

**SHREE SWAMINARAYAN COLLEGE OF COMPUTER SCIENCE**

**(Affiliated to M. K. Bhavnagar University)**

#### BIGSTORE BY

2021010 – ANKIT R. KHAVADIYA

2021025 – CHETAN N. PANARA

**UNDER GUIDANCE OF**

Mr. BHAVESH DHANDHUKIYA

**SUBMITTED TO**

SHREE SWAMINARAYAN COLLEGE OF COMPUTER SCIENCE

**FOR DEGREE OF**

BACHELOR OF COMPUTER APPLICATION

### **S**HREE **S**WAMINARAYAN **C**OLLEGE **O**F **C**OMPUTER **S**CIENCE

**(Affiliated to M. K. Bhavnagar University) GURUKUL CAMPUS, SARDARNAGAR, BHAVNAGAR-364001**



**AKNOWLEDGMENT**

The pleasure that follows the successful completion of project would remain incomplete without a word of gratitude for the people and without whose cooperation the achievement would remain a distant dream. Therefore, we would like to acknowledge with thanks, the individuals who had guided us in making this project.

Firstly, we are thankful to **Mr. Paresh Rathod**, I/C Principal at Shree Swaminarayan College Of Computer Science for his scholarly guidance, advice and encouragement and also providing necessary facilities to carry out the dissertation in prescribed period.

I convey my respectable thanks to **Dr. Kalpesh Gundigara**, Academic Head at Shree Swaminarayan College Of Computer Science for giving me continuous support throughout my project.

We would like to thanks to our project lecturer, **Mr. Sagar Vala**. he had guided us on how to complete this project. Every lecture in class, he always reminds and gives guideline on how to search information about this project.

We would like to thank our project guide **Mr. Bhavesh Dhandhukiya**, Lecturer at Shree Swaminarayan College Of Computer Science for his guidance and providing necessary information. Then, we express our thanks to our faculty members for their valuable guidance and intellectual suggestions.

Then, we would like to thanks to all our friends. Who had helped and shared ideas with us. They also gave us supports and advices.

Lastly but not least, we want to thanks to our parents who always prayed well forus and giving their time to hear our problem.



**PREFACE**

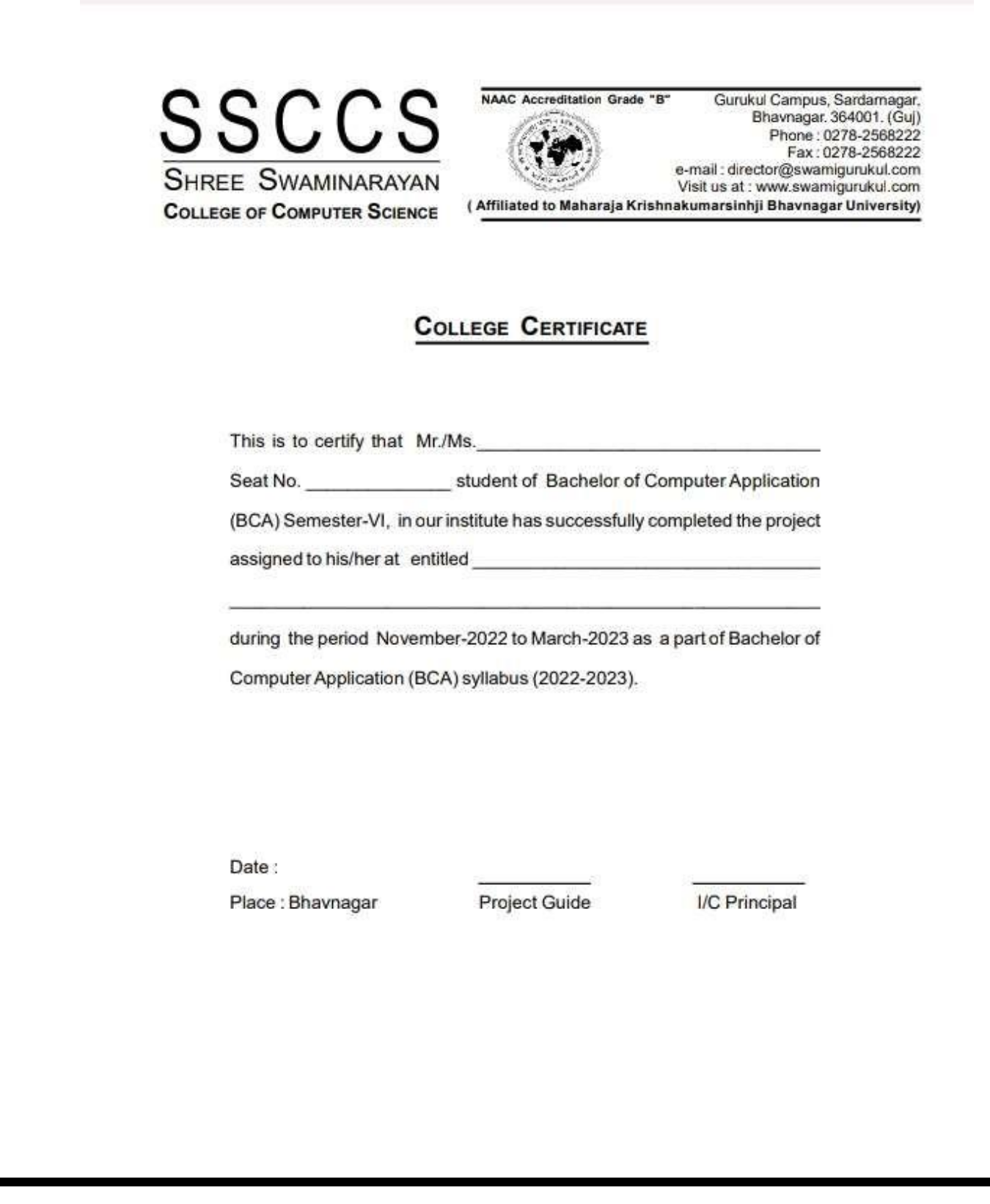
There is a project wo0rk as a subject in BCA-6. The main purpose of this project is to get practical as well as theoretical knowledge in any business firm or in any organization.

Project helps student in growing in the direction of practical implementation of any system. The project checks the patience and working ability of the student and help in achieving very important thing that is experience of project.

During this project work we try our level best to be the professional and also try to realize the fact that we are capable enough to face any challenges.

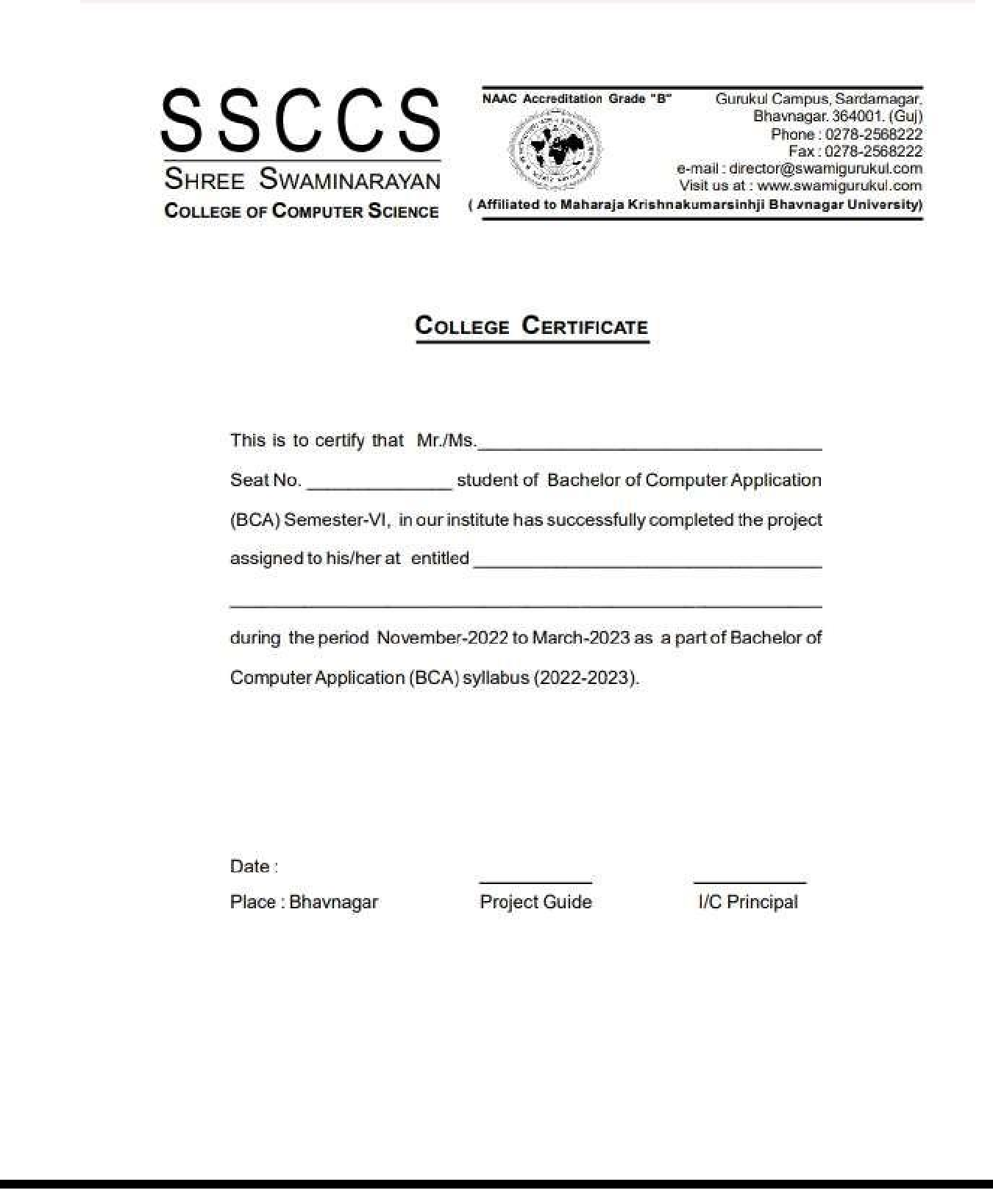
We study documents of various websites for the reference to our project. This is our pleasure to present this project work as our subject we learn a lot by doing this project. We realize that only the knowledge of theoretical subject is not sufficient. But practical knowledge is also involved.

The project report is about Courier Service. Generally, it is used to provide the highest quality and comprehensive food services in our clients' table. We also provide personal services like home delivery.





Page **4** of **79**





Page **5** of **79**



**TABLE OF CONTENT**

|  |  |
| --- | --- |
| **Topic name** | **Page no.** |
| **1. Introduction** | **10** |
| **1.1. Background** | **11** |
| **1.2. Objective** | **11** |
| **1.3. Advantages and Disadvantages** | **12** |
| 1.3.1. Advantages | **12** |
| 1.3.2. Disadvantages | **12** |
| **1.4. Purpose and Scope** | **12** |
| 1.4.1. Purpose | 12 |
| 1.4.2. Scope | 13 |
| **2. Requirement Analysis** | **14** |
| **2.1. Problem Definition** | **15** |
| **2.2. Requirement specification** | **15** |
| **2.2.1. Functional** | **15** |
| 2.2.1.1. Admin | **15** |
| 2.2.1.2. Customer | **15** |
| **2.2.2. Non functional** | 16 |
| 2.2.2.1. Performance | 16 |
| 2.2.2.2. Security | 16 |



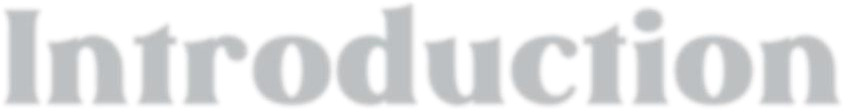
|  |  |
| --- | --- |
| 2.2.2.3. Availability | 16 |
| 2.2.2.4. Scalability | 16 |
| 2.2.2.5. Integrity | 16 |
| 2.2.2.6. Reliability | 16 |
| **2.3. Software and Hardware requirements** | **17** |
| 2.3.1. Hardware | 17 |
| 2.3.2. Software | 17 |
| **2.4. Planning and scheduling** | **18** |
| **2.5. Preliminary product description** | 19 |
| **2.6. References** | 19 |
| **3. System Design** | **20** |
| **3.1. Database Design** | **21** |
| 3.1.1. Data Dictionary | 21 |
| 3.1.2. Why Data Dictionary is Important? | 22 |
| 3.1.3. Database Design | 23 |
| **3.2. Data Flow Diagram** | **28** |
| 3.2.1. What is DFD | 28 |
| 3.2.2. Components of DFD | 28 |
| 3.2.3. Rules for creating DFD | 30 |
| 3.2.4. Levels of DFD | 31 |
| 3.2.5. Advantages of DFD | 31 |
| **3.3. Entity-Relationship (ER) Diagram** | **38** |



|  |  |
| --- | --- |
| **3.4. Input/Output Design** | **40** |
| **4. Testing & Implementation** | **59** |
| **4.1. Introduction to Testing** | **60** |
| 4.1.1. Errors can be present in the software due to the following reasons | 60 |
| **4.2. Characteristics of Software Test** | **61** |
| 4.2.1. High probability of detecting errors | 61 |
| 4.2.2. No redundancy | 61 |
| 4.2.3. Choose the most appropriate test | 62 |
| 4.2.4. Moderate | 62 |
| **4.3. The different types of tests** | **63** |
| 4.3.1. Unit tests | 63 |
| 4.3.2. Integration tests | 63 |
| 4.3.3. Functional tests | 64 |
| 4.3.4. End-to-end tests | 64 |
| 4.3.5. Acceptance testing | 65 |
| 4.3.6. Performance testing | 65 |
| 4.3.7. Smoke testing | 65 |
| **4.4. Testing Methods** | **66** |
| 4.4.1. Black-Box Testing | 66 |
| 4.4.2. White-Box Testing | 66 |
| 4.4.3. Grey-Box Testing | 66 |
| **4.5. Test Cases** | **68** |



|  |  |
| --- | --- |
| 4.5.1. Test case 1: Customer Registration | 68 |
| 4.5.2. Test case 2: Customer Login | 69 |
| 4.5.3. Test case 3: Add new Admin | 70 |
| 4.5.4. Test case 4: Add new customer | 71 |
| 4.5.5. Test case 5: Add new category | 72 |
| 4.5.6. Test case 6: Add new product | 74 |
| **5. Conclusion** | **77** |
| **5.1. Conclusion** | **78** |
| **5.2. Limitations and Future Enhancement** | **78** |
| 5.2.1. Limitations | 78 |
| 5.2.2. Future Enhancement | 79 |
| **5.3. Bibliography** | **79** |



# Introduction



## INTRODUCTION

##### Background:

* In modern age, as time increase, needs & requirements of the person are also increased. They want more facility & try to do their task quickly & within time. Butthey cannot get all the things at nearest market or area, so they have to import the things from any place in the world.
* The Grocery store is one of the solutions of these problems. It is used to purchase grocery items will be able to shop any time of the day.

##### Objective:

* The objective of this system is to save time and effort for the consumer. The objective of this study is to help the consumer of Savemore to make their ordering more convenient and easier. For the customer,it can minimize the workload and effort of roaming around the grocery store. They can search the grocery items that they’re looking for. And they can buy it.
* Its administration can take immediateorders and provide a receipt which will include all the details of the grocery products along with appropriate price to their customers.
* Thus, saving time and eliminating line making process.



##### Advantages & Disadvantages of system

###### Advantages

* Easy to update information.
* Compare Price To Save Money
* Customer supports (24\*7)
* Helps Save Time and Money o Greater Variety of Choice

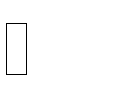
###### Disadvantages

* Compromise With the Grocery Quality
* Customer can not check items physically.

##### Purpose & scope

###### Purpose

* The main purpose of online grocery is to create and develop new models, and to optimize the relationships between a grocery company and its customers.
* whenever you purchase your groceries online you will be able to shop any time of the day or night at your own convenience, regardless of what the weather outside may be and still get evething that you need and want.



###### Scope

* The customer can easily see for the Grocery products in shop and can add immediately toher/his shopping cart.
* The customer can pay through UPI and Cash on Delivery.
* Customer can see the order details and the actions done to her/his orders. Customers can easily avail discounts on grocery products.
* The system can print the receipt of the customer’s order
* The customer can print her/his own receipt

Page **13** of **79**



Requirement

Analysis



## REQUIREMENT ANALYSIS

#### Problem Defination

* This system is used for daily activities such as purchase grocery items ,products history, maintain customers details and their orders details.
* It is very difficult to do this process manually. Hence it is recommended to computerize the process by developing the relative software as the world is turning into information and technology; computerization becomes a necessity in all walks of life.
* When a customer can buy their products using a grocery store, they want to know if their products have been moved to their right location or not. If not then by what time will he be shifted and where is he now. It has to wait.
* Taking all this information manually is very difficult and time taking process. To handle all these activities, include various processes and paper work from the management also have to be involved.

#### Requierment Specification

##### Functional

###### Admin

* + - * + Admin can add/update/delete the products/category/orders from the website. Admin manage the system.

###### Customers

* + - * + Customers can view the products with its detail. He can add products in the add to cart page in this website.he can see total price of their buy products and checkout option to order placed.Customer can also manage their account.



##### Non Funcational

###### Performance

* + - * + Although the system is a simple one, a literate organizer who is able to
        + understand simple computer process is needed to run the system.

###### Security

* + - * + The users have respective accounts with password that enables only the authorize user to login onto the system.
        + Password are required so that no one else can access the system or database. In the case of the administrator needs to have the adequate knowledge about maintaining database should the system encounter problems.

###### Availability

* + - * + The system is accessible for a user at a given point in time

###### Scalability

* + - * + The system should easily allow for future improvements and upgrades.

###### Integrity

* + - * + The system should make sure that stored data is altered or corrupted.

###### Reliability

* + - * + The system should be reliable to perform user tasks



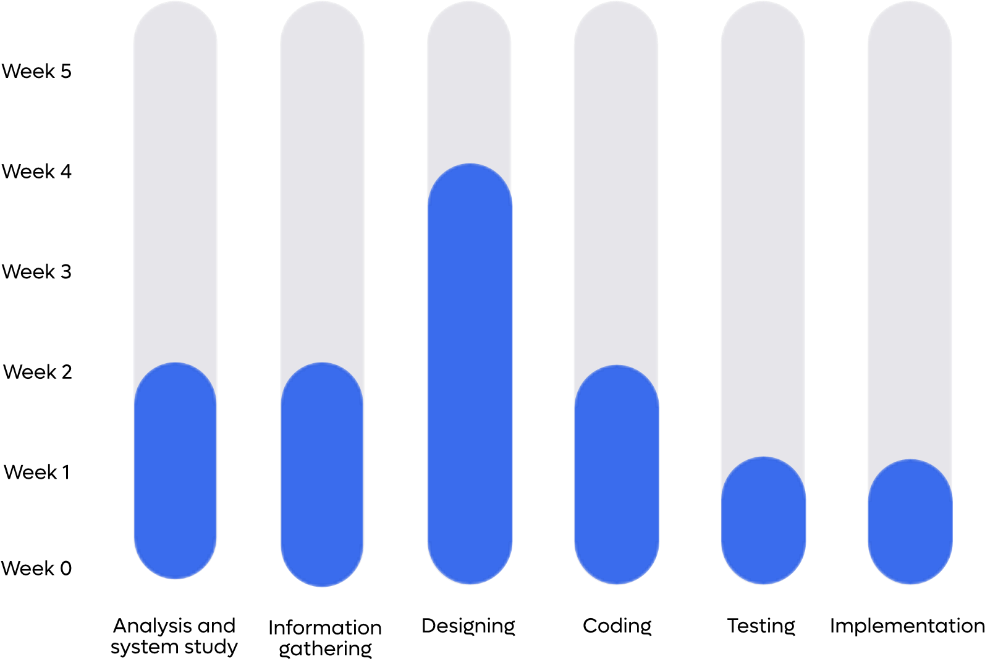
#### S/w & h/w requierments

* + 1. **Software Requirement**

|  |  |
| --- | --- |
| S/W | Version |
| O.S | Microsoft XP, Window-7,Window-10 |
| Browser | Google Chrome, Mozilla FireFox |
| XAMP | MySQL |
| Fronted | PHP,HTML,CSS,JS,Bootstrapt,Jquery,Ajex |
| Backend | MYSQL |

* + 1. **Hardware Requirements**

|  |  |
| --- | --- |
| Component | Requirement |
| Processor | Pentinum 4 OR Dualcore |
| RAM | 592MB,1GB,4GM |
| Hard disk space | 50GB |



#### Planning & scheduling



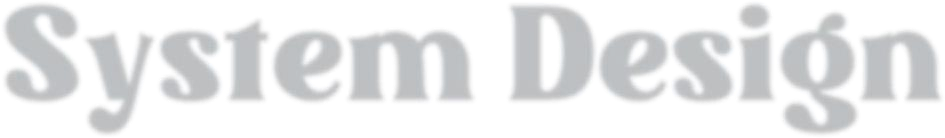
* 1. **Preliminary product description (Investication)**
* **PHP** : For communicate with database and maintain session of user
* **HTML** : Creating a web page
* **CSS** : For formatting and designing of web page
* **Bootstrap** : Use to create Responsive and mobile friendly website.
* **JavaScript** : For client side scripting
* **MYSQL** : To Store and maintain data in database

###### Tools :

* + XAMMP : XAMMP sever for localhost and run PHP script o
  + VS Code : For coding in html or PHP file o Browser : Use to view output or was page.

#### References

* [www.youtube.com](http://www.youtube.com/)
* Conceptual Programming Using PHP



System Design



## SYSTEM DESIGN

#### Database Design

##### Data Dictionary

* A data dictionary contains metadata i.e., data about the database.
* The data dictionary is very important as it contains information such as what is in the database, who is allowed to access it, where is the database physically stored etc
* The users of the database normally don't interact with the data dictionary, it is only handled by the database administrators.
* The data dictionary in general contains information about the following :
  + Names of all the database tables and their schemas
  + Details about all the tables in the database, such as their owners, theirsecurity constraints, when they were created etc
  + Physical information about the tables such as where they are stored andhow.
  + Table constraints such as primary key attributes, foreign key information etc.
  + Information about the database views that are visible.



##### Why Data Dictionary is important?

* To communicate a common meaning for all system elements.
* To document the features of the system.
* To facilitate analysis of the details on order to evaluate characteristics and determine where system changes should be made.
* There is a style of data dictionaries known as a middleware data dictionary.
* Middleware is computer software that connects software components or applications. The software consists of a set of services that allows multiple processes running on one or more machines to interact.
* Traditional data dictionaries provide structure and basic function to the database. Middleware data dictionaries are located within the DBMS itself and operate on a higher level. Middleware data dictionaries can provide alternate entity relationship structures that can be tailored to fit different users that interact with the same database.
* Middleware data dictionaries can also assist in query optimization as well as distributed database.
* Middleware also helps database designer by reducing the amount of time it takes to create form, queries, reports, menu and many other database components. they do this by automatically generated sql code for commonitem such as forms and views.
* Some middle wear database dictionary can also help with data security as well as database optimization.



##### Database design

**Database Name: bigstore**

* **Table 1:** admin
  + This table stores information about admin.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Srno | Field Name | Datatype | Size | Constrains | Description |
| 1. | id | int | 10 | Primary key | Admin id |
| 2. | fullname | varchar | 30 | Not null | Full name |
| 3. | email | varchar | 30 | Not null | Email id |
| 4. | password | varchar | 20 | Not null | Password |

* **Table 2:** customer
  + This table stores information about customer.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Srno | Field Name | Datatype | Size | Constrains | Description |
| 1. | id | int | 10 | Primary key | customer id |
| 2. | fullname | varchar | 30 | Not null | Full name |
| 3. | email | varchar | 30 | Not null | Email id |
| 4. | password | varchar | 20 | Not null | Password |



* **Table 3:** category
  + This table stores information about category.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Srno | Field Name | Datatype | Size | Constrains | Description |
| 1. | category\_id | int | 10 | Primary key | Category id |
| 2. | category\_name | varchar | 30 | Not null | Category name |
| 3. | category\_image | varchar | 255 | Not null | Category image |
| 4. | discount | int | 10 | Not null | Discount |
| 5. | active | varchar | 10 | Not null | Active |

* **Table 4:** product
  + This table stores information about product.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Srno | Field Name | Datatype | Size | Constrains | Description |
| 1. | id | int | 10 | Primary key | Product Id |
| 2. | product\_name | varchar | 30 | Not null | Product name |
| 3. | company\_name | varchar | 30 | Not null | Company name |
| 4. | category | varchar | 30 | Not null | Category |
| 5. | sub\_category | varchar | 30 | Not null | Sub Category |
| 6. | price | int | 10 | Not null | Price |
| 7. | mrp | int | 10 | Not null | MRP |
| 8. | product\_image | varchar | 255 | Not null | Product image |
| 9. | description | text |  | Not null | Description |
| 10. | active | varchar | 10 | Not null | Active |



* **Table 5:** sub\_category
  + This table stores information about sub category.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Srno | Field Name | Datatype | Size | Constrains | Description |
| 1. | scategory\_id | int | 10 | Primary key | Sub category id |
| 2. | category\_name | varchar | 30 | Not null | Category name |
| 3. | sub\_category\_name | varchar | 30 | Not null | Sub category image |
| 4. | active | varchar | 10 | Not null | Active |

* **Table 6:** cart
  + This table stores information about add to cart.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Srno | Field Name | Datatype | Size | Constrains | Description |
| 1. | id | int | 10 | Primary key | Cart Id |
| 2. | pname | varchar | 40 | Not null | Product name |
| 3. | pimage | varchar | 255 | Not null | product  image |
| 4. | price | int | 10 | Not null | Price |
| 5. | mrp | int | 10 | Not null | MRP |
| 6. | quantity | int | 10 | Not null | Quantity |
| 7. | user\_id | int | 10 | Not null | User id |

Page **25** of **79**



* **Table 7:** order\_detail
  + This table stores information about order detail.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Srno | Field Name | Datatype | Size | Constrains | Description |
| 1. | order\_id | int | 10 | Primary key | Order id |
| 2. | fullname | varchar | 30 | Not null | Full name |
| 3. | email | varchar | 30 | Not null | Email id |
| 4. | order\_date | date | - | Not null | Order date |
| 5. | status | varchar | 20 | Not null | Status |
| 6. | address | varchar | 100 | Not null | Address |
| 7. | city | varchar | 20 | Not null | City |
| 8. | pincode | int | 10 | Not null | Pincode |
| 9. | country | varchar | 20 | Not null | Country |
| 10 | phoneno | int | 10 | Not null | phoneno |
| 11. | payment | varchar | 20 | Not null | Payment |
| 12. | user\_id | int | 10 | Not null | User id |
| 13. | product\_name | text | - | Not null | Product name |
| 14. | quantity | varchar | 100 | Not null | Quantity |
| 15. | mrp | varchar | 100 | Not null | MRP |
| 16. | price | varchar | 100 | Not null | Price |
| 17 | total\_amount | int | 10 | Not null | Total amount |



* **Table 8:** bill
  + This table stores information about Bill.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Srno | Field Name | Datatype | Size | Constrains | Description |
| 1. | id | int | 10 | Primary key | Bill id |
| 2. | fullname | varchar | 30 | Not null | Full name |
| 3. | order\_date | date |  | Not null | Order date |
| 4. | address | varchar | 100 | Not null | address |
| 5. | city | varchar | 20 | Not null | city |
| 6. | pincode | Int | 10 | Not null | Pincode |
| 7. | country | varchar | 20 | Not null | country |
| 8. | phoneno | int | 10 | Not null | phoneno |
| 9. | payment | varchar | 20 | Not null | Parment |
| 10. | user\_id | Int | 10 | Not null | User id |
| 11. | product\_name | text |  | Not null | Product name |
| 12. | quantity | varchar | 100 | Not null | Quantity |
| 13. | price | varchar | 100 | Not null | Total price |
| 14. | mrp | varchar | 100 | Not null | MRP |
| 15 | Total\_amount | int | 10 | Not null | Total amount |

* **Table 9:** Contact
  + This table stores information about Contact.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Srno | Field Name | Datatype | Size | Constrains | Description |
| 1. | contact\_id | int | 10 | Primary key | contact id |
| 2. | fullname | varchar | 30 | Not null | Full name |
| 3. | email | varchar | 30 | Not null | Email id |
| 4. | subject | varchar | 30 | Not null | subject |
| 5, | message | text | - | Not null | message |
| 6, | user\_id | int | 10 | Not null | User id |



#### Data Flow Diagram

##### What is DFD (Data Flow Diagram)?

* DFD is the abbreviation for Data Flow Diagram.
* The flow of data of a system or a process is represented by DFD.
* It also gives insight into the inputs and outputs of each entity and the processitself.
* DFD does not have control flow and no loops or decision rules are present.
* Specific operations depending on the type of data can be explained by a flowchart.
* Data Flow Diagram can be represented in several ways.
* The DFD belongs to structured-analysis modelling tools.
* Data Flow diagrams are very popular because they help us to visualize the major steps and data involved in software- system processes.

##### Components of DFD

* DFD is the abbreviation for Data Flow Diagram.
* The two main types of notations used for data flow diagrams are Yourdon Coad and Gane-Sarson, both named after their creators, all experts who helped develop DFD methodology: Ed Yourdon, Peter Coad, Chris Gane and Trish Sarson.
* There are some differences in style between the notation types.
* For example, Yourdon and Coad notation uses circles to represent processes, whereas Gane and Sarson notation use rectangles with rounded corners.
* Another variation is the symbol used for data stores Yourdon and Coad use parallel lines while Gane and Sarson notation uses an open-ended rectangle.
  + Because DFD symbols vary, it’s important to be consistent with whatevernotation you choose in order to avoid confusion.
  + If you’re using DFD software, it will likely dictate which set of symbols areavailable to use.
  + All data flow diagrams include four main elements: entity, process, datastore and data flow.

##### External Entity

* + Also known as actors, sources or sinks, and terminators, external entities produce and consume data that flows between the entity and thesystem being diagrammed.
  + These data flows are the inputs and outputs of the DFD.
  + Since they are external to the system being analysed, these entities are typically placed at the boundaries of the diagram.
  + They can represent another system or indicate a subsystem.

##### Process

* An activity that changes or transforms data flows.
* Since they transform incoming data to outgoing data, all processes musthave inputs and outputs on a DFD.
* Since This symbol is given a simple name based on its function, such as “Ship Order,” rather than being labelled “process” on a diagram.
* In Gane-Sarson notation, a rectangular box is used and may be labelled with a reference number, location of where in the system the process occurs and a short title that describes its function.
* Processes are typically oriented from top to bottom and left to right ona data flow diagram.



##### Data Store



* + - A data store does not generate any operations but simply holdsdata forlater access.

|  |  |  |
| --- | --- | --- |
|  | Yourdon and Coad | Gane and Sasron |
| External Entity |  |  |
| Process |  |  |
| Data Storage |  |  |
| Data Flow |  |  |

##### Rules for creating DFD

* The name of the entity should be easy and understandablewithout any extraassistance (like comments).
* The processes should be numbered or put in ordered list tobe referredeasily.
* The DFD should maintain consistency across all the DFDlevels.
* A single DFD can have maximum processes up to 9 andminimum 3processes.

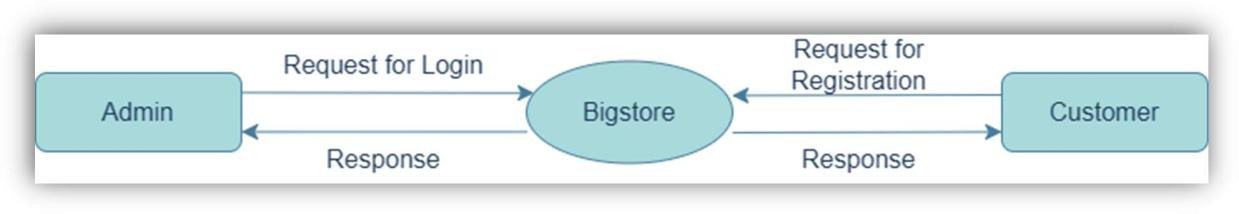


##### Levels of DFD

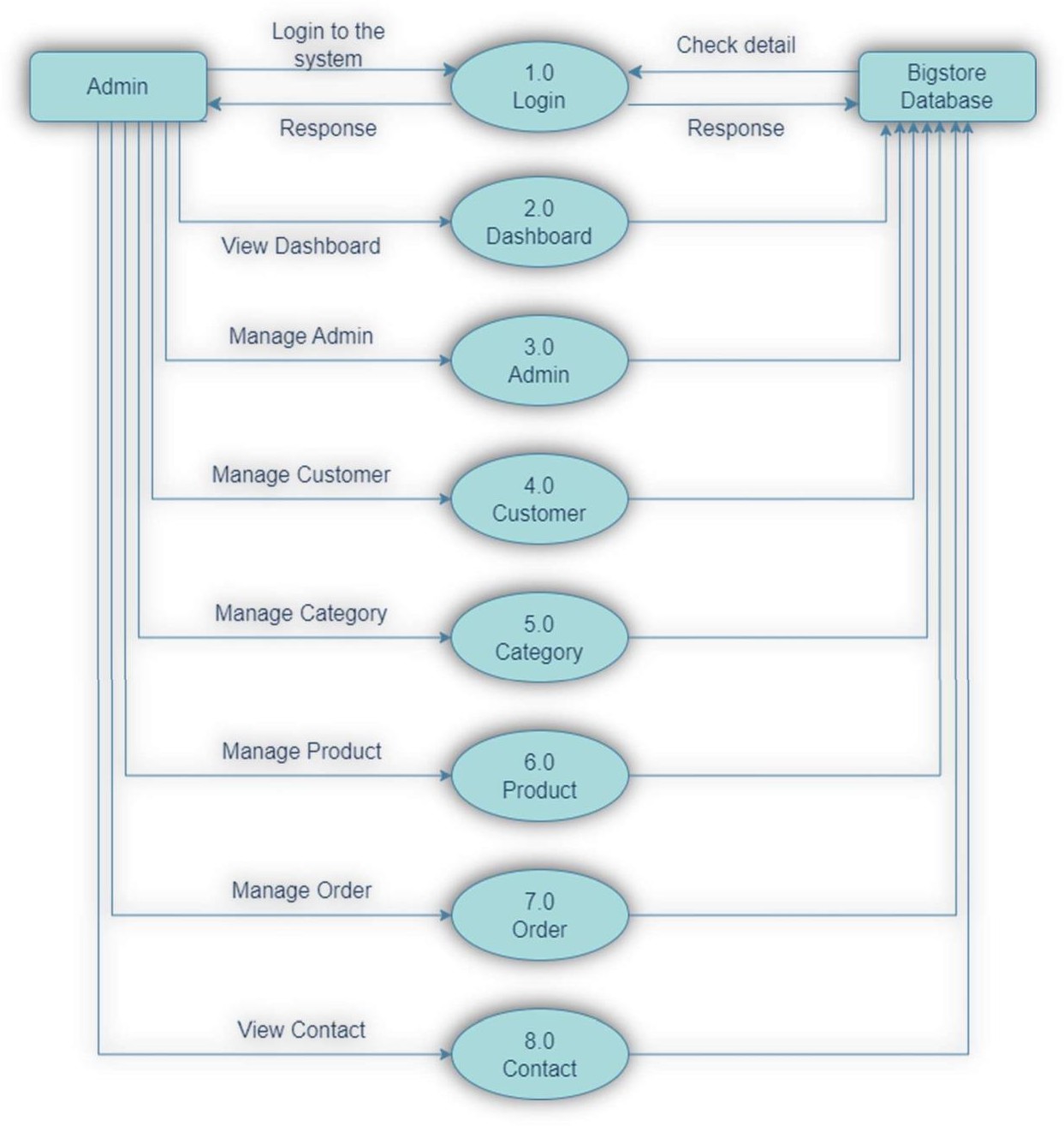
* DFD uses hierarchy to maintain transparency thus multilevel DFD’s can be created.
  + Levels of DFD are as follows:
    - 0-level DFD
    - 1-level DFD
    - 2-level DFD:

##### Advantages of DFD

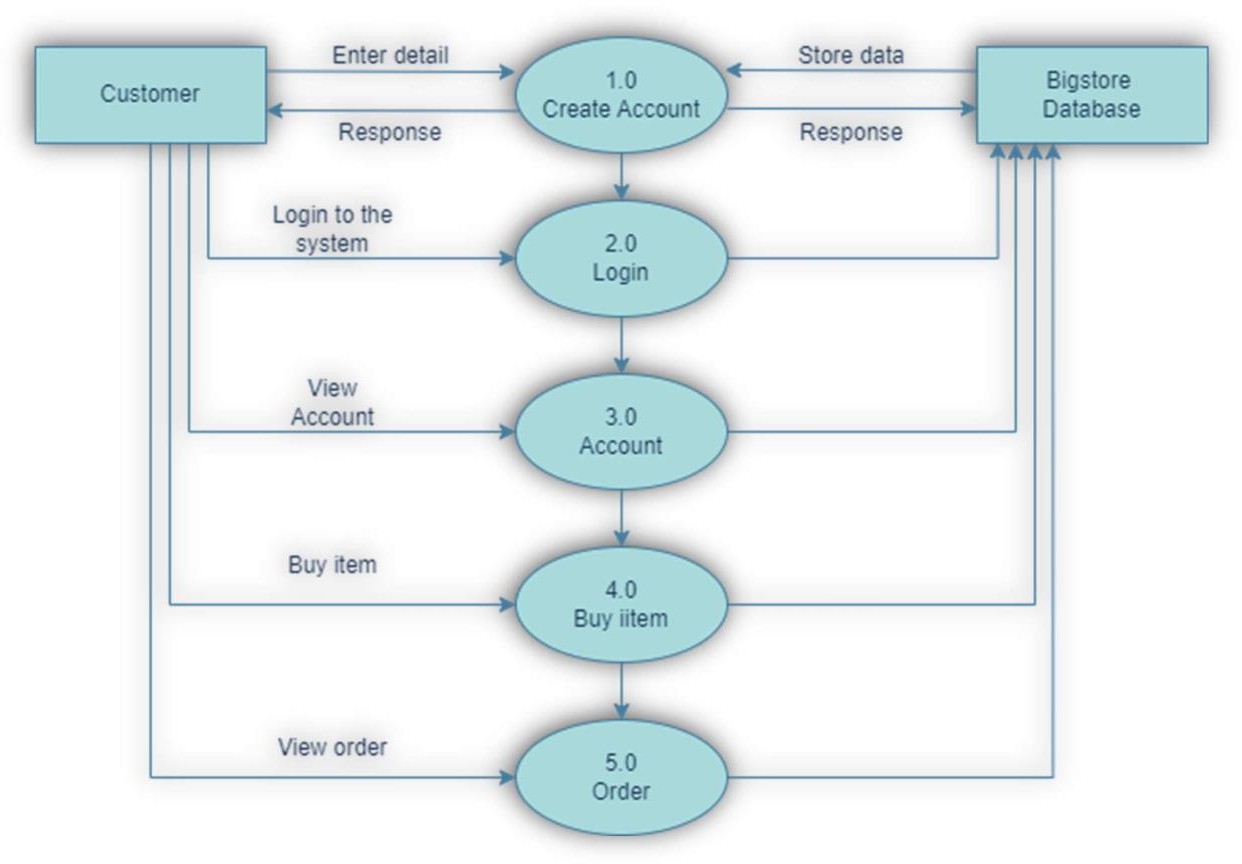
* It helps us to understand the functioning and the limits of a system.
* It is a graphical representation which is very easy to understand as it helps visualize contents.
* Data Flow Diagram represent detailed and well explained diagram of system components.
* It is used as the part of system documentation file.
* Data Flow Diagrams can be understood by both technical or nontechnical person because they are very easy to understand.



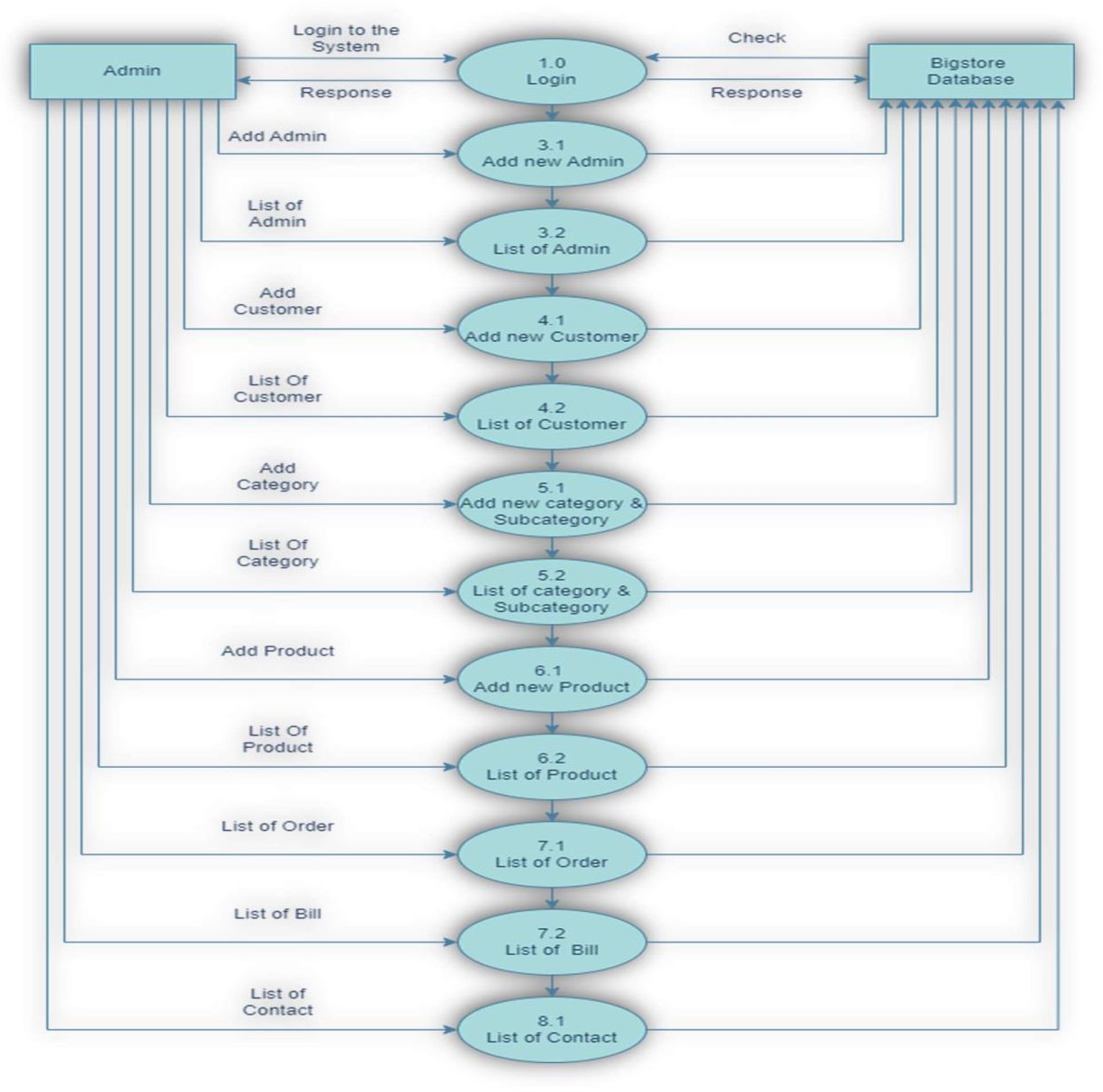
* **DFD Level 0 Or Context level**



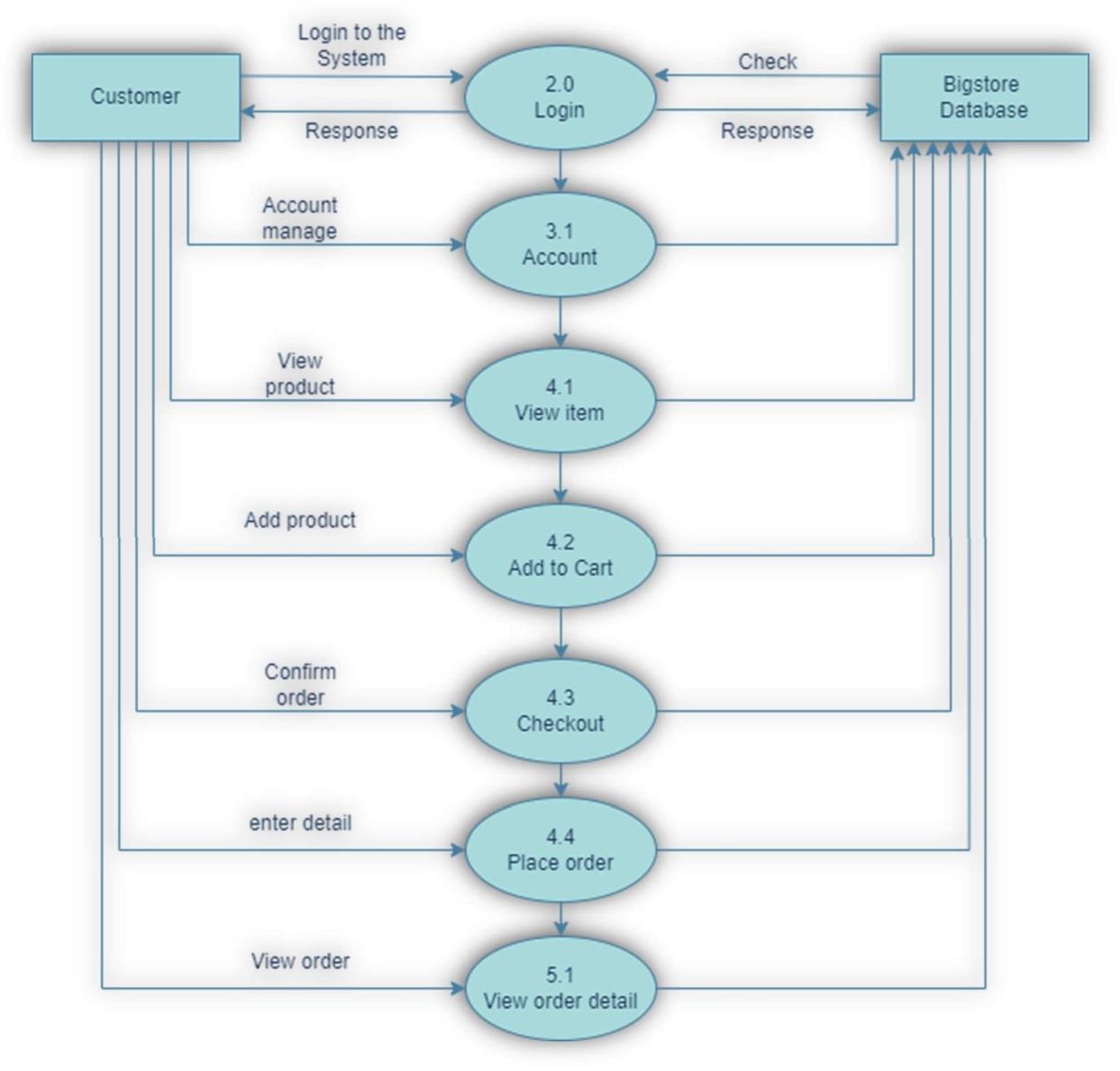
* **1st Level DFD(Admin)**



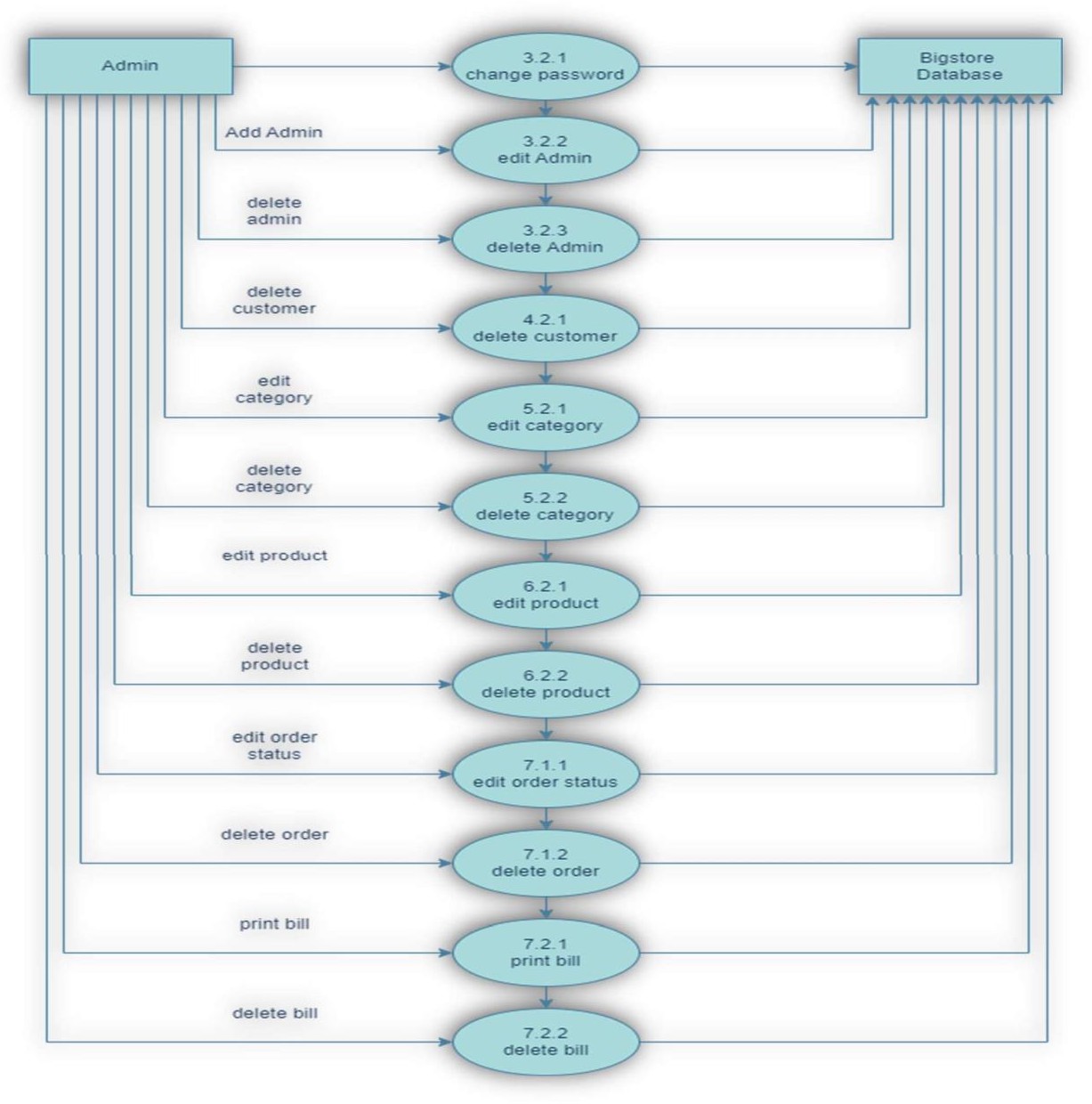
* **1st Level DFD(Customer)**



* **2st Level DFD(Admin)**



* **2st Level DFD(Customer)**

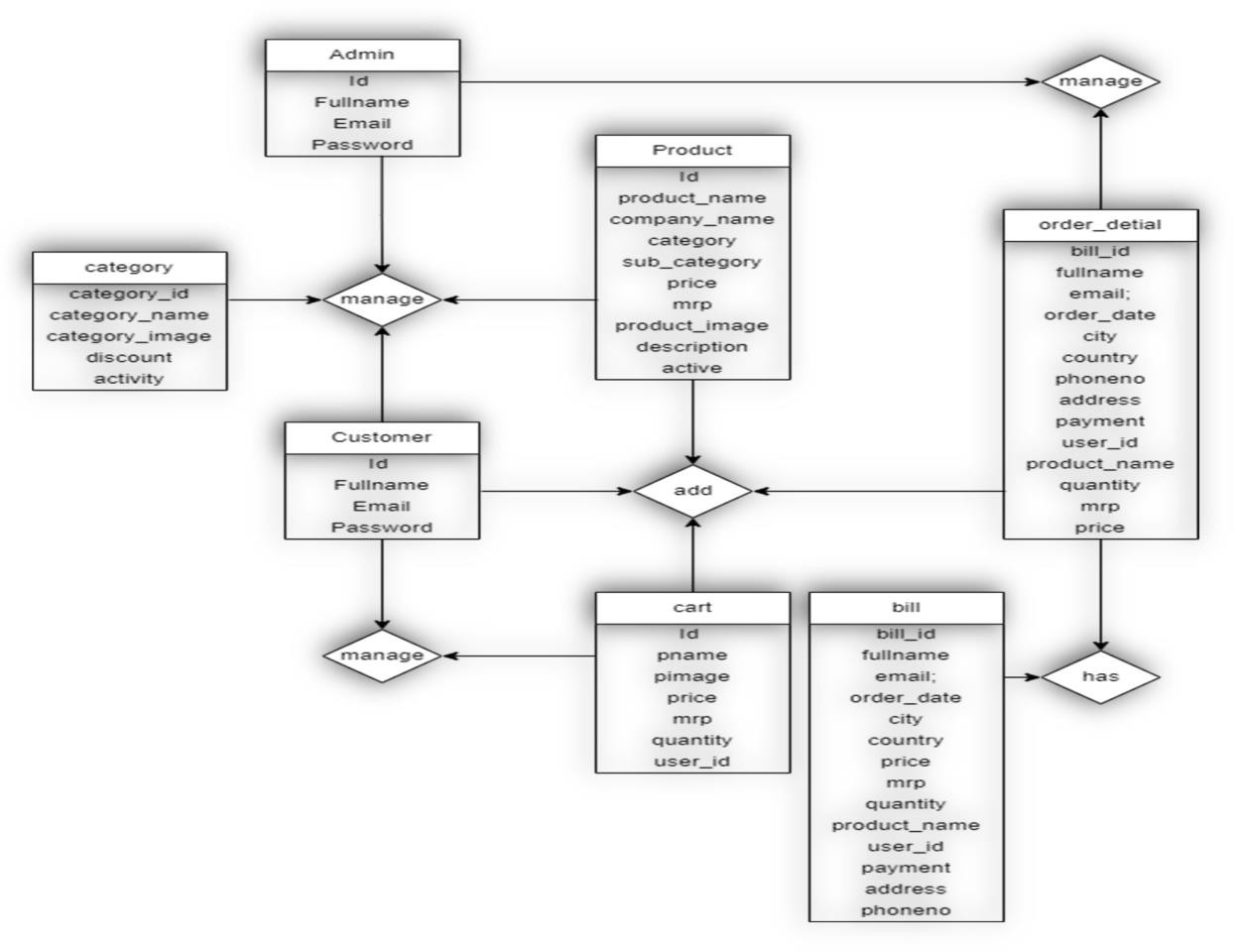


* **3st Level DFD(Admin)**

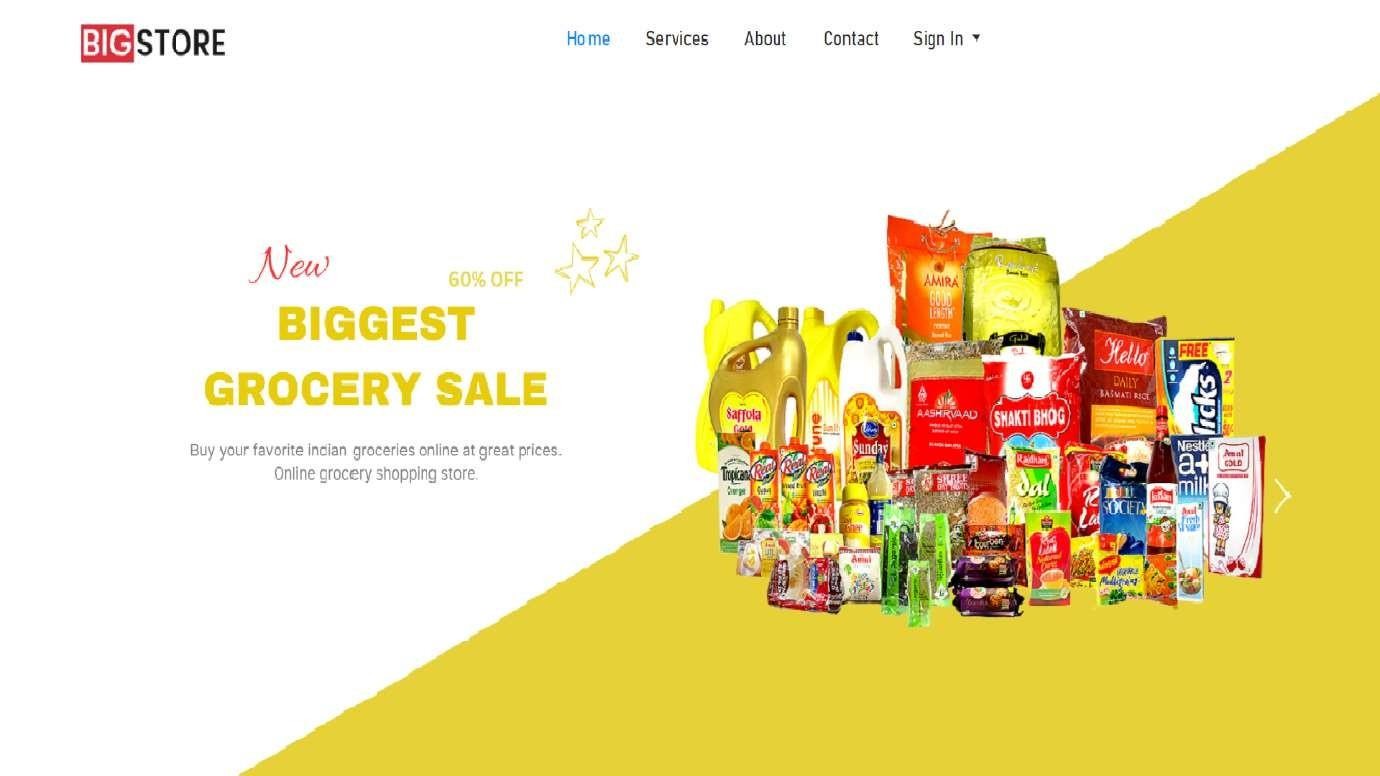


#### Entity-Relationship (ER) Diagram

* Entity Relationship Diagram, also known as ER diagram or ER model, is a type of structural diagram for use in database design.
* An ERD contains different symbols and connectors that visualize two important information: The major entities within the system scope, and the inter- relationships among these entities.
* An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system.
* ER diagrams also are often used in conjunction with dataflow diagrams (DFDs), which map out the flow of information for processes or systems. Let us of learn how the ER model is represented by means of ER diagram.
* Any object for example, entities, relationship sets, and attributes of relationship of sets, can be represented with the help of an ER diagram. An entity-relationship diagram (ERD) is crucial to creating a good database design.
* It is used as a high-level logical data model, which is useful in developing a conceptual design for databases. An entity is a real-world item or concept that exists on its own. Entities are equivalent to database tables in a relational database, with each row of the table representing an instance of that entity. An attribute of an entity is a particular property that describes the entity.
* A relationship is the association that describes the interaction between entities. Cardinality, in the context of ERD, is the number of instances of one entity that can, or must, be associated with each instance of another entity. In general, there may be one-to-one, one-to-many, or many-to many relationships.
* Here, are prime reasons for using the ER Diagram Helps you to define terms related to entity relationship modelling Provide a preview of how all your tables should connect, what fields are going to be on each table helps to describe entities,attributes, relationships.
* ER diagrams are translatable into relational tables which allows you to build databases quickly.

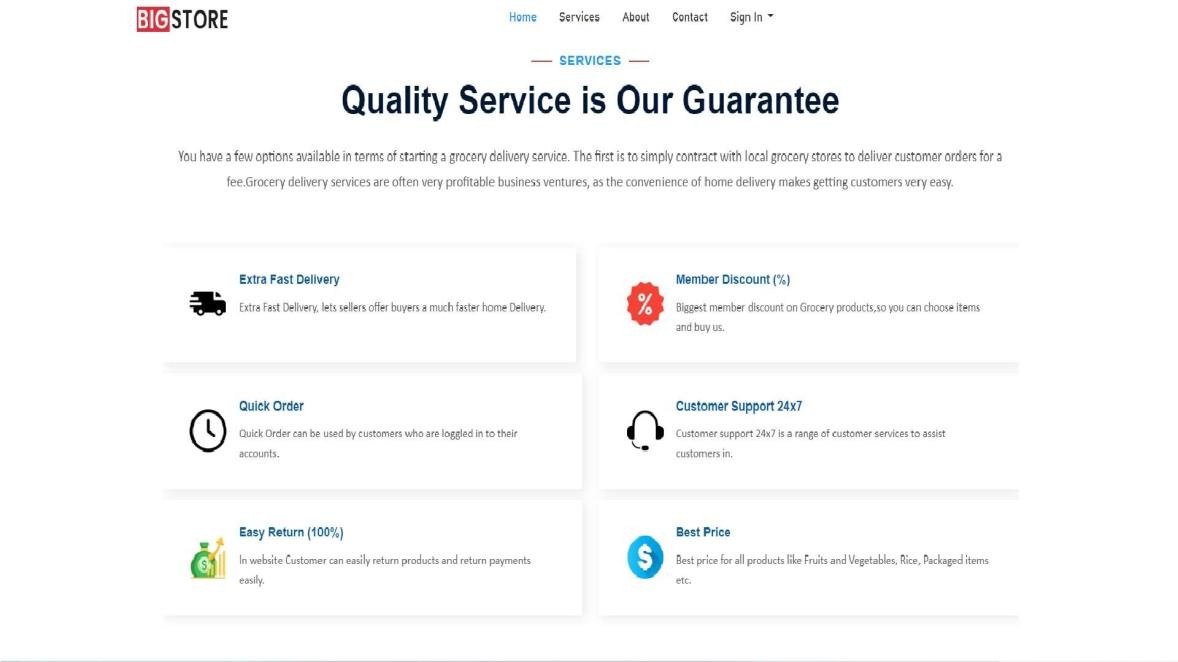


* ER diagrams can be used by database designers as a blueprint for implementing data in specific software applications.
* The database designer gains a better understanding of the information to be contained in the database with the help of ERP diagram.
* ERD is allowed you to communicate with the logical structure of the database tousers

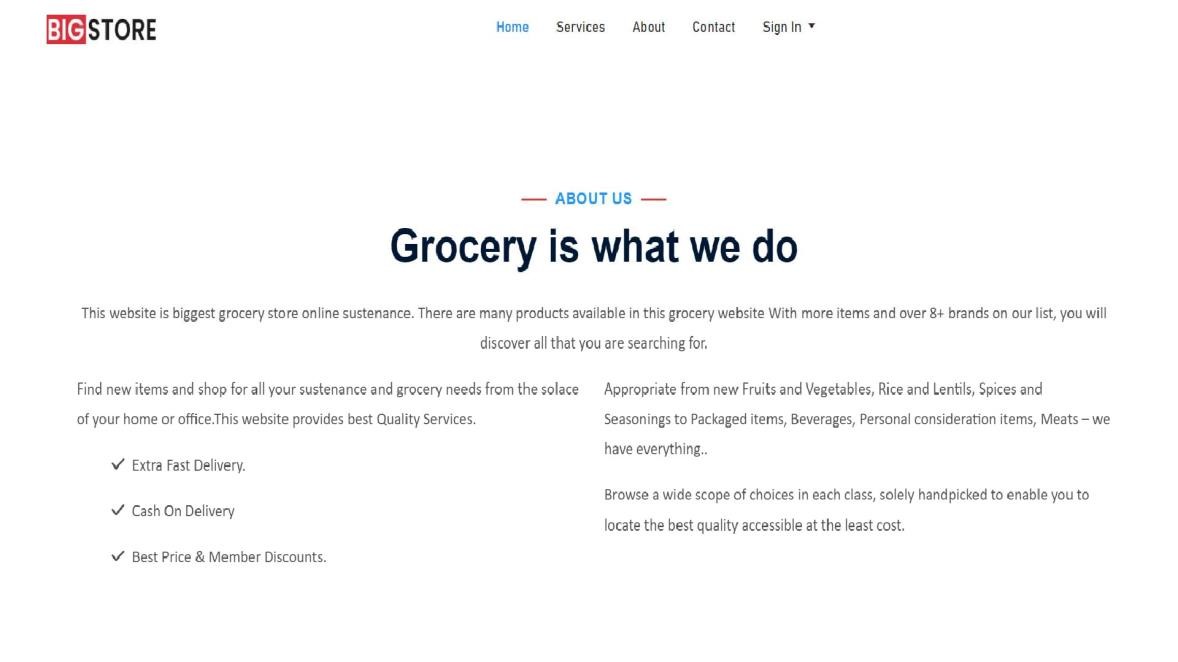


#### Input/Output Design

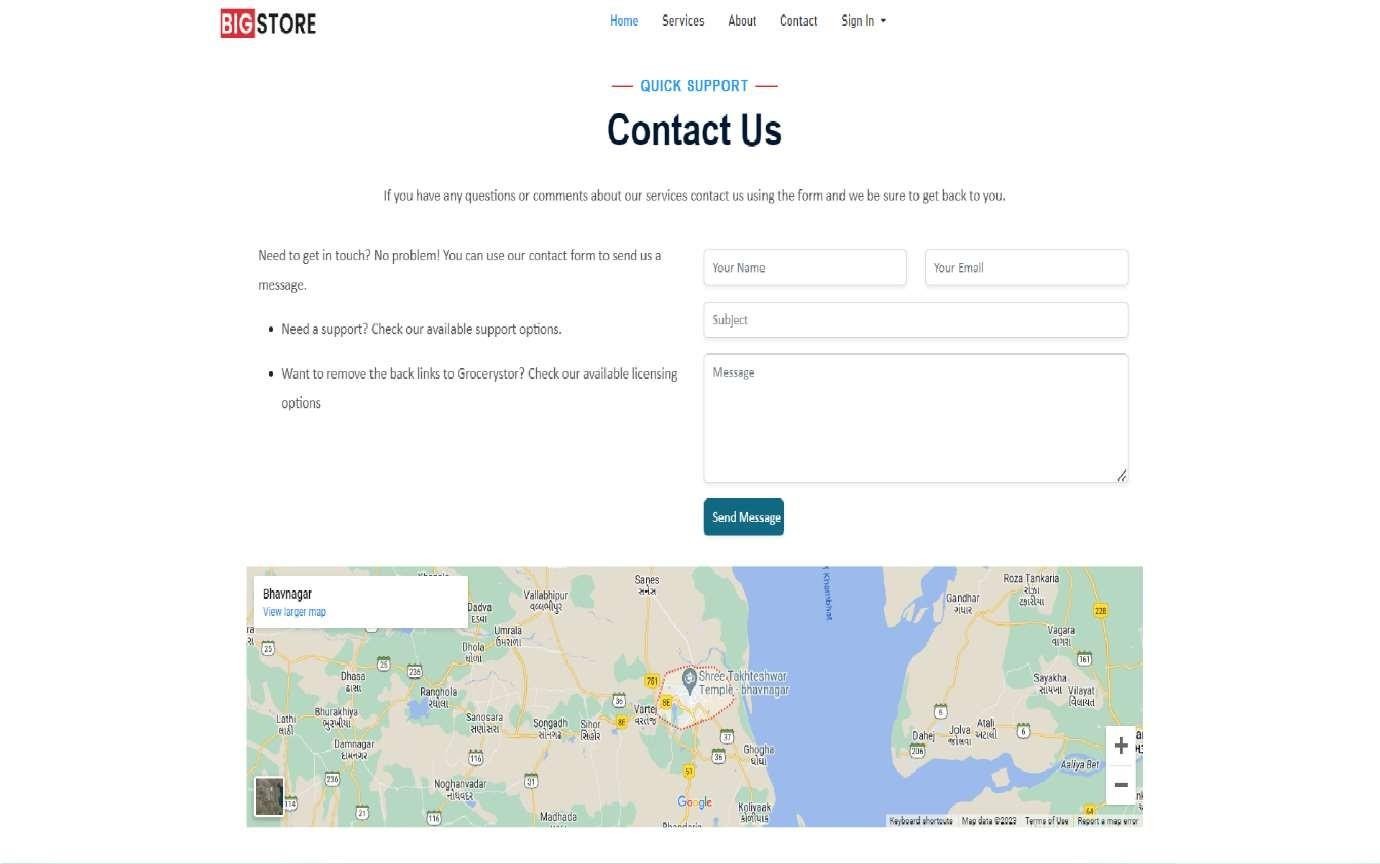
* **Home**



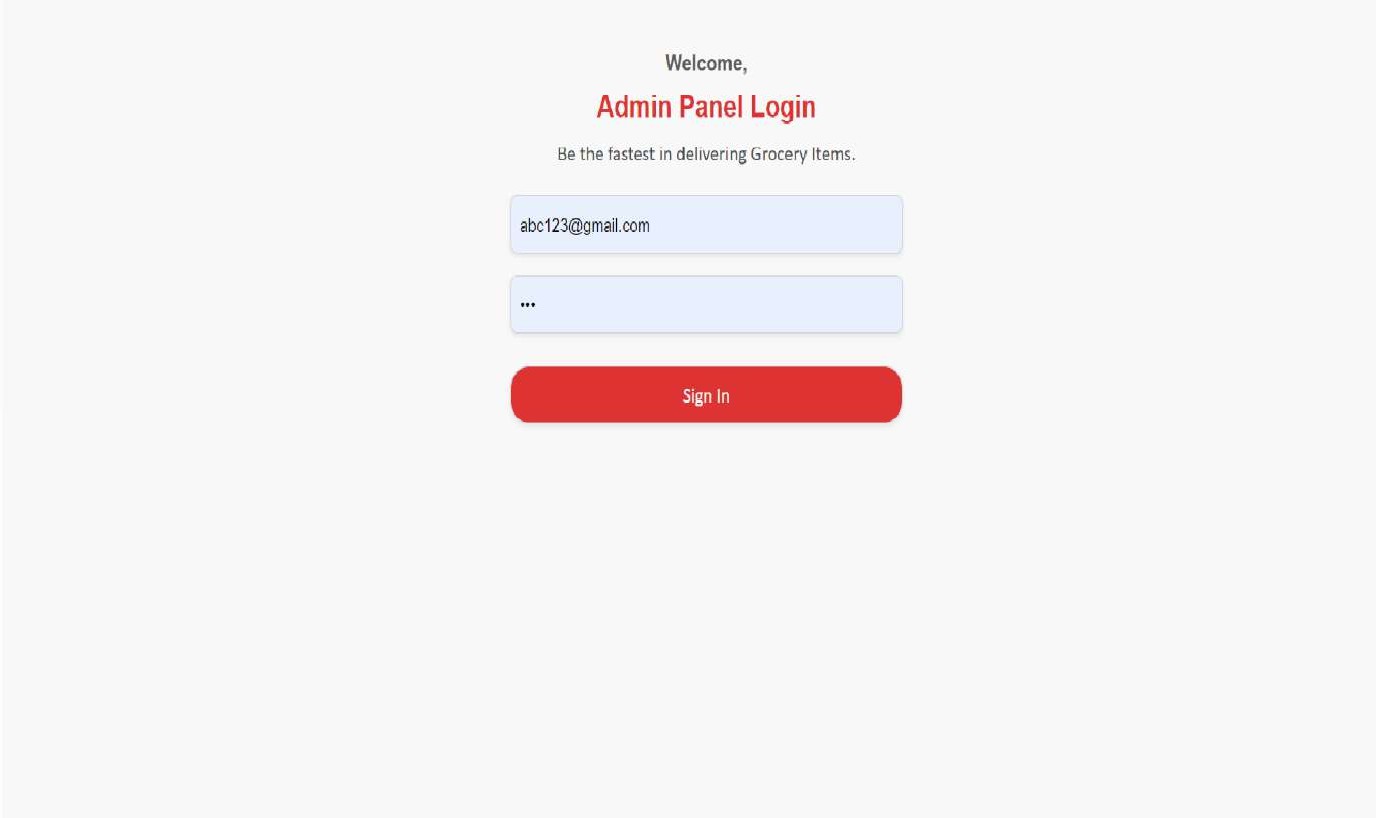
* **Services**



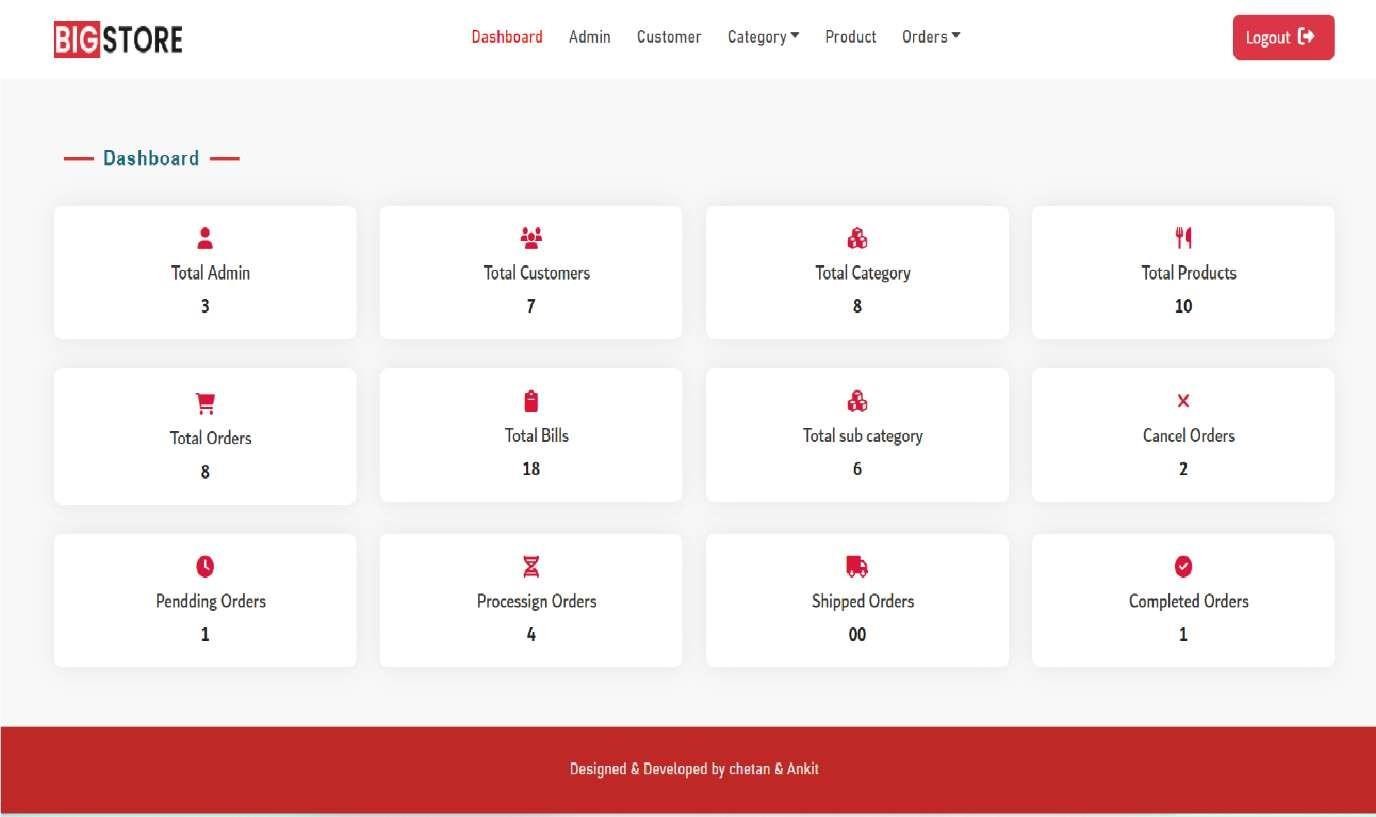
* **About**



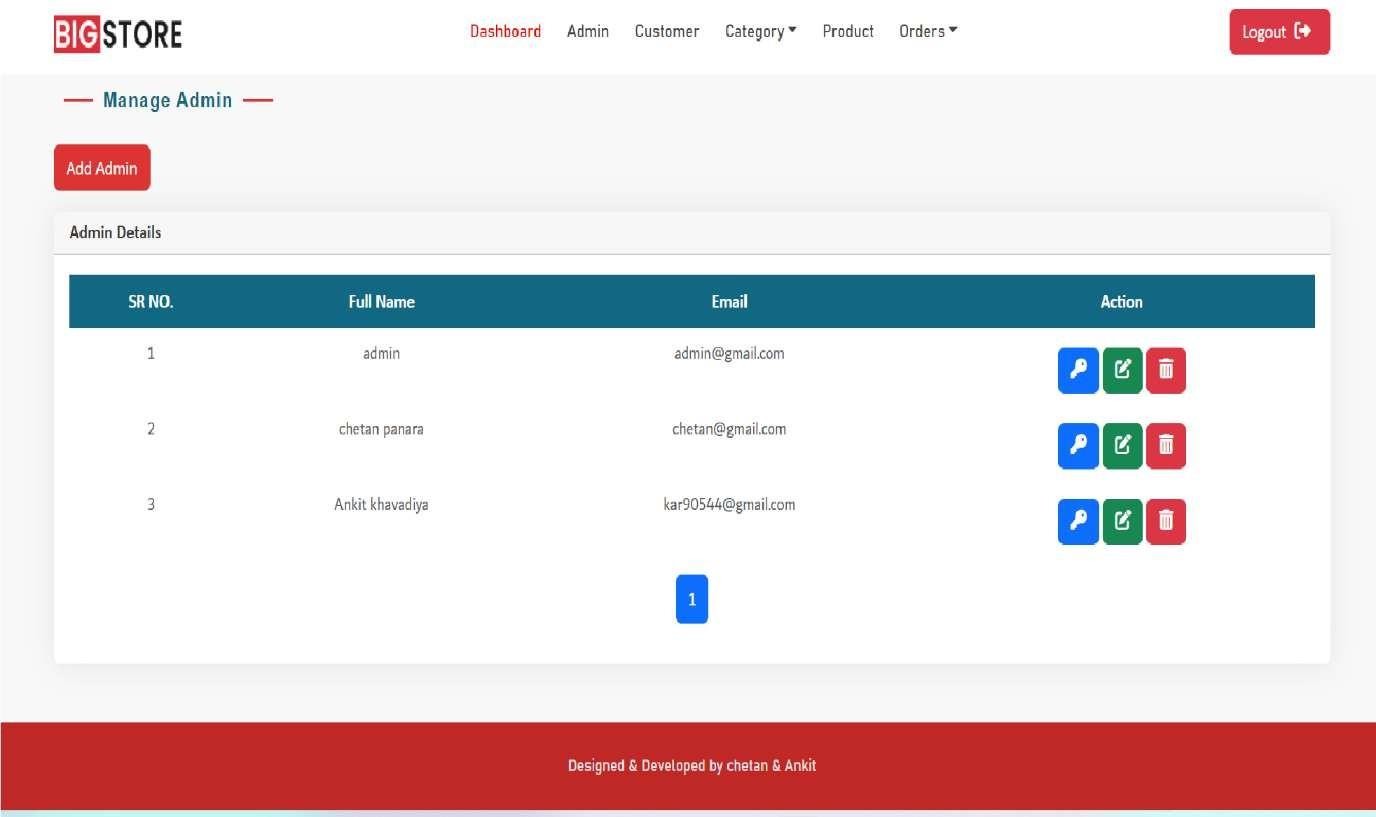
* **Contact**



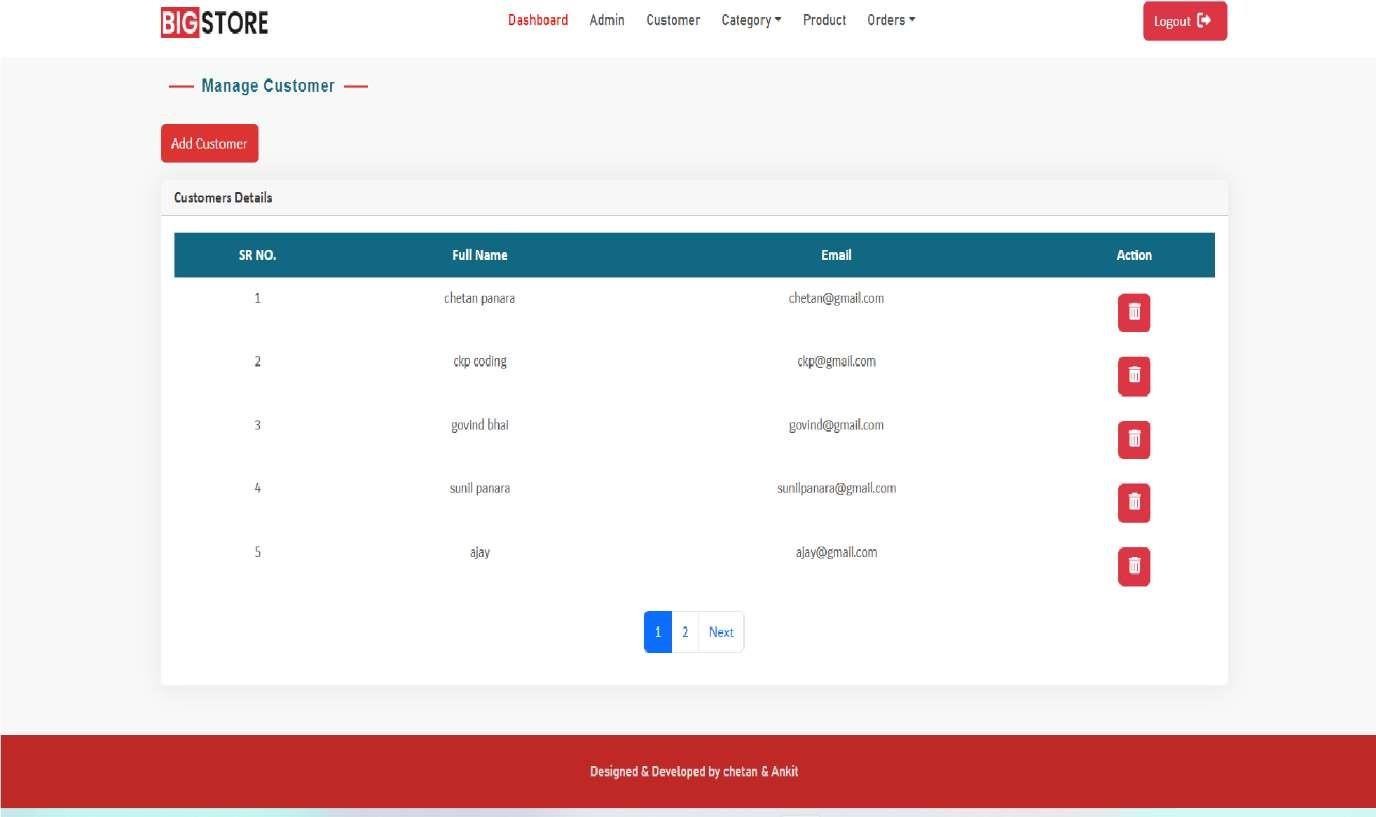
* **Sige in**
  + Admin



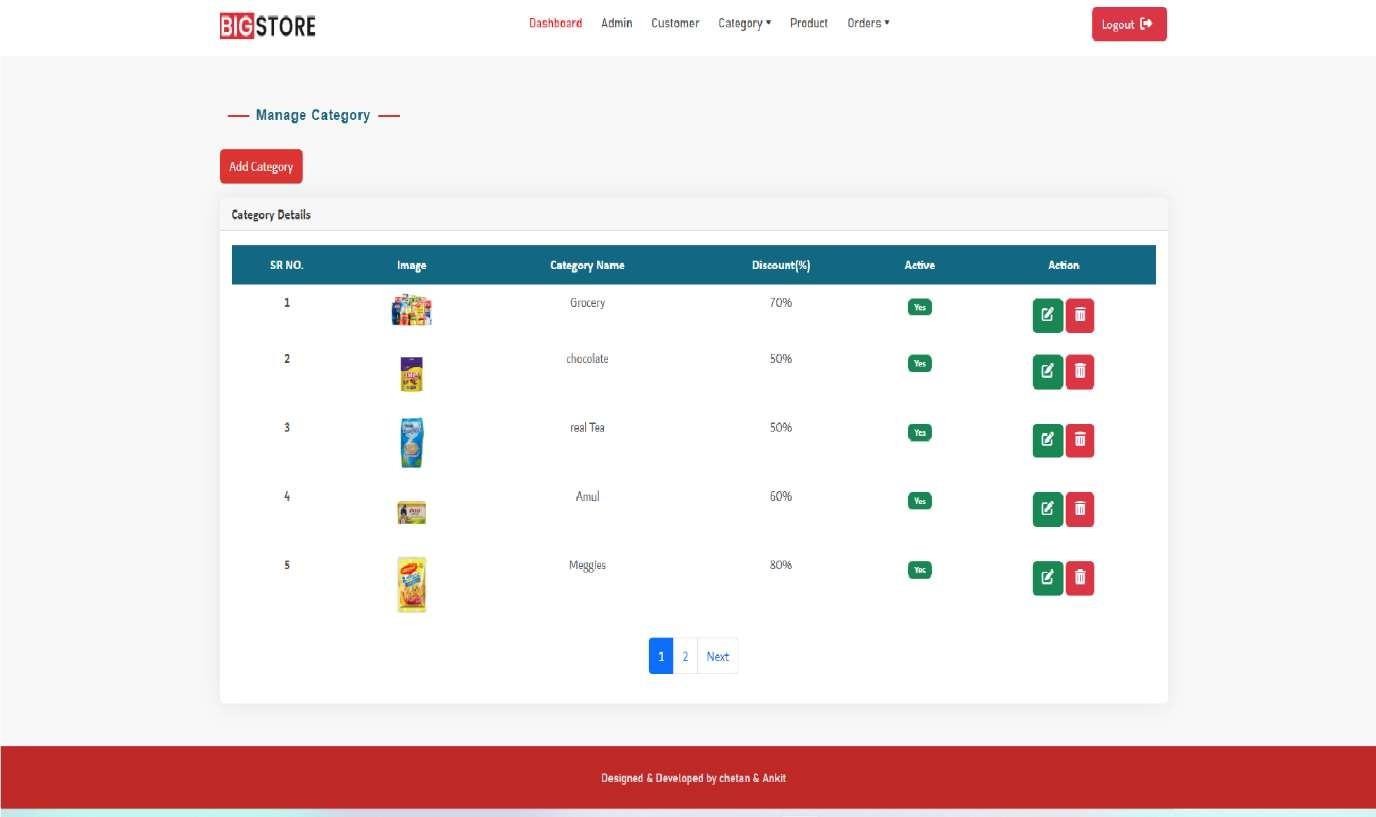
* + - Dashboard



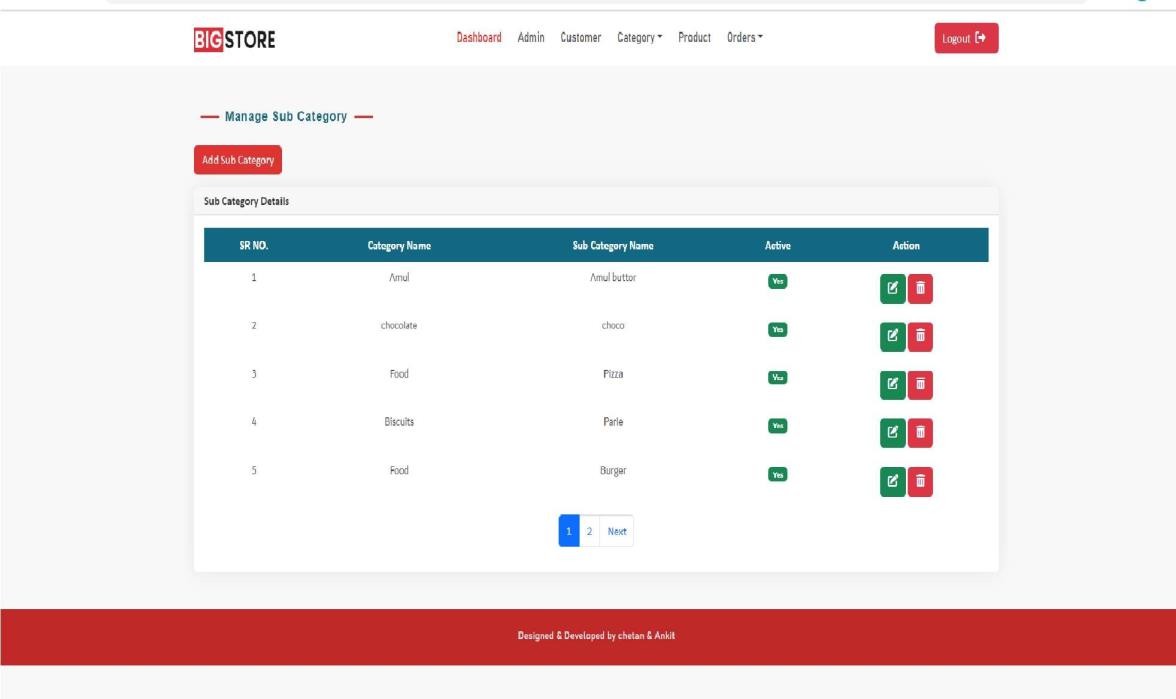
* + - Admin



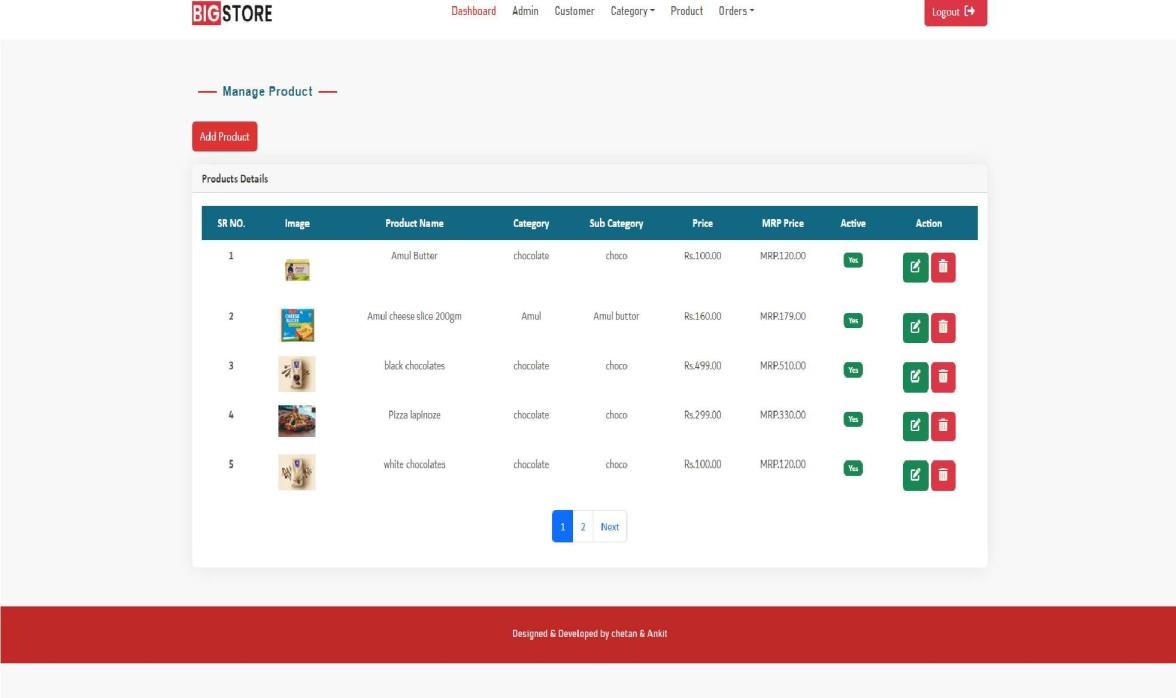
* + - Customer



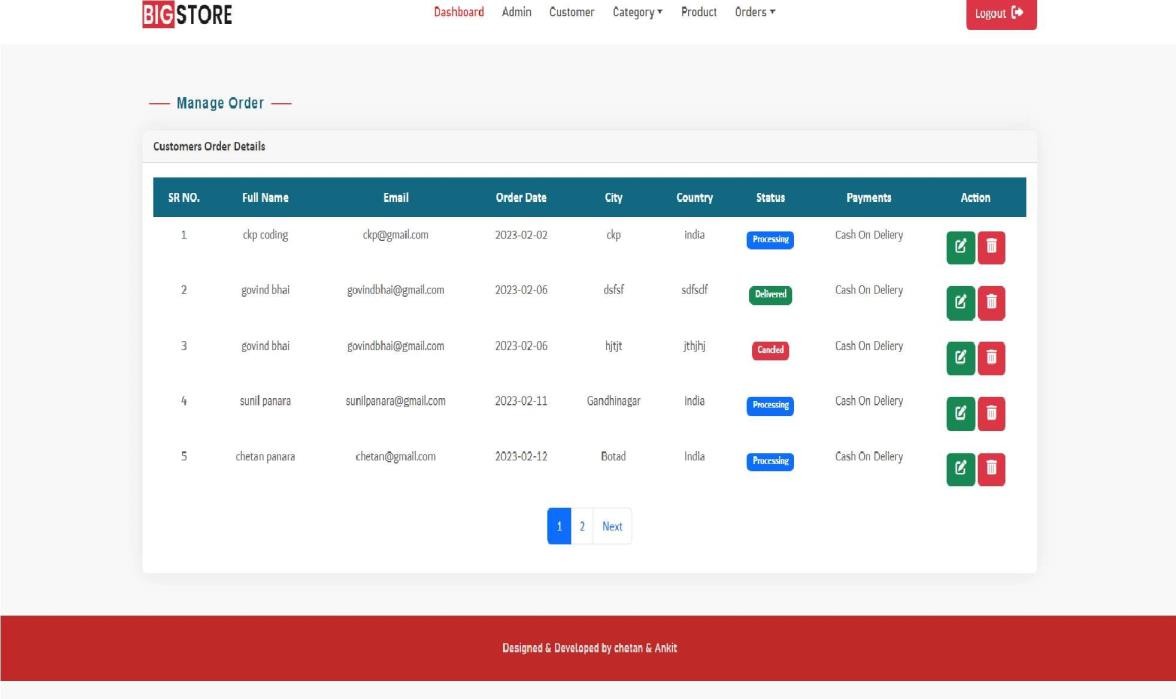
* + - Category
      * Add Category



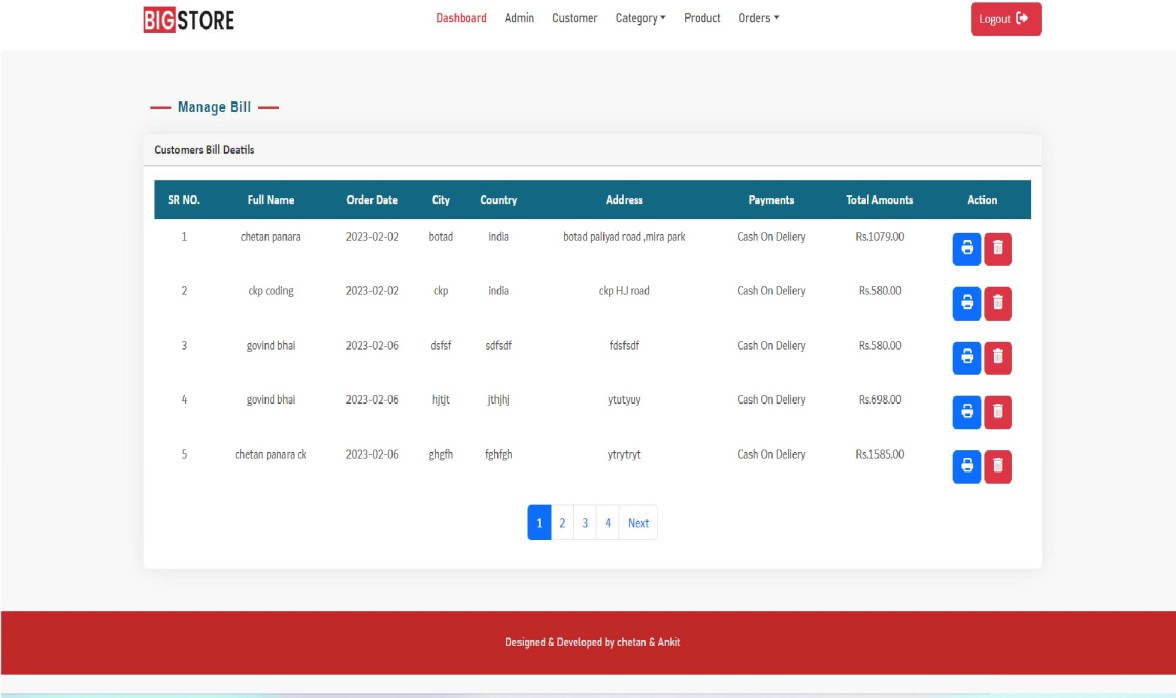
* + - * Add Sub Category



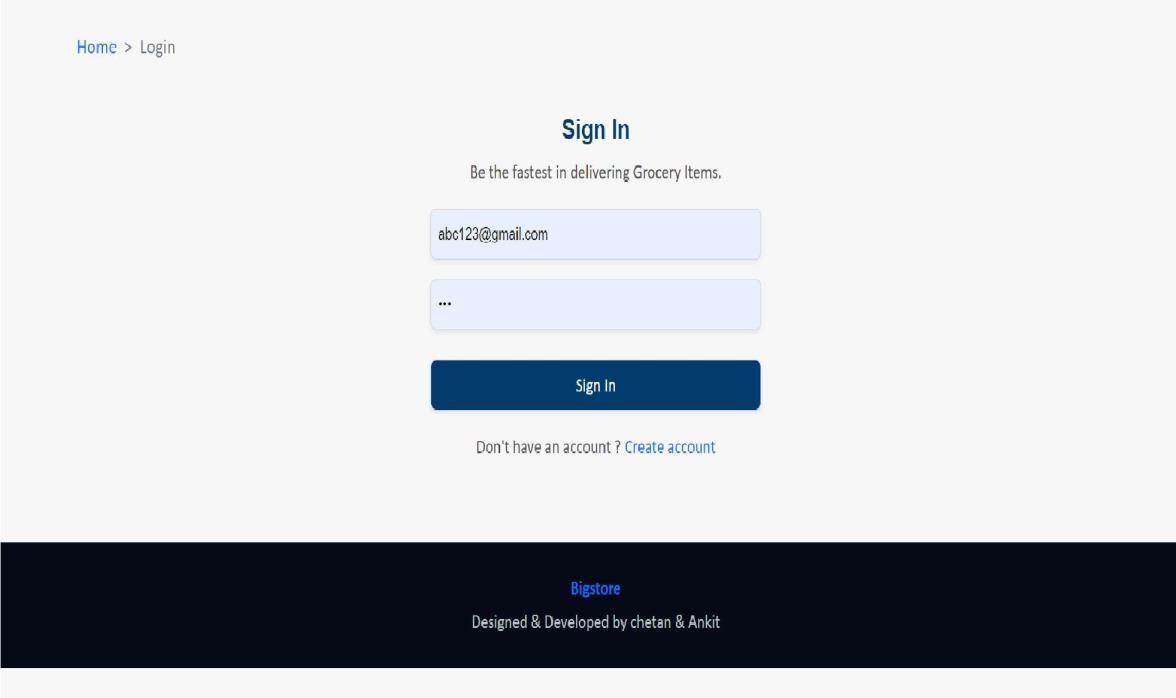
* + - Product



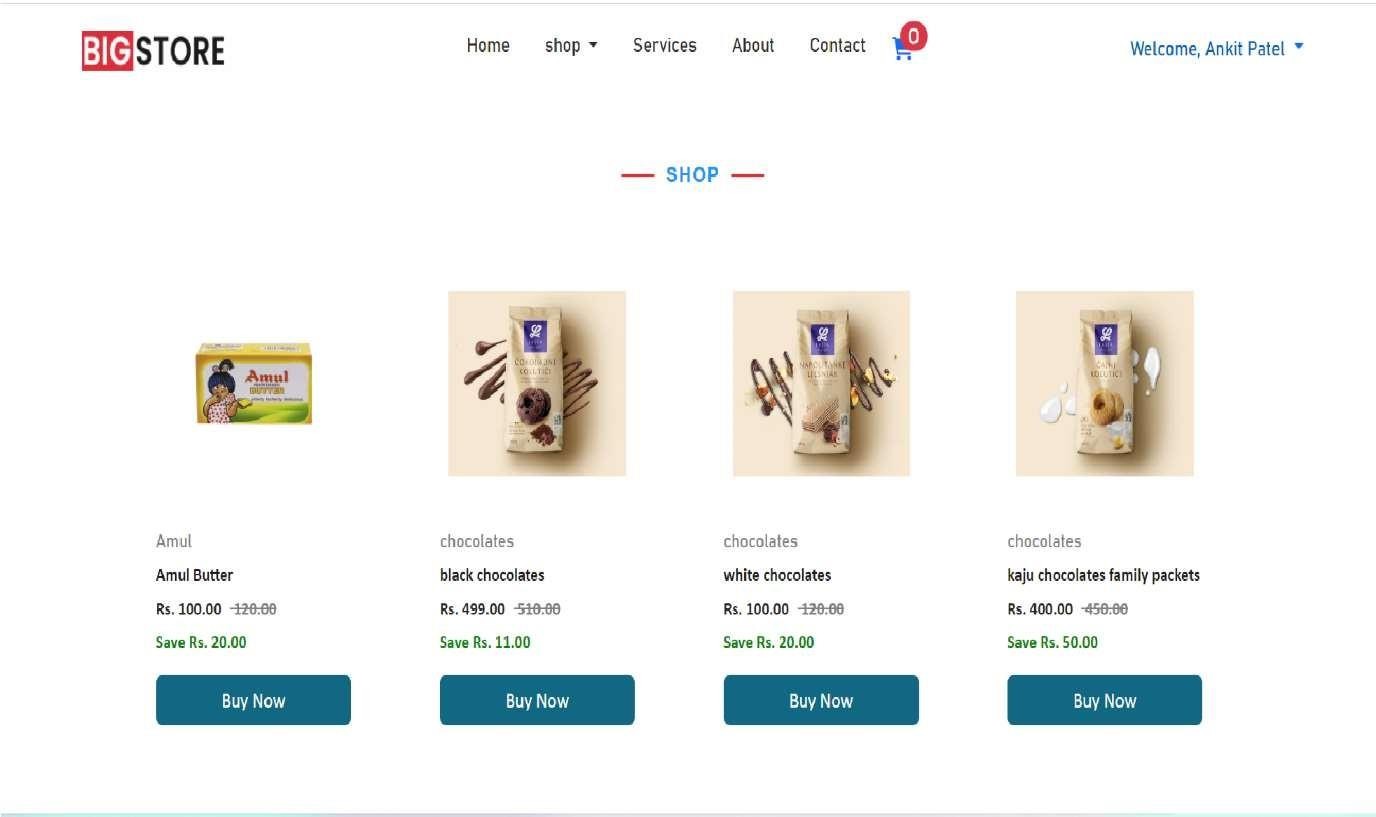
* + - Order
      * Order details



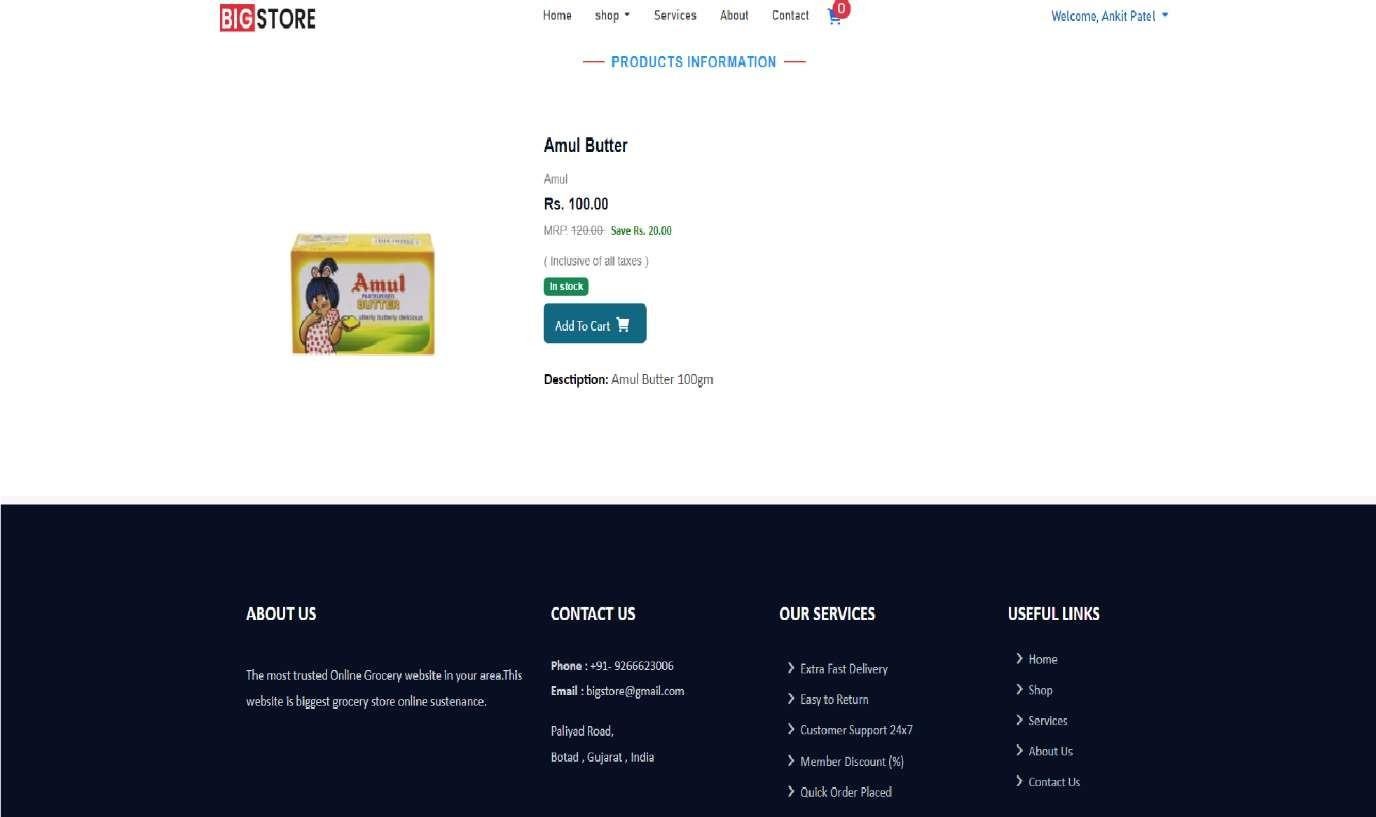
* + - * Bill



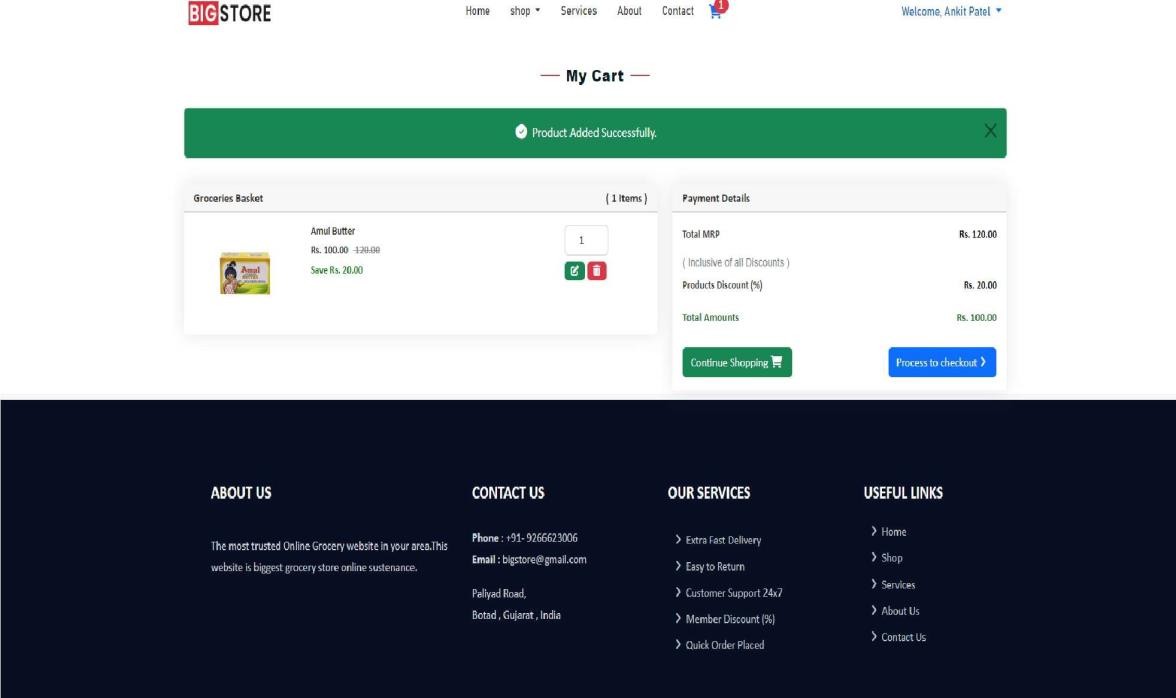
* + Customer



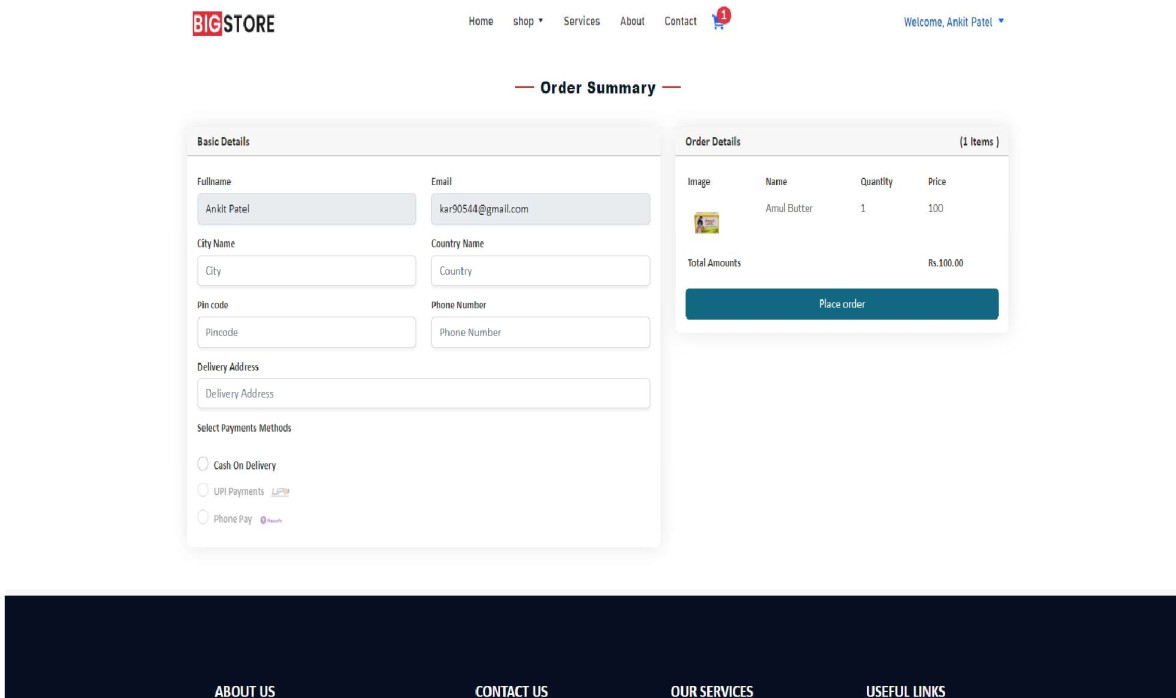
* + Create Customer
    - Shop



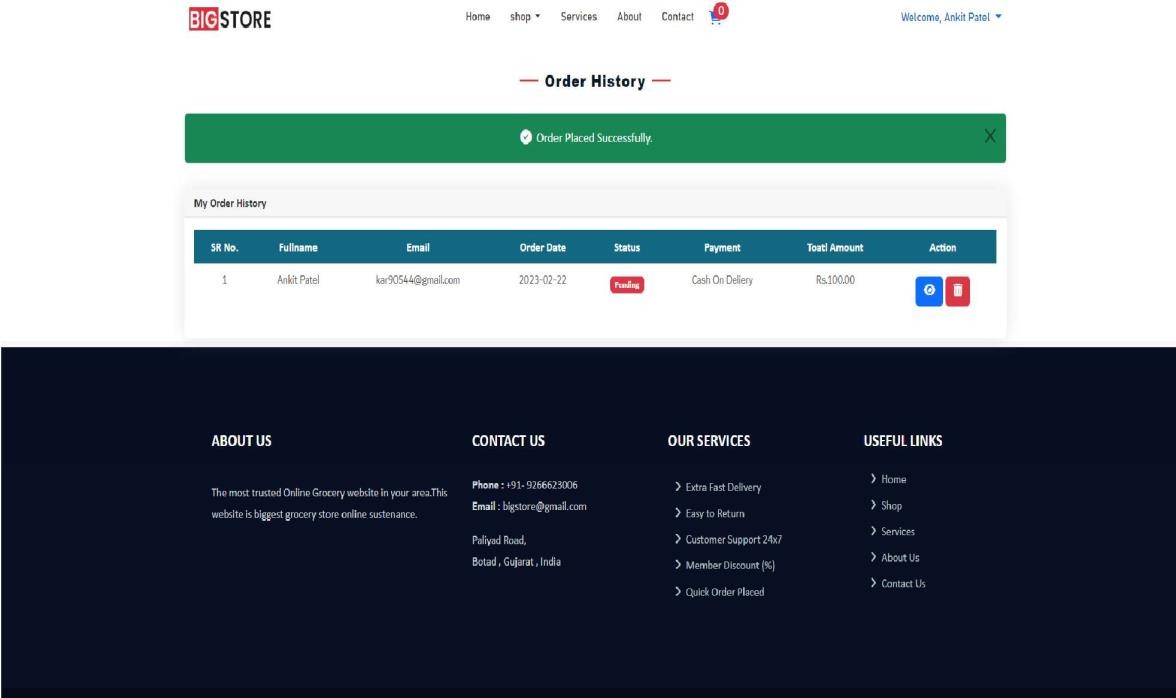
* + - * Product information



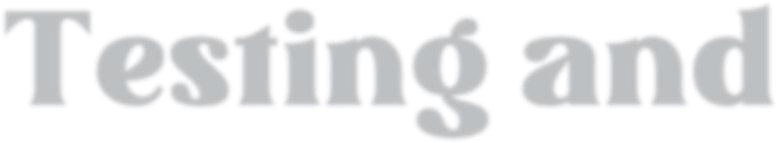
* + - * Add to cart



* + - * Order Summary



* + - * Order History



# Testing and Implementation



## Testing & Implementation

#### Introduction to Testing

* Software Testing is a method to check whether the actual software product matches expected requirements and to ensure that software product is Defect free.
* It involves execution of software/system components using manual or automatedtools to evaluate one or more properties of interest.
* The purpose of software testing is to identify errors, gaps or missing requirementsin contrast to actual requirements.

##### Errors can be present in the software due to the following reasons.

###### Programming errors:

* + Programmers can make mistakes while developing the source code.

###### Unclear requirements:

* + The user is not clear about the desired requirements or the developers are unable to understand the user requirements in a clear and concise manner.

###### Software complexity:

* + The greater the complexity of the software, the more the scope of committing an error (especially by an inexperienced developer).



###### Changing requirements:

* + The users usually keep on changing their requirements, and it becomes difficult to handle such changes in the later stage of development process. Therefore, there are chances of making mistakes while incorporating these changes in the software.

###### Time pressures:

* + Maintaining schedule of software projects is difficult. When deadlines are not met, the attempt to speed up the work causes errors.

###### Poorly documented code:

* + If the code is not well documented or well written, then maintaining and modifying it becomes difficult. This causes errors to occur.

#### Characteristics of Software Test

##### High probability of detecting errors:

* To detect maximum errors, the tester should understand the software thoroughly and try to find the possible ways in which the software can fail.
* For example, in a program to divide two numbers, the possible way in which the program can fail is when 2 and 0 are given as inputs and 2 is to be divided by 0.
* In this case, a set of tests should be developed that can demonstrate an error in the division operator.

##### No redundancy:



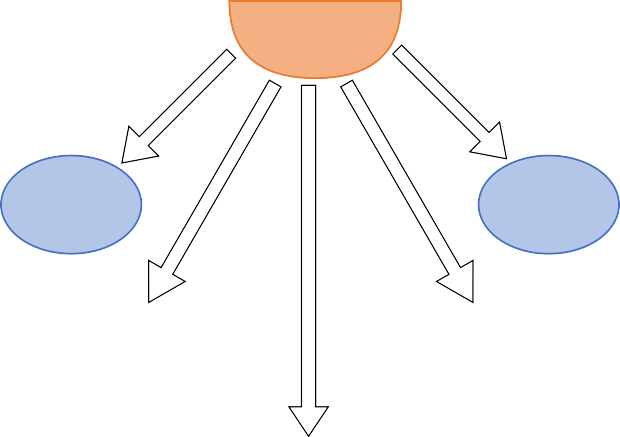
* To Resources and testing time are limited in software developmentprocess.
* Thus, it is not beneficial to develop several tests, which have the sameintended purpose.
* In Every test should have a distinct purpose.

##### Choose the most appropriate test:

* There can be different tests that have the same intent but due to certain limitations such as time and resource constraint, only fewof them are used.
* In such a case, the tests, which are likely to find a greater number of errors, should be considered.

##### Moderate:

* A test is considered good if it is neither too simple, nor too complex.
* Many tests can be combined to form one test case.
* However, this can increase the complexity and leave many errors undetected.
* Hence, all tests should be performed separately.



#### The different types of tests

##### Unit Testing

* Unit tests are very low level, close to the source of your application.
* They consist in testing individual methods and functions of the classes,components or modules used by your software.
* Unit tests are in general quite cheap to automate and can be run very quicklyby a continuous integration server.

##### Integration Testing

* Integration tests verify that different modules or services used by yourapplication work well together.



* For example, it can be testing the interaction with the database or makingsure that microservices work together as expected.
* These types of tests are more expensive to run as they require multiple partsof the application to be up and running.

##### Functional Testing

* Functional tests focus on the business requirements of an application.
* They only verify the output of an action and do not check the intermediatestates of the system when performing that action.
* There is sometimes a confusion between integration tests and functional tests as they both require multiple components to interact with each other.
* The difference is that an integration test may simply verify that you can query the database while a functional test would expect to get a specific value from the database as defined by the product requirements.

##### End-to-end Testing

* End-to-end testing replicates a user behavior with the software in a complete application environment.
* It verifies that various user flows work as expected and can be as simple as loading a web page or logging in or much more complex scenarios verifyingemail notifications, online payments, etc...
* End-to-end tests are very useful, but they're expensive to perform and can be hard to maintain when they're automated.
* It is recommended to have a few key end-to-end tests and rely more on lower-level types of testing (unit and integration tests) to be able to quicklyidentify breaking changes.



##### Acceptance Testing

* Acceptance tests are formal tests executed to verify if a system satisfies itsbusiness requirements.
* They require the entire application to be up and running and focus onreplicating user behaviors.
* But they can also go further and measure the performance of the system andreject changes if certain goals are not met.

##### Performance Testing

* Performance tests check the behaviors of the system when it is undersignificant load.
* These tests are non-functional and can have the various form to understandthe reliability, stability, and availability of the platform.
* For instance, it can be observing response times when executing a high number of requests, or seeing how the system behaves with a significant ofdata.
* Performance tests are by their nature quite costly to implement and run, but they can help you understand if new changes are going to degrade your system.

##### Smoke Testing

* Smoke tests are basic tests that check basic functionality of the application.
* They are meant to be quick to execute, and their goal is to give you the assurance that the major features of your system are working as expected.
* Smoke tests can be useful right after a new build is made to decide



whether or not you can run more expensive tests, or right after a deployment to make sure that they application is running properly in the newly deployed environment.

#### Testing Methods

##### Black-Box Testing

* The technique of testing without having any knowledge of the interiorworkings of the application is called black-box testing.
* The tester does not have access to the source code.
* Typically, while performing a black-box test, a tester will interact with the system's user interface by providing inputs and examining outputs withoutknowing how and where the inputs are worked upon.

##### White-Box Testing

* White-box testing is the detailed investigation of internal logic andstructure of the code.
* White-box testing is also called glass testing or open-box testing.
* In order to perform white-box testing on an application, a tester needs toknow the internal workings of the code.
* The tester needs to have a look inside the source code and find out whichunit/chunk of the code is behaving inappropriately.

##### Grey-Box Testing

* Grey-box testing is a technique to test the application with having a limited knowledge of the internal workings of an application.



* + In software testing, the phrase the more you know, the better carriesa lotof weight while testing an application.
  + Mastering the domain of a system always gives the tester an edgeoversomeone with limited domain knowledge.
  + Unlike black-box testing, where the tester only tests the application's user interface; in grey-box testing, the tester has access to design documents and the database.
  + Having this knowledge, a tester can prepare better test dataand testscenarios while making a test plan



#### Test Cases

* 1. **Test Case 1: Customer Registration:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case Id | Test Description | Expected Result | Actual Result | Solution | Status Pass/Fail |
| 1. | Enter empty fullname | Display error message “Enter  fullname” | Display error message “Enter  fullname” | Enter fullname | Pass |
| 2. | Enter fullname in numeric value | Display error message “Enter fullname in character” | Display error message “Enter fullname in character” | Enter fullname in character | Pass |
| 3. | Enter empty email | Display error message “Enter email” | Display error message “Enter email” | Enter email | Pass |
| 4. | Enter Invalid Email | Display error message “Email  is invalid” | Display error message “Email is invalid” | Enter valid email | Pass |



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 5 | Enter less than eight length password. | Display error message “Password length should be more than 8 characters.” | Display error message “Password length  should be more than 8 characters.” | Enter greater than or equal to eight length password | Pass |
| 6. | Enter password but confirm password not same | Display error message “Password not same” | Display error message “Password not same” | Enter password and confirm password matching | Pass |

* 1. **Test Case 2: Customer Login**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case Id | Test Description | Expected Result | Actual Result | Solution | Status Pass/Fail |
| 1 | Enter empty email | Display error message | Display error message | Enter email | Pass |
| 2 | Enter empty password | Display error message “Enter  password” | Display error message “Enter  password” | Enter password | Pass |
| 3 | Valid Email but wrong Password | Display error message “Email is invalid” | Display error message “Email is invalid” | Enter valid password | Pass |



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 4. | Wrong Email but valid password | Display error message “Password is invalid” | Display error message “Password is invalid” | Enter valid email | Pass |
| 5. | Wrong Email and wrong password | Display error message “Email or Password wrong” | Display error message “Email or Password wrong” | Enter valid email and password | Pass |

* 1. **Test Case 3: Add new admin**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case Id | Test Description | Expected Result | Actual Result | Solution | Status Pass/Fail |
| 1 | Enter empty fullname | Display error message “Enter  fullname” | Display error message “Enter  fullname” | Enter fullname | Pass |
| 2 | Enter fullname in character value | Display error message “Enter fullname in character” | Display error message “Enter fullname in  character” | Enter fullname in character | Pass |
| 3 | Enter empty email | Display error message “Enter  email” | Display error message “Enter  email” | Enter email | Pass |



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 4 | Enter Invalid Email | Display error message  “Email is invalid” | Display error message  “Email is invalid” | Enter valid email | Pass |
| 5 | Enter less than eight length password. | Display error message “Password length  should be more than 8 characters. | Display error message “Password length should be more than 8  characters. | Enter greater than or equal to eight length password | Pass |
| 6 | Enter password but confirm password not  same | Display error message “Password not same” | Display error message “Password not same” | Enter password and confirm password matching | Pass |

* 1. **Test Case 4: Add new customer**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test  Cas e Id | Test Description | Expecte d Result | Actual Result | Solution | Status  Pass/Fai l |
| 1 | Enter empty fullname | Display error message “Enter  fullname” | Display error message “Enter  fullname” | Enter fullnam e | Pass |
| 2 | Enter fullname in character value | Display error message “Enter fullname in character” | Display error message “Enter fullname in  character” | Enter fullname in character | Pass |



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 3 | Enter empty email | Display error message  “Enter email” | Display error message  “Enter email” | Enter email | Pass |
| 4 | Enter Invalid Email | Display error message  “Email is invalid” | Display error message “Email is  invalid” | Enter valid email | Pass |
| 5 | Enter less than eight | Display error message | Display error message | Enter greater than or equal | Pass |
|  | length password  . | “Password length should be more than 8 characters. | “Passwor d length should be more than 8  characters  . | to eight length password |  |
| 6 | Enter password but confirm password not  same | Display error message “Password not same” | Display error message “Passwor d not  same” | Enter password and confirm password matching | Pass |

* 1. **Test Case 5: Add new category**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Cas  e Id | Test Description | Expecte d Result | Actual Result | Solution | Status Pass/Fai  l |



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | Enter empty category name | Display error message “Enter  category name” | Display error message “Enter  category name” | Enter categor y name | Pass |
| 2 | Enter category name in character value | Display error message “Enter category name in character” | Display error message “Enter category name in character  ” | Enter category name in character | Pass |
| 3 | Select categor y image | Display error message “select  category image” | Display error message “select  category image” | Choose categor y image | Pass |
| 4. | Enter empty discount | Display error message “Enter  discount” | Display error message “Enter  discount” | Enter discoun t | Pass |
| 5. | Enter discount in numeric value | Display error message “Enter category name in  numeric ” | Display error message “Enter category name in  numeric ” | Enter discount in only numeric | Pass |
| 6. | Select one of the these active option | Display error message ”select one of these active option” | Display error message ”select one of these active  option” | Select active option Yes/No | Pass |



* 1. **Test Case 6 : Add new Product**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Cas e Id. | Test Description | Expecte d Result | Actual Result | Solution | Status Pass/Fai l |
| 1 | Enter empty Product name | Display error message “Enter  Product name” | Display error message “Enter  Product name” | Enter Proudc t name | Pass |
| 2. | Enter empty company name | Display error message “Enter company  name” | Display error message “Enter company  name” | Enter compan y name | Pass |
| 3. | Enter company name in character value | Display error message “Enter company name in character” | Display error message “Enter company name in  character ” | Enter company name in character | Pass |
| 4. | Enter empty Product price | Display error message “Enter Product  price” | Display error message “Enter Product  price” | Enter product price | Pass |
| 5. | Enter product price in  numeric value | Display error message “Enter product price in numeric ” | Display error message “Enter product price in numeric ” | Enter price in only numeric | Pass |

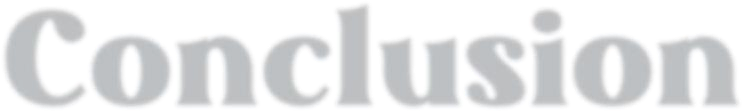


|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 6. | Enter empty Product MRP | Display error message “Enter  Product MRP” | Display error message “Enter  Product MRP” | Enter product MRP | Pass |
| 7. | Enter product MRP in numeric value | Display error message “Enter product MRP in  numeric ” | Display error message “Enter product MRP in  numeric ” | Enter MRP in only numeric | Pass |
| 8. | Select product image | Display error message “select  product image” | Display error message “select  product image” | Choose Produc t image | Pass |
| 9. | Select category name in dropdownlis t | Display error message “select  category name” | Display error message “select  category name” | Select categor y name | Pass |
| 10. | Select sub category name in | Display error message “select sub | Display error message  “select sub | Select sub category name | Pass |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | dropdownlist | category  name” | category  name” |  |  |
| 11. | Select one of the these active option | Display error message ”select one of these active option” | Display error message ”select one of these active option” | Select active option Yes/No | Pass |



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 12. | Enter empty Product Description | Display error message “Enter  Product Description” | Display error message “Enter  Product Description” | Enter Product Description | Pass |



# Conclusion



## Conclusion

#### Conclusion

* To summarize, the world is rapidly evolving and heading toward technical expertise. Technology is not a static or stagnant field, but rather one that is constantly changing as new trends arise. As patterns change and improve, it's pasttime for us to change with them.
* The use of flybox courier systems is important for getting accountability and making goods get delivered quickly and making the work easier.
* Flybox courier is an advanced system. Using this system customer can bookcourier online.
* So, this system will be very useful because today people don’t have time to go courier service office and book a courier. This system saves customer’s time.

#### Limitations and Future Enhancement

##### Limitations

* Any system will never complete, always there is change of improving. it is due to many reasons for example due to lack of technology or current requirement of user is limited.
* As the time progresses, new requirements always emerge, therefore the system which is today looking complete, in future may not be.
* The limitation of our system is defined and the lack of functionality in oursystem as below:



* + In our website Online payment facility is not available.
  + After book courier user can not get detail via SMS
  + Substantial hardware and software start-up costs.

##### Bibliography

* Google: https://google.com/
* Stack Overflow: https://stackoverflow.com/
* W3Schools: https://[www.w3schools.com/](http://www.w3schools.com/)
* Bootstrap: https://getbootstrap.com/
* YouTube: https://youtube.com/
* PHP: https://php.net/