

**Tribhuvan University**

**Faculty of Humanities and Social Sciences**

**“ONLINE CAR RENTAL SYSTEM ”**

**A PROJECT REPORT**

**Submitted to:**

**Department of Computer Application**

**Kathmandu College of Central State**

**Soalteemode, Kathmandu**

***In partial fulfillment of the requirements for the Bachelors in Computer Application***

Submitted by:

Payal Kumari Jha (6-2-528-11-2020)

November 2024

Under the Supervision of

**Mr. Bishranta Bhattarai**



**Tribhuvan University**

**Faculty of Humanities and Social Sciences**

**Kathmandu College of Central State**

# **Supervisor’s Recommendation**

I hereby recommend that this project prepared under my supervision by **Payal Kumari Jha**  entitled “**Online Car Rental System**” in partial fulfillment of the requirements for the degree of Bachelor of Computer Application is recommended for the final evaluation.

**Mr. Bishranta Bhattarai**

**Project Supervisor**

BCA Department

Kathmandu College of Central State



**Tribhuvan University**

**Faculty of Humanities and Social Sciences**

**Kathmandu College of Central State**

# **LETTER OF APPROVAL**

This is to certify that this project prepared by **Payal Kumari Jha** entitled “**ONLINE CAR RENTAL SYSTEM**” in partial fulfillment of the requirements for the degree of Bachelor in Computer Application has been evaluated. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

|  |  |
| --- | --- |
| **Mr. Bishranta Bhattarai**  Supervisor  Department of BCA  Kathmandu College of Central State  Soalteemode, Kalimati, Kathmandu | **Mrs. Ambika Deoja Kunwar**  Head of Department  Department of BCA  Kathmandu College of Central State  Soalteemode, Kalimati, Kathmandu |
| Internal Examiner | External Examiner |

# **ABSTRACT**

This project presents Online Car Rental System, an e-commerce website designed to facilitate the buying and selling of plants and related accessories. The platform is designed to provide users with a seamless shopping experience by allowing them to browse multiple products, manage their shopping carts, and complete purchases with ease. The website has a user-friendly interface designed using HTML, CSS, JavaScript, and PHP to provide a great experience. The application uses a three-tier architecture that separates the user interface, business logic, and data management layers, promoting efficiency and maintainability. MySQL is used for data storage, handling all user information, product details, and order history. The project addresses security through user authentication and aims to deliver a reliable platform for both customers and administrators. Overall, Online Car rental System is trying to create an easy online space for plant enthusiasts to explore and purchase their favorite items.

***Keywords****: E-commerce, Online Bussiness, My SQL, HTML, CSS, PHP, Web application*

# **ACKNOWLEDGEMENT**

It is a great pleasure to have the opportunity to extend our heartfelt gratitude to everyone who helped me throughout the project. I would like to express my profound gratitude to my supervisor**, Mr. Bishranta Bhattarai,** for his guidance and support in each step of this project's progress.His continuous inspiration and valuable suggestions have helped me complete this project and achieve its target.

I express my sincere gratitude to **Mr. Suman Rijal** for his invaluable guidance and unwavering supervision, which played a pivotal role in supporting me throughout the completion of our project. His mentorship has been instrumental in shaping my work, and I am truly thankful for his dedication and support.

I want to express our sincere gratitude towards the **Department of** **Computer Application, Kathmandu College of Central State**, for providing me with the wonderful opportunity, encouragement, and environment to explore my skills and learning experience through this major project.

I would also like to express our deepest appreciation to**, Mrs. Ambika Deoja** **Kunwar**, head of department, **Kathmandu College of Central State**, for her constant motivation, and support and for providing me with a suitable working environment.

Last but not least; I would like to thank all the faculty members and friends who have direct and indirect contributions to the project. Any kind of suggestions or criticism will be greatly appreciated and acknowledged.

Payal Kumari Jha (T.U. Regd. No: 6-2-528-11-2020)

**TABLE OF CONTENTS**

Supervisor’s Recommendationi

LETTER OF APPROVALii

ABSTRACTiii

ACKNOWLEDGEMENTiv

LIST OF FIGURESv

LIST OF TABLESvi

LIST OF ABBREVIATIONvii

CHAPTER 1: INTRODUCTION1

1.1 Introduction1

1.2 Problem Statement2

1.3 Objectives2

1.4 Scope and Limitations2

1.4.1 Scope2

1.4.2 Limitations3

1.5 Development Methodology3

1.6 Report Organization4

CHAPTER 2: BACKGROUND STUDY AND LITERATURE REVIEW5

2.1 Background Study5

2.2 Literature Review6

CHAPTER 3: SYSTEM ANALYSIS AND DESIGN7

3.1 System Analysis7

3.1.1 Requirement Analysis7

3.1.1.1 Functional Requirements7

3.1.1.2 Non-Functional Requirements8

3.1.2 Feasibility Analysis9

i. Technical Feasibility9

ii. Operational Feasibility9

iii. Economic Feasibility10

iv. Schedule Feasibility10

3.1.3 Data Modeling11

3.1.4 Process Modeling11

3.2 System Design13

3.2.1 Architectural Design14

3.2.2 Database Design Schema14

3.2.4 Interface Design (Interface Structure Diagrams)15

3.2.5 Physical DFD18

3.3 Algorithm details19

CHAPTER 4: IMPLEMENTING AND TESTING22

4.1 Implementation22

4.1.1 Tools Used22

4.1.2 Implementation Details of Modules24

4.2 Testing25

4.2.1 Test Cases for Unit Testing 25

4.2.2 Test Cases for System Testing29

CHAPTER 5: CONCLUSION AND FUTURE RECOMMENDATION33

5.1 Conclusion 33

5.2 Lession Learnt/Outcome 33

5.3 Future Recommendation34

REFERENCES35

# **LIST OF FIGURES**

[Figure 1.1:Waterfall Model 3](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735370)

[Figure 3.1:Use Case Diagram of Brikshya Griha 8](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735371)

[Figure 3.2:Gantt Chart of Brikshya Griha 10](#_Toc181735372)

[Figure 3.3:ER Diagram of Brikshya Griha 11](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735373)

[Figure 3.4:Context Diagram of Brikshya Griha 12](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735374)

[Figure 3.5:Level-1 DFD for Admin 12](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735375)

[Figure 3.6:Level-1 DFD for User 13](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735376)

[Figure 3.7:Architectural Design of Brikshya Griha 14](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735377)

[Figure 3.8:Database Design Schema 15](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735378)

[Figure 3.9:Interface Design for User Module 16](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735379)

[Figure 3.10:Interface Design for User Registration 17](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735380)

[Figure 3.11:Interface Design for User Login 17](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735381)

[Figure 3.12:Physical DFD for User Registration 18](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735384)

[Figure 3.13:Physical DFD for User Login 18](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735383)

[Figure 3.14:Physical DFD for Order 18](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735382)

[Figure 3.15:Physical DFD for Admin Login 18](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735385)

[Figure 3.16:Physical DFD for Category 19](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735387)

[Figure 3.17:Physical DFD for Product 19](file:///C:\Users\Dell\OneDrive\Desktop\Ptoject2Report.docx#_Toc181735386)

# 

# **LIST OF TABLES**

[Table 4.1:Test Case for User Registration Module 25](#_Toc181700872)

[Table 4.2:Test Case for User Login 26](#_Toc181700873)

[Table 4.3:Test Case for Product Management Module 27](#_Toc181700874)

[Table 4.4:Test Case for Cart Module 28](#_Toc181700875)

[Table 4.5:Test Case for Checkout Module 28](#_Toc181700876)

[Table 4.6:Test Case for Logout Module 29](#_Toc181700877)

[Table 4.7:Test Case to verify Admin Module 29](#_Toc181700878)

[Table 4.8:Test Case to verify User Module 31](#_Toc181700879)

# 

# **LIST OF ABBREVIATION**

BCA Bachelor in Computer Application

CRUD Create, Read, Update and Delete

CSS Cascading Style Sheet

DFD Data Flow Diagram

ER Entity Relationship

FAQ Frequently Asked Questions

GANTT Generalized Activity Normalization Time Table

GUI Graphical User Interface

HTML Hypertext Markup Language

IT Information Technology

MySQL My Structured Query Language

PHP Hypertext Preprocessor

TU Tribhuvan University

# **CHAPTER 1: INTRODUCTION**

## **Introduction**

E-commerce has changed the way people shop, allowing people to shop from the comfort of their homes. Brikshya Griha is an e-commerce website that caters to the growing demand for plant-related products, including live plants, seeds, pots, and gardening accessories. The platform aims to provide a user-friendly, secure, and efficient way for plant enthusiasts and hobbyists to browse, purchase, and have their desired products delivered to their doorstep.

This project is focused on developing a feature-rich website where users can explore a wide range of plant and garden products. It is web based online project where you can find different types of plants, pots fertilizers and other equipment’s for decoration purpose. Brikshya Griha is built using a combination of HTML, CSS, PHP, MySQL, and JavaScript, ensuring a robust and dynamic experience for its users. The website follows a three-tier architecture, which separates the presentation, business logic, and data management layers, contributing to enhanced maintainability and scalability.

The platform incorporates various features such as user registration, secure login, product search functionality, product reviews, product recommendation based on popularity, a shopping cart, and a checkout system. It also offers a personalized experience through a recommendation system, providing users with suggestions based on highest sales or purchases. The admin panel allows administrators to efficiently manage product listings, orders, and customer queries.

When logged in as admin, it allows admin to add, update, and remove products, and handle user accounts. They are responsible for processing orders, updating order statuses, and customer enquiries and viewing sales report.

Brikshya Griha aims to bridge the gap between plant suppliers and customers, creating a community where nature lovers can easily access the resources they need to grow and care for their plants. With a focus on simplicity, accessibility, and user satisfaction, this project brings together modern web technologies to create a seamless and enjoyable shopping experience.

## **Problem Statement**

With the rising interest in gardening, customers struggle to find a variety of plants and gardening supplies at local nurseries. The lack of a good online platform makes it challenging to browse, compare, and buy products from home. Issues such as limited choices, unclear order tracking, and poor order updates add to the inconvenience. Nursery owners also face challenges in managing inventory, customer service, and order processing. Brikshya Griha addresses these problems by providing a simple online store where customers can explore and purchase plants, pots, fertilizers, and accessories with easy order tracking. It also offers nursery owners tools for efficient business management.

## **1.3 Objectives**

The main objectives are listed below:

* Create a complete online store with a range of plants and gardening items, making shopping easy and convenient.
* To add a simple search and recommendation feature so customers can quickly find and compare items based on their needs.
* To integrate an order tracking feature, so customers know exactly where their order is from start to finish.

## **1.4 Scope and Limitations**

The scope and limitations of a project help to define its boundaries and provide clarity on what the project aims to achieve and what it may not encompass.

### **1.4.1 Scope**

The scope of Brikshya Griha is given below:

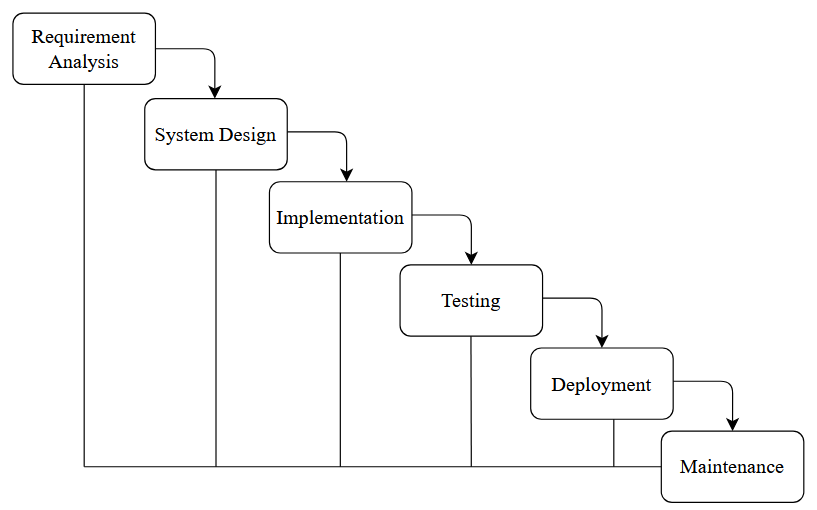
* The platform offers a range of plants and accessories to fit different customer preferences, all available for online shopping.
* Customers can track their orders from the moment they place them until delivery.
* Brikshya Griha provides customer support through FAQ sections and contact forms, helping users with any questions or issues.
* It uses cash on delivery as the payment method, giving customers a safe way to pay without needing online payment options.

### **1.4.2 Limitations**

The limitations of Brikshya Griha are given below:

* The platform currently only supports cash on delivery, which may not work for customers who prefer other payment options.
* The selection of plants and accessories might not be broad enough to meet everyone’s needs.

## **1.5 Development Methodology**

The Brikshya Griha project employs the **Waterfall model**, a structured, step-by-step approach that ensures each phase is completed fully before moving to the next. This model helps maintain clarity and organization throughout development.

**Figure 1.1:Waterfall Model**

1. **Requirement Analysis**: This phase gathers user needs to define essential features such as product browsing, cart functionality, and secure login. Requirements are documented to ensure the project aligns with user expectations.
2. **System Design**: The system was designed using a three-tier architecture, consisting of a presentation layer (HTML, CSS, JavaScript), a business logic layer (PHP), and a data management layer (MySQL). The focus was on creating a user-friendly interface and ensuring efficient data handling. The design emphasizes scalability, maintainability, and a seamless user experience, with careful planning to support future growth and updates.
3. **Implementation**: The actual coding is done here, with HTML, CSS, and JavaScript for the front end, PHP for server logic, and MySQL for data storage. Visual Studio Code is used for development and version control, ensuring a robust build.
4. **Testing**: The platform was manually tested to ensure all features, such as user registration, login, product browsing, and checkout, functioned correctly. Any issues found were resolved to improve the system’s stability and user experience.
5. **Deployment**: The platform was deployed locally using XAMPP (or another local server setup), running on localhost for development and testing. All files, including HTML, CSS, PHP, and MySQL, were placed in the local server directory. It is currently not deployed to a live server.
6. **Maintenance and Support**: Ongoing updates, fix bugs, and improve features, ensuring Brikshya Griha remains user-friendly and up-to-date.

## **1.6 Report Organization**

Talking about the report organization, in our report chapter one includes the introduction of this system, i.e. the reason why this system is being made which includes as problem statement, the major objectives and goals of our system, scopes of our system, and the limitations of our system.

Second chapter contains the background of the study and literature review of system which illustrates the work that has been carried out in the field of data mining and also describes the features of some existing applications related to the online nursery system.

The third chapter includes system analysis and system design. We included requirement analysis of both functional and non-functional requirements in system analysis. Also, we mentioned about technical feasibility, operational feasibility, economic feasibility and schedule feasibility in system analysis. We prepared entity relationship diagram and data flow diagram which were also embraced in system analysis. The architectural design, database schema design, interface design and physical DFD were put in system design part.

Fourth chapter embraces the part of system implementation and testing. And finally we have done implementation and testing of our system on different platform. Unit testing and system testing are done properly to erase bugs and defects in our system. And the fifth chapter highlights the summary of the lesson learned, outcome, and conclusion of the whole project and explains what has been done and what further improvements could be done.

# **CHAPTER 2: BACKGROUND STUDY AND LITERATURE REVIEW**

## **2.1 Background Study**

Today interest in using proper systems and software is so great that it is the most active research area in software. Moreover, users are interested in using proper systems for their daily tasks. It can be used in wide range. A plant is more than just an item of decoration. Plants help to absorb carbon dioxide and keep the oxygen flowing, they purify the air, ease tension and create a relaxed environment. [1]

Brikshya Griha is developed to be used for managing plants, plant items, and fertilizers. Brikshya Griha is a computer vision task which helps to manage stock of the plant nursery shop and facilitate the vendor for operational tasks of the shop. This is typically done by making a proper system for customer login, registration, admin login, ordering desired materials, and maintaining a proper stock for users and items in the shop. Brikshya Griha can be beneficial for home, office, or temple where plants are cultivated, grown, and for decoration purposes.

In the near future, buying plants and their materials will be operated and influenced by the computerized world, so proper software for ordering plant materials and managing stock are basic jobs. There are a variety of uses for it. On Brikshya Griha, customer registration, login details, accepting orders from customers, and stock materials will be the input viewing orders from the customer, billing according to the order, and delivering the items are the operational procedures. The system also includes some modules that are viewable only to the Administrator. These modules are mostly for managing, maintaining, and monitoring websites, orders, users, and products. One of these modules is product inventory and monitoring. This is where the Admin can view the products that are in the stock and out of the stock. Maintenance allows the Admin to add/update/delete products and categories. To add new items to the store, the Admin must input the required text fields such as product name, selling price, stock, short description etc. in the admin panel.

## **2.2 Literature Review**

Brikshya Griha is introduced to facilitate people by providing plants and other components related to plant. It is web based online project where you can find different types of plants, seeds, pots and plotters, fertilizers and other equipment’s for decoration purpose.

By the research, I have found that, the “Plant and Decor” which is an online marketplace that will help you find the home decoration products of your liking. It is giving services in Nepal and delivers in cash on delivery. It is a place where independent buyers and sellers can promote, sell, and buy their plant and decor products online in Nepal. The independent vendors from all around Nepal provides you with the home decor products such as paintings, furniture, antiques, and even handmade crafts according to your taste preference. You can find all things related to indoor and outdoor plants, pots, and accessories. But this system doesn’t have seeds and fertilizers. [2]

“e-NurseryNepal” is another website selling plants and it’s accessories. It has various categories for plants and their accessories.It has problem that it’s forgot password feature doesnot works and shows server error.It has varioud categories but all categories doesnot have products and also doesnot show message no product in this categories. [3]

**“**PashupatiNursery” of Nepal serves people for ordering wide varieties of plants, flowers, trees, carpet grass, etc. It doesn’t provide all items required for planting and provides plants only. This software is not updated. This website doesn’t consist nice cart system and it’s confusing for the user to use. [4]

“Gamala Ghar” was established in 2020, which provides a wide range of natural and artificial plants, gardening accessories and everything you need to make your space beautiful. They have a large array of products ranging from flowering plants, cactus, succulents, bonsai, ground covers, ceramic pots, designer pots and other varieties at the most reasonable prices. They deliver plants & planters to customers. In this system there is very less amount of product and we can’t find all the products in same place. [5]

As remedy of these drawback, we are inspired to develop new software which can fulfill the sectors under looked by this software’s. Our main focus is to facilitate customer with all items required to grow a plant. Many people love to grow plant as their interest. So, we aimed to feature our software by providing all the plants and its materials. We provide items delivery service and provide cash on delivery as payment system.

# **CHAPTER 3: SYSTEM ANALYSIS AND DESIGN**

## **3.1 System Analysis**

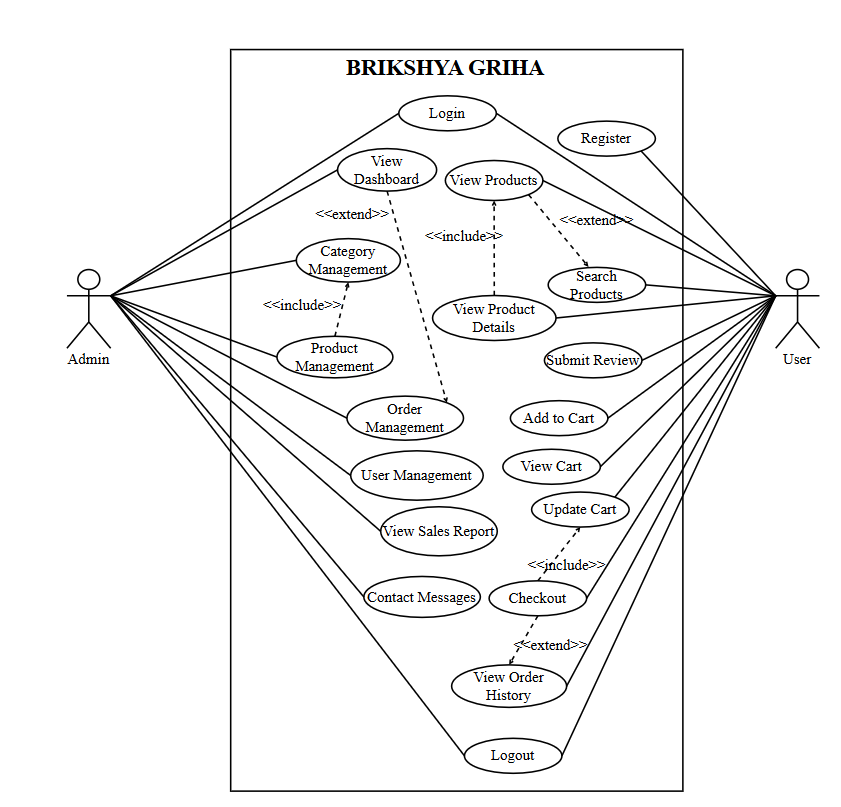
### **3.1.1 Requirement Analysis**

#### **i. Functional requirements**

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. They specify the specific functionalities and capabilities that a system must possess to fulfill the needs and objectives of its users. These requirements define what the system should do and how it should behave in response to different inputs or events. Functional requirements typically describe the system's features, behavior, and interactions with users and other system components. The functional requirements of our project are as follows:

* Users must be able to register and log in.
* The platform should allow users to browse, search, and filter products based on categories, price, and availability.
* Users must be able to add products to a shopping cart, view the cart, update quantities, and remove items before checkout.
* Orders should be placed using a "Cash on Delivery" payment method, with confirmation provided to the user.
* The admin should have the ability to add, update, and remove products while monitoring inventory levels in real-time.
* Users should be able to track the status of their orders from placement to delivery.
* A user profile management feature should allow users to view and edit their details, such as address and contact information.

**Use-case Diagram**

In this application, the admin and users are the two actors who interact with the system by involving the related information. The admin is the head of this system who can manage the users, manage the categories, manage the products, manage orders and manages user enquiries etc. However, users are the one who uses this system. He / She can select products, add to cart, view their cart, increase, decrease and remove the cart and checkout using cash on delivery through this website. Users can register their email and password and logged in using their email and password which they have registered. As an admin, admin should enter his email and password which is stored in the database at the time of development.

**Figure 3.1: Use Case Diagram of Brikshya Griha**

**Figure 3.1:Use Case Diagram of Brikshya Griha**

#### **Non-functional requirements**

Non-functional requirement describes how Brikshya Griha should perform, rather than what it should do.The non-functional requirements of Brikshya Griha are listed below: The system should have an intuitive and user-friendly interface, ensuring ease of use for all users.

* The system should respond quickly to user actions and handle multiple concurrent users without performance degradation.
* Robust security measures should be implemented to protect user data and ensure confidentiality and integrity.
* The system should be highly reliable, minimizing system failures and ensuring data integrity.
* The system should be scalable to accommodate an increasing number of users, products, and checkout process without sacrificing performance.

### **3.1.2 Feasibility Analysis**

A feasibility study assesses the operational, technical and economic aspects of this project. The feasibility study is a management-oriented activity. The objective of a feasibility study is to find out if an information system project can be done and to suggest possible alternative solutions. [6]

Various types of feasibility study is performed in order to develop the system successfully.

#### **i. Technical Feasibility**

Technical feasibility considers the technical requirements of Brikshya Griha. The technical requirements are then compared to the technical capability of the organization. This project is feasible on technical remarks as it can run on any machines supporting Windows and Internet services and works on the best software and hardware that had been used while designing the system so it would be feasible in all technical terms of feasibility. The systems project is considered technically feasible if the internal technical capability is sufficient to support the project requirements. [7]

The main tools and technologies to be used in this system to make this project more feasible are:

Hardware Requirements: A laptop with 8 GB RAM, and 12th Gen Intel i7 processor and 512 GB SSD

Technologies: PHP, HTML, CSS, JavaScript

Software Requirements: Visual Studio Code, XAMPP Server, MS Word

#### **ii. Operational Feasibility**

**Operational feasibility** is a measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development. The system is easy to operate with the basic knowledge of computer and internet and well-trained manpower is not necessary. User can also easily access the system as it is user friendly in many aspects with good Graphical User Interface (GUI). This system includes all the requirements used for Brikshya Griha is completely operational and can be successfully implemented and admin feel easy to use this app as it is user-friendly. This system include all the requirements used for Brikshya Griha is completely operational and can be successfully implemented and admin feel easy to use this App as it is user-friendly. [8]

#### **iii. Economic Feasibility**

**Conducting an economic feasibility** is crucial to determine the financial viability of the project. The proposed system is feasible and is economical regarding its pre-assumed cost for making a system. We classified the costs of Brikshya Griha according to the phase in which they occur. As we know that the system development costs are usually one-time costs that will not reoccur after the project that been completed.

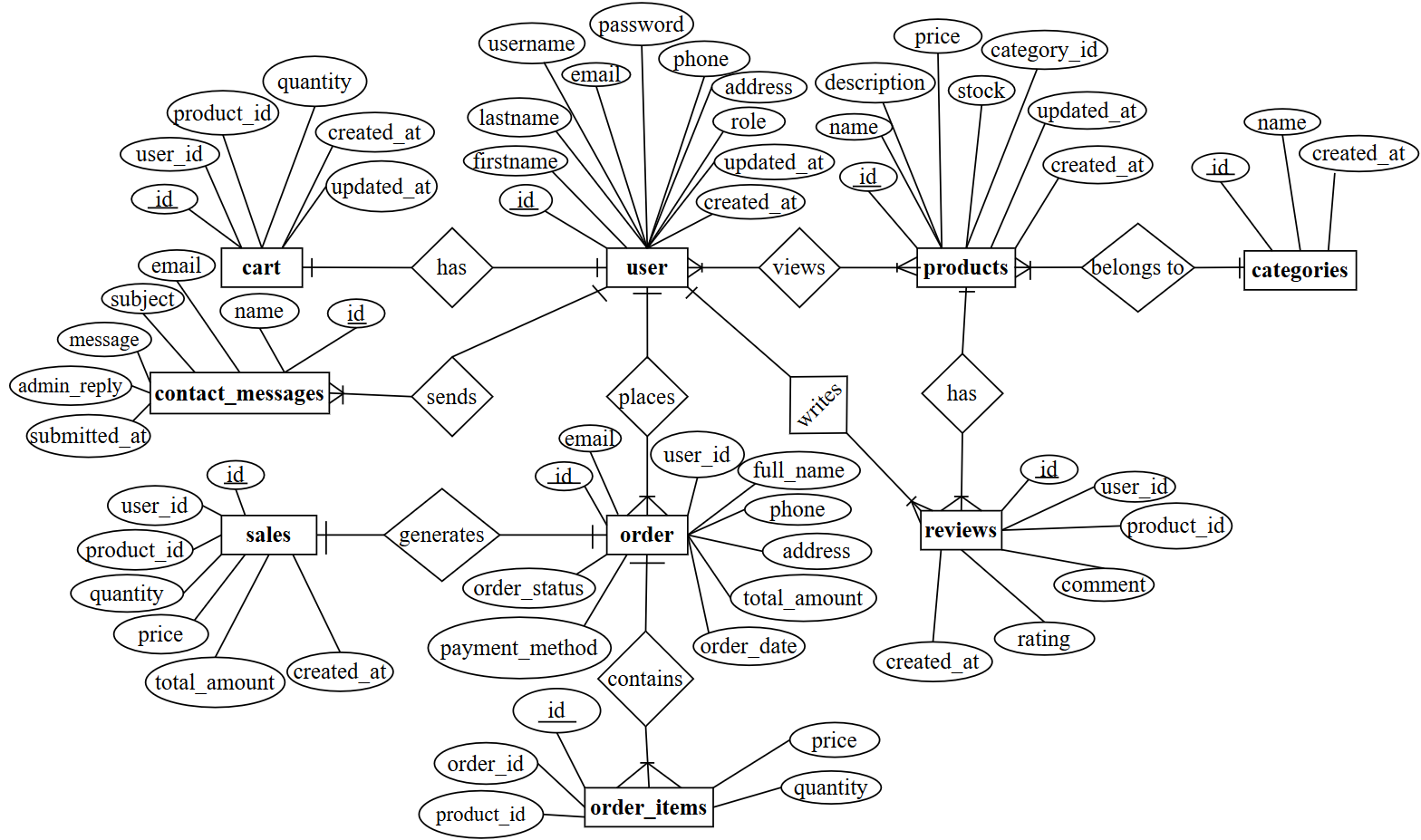
#### **iv. Schedule Feasibility**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Process | No. of Weeks | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Requirement Gathering |  |  |  |  |  |  |  |  |  |  |
| Planning |  |  |  |  |  |  |  |  |  |  |
| Designing |  |  |  |  |  |  |  |  |  |  |
| Coding |  |  |  |  |  |  |  |  |  |  |
| Testing and debugging |  |  |  |  |  |  |  |  |  |  |
| Implementation |  |  |  |  |  |  |  |  |  |  |

The system which we had developed is completed within the scheduled time and did not exceed the scheduled time. For this schedule, project planning is carried out by the Gantt chart. A Gantt chart is one of the most popular and useful ways of showing activities (tasks or events) displayed against time. It is used to show tasks or the activities performed in contradiction to time as it is one of the graphical tools.

**Figure 3.2:Gantt Chart of Brikshya Griha**

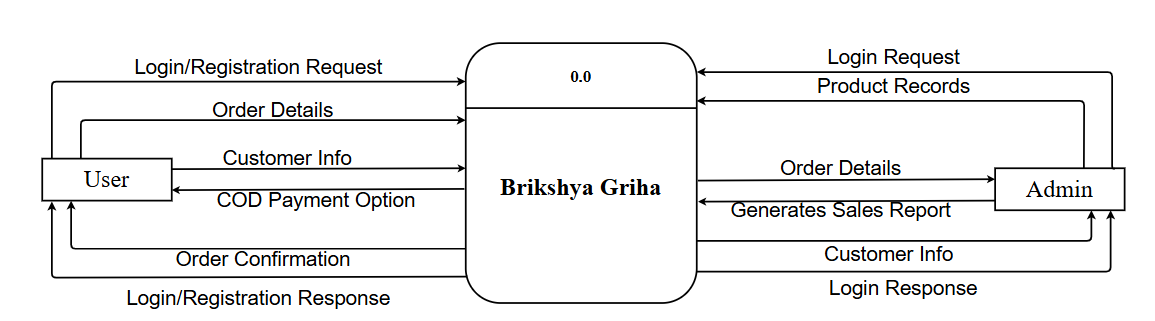
### **3.1.3 Data Modeling**

Data modeling is a crucial aspect of the Brikshya Griha project as it defines the structure and relationships of the database. Through the creation of an entity-relationship (ER) diagram, the system's entities, such as users, cart, products, categories, orders, order\_items, review, sales and contact\_messages are visually represented along with their attributes. The ER diagram illustrates how these entities are interconnected through relationships, such as admin adding categories, categories containing multiple products, and products having various reviews. This diagram provides a clear overview of the data structure and serves as a blueprint for database design, ensuring that the system accurately stores and organizes the necessary information.

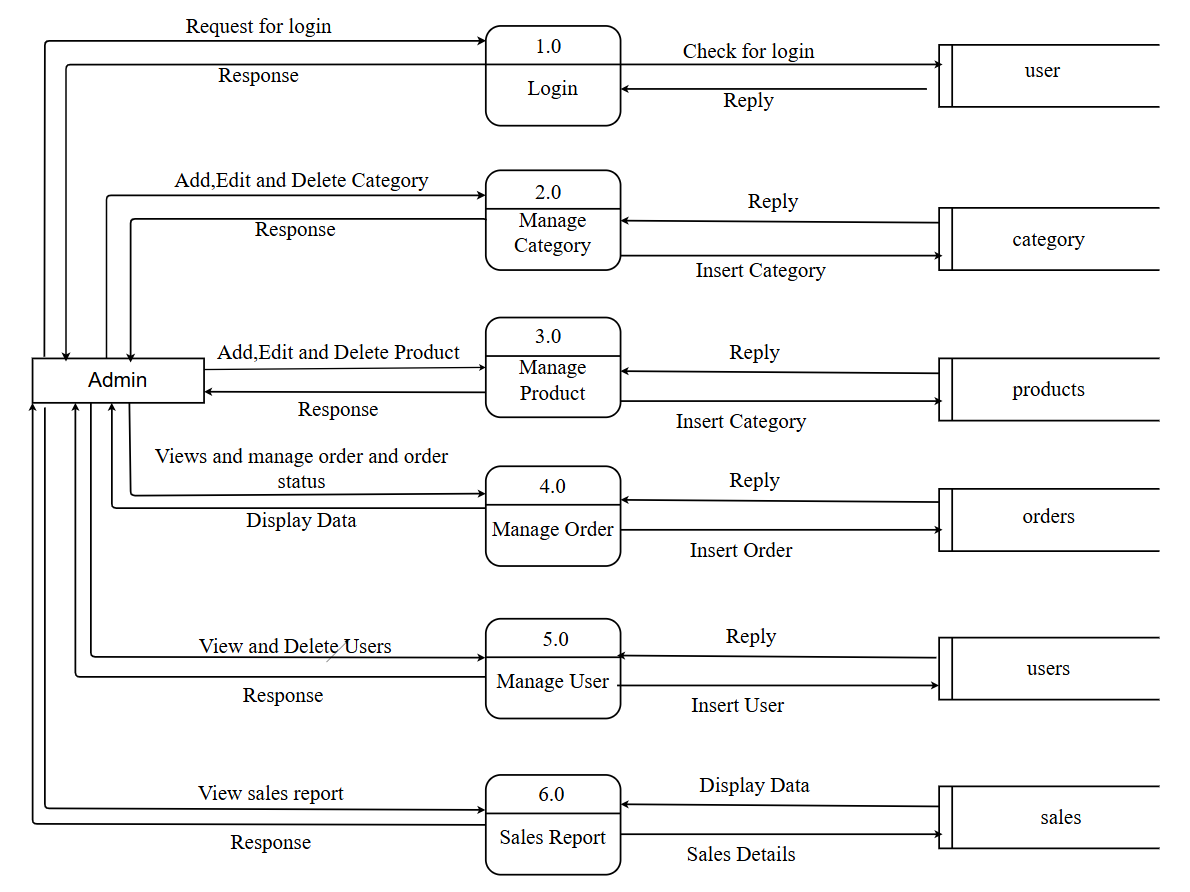
**Figure 3.3:ER Diagram of Brikshya Griha**

### **3.1.4 Process Modeling**

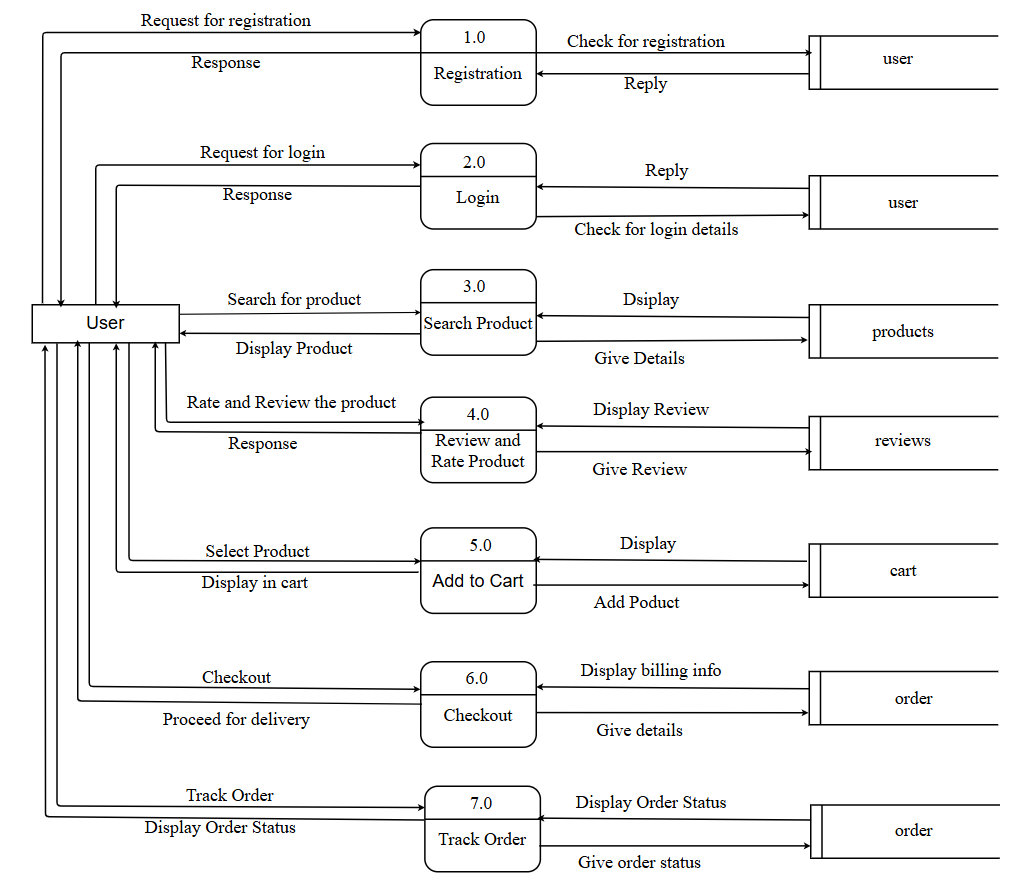
Process modeling is a technique designed to understand and describe the process. It connects and improves the communication between the current and the future state of a process. [9]

The 0-level Data Flow Diagram (DFD) for the Brikshya Griha system offers a high-level view of key processes, such as user registration and product management, while illustrating interactions between users and administrators to clarify the overall flow of information.

**Figure 3.4:Context Diagram of Brikshya Griha**

****First level DFD of Brikshya Griha shows how the system is divided into subsystems, each of which deals with one or more of the data flows to or from an external agent and which together provide all the functionalities of the Brikshya Griha as a whole. It also identifies internal data stores of admin and users. It shows the flow of data between various part of tests. DFD level 1 provides the more detailed breakout of pieces of the zero level DFD. The Level-1 DFD for both admin and users are shown below:

**Figure 3.5:Level-1 DFD for Admin**



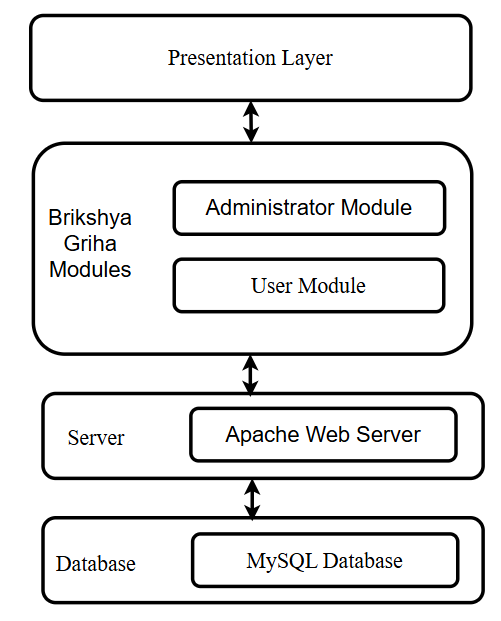
**Figure 3.6:Level-1 DFD for User**

## **3.2 System Design**

The System Design section outlines the detailed design of the Brikshya Griha project. This phase focuses on translating requirements into a well-structured and efficient system. The design ensures smooth functionality, user-friendliness, security, and scalability. This section highlights the key design aspects, such as system architecture, database schema & user interface The introduction provides an overview of the design goals, target audience, design principles/methodologies used, and any unique considerations for the project's success.

### **3.2.1 Architectural Design**

The architectural design for the Brikshya Griha follows a three-tier architecture consisting of the presentation layer, business logic layer, and data storage layer.

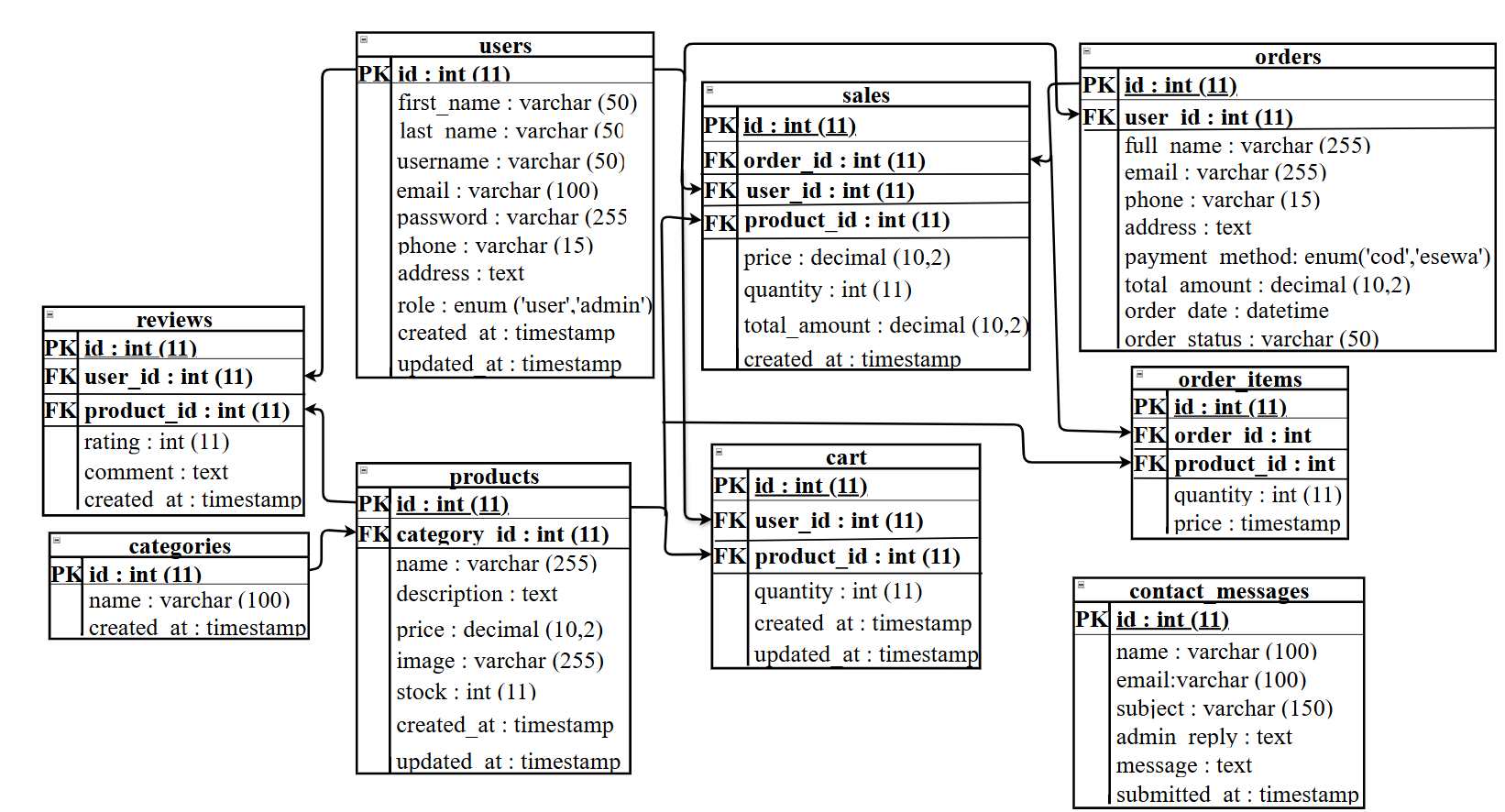


**Figure 3.7:Architectural Design of Brikshya Griha**

### **3.2.2 Database Design Schema**

In this topic, we will explore the structure and organization of the database that underlies our project. A well-designed database schema is essential for storing and managing data efficiently, ensuring data integrity, and supporting the functionalities of our system. In this section, we will discuss the main entities or tables in the database, their relationships, and the overall design principles employed. By establishing a solid database schema, we lay the foundation for a robust and reliable system that can effectively handle data storage and retrieval operations.Database name ‘brikshya\_griha’ was created to perform all the database operations of our project ‘Brikshya Griha’. There are various number of tables created in the database.

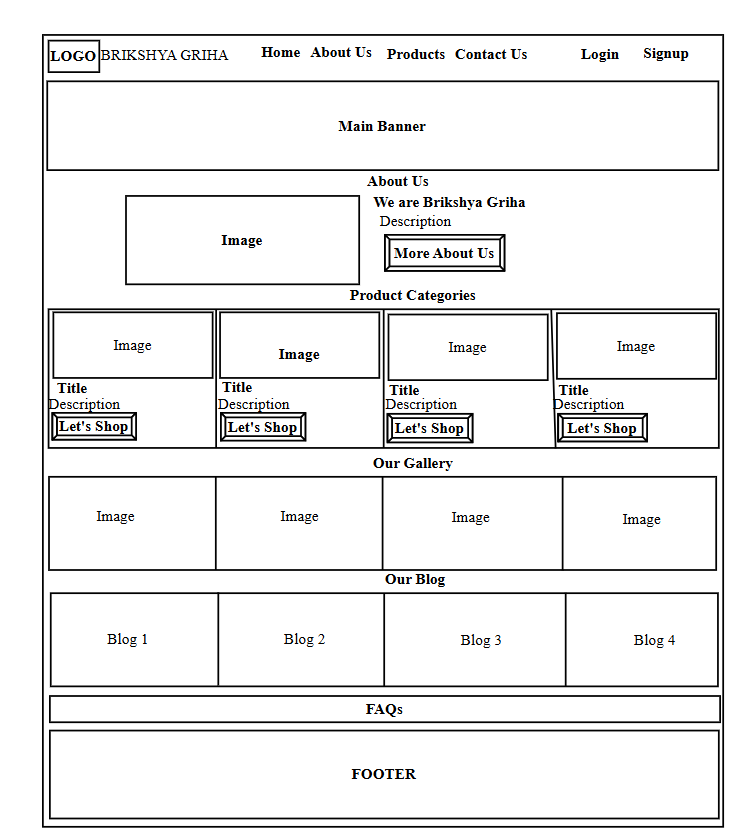
Database schema design is used to show basic structure of the system. In Brikshya Griha, there are nine tables in the databases each of them has their own fields where their id is

primary key and if that id is used in another table, it becomes foreign key and foreign key are connected to another table with a line. There is data type of each entity and the foreign key in schema is represented by the arrow as shown in the diagram.

**Figure 3.8:Database Design Schema**

### **3.2.3 Interface Design (Interface Structure Diagrams)**

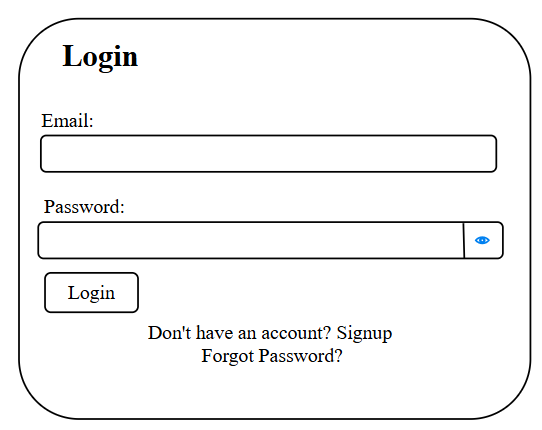
In the Interface Design part, we have focused on creating a visually appealing and user-friendly interface for our project, the Brikshya Griha. The interface follows modern design principles and emphasizes simplicity and clarity to enhance the user experience. We have carefully organized the layout of each screen, ensuring easy navigation and intuitive placement of interactive elements such as buttons, forms, and menus. The user interface of the system has been designed using HTML, CSS, JavaScript and PHP.



**Figure 3.9:Interface Design for User Module**

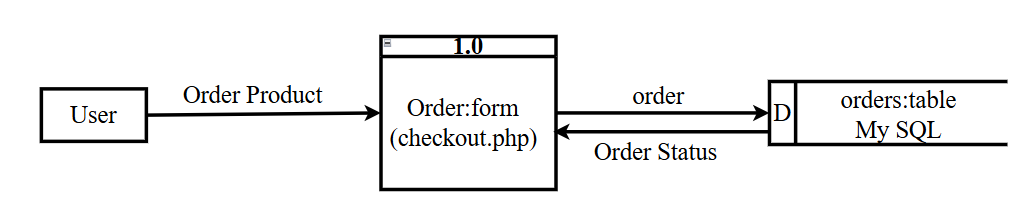
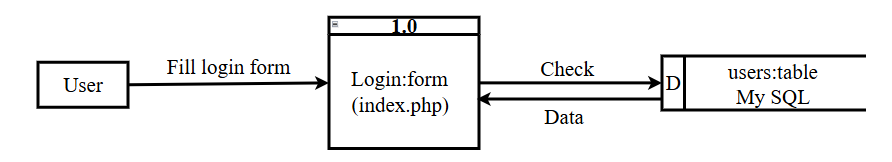
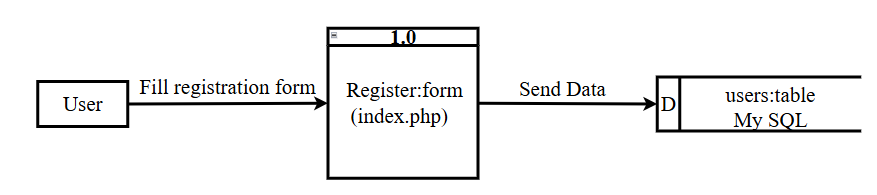


**Figure 3.10:Interface Design for User Registration**

****

**Figure 3.11:Interface Design for User Login**

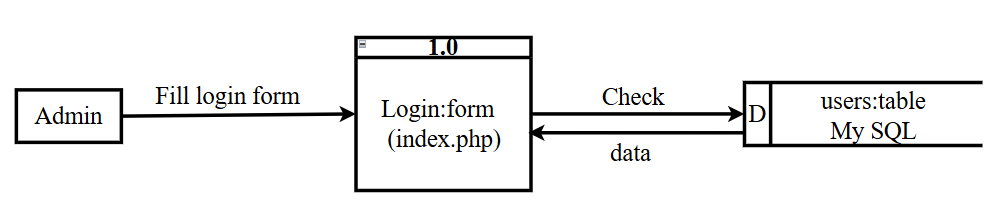
### **3.2.4 Physical DFD**

****Physical DFD goes one step deeper than level one DFD where user account is created when user enter all details correctly and user data is stored in the database. User could have logged in to the system after successfully creation of an account and users can be able to take test. Likewise, admin could have logged in to the system by entering admin detail correctly and can manage users, and the orders by accessing the user’s database.

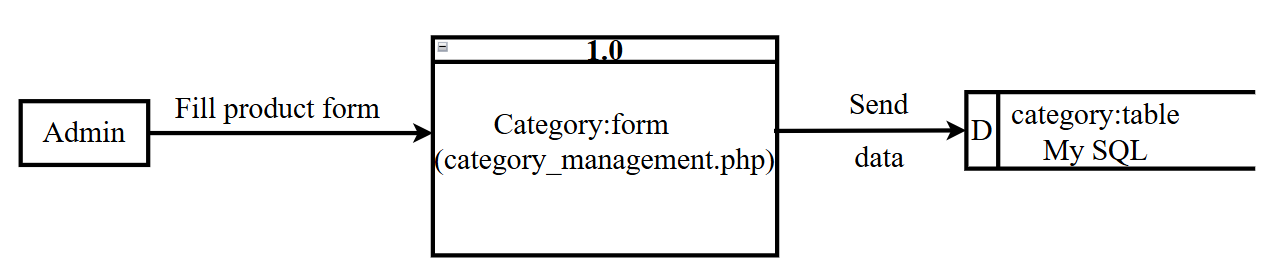
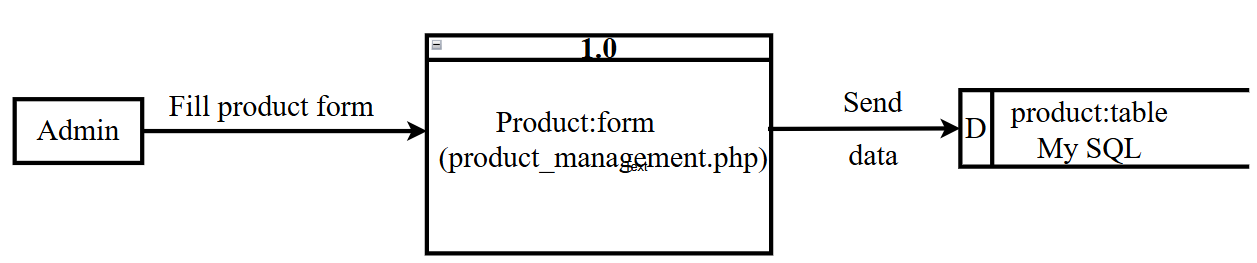
**Figure 3.12:Physical DFD for Order**

**Figure 3.13:Physical DFD for User Registration**

**Figure 3.14:Physical DFD for User Login**

Above figures are the physical DFD of user which includes user register, login, and place order. Here, the physical DFD shows how all the components of the developed system works for user.

**Figure 3.15:Physical DFD for Admin Login**

****Above figures are the physical DFD of admin which includes admin login, manage category and manage products. Here, the physical DFD shows how all the components of the developed system works for admin.

**Figure 3.16:Physical DFD for Product**

**Figure 3.17:Physical DFD for Category**

## **3.3 Algorithm Details**

In this project, we employ two primary algorithms: **Bubble Sort** for sorting products based on price and name and **Popularity Based Product Recommendation algorithm using Bubble Sort**. These algorithms enhance user experience by improving the accessibility of products and tailoring suggestions based on user preferences.

* **Bubble Sort Algorithm for Product Sorting**

Bubble Sort is a straightforward sorting algorithm often used in small-scale applications where simplicity is preferred over speed. In this project, the Bubble Sort algorithm is utilized to arrange products on the user interface based on various criteria such as price or name, allowing users to sort items as they browse. It’s purpose is to enable users to view products in a sorted order by price or name. enhancing the usability and navigability of the interface.

**Steps of the algorithm:**

**i. Fetch Products Data**

The first step involves retrieving product data from the database based on the selected category and search term. This query fetches various product details such as name, price, description, and other relevant attributes, and stores them in an array called $products. This data is then used for further processing and sorting based on user preferences.

**ii. Define Bubble Sort Function**

In this step, a function called bubbleSort() is defined, which takes two parameters: the $products array and a sorting criterion $sortBy (either 'price' or 'name'). This function will be responsible for sorting the products based on the specified criterion. The Bubble Sort algorithm works by repeatedly comparing adjacent elements in the array and swapping them if they are in the wrong order, and it continues until the array is fully sorted.

**iii. Apply Bubble Sort to Products**

Once the bubbleSort() function is defined, it is used to sort the $products array. The algorithm checks each pair of adjacent products and compares their attributes based on the $sortBy criterion. If the criterion is 'price', the function compares the prices of adjacent products; if the criterion is 'name', it compares the product names. If the products are not in the desired order, the algorithm swaps them to ensure the correct order is maintained.

**iv. Repeat Comparison and Swapping**

The Bubble Sort algorithm continues to iterate over the array, performing comparisons and swaps until all products are in the correct order. Each pass through the array moves the next largest (or smallest) product into its correct position. The process is repeated until no more swaps are needed, indicating that the array is fully sorted based on the selected criterion, whether by price (ascending or descending) or name (alphabetically).

**v. Return Sorted Products**

After applying the Bubble Sort algorithm, the $products array is now sorted according to the selected criterion. The sorted array is returned and can be used to display the products on the user interface, ensuring that the products are shown in the desired order, whether by price (in ascending or descending order) or alphabetically by name.

* **Popularity-Based Product Recommendation Algorithm**

The Popularity-Based Product Recommendation Algorithm is designed to recommend products that have been most frequently purchased or have the highest total sales. This algorithm uses a Bubble Sort approach to rank the products by the total quantity sold, allowing the application to suggest the top-selling products to users.

**Steps of the Algorithm:**

1. **Fetch Sales Data**

The first step involves executing an SQL query to retrieve sales data from the database. The query calculates the total quantity sold for each product by joining the order\_items and orders tables.

1. **Store Sales Data:**

The retrieved data, which includes the product ID and its corresponding total sales quantity, is stored in an array called $salesData. This array will later be used to rank the products based on their popularity.

1. **Apply Bubble Sort:**

The Bubble Sort algorithm sorts the products based on the total quantity sold. The sorting is done in descending order so that the products with the highest sales come first. Bubble Sort repeatedly compares adjacent elements and swaps them if they are in the wrong order. This process continues until the list is fully sorted. Bubble Sort is performed by comparing each pair of adjacent products. Swap their positions if the product’s total quantity is less than the next product. This process continues until all products are sorted by their sales or order quantity.

1. **Select Top N Popular Products**

After the Bubble Sort, the array of products is now sorted by popularity (total sales). The top N products are selected based on the sorted data. The array\_slice() function extracts the top N products, where N is typically set to 6 by default.

1. **Return Popular Product IDs:**

The product IDs of the top N most popular products are extracted from the sorted array and returned by the algorithm.

# **CHAPTER 4: IMPLEMENTATION AND TESTING**

## **4.1 Implementation**

The implementation phase involves the conversion of the well-defined system design into a fully functioning application. It encompasses writing the programming code, creating databases, designing user interfaces, and integrating various functionalities to provide a seamless user experience. This section provides an overview of the technical aspects of the development process, including the languages, frameworks, and tools used to build the presentation layer, business layer, and data layer. The chapter aims to highlight the successful realization of the project's objectives and the strategies employed to overcome challenges during development.

### **4.1.1 Tools Used**

The various system tools that have been used in developing both the front-end and back-end of the project are being discussed in this section.

**4.1.1.1 Front-End Tools**

* **HTML**

HTML, short for Hyper Text Markup Language, is the fundamental building block of web pages. It uses a series of tags to structure and organize the content, defining headings, paragraphs, images, links, and other elements that make up a web page. It provides the necessary structure for creating a cohesive and well-organized web page, forming the backbone of the entire web development process.

* **CSS**

CSS stands for Cascading Style Sheet, is a stylesheet language used to define the presentation and layout of HTML documents. It works hand-in-hand with HTML, allowing web developers to control the appearance of web pages, including colors, fonts, spacing, and positioning of elements.

* **JavaScript**

JavaScript is used to make our webpages interactive and many JavaScript functions such as dialogue box is used in this project to make webpages interactive and user friendly.

**4.1.1.2 Back End tools**

* **PHP**

PHP, which stands for Hypertext Preprocessor, is a server-side scripting language widely used for web development. It is embedded within HTML code to perform dynamic tasks, such as database access, form processing, and generating dynamic content. PHP code is executed on the server, generating HTML that is sent to the client's web browser. Its flexibility and ease of integration with databases and other technologies have made it a popular choice for building dynamic websites and web applications. Simply, it is used to develop dynamic and interactive webpages.

**4.1.1.3 Database**

* **MYSQL**

MySQL is an open-source and widely-used relational database management system that plays a crucial role in the development of dynamic web applications. It provides a robust and efficient platform for storing, managing, and retrieving data, making it a popular choice for developers seeking to build interactive and data-driven websites.

**4.1.1.4 Server**

* **Apache Server**

In Brikshya Griha, Apache server is used to run PHP files and creating fast and dynamic webpages.

**4.1.1.5 Documentation Tools**

* **MS Office**

This is used for writing and editing the documentation of ‘Brikshya Griha’.

* **Draw.io**

This is used to generate diagrams for system analysis and design of Brikshya Griha. All the diagrams were drawn using this tool in order to save time since all components are available with drag and drop functions.

### **4.1.2 Implementation Details of Modules**

The proposed system consists of following modules:

* **User Management Module:**

This module handles user registration, login, and profile management. It ensures secure user authentication and authorization, allowing users to create accounts, log in, and manage their personal information. It also includes features to view order history and update user details.

* **Product Management Module**

The product management module allows administrators to add, edit, and delete products from the inventory. This module includes functionalities to manage product categories, set pricing, and manage stock levels, including alerts for low stock situations. Admins can view product details and images, ensuring that the e-commerce platform is up-to-date with available inventory.

* **Cart Module**

The cart module enables users to add products to their shopping cart, view the items in the cart, and proceed to checkout. Users can adjust quantities, remove items, and calculate the total price before finalizing their orders. This module ensures that a seamless shopping experience is maintained throughout the purchasing process.

* **Order Management Module**

This module processes user orders after they complete the checkout process. It tracks order status (pending, shipped, completed, etc.) and manages order details, including payment information and shipping addresses. Admins can view all orders, update their statuses, and generate order reports for better inventory management.

* **Inventory Management Module**

The inventory management module monitors stock levels for all products. It alerts administrators when stock levels fall below a predefined threshold, ensuring timely restocking of popular items. This module helps prevent stockouts and improves inventory turnover.

* **Admin Dashboard Module**

The admin dashboard provides a comprehensive overview of the e-commerce platform's performance. It displays key statistics such as total users, total products, and recent orders. This module is crucial for admins to make informed decisions regarding product management and customer service.

* **Contact Management Module**

This module handles user inquiries and feedback. It allows users to submit contact forms, which are then stored for admin review. Admins can respond to user queries, improving customer engagement and satisfaction.

## **4.2 Testing**

On the basis of software requirement specification document, testing was performed to investigate and validate the behavior of a fully integrated software product. Before deploying an application or website, it must be thoroughly tested. Some of the types of testing that we did are described below:

### **4.2.1 Test Cases for Unit Testing**

Unit testing is performed to ensure the individual units or components of the software, such as functions, methods, or classes, work correctly and produce the expected output. It aims to validate that each isolated unit behaves as intended and is free from defects. We can catch bugs early in the development process, making it easier to fix issues and prevent them from propagating to other parts of the systems.

The unit tests that were performed in this project are tabulated below:

**Table 4.1:Test Case for User Registration Module**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Case Title** | **Test Steps** | **Test Data** | **Expected Result** | **Actual Result** | **Status** |
| TC-UR-01 | Validate successful user registration | Navigate to the registration page, fill out and submit the form. | First Name: Tara  Last Name: Sapkota  Email: [tara@gmail.com](mailto:tara@gmail.com)  Password: Tara@123  Phone:9808983861  Address: Tinthana | Registration Successful! | As Expected | Pass |
| TC-UR-02 | Validate user registration with incorrect email format | Navigate to the registration page. Fill and submit the form with an incorrect email format. | First Name: Tara  Last Name: Sapkota  Email: tara123  Password: Tara@123  Phone:9808983861  Address: Tinthana | Incorrect Email Format | As Expected | Pass |
| TC-UR-03 | Validate user registration with weak password | Navigate to the registration page. Fill and submit the form with a weak password. | First Name: Tara  Last Name: Sapkota  Email: tara123  Password: Tara@123  Phone:9808983861  Address: Tinthana | Password must be at least 8 characters long, with one uppercase, lowercase, number, and one special character. | As Expected | Pass |

**Table 4.2:Test Case for User Login**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Case Title** | **Test Steps** | **Test Data** | **Expected Result** | **Actual Result** | **Status** |
| TC-UL-01 | Validate successful user login | Navigate to the login page and enter valid credentials. | Email: [tara@gmail.com](mailto:tara@gmail.com)  Password: Tara@123 | Login successful | As Expected | Pass |
| TC-UL-02 | Validate user login with invalid password | Navigate to the login page and enter valid email and invalid password. | Email: [tara@gmail.com](mailto:tara@gmail.com)  Password: Tara123 | Incorrect password | As Expected | Pass |
| TC-UL-03 | Validate user login with non-registered email | Navigate to login page and enter a non-registered email. | Email: [sarada@gmail.com](mailto:sarada@gmail.com)  Password:Sarada@123 | Email not found | As Expected | Pass |

**Table 4.3:Test Case for Product Management Module**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Case Title** | **Test Steps** | **Test Data** | **Expected Result** | **Actual Result** | **Status** |
| TC-PM-01 | Validate adding a new product | Navigate to product management, fill in product details and click add product | Product Name:’Fern’  Price: Rs.1000  Quantity:10 | Product added successfully | As Expected | Pass |
| TC-PM-02 | Validate adding a product with missing fields | Navigate to product management and leave required field empty and click add product | Product Name:’’  Price: Rs.1000  Quantity:10 | Product name is required | As Expected | Pass |
| TC-PM-02 | Validate updating an existing product | Navigate to product management, select an existing product and click on edit. | Product ID: 1 New Price: 12.99 | "Product updated successfully!” | As Expected | Pass |

**Table 4.4:Test Case for Cart Module**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Case Title** | **Test Steps** | **Test Data** | **Expected Result** | **Actual Result** | **Status** |
| TC-  CM-  01 | Validate adding a product to the cart | Navigate to the shop or product\_details page and Click on add to cart. | Product ID: 1 | Product added to cart | As Expected | Pass |
| TC-CM-02 | Validate removing a product from the cart | Navigate to the cart and click remove on a product | Product ID: 1 | Product removed from cart. | As Expected | Pass |
| TC-CM-03 | Validate viewing cart items | Navigate to the cart and check displayed products | View cart page | Displayed products match cart items. | As Expected | Pass |

**Table 4.5:Test Case for Checkout Module**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Case Title** | **Test Steps** | **Test Data** | **Expected Result** | **Actual Result** | **Status** |
| TC-CM-01 | Validate successful checkout | Navigate to checkout page and fill in shipping details and confirm order. | Address: Tinthana  Phone number:9812936471 | Order placed successfully! | As expected | Pass |
| TC-CM-02 | Validate checkout with empty cart | Navigate to checkout page. Try to confirm order with an empty cart. | Empty the cart and try to confirm order. | Your cart is empty | As expected | Pass |

**Table 4.6:Test Case for Logout Module**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Case Title** | **Test Steps** | **Test Data** | **Expected Result** | **Actual Result** | **Status** |
| TC-LO-01 | Validate successful logout | Click Logout | Click on logout | You have been logged out. | As expected | Pass |
| TC-LO-02 | Validate logout from any page. | Navigate to any page and click logout | Click on logout | You have been logged out | As expected | Pass |

### **4.2.2 Test Case for System Testing**

System testing is focused on evaluating the entire system as a whole to ensure that all the integrated components work together as expected. System testing is done by giving different training and testing datasets. This test is done to evaluate whether the system is providing accurate summary or not. During the phase of the development of the system, our system is tested time and again. In system testing, whole system is tested as below:

**Table 4.7:Test Case to verify Admin Module**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Module** | **Test Objective** | **Preconditions** | **Test Steps** | **Expected Result** | **Actual Result** | **Status** |
| ADM-001 | Admin Login | Verify login with correct admin credentials | Admin account exists in the system. | Enter valid admin email ad password and click Login | Admin is logged in and redirected to admin dashboard. | As expected | Pass |
| ADM-002 | View Sales Report | Verify admin can view sales report. | Sales data exists in the database | Navigate to “Sales Report” section from the dashboard | A list of all sales record displays with details. | As expected | Pass |
| ADM-003 | Download Sales Report | Ensure sales report can be downloaded. | Sales report page is accessible. | Click “Download Report” | Sales report downloads as a CSV file with all relevant sales data. | As expected | Pass |
| ADM-004 | Manage Products | Confirm admin can add, edit and delete products | Admin is on the product management page. | Fill in product details and click on add product. | Product is successfully added, edited and deleted confirmation message appears. | As expected | Pass |
| ADM-005 | Manage Categories | Ensure admin can add categories | Admin is on the category management page. | Enter new category name and click on add category. | Category is added and confirmation message appears. | As expected | Pass |
| ADM-006 | Manage Orders | Verify order status update | Order exists in the database. | Open “Orders” page and change the order status. | The order status is updated and user can see in track order. | As expected | Pass |
| ADM-007 | Logout | Verify admin can logout successfully. | Admin is logged in. | Click Logout button. | Admin logged out and redirected to index page. | As expected | Pass |

**Table 4.8:Test Case to verify User Module**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Module** | **Test Objective** | **Preconditions** | **Test Steps** | **Expected Result** | **Actual Result** | **Status** |
| USR-001 | User Registration | Verify successful user registration with valid data | User is on the registration page. | Fill out registration form with valid data and submit the form. | Account is created and a success message is displayed. | As expected | Pass |
| USR-002 | User Login | Validate login with correct credentials. | User is registered. | Enter valid email and password and click on Login. | User is redirected to shop page | As expected | Pass |
| USR-003 | Product Search | Ensure product search works as expected. | Products exist in database | Enter a product name in the search bar and click search. | Relevant products are displayed based on search query. | As expected | Pass |
| USR-004 | Add to Cart | Confirm items can be added to cart. | User is logged in and viewing products. | Select a product and click add to cart | Item is added to the cart and a success message id displayed. | As expected | Pass |
| USR-005 | View Cart | Verify items in the cart are displayed accurately | Items are added to the cart. | Go to the cart page. | All added items are shown with correct name, quantities and prices. | As expected | Pass |
| USR-006 | Update Cart Quantity | Ensure cart updates correctly when quantity changes. | Items are in the cart | Change quantity of an item in the cart and click update. | Cart updates with new quantity and the total price is recalculates. | As expected | Pass |
| USR-007 | Remove from Cart | Validate removal of items from cart. | Items are in the cart. | Click “Remove” on an item in the cart. | Item is removed and cart updates the total. | As expected | Pass |
| USR-008 | Checkout | Test checkout with “Cash on Delivery”. | Items are in the cart and user is logged in. | Goto the checkout page and select cash on delivery and confirm the order. | Order is placed and confirmation message is displayed. | As expected | Pass |
| USR-009 | Order History | Confirm user can view past orders. | User has placed previous orders. | Go to Order History. | User’s order history shows past orders with details. | As expected | Pass |
| USR-010 | Submit Review | Ensure users can submit product reviews | User has purchased product | Goto purchased product and give a rating and comment. | Review is saved and a success message is displayed. | As expected | Pass |
| USR-011 | Logout | Verify user can logout successfully. | User is logged in. | Click Logout button. | User logged out and redirected to index page. | As expected | Pass |

# **CHAPTER 5: CONCLUSION AND FUTURE RECOMMENDATION**

## **5.1 Conclusion**

The development of this e-commerce website has successfully achieved the primary goals of creating a user-friendly, reliable platform for purchasing plants and plant-related accessories. Throughout the project, I implemented essential features such as product browsing, user registration and login, a secure checkout process, and efficient order management. Additionally, I incorporated administrative capabilities, allowing for streamlined management of products, sales, and user interactions. This system not only improves the shopping experience for customers but also provides an efficient management tool for administrators. The result is a functional, accessible, and scalable e-commerce platform that can accommodate a variety of future enhancements.

## **5.2 Lesson Learnt / Outcome**

Every project makes us to learn and gain the knowledge in different aspects. From this project, I have learned lots of problem-solving skills and learn things like improving technical skills, understanding full software development cycle, database design and management, problem solving, troubleshooting, documentation and communication. Here are some potential lessons learned and outcomes of our project:

* **Improves Technical Skill**

During the development of our project, I likely enhanced my technical skills in programming languages, database management, web development, and working with frameworks and libraries. These skills can be valuable for future projects and career opportunities

* **Understanding full software development life cycle**

I gained practical experience in the entire software development cycle, from requirements gathering and planning to implementation, testing, and deployment. This understanding can help me to become more efficient and effective in future development projects.

* **Database Design and Management**

Developing a web application with a database backend requires careful database design and management. Because of this project, I likely learned how to create and manage database tables, set up relationships between tables, and handle data efficiently.

* **Problem Solving and Troubleshooting**

Throughout the development process, I likely encountered various challenges and issues. My ability to troubleshoot and find solutions to these problems has likely improved.

* **Documentation and Communication**

Throughout the development process, I likely encountered various challenges and issues. I likely learned how to document the report as well as increase my communication skills.

## **5.3 Future Recommendation**

There are many things to be added in future to improve this system. We are planning to keep managing the project and improving it based on user feedback.

Here is my to do list for future:

* Introduce multiple payment methods, such as online payments, credit cards, and digital wallets, to increase convenience and attract a wider customer base.
* Create a fully responsive mobile version of the website to enhance accessibility for users shopping on smartphones and tablets.
* Improve the product search and filtering options by adding filters like price range, category, and popularity, enabling users to find products more easily.
* **Add more product categories and varieties** to provide a wider selection for customers, making it a comprehensive platform for plant-related products.

# **REFERENCES**

[1] "Plant," Collins Dictionary. Available: <https://www.collinsdictionary.com/dictionary/english/plant>. Accessed: Nov. 10, 2024.

[2] "Plants and Decor," Available: <https://plantanddecor.com/>. Accessed: Nov. 10, 2024.

[3] E-nursery Nepal. Available: <https://www.enurserynepal.com/>. Accessed: Nov. 10, 2024.

[4] "Pashupati Nursery," Available: <https://pashupatinursery.com/>. Accessed: Nov. 10, 2024.

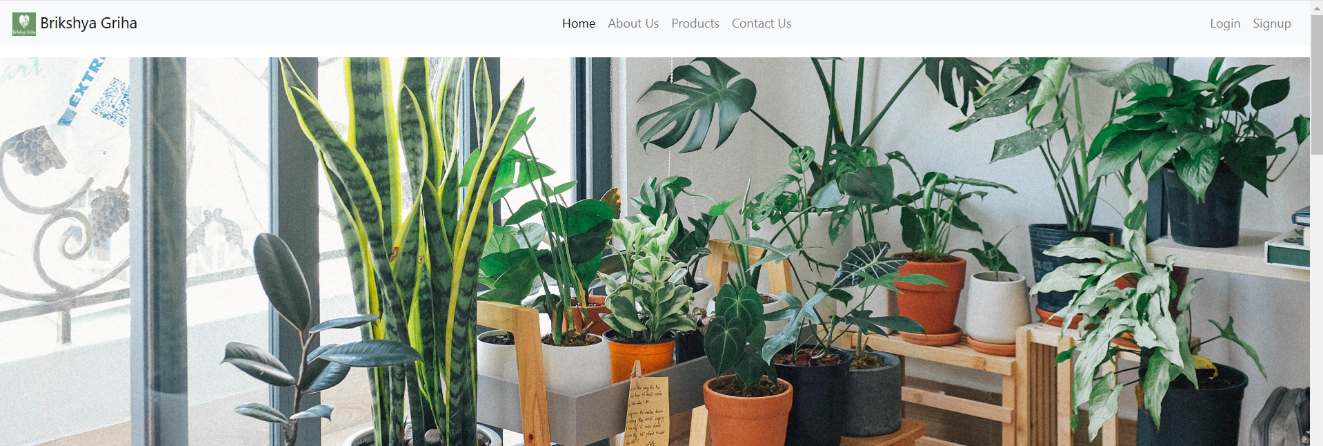
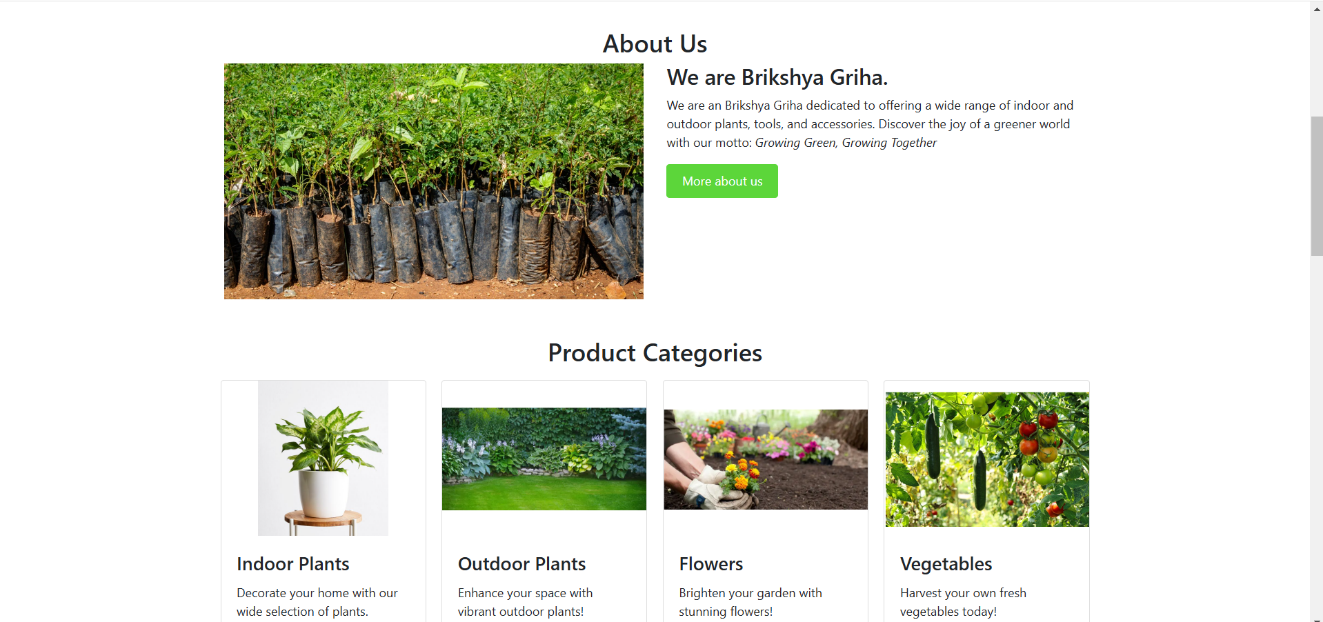
[5] "Gamala Ghar," Available: <https://www.gamalaghar.com/>. Accessed: Nov. 10, 2024.

[6] "Feasibility Analysis," Simplilearn. Available: <https://www.simplilearn.com/feasibility-study-article#:~:text=As%20the%20name%20implies%2C%20a,project%20may%20not%20be%20doable>.. Accessed: Nov. 10, 2024.

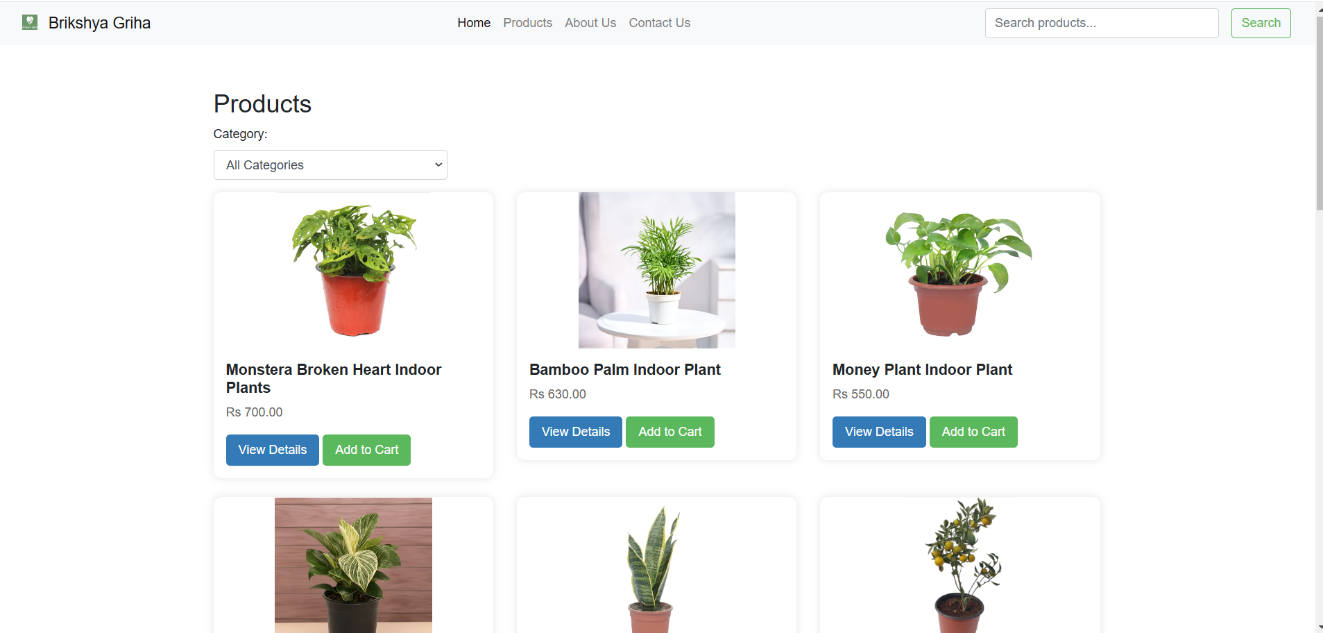
[7] "Technical Feasibility," Indeed. Available: <https://in.indeed.com/career-advice/career-development/what-is-technical-feasibility#:~:text=Technical%20feasibility%20is%20the%20formal,every%20step%20of%20the%20operation>.. Accessed: Nov. 10, 2024.

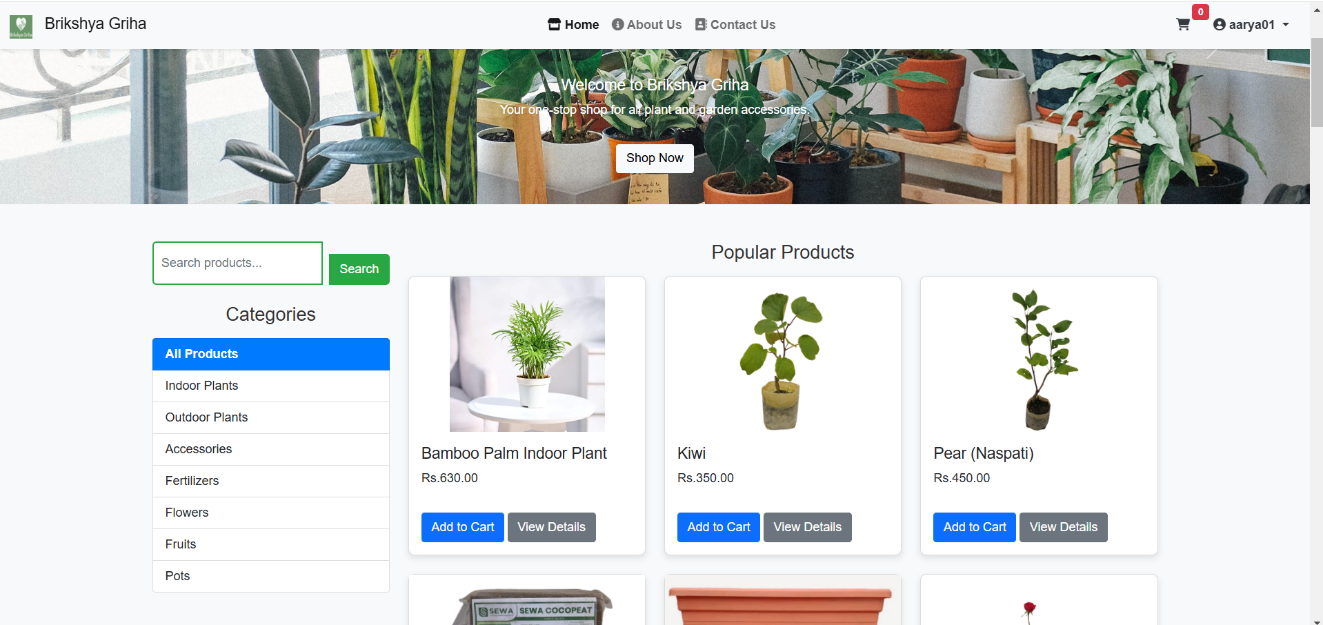
[8] "Operational Feasibility," QS Study. Available: <https://qsstudy.com/operational-feasibility/>. Accessed: Nov. 10, 2024.

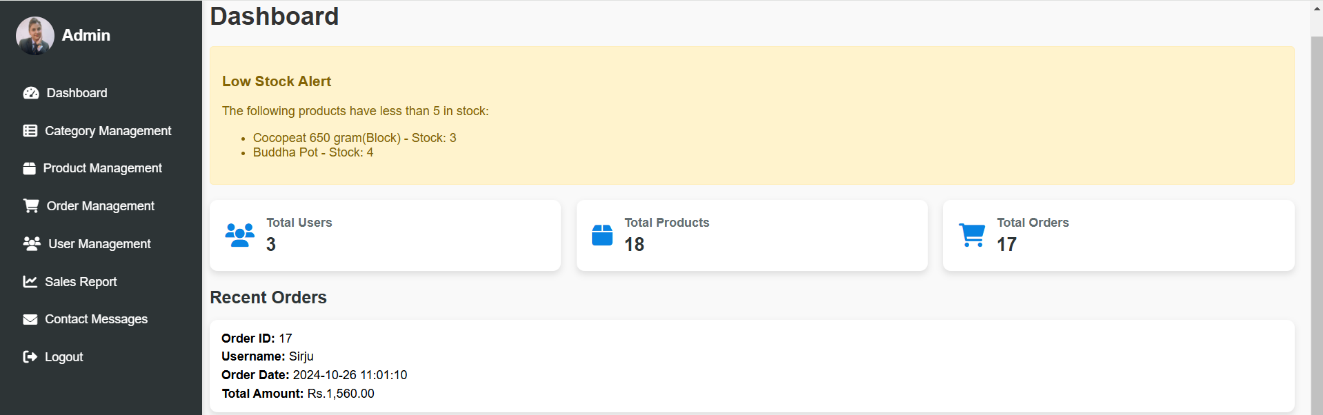
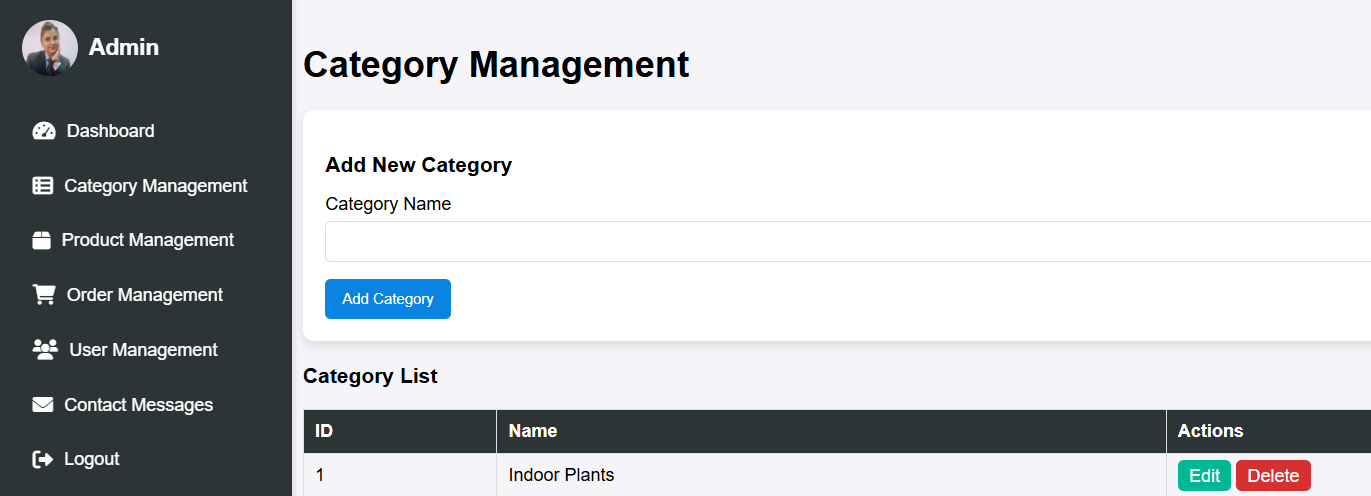
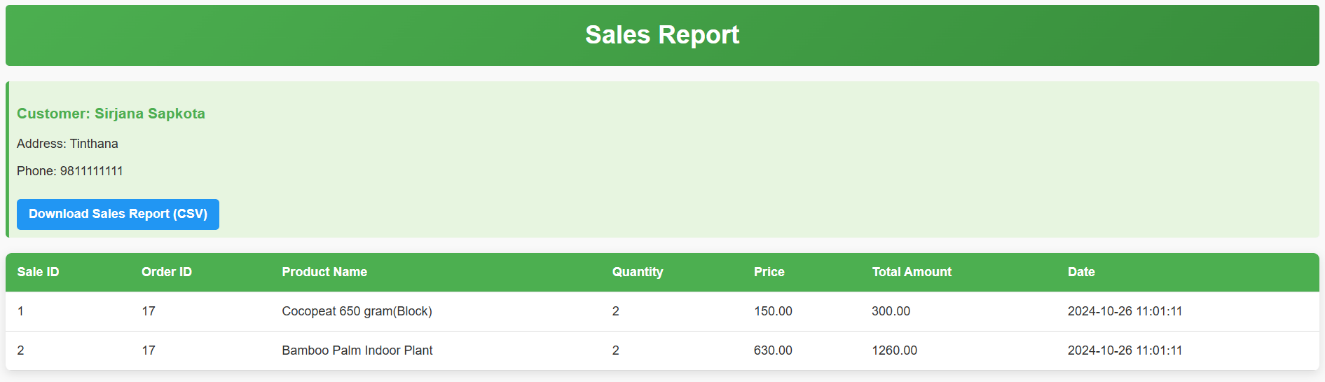
[9] "Process Modeling and Mapping," Bizagi. Available: <https://www.bizagi.com/en/blog/process-modeling-and-mapping/what-is-process-modeling-6-essential-questions-answered#:~:text=Process%20modeling%20is%20the%20graphical,context%20of%20the%20business%20environment>.. Accessed: Nov. 10, 2024.

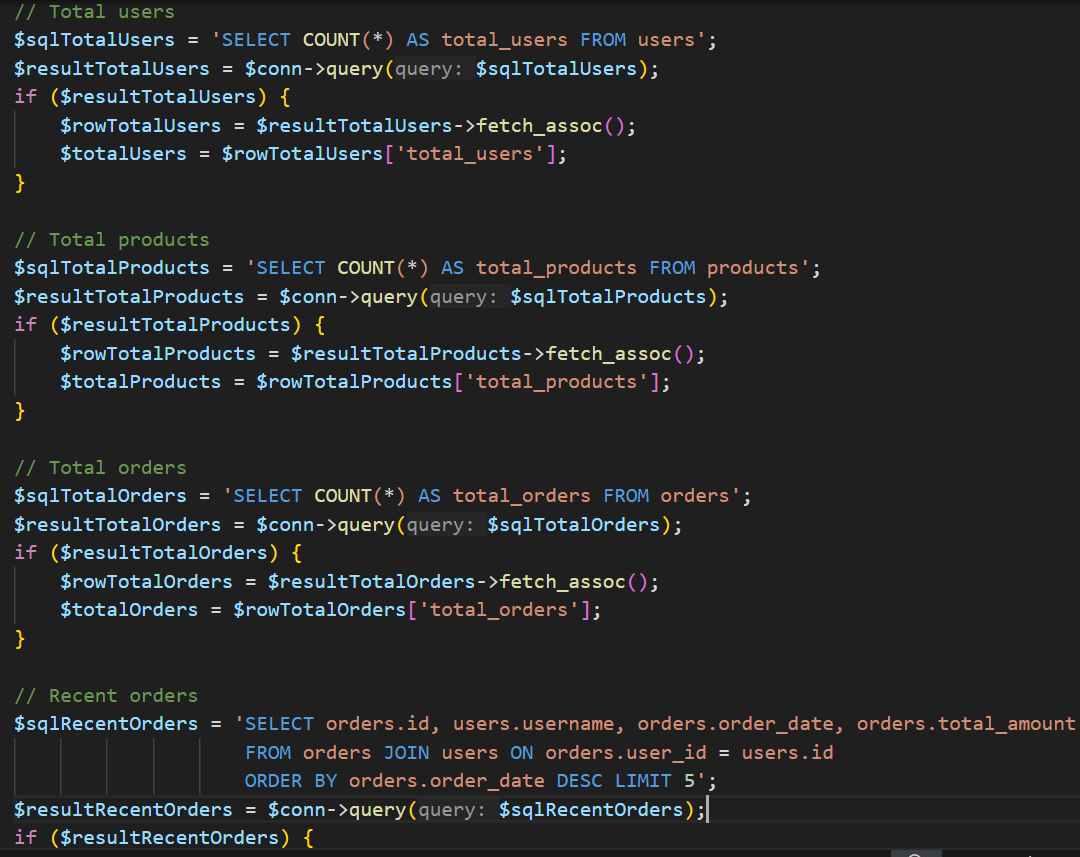
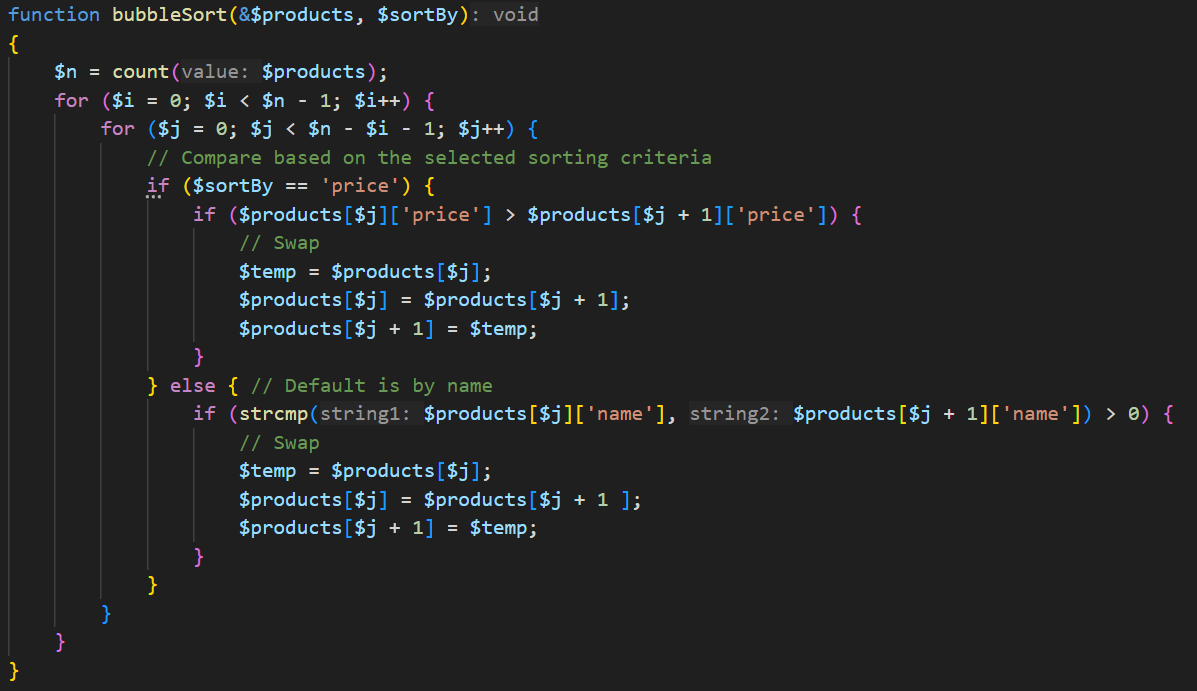
**APPENDICES**

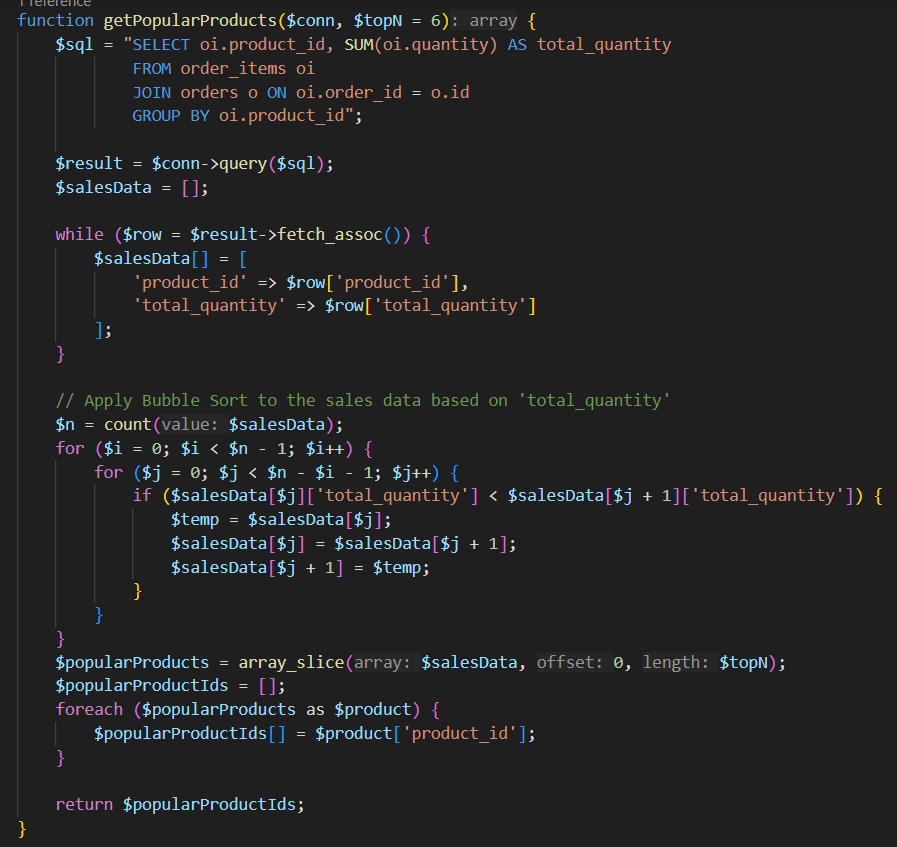
|  |
| --- |
|  |



****

****

**Source Code**

****