

BASAVARAJESWARI GROUP OF INSTITUTIONS

BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT

(An Autonomous Institution under Visvesvaraya Technological University)



NACC & NBA Accredited Institution*
(Recognized by Govt. of Karnataka, approved by AICTE, New Delhi & Affiliated to
Visvesvaraya Technological University, Belagavi)
"JnanaGangotri" Campus, No.873/2, Ballari-Hospet Road, Allipur, Ballari-583
104 (Karnataka) (India)
Ph: 08392 – 237100 / 237190, Fax: 08392 – 237197



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

A Mini Project Report On

“Agro-Culture System”

Submitted to the Department of Computer Science and Engineering of
Visvesvaraya Technological University, Belagavi in partial fulfilment for the
requirement of Bachelor of Engineering

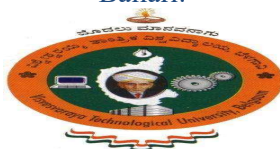
Project Associates:

Sadiya Farhad

3BR22CS415

Under the Guidance of

Dr P Paniram Prasad
Dept of CS&E, BITM,
Ballari.



Visvesvaraya Technological University

Belagavi, Karnataka

2024

BASAVARAJESWARI GROUP OF INSTITUTIONS

BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT
(An Autonomous Institution under Visveswaraya Technological University)



NACC & NBA Accredited Institution*
(Recognized by Govt. of Karnataka, approved by AICTE, New Delhi & Affiliated to
Visveswaraya Technological University, Belgavi)
"JnanaGangotri" Campus, No.873/2, Ballari-Hospet Road, Allipur, Ballari-583
104 (Karnataka) (India)
Ph: 08392 – 237100 / 237190, Fax: 08392 – 237197



**DEPARTMENT OF COMPUTER SCIENCE &
ENGINEERING**
CERTIFICATE

This is to certify that the Mini project work entitled “Agro-Culture System” is a bonafide work carried out by **Sadiya Farhad** in partial fulfilment for the requirement of **Bachelor Degree in Computer Science & Engineering** in the VISVESVARAYA TECHNOLOGICAL UNIVERSITY, Belagavi during the academic year 2023-2024. It is certified that all corrections and suggestions indicated for internal assessment have been incorporated in the report deposited in the library. The project has been approved as it satisfies the academic requirements in respect of Mini project work prescribed for Bachelor of Engineering Degree.

Signature of Project guide

Dr P Paniram Prasad

Signature Coordinator

Dr P Paniram Prasad

Signature of HOD

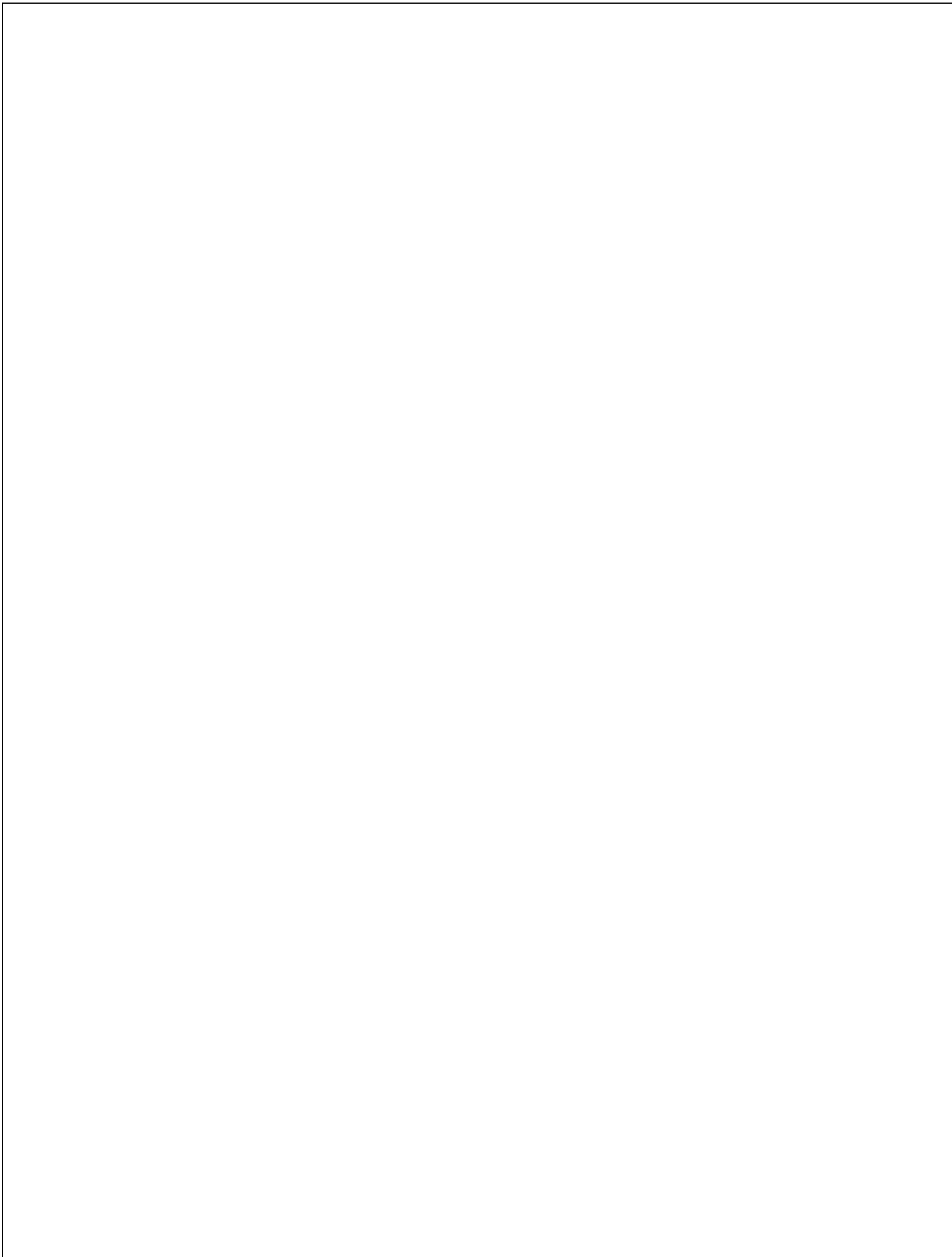
Dr.R.N.Kulkarni

Name of the Examiners

1.

2.

Signature with Date



ABSTRACT

Agro-Culture System acts as interface between farmer and buyer for commercial purpose. Agro-Culture system will make better connection among the farmers and buyers ensure quality food. Agro-Culture System in PHP is developed using PHP, CSS, Bootstrap, and Java Script. This project contains an admin side where he/she can manage all the agricultural and farming records. The admin plays important role in the management of this system. The admin side where farmer can post all his good which are available for sale. This project also has user side where he/she can view the goods like fruits, vegetables details like price, quantity etc. and can place order for it. Agro-Culture System is eco-friendly farming system

ACKNOWLEDGEMENTS

Salutations to our beloved and highly esteemed institute, “**BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT**” for having well qualified staff and labs furnished with necessary Computing facilities. We express our sincere thanks to our guide **Dr. P Pani rama Prasad** for giving us constant encouragement, support and valuable guidance throughout the course of the project without whose stable guidance this project would not have been achieved. We would also like to thank our Mini Project Coordinator **Dr P Pani rama Prasad** for their continuous motivation towards completion of the Project work. We express wholehearted gratitude to **Dr.R.N.KULKARNI** who is our respectable HOD of Computer Science & Engineering Department. We wish to acknowledge his help who made our task easy by providing with his valuable help and encouragement and also our due thanks to **Dr.Yadawalli Basavaraj**, Principal, as we consider ourselves very lucky to have such excellent computing facilities and their inspiration throughout our professional course. We also thank the non-teaching staff of CSE Department who guided at the time of difficulties. Finally we thank all those who are involved directly and indirectly in completion of our project.

Table of Contents		
Chapter No	Chapter Name	Page No
	Abstract	I
	Acknowledgement	II
	Table of Contents	III
	List of Figures	IV
1	Chapter 1 Introduction 1.1 Introduction to Project 1.2 Problem Statement 1.3 Objectives 1.4 Scope of the Project	1
2	Chapter 2 Literature survey	2
3	Chapter 3 System Analysis 3.1 Existing System 3.2 Proposed System	3
4	Chapter 4 System Requirement Specification 4.1 Software Requirement Specification 4.1.1 Functional Requirements 4.1.2 Non Functional Requirements 4.1.3 Hardware Requirements 4.1.4 Software Requirements 4.2 Software Tools	4
5	Chapter 5 Description of Modules 5.1 System Architecture 5.2 Design	5-6
6	Chapter 6 Results and Discussion 6.1 Implementation 6.2 Screenshots of output	7-9
7	Chapter 7 Testing 7.1 Types of Testing 7.2 Test Cases	10-11
	Conclusion	12
	References	13

CHAPTER I: INTRODUCTION

1.1 INTRODUCTION

Agro Culture is the farmer system where they can plan, monitor and analyze the activity of the farmers production system. It manages farmer operation with one system and organizes data in one place. It helps smart farmers become even smarter. This creates in partnership with growers and buyers. It inspire farmer to produce and buyers to consume fresh goods.

Agro Culture System will make better connection among Farmers and Buyers ensure quality food. Standardize and increase efficiency of agro culture process.

1.2 PROBLEM STATEMENT

In many rural and semi-urban areas, farmers face challenges in reaching potential buyers for their produce. Traditional marketplaces often involve multiple intermediaries, which can reduce the farmers' profit margins and increase the cost for buyers. There is a need for a digital platform that connects farmers directly with buyers, allowing for a more efficient and transparent transaction process.

1.3 OBJECTIVES

The specific objectives of the project include:

- To provide qualitative foods to the buyers.
- Implementing an automated/online agro culture system.
- To inspire farmer to produce quality goods and supply to the buyers.
- Eco-friendly farming system.

1.4 Scope of Project

It is focused on studying the existing system of agro culture in and to make sure that the peoples are getting quality fresh goods. This is also will produce:

- Less effort and less labour intensive, as the primary cost and focus primary on creating, managing, and running a secure quality food supply.
- Increasing number of buyers as individuals will find it easier and more convenient to buy goods.
- Easy management.

CHAPTER 2: LITERATURE SURVEY

The development of an online agriculture product purchase system is inspired by several existing studies and platforms that aim to improve market access and profitability for farmers. This literature survey reviews key contributions in the field of digital agriculture platforms, online marketplaces, and e-commerce solutions tailored for the agricultural sector.

1. Digital Platforms in Agriculture:

Studies on digital platforms in agriculture have highlighted the potential of Information and Communication Technology (ICT) in connecting farmers with markets. Platforms like e-Chou pal and Agri Bazaar have demonstrated how ICT can bridge the gap between rural farmers and urban markets, reducing dependency on traditional intermediaries. Research by the Food and Agriculture Organization (FAO) underscores that direct access to markets via digital platforms can significantly increase farmers' income and reduce post-harvest losses.

2. Online Marketplaces for Agricultural Products:

Online marketplaces, such as AgroTrade and Farm Link, have been studied for their role in providing farmers with direct access to buyers. These platforms focus on enabling farmers to list their produce and receive bids from buyers, which encourages competitive pricing and ensures better profit margins for the farmers.

3. E-Commerce and Payment Systems in Agriculture:

The adoption of e-commerce in agriculture has been transformative, particularly with the integration of secure payment gateways. Studies on platforms like Kisan Network and DeHaat show that seamless payment solutions are critical for gaining user trust and ensuring successful transactions. The literature emphasizes the importance of secure, user-friendly payment systems to enhance the adoption of digital marketplaces in rural areas.

The reviewed literature highlights the significant impact of digital platforms in transforming agricultural markets by directly connecting farmers with buyers. While there are challenges, including the need for reliable internet access, user education, and data security, the benefits of increased market access and reduced dependency on intermediaries are substantial. The proposed Agriculture Product Purchase System builds upon these insights to create a user-friendly, secure, and efficient platform that empowers farmers and facilitates direct transactions in the agricultural sector.

CHAPTER 3: SYSTEM ANALYSIS

System Analysis is a detailed study of the various operations performed by a system and their relationships within and outside of the system. Here the key question is- why all problems exist in the present system? What must be done to solve the problem? Analysis begins when a user or manager begins a study of the program using existing system. During analysis, data collected on the various files, decision points and transactions handled by the present system. A good analysis model should provide not only the mechanisms of problem understanding but also the frame work of the solution. Thus, it should be studied thoroughly by collecting data about the system.

3.1 EXISTING SYSTEM

In our existing system the recording of user's information is done manually, So taking more time for searching the information of the users. Another major disadvantage is that preparing the list of members that viewed any user's information takes more time.

The current agricultural marketplace largely depends on a traditional system where multiple intermediaries, such as wholesalers and distributors, play a central role in connecting farmers with buyers. This system often limits market access for farmers, especially those in rural areas, resulting in lower profit margins due to the significant share taken by intermediaries. Moreover, the lack of transparency in the supply chain makes it difficult for farmers to receive fair prices, while buyers have limited visibility into the origin, quality, and pricing of products. The system's inefficiencies, such as delayed payments and post-harvest losses, further exacerbate the challenges, with minimal use of digital technology to streamline processes.

3.2 PROPOSED SYSTEM

The proposed Agriculture Product Purchase System aims to revolutionize this process by providing farmers with direct access to a broader market through a digital platform, eliminating the need for intermediaries. The system enhances transparency by allowing users to view detailed product information, pricing, and transaction history, while a rating and feedback mechanism builds trust between farmers and buyers. The proposed system has:

1. The platform provides farmers with direct access to a broader market of buyers, eliminating the need for intermediaries.
2. The system promotes transparency by allowing both farmers and buyers to view product details, pricing, and transaction history.
3. The platform integrates a secure payment gateway, enabling seamless transactions between farmers and buyers.
4. The system is designed with an intuitive user interface that is accessible to users with varying levels of technological literacy.

CHAPTER 4: SYSTEM REQUIREMENT SPECIFICATION

4.1.1 SOFTWARE REQUIREMENT SPECIFICATION

1. Crop Management:-where it shows for all the buyers that availability of crops along with the price
2. Live Stock:-This page is dedicated for only displaying the availability of crops which is done by the former
3. online mode transaction:-This allow the user to make easy and secure online transaction payment

4.1.2 NON-FUNCTIONAL REQUIREMENTS

1. Easy to use:-where it requires simple knowledge to use this application not like only technical people can use it
2. Reduce Time and Cost:-It can reduce time of buying and selling the crops compare to offline and where the former directly selling the products which doesn't require of retailer
3. Security:-It provides for security for the user information by giving login process for already registered user if not it will ask for new user for registration

4.1.3 HARDWARE REQUIREMENTS

1. Processor: - Intel R core TM i3-1005 GI Pentium Core i5
2. Hard Disk:- 4GB
3. RAM: -8GB

4.1.4 SOFTWARE REQUIREMENTS

1. Front End: -HTML, CSS, Java script, Bootstrap
 2. Back End: -PHP, MYSQL
 3. Design constraint: -OS(windows 11)
- 4.2 Software Tools

Xampp:

- Apache:
(Application Server) Apache , often referred to as Server, is an open-source Java Servlet Container developed by the Apache Software Foundation.
- MySqlServer:
It handles large databases much faster than existing solutions.
It consists of multi-threaded SQL server that supports different back ends, several different client programs and libraries, administrative tools, and application programming interfaces (APIs)

CHAPTER 5 DESCRIPTION OF MODULES

5.1 System Architecture

The agriculture system you're working on seems to be a platform that connects farmers and buyers, allowing them to interact through registration, product listings, purchases, and transaction management. Below is a detailed description of the working flow:

1. User Registration (register.html & register.php)
 - Frontend (register.html): The registration page presents a form where new users (farmers or buyers) enter their details such as name, email, password, user type (farmer or buyer), and possibly more information like location, farm details, etc.
 - Backend (register.php): When the form is submitted, the data is sent to the server via POST request. The PHP script processes this data by validating it, checking for existing users with the same email, and then storing the data in the database. A success message is returned, or errors are displayed if registration fails.
2. User Login (login.html & login.php)
 - Frontend (login.html): The login page allows users to enter their email and password to access the system.
 - Backend (login.php): When the login form is submitted, the credentials are sent to the server for verification. The PHP script checks if the email exists and if the password matches. If successful, the user is redirected to their profile or dashboard page. If not, an error message is displayed.
3. User Profile Management (profile.html & profile.php)
 - Frontend (profile.html): After login, users are taken to their profile page where they can view and update their personal information, view past transactions, or list products (for farmers).
 - Backend (profile.php): This script retrieves user information from the database and updates any changes made by the user. It also pulls data like past transactions and product listings related to the user.
4. Product Listing (product.html & product.php) □
 - For Farmers:
 - Frontend (product.html): Farmers can list their products by filling out a form that includes details like product name, description, quantity, price, and images.
 - Backend (product.php): The submitted product details are sent to the server, where the PHP script saves the product information in the database and links it to the farmer's profile.
 - For Buyers:
 - Frontend (product.html): Buyers can browse through the products listed by various farmers. They can filter products based on criteria like price, location, or product type.
 - Backend (product.php): The PHP script retrieves product listings from the database and displays them on the page.
5. Product Purchase (buy.html & buy.php)
 - Frontend (buy.html): Once a buyer selects a product, they are taken to the purchase page where they can review the product details, confirm the quantity, and proceed to buy.

- Backend (buy.php): This script processes the purchase request by validating stock availability, reducing the product quantity in the database, and creating a transaction record. A confirmation message is displayed after the purchase is successful.
6. Transaction Management (transactions.html & transactions.php)
- Frontend (transactions.html): Users (both farmers and buyers) can view their transaction history on this page. Farmers see the products they have sold, while buyers see the products they have purchased.
 - Backend (transactions.php): The PHP script queries the database for transaction records related to the logged-in user and displays them on the page.
7. Styling and Interactivity (styles.css & script.js)
- CSS (styles.css): This file is used to style all the HTML pages, ensuring a consistent and userfriendly interface.
 - JavaScript (script.js): JavaScript is used to add interactivity to the pages, such as form validation, dynamic content updates, and handling user events like clicks or form submissions.
8. Database Interaction
- The system uses a database (likely MySQL or a similar relational database) to store user information, product listings, transaction records, and other necessary data.
 - PHP scripts interact with the database using SQL queries to fetch, insert, update, or delete records as needed for the various operations described above.

5.2 DESIGN

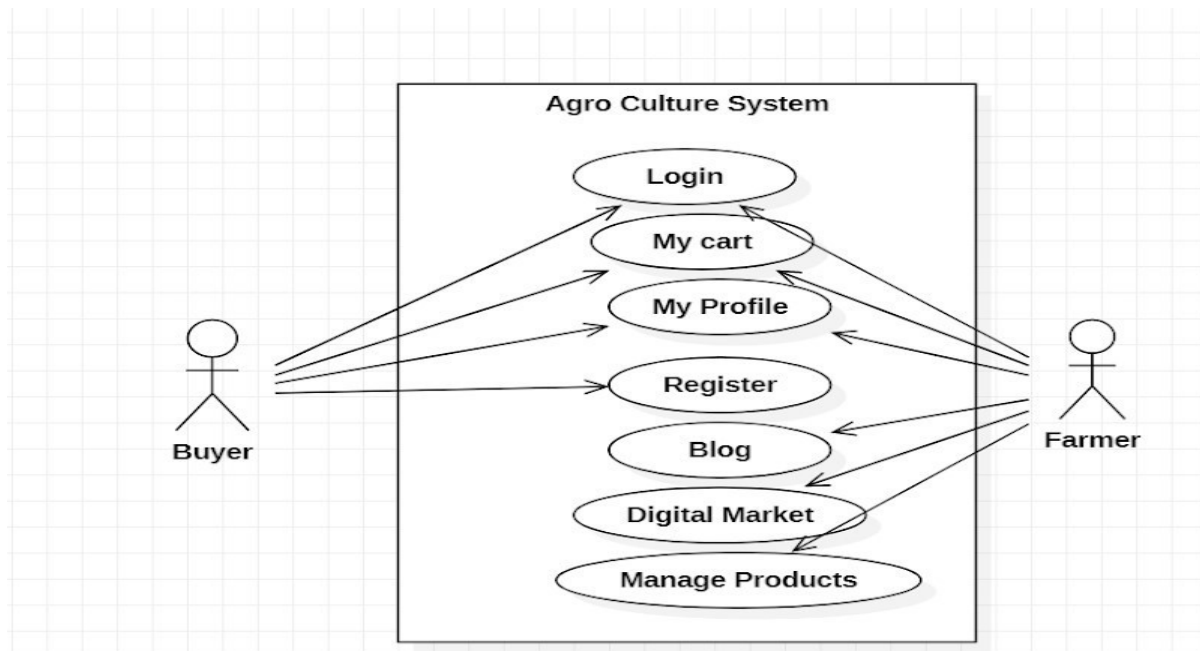


Fig 5.2.1 Use Case Diagram

CHAPTER 6 RESULTS AND DISCUSSION

6.1 IMPLEMENTATION

Here's a concise version of the algorithmic flow for the agriculture system:

1. User Registration

1. Input: User provides name, email, password, and user type.
2. Check: Validate input; check if email exists.
3. Process: Hash password and insert data into the database.
4. Output: Success or error message.

2. User Login

1. Input: User provides email and password.
2. Check: Validate input; verify email and password.
3. Process: Authenticate user and create session.
4. Output: Redirect to profile or display error.

3. Profile Management

1. Input: User views or updates their profile.
2. Check: Validate session and input.
3. Process: Retrieve or update user data in the database.
4. Output: Display profile information.

4. Product Listing (Farmer)

1. Input: Farmer enters product details.
 2. Check: Validate input.
 3. Process: Insert product into the database.
 4. Output: Success or error message.
5. Product Browsing (Buyer)
1. Input: Buyer browses products.
 2. Check: Apply filters if needed.
 3. Process: Retrieve products from the database.
 4. Output: Display product list.

6. Product Purchase

1. Input: Buyer selects product and quantity.
2. Check: Validate stock availability.
3. Process: Update product quantity; log transaction.
4. Output: Display purchase confirmation.

7. Transaction Management

1. Input: User views transaction history.
2. Check: Validate session.
3. Process: Retrieve transactions from the database.
4. Output: Display transaction history.

6.2 SCREENSHOTS OF OUTPUT

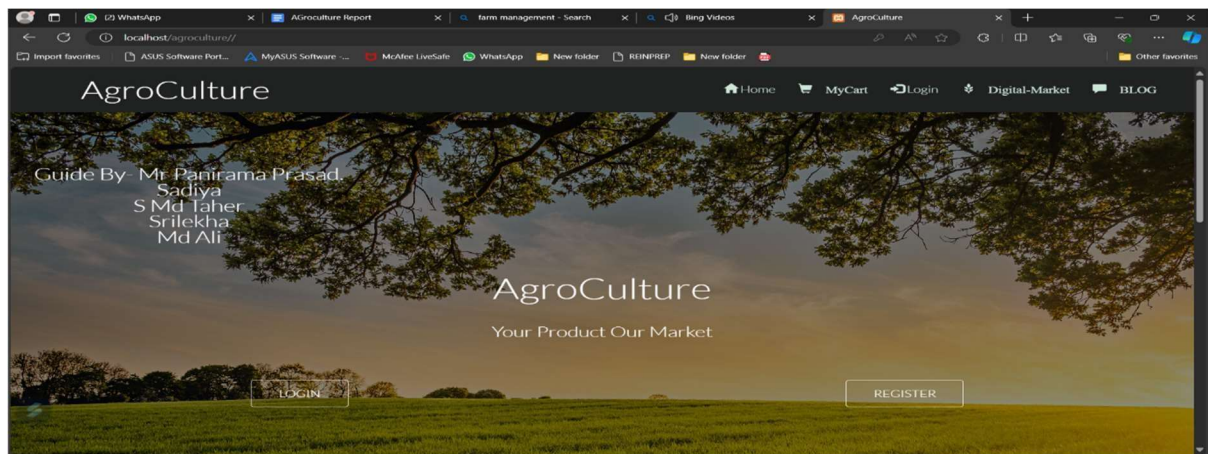


Fig 6.2.1 Welcome Page

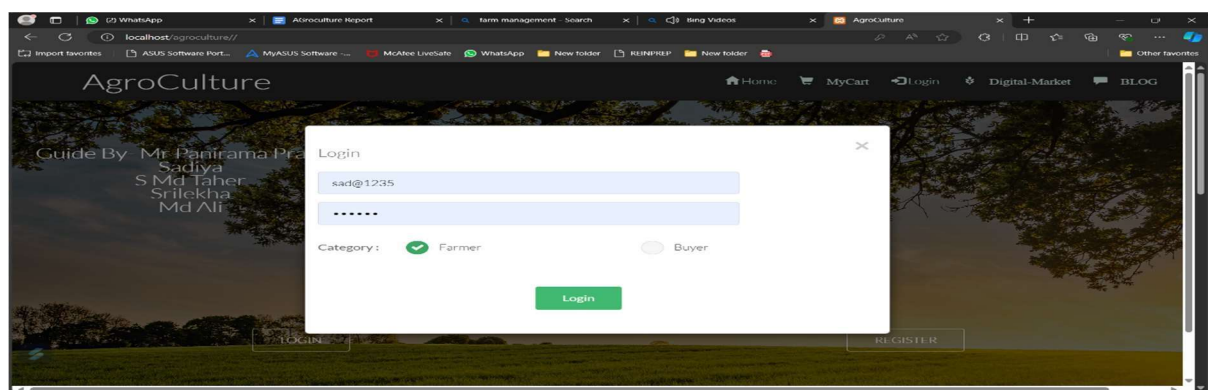


Fig 6.2.2 Login Page

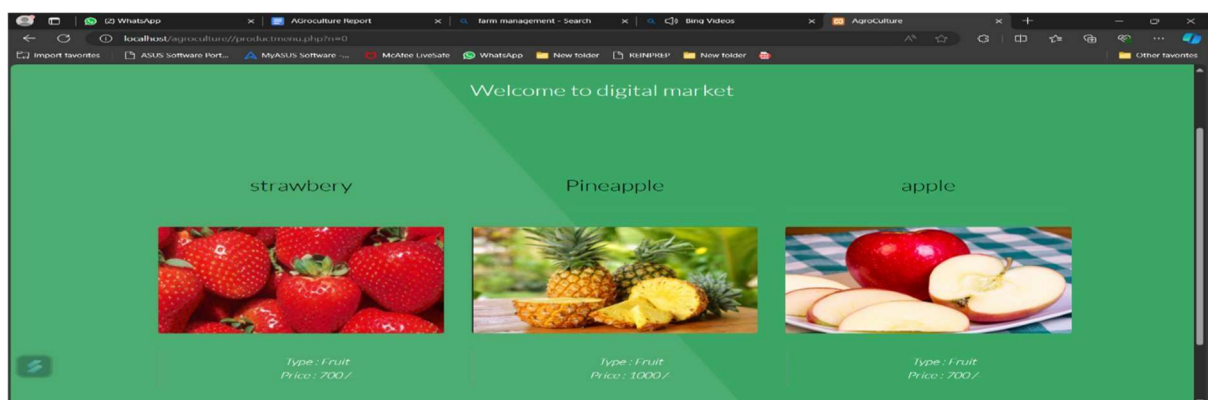
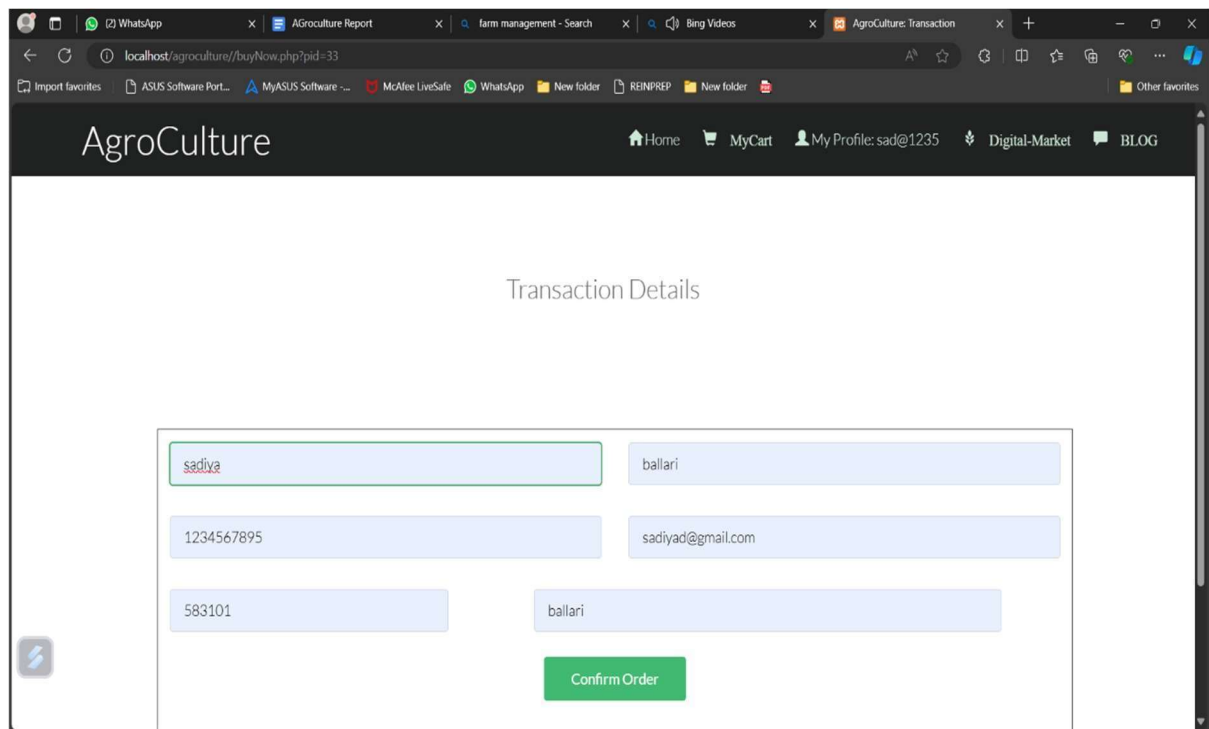


Fig 6.2.3 Product Display Page



AgroCulture

Home MyCart My Profile: sad@1235 Digital-Market BLOG

Transaction Details

sadiya ballari

1234567895 sadiyad@gmail.com

583101 ballari

Confirm Order

Fig 6.2.4 Transaction Page

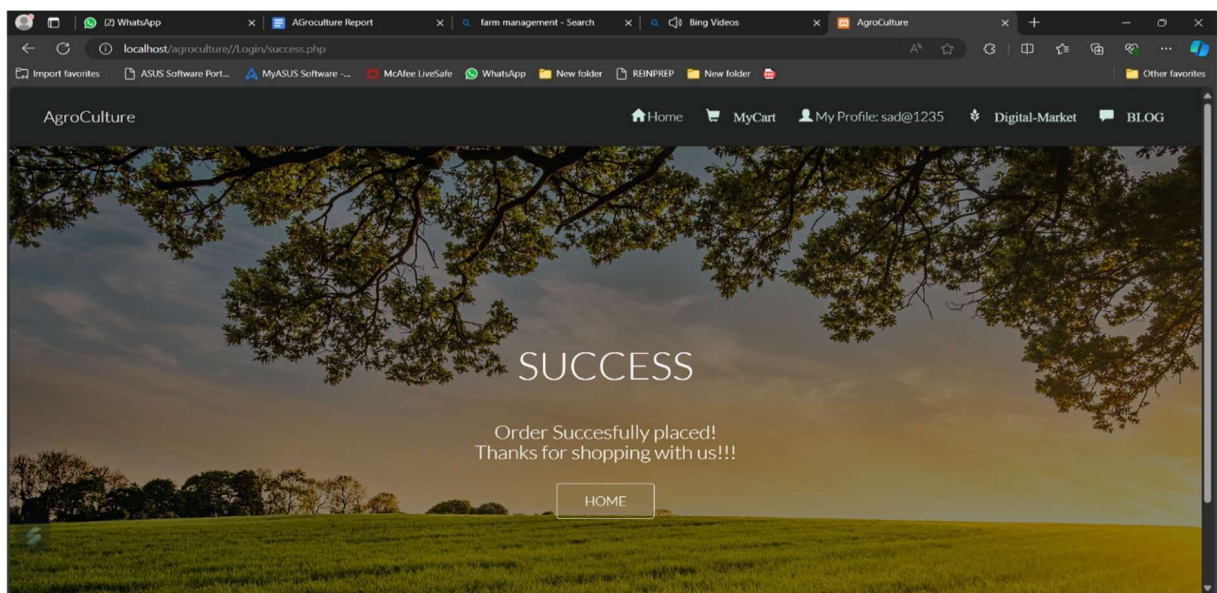


Fig 6.2.5 Success Page

CHAPTER 7 TESTING

Testing is evaluation of the software against requirements gathered from users and system specifications. Testing identifies important defects, flaws, or an error in the application code that must be fixed. It also assesses the feature of a system. Testing assesses the quality of the product.

7.1 TYPES OF TESTING

7.1.1. Unit Testing

Unit testing refers to the testing certain functions and areas of the code. It gives the ability to verify that all the functions work as expected. Eventually, it helps to identify failures in the algorithms as well a logic to help improve the quality of the code that composes a certain function.

7.1.2 Integration Testing

Integration testing is basically a logical extension of unit testing. In simple words, two tested units are combined into a component and the interface between them is tested. It identifies problems that occur when different units are combined. The different modules of this project have undergone integration testing while being merged.

7.1.3 System Testing

System testing tests the behavior of whole system as defined by the scope of the development project. It might include tests based on risks as well as requirement specifications, business process, use cases or other high level descriptions of system behavior, interactions with the operating systems and system resources. It is most often the final test performed to verify that the system meets the specification and its objectives. System testing has been performed at the completion of each feature and is still taking place to make improvements on the existing system.

7.2 TEST CASES

Here are some test cases for each component of the agriculture system:

1. User Registration

- Test Case 1: Successful Registration
 - Input: Name: "John Doe", Email: "john@example.com", Password: "Password123", User Type: "Farmer"
 - Expected Result: Registration successful message; user data stored in the database.
- Test Case 2: Registration with Existing Email
 - Input: Name: "Jane Doe", Email: "john@example.com", Password: "Password456", User Type: "Buyer"
 - Expected Result: Error message indicating the email is already registered.
- Test Case 3: Missing Required Field

- Input: Name: "John Doe", Email: "", Password: "Password123", User Type: "Farmer"

Expected Result: Error message indicating all fields must be filled.

2. User Login

- Test Case 1: Successful Login
- Input: Email: "john@example.com", Password: "Password123"
- Expected Result: User redirected to profile page; session created.
- Test Case 2: Login with Incorrect Password
- Input: Email: "john@example.com", Password: "WrongPassword"
- Expected Result: Error message indicating incorrect password.
- Test Case 3: Login with Non-existent Email
- Input: Email: "nonexistent@example.com", Password: "Password123"

Expected Result: Error message indicating the email is not registered.

3. Profile Management

- Test Case 1: View Profile Information

Input: User is logged in.

- Expected Result: User's profile information is displayed.

- Test Case 2: Update Profile Information

Input: User updates name to "John Smith".

- Expected Result: Profile information is updated and displayed with the new name.

- Test Case 3: Access Profile Without Login

Input: User is not logged in.

- Expected Result: Redirect to login page or error message indicating access is denied.

4. Product Listing (Farmer)

- Test Case 1: Successful Product Listing

Input: Name: "Apples", Description: "Fresh apples", Quantity: 100, Price: 2.50 ◦

Expected Result: Product is listed and stored in the database.

CONCLUSION

The agriculture system is designed to support various functionalities, including user registration, login, profile management, product listing, browsing, purchasing, and transaction management. Each component of the system involves specific processes such as input validation, database interactions, and security protocols to ensure smooth and secure operations. Comprehensive testing through structured test cases helps evaluate the system's effectiveness across typical user scenarios, ensuring that it performs as expected while handling errors and maintaining security. This approach ensures a seamless experience for users—whether they are farmers or buyers—while safeguarding data integrity and system security throughout the entire process.

REFERENCES

1. www.google.com
2. <https://www.youtube.com/watch?si=EJQoBaL117N97wbN&v=MK5R0yxUac&feature=youtu.be>
3. www.geeksforgeeks.com
4. www.w3schools.com
5. MySQL scratch course
6. <https://getliner.com/main>
7. <https://libgen.is/>
8. https://docs.google.com/document/u/0/d/1rmDI_d0A6kbyvMx7N7Pw3pThW8ue90q1/mobilenbasic
9. MySql Cookbook by Paul DuBois
10. Modern PHP by Josh Lockhart