****

**Tribhuvan University**

**Faculty of Humanities and Social Sciences**

A PROJECT REPORT

ON

“ORDERING WEB APPLICATION”

Submitted in partial fulfillment of the requirement of

Project – I CACS256

Of

Bachelor of Information Technology

**Submitted By**

**Bikash Mainali 6-2-263-05-2020**

Sudip Shrestha 6-2-263-27-2020

**Janamaitri Multiple Campus**

**Kuleshwor - Height, Kathmandu**

December 10,2023

A PROJECT REPORT

ON

“ORDERING WEB APPLICATION”

Submitted in partial fulfillment of the requirement of

Project – I ()

Of

Bachelor of Information Technology

**Submitted to**

Tribhuvan University

Kathmandu, Nepal

**Submitted By**

**Bikash Mainali 6-2-263-05-2020**

Sudip Shrestha 6-2-263-27-2020

**Project Supervisor**

**Kamal Tamrakar**

HOD

**Janamaitri Multiple Campus**

**Kuleshwor - Height, Kathmandu**

December 10,2023



**Tribhuvan University**

**Faculty of Humanities and Social Sciences**

Janamaitri Multiple Campus

**SUPERVISOR’S RECOMMENDATION**

I hereby recommend that this project prepared under my supervision by*,* Sudip Shrestha *6-2-263-27-2020* and Bikash Mainali *6-2-263-04-2020* entitled “**ORDERING WEB APPLICATION**” in partial fulfillment of the requirements for the degree of Bachelor of Computer Application is recommended for the final evaluation.

**Signature**

Kamal Tamrakar

Department of Math and ICT

Janamaitri Multiple Campus

Kuleshwor-Height, Kathmandu



Tribhuvan University

**Faculty of Humanities and Social Sciences**

**JANAMAITRI MULTIPLE CAMPUS**

LETTER OF APPROVAL

This is to certify that this project prepared by Sudip Shrestha, 6-2-263-27-2020 and Bikash Mainali, 6-2-263-05-2020 entitled “**ORDERING WEB APPLICATION**” 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.

|  |  |
| --- | --- |
| **SUPERVISOR**  **Kamal Tamrakar**  Janamaitri Multiple Campus  Kuleshwor-Height, Kathmandu | **HOD/ Coordinator**  **Kamal Tamrakar**  Head of Math And ICT Department  Janamaitri Multiple Campus  Kuleshwor-Height, Kathmandu |
|  |  |

# ACKNOWLEDGEMENT

We are the students of BCA 4th semester of Janamaitri Multiple Campus in the BCA. We have to do minor project work in this semester for practical work so we can easily develop the web base application as its part as a project. We are so glad to present our project ***“Ordering web application”*** which is implementation of our study in real life with a practical knowledge.

We express special thanks to our project supervisor Kamal Tamrakar for his valuable and great support and guidance in all the happening regarding to the project. We are also express special thanks to BCA coordinator Kamal Pathak for his encouragement and guidance throughout the project.

We are indebted to the department of Information and Technology for providing support to add on out venture. At last we would like to express special thanks to our teacher and college member who have been directly and indirectly the part of this project and support us and views during the entire development time.

# ABSTRACT

This project “Ordering web application” is a web based application which primary objective is to create a user-friendly platform that seamlessly connects sellers and buyers. Sellers can register and log in to the application, gaining access to a dashboard where they can add, edit, and delete their products. The CRUD system ensures efficient management of the product catalog, allowing sellers to maintain accurate information and availability. On the other hand, buyers can explore the home page, which displays a list of available products. They can browse through the product catalog, view detailed information, and make purchasing decisions. The application facilitates the ordering process by providing a user-friendly interface that allows buyers to select desired products, specify quantities, and proceed with the order placement. The technologies used in this application contribute to its overall functionality and visual appeal. HTML provides the structure of the web pages, while jQuery enhance interactivity and dynamic content. Bootstrap ensures a responsive and aesthetically pleasing design across various devices and screen sizes. PHP handles server-side processing, facilitating seamless communication between the front-end and back-end components.

# Table of Contents

Contents

[ACKNOWLEDGEMENT 5](#_Toc154161227)

[ABSTRACT 6](#_Toc154161228)

[Table of Contents 7](#_Toc154161229)

[LIST OF ABBREVIATIONS 9](#_Toc154161230)

[List of Figures 10](#_Toc154161231)

[List of Tables 11](#_Toc154161232)

[References 12](#_Toc154161233)

[1 Chapter 1: INTRODUCTION 13](#_Toc154161234)

[1.1 Introduction of Ordering web application 13](#_Toc154161235)

[1.2 Problem statement 13](#_Toc154161236)

[1.3 Objectives 14](#_Toc154161237)

[1.4 Scope and limitation 14](#_Toc154161238)

[2 Chapter 2: BACKGROUND STUDY AND LITRATURE REVIEW 16](#_Toc154161239)

[2.1 Background Study 16](#_Toc154161240)

[2.2 Literature Review 17](#_Toc154161241)

[2.3 Existing Systems Overview 17](#_Toc154161242)

[2.3.1 Big Basket 17](#_Toc154161243)

[3 SYSTEM ANALYSIS 18](#_Toc154161244)

[3.1 Software Development Model 18](#_Toc154161245)

[3.1.1 Waterfall Model 18](#_Toc154161246)

[Requirement analysis: 19](#_Toc154161247)

[3.2 Requirements specification 20](#_Toc154161248)

[3.2.1 Functional Requirements 20](#_Toc154161249)

[3.2.2 Nonfunctional requirements 21](#_Toc154161250)

[3.3 Feasibility Study 22](#_Toc154161251)

[3.3.1 Technical Feasibility: 22](#_Toc154161252)

[3.3.2 Economic Feasibility: 22](#_Toc154161253)

[3.3.3 Operational Feasibility 22](#_Toc154161254)

[3.3.4 Scheduling Feasibility 23](#_Toc154161255)

[4 System Design 24](#_Toc154161256)

[4.1 System Architecture 24](#_Toc154161257)

[4.2 Context Diagram and Data Flow Diagram 24](#_Toc154161258)

[4.2.1 DFD level 0 25](#_Toc154161259)

[4.2.2 DFD level 1 26](#_Toc154161260)

[4.3 Use Case Diagram 26](#_Toc154161261)

[4.4 Activity Diagram 27](#_Toc154161262)

[4.5 Class Diagram 28](#_Toc154161263)

[4.6 Database Design 28](#_Toc154161264)

[4.6.1 E-R diagram 28](#_Toc154161265)

[4.6.2 Data Dictionary 29](#_Toc154161266)

[5 System Development and Implementation 31](#_Toc154161267)

[5.1 Programming platform 31](#_Toc154161268)

[5.1.1 Tools Used 31](#_Toc154161269)

[5.2 Operating environment 31](#_Toc154161270)

[5.2.1 Software Requirements 32](#_Toc154161271)

[5.2.2 Hardware Requirements: 32](#_Toc154161272)

[5.3 Testing and debugging 32](#_Toc154161273)

[6 Conclusion and Future Enhancement 39](#_Toc154161274)

[6.1 Conclusion 39](#_Toc154161275)

[6.2 Future Recommendation 39](#_Toc154161276)

# LIST OF ABBREVIATIONS

**Abbreviations**

HTML: Hypertext Markup Language

PHP: Hypertext Preprocessor

RDBMS: Relational Database Management System

SQL: Structured Query Language

DFD: Data Flow Diagram

ER: Entity Relationship

DD: Data Dictionary

# List of Figures

**Figures**

[Figure 1 Waterfall Model 17](#_Toc154160155)

[Figure 2 System Architecture 22](#_Toc154160156)

[Figure 3 Context diagram 23](#_Toc154160157)

[Figure 4 DFD level 0 23](#_Toc154160158)

[Figure 5: DFD level 1 24](#_Toc154160159)

[Figure 6 Use Case Diagram 24](#_Toc154160160)

[Figure 7 Activity Diagram 25](#_Toc154160161)

[Figure 8 Class Diagram 26](#_Toc154160162)

[Figure 9 E-R Diagram 27](#_Toc154160163)

[Figure 10 Login Test 32](#_Toc154160164)

[Figure 11 Register Now 32](#_Toc154160165)

[Figure 12 Login Successfull 33](#_Toc154160166)

[Figure 13 Add Categories 33](#_Toc154160167)

[Figure 14 Manage Categories Test 34](#_Toc154160168)

[Figure 15 Add Product 34](#_Toc154160169)

[Figure 16 Manage Products 35](#_Toc154160170)

[Figure 17 Manage Invoice 35](#_Toc154160171)

[Figure 18 Sales Report 36](#_Toc154160172)

# List of Tables

**Tables**

[Table 2 Data Dictionary of Seller 30](#_Toc154197827)

[Table 3 Data Dictionary of Table 30](#_Toc154197828)

[Table 4 Data Dictionary of Product 30](#_Toc154197829)

[Table 5 Data Dictionary of Category 31](#_Toc154197830)

[Table 6Test case 32](#_Toc154197831)

# References

1. [Agricultural transformation in Nepal: Trends, prospects, and policy options](https://pim.cgiar.org/2020/11/09/agricultural-transformation-in-nepal-trends-prospects-and-policy-options/)
2. [Nepal’s Agriculture: Farmers’ Problems and Their Solutions](https://medium.com/@theviperidae/nepals-agriculture-farmers-problems-and-their-solutions-2ab8bc5104ec)
3. [10 Major Problems Of Agriculture in Nepal - Nepali Infopedia](https://nepaliinfopedia.com/problems-of-agriculture-in-nepal/)
4. [Buyers' trust and mistrust in e-commerce platforms: a ... - Springer](https://link.springer.com/article/10.1007/s10257-021-00545-0)
5. [E-commerce growth in Nepal – Let’s talk facts!](https://techlekh.com/ecommerce-growth-nepal/)
6. [The Key Success Factors in E-Marketplace Implementation: A Systematic ...](https://ieeexplore.ieee.org/document/8528189)
7. [E-Commerce E-Commerce in Nepal ( legalities ) - Trilegal Nepal](https://trilegalnepal.com/blog/business/e-commerce-in-nepal-legalities)
8. [Digital development and e-commerce in Nepal - The Kathmandu Post](https://kathmandupost.com/columns/2019/07/15/digital-development-and-e-commerce-in-nepal)
9. [Government implements national strategy for e-commerce sector](https://kathmandupost.com/money/2019/08/24/government-implements-national-strategy-for-e-commerce-sector)
10. <http://www.foodmario.com/>

# Chapter 1: INTRODUCTION

## Introduction of Ordering web application

The Ordering Web Application plays a crucial role in efficiently managing facilities and optimizing distribution networks. Designed to address the challenges faced by small businesses in Nepal, this platform serves as a virtual marketplace. It enables sellers to display their products and allows buyers to explore and make purchases seamlessly.

In a country dominated by numerous small enterprises like Nepal, the common hurdle is the limited access to broader markets. Small businesses often find it challenging to reach a wider customer base and expand their operations. The Ordering Web Application aims to bridge this gap by providing a platform that streamlines transactions, connecting sellers and buyers in a user-friendly manner.

Acting as a centralized hub, the platform empowers sellers to effectively showcase their products. This opportunity allows small businesses to present their offerings to a much larger audience, overcoming the constraints of their physical locations. By leveraging this innovative system, businesses in Nepal can break through geographic limitations, gain visibility, and access a broader market, ultimately fostering growth and expansion.

## Problem statement

In Nepal, the prevalence of numerous small businesses is hindered by a common challenge—limited access to broader markets. Despite the rich tapestry of products offered by these enterprises, their potential for growth and expansion is constrained by geographic limitations and a lack of visibility. Small businesses struggle to reach a wider customer base, impeding their ability to transcend physical boundaries and maximize their operations. The Ordering Web Application seeks to address this critical issue by providing a platform that streamlines transactions, connecting sellers and buyers seamlessly. However, the extent of its effectiveness in mitigating the challenges faced by small businesses in Nepal remains a key question. This problem statement aims to investigate the impact of the Ordering Web Application in overcoming market access limitations, enhancing visibility, and fostering growth for small enterprises in the Nepalese business landscape.

## Objectives

The main objective of this project is to provide quality order in today’s era of busy schedules.

* Assess the extent to which the Ordering Web Application facilitates increased market access for small businesses in Nepal, considering geographic constraints.
* Investigate the platform's effectiveness in enhancing visibility for small enterprises, allowing them to overcome challenges related to limited exposure and recognition.
* To reduce operating costs associated with business and vendors.
* To establish effective collaboration with suppliers, ensuring a reliable supply of materials and components.
* Share insights gained from the evaluation to contribute valuable information for policymakers and developers seeking to support the growth of small businesses in Nepal through digital platforms.

## Scope and limitation

**Scope**

* Save time
* It encompasses an analysis of market access, visibility enhancement, and business expansion facilitated by the platform.
* Implement a robust product management system, enabling sellers to add, edit, and delete products, ensuring accurate and up-to-date product information.
* The scope extends to understanding user experiences, transaction efficiency, and identifying opportunities for improvement.
* Ensure geographical accessibility, allowing individuals across Nepal to conveniently access products from local small businesses through the Ordering web application.

**Limitation**

* The study may be limited by the availability of comprehensive data from a diverse range of small businesses using the Ordering Web Application.
* External factors such as economic conditions or regulatory changes that could impact small businesses are beyond the immediate scope.
* The study does not delve into the technical aspects of the Ordering Web Application but focuses on its practical implications for businesses.
* Generalization of findings may be constrained by the specific context of Nepal, and the applicability to other regions may require additional investigation.

# Chapter 2: BACKGROUND STUDY AND LITRATURE REVIEW

## Background Study

In the context of Nepal, the background study for the ordering project encompasses an examination of the unique challenges and opportunities that exist within the country's landscape.

The topographical environment of Nepal presents obstacles to the ordering and supply system. Transportation and logistics are complicated by the harsh geography, which includes mountains and rural locations. Infrastructure constraints, such as inadequate roads and a lack of connectivity, exacerbate the inefficiency of supply chain operations. These geographical limitations have an impact on the timely and cost-effective delivery of commodities, thus the ordering and supply system must adapt to the unique challenges given by Nepal's diversified geography.

Small businesses in Nepal face challenges in market access caused by quantity-related factors. Limited production capacity may hinder their ability to meet high demand, resulting in potential loss of customers to larger competitors. Insufficient resources, such as inventory and manpower, can restrict a small business’s ability to scale up to match market needs. Additionally, challenges may arise in negotiating favorable terms with suppliers and distributors due to the smaller quantities involved, impacting the overall competitiveness of the business in the market. Addressing these quantity-related challenges requires strategic planning, efficient resource management, and creative solutions to enhance market access for small businesses

The regulatory environment for e-commerce in Nepal is still in its early stages of development. Currently, there is no specific law to regulate the e-commerce sector.

## Literature Review

In general, involvement in e-commerce platforms can help small businesses develop and sustain themselves by giving them access to a larger client base, lowering transaction costs, and enhancing supply chain management efficiency. E-commerce platforms can also assist small enterprises in breaking down geographical barriers and reaching customers in outlying places. Furthermore, e-commerce platforms may provide significant data insights to small businesses, allowing them to streamline their operations and improve their products and services. These advantages are not confined to any one country or location, but are available worldwide and in similar economic settings.

## Existing Systems Overview

### Big Basket

**Introduction**

BigBasket is an Indian online grocer headquartered in Bangalore, India, and currently owned by Tata Digital. It was the first online grocer in India, set up in 2011. It is a registered company with name as the Supermarket Grocery Supplies Pvt. Ltd.

**Pros**

* A wide variety of options are available in all the categories at one place for purchasing
* You can find best options at lower prices, right from fresh fruits, spices, dals, seasonings and vegetables to packaged products.

**Cons**

* Only basis on India market
* Security issue

# SYSTEM ANALYSIS

The system analysis phase is pivotal in developing our web-based application for ordering. It involves thoroughly assessing current operations, gathering user requirements, and identifying constraints. The primary objective is to precisely define the system's scope and functionality, ensuring alignment with the goal of overcoming market reach challenges.

## Software Development Model

The Ordering Web Application was developed using the waterfall model for its simplicity, ease, and compatibility with well-defined requirements and familiar technology. With all basic requirements collected and documented, and a clear understanding of the system's end product, the waterfall model proved ideal. No regression to previous stages was necessary, as requirements had been meticulously gathered beforehand. The project management steps and development phases are encapsulated in the waterfall model, ensuring a systematic and well-structured approach to creating the system.

### Waterfall Model

The system follows the waterfall model, chosen for its suitability in handling pre-collected requirements, prepared documentation, and a well-understood technology stack. Its sequential approach aligns well with our clear understanding of the final product. The model's linear progression makes it an ideal fit, eliminating the need to revisit previous stages since requirements were meticulously gathered. The accompanying waterfall model image illustrates the systematic project management steps and phases involved in creating the system.

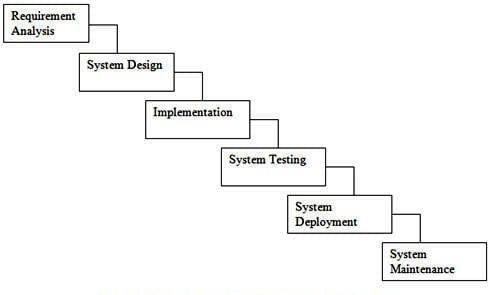


Figure 1 Waterfall Model

Requirement analysis:

During this phase, various requirements were gathered through additional research, new systems, and already existing systems. Gathered and filtered various requirements through research and analysis. Unsuitable needs were discarded.

**System design:**

Structured the system for subsequent actions, examining required paperwork to build the system architecture. Completed tasks include use case, database schema, UI design, DFD, and ERD.

**Implementation & Coding:**

Translated requirement documents and system design into programs, building small elements first and then integrating them into larger modules with connections.

**System Testing:**

Combined modules to form a system, thoroughly testing for functionality and resolving errors. Ensured all components met software criteria.

**System Deployment:**

In this phaseafter the functional and non-functional testing was done, the product is deployed in the customer environment or released into the market. The deployment phase includes installation, migration, and support of the complete system to the user or customer environment.

**System Maintenance:**

After functional and non-functional testing, deployed the product into the customer environment or released it to the market. Included installation, migration, and ongoing support.

## Requirements specification

A software requirements specification (SRS) is a document that describes the functionality, features, design, limitations, and goals of a software system to be developed.

### Functional Requirements

**For Vendor/User**

* The system should display a comprehensive list of available products, including details such as product name, description, price, and availability.
* Customers should be able to use the system to place bookings for products. This includes selecting products, specifying quantities, and confirming the booking.

**For Business Owner:**

* The system should allow the business owner to add new products to the catalog. This includes specifying product details such as name, description, price, and Quantity.
* The business owner should have the capability to delete products from the catalog when necessary.
* The system should allow the business owner to securely log in and log out from the system using unique credentials.
* The business owner should be able to view and update customer requests. This includes confirming bookings, updating order statuses, and addressing customer inquiries.

### Nonfunctional requirements

Nonfunctional requirements define the qualities and characteristics of the system that are not directly related to its specific behaviors or functions. They often represent the performance, reliability, security, and usability aspects of the system. Here are the nonfunctional requirements for the "Ordering web application":

**Performance**

* The system should respond to user interactions within 2 seconds to ensure a responsive user experience.
* The system should be scalable to handle an increasing number of users and products without significant degradation in performance.
* The system should support a minimum of 1000 concurrent users without a significant decrease in throughput.

**Reliability**

* The system should have at least 99.9% uptime, ensuring it is available to users except during scheduled maintenance.
* The system should be designed to handle server failures gracefully, minimizing disruption to users.
* The system should ensure the integrity of data, with a backup and recovery mechanism to prevent data loss in case of unexpected events.

**Security**

* All sensitive data, including user credentials and transaction information, should be encrypted using industry-standard encryption protocols
* Role-based access control should be implemented to restrict unauthorized access to sensitive functionalities.

**Compatibility**

* The system should be compatible with major web browsers, including Chrome, Firefox, Safari, and Edge.
* The user interface should be responsive and optimized for various devices, including smartphones and tablets.

## Feasibility Study

Conducted to evaluate the viability of implementing the "Ordering Web Application." Examines technical, economic, operational, and scheduling aspects to determine project feasibility. A comprehensive assessment is undertaken to decide whether to proceed with the implementation.

### **Technical Feasibility:**

Evaluate if technical requirements can be met using available resources, considering compatibility with existing hardware and software, and the availability of expertise for system development and maintenance.

### Economic Feasibility:

Assess the financial viability of the Ordering Web Application, including a comprehensive analysis of costs, benefits, and potential return on investment.

### Operational Feasibility

Evaluate the operational capabilities of the Ordering Web Application, focusing on its integration into existing business processes, impact on operations, and the availability of resources for system management.

### Scheduling Feasibility

Assess the feasibility of meeting project milestones and deadlines within the desired timeframe, ensuring the availability of necessary resources, including personnel and technology, to adhere to the project schedule.

# System Design

It provides a comprehensive understanding of how the system will be structured and how its various parts will interact

## System Architecture

Provide a high-level overview of the system architecture, including the client-side (frontend), server-side (backend), database, and any external services or APIs.

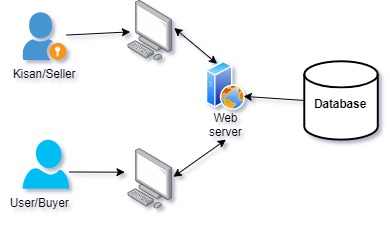


Figure 2 System Architecture

## Context Diagram and Data Flow Diagram

A context diagram is a high-level visual representation that provides an overview of a system and its interactions with external entities. It is used to depict the boundaries of the system and illustrate how it interacts with its environment. The primary purpose of a context diagram is to show the external entities that communicate with the system without going into the internal details of the system. in sort

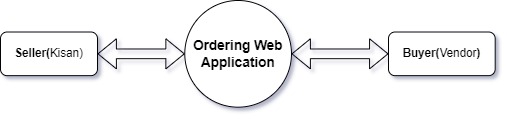


Figure 3 Context diagram

### DFD level 0

A simplified and high-level view of the Ordering Web Application, emphasizing major processes in product and order management. Detailed processes, such as adding, removing, editing, updating, searching, and booking products, are elaborated in subsequent levels of the DFD hierarchy.

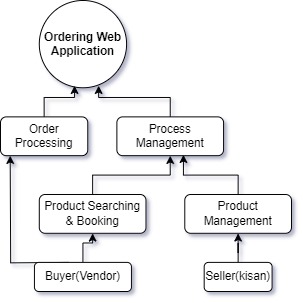


Figure 4 DFD level 0

**Order Processing:** The process responsible for handling order-related operations, including order placement and fulfillment.

**Product Management:** The process responsible for managing product information, including search and booking functionalities.

The arrows represent the flow of data between the processes, indicating how information is exchanged within the system.

### DFD level 1

A Level 1 Data Flow Diagram (DFD) provides a more detailed breakdown of the processes identified in the Level 0 DFD.

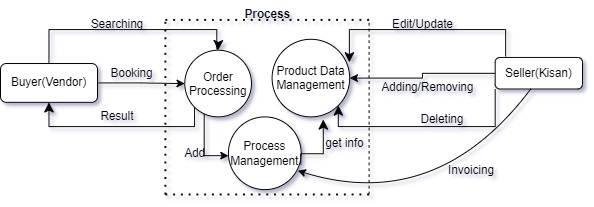


Figure 5: DFD level 1

Elaborates on sub-processes and data flows within major processes, such as "Order Processing" and "Product Data Management" in the Ordering Web Application.

## Use Case Diagram

A use case diagram is a graphical representation of the system's functionality and the actors that interact with the system.

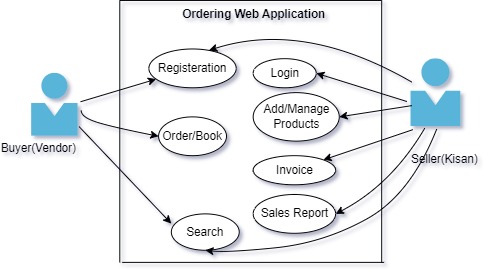


Figure 6 Use Case Diagram

## Activity Diagram

A type of UML diagram depicting the flow of activities and actions within a system or business process. Useful for illustrating dynamic system characteristics, including the sequence of actions, decision points, and concurrent operations.

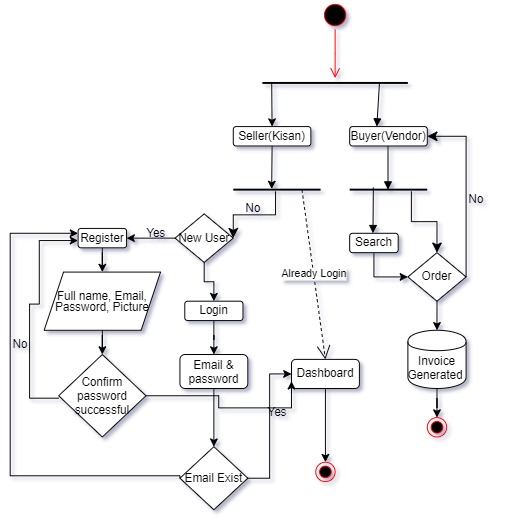


Figure 7 Activity Diagram

## Class Diagram

Class diagrams are often used to model the data structure of an application, representing how different entities (objects or classes) are organized and how they relate to each other. Each class in a class diagram corresponds to a table in a relational database, and the attributes of the class correspond to the columns of the table.

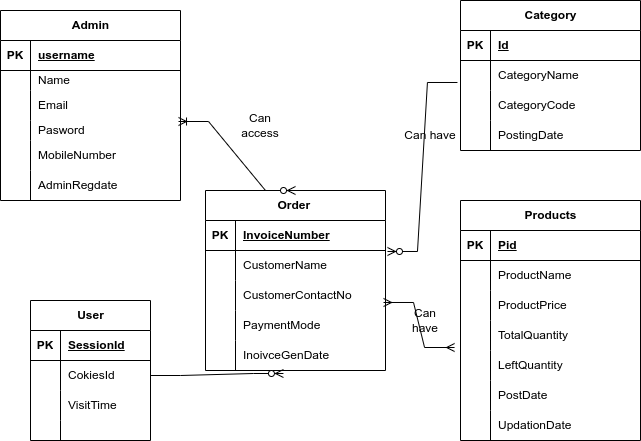


Figure 8 Class Diagram

## Database Design

It is a critical aspect of software development that involves creating a structured and efficient organization of data to meet the requirements of an application or system.

### E-R diagram

By using an E-R diagram, the application development team can ensure that the database schema is well-designed and efficient, which can improve the performance of the application. It can also help to identify any potential issues or inconsistencies in the database schema before the application is launched, allowing them to be addressed before they become bigger problems.

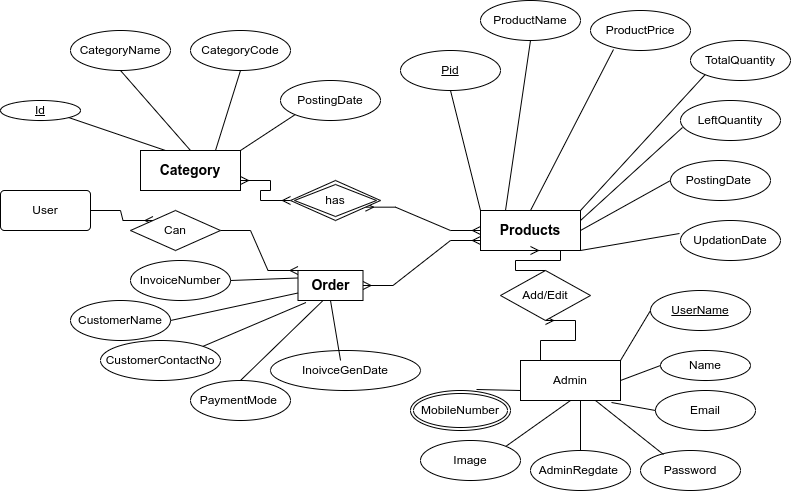


Figure 9 E-R Diagram

In this system, there are seven different entities namely: category, orders, products and user form. The **category** has various attributes including: Id, CategoryName, CategoryCode, PostingDate.**Order** has attributes including: id, ProductId, QuantityId, InvoiceNumber, CustomerName, CustomerContactNo, PaymentMode and InoivceGenDate. **Products** also has Id, ProductName, ProductPrice, LeftQuantity, TotalQuantity, CategoryCode, CategoryName, PostingDate and UpdationDate, at last **user\_form** has attribute called Id, Name, UserName, MobileNumber, Email, Password, Image, AdminRegdate and UpdationDate.

### Data Dictionary

A data dictionary is a centralized repository that stores metadata about the data within a database. It contains information about the structure, organization, and usage of data in a database. This provide details for each field in the tables, specifying the field name, type, description, and constraints (e.g., Primary Key, Foreign Key).

**Seller Table**

Table 2 Seller DD

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Type** | **Description** | **Constrains** |
| Username | Int(100) | Unique seller ID | Primary key |
| Full Name | Varchar(100) | Full Name of Seller | Not Null |
| Email | Varchar(50) | Valid Email Id | Not Null |
| MobileNumber | Bigint(14) | Valid Mobile Number | Not Null |
| Password | Nvarchar(24) | Password of seller | Not Null |
| AdminRegdate | Timestamp | Autogenerated | Current\_timestap() |

**Order Table**

Table 3 Table DD

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Type** | **Description** | **Constrains** |
| CustomerContactNo | Bigint(14) | Contact number | Primary key |
| Customer Name | Nvarchar(100) | Name of customer | Not null |
| PaymentMode | Varchar(10) | Method of payment | Default Cash |
| InvoiceGenDate | Timestamp | Auto generated | Current\_timestap() |

**Product Table**

Table 4 Product DD

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Type** | **Description** | **Constrains** |
| Pid | Int(11) | Product id | Primary key |
| Product Name | Nvarchar(100) | Product Name | Not Null |
| Product Price | Int(11) | Price of product | Not null |
| Total Quantity | Int(11) | Total product | Not Null |
| Left Quantity | Int(11) | Left product | Not Null |
| Post Date | Timestamp | Auto generated | Current\_timestap() |
| Update Date | Timestamp | Auto generated | Current\_timestap() |

**Category**

Table 5 Category DD

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Type** | **Description** | **Constrains** |
| Cid | Int(11) | Id of category | Primary key |
| CategoryName | Nvarchar(200) | Name of category | Not Null |
| CategoryCode | Varchar(50) | Code of Category | Default null |

# System Development and Implementation

## Programming platform

The programming platform for this project encompasses the utilization of MariaDB for database creation and management, PHP on the back-end to handle static data generation, and Bootstrap for designing and styling interfaces. Additionally, jQuery is employed on the front-end to enhance functionality. In summary, the technology stack includes MariaDB, PHP, Bootstrap, and jQuery to deliver a robust and feature-rich web application.

### Tools Used

For the Ordering web application

**Operating System:** Manjaro Linux

**Technology [Front-End]:** HTML, CSS, JAVASCRIPT, BOOTSTRAP, jQuery

**Technology [Back-End]:** PHP

**Version Control:** git

**Code Editor:** Visual studio code.

**Database:** MariaDB

**Server:** Xamp apache server

## Operating environment

To ensure optimal performance and functionality of the system developed in this project, the following minimum requirements for the operating environment, encompassing both software and hardware components, should be met:

### Software Requirements

* Latest versions of popular web browsers such as Google Chrome, Mozilla Firefox, Safari, or Microsoft Edge.
* MariaDB (or compatible relational database systems) with the necessary extensions and configurations to support the application's database structure.
* PHP 7.x or later with required extensions and modules.
* Bootstrap 4 or later for responsive and visually appealing user interfaces.
* jQuery for enhanced interactivity and dynamic content.

### Hardware Requirements:

* Dual-core processor (or equivalent) with a clock speed of 2.0 GHz or higher.
* Minimum 2 GB of RAM for adequate system responsiveness.
* Sufficient disk space to accommodate the application code, database, and any additional resources. A minimum of 20 GB is recommended.
* Stable internet connection for accessing and interacting with the web application

## Testing and debugging

In the testing phase following tests were done.

Table 6Test case

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **S.N** | **Unit Test** | **Test** | **Expected Result** | **Test**  **Outcomes** | **Evidence** | **Results** |
| 1 | Login test | Check login credentials for valid admin name and password | Invalid email and password | Invalid email and password | Test 1.1 | pass |
| 2 | Registration | New admin created | Created new user save on database | Info successfully Submitted | Test 2.1 | pass |
| 3 | Login Admin | Check login credentials for valid email and password | Redirect to Dashboard | Dashboard Details | Test 3.1 | Pass |
| 4 | Add Categories | Add the new Categories | New Categories is added and show on database | successful | Test 4.1 | Pass |
| 5 | Manage Categories | Edit Functionality on Categories | Show all categories | Successful | Test 5.1 | pass |
| 6 | Add Products | Add the new Product | Add product with all requirement | Successful | Test 6.1 | pass |
| 7 | Manage Products | CRUD functionality | Show all products | successful | Test 7.1 | pass |
| 8 | Manage Invoices | Show all order | Generated invoice on PDF | successful | Test 8.1 | pass |
| 9 | Sales Report | Selected range of date | Show result from selected date | Shows total sales | Test 9.1 | pass |

**EVIDENCES 1.1**

**Unit Test**: Login test

**Test**: Check Login credentials for valid admin name and password

**Expected Result**: incorrect user name or password.

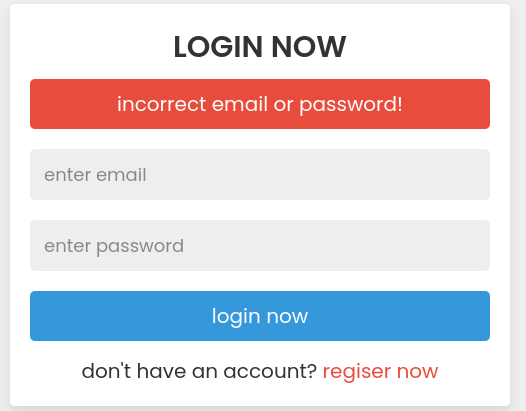


Figure 10 Login Test

**TEST EVIDENCE 2.1**

**Unit Test:** Registration

**Test:** Register to the system as Admin

**Expected Result:** Created new Kisan save on database

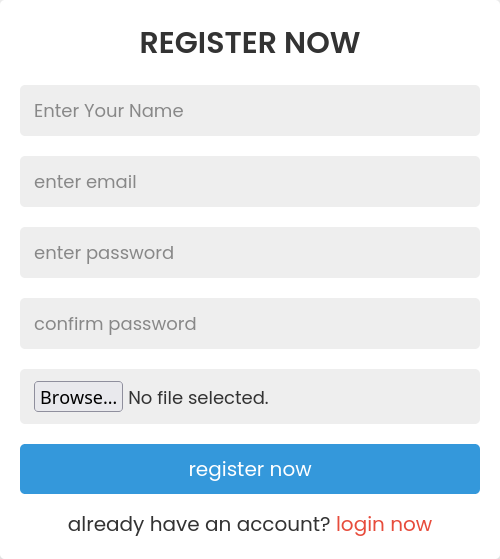


Figure 11 Register Now

**TEST EVIDENCE 3.1**

**Unit Test:** Log In & Dashboard

**Test:** Check login credentials for valid email and password

**Expected Result**: Redirected to Dashboard Page

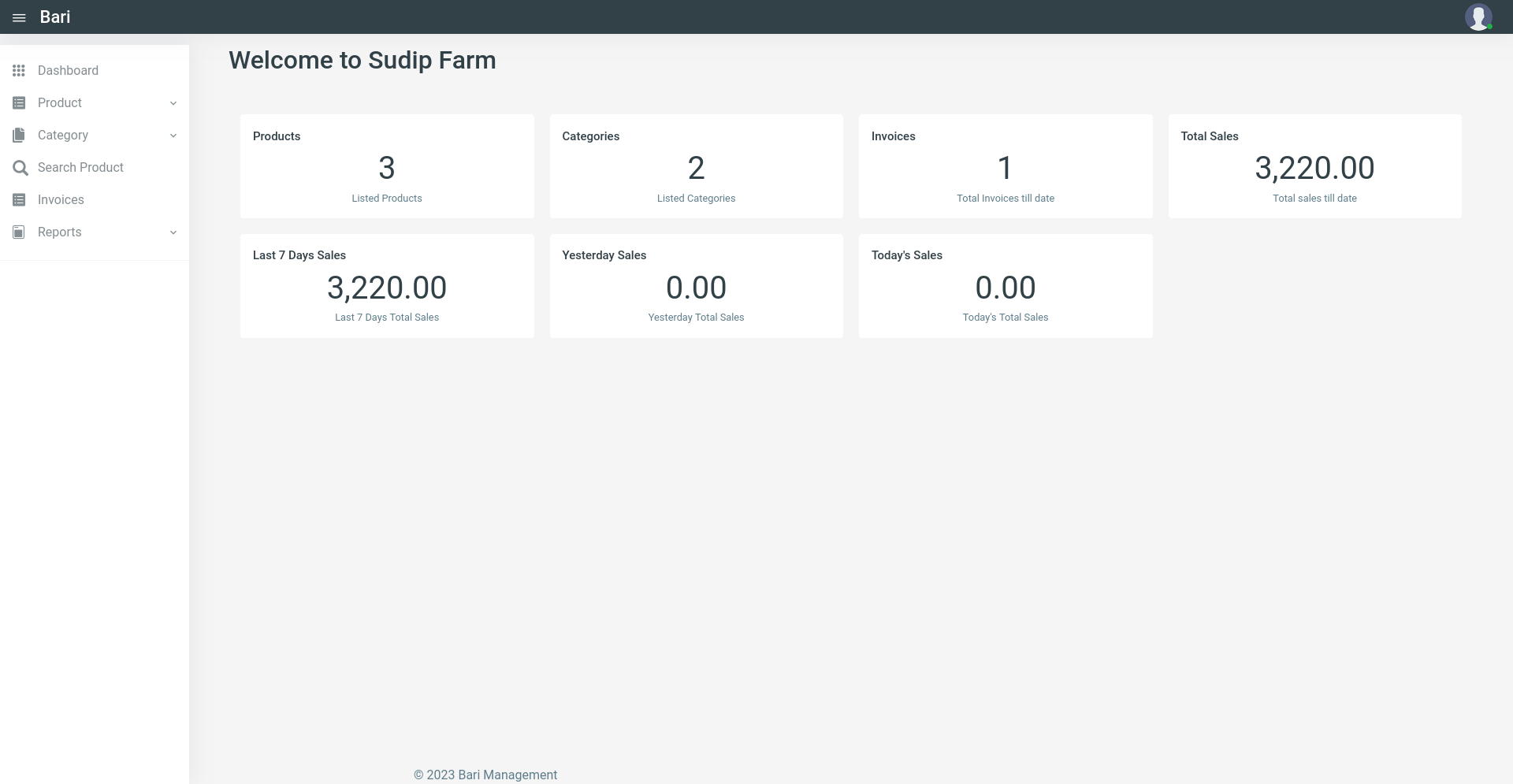


Figure 12 Login Successful

**TEST EVIDENCE 4.1**

**Unit Test**: Add Categories

**Test**: Add the new Categories

**Expected Result:** New Categories is added and show on database

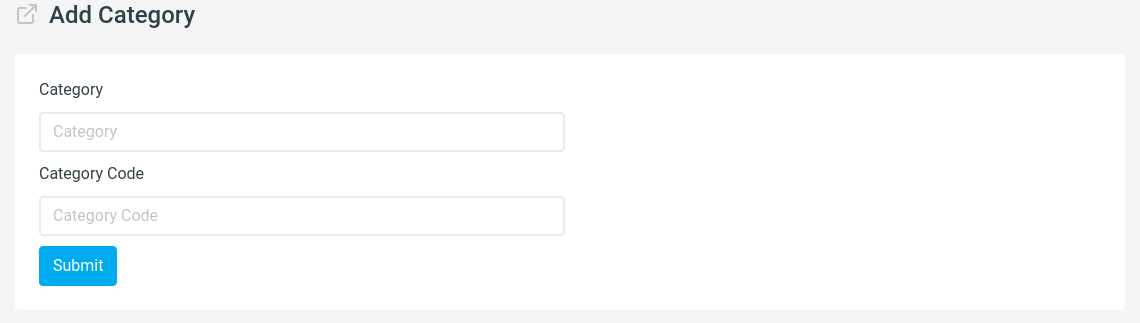


Figure 13 Add Categories

**TEST EVIDENCE 5.1**

**Unit Test**: Manage Categories

**Test**: Edit Functionality on Categories

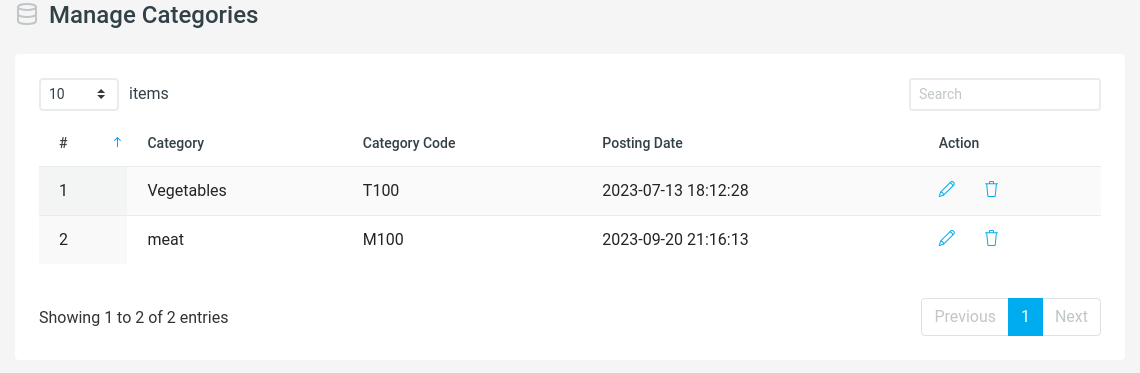
**Expect Result**: Show all categories

Figure 14 Manage Categories Test

**TEST EVIDENCE 6.1**

**Unit Test**: Add Product

**Test:** Add the new Product

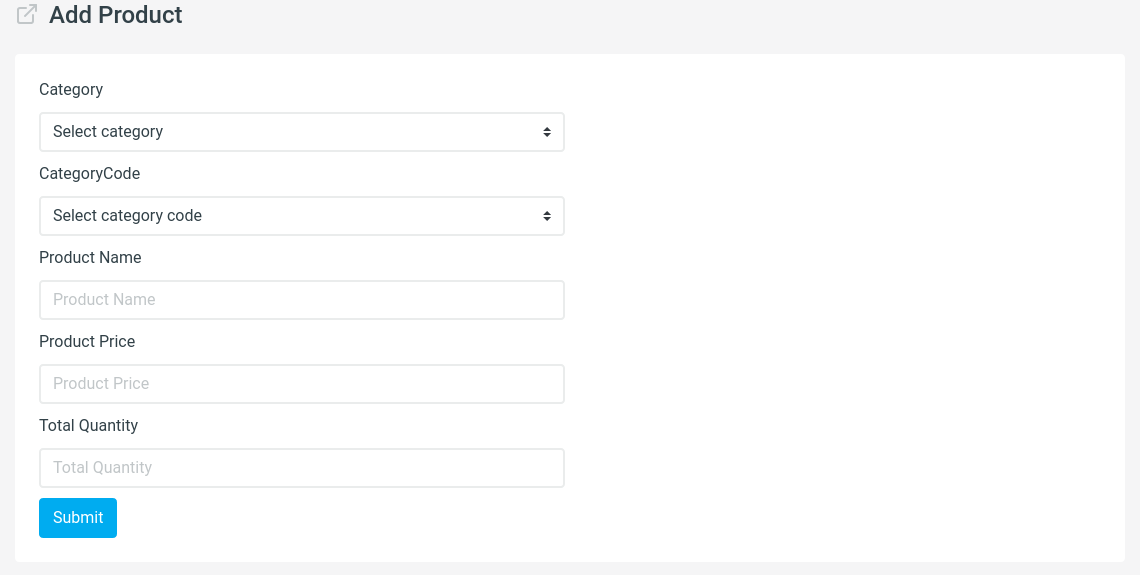
**Expect Result**: Add product with all requirement

Figure 15 Add Product

**TEST EVIDENCE 7.1**

**Unit Test**: Manage Products

**Test**: CRUD functionality on Products

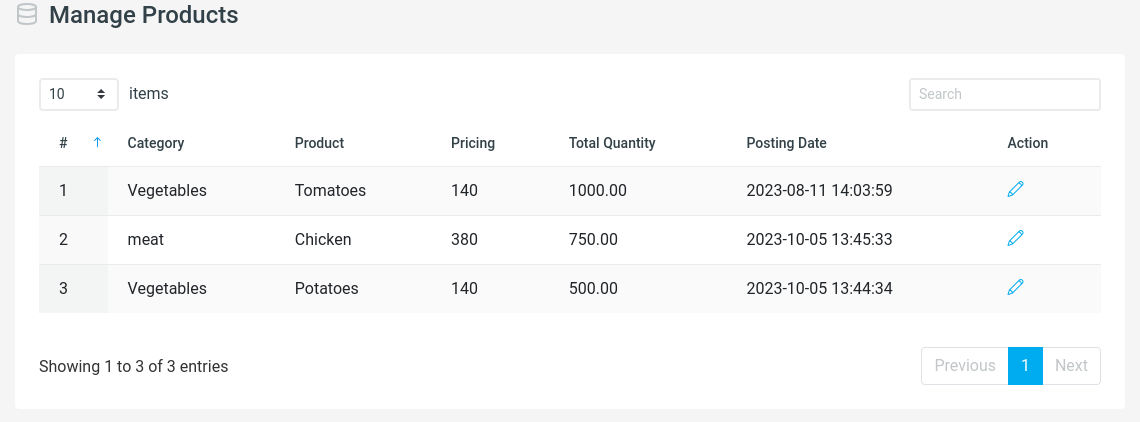
**Expect Result**: Show all products

Figure 16 Manage Products

**TEST EVIDENCE 8.1**

**Unit**: Test: Manage Invoices

**Test:** Show all order

**Expect Result:** Generated invoice on PDF

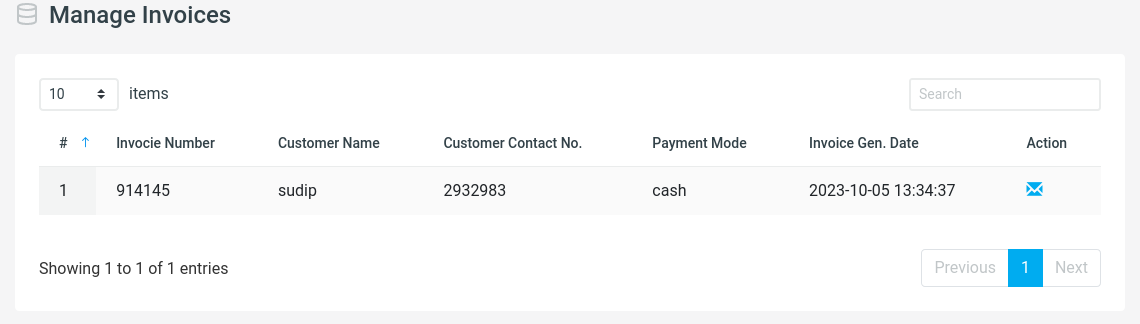


Figure 17 Manage Invoice

**TEST EVIDENCE 9.1**

**Unit Test:** Sales report

**Test:** Selected range of date

**Expect Result**: Show result from selected date

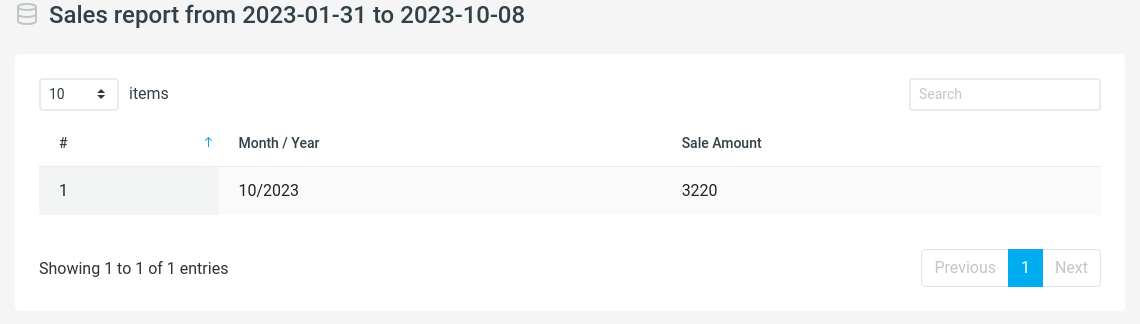


Figure 18 Sales Report

These testing and debugging strategies, along with specific test cases, ensure the systematic identification and resolution of issues during the development and testing phases. Regular testing and debugging practices contribute to the delivery of a stable and reliable software product.

# Conclusion and Future Enhancement

## Conclusion

The project was successfully completed within the stipulated timeframe, marked by dedicated efforts and meticulous attention to detail. All functionalities outlined earlier have been seamlessly incorporated into both the admin and user modules, ensuring the project's objectives are met. The primary aim of this endeavor is to ensure the product's resilience in the challenging contemporary landscape.

At its current stage, the system operates effortlessly, thanks to rigorous development and testing. The admin module facilitates efficient management, while users can easily and conveniently access the services provided by the website. Moving forward, the project will be well-maintained, enabling farmers to register as users, adding or modifying products as needed.

In conclusion, the project stands as a testament to effective management and thorough implementation. As it reaches its completion, farmers will benefit from a user-friendly platform that caters to their specific needs, contributing to a seamless and accessible experience for all users.

## Future Recommendation

Anticipating a continuous evolution, we have an array of exciting features planned for future upgrades. The system is poised for a significant enhancement through the integration of a secure payment module. User-centric improvements will be a top priority, driven by valuable feedback, comments, and ratings.

Key areas for future development include:

**Enhanced Payment Options**: Integration of an online payment feature, allowing users to make full or partial settlements securely.

**Direct Messaging Feature:** Implementation of a messaging feature enabling direct communication between buyers and sellers.

**Database Optimization:** Continuous improvement and regular updates to enhance database functionality.

**Back-End Maintenance and Upgrades:** Ongoing efforts to maintain and upgrade the back-end infrastructure for optimal performance.

**User Interface Enhancements:** Regular updates to user interfaces to align with evolving user needs and technological advancements.

**Performance Optimization**: Focus on improving overall system performance for a seamless and responsive user experience.

Additionally, the incorporation of several live-updated page elements will ensure that users have access to real-time information, contributing to a dynamic and engaging platform. These future recommendations align with our commitment to delivering a cutting-edge platform that adapts to user requirements and technological advancements, ensuring a continuously improving and enriching user experience.

# Source code

Login.php

|  |
| --- |
| <?php  include('includes/config.php');  // session\_start();  if (isset($\_POST['submit'])) {      $email = mysqli\_real\_escape\_string($con, $\_POST['email']);      $pass = mysqli\_real\_escape\_string($con, ($\_POST['password']));      $select = mysqli\_query($con, "SELECT \* FROM `user\_form` WHERE Email = '$email' AND Password = '$pass'") or die('query failed');      if (mysqli\_num\_rows($select) > 0) {          $row = mysqli\_fetch\_assoc($select);          $\_SESSION['user\_id'] = $row['ID'];          // Set the cookie with the username          $username = $row['Username'];          setcookie('username', $username, time() + 3600, '/');          header('location: dashboard.php');      } else {          $message[] = 'incorrect email or password!';      }  }  ?>  <html lang="en">  <head>      <title>login</title>         <link rel="stylesheet" href="dist/css/userForm.css">      <link rel="stylesheet" href="dist/css/navbar.css">  </head>  <body>      <nav class="navbar">          <div class="navbar-brand">BARI</div>          <ul class="navbar-items">              <li><a href="../index.php">home</a></li>              <li><a href="logout.php">logout</a></li>          </ul>      </nav>      <div class="form-container">          <form action="" method="post" enctype="multipart/form-data">              <h3>login now</h3>              <?php if (isset($message)) {                  foreach ($message as $message) {                      echo '<div class="message">'. essage.'</div>';} } ?>              <input type="email" name="email" placeholder="enter email" class="box" required>              <input type="password" name="password" placeholder="enter password" class="box" required>              <input type="submit" name="submit" value="login now" class="btn">              <p>don't have an account? <a href="register.php">regiser now</a></p>          </form>      </div>  </body>  </html> |

Register.php

|  |
| --- |
| <?php  include 'includes/config.php';  if (isset($\_POST['submit'])) {     $name = mysqli\_real\_escape\_string($con, $\_POST['name']);     $email = mysqli\_real\_escape\_string($con, $\_POST['email']);     $pass = mysqli\_real\_escape\_string($con, ($\_POST['password']));     $cpass = mysqli\_real\_escape\_string($con, ($\_POST['cpassword']));     $image = $\_FILES['image']['name'];     $image\_size = $\_FILES['image']['size'];     $image\_tmp\_name = $\_FILES['image']['tmp\_name'];     $image\_folder = 'uploaded\_img/' . $image;     $select = mysqli\_query($con, "SELECT \* FROM `user\_form` WHERE Email = '$email' AND Password = '$pass'") or die('query failed');     if (mysqli\_num\_rows($select) > 0) {        $message[] = 'user already exist';     } else {        if ($pass != $cpass) {           $message[] = 'confirm password not matched!';        } elseif ($image\_size > 2000000) {           $message[] = 'image size is too large!';        } else {           $insert = mysqli\_query($con, "INSERT INTO `user\_form`(Name, Email, Password, Image) VALUES('$name', '$email', '$pass', '$image')") or die('query failed');           if ($insert) {              move\_uploaded\_file($image\_tmp\_name, $image\_folder);              $message[] = 'registered successfully!';              header('location: login.php');           } else {              $message[] = 'registeration failed!';           }        }     }  }  ?>  <html lang="en">  <head>      <title>register</title>        <link rel="stylesheet" href="dist/css/userForm.css">  </head>  <body>     <div class="form-container">        <form action="" method="post" enctype="multipart/form-data">           <h3>register now</h3>           <?php           if (isset($message)) {              foreach ($message as $message) {                 echo '<div class="message">' . $message . '</div>';              }           }           ?>           <input type="text" name="name" placeholder="Enter Your Name" class="box" required>           <input type="email" name="email" placeholder="enter email" class="box" required>           <input type="password" name="password" placeholder="enter password" class="box" required>           <input type="password" name="cpassword" placeholder="confirm password" class="box" required>           <input type="file" name="image" class="box" accept="image/jpg, image/jpeg, image/png">           <input type="submit" name="submit" value="register now" class="btn">           <p>already have an account? <a href="login.php">login now</a></p>        </form>     </div>  </body>  </html> |

Addproduct.php

|  |
| --- |
| <?php  error\_reporting(E\_ALL);  ini\_set('display\_errors', 1);  include('includes/config.php');  if (strlen($\_SESSION['user\_id'] == 0)) {      header('location:logout.php');  } else {      if (isset($\_POST['submit'])) {          //Getting Post Values          $catname = $\_POST['category'];          $catcode = $\_POST['categorycode'];          $pname = $\_POST['productname'];          $pprice = $\_POST['productprice'];          $tquantity = $\_POST['quantity'];          $query = mysqli\_query($con, "insert into products(CategoryName,CategoryCode,ProductName,ProductPrice,TotalQuantity) values('$catname','$catcode','$pname','$pprice','$tquantity')");          if ($query) {              echo "<script>alert('Product added successfully.');</script>";              echo "<script>window.location.href='add-product.php'</script>";          } else {              echo "<script>alert('Something went wrong. Please try again.');</script>";              echo "<script>window.location.href='add-product.php'</script>";          }      }  ?>  <html lang="en">  <head>  <link href="vendors/jquery-toggles/css/toggles.css" rel="stylesheet" type="text/css">  <link href="vendors/jquery-toggles/css/themes/toggles-light.css" rel="stylesheet" type="text/css">  <link href="dist/css/style.css" rel="stylesheet" type="text/css">      </head>      <body>          <div class="hk-wrapper hk-vertical-nav">   <?php include\_once('includes/navbar.php');  include\_once('includes/sidebar.php');         ?>       <div id="hk\_nav\_backdrop" class="hk-nav-backdrop"></div>   <div class="hk-pg-wrapper">         <nav class="hk-breadcrumb" aria-label="breadcrumb">     <ol class="breadcrumb breadcrumb-light bg-transparent">   <li class="breadcrumb-item"><a ref="#">Product</a></li>     <li class="breadcrumb-item active" aria-current="page">Add</li>                      </ol>                  </nav>                  <div class="container">  <div class="hk-pg-header"> <h4 class="hk-pg-title"><span class="pg-title-icon"><span class="feather-icon"><i data-feather="external-link"></i></span></span>Add Product</h4>                      </div>                      <div class="row">                          <div class="col-xl-12">                              <section class="hk-sec-wrapper">                                <div class="row">                                      <div class="col-sm">                                          <form class="needs-validation" method="post" novalidate>                                              <div class="form-row">                                                  <div class="col-md-6 mb-10">                                                      <label for="validationCustom03">Category</label>                                                      <select class="form-control custom-select" name="category" required>                                                          <option value="">Select category</option>                                                          <?php                                                          $ret = mysqli\_query($con, "select CategoryName from category");                                                          while ($row = mysqli\_fetch\_array($ret)) { ?>                                                              <option value="<?php echo $row['CategoryName']; ?>"><?php echo $row['CategoryName']; ?></option>                                                          <?php } ?>                                                      </select>                                                      <div class="invalid-feedback">Please select a category.</div>                                                  </div>                                              </div>                                              <div class="form-row">                                                  <div class="col-md-6 mb-10">                                                      <label for="validationCustom03">CategoryCode</label>                                                      <select class="form-control custom-select" name="categorycode" required>                                                          <option value="">Select category code</option>                                                          <?php                                                          $ret = mysqli\_query($con, "select CategoryCode from category");                                                          while ($row = mysqli\_fetch\_array($ret)) { ?>                                                              <option value="<?php echo $row['CategoryCode']; ?>"><?php echo $row['CategoryCode']; ?></option>                                                          <?php } ?>                                                      </select>                                                      <div class="invalid-feedback">Please select a category code.</div>                                                  </div>                                              </div>                                              <div class="form-row">                                                  <div class="col-md-6 mb-10">                                                      <label for="validationCustom03">Product Name</label>                                                      <input type="text" class="form-control" id="validationCustom03" placeholder="Product Name" name="productname" required>                                                      <div class="invalid-feedback">Please provide a valid product name.</div>                                                  </div>                                              </div>                                              <div class="form-row">                                                  <div class="col-md-6 mb-10">                                                      <label for="validationCustom03">Product Price</label>                                                      <input type="text" class="form-control" id="validationCustom03" placeholder="Product Price" name="productprice" required>                                                      <div class="invalid-feedback">Please provide a valid product price.</div>                                                  </div>                                              </div>                                              <div class="form-row">                                                  <div class="col-md-6 mb-10">                                                      <label for="validationCustom03">Total Quantity</label>                                                      <input type="text" class="form-control" id="validationCustom03" placeholder="Total Quantity" name="quantity" required>                                                      <div class="invalid-feedback">Please provide a valid Total Quantity.</div>                                                  </div>                                              </div>                                              <button class="btn btn-primary" type="submit" name="submit">Submit</button>                                          </form>                                      </div>                                  </div>                              </section>                          </div>                      </div>                  </div>                  <!-- Footer -->                  <?php include\_once('includes/footer.php'); ?>                  <!-- /Footer -->              </div>              <!-- /Main Content -->          </div>          <script src="vendors/jquery/dist/jquery.min.js"></script>          <script src="vendors/popper.js/dist/umd/popper.min.js"></script>          <script src="vendors/bootstrap/dist/js/bootstrap.min.js"></script>          <script src="vendors/jasny-bootstrap/dist/js/jasny-bootstrap.min.js"></script>          <script src="dist/js/jquery.slimscroll.js"></script>          <script src="dist/js/dropdown-bootstrap-extended.js"></script>          <script src="dist/js/feather.min.js"></script>          <script src="vendors/jquery-toggles/toggles.min.js"></script>          <script src="dist/js/toggle-data.js"></script>          <script src="dist/js/init.js"></script>          <script src="dist/js/validation-data.js"></script>      </body>      </html>  <?php } ?> |