1">
                                            <h3>Save <span>Upto</span> $10</h3>
                                        </div>
                                    </div>
                                </div>
                                <div class="clearfix"> </div>
                            </div>
                            <div class="clearfix"> </div>
  </div>
<!-- top-brands -->
  <div class="top-brands">
        <div class="container">
            <h3>Hot Offers</h3>
            <div class="agile_top_brands_grids">
                <div class="col-md-3 top_brand_left">
                    <div class="hover14 column">
                        <div class="agile_top_brand_left_grid">
                            <div class="tag"><img
src="images/tag.png" alt=" " class="img-responsive" /></div>
                            <div class="agile_top_brand_left_grid1">
                                <figure>
                                    <div class="snipcart-item
block" >
                                        <div
class="snipcart-thumb">
                                            <a
href="single.php"><img title=" " alt=" " src="images/1.png" /></a>

  <p>fortune sunflower oil</p>
                                            <h4>100
<span>$10</span></h4>
                                        </div>
                                        <div
class="snipcart-details top_brand_home_details">
```

```
                                                                    <form
action="#" method="post">
  <fieldset>
  <input type="hidden" name="cmd" value="_cart" />
  <input type="hidden" name="add" value="1" />
  <input type="hidden" name="business" value=" " />
  <input type="hidden" name="item_name" value="Fortune Sunflower Oil" />
  <input type="hidden" name="amount" value="7.99" />
  <input type="hidden" name="discount_amount" value="1.00" />
  <input type="hidden" name="currency_code" value="USD" />
  <input type="hidden" name="return" value=" " />
  <input type="hidden" name="cancel_return" value=" " />
  <input type="submit" name="submit" value="Add to cart" class="button" />
  </fieldset>
                                                                    </form>
                                                                </div>
                                                            </div>
                                                        </figure>
                                                    </div>
                                                </div>
                                            </div>
                                        </div>
                                        <div class="col-md-3 top_brand_left">
                                            <div class="hover14 column">
                                                <div class="agile_top_brand_left_grid">
```

```
                                        <div class="agile_top_brand_left_grid1">
                                            <figure>
                                                <div class="snipcart-item
block" >
                                                    <div
class="snipcart-thumb">
                                                        <a
href="single.php"><img title=" " alt=" " src="images/3.png" /></a>
  <p>basmati rise (5 Kg)</p>
  <h4>$11.99 <span>$15.00</span></h4>
                                                    </div>
                                                    <div
class="snipcart-details top_brand_home_details">
                                                        <form
action="#" method="post">
  <fieldset>
  <input type="hidden" name="cmd" value="_cart" />
  <input type="hidden" name="add" value="1" />
  <input type="hidden" name="business" value=" " />
  <input type="hidden" name="item_name" value="basmati rise" />
  <input type="hidden" name="amount" value="11.99" />
  <input type="hidden" name="discount_amount" value="1.00" />
  <input type="hidden" name="currency_code" value="USD" />
  <input type="hidden" name="return" value=" " />
  <input type="hidden" name="cancel_return" value=" " />
```

```
<input type="submit" name="submit" value="Add to cart" class="button" />

</fieldset>
                                                                    </form>
                                                                </div>
                                                            </div>
                                                        </figure>
                                                    </div>
                                                </div>
                                            </div>
                                        </div>
                                        <div class="col-md-3 top_brand_left">
                                            <div class="hover14 column">
                                                <div class="agile_top_brand_left_grid">
                                                    <div
class="agile_top_brand_left_grid_pos">
                                                        <img src="images/offer.png"
alt=" " class="img-responsive" />
                                                    </div>
                                                    <div class="agile_top_brand_left_grid1">
                                                        <figure>
                                                            <div class="snipcart-item
block">
                                                                <div
class="snipcart-thumb">
                                                                    <a
href="single.php"><img src="images/2.png" alt=" " class="img-responsive" /></a>
                                                                    <p>Pepsi
soft drink (2 Ltr)</p>

<h4>$8.00 <span>$10.00</span></h4>
                                                                </div>
                                                                <div
class="snipcart-details top_brand_home_details">
                                                                    <form
action="#" method="post">
```

```
<fieldset>

<input type="hidden" name="cmd" value="_cart" />

<input type="hidden" name="add" value="1" />

<input type="hidden" name="business" value=" " />

<input type="hidden" name="item_name" value="Pepsi soft drink" />

<input type="hidden" name="amount" value="8.00" />

<input type="hidden" name="discount_amount" value="1.00" />

<input type="hidden" name="currency_code" value="USD" />

<input type="hidden" name="return" value=" " />

<input type="hidden" name="cancel_return" value=" " />

<input type="submit" name="submit" value="Add to cart" class="button" />

</fieldset>
                                                                    </form>
                                                                </div>
                                                            </div>
                                                        </figure>
                                                    </div>
                                                </div>
                                            </div>
                                        </div>
                                        <div class="col-md-3 top_brand_left">
                                            <div class="hover14 column">
                                                <div class="agile_top_brand_left_grid">
                                                    <div
class="agile_top_brand_left_grid_pos">
```

```
                                        <img src="images/offer.png"
alt=" " class="img-responsive" />
                                    </div>
                                    <div class="agile_top_brand_left_grid1">
                                        <figure>
                                            <div class="snipcart-item
block">
                                                <div
class="snipcart-thumb">
                                                    <a
href="single.php"><img src="images/4.png" alt=" " class="img-responsive" /></a>
                                                    <p>dogs
food (4 Kg)</p>

  <h4>$9.00 <span>$11.00</span></h4>
                                                </div>
                                                <div
class="snipcart-details top_brand_home_details">
                                                    <form
action="#" method="post">

  <fieldset>

  <input type="hidden" name="cmd" value="_cart" />

  <input type="hidden" name="add" value="1" />

  <input type="hidden" name="business" value=" " />

  <input type="hidden" name="item_name" value="dogs food" />

  <input type="hidden" name="amount" value="9.00" />

  <input type="hidden" name="discount_amount" value="1.00" />

  <input type="hidden" name="currency_code" value="USD" />
```

```
<input type="hidden" name="return" value=" " />

<input type="hidden" name="cancel_return" value=" " />

<input type="submit" name="submit" value="Add to cart" class="button" />

</fieldset>
                                                                    </form>
                                                                </div>
                                                            </div>
                                                        </figure>
                                                    </div>
                                                </div>
                                            </div>
                                        </div>
                                        <div class="clearfix"> </div>
                                    </div>
                                </div>
                            </div>
<!-- //top-brands -->
<!-- fresh-vegetables -->
    <div class="fresh-vegetables">
            <div class="container">
                    <h3>Top Products</h3>
                    <div class="w3l_fresh_vegetables_grids">
                            <div class="col-md-3 w3l_fresh_vegetables_grid
w3l_fresh_vegetables_grid_left">
                                    <div class="w3l_fresh_vegetables_grid2">
                                            <ul>
                                                    <li><i class="fa fa-check" aria-
hidden="true"></i><a href="products.php">All Brands</a></li>
                                                    <li><i class="fa fa-check" aria-
hidden="true"></i><a href="vegetables.php">Vegetables</a></li>
                                                    <li><i class="fa fa-check" aria-
hidden="true"></i><a href="vegetables.php">Fruits</a></li>
```

```
                                        <li><i class="fa fa-check" aria-
hidden="true"></i><a href="drinks.php">Juices</a></li>
                                        <li><i class="fa fa-check" aria-
hidden="true"></i><a href="pet.php">Pet Food</a></li>
                                        <li><i class="fa fa-check" aria-
hidden="true"></i><a href="bread.php">Bread & Bakery</a></li>
                                        <li><i class="fa fa-check" aria-
hidden="true"></i><a href="household.php">Cleaning</a></li>
                                        <li><i class="fa fa-check" aria-
hidden="true"></i><a href="products.php">Spices</a></li>
                                        <li><i class="fa fa-check" aria-
hidden="true"></i><a href="products.php">Dry Fruits</a></li>
                                        <li><i class="fa fa-check" aria-
hidden="true"></i><a href="products.php">Dairy Products</a></li>
                                    </ul>
                                </div>
                            </div>
                            <div class="col-md-9 w3l_fresh_vegetables_grid_right">
                                <div class="col-md-4 w3l_fresh_vegetables_grid">
                                    <div class="w3l_fresh_vegetables_grid1">
                                        <img src="images/8.jpg" alt=" "
class="img-responsive" />
                                    </div>
                                </div>
                                <div class="col-md-4 w3l_fresh_vegetables_grid">
                                    <div class="w3l_fresh_vegetables_grid1">
                                        <div
class="w3l_fresh_vegetables_grid1_rel">
                                            <img src="images/7.jpg" alt=" "
class="img-responsive" />
                                            <div
class="w3l_fresh_vegetables_grid1_rel_pos">
                                                <div class="more m1">
                                                    <a
href="products.php" class="button--saqui button--round-l button--text-thick" data-
text="Shop now">Shop now</a>
                                                </div>
```

```
                                        </div>
                                    </div>
                                </div>
                                <div
class="w3l_fresh_vegetables_grid1_bottom">
                                    <img src="images/10.jpg" alt=" "
class="img-responsive" />
                                    <div
class="w3l_fresh_vegetables_grid1_bottom_pos">
                                        <h5>Special Offers</h5>
                                    </div>
                                </div>
                            </div>
                            <div class="col-md-4 w3l_fresh_vegetables_grid">
                                <div class="w3l_fresh_vegetables_grid1">
                                    <img src="images/9.jpg" alt=" "
class="img-responsive" />
                                </div>
                                <div
class="w3l_fresh_vegetables_grid1_bottom">
                                    <img src="images/11.jpg" alt=" "
class="img-responsive" />
                                </div>
                            </div>
                            <div class="clearfix"> </div>
                            <div class="agileinfo_move_text">
                                <div class="agileinfo_marquee">
                                    <h4>get <span class="blink_me">25%
off</span> on first order and also get gift voucher</h4>
                                </div>
                                <div class="agileinfo_breaking_news">
                                    <span> </span>
                                </div>
                                <div class="clearfix"></div>
                            </div>
                        </div>
                        <div class="clearfix"> </div>
```

```
                    </div>
                </div>
            </div>
        <!-- //fresh-vegetables -->
        <!-- newsletter -->
            <div class="newsletter">
                <div class="container">
                    <div class="w3agile_newsletter_left">
                        <h3>sign up for our newsletter</h3>
                    </div>
                    <div class="w3agile_newsletter_right">
                        <form action="#" method="post">
                            <input type="email" name="Email" value="Email"
onfocus="this.value = ';" onblur="if (this.value == '') {this.value = 'Email';}" required="">
                            <input type="submit" value="subscribe now">
                        </form>
                    </div>
                    <div class="clearfix"> </div>
                </div>
            </div>
        <!-- //newsletter -->
        <!-- footer -->
            <div class="footer">
                <div class="container">
                    <div class="col-md-3 w3_footer_grid">
                        <h3>information</h3>
                        <ul class="w3_footer_grid_list">
                            <li><a href="events.php"></a></li>
                            <li><a href="about.php">About Us</a></li>
                            <li><a href="products.php">Best Deals</a></li>
                            <li><a href="services.php">Services</a></li>
                            <li><a href="short-codes.php">Short Codes</a></li>
                        </ul>
                    </div>
                    <div class="col-md-3 w3_footer_grid">
                        <h3>policy info</h3>
                        <ul class="w3_footer_grid_list">
```

```
                                <li><a href="faqs.php">FAQ</a></li>
                                <li><a href="privacy.php">privacy policy</a></li>
                                <li><a href="privacy.php">terms of use</a></li>
                        </ul>
                </div>
                <div class="col-md-3 w3_footer_grid">
                        <h3>what in stores</h3>
                        <ul class="w3_footer_grid_list">
                                <li><a href="pet.php">Pet Food</a></li>
                                <li><a href="frozen.php">Frozen Snacks</a></li>
                                <li><a href="kitchen.php">Kitchen</a></li>
                                <li><a href="products.php">Branded Foods</a></li>
                                <li><a href="household.php">Households</a></li>
                        </ul>
                </div>

                <div class="clearfix"> </div>
                <div class="agile_footer_grids">
                        <div class="col-md-3 w3_footer_grid
agile_footer_grids_w3_footer">
                                <div class="w3_footer_grid_bottom">
                                        <h4>100% secure payments</h4>
                                        <img src="images/card.png" alt=" " class="img-
responsive" />
                                </div>
                        </div>
                        <div class="col-md-3 w3_footer_grid
agile_footer_grids_w3_footer">
                                <div class="w3_footer_grid_bottom">
                                        <h5>connect with us</h5>
                                        <ul class="agileits_social_icons">
                                                <li><a href="#" class="facebook"><i
class="fa fa-facebook" aria-hidden="true"></i></a></li>
                                                <li><a href="#" class="twitter"><i
class="fa fa-twitter" aria-hidden="true"></i></a></li>
                                                <li><a href="#" class="google"><i
class="fa fa-google-plus" aria-hidden="true"></i></a></li>
```

```
                                                    <li><a href="#" class="instagram"><i
class="fa fa-instagram" aria-hidden="true"></i></a></li>
                                                    <li><a href="#" class="dribbble"><i
class="fa fa-dribbble" aria-hidden="true"></i></a></li>
                                                </ul>
                                            </div>
                                        </div>
                                        <div class="clearfix"> </div>
                                    </div>
                                    <div class="wthree_footer_copy">
                                        <p>© 2016 @Your Door. All rights reserved | Design by <a
href="http://w3layouts.com/">Ancy Alexander</a></p>
                                    </div>
                                </div>
                            </div>
<!-- //footer -->
<!-- Bootstrap Core JavaScript -->
<script src="js/bootstrap.min.js"></script>
<script>
$(document).ready(function(){
    $(".dropdown").hover(
        function() {
            $('.dropdown-menu', this).stop( true, true ).slideDown("fast");
            $(this).toggleClass('open');
        },
        function() {
            $('.dropdown-menu', this).stop( true, true ).slideUp("fast");
            $(this).toggleClass('open');
        }
    );
});
</script>
<!-- here stars scrolling icon -->
    <script type="text/javascript">
            $(document).ready(function() {
                /*
                    var defaults = {
```

```
                        containerID: 'toTop', // fading element id
                        containerHoverID: 'toTopHover', // fading element hover id
                        scrollSpeed: 1200,
                        easingType: 'linear'
                        };
                */

                $().UItoTop({ easingType: 'easeOutQuart' });

                });
    </script>
<!-- //here ends scrolling icon -->
<script src="js/minicart.min.js"></script>
<script>
    // Mini Cart
    paypal.minicart.render({
            action: '#'
    });

    if (~window.location.search.indexOf('reset=true')) {
            paypal.minicart.reset();
    }
</script>
</body>
</html>
```

Admin

```
<?php
include 'db2.php';
```

```
$sql = mysqli_query($con, "SELECT * FROM `order_db` join product_tbl on
    product_tbl.product_id = order_db.order_id join register on register.login_id =
    order_db.login_id");
$sql2 = mysqli_query($con, "SELECT * FROM `login_tbl` INNER join register on
    login_tbl.login_id = register.login_id");


?>


<!DOCTYPE html>
<html lang="en">


<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>At Your Door</title>
  <link rel="stylesheet" type="text/css" href="css/styleadmin.css">
  <link rel="stylesheet" href="path/to/font-awesome/css/font-awesome.min.css">
  <link href="https://maxcdn.bootstrapcdn.com/font-awesome/4.2.0/css/font-awesome.min.css"
    rel="stylesheet">
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/@fortawesome/fontawesome-
    free@5.15.4/css/fontawesome.min.css" integrity="sha384-
    jLKHWM3JRmfMU0A5x5AkjWkw/EYfGUAGagvnfryNV3F9VqM98XiIH7VBGVoxVSc7
    " crossorigin="anonymous">


</head>


<body>
  <?php include 'sidebar.php'; ?>


  <div class="main">
    <?php include 'topbar.php'; ?>
```

```
<div class="cardBox">
   <div class="card">
      <div>
         <div class="numbers">1,042</div>
         <div class="cardName">Today's order</div>
      </div>
      <div class="iconBox">
         <i class="fa fa-eye" aria-hidden="true"></i>
      </div>
   </div>
   <div class="card">
      <div>
         <div class="numbers">208</div>
         <div class="cardName">Sales</div>
      </div>
      <div class="iconBox">
         <i class="fa fa-shopping-cart" aria-hidden="true"></i>
      </div>
   </div>
   <div class="card">
      <div>
         <div class="numbers">5</div>
         <div class="cardName">Comments</div>
      </div>
      <div class="iconBox">
         <i class="fa fa-comment" aria-hidden="true"></i>
      </div>
   </div>
   <div class="card">
```

```
            <div>
                <div class="numbers">3999</div>
                <div class="cardName">Earnings</div>
            </div>
            <div class="iconBox">
                <i class="fa fa-inr" aria-hidden="true"></i>
            </div>
        </div>
    </div>



    <div class="details">
        <div class="recentOrders">
            <div class="cardHeader">
                <h2>Recent Orders</h2>
                <a href="#" class="btn">View All</a>
            </div>
            <table>
                <thead>
                    <tr>
                        <td>Name</td>
                        <td>Product</td>
                        <td>Price</td>
                        <td>Status</td>
                    </tr>
                </thead>
                <tbody>
                    <?php while ($row = mysqli_fetch_assoc($sql)) { ?>
                        <tr>
                            <td><?php echo $row['uname']?></td>
```

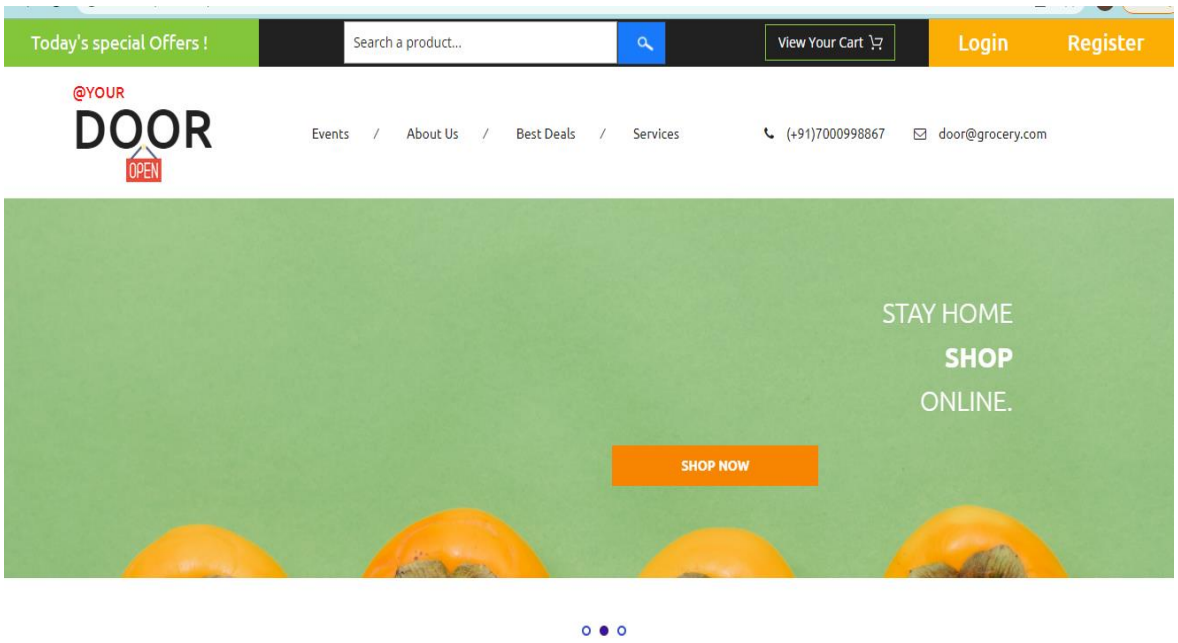
```
                    <td><?php echo $row['product_name']?></td>
                    <td><?php echo $row['price']?></td>
                    <td><span class="status inprogress"><?php echo
$row['status']?></span></td>
                </tr>
                <!-- <tr>
                <td>Mango</td>
                <td>100</td>
                <td>Due</td>
                <td><span class="status pending">Pending</span></td>
              </tr>
              <tr>
                <td>Maggi</td>
                <td>50</td>
                <td>Paid</td>
                <td><span class="status return">Return</span></td>
              </tr>
              <tr>
                <td>Sugar</td>
                <td>40</td>
                <td>due</td>
                <td><span class="status delivered">In Progress</span></td>
              </tr> -->
              <?php } ?>
            </tbody>
          </table>

        </div>
        <div class="recentCustomers">
          <div class="cardHeader">
```

```
                <h2>Recent Customers</h2>


            </div>
            <table>
                <tbody>
                    <?php while ($row = mysqli_fetch_assoc($sql2)) { ?>
                    <tr>
                        <td width="60px">
                            <div class="imgBx"><img src="profile.png"></div>
                        </td>
                        <td>
                            <h4><?php echo $row['uname']?><br><span><?php echo
    $row['email']?></span></h4>
                        </td>
                    </tr>
                    <?php } ?>
                </tbody>
            </table>
        </div>
    </div>
</div>


<script>
    function toggleMenu() {
        let toggle = document.querySelector('.toggle');
        let navigation = document.querySelector('.navigation');
        let main = document.querySelector('.main');
        toggle.classList.toggle('active');
        navigation.classList.toggle('active')
        main.classList.toggle('active')
```

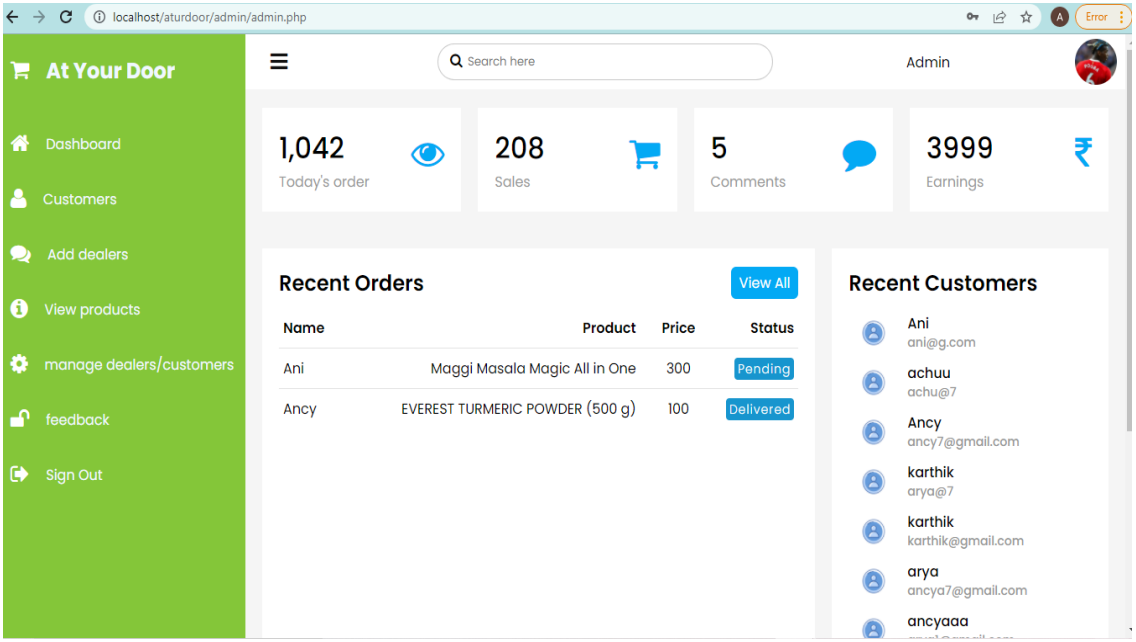
```
    }
  </script>

</body>

    </html>
```

9.2 Screen Shots

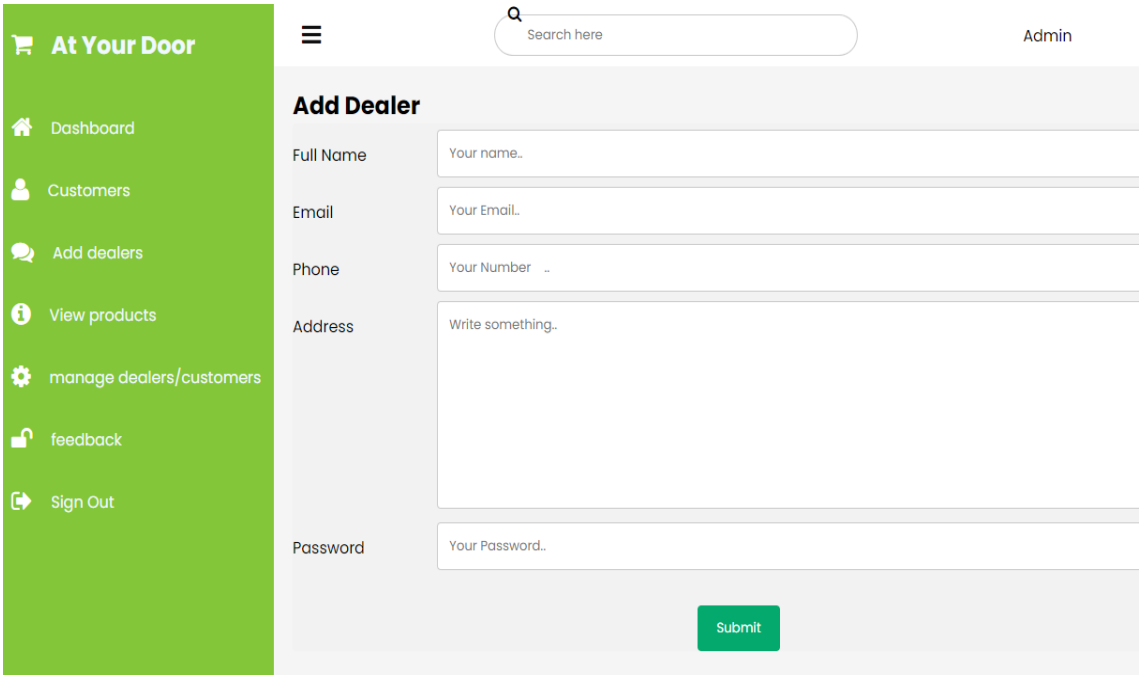
Home page



Admin



Add dealers


At Your Door
Dashboard
Customers
Add dealers
View products
manage dealers/customers
feedback
Sign Out
Search here
Admin
Add Dealer
Full Name
Your name..
Email
Your Email..
Phone
Your Number ..
Address
Write something..
Password
Your Password..
Submit

View products