
Certifications

This is to certify that the project titled **SHOP MANAGEMENT SYSTEM** presented by **RASHMA DEV** (ID: 1703310201530) and **BIBI KHADIJA SADIYA** (ID: 1703310201539) has been deemed satisfactory in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science & Engineering (CSE) to be conferred by Premier University, Chattogram.

Prof. Dr. Taufique Sayeed

Chairman

Department of Computer Science & Engineering

Premier University, Chattogram

Premier University (Signature & Date)

Chattogram

[Supervisor]

Asma Joshita Trisha

Assistant Professor

Department of Computer Science & Engineering

Premier University, Chattogram

Premier University (Signature & Date)

Chattogram

Author's Declaration of Originality

We hereby declare that the project work entitled **SHOP MANAGEMENT SYSTEM** submitted to Premier University is an original work conducted by us, under the guidance of **Asma Joshita Trisha**, Assistant Professor in the Department of Computer Science & Engineering at Premier University, Chattogram. This work is being submitted to fulfill the requirements for the degree of Bachelor of Science in Computer Science & Engineering.

We affirm that the outcome of this project has not been previously submitted to any other university or educational institution.

RASHMA DEV

Student ID: 1703310201530

Department of Computer Science & Engineering

Premier University, Chattogram

BIBI kHADIJA SADIYA

Student ID: 1703310201539

Department of Computer Science & Engineering

Premier University, Chattogram

Dedications

In profound gratitude and sincere appreciation, We dedicate this project titled **SHOP MANAGEMENT SYSTEM** to our esteemed mentor, Asma Joshta Trisha mem, Assistant Professor in the Department of Computer Science & Engineering at Premier University, Chattogram. Her invaluable guidance, unwavering support, and profound knowledge have been instrumental in shaping our journey throughout this project. Her dedication and commitment to fostering our academic growth have inspired us to reach new heights. This dedication is a token of our deepest appreciation and a testament to the invaluable role played by our mentor and the institution in our academic journey.

Acknowledgement

We extend our heartfelt gratitude and appreciation to all those who have contributed to the successful completion of our project entitled **SHOP MANAGEMENT SYSTEM** for Premier University. This work has been a result of collective efforts, guidance, and support from various individuals and institutions, and we would like to express our sincerest thanks to each one of them.

First and foremost, we would like to express our deepest gratitude to our project supervisor, **Asma Joshta Trisha** mem, Assistant Professor in the Department of Computer Science & Engineering at Premier University, Chattogram. Her expertise, continuous guidance, and unwavering support have been invaluable throughout the entire project. Her insightful suggestions, timely feedback, and encouragement have greatly contributed to the success of this endeavor. We are truly grateful for her mentorship.

Furthermore, we would also like to acknowledge with much appreciation the crucial role of Professor **Dr. Taufique Sayeed**, Chairman, Department of CSE, Premier University and every teacher of Department of Computer Science and Engineering for their guidance as well as constant supervision and persistent motivation.

Thank you all once again for your unwavering support, guidance, and encouragement.

Abstract

The **Shop Management System** is a user-friendly and accessible solution designed to simplify daily business operations. Unlike traditional software, our system does not require any installation or local processing, as it can be accessed directly through a web browser. This eliminates the need for complex setups and allows users to manage their shop from anywhere in the world, using any computer with an internet connection. Our software offers a wide range of features that cater to the specific needs of shop management. From inventory management and sales tracking to customer relationship management and employee management, our system provides comprehensive tools to streamline processes and increase efficiency. The software ensures that even non-technical users can navigate through the system effortlessly. By providing real-time insights and accurate data, our software empowers shop owners to make informed decisions and optimize their operations for maximum productivity. The manufacturer design of this project uses React JS, Chart JS, Custom CSS, HTML, as a frontend and Node JS, Express JS, Node Malier as a backend development of this project. For storing data it uses Mysql database system. In conclusion, the **Shop Management System** is a practical and efficient solution that revolutionizes the way shops are managed. By leveraging the power of technology, our software simplifies shop management, enhances productivity, and drives growth.

Table of Contents

Certifications	i
Author's Declaration of Originality	ii
Dedications	iii
Acknowledgement	iv
Abstract	v
1 Introduction	2
1.1 Introduction	2
1.2 Motivation	2
1.3 Objectives	2
1.4 Expected Outcomes	3
1.5 Challanges	3
1.6 Summary	3
2 Literature Review	4
2.1 Introduction	4
2.2 Related Works	4
2.3 Comparative Studies	4
2.4 Summary	5
3 Methodology	6
3.1 Methodology	6
3.2 Software Development Methodology(SDM's)	6
3.3 Process Models	6
3.4 Software Process Model	7
3.5 Justification's of the selected Methodology	7
3.6 Summary	7
4 System Analysis	8
4.1 Requirement collections and analysis	8
4.2 Software Requirements	8
4.3 Design Requirements:	9
4.4 Use Case Modelling and Description	9
4.5 Use Case Diagram	9

4.6	Use Case Diagram Components	10
4.7	Use Case Symbols and Notation's	10
4.8	Use Case Diagram of Our System(Customer order process)	11
4.9	Components Entity Relationship Diagram	12
4.10	ER Diagram of Our System	12
4.11	Flow diagram of system	13
4.12	Activity diagram of Shop Management system	17
5	Design specifications	20
5.1	Front-End-Design	20
5.2	Back-End-Design	20
5.3	Implementation Requirements	21
5.3.1	Hardware Requirements	21
5.3.2	Software Requirements	21
5.3.3	Development Tools	21
5.3.4	Third-Party Libraries and Frameworks	21
5.4	Summary	21
6	Implementation & Testing	22
6.1	Implementation of database	22
6.1.1	Customer Registration	22
6.1.2	Customer Orders	22
6.1.3	Customer Payments	23
6.1.4	Order State	23
6.1.5	Cart Product	23
6.1.6	Cart Payment Info	24
6.1.7	Cart Invoice	24
6.1.8	Order Confirmation	24
6.1.9	Implementation of deliveryman database	24
6.1.10	Implementation of Products database	26
6.1.11	Implementation of Satff's database	26
6.2	Implementation of Front-end Design	27
6.3	Test result and Report	45
6.4	Summary	46
7	Limitation , Future scope & Conclusion	47
7.1	Limitation	47
7.2	Future Scope	47
7.3	Conclusion	48
7.4	References	48

List of Figures

3.1	Iterative Incremental model in Designing System	6
4.1	Use Case Notation and Symbols	10
4.2	Use case diagram of order process and office management	11
4.3	Entity Relation Diagram of Shop Management system	12
4.4	Flow diagram of Admin Panel	13
4.5	Flow diagram of Staff Panel	14
4.6	Flow diagram of Deliveryman Panel	15
4.7	Flow diagram of Customer Panel	16
4.8	Activity diagram of Admin Panel	17
4.9	Activity diagram of Customer Order Process	18
4.10	Activity diagram of Deliveryman	19
6.1	Shop Management Software System Homepage	27
6.2	Section of Shop management homepage	28
6.3	Product Details	29
6.4	Order Create Process	30
6.5	Create Cart Order	31
6.6	Cash on delivery method	31
6.7	Auto generated Cart Invoice	32
6.8	Check order state by searching order ID	33
6.9	Product Filtering	33
6.10	Staff Registration Form	34
6.11	Staff Login Form	35
6.12	Staff dashboard	35
6.13	Visual and Manual Implementations of Revenue	36
6.14	Check orders	36
6.15	Hire Delivery man or selects services	37
6.16	Costs specifications	38
6.17	CRUD operations on Products	38
6.18	Insert New Products	39
6.19	Insert New Products Category	40

6.20 Update existing product	40
6.21 Find products via search and category wise	41
6.22 Hire new deliveryman form	41
6.23 Active delivery persons	42
6.24 Delivery man Login	42
6.25 Delivery man Dashboard	42
6.26 Check new orders	43
6.27 Enter the order id To Validate customers	43
6.28 Handled orders to the customers	44

List of Tables

6.1	Test cases for User Interface	45
6.2	Test cases for Product Buying Process	45
6.3	Test cases for Staff Management Process	45
6.4	Test cases for Delivery Management Process	45

Chapter 1

Introduction

1.1 Introduction

Introducing the revolutionary **Shop Management System**, a real-time inventory software that transforms shop management. This user-friendly system offers an online platform accessible from anywhere, ensuring convenience and efficiency. With robust database management and advanced features, it streamlines shop operations and revolutionizes business management. From sales order management to real-time tracking, it enhances productivity and provides seamless organization of shop data. With additional features like bill generation, multi-channel sales support, and inventory value views, it empowers shop owners to optimize sales processes and effectively manage their businesses. Experience a new level of efficiency with the **Shop Management System**.

1.2 Motivation

The motivation behind the development of the **Shop Management System** is to introduce shop management by addressing business challenges in the digital era. The system streamlines operations, improves productivity, and optimizes customer management. It offers flexibility, security, and features like order management, real-time tracking, and inventory insights. The goal is to empower shop owners with an efficient and comprehensive solution for effective business management.

1.3 Objectives

1. Enhance operational efficiency with a user-friendly interface and intuitive features.
2. Optimize the sales process with features like order management.
3. Ensure seamless organization and retrieval of shop data with a well-defined database management system.
4. Gain insights into inventory value for informed decision-making.
5. Improve administrative efficiency with integrated office management tasks.
6. Efficiently track and manage product deliveries with a delivery management system.

1.4 Expected Outcomes

1. Customers have a better buying experience with smooth order management and tracking.
2. The system keeps shop data organized, helping owners make smart decisions about inventory and money.
3. It makes office work easier by putting everything in one place and saving time.
4. Deliveries are managed well, ensuring orders arrive on time.
5. Overall, The system makes shop tasks easier, like managing inventory, processing orders, and keeping customers happy.

1.5 Challenges

Employees require proper training and support to effectively learn and adapt to the new system. Careful planning and execution are needed to transfer and integrate existing shop data with other systems. An intuitive and user-friendly interface enhances user experience. Strong security measures are necessary to protect shop data. Choosing a reliable software vendor is vital for successful implementation. Evaluating costs and benefits helps businesses make informed decisions.

1.6 Summary

The **"Shop Management System"** revolutionizes shop management with its real-time inventory software. It provides a user-friendly frontend, robust database management, and CRM systems. The system streamlines operations, ensures data security, tracks orders, and offers insights into inventory value. Additional features include automatic bill generation, support for multiple sales channels, and delivery management. Overall, the system empowers businesses to optimize efficiency, improve sales processes, and effectively manage their shops, leading to increased productivity and customer satisfaction.

Chapter 2

Literature Review

2.1 Introduction

The literature review of the **Shop Management System** aims to provide an overview of existing related works to the topic. This section explores various studies that have examined the concept of shop management software systems.

2.2 Related Works

For accomplish the advanced software system we examined different types of online platform which has best recognition in field of Online shopping system.Those are :

- **Aarong:** An online platform known for ethically handcrafted products.
- **Daraz:** A multi-product seller platform in Bangladesh.
- **Ryans Computer:** The largest chain of computer stores in BD.
- **Smart Eshop BD:** An online shopping platform in Bangladesh.

These platforms were analyzed for their user interface, inventory management, customer relationship management, and security aspects. Emphasis was given to intuitive navigation, efficient inventory control, and integration of features to enhance customer satisfaction and targeted marketing. The studies also considered security measures, encryption protocols, and access controls to safeguard sensitive shop data.

2.3 Comparative Studies

The studies aimed to assess various shop management software systems for different businesses. We compared their user interface, inventory management, and customer relationship features. We evaluated the systems ease of use, navigation, and user satisfaction.

2.4 Summary

The literature review of this project explores the current state of knowledge regarding **Shop Management Systems**. The review reveals the importance of user-friendly interfaces, efficient inventory tracking, and effective customer data management. The literature review provides a solid foundation for the project, identifying gaps and informing future directions.

Chapter 3

Methodology

3.1 Methodology

The project methodology follows a systematic approach, involving data collection from various sources and practical implementation. It evaluates the functionalities and features of Shop Management Systems.

3.2 Software Development Methodology(SDM's)

The chosen SDM for this project is the Agile framework. It involves iterative development with short sprints, prioritizing tasks and collaboration. This approach promotes adaptability, transparency, and customer satisfaction, ensuring efficient software development.

3.3 Process Models

The process model utilized in this project follows a structured approach to guide the execution and management of project activities. It encompasses several key stages, including project initiation, planning, executions monitoring.

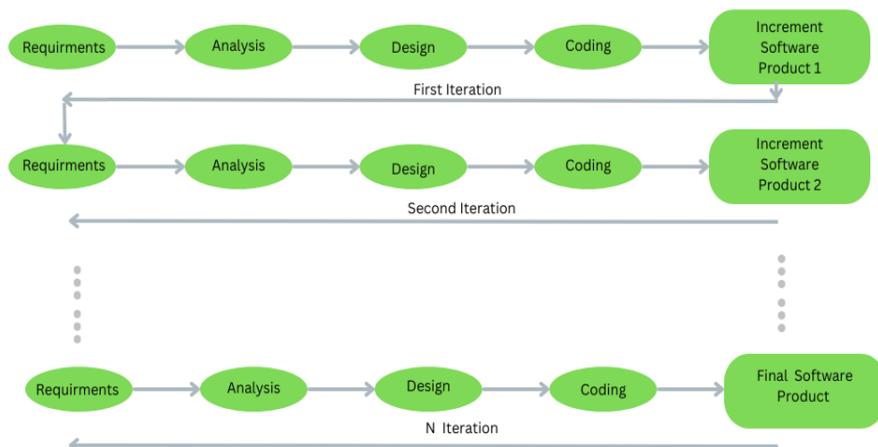


Figure 3.1: Iterative Incremental model in Designing System

3.4 Software Process Model

The software process model adopted for this project is the iterative and incremental model. The incremental and iterative development process is closely associated with Agile project management. Each iteration produces a partial working solution, which is incrementally refined and improved in subsequent iterations.

3.5 Justification's of the selected Methodology

- Incremental and iterative development is crucial in Agile software development.
- It allows for early changes and avoids delays and wasted resources.
- Incremental development ensures progress is made throughout the project timeline.
- Iterative development leads to ongoing improvements and higher-quality outcomes.

3.6 Summary

The chosen methodology for this project is the iterative and incremental model, which emphasizes the delivery of working software in iterative cycles. It promotes collaboration, adaptability, and prioritization of features, allowing for early feedback and continuous improvement. This approach is well-suited for projects with evolving requirements and ensures an efficient and effective software development process.

Chapter 4

System Analysis

4.1 Requirement collections and analysis

Requirements collection and analysis are crucial in system analysis. It ensures a clear understanding of project objectives and facilitates informed decision-making throughout the project lifecycle.

4.2 Software Requirements

To develop a Shop Management System, the following software requirements are used:

1. **React**: A JavaScript library for creating interactive user interfaces in web development.
2. **Node.js**: A runtime environment used for server-side development, allowing the system to handle server logic and connect with databases and APIs.
3. **XAMPP**: A software package that includes an Apache web server, MySQL database, and PHP, providing a local testing environment for the system.
4. **Bootstrap**: A popular framework with pre-designed components and stylesheets, making it easier to create visually appealing and mobile-friendly interfaces.
5. **Chart.js**: A JavaScript library for visualizing data, enabling the creation of various charts and graphs to display sales data and analytics within the system.

These software requirements form the essential components for developing a robust and functional Shop Management Software System. They cover frontend and backend technologies, database management, and data visualization capabilities.

1. **Express.js**: A web application framework for Node.js, Express.js is used to build the backend API and handle HTTP requests and responses.
2. **JWT (JSON Web Tokens)**: JWT is employed for authentication and secure communication between the client and server, ensuring that only authorized users can access the system.

-
3. **Email Notifications:** The system can be equipped with email notification functionality to send automated order confirmations, shipping updates, and promotional offers to customers.

These additional requirements enhance the functionality and user experience of the Shop Management System, ensuring secure authentication, seamless payment processing, efficient communication, and enhanced customer engagement.

4.3 Design Requirements:

- User-friendly interface for efficient navigation and task performance.
- Responsive design to ensure consistent user experience across devices.
- Robust database design for efficient storage and retrieval of shop-related data.
- Scalability to accommodate increasing data volumes and user traffic.
- Strong security measures to protect sensitive shop data.
- Reporting and analytics capabilities for generating sales reports and inventory analysis.
- Integration capabilities with payment gateways, shipping providers, and third-party applications.
- Customizability options to meet individual shop requirements.

4.4 Use Case Modelling and Description

1. **Manage Inventory:** Add, update, and track inventory items.
2. **Process Sales Orders:** Handle customer sales orders.
3. **Customer Management:** Manage customer information.
4. **Generate Reports:** Generate sales, inventory, and financial reports.
5. **User Management:** Manage user accounts and access levels.
6. **Payment Gateway Integration:** Enable E-payments.

These use cases define the key functionalities of the Shop Management System, including inventory management, sales order processing, customer management, reporting, user access control, payment integration, and data analysis.

4.5 Use Case Diagram

A Use Case Diagram is a visual representation of a system's functionalities and interactions between actors and the system. It helps define system boundaries and guide system design and development.

4.6 Use Case Diagram Components

Use Cases : represent system functionalities and interactions with actors.

Relationships : define connections and dependencies between actors and use cases.

System Boundary : sets the scope of the system. These components provide a visual representation to understand system behavior and requirements.

4.7 Use Case Symbols and Notation's

Actor: Represents a user or external system interacting with the system.

Use Case: Represents a specific functionality or task provided by the system.

Association: Shows the association between an actor and a use case.

Generalization/Inheritance: Indicates specialization and inheritance between use cases or actors.

Include: Represents the inclusion of one use case within another.

Extend: Indicates an extension of one use case by another.

These symbols simplify the representation of interactions and relationships in a use case diagram.

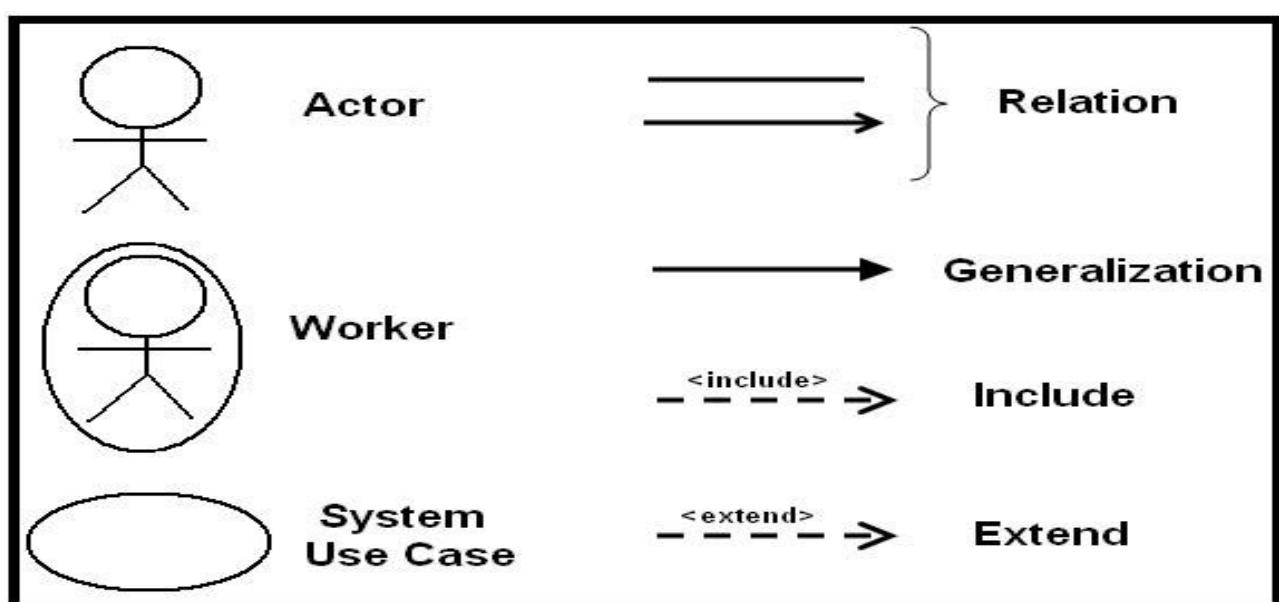


Figure 4.1: Use Case Notation and Symbols

4.8 Use Case Diagram of Our System(Customer order process)

Use case diagram of shop management system 4.2



Figure 4.2: Use case diagram of order process and office management

4.9 Components Entity Relationship Diagram

An ER diagram includes entities, attributes, relationships, cardinality, primary and foreign keys, weak entities, and aggregations. It visualizes a database's structure and constraints.

4.10 ER Diagram of Our System

Entity Relation's diagram of shop management system in 4.3

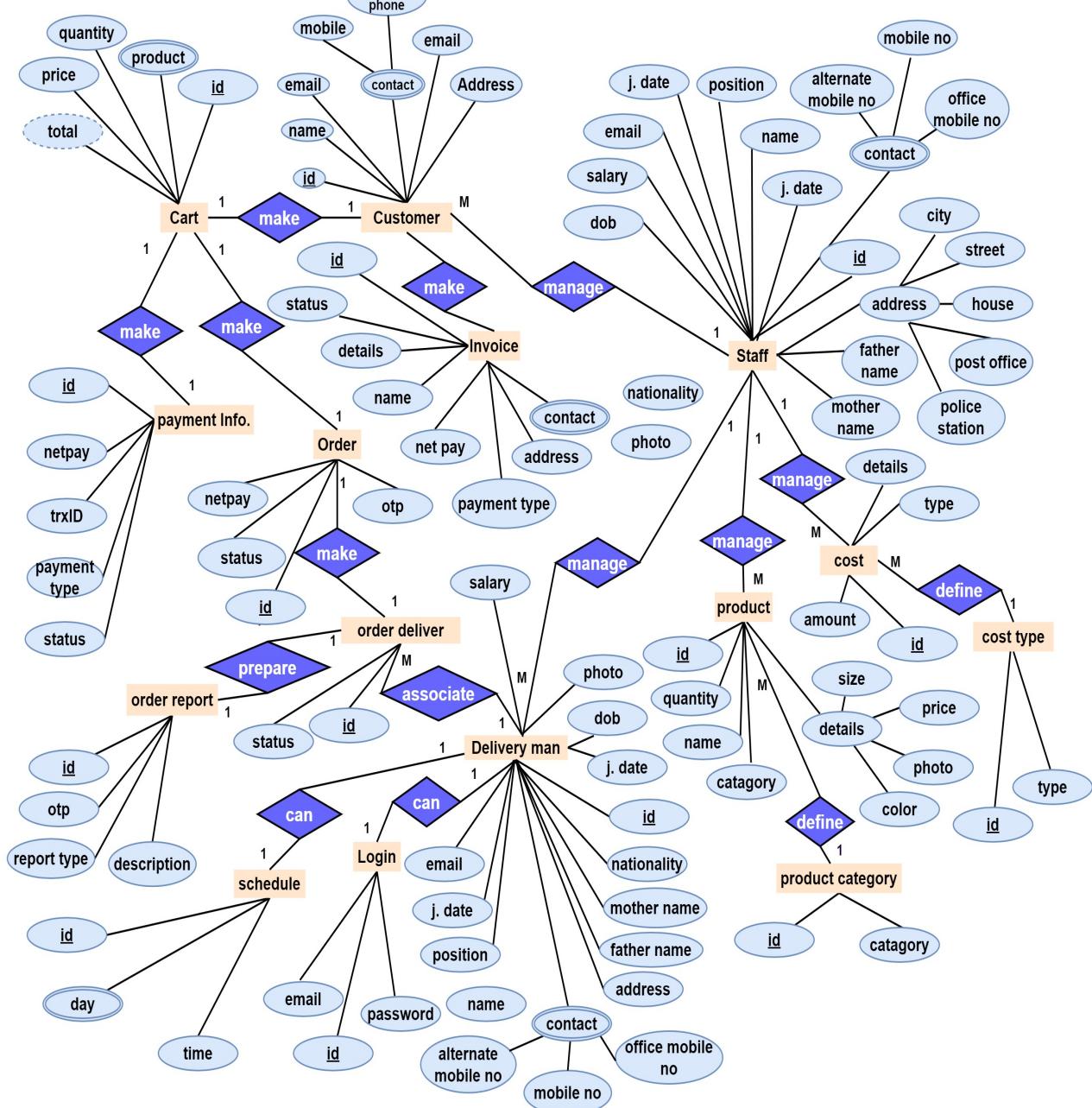


Figure 4.3: Entity Relation Diagram of Shop Management system

4.11 Flow diagram of system

Workflow diagram for Admin in 4.4

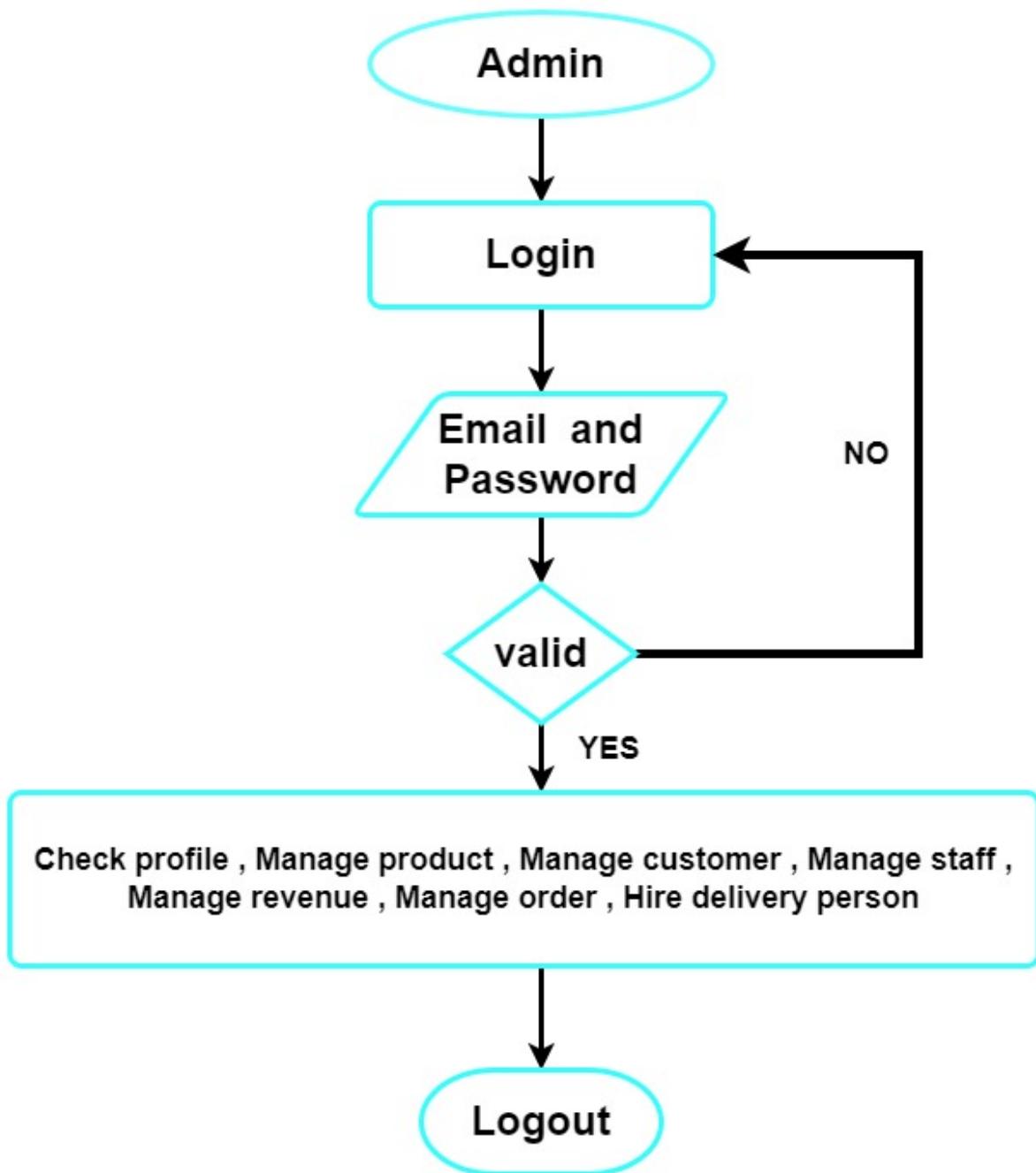


Figure 4.4: Flow diagram of Admin Panel

Workflow diagram for Staff in 4.5

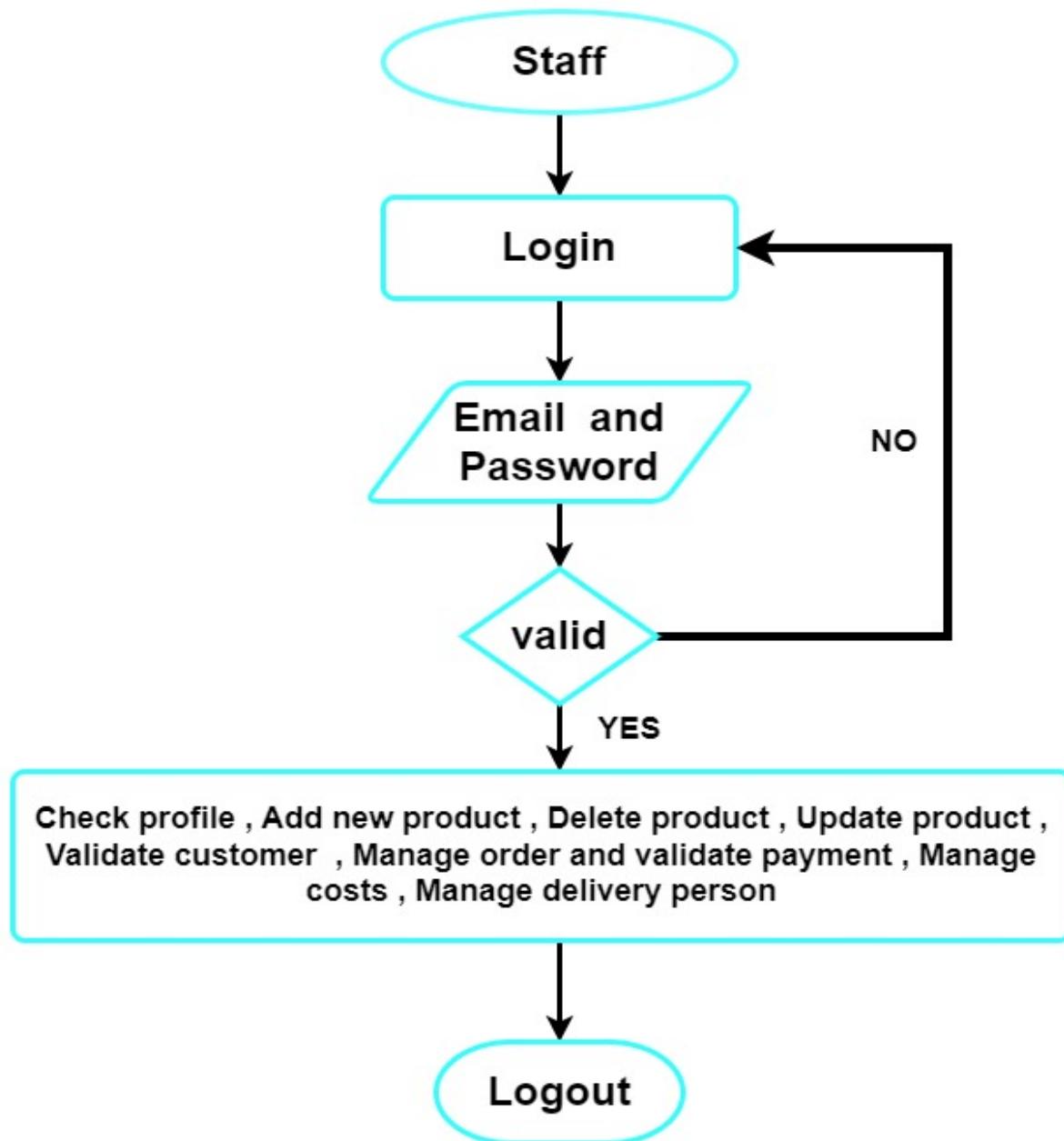


Figure 4.5: Flow diagram of Staff Panel

Workflow diagram for Delivery Person in 4.6

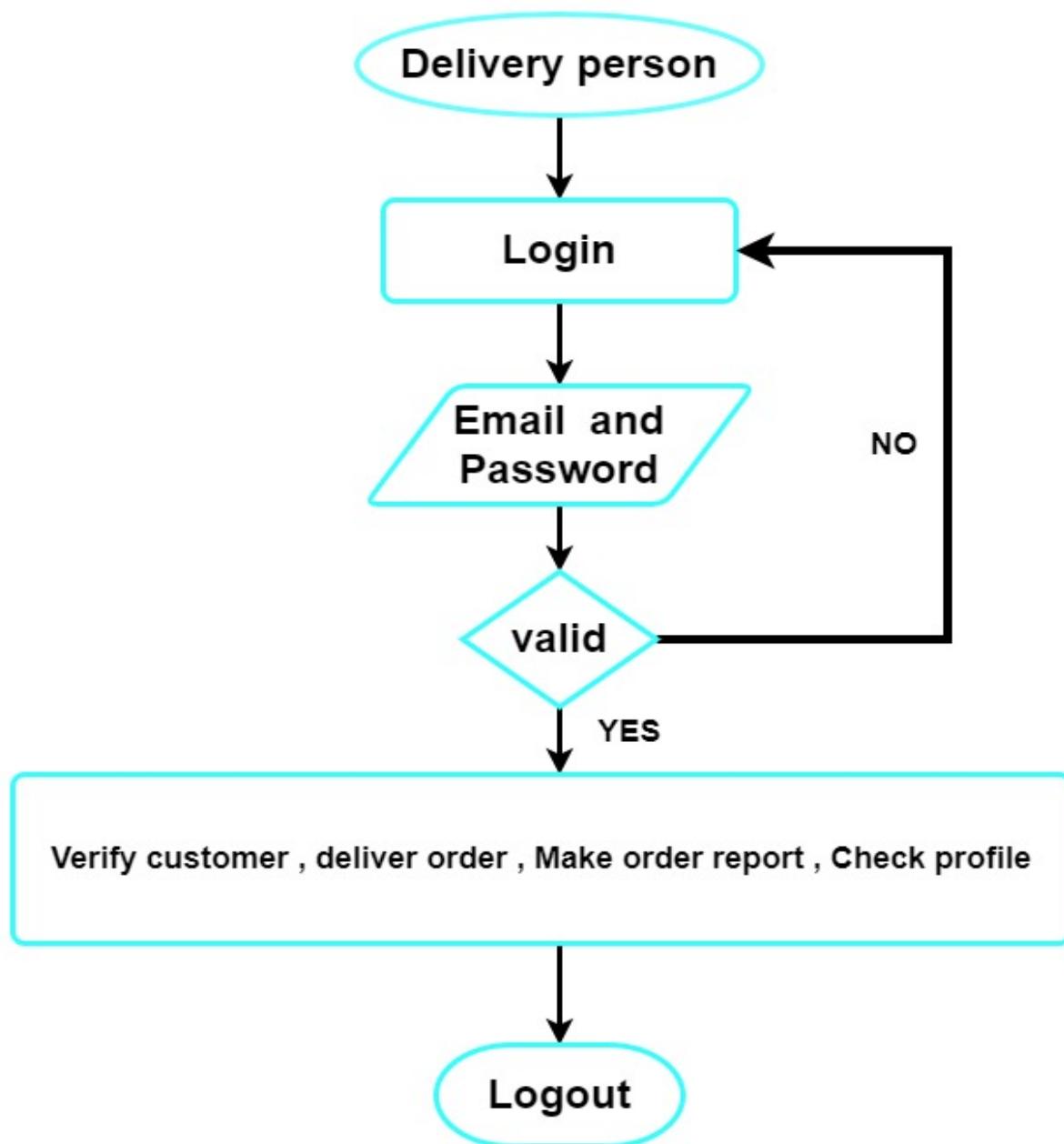


Figure 4.6: Flow diagram of Deliveryman Panel

Workflow diagram of Customer Order process in 4.7

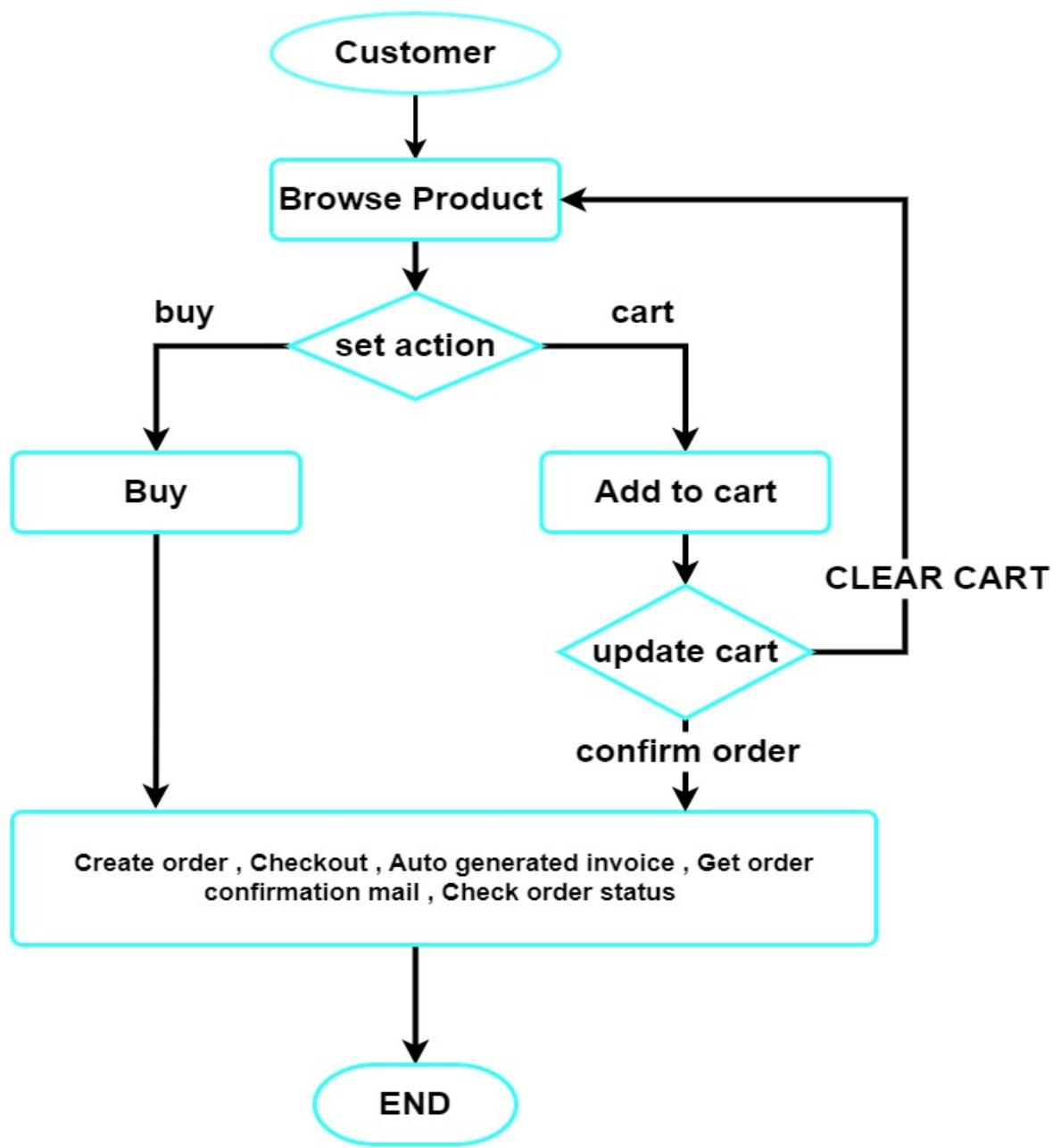


Figure 4.7: Flow diagram of Customer Panel

4.12 Activity diagram of Shop Management system

Activity diagram for Admin

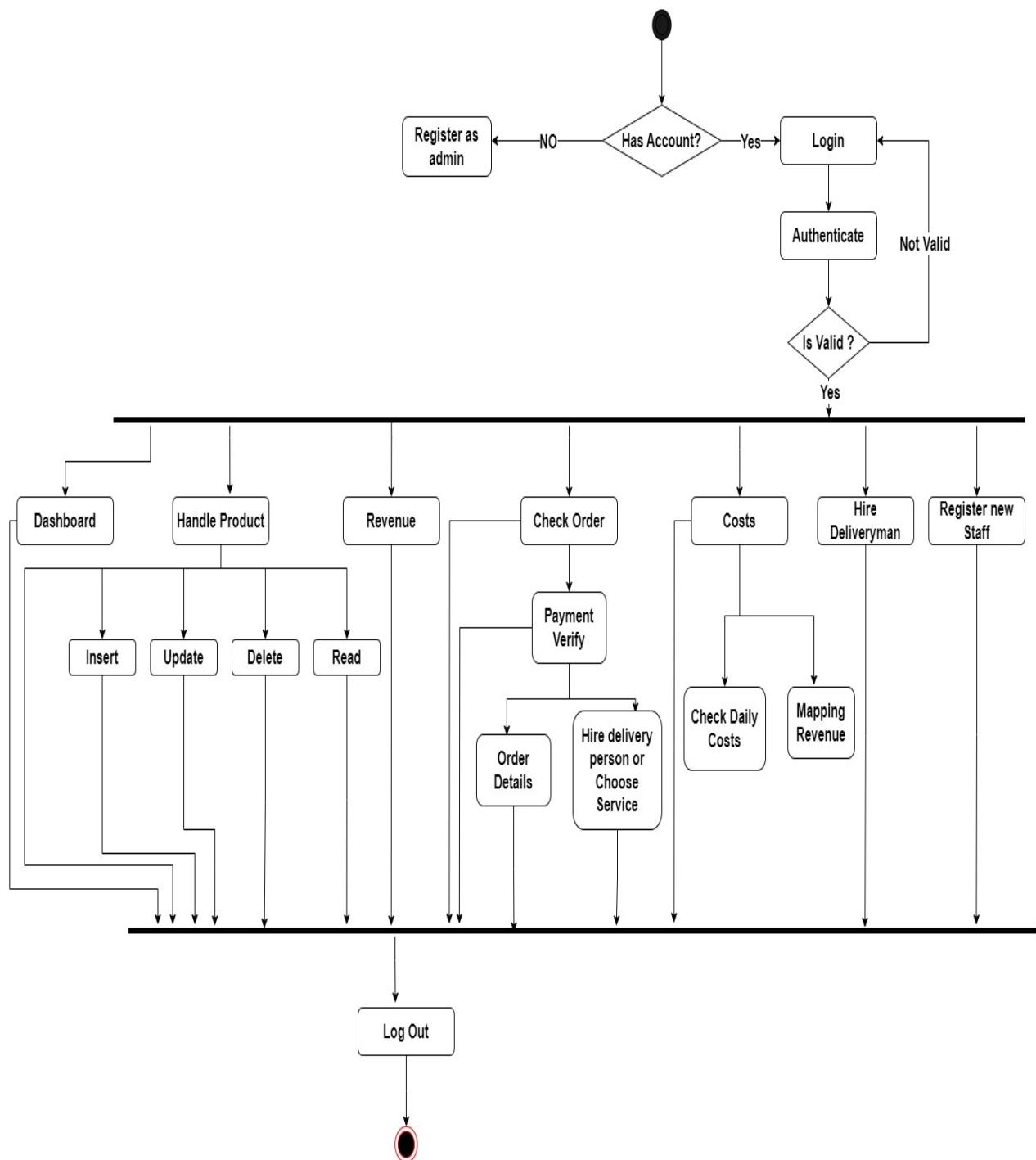


Figure 4.8: Activity diagram of Admin Panel

Activity diagram for Customer Order Process

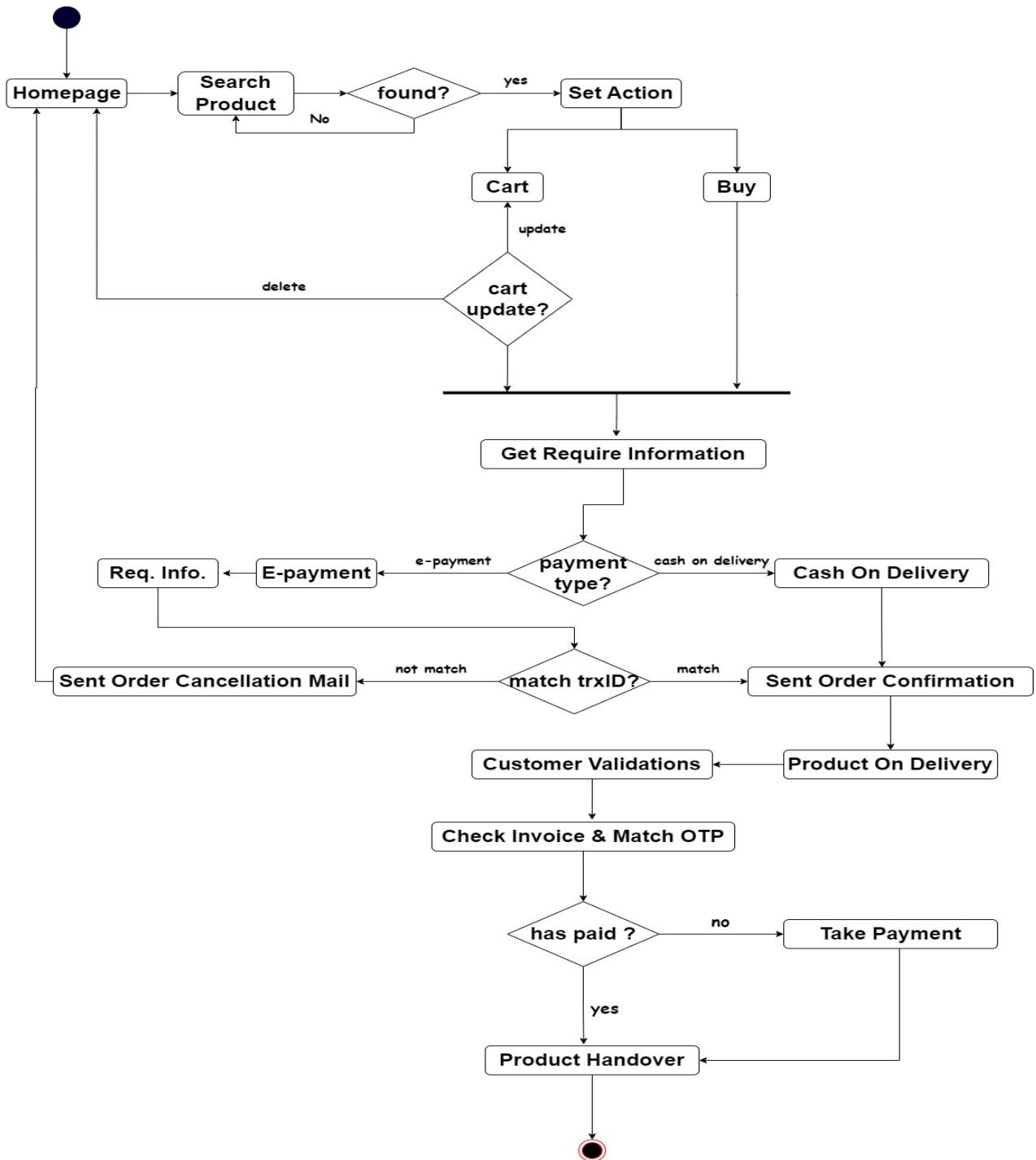


Figure 4.9: Activity diagram of Customer Order Process

Activity diagram for Deliveryman

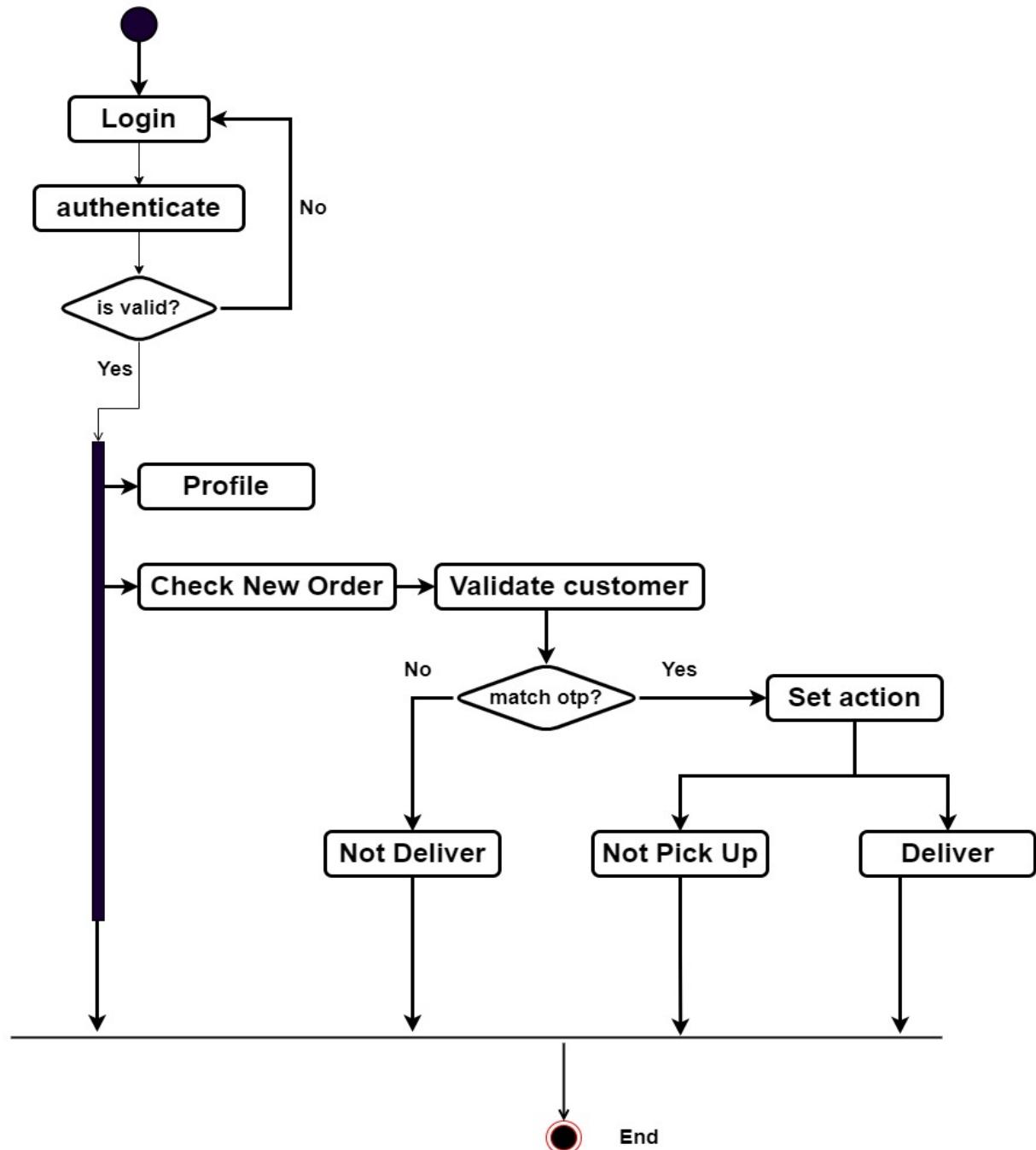


Figure 4.10: Activity diagram of Deliveryman

Chapter 5

Design specifications

5.1 Front-End-Design

- User-Friendly Interface: Easy-to-use design for smooth interaction.
- Clean and Modern Design: Attractive and up-to-date visuals.
- Interactive Elements: Effortless actions and data input.
- Focus on Usability: Simplifying shop management for convenience.

5.2 Back-End-Design

Our **Shop Management System** is built on the Express framework, a fast and minimalist web framework for Node.js. It leverages various packages and models to manage different aspects of the shop, including products, staff, customers, and delivery.

The dependencies used in our backend design include:

- **express**: Fast, unopinionated, and minimalist web framework for Node.js.
- **productmodel**: Model for managing product data.
- **staffmodel**: Model for managing staff data.
- **customermodel**: Model for managing customer data.
- **deliverymodel**: Model for managing delivery man data.

Our Express app serves as the main entry point for the backend, handling routes, middleware, and server configuration. We utilize the following middleware:

- **cors**: Middleware for enabling Cross-Origin Resource Sharing.
- **bodyParser**: Middleware for parsing HTTP request bodies.

The backend runs on port 3001, and the server listens for incoming requests on this port.

5.3 Implementation Requirements

The successful implementation of the **Shop Management System** requires the following:

5.3.1 Hardware Requirements

- A computer with a minimum of 4GB RAM.
- Sufficient storage capacity to accommodate the software and database.
- A stable internet connection for accessing the system.

5.3.2 Software Requirements

- Web Browser: Google Chrome, Mozilla Firefox, or Microsoft Edge.
- Node.js: Version 12 or above.
- Express.js: Version 4 or above.

5.3.3 Development Tools

- Integrated Development Environment (IDE): Visual Studio Code.
- Package Manager: npm (Node Package Manager).

5.3.4 Third-Party Libraries and Frameworks

- Express.js: Fast and minimalist web framework for Node.js.
- body-parser: Middleware for parsing HTTP request bodies.
- cors: Middleware for enabling Cross-Origin Resource Sharing.

It is essential to fulfill these requirements to guarantee the smooth implementation and functionality of the **Shop Management System**.

5.4 Summary

The implementation requirements for the **Shop Management System** encompass several key components. A stable internet connection is essential for seamless communication between the software and the users.

Chapter 6

Implementation & Testing

6.1 Implementation of database

The database implementation for the Shop Management System project involves defining and creating several database tables using Sequelize, a popular Object-Relational Mapping (ORM) library for Node.js.

6.1.1 Customer Registration

The `customerdetail` table is used to store customer registration details. It has the following columns:

- `id` (Primary Key): String type, uniquely identifies each customer.
- `name`: String type, stores the customer's name.
- `email`: String type, stores the customer's email.
- `mobile`: String type, stores the customer's mobile number.
- `alternative_mobile`: String type, stores an alternative contact number for the customer.
- `street`: String type, stores the customer's street address.
- `city`: String type, stores the customer's city.

6.1.2 Customer Orders

The `customerorders` table is used to store customer order details. It has the following columns:

- `id` (Primary Key): String type, uniquely identifies each order.
- `productid`: String type, stores the ID of the product ordered.
- `productname`: String type, stores the name of the product ordered.
- `price`: String type, stores the price of the product.
- `quantity`: String type, stores the quantity of the product ordered.
- `totalprice`: String type, stores the total price of the order.

6.1.3 Customer Payments

The `totalbill` table is used to store the total bill amount for each customer. It has the following columns:

- `id` (Primary Key): String type, uniquely identifies each bill.
- `customerid`: String type, stores the ID of the customer.
- `netpay`: String type, stores the net payment amount.

The `paymentinfo` table is used to store payment information for customer orders. It has the following columns:

- `id` (Primary Key): String type, uniquely identifies each payment.
- `customerid`: String type, stores the ID of the customer.
- `netpay`: String type, stores the net payment amount.
- `paymenttype`: String type, stores the type of payment (e.g., cash, credit card).
- `bakshnumber`: String type, stores the Baksh number for the payment.
- `trxid`: String type, stores the transaction ID for the payment.
- `status`: String type, stores the status of the payment.

6.1.4 Order State

The `orderstate` table is used to store the order state (status) for each customer order. It has the following columns:

- `id` (Primary Key): String type, uniquely identifies each order state.
- `customerid`: String type, stores the ID of the customer.
- `status`: String type, stores the status of the order.

6.1.5 Cart Product

The `cartproduct` table is used to store products added to the cart by customers. It has the following columns:

- `bill_id` (Primary Key): String type, uniquely identifies each cart product.
- `customerid`: String type, stores the ID of the customer.
- `productid`: String type, stores the ID of the product.
- `productname`: String type, stores the name of the product.
- `price`: String type, stores the price of the product.
- `quantity`: String type, stores the quantity of the product.
- `totalprice`: String type, stores the total price of the product in the cart.
- `invoice_id`: String type, stores the ID of the invoice associated with the cart product.

6.1.6 Cart Payment Info

The `cart_payment` table is used to store payment information for cart products. It has the following columns:

- `invoice_id` (Primary Key): String type, uniquely identifies each cart payment.
- `bill_id`: String type, uniquely identifies the associated cart product.
- `customerid`: String type, stores the ID of the customer.
- `netpay`: String type, stores the net payment amount.
- `paymenttype`: String type, stores the type of payment (e.g., cash, credit card).
- `bakshnumber`: String type, stores the Baksh number for the payment.
- `trxid`: String type, stores the transaction ID for the payment.
- `status`: String type, stores the status of the payment.

6.1.7 Cart Invoice

The `cartinvoice` table is used to store invoices for cart products. It has the following columns:

- `bill_id` (Primary Key): String type, uniquely identifies each cart invoice.
- `invoice_id`: String type, stores the ID of the invoice.
- `customerid`: String type, stores the ID of the customer.
- `status`: String type, stores the status of the invoice.

6.1.8 Order Confirmation

The `orderconfirmation` table is used to store order confirmations with OTP (One-Time Password). It has the following columns:

- `orderid` (Primary Key): String type, uniquely identifies each order confirmation.
- `otp`: String type, stores the One-Time Password for the order confirmation.

To create the above-mentioned tables, the `sequelize.sync()` function is called with the `alter:true` option, which ensures that the tables are synchronized with the database and any necessary changes are applied

6.1.9 Implementation of deliveryman database

- `Deliverystafflogin (deliverystafflogin):`
 - `id` (Primary Key): String type, uniquely identifies the delivery staff login.
 - `email`: String type, stores the email of the delivery staff.
 - `password`: String type, stores the password of the delivery staff.

-
- **Schedule (schedule):**
 - **staffId** (Primary Key): String type, uniquely identifies the schedule.
 - **startsTime**: Date type, stores the start time of the schedule.
 - **endTime**: Date type, stores the end time of the schedule.
 - **Weekday (weekday):**
 - **staffid** (Primary Key): String type, uniquely identifies the weekday.
 - **day1**: String type, stores the name of the first day of the week.
 - **day2**: String type, stores the name of the second day of the week.
 - **day3**: String type, stores the name of the third day of the week.
 - **LoginState (loginState):**
 - **id** (Primary Key): String type, uniquely identifies the login state.
 - **state**: Enum type (active, scheduled), stores the state of the login (default value: scheduled).
 - **DeliverymanPaymentInfo (deliverymanpaymentinfo):**
 - **id** (Primary Key): String type, uniquely identifies the payment info.
 - **paymeenttype**: String type, stores the type of payment.
 - **bakshnumber**: String type, stores the Baksh number for the payment.
 - **netpay**: String type, stores the net payment amount.
 - **status**: String type, stores the status of the payment.
 - **DeliveryReport (deliveryreport):**
 - **id** (Primary Key): String type, uniquely identifies the delivery report.
 - **otp**: String type, stores the OTP (One-Time Password) for the report.
 - **reporttype**: String type, stores the type of the report.
 - **description**: String type, stores the description of the report.
 - **Jobs (jobs):**
 - **id** (Primary Key): String type, uniquely identifies the job.
 - **deliverymanid**: String type, stores the ID of the deliveryman.
 - **order_id**: String type, stores the ID of the order.
 - **day**: String type, stores the day of the job.
 - **date**: String type, stores the date of the job.
 - **time**: String type, stores the time of the job.
 - **type**: String type, stores the type of the job.
 - **Ondelivery (ondelivery):**

-
- **id** (Primary Key): String type, uniquely identifies the on delivery status.
 - **deliverymanid**: String type, stores the ID of the deliveryman.
 - **status**: String type, stores the status of the delivery (default value: true).

6.1.10 Implementation of Products database

- **Product**:

- **id** (Primary Key): STRING type, uniquely identifies the product.
- **catagory**: STRING type, not nullable, stores the category of the product.
- **productName**: STRING type, not nullable, stores the name of the product.
- **productDetail**: STRING type, not nullable, stores the details of the product.
- **size**: STRING type, not nullable, stores the size of the product.
- **color**: STRING type, not nullable, stores the color of the product.
- **price**: INTEGER type, not nullable, stores the price of the product.
- **quantity**: INTEGER type, not nullable, stores the quantity of the product.
- **photo**: STRING type, nullable, stores the photo of the product.

- **catagory**:

- **id** (Primary Key): INTEGER type with auto-increment, uniquely identifies the category.
- **catagory**: STRING type, not nullable, stores the category name.

6.1.11 Implementation of Satff's database

- **Login**:

- **staffid** (Primary Key): STRING type, uniquely identifies the staff login.
- **email**: STRING type, not nullable, stores the email of the staff.
- **password**: STRING type, not nullable, stores the password of the staff.

- **costtype**:

- **id** (Primary Key): INTEGER type with auto-increment, uniquely identifies the cost type.
- **costs**: STRING type, not nullable, uniquely identifies the cost.

- **costs**:

- **costid** (Primary Key): STRING type, uniquely identifies the cost.
- **type**: STRING type, not nullable, stores the type of cost.
- **details**: STRING type, nullable, stores additional details of the cost.
- **amount**: INTEGER type, not nullable, stores the amount of the cost.

6.2 Implementation of Front-end Design

Homepage of shop management system in 6.1

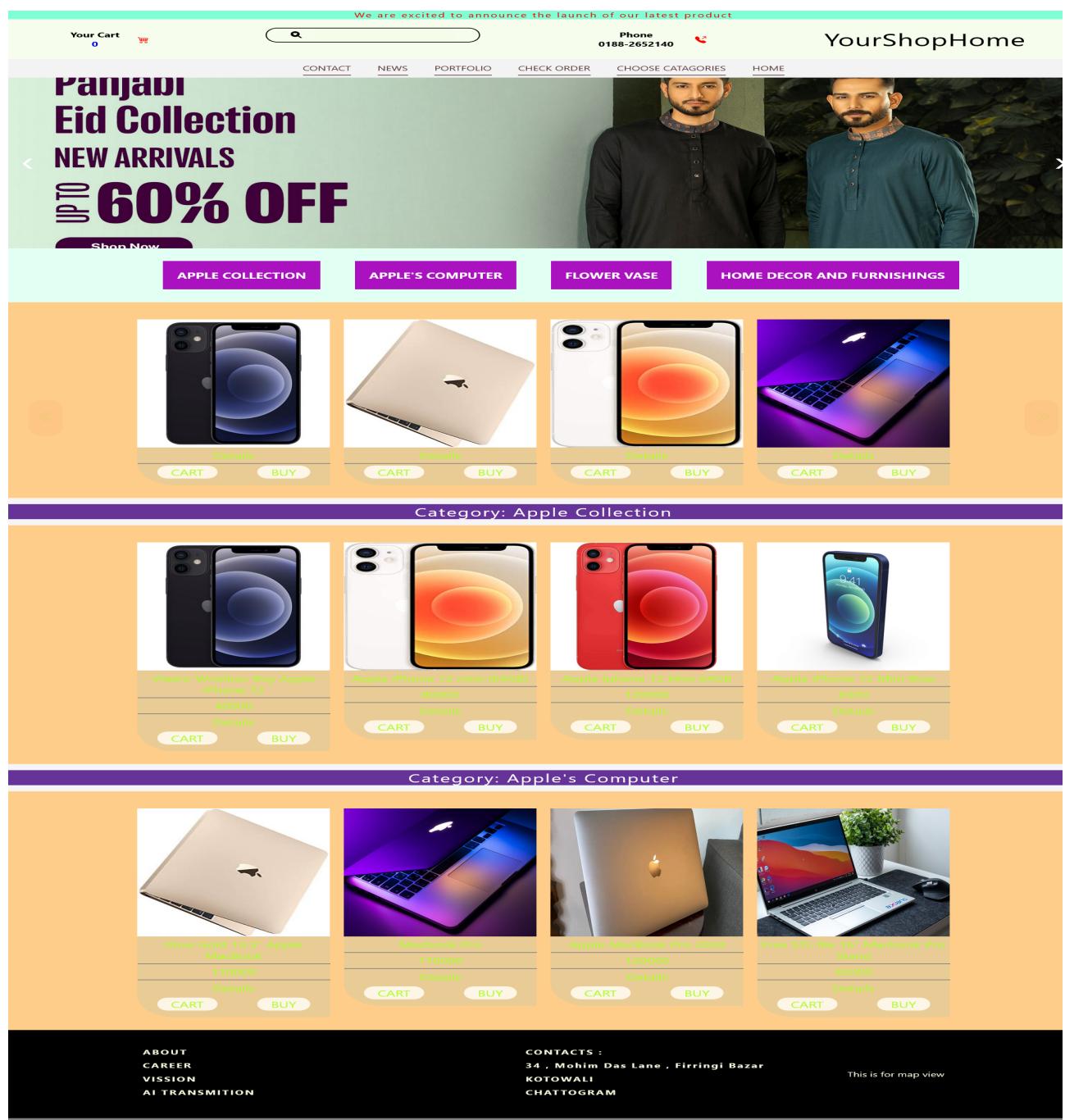


Figure 6.1: Shop Management Software System Homepage

Design Specification of homepage in 6.2

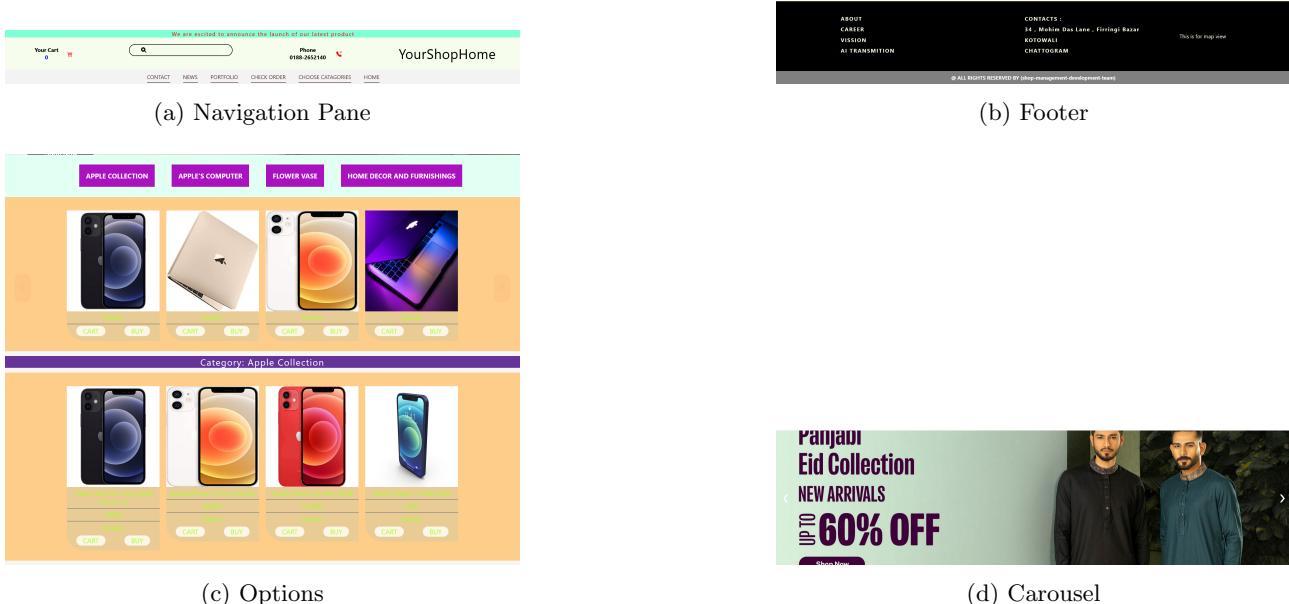


Figure 6.2: Section of Shop management homepage

Product details section and Product buy form of shop management system in 6.3a and 6.3b



(a) Shop Management Software System Homepage

The screenshot shows a "Product Buy Form" interface. On the left, there are six input fields: "Write your Name", "E-mail address", "mobile number", "Alternate Number", "Street Address", and "City". Below these is a large blue "Submit" button. On the right, there is a table with columns: "Product Name", "Price per Unit", "Quantity", and "Total Price". The table contains two rows: one for "Viaero Wireless: Buy Apple iPhone 12" with a price of 40000, quantity of 1, and total price of 40000; and another row for "Net Pay" with a value of 40000.

Product Name	Price per Unit	Quantity	Total Price
Viaero Wireless: Buy Apple iPhone 12	40000	1	40000
Net Pay			40000

(b) Product Buy Form

Figure 6.3: Product Details

Submit payment details and create invoice of shop management system in 6.4a , 6.4b and 6.4c

INVOICE FORM EE79D208

B-kash

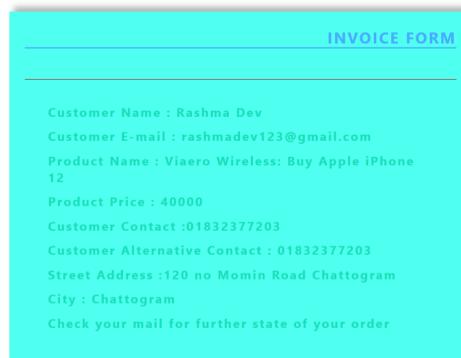
40000

01832377203

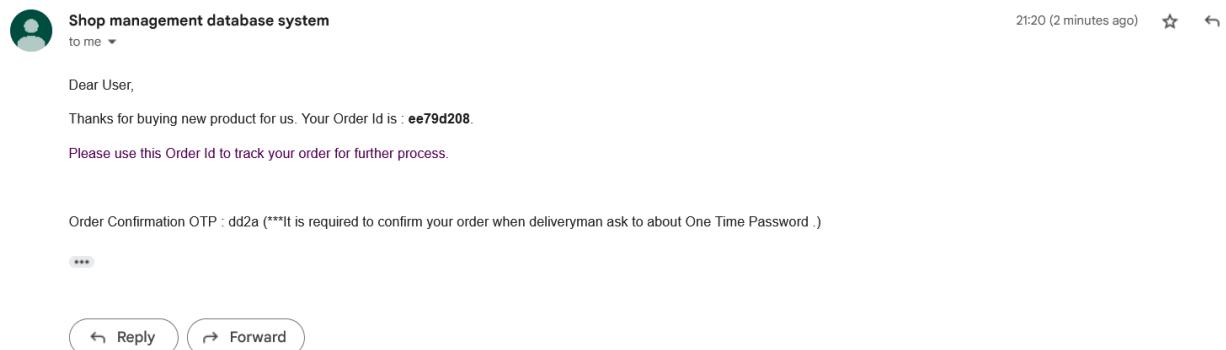
qwer345

Submit

(a) Submit Payment Details



(b) Auto generated invoice



(c) Get order confirmation mail

Figure 6.4: Order Create Process

Create order form cart using shop management system in 6.5

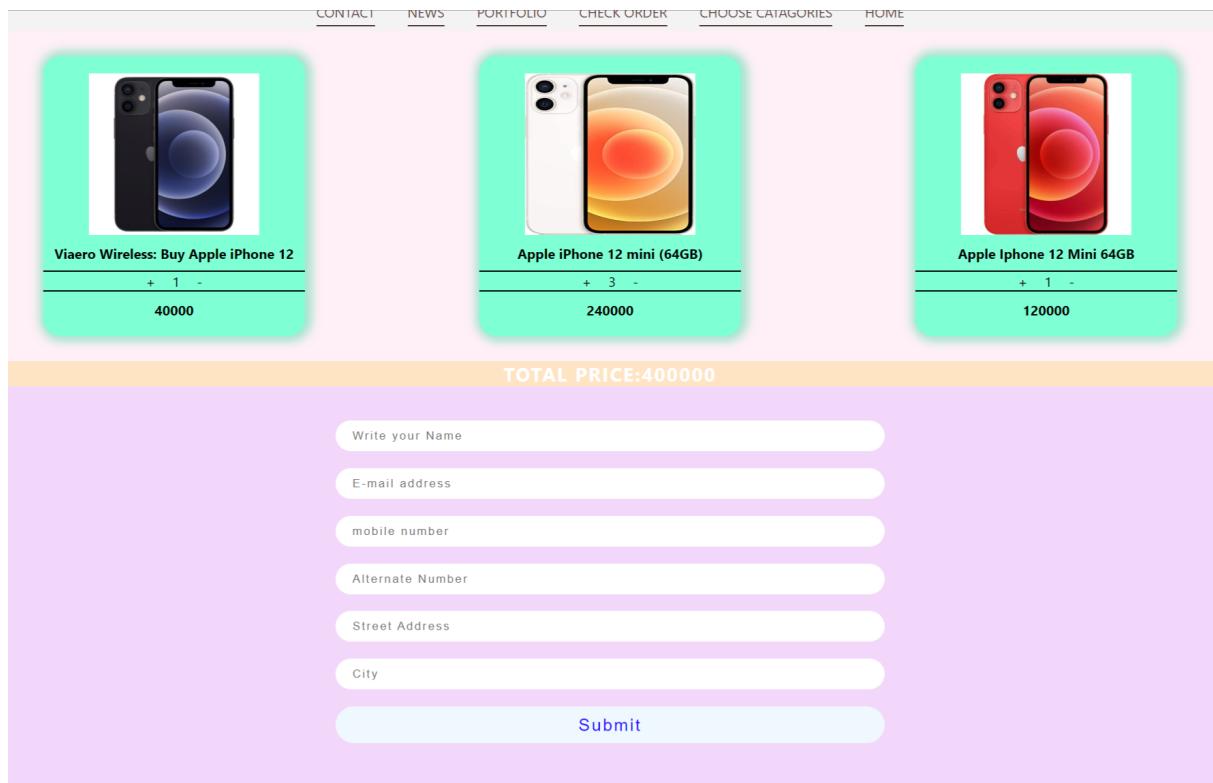


Figure 6.5: Create Cart Order

A dropdown menu for delivery method is open, showing "Cash-On Delivery".

Total bill to pay : undefined

write your email

Submit

Figure 6.6: Cash on delivery method

Create order form cart using shop management system in 6.7

Billing Details																																
# Invoice No : 2de4f224		# Customer Registration No :6d313bd3																														
Bills type : Online Shopping (status : pending)																																
YOUR SHOP HOME 24 Mohim Das Lane, Firingibazar, Kotowali , Chattogram Contact : 0188-2652140 E-mail : shophome98@gmail.com																																
Authorised By : Manager : Rashma Dev Branch : Firingeebazar Branch Contact : 0173-8053924 Email rashmadev@gmail.com																																
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; padding: 5px;">PRODUCT NAME</th> <th style="text-align: left; padding: 5px;">QUANTITY</th> <th style="text-align: left; padding: 5px;">PRICE</th> <th style="text-align: left; padding: 5px;">TOTAT PRICE</th> </tr> </thead> <tbody> <tr> <td style="padding: 5px;">Apple iPhone 12 mini (64GB)</td> <td style="padding: 5px; text-align: center;">1</td> <td style="padding: 5px; text-align: center;">80000</td> <td style="padding: 5px; text-align: center;">80000</td> </tr> <tr> <td style="padding: 5px;">Macbook Pro</td> <td style="padding: 5px; text-align: center;">1</td> <td style="padding: 5px; text-align: center;">170000</td> <td style="padding: 5px; text-align: center;">170000</td> </tr> <tr> <td style="padding: 5px;">Viaero Wireless: Buy Apple iPhone 12</td> <td style="padding: 5px; text-align: center;">1</td> <td style="padding: 5px; text-align: center;">40000</td> <td style="padding: 5px; text-align: center;">40000</td> </tr> <tr> <td style="padding: 5px;">Rose Gold 13.3" Apple MacBook</td> <td style="padding: 5px; text-align: center;">1</td> <td style="padding: 5px; text-align: center;">110000</td> <td style="padding: 5px; text-align: center;">110000</td> </tr> <tr> <td style="padding: 5px;">Apple Iphone 12 Mini 64GB</td> <td style="padding: 5px; text-align: center;">1</td> <td style="padding: 5px; text-align: center;">120000</td> <td style="padding: 5px; text-align: center;">120000</td> </tr> <tr> <td colspan="3" style="padding: 5px; text-align: right;">Total Pay :</td> <td style="padding: 5px; text-align: center;">520000</td> </tr> </tbody> </table>					PRODUCT NAME	QUANTITY	PRICE	TOTAT PRICE	Apple iPhone 12 mini (64GB)	1	80000	80000	Macbook Pro	1	170000	170000	Viaero Wireless: Buy Apple iPhone 12	1	40000	40000	Rose Gold 13.3" Apple MacBook	1	110000	110000	Apple Iphone 12 Mini 64GB	1	120000	120000	Total Pay :			520000
PRODUCT NAME	QUANTITY	PRICE	TOTAT PRICE																													
Apple iPhone 12 mini (64GB)	1	80000	80000																													
Macbook Pro	1	170000	170000																													
Viaero Wireless: Buy Apple iPhone 12	1	40000	40000																													
Rose Gold 13.3" Apple MacBook	1	110000	110000																													
Apple Iphone 12 Mini 64GB	1	120000	120000																													
Total Pay :			520000																													
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%; padding: 5px;">Name</td> <td style="width: 50%; padding: 5px;">Rashma Dev</td> </tr> <tr> <td style="padding: 5px;">E-mail</td> <td style="padding: 5px;">rashmadev123@gmail.com</td> </tr> <tr> <td style="padding: 5px;">Mobile</td> <td style="padding: 5px;">01832377203</td> </tr> <tr> <td style="padding: 5px;">Alternate Mobile No</td> <td style="padding: 5px;">01832377203</td> </tr> <tr> <td colspan="2" style="padding: 5px;">Address Info</td> </tr> <tr> <td style="padding: 5px;">Street</td> <td style="padding: 5px;">120 no Momin Road Chattogram</td> </tr> <tr> <td style="padding: 5px;">City</td> <td style="padding: 5px;">Chattogram</td> </tr> </table>					Name	Rashma Dev	E-mail	rashmadev123@gmail.com	Mobile	01832377203	Alternate Mobile No	01832377203	Address Info		Street	120 no Momin Road Chattogram	City	Chattogram														
Name	Rashma Dev																															
E-mail	rashmadev123@gmail.com																															
Mobile	01832377203																															
Alternate Mobile No	01832377203																															
Address Info																																
Street	120 no Momin Road Chattogram																															
City	Chattogram																															
<small>This is a computer generated invoice. No signature is required. If you have any query, Please call our Credit & Customer Management Department, Number 09619265231, Ext.: 113 for details.</small>																																
Download Panal																																

Figure 6.7: Auto generated Cart Invoice

Create order form cart using shop management system in 6.8

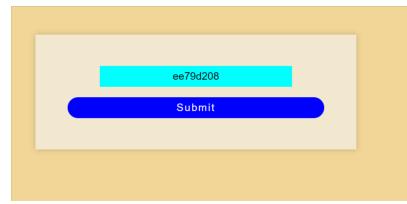


Figure 6.8: Check order state by searching order ID

Product filtering shop management system in 6.9

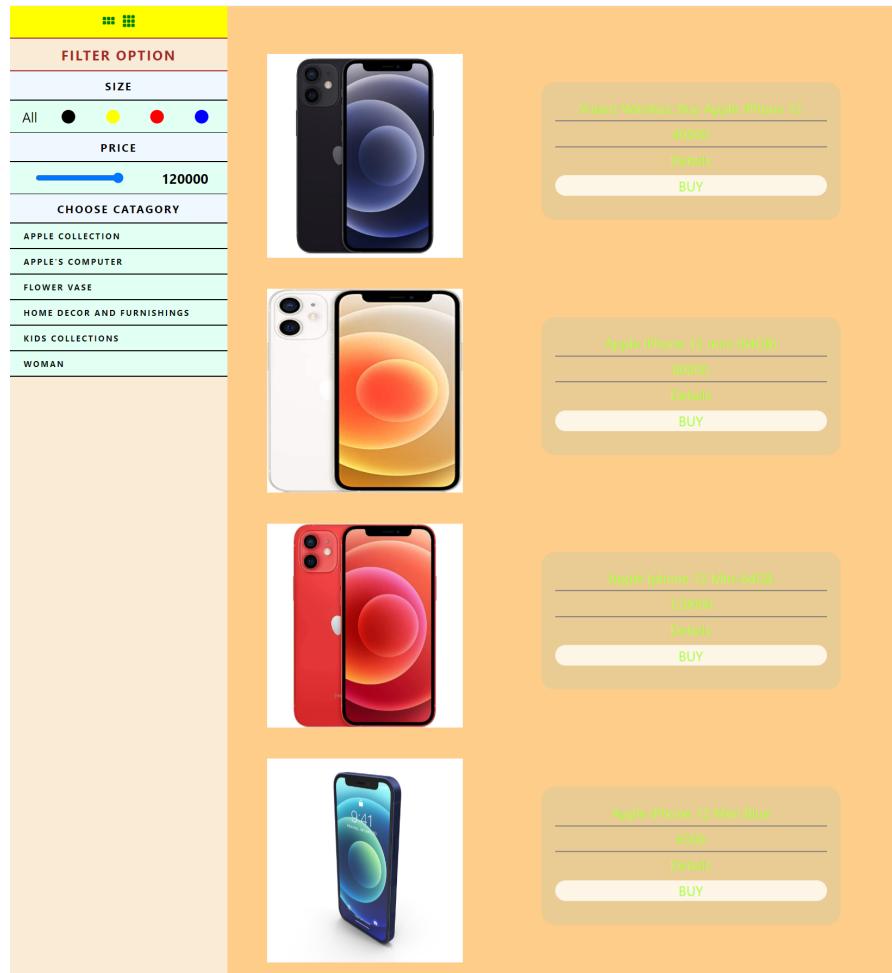


Figure 6.9: Product Filtering

New Staff Registration in Shop Management System (Figure 6.10)

CONTACT NEWS PORTFOLIO CHECK ORDER CHOOSE CATEGORIES HOME

CHOOSE POSITION

Write Employee Name

What's your father name ?

What's your mother's name ?

What is your Nationality ?

dd/mm/yyyy

Choose File No file chosen

PARMAMENT ADDRESS

STREET

HOUSE or HOME TOWN

CITY

POST OFFICE

POST CODE

PRESENT ADDRESS

STREET

HOUSE or HOME TOWN

CITY

POST OFFICE

POST CODE

CONTACT

mobile number

Alternate Number

OFFICE NUMBER

E-mail address

OTHERS

dd/mm/yyyy

Renumearction

Submit

Figure 6.10: Staff Registration Form

Staff Login in Shop Management System (Figure 6.11)



The image shows a staff login form. It features a light blue header with the text "CHOOSE POSITION" in a dropdown menu. Below the header are two input fields: "E-mail address" and "Password". At the bottom is a green "Submit" button.

Figure 6.11: Staff Login Form



The dashboard has a top navigation bar with links: Dashboard, Wages, Check Orders, Costs, Products, Delivery, and Logout. Below the navigation, there are sections for "Staff Personal Information" and "Personal Information". A circular profile picture of a woman is displayed, with the caption "Position : Chief Executives" underneath. To the right of the profile picture, personal details are listed: Name : Reshma dev, Father Name : Badal Dev, Mother Name : Runu Dey, Date of Birth : 1997-09-10, and Nationality : Bangladeshi. The "Address Informations" section contains two tables: "Parmanent Address" and "Present Address", both listing the same address information: Street : 34 firringi Bazar, House : Chowdhury Villa 5th floor, City : Chittagong, Zilla : Chittagong, Post Office : GPO, and Post Code : 4000. The "Contact Informations" section lists three contact details: rashmadev123@gmail.com, 01738053924, and 01738053924.

Figure 6.12: Staff dashboard

Revenue implementation visually and manually in Shop Management System (Figure 6.13)



Figure 6.13: Visual and Manual Implementations of Revenue

Validate transaction and orders in Shop Management System (Figure 6.14)

FOR SINGLE PRODUCT

Order ID	Customer ID	Total Pay	Payment Type	Transaction No.	Trx ID	Status	Actions
#ee79d208	#5a6b31eb	40000	B-kash	01832377203	qwer345	PENDING DELETE ORDER	HIRE DELIVERY MAN
#7cea0b34	#8c2e3041	40000	B-kash	01832377203	12nmgjuyi	PENDING DELETE ORDER	HIRE DELIVERY MAN
#14606d25	#3052ff22	40000	B-kash	01832377203	12nmgjuyi	DELIVERED	AKKAS ALI
#c9d3560e	#bb08abab	40000	B-kash	01832377203	12nmgjuyi	DELIVERED	RAHAMAT UDDIN
#dbb32358	#221ce20b	40000	cashon-delivery	null	null	NOT DELIVERED	AKKAS ALI

FOR MULTI PRODUCT

Order ID	Customer ID	Total Pay	Payment Type	Transaction No.	Trx ID	Status	Actions
#4d476f65	#6d313bd3	520000	cashon-delivery	null	null	PENDING DELETE ORDER	HIRE DELIVERY MAN
#f1db9594	#d6d8dc47	665000	cashon-delivery	null	null	CUSTOMER DO NOT PICK	AKKAS ALI

Figure 6.14: Check orders

Hire deliveryman and select service in Shop Management System (Figure 6.15)

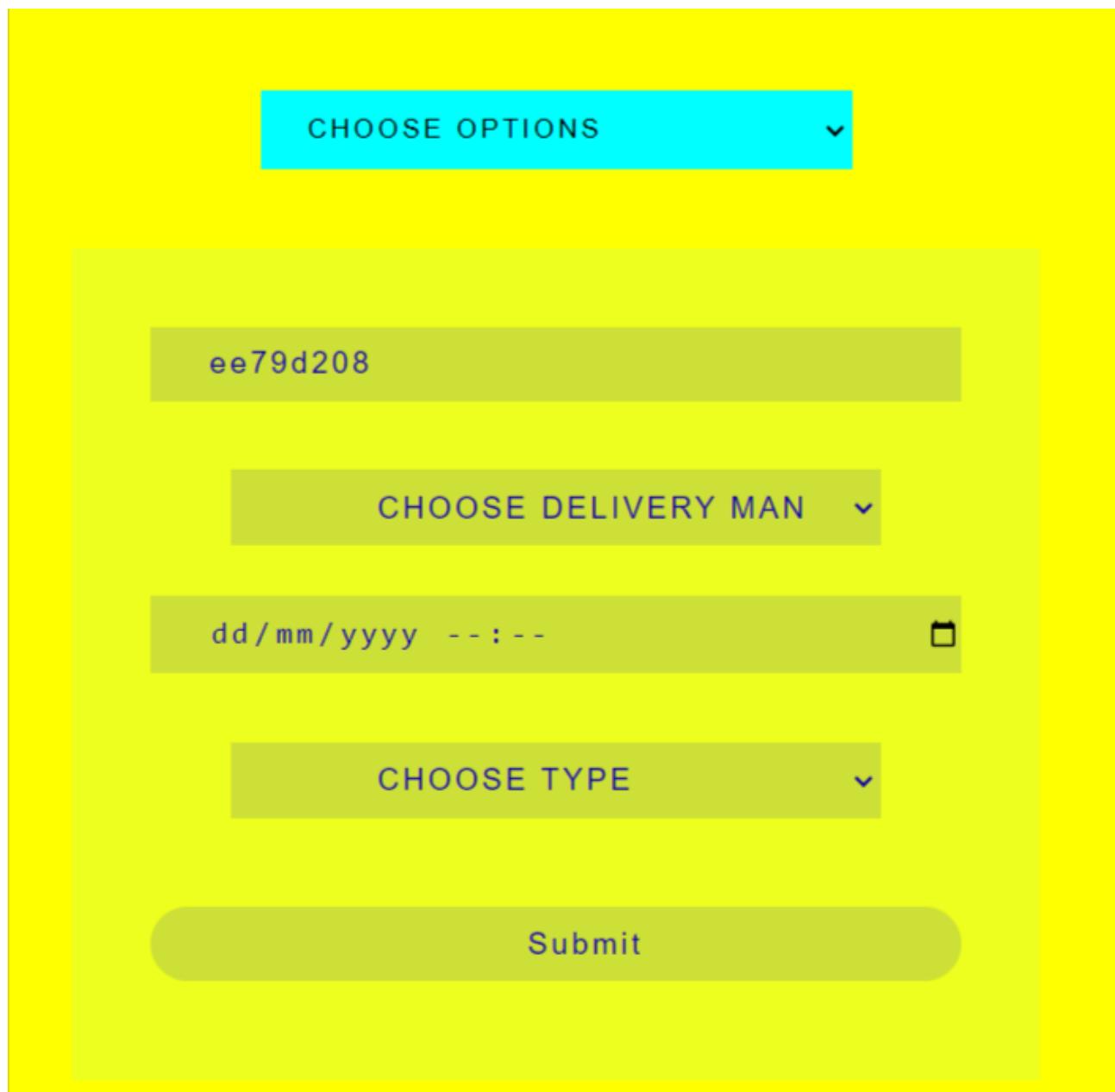


Figure 6.15: Hire Delivery man or selects services

Handled daily costs in Shop Management System (Figure 6.16)

The screenshot shows two main components: a cost entry form on the left and a daily costs report on the right.

Entry Daily Costs:

- Form fields include: "Select cost type" dropdown, "Write a short notes about the cost" text area, and "Amount" input field.
- A "Submit" button is at the bottom.

Daily Costs Report:

Daily Costs Report				
Cost ID	Cost Type	Cost Details	Amount	Update
2227	breakfast	200 taka	200	2023-05-29T06:47:30.000Z
59ad	Buying Product	Buy new cloth from R&D Traders	12000	2023-05-28T19:59:11.000Z
8468	Customer refund	Customer ID:#20302 Refund money 3000 taka due to unconditional payment failure	3000	2023-05-28T20:00:48.000Z

Figure 6.16: Costs specifications

Product CRUD operations in Shop Management System (Figure 6.17)

The screenshot shows a product list and search interface.

Product List:

ID	Name	Price	Update	Delete
Apple Collection	Viaero Wireless: Buy Apple iPhone 12	40000	UPDATE	DELETE
Apple's Computer	Rose Gold 13.3" Apple MacBook	110000	UPDATE	DELETE
Apple Collection	Apple iPhone 12 mini (64GB)	80000	UPDATE	DELETE
Apple's Computer	Macbook Pro	170000	UPDATE	DELETE
Apple Collection	Apple Iphone 12 Mini 64GB	120000	UPDATE	DELETE
Apple's Computer	Apple MacBook Pro 2020	120000	UPDATE	DELETE
Apple Collection	Apple iPhone 12 Mini Blue	6500	UPDATE	DELETE
Apple's Computer	Free STL file 16" Macbook Pro Stand	65000	UPDATE	DELETE

Search and Filter:

- Buttons: "List Of Product", "Insert New Product", "Insert New Catagory".
- Search bar: "SEARCH BY CATAGORY....".
- Category dropdown: "SELECT CATAGORY".
- Pagination: "«" and "»" icons.

Figure 6.17: CRUD operations on Products

Products Insert

Select Catagory

Product Name

Product Color

Product Size

Product Details

Product Price

No file selected.

Quantity

Submit

Figure 6.18: Insert New Products

Product Catagory Insert

Insert Catagory

Submit

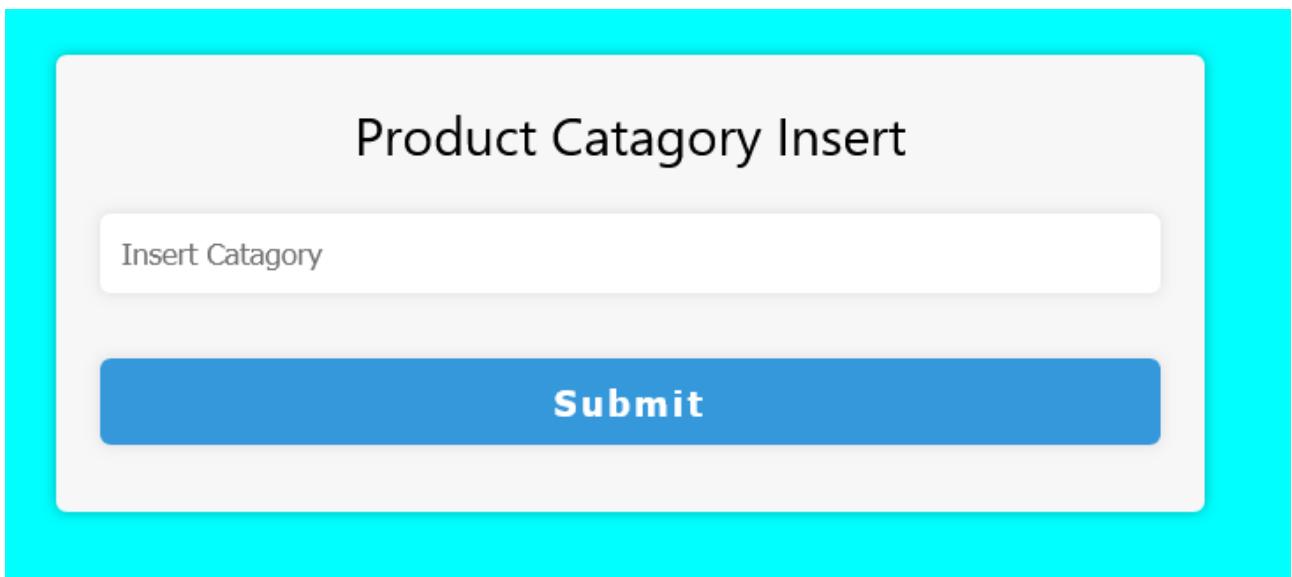


Figure 6.19: Insert New Products Category

Product Update

Select Catagory

Viaero Wireless: Buy Apple iPhone 12

Electronics encompass a wide range of products that have become essential in our daily lives. Smartphones are powerful devices that provide communication, internet access, and a multitude of features.

40000

Browse... No file selected.

1

Update

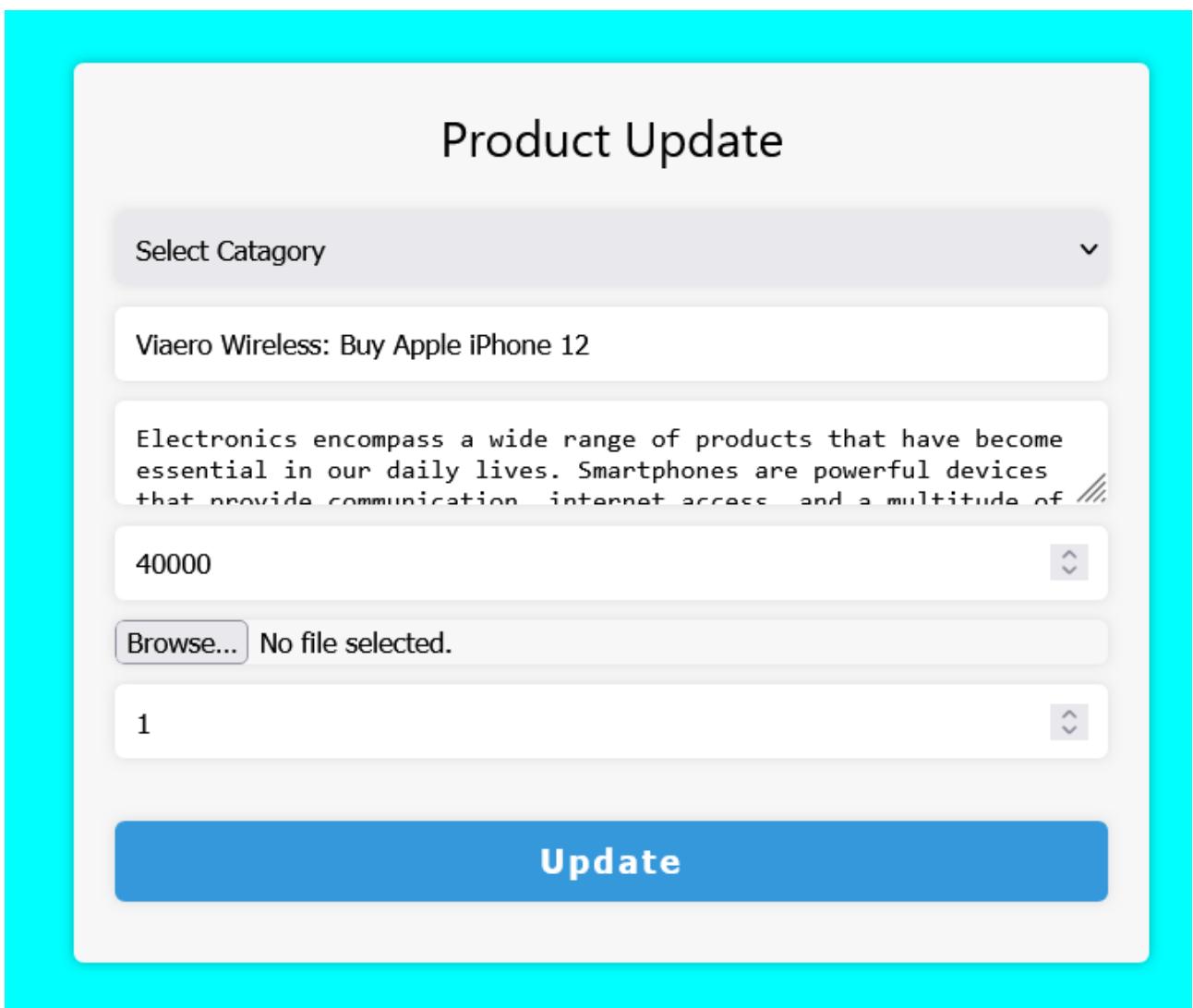


Figure 6.20: Update existing product

List Of Product	Insert New Product	Insert New Catagory
<input type="text" value="SEARCH BY CATAGORY...."/> SELECT CATAGORY		

Product List

Figure 6.21: Find products via search and category wise

Deliveryman	
Write Employee Name	
What's your father name ?	
What's your mother's name ?	
What is your Nationality ?	
dd / mm / yyyy	
<input type="button" value="Browse..."/> No file selected.	
PARMAMENT ADDRESS	
STREET	
HOUSE or HOME TOWN	
CITY	
POST OFFICE	
POST CODE	
PRESENT ADDRESS	
STREET	
HOUSE or HOME TOWN	
CITY	
POST OFFICE	
POST CODE	
CONTACT	
mobile number	
Alternate Number	
OFFICE NUMBER	
E-mail address	
OTHERS	
dd / mm / yyyy	
Renumeartion	
SET SCHEDULE	
<input type="button" value="CHOOSE DAY-1"/> ▼	
<input type="button" value="CHOOSE DAY-1"/> ▼	
<input type="button" value="CHOOSE DAY-1"/> ▼	
<input type="button" value="Submit"/>	

Figure 6.22: Hire new deliveryman form

List Of Delivery Persons	Add new delivery member	Active Delivery
--------------------------	-------------------------	-----------------

ID	Name	Email	Phone	Status
8ab895453c2eb847	Akkas Ali	rashmadev123@gmail.com	01832377203	Scheduled
8d31b01b17471df2	Rahamat Uddin	rashmadev123@gmail.com	01832377203	Scheduled

Figure 6.23: Active delivery persons

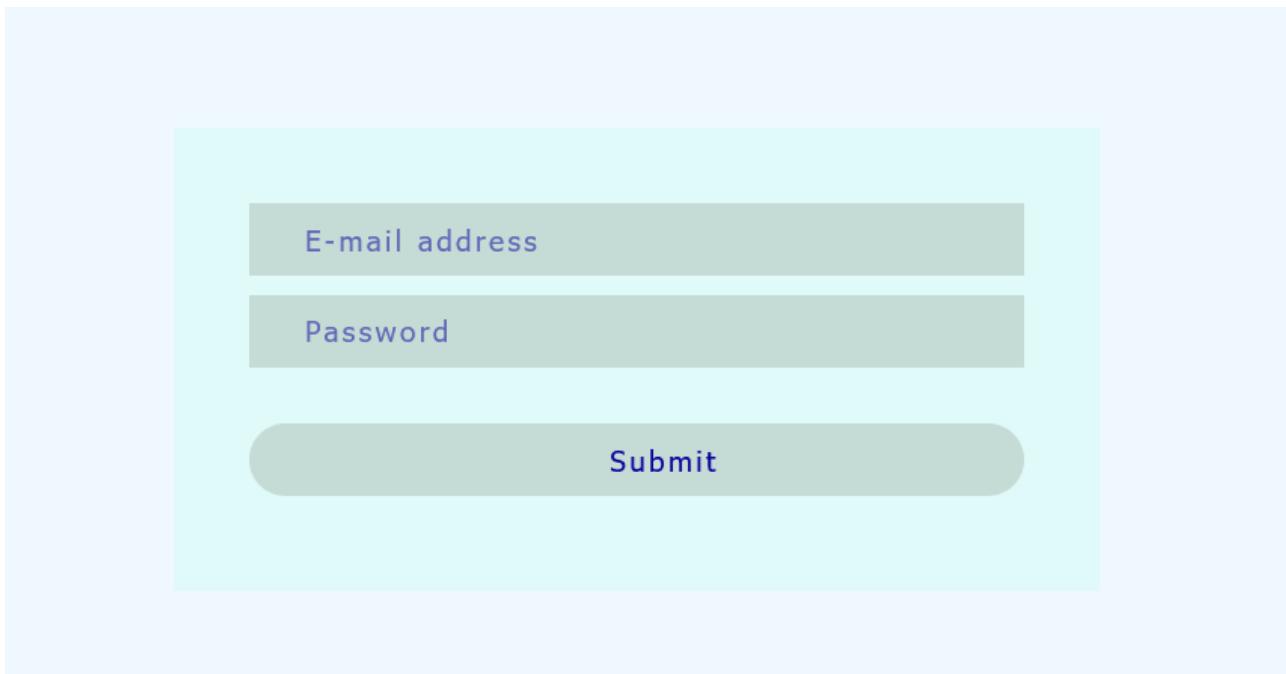


Figure 6.24: Delivery man Login

YOUR SHOP HOME
Search your customer , invoice , Deliver Product
Submit Query
Dashboard
Orders
Logout

Position : Delivery Man

Name : Akkas Ali
Father name : Kuddus Ali
Mother name : Jhankar Ali
Date of Birth : 1997-03-28T00:00:00.000Z
Nationality : Bangladeshi

ADDRESS INFORMATIONS

PARMAMENT ADDRESS	PRESENT ADDRESS
Street : 34 firringi Bazar	Street : 120 no Momin road
House : Chowdhury Villa 5th floor	House : Chowdhury Villa 5th floor
City : Chittagong	City : Chittagong
Zilla : Chittagong	Zilla : Chittagong
Post Office : GPO	Post Office : GPO
Post Code : 4000	Post Code : 4000

[ABOUT](#)
[CAREER](#)
[VISSION](#)
[AI TRANSMITION](#)

CONTACTS :

34 , Mohim Das Lane , Firringi Bazar
KOTOWALI
CHATTOGRAM

This is for map view

© ALL RIGHTS RESERVED BY (shop-management-development-team)

Figure 6.25: Delivery man Dashboard

NEW ORDERS ARRIVES...			
INVOICE ID	SCHEDULED	DETAILS	ACTIONS
DBB32358	05/31/2023_WEDNESDAY	CLICK HERE FOR MORE DETAILS..	SUBMIT

Figure 6.26: Check new orders

The interface features a light blue background with a green header bar at the top. Below the header, there is a green input field containing the placeholder text "Enter your order id..". Below the input field is a green button with the word "Submit" in white.

Figure 6.27: Enter the order id To Validate customers

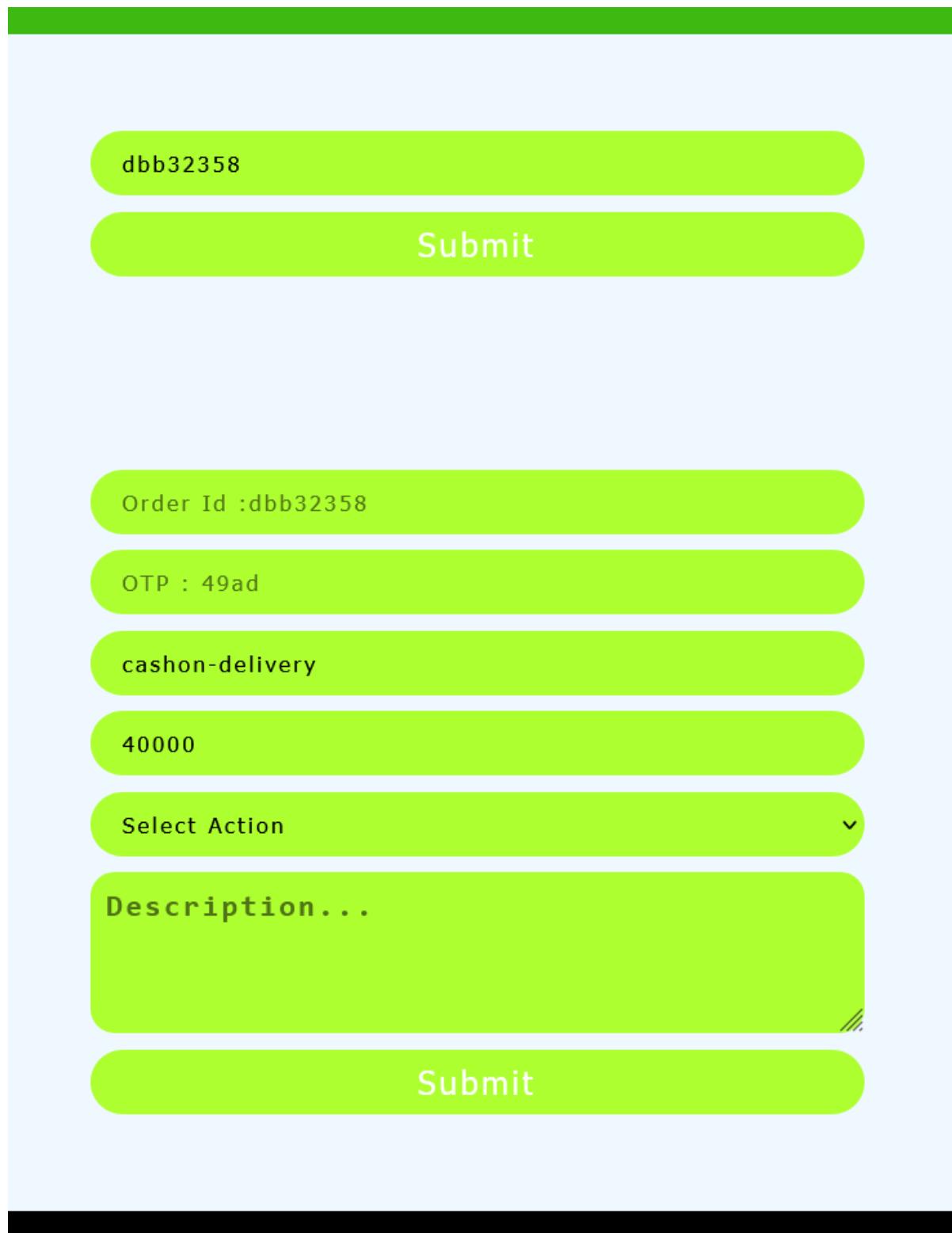


Figure 6.28: Handled orders to the customers

6.3 Test result and Report

Table 6.1: Test cases for User Interface

SI No	Test Case	Input	Expected Outcome	Actual Output	Result
1	Display web page	Tested On Browser	Display Successful	Web page displayed	Passed
2	Check navigation link	Clicking On nav Button	Linked Successful	Navigate on expected page	Passed
3	Cart Product	Clicking On Cart Button	Product listing on Cart	Product set in cart	Passed
4	Filter product by Color	Clicking on different color filter option	Showed product with existing color	Product filter with color	Passed
5	Filter product by Price	Change the slider of Price filter option	Showed product with existing Price	Product filter with price	Passed
6	Filter product with different categories	Clicking on different categories option	Showed product with existing category	Product filter with category	Passed

Table 6.2: Test cases for Product Buying Process

SI No	Test Case	Input	Expected Outcome	Actual Output	Result
1	Input field required	Blank Input	Failed Submission	submission failed with warning	Failed
2	Valid E-mail Format	Invalid e-mail format	Failed Submission	submission failed with warning	Failed
3	Generate invoice form	Submit Customer details	Generate Invoice ID on Invoice page	Shows generated invoice	Passed
4	Order Confirmation Mail	Check mail	Getting Order id and OTP	Gets confirmation mail	Passed
5	Check Order detail	Order ID	Getting Billing Invoice	Shows generated invoice	Passed

Table 6.3: Test cases for Staff Management Process

SI No	Test Case	Input	Expected Outcome	Actual Output	Result
1	Get registration mail	Check mail	Get password	Password get in mail	Passed
2	Blank or incorrect Password	Blank Password	Login failed	Login failed	Failed

Table 6.4: Test cases for Delivery Management Process

SI No	Test Case	Input	Expected Outcome	Actual Output	Result
1	Login credential	Invalid password	Failed Login	Failed Login	Failed
2	Check Order detail	Clicking on details button	Show invoice form	Open Invoice form	Passed
3	Order Confirmation	Submit Order Id	Match OTP with customer	open delivery form	Passed

6.4 Summary

The "Implementation and Testing" section of the project report for the "Shop Management Software System" focused on key aspects of system development. It involved the implementation of the database, including the design, creation, and population of relevant tables. Furthermore, the front-end design was developed to create an aesthetically pleasing and user-friendly interface. The interaction components, such as API endpoints and functionality, were also implemented. Thorough testing was conducted, and the results were documented in the test report. This section emphasizes the successful implementation of the system's core components and the rigorous testing process.

Chapter 7

Limitation , Future scope & Conclusion

7.1 Limitation

- Lack of a dedicated mobile app.
- Inadequate responsiveness for mobile device and notepad.
- Dependence on plain CSS for designing user interface.
- Absence of integration with payment gateways.
- Lacking of Advanced Filtering options like limited user interface and feature options.
- Potential database organization issues like data redundancy , data inconsistency and Lack of Normalization.
- Inadequate feature of delivery management system.

By addressing these limitations and taking proactive measures to mitigate them, businesses can ensure a smoother implementation and optimize the benefits of their shop management software system.

7.2 Future Scope

The future scope of our project holds several potential opportunities for expansion and enhancement. Here are some key areas of future development:

- Global hosting for worldwide accessibility.
- Integration of globally accepted payment gateways.
- Enhancing the design for improved feasibility, attractiveness, and responsiveness.
- Developing a mobile application with additional useful features.
- Added new features on Delivery Management.

By pursuing these future developments, our project can continue to evolve, meet the changing needs of the market, and stay competitive in the dynamic landscape of shop management software systems.

7.3 Conclusion

Despite the limitations mentioned earlier, our **Shop Management System** effectively addresses the challenges and requirements of efficient shop management. With its real-time sales management, inventory tools, customer handling, auto-generated payment system, and delivery management system, our software optimizes shop operations, enhances customer satisfaction, and drives business growth. We are confident that the successful implementation of this project will contribute to the long-term success and profitability of our shop, making it a valuable asset in the retail industry.

7.4 References

1. <https://www.wrike.com/project-management-guide/>
2. <https://www.edvantis.com/blog/software-development-methodologies/>
3. <https://www.geeksforgeeks.org/iterative-incremental-model-in-designing-system/>
4. <https://www.aarong.com/>
5. <https://www.daraz.com.bd/>
6. <https://www.ryanscomputers.com/>
7. <https://www.smarteshopbd.com/>