**E - FARMER**

**A PROJECT REPORT**

***Submitted by***

**Mr. DILEEP V - 20201CSE0691**

**Mr. YASHAS M - 20201CSE0700**

**Mr. D SHIVASHANKAR - 20201CSE0711**

***Under the guidance of,***

**Dr. JOSEPH MICHAEL JERARD V**

**Professor**

***in partial fulfillment for the award of the degree of***

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

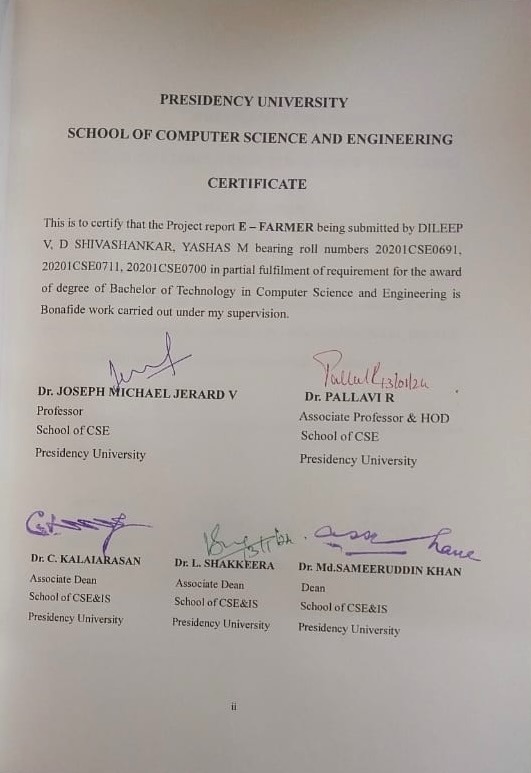
**At**

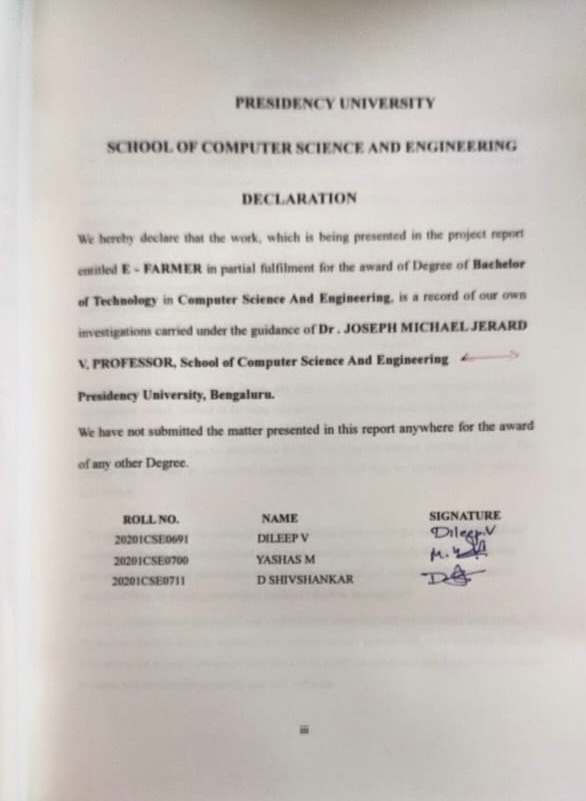
****

**PRESIDENCY UNIVERSITY**

**BENGALURU**

**JANUARY 2024**





**ABSTRACT**

The goal of the E-Agriculture System is to automate the current manual system with the aid of fully functional computer software and computerized equipment, meeting their needs and enabling the storage of their important data and information for an extended amount of time with simple manipulation and access. The necessary gear and software are readily available and simple to use.

The e-agriculture system mentioned above has the potential to produce a rapid, safe, dependable, and error-free management system. Instead of focusing on maintaining records, it might help the user focus on other activities. As a result, it will aid organizations in making better use of their resources. Computerized records can be maintained by the organization without duplicate entries. This implies that irrelevant information need not divert attention.

E-Agriculture System, as stated above, can lead to error free, secure, dependable and rapid management system. Instead of focusing on maintaining records, it might help the user focus on other activities. As a result, it will aid organizations in making better use of their resources. Computerized records can be maintained by the organization without duplicate entries. This implies that in order to access the knowledge, one need not be sidetracked by irrelevant information.

The goal is to automate the company's current manual system with the use of fully functional computer software and computerized equipment, meeting their needs in the process, so that their important data and information may be easily accessed and manipulated for extended periods of time. In essence, the project explains effective management

By using computerized equipment and sophisticated computer software to meet their needs, the Agriculture System aims to automate the current manual system and store valuable data and information for a longer amount of time that can be easily accessed and manipulated. It is simple to obtain and operate the necessary gear and software.

**ACKNOWLEDGEMENT**

First of all, we indebted to the GOD ALMIGHTY for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected **Dr. Md. Sameeruddin Khan**, Dean, School of Computer Science and Engineering & Information Science, Presidency University for getting us permission to undergo the project.

We record our heartfelt gratitude to our beloved Associate Deans **Dr. Kalaiarasan C** **and Dr. Shakkeera L**, School of Computer Science and Engineering & Information Science, Presidency University and **Dr. Pallavi R**, Head of the Department, School of Computer Science and Engineering & Information Science, Presidency University for rendering timely help for the successful completion of this project.

We are greatly indebted to our guide **Dr. Joseph Michael Jerard V**, Professor, School of Computer Science and Engineering & Information Science, Presidency University for his inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the project work.

We would like to convey our gratitude and heartfelt thanks to the University Project-II Coordinators **Dr. Sanjeev P Kaulgud**, **Dr. Mrutyunjaya MS** and also the department Project Coordinators **Dr. Zia Ur Rahman, Mr.Peniel John Whistely**.

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

**DILEEP V**

**YASHAS M**

**D SHIVSHANKAR**

# **LIST OF FIGURES**

**Sl. No. Figure Name Caption Page No.**

1 Figure 4.1 Agile Model 6

2 Figure 4.2 Use Case Diagram 7

3 Figure 4.3 Flow Chart 8

4 Figure 6.1 Scheme of use cases for admin 12

5 Figure 6.2 Use Case Diagram of User 13

6 Figure 6.3 E-R Diagram 14

7 Figure 6.4 Zerotaseme DFD Diagram 15

8 Figure 6.5 First Level DFD Diagram 16

9 Figure 7.1 Gantt Chart 17

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO.** | **TITLE** | **PAGE NO** |
| **1.** | **INTRODUCTION** | 1 |
|  | 1.1 PROBLEM STATEMENT | 1 |
|  | 1.2 SCOPE OF PROJECT | 1 |
| **2.** | **LITERATURE SURVEY** | 2-3 |
| **3.** | **RESEARCH GAPS OF EXISTING METHODS** | 4 |
| **4.** | **PROPOSED METHODOLOGY** | 5 |
|  | 4.1 DATA COLLECTION INSTRUMENT | 5 |
|  | 4.1.1 AGILE MODEL | 6 |
|  | 4.1.2 USE – CASE DIAGRAM | 7-8 |
| **5.** | **OBJECTIVES** | 9 |
| **6.** | **SYSTEM DESIGN AND IMPLEMENTATION** | 10 |
|  | 6.1 HARDWARE & SOFTWARE REQUIREMENTS | 10 |
|  | 6.2 FUNCTIONAL REQUIREMENTS | 11 |
|  | 6.3 ANALYSIS AND DESIGN | 12 |
|  | 6.4 USE – CASE DIAGRAM OF USER | 13 |
|  | 6.5 ER – DIAGRAM | 14 |
|  | 6.6 ZERO LEVEL DFD DIAGRAM | 15 |
|  | 6.7 FIRST LEVEL DFD DIAGRAM | 16 |
| **7.** | **TIMELINE FOR EXECUTION OF PROJECT** | 17 |
|  | 7.1 GANTT CHART | 17 |
| **8.** | **OUTCOMES** | 18-19 |
| **9.** | **RESULTS AND DISCUSSION** | 20 |
| 10. | **CONCLUSION** | 21 |
| 11. | **REFERENCES** | 22 |
| 12. | **APPENDIX – A** | 23 |
|  | 1. PSEUDOCODE | 23-31 |
| 13. | **APPENDIX – B** | 32 |
|  | 1. Screenshot | 32-53 |
| 14. | **APPENDIX – C** | 54 |
|  | 1. ENCLOSURES | 54-58 |
| 15. | **SDG** | 59 |

**CHAPTER-1**

**INTRODUCTION**

**Problem Statement**

The current globalization of generation and supply of rural items has emphasized the got to keep up security, effectiveness and competence all through the nourishment and rural supply chain. Compelling traceability arrangements are direly required to guarantee item security due to expanding nourishment security and debasement dangers. Our conventional method of recording information on papers and registers is the foundation of the outdated systems that currently power the agricultural web portal. It takes a lot of work to operate and administer a sizable web portal with paper records.

The Agricultural Web Portal encountered the following issues.

* It was time-consuming and stressful.
* It was difficult to establish a secure database.
* It was difficult to maintain record books for its effective use.
* There were problems with the effective communication and interaction between a farmer and the shopkeeper.

**Scope of the project**

The following are the opportunities for building an agricultural online portal:

* This approach assists in compiling the ideal management in the specifics. With the aid of this project, data collecting and storage will be easy, rational, and efficient. It also aids in carrying out the several tasks that the Agricultural Web Portal requires.
* The program is developed using Java technology.

Both administrators and regular clients will be able to utilize the system efficiently.

* Through this platform, farmers can sell their food directly to consumers and obtain information on goods that are beneficial for their operations.

**CHAPTER-2**

**LITERATURE SURVEY**

**The Role of Online Platforms in Agricultural Markets. - Marelli, L., & Whittaker, G. (2018).**

This paper highlights the potential of online platforms to revolutionize the process of selling farms through bidding mechanisms. This innovative approach introduces a level of transparency and competitiveness previously unseen is traditional farm transactions, leading to fair market values and empowering both sellers and buyers.

**An Analysis of Online Auctions in the Agricultural Sector. - Marelli, L., & Whittaker, G. (2018).**

This paper emphasizes that the adoption of online auction platforms in the agricultural sector holds great promise for revolutionizing the way agricultural products and land are traded. These platforms contribute to reducing transaction costs, widening market access, and promoting equitable trade practices.

**Online B2B auction system for agricultural products. -Nan, Y., & Ding, J. (2017).**

This article highlights the potential of online stages to revolutionize the cultivate deals handle through offering components. This imaginative approach brings straightforwardness and competitiveness not already seen in conventional rural businesses, coming about in reasonable showcase values and strengthening of both dealers and buyers.

**Food and Agriculture Organization of the United Nations (FAO)- "Agriculture Strategy Guide".**

This direction recognizes the significance of adjusting e-agriculture methodologies to the oneof-a-kind needs and settings of person districts and nations. It energizes flexibility and advancement within the appropriation of computerized arrangements, reflecting FAO's commitment to advancing maintainable and context-specific rural hones.

**E-Agriculture in India: An Exploratory Study. - Kaur, J. and Kaur, M.**

**(2020).**

Arrangements in India. In the event that utilized viably, these innovations can act as catalysts for rustic advancement, destitution diminishment and nourishment security. Creators and#039; work emphasize the significance This paper. emphasizes the require for proceeded endeavours to advance and create e-agriculture of key arrangement and venture in advanced foundation to guarantee that the benefits of e-agriculture reach all corners of the nation, particularly in farther and underserved regions.

**CHAPTER-3**

**RESEARCH GAPS OF EXISTING METHODS**

Manual forms are known to be much more labour seriously, time devouring, exact and proficient than robotized frameworks. Thus, the following are a few drawbacks in the previous 1. Inaccurate

1. Inefficient
2. A ton of paperwork
3. Unfriendly surroundings
4. It's hard to maintain ancient records
5. More manpower.
6. Uses a significant amount of payload.
7. Higher officials have no direct involvement.

**Modules of E-Agriculture System**:

* + Crops Management Module: Used for managing the Crops details.
  + Customer Module: Used for managing the details of Customer
  + System User Module: Used for managing the details of System User
  + Equipment Management Module: Used for managing the information and details of the Equipment.
  + Insecticides Module: Used for managing the Insecticides details
  + Pesticides Module: Used for managing the Pesticides information
  + Login Module: Used for managing the login details
  + Users Module: Used for managing the users of the system

**CHAPTER-4**

**PROPOSED METHODOLOGY**

Electronic Agrarian Administration Framework (EAMS) may be a arranged framework. This web-based program is outlined to supplant an error-prone and questionable manual cultivate information framework. Facilitated on Internet servers, the EAMS application can be gotten to from both computers and portable gadgets. The framework collects and forms data almost ranchers, agrarian items, buyers and providers, such as names, addresses, phone numbers and mail addresses.

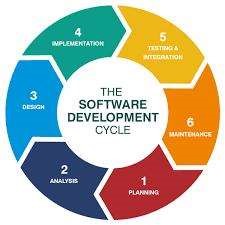
**Data Collection Instrument**

The device chosen for information collection could be a chat bar The reason of the Electronic Agribusiness Meet is to assemble data from different agriculturists and wholesalers of cultivate items. The arranged framework is called the Administration Framework (EAMS). This web-based application is implied to supplant the labour-intensive and error-prone manual framework utilized to handle cultivate information.

The EAMS application is accessible on both computers and portable gadgets because it is facilitated on an online server. The framework collects and forms data almost ranchers, agrarian items, buyers and providers, counting names, addresses, phone numbers and e-mail addresses. utilize inquire about strategy. The EAMS application is accessible on both computers and versatile gadgets because it is facilitated on a web server. The framework collects and forms data around ranchers, rural items, buyers and providers, counting names, addresses, phone numbers and e-mail addresses. utilize investigate strategy.

**Agile Model**

Agile is a software development life cycle model and is used in EAMS development due to its rapid and flexible response to change.



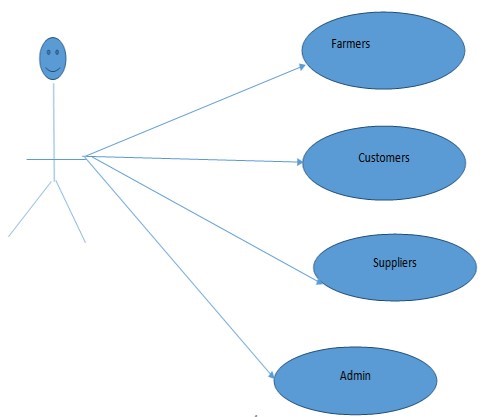
**Fig 4.1: Agile Methodology**

**Advantages of Agile Model**

1. Requirements changes can be easily implemented in the system.
2. Compared to the waterfall model, starting software development requires only a few planning steps.

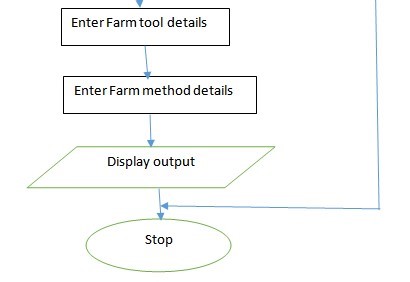
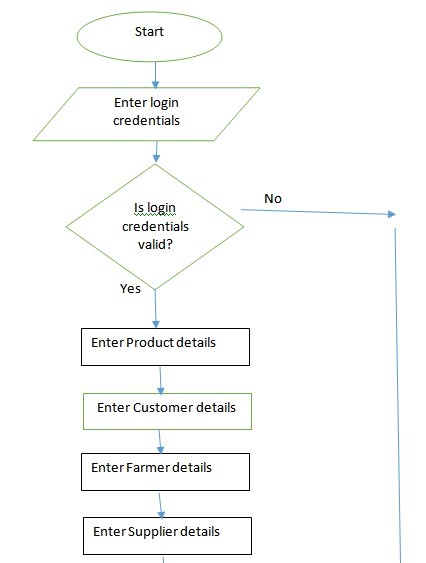
**Use Case Diagram**

The use case diagram below illustrates the interaction between the proposed system and the users.



**Fig 4.2: Use Case Diagram**

A flowchart is used to design the proposed EAMS. The flowchart below shows a logical and pictorial representation of the proposed system.



**Fig 4.3: Flow Chart**

**CHAPTER-5**

**OBJECTIVES**

The aim of this project is to provide a technologically-based approach to augment transparency and strengthen the customer-farmer relationship. the capability for farmers to place orders for fertilizer and seeds. the flexibility to pay bills in any method, whether offline or online.

Workflow is continuously tracked via the site.

contacting farmers by phone and encrypted messaging.

Permit customers to ask or bid for the farm's produce.

Improving data security is essential, which emphasizes the necessity of putting strong security measures in place, especially when it comes to user authentication and authorization. The purpose of this strategic endeavor is to strengthen the protection of private data stored in the MySQL database so that it is not susceptible to hacking or other unwanted access.

There are several opportunities to create an online site for agriculture:

* Agriculture: This method helps to assemble the best management in the details. This project will make gathering and storing data simple, logical, and effective. It also helps with completing the various activities needed by the Agricultural Web Portal. In order to facilitate the maintenance of records, documentation, and the six inventories on the portal—all of which require proper record-keeping and management—this management system was developed.
* Java technology is utilized in the program's development.

The system will be effective for regular clients as well as administrators to use. Farmers can use this platform to sell their produce directly to customers and get information about products that can help them.

**CHAPTER-6**

**SYSTEM DESIGN & IMPLEMENTATION**

**Hardware Requirements**

Processor - I3/Intel Processor

Hard Drive - 160 GB

Keyboard - Standard Windows Keyboard

Mouse - Two or Three Button Mouse

RAM - 8 GB

**Software requirements**

Operating system: Windows 7/8/10

Server-side scripting: HTML, CSS, Bootstrap and JS

Programming language: Java Server implementation: Apache Tomcat Database: MySQL.

## **FUNCTIONAL REQUIREMENTS**

**A. INPUT/OUTPUT**

1. The system must have a form to receive customer data.
2. The system must have a form to receive information from buyers.
3. The system displays the order information.
4. The system should provide a search function for customer name, order placed, order date, order shipped date, event date, etc.
5. The system should provide an option to change the address/name.
6. The system must record information about the order/shipment or order processing, i.e. order status.

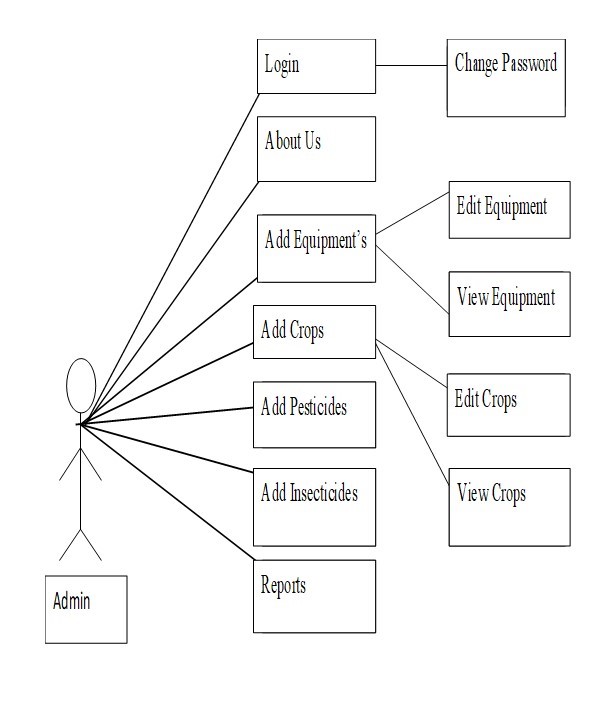
**B. PROCESSING**

1. The system should automatically generate an order ID.
2. The system should report a pending order and make changes when the order is shipped.

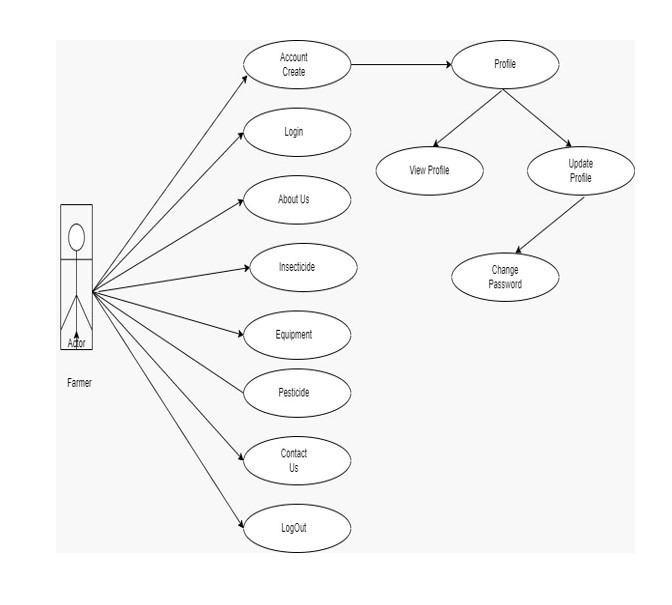
**C. ERROR HANDLING**

1. Errors in duplicate master keys should be reported.
2. Should report all "out of range" values in numeric fields.
3. Any data types that do not match any of the form fields must be reported in the report.

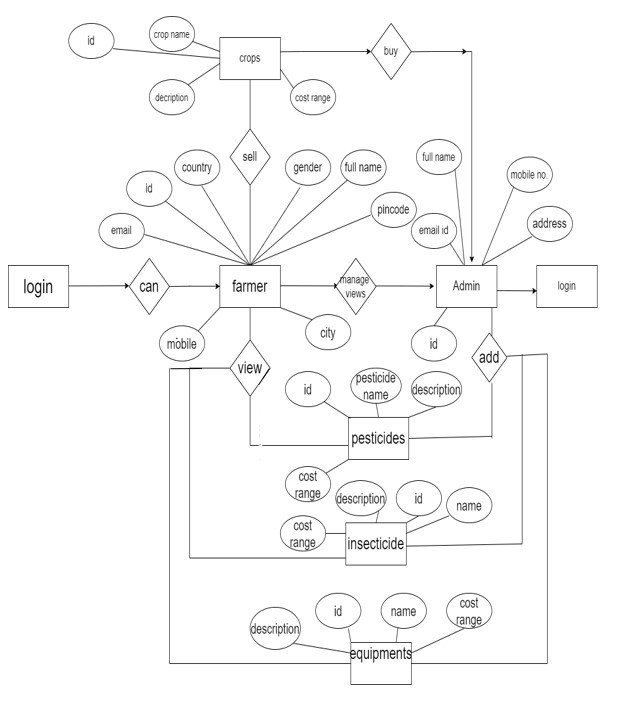
**ANALYSIS AND DESIGN**



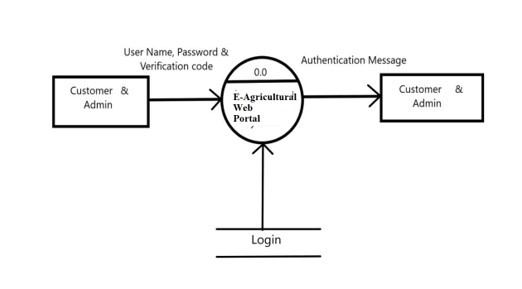
**Fig 6.1: SCHEME OF USE CASES FOR ADMIN.**



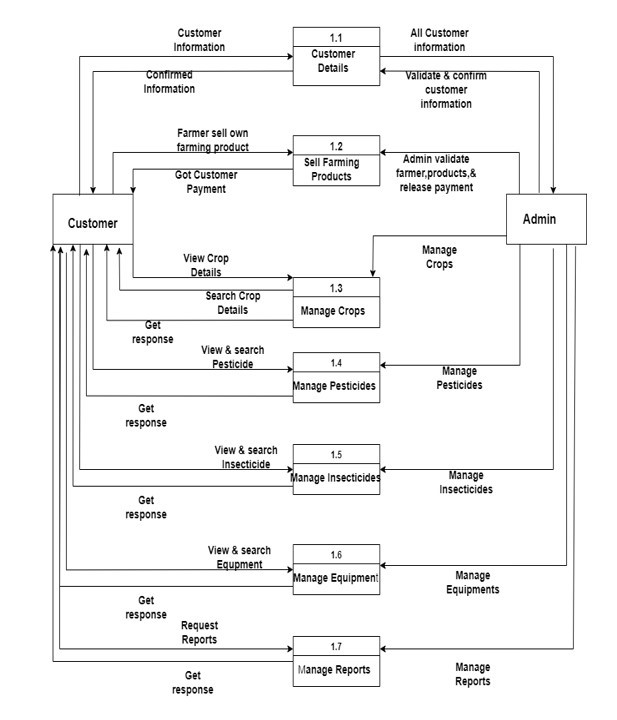
**Fig 6.2: USER USE CASE DIAGRAM**



**Fig 6.3: E-R SCHEMA**



**Fig 6.4: ZEROTASEME DFD DIAGRAM.**

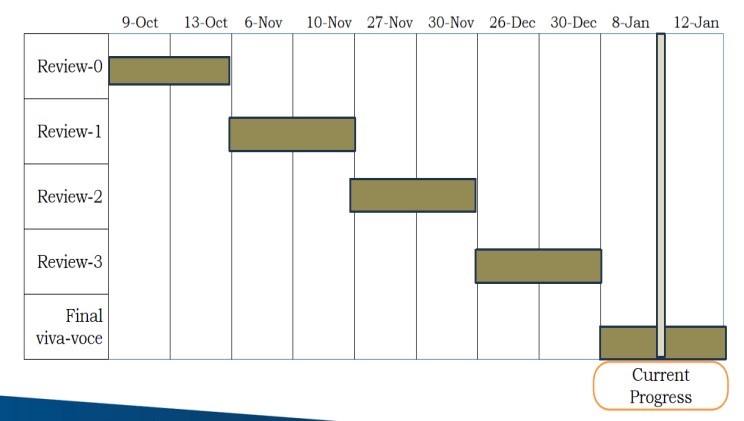


**Fig 6.5: FIRST LEVEL DFD DIAGRAM.**

**CHAPTER-7**

**TIMELINE FOR EXECUTION OF PROJECT**

**(GANTT CHART)**



**Fig 7.1: Gantt Chart**

**CHAPTER-8**

**OUTCOMES**

**1. Marketplace for Agricultural Products:**

Connect farmers with buyers, retailers, or wholesalers. Allow farmers to showcase and sell their produce online.

**2. Crop Management Tools:**

Provide tools for farmers to manage their crops efficiently.

Offer planting calendars, weather forecasts, and pest management advice.

**3. Knowledge Hub:**

Include a resource section with articles, guides, and tutorials for farmers.

Facilitate knowledge sharing among the farming community.

**4. Online Farming Communities:**

Create forums or discussion boards for farmers to share experiences and ask questions.

Foster a sense of community among farmers.

**5. Supply Chain Management:**

Streamline the supply chain by connecting farmers with logistics and transportation services. Ensure efficient delivery of agricultural products to the market.

**6. Financial Tools:**

Integrate tools for financial planning, budgeting, and tracking expenses.

Provide access to loans or financial support for farmers.

**7. Weather and Climate Information:**

Include real-time weather updates and climate information to help farmers make

informed decisions.

**8. Precision Farming Support:**

Integrate technologies like IoT or AI to support precision farming practices.

Offer tools for data analysis and decision-making.

**9. Government Schemes and Policies:**

Provide information on government agricultural schemes, subsidies, and policies.

Assist farmers in accessing relevant support programs.

**10. Mobile Accessibility:**

Ensure that the website is mobile-friendly for easy access in rural areas with limited

internet connectivity.

**11. Education and Training:**

Offer online courses or training programs for farmers to enhance their skills.

Provide certifications for completing courses.

**12. Feedback and Reviews:**

Allow farmers to leave reviews and feedback for products, services, or equipment.

Build a reputation system to help farmers make informed choices.

**CHAPTER-9**

**RESULTS AND DISCUSSIONS**

In short, the scope and future of the project is focused on recording the following information:

* We can provide advanced farm management system software with additional capabilities.
* To make the platform accessible to everyone, we want to put it on web servers.
* Build a master-slave database structure to reduce database query overhead.

• Integrate multiple load balancers to distribute system load.

* Implement a backup system that regularly backs up the database and code base to different servers.

The improvements listed above can be made to make this project more workable and useful. We can track the catch and equipment here.

Our initiative is merely a modest attempt to meet their needs for project management. A lot of user-friendly coding has also been implemented. This bundle will show to be quite effective in meeting all of the school's requirements. The goal of software planning is to give the manager a framework that, at the start of the software project, allows them to produce realistic estimations in a constrained amount of time. This framework should be updated on a regular basis as the project moves forward.

**CHAPTER-10**

**CONCLUSION**

The goal of this project was to create a web application that allows users to buy products from farmers. Through this project, we gained useful knowledge and practical experience on various topics including creating web pages using JSP servlets, Bootstrap, CSS and HTML databases using MySQL. Every part of the system is secure. In addition, the project increased our awareness of the software development life cycle and its development phases.

Our initiative is only a modest attempt to fulfil their project management needs. A lot of user-friendly coding has also been implemented. This package seems to be effective enough to meet all the requirements of school. The purpose of software design is to provide managers with a framework within which they can make decisions.

An explanation of the project's history, setting, and relationship to previous work in the field.

made a declaration outlining the project's goals and objectives.

The description of Purpose, Scope, and applicability.

We specify the issue that the initiative is focusing on.

We outline the prerequisite. system specifications and the possible actions related to these items.

After gaining an understanding of the problem domain, we create a system model that outlines the operations that may be carried out on the system.

We went into great detail about the features and functionality, including screen layouts.

We created the system's user interface and addressed security concerns.

Ultimately, test cases are used to guide the implementation and testing of the system.

**REFERENCES**

[1] Construction of E-commerce Platform System for Targeted Poverty Alleviation IEEE 2020.

[2] E-Commerce Application for the farmers IEEE 2018.

[3] Krishi Portal Web Based Farmer Help Assistance International Journal of Advanced Science and Technology Vol. 29, No. 6, (2020), pp. 4783 – 4786.

[4] Design of Web Portal for E-Trading for Farmers’. Swapnil International Journal on Future Revolution in Computer Science & Communication Engineering IJFRCSCE March 2018, Available @ http://www.ijfrcsce.org.

[5] “FARMER TO CUSTOMER E-TRADE ’’International Journal of Latest Trends in Engineering and Technology Vol. (13) Issue(3), pp.050-056 DOI: http://dx.doi.org/

10.21172/1.133.0 eISSN:2278-621.

[6] Smart E-Marketing in Agricultural Products International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Published by, www.ijert.org ICRADL – 2021.

[7] L. A. Tawalbeh, R. Mehmood, E. Benkhlifa, and H. Song, ‘‘Mobile cloud computing model and big data analysis for healthcare applications,’’ IEEE Access, vol. 4, pp. 6171–6180, 2016.

[8] S. M. R. Islam, D. Kwak, M. H. Kabir, M. Hossain, and K.-S. Kwak, ‘‘The Internet of Things for health care: A comprehensive survey,’’ IEEE Access, vol. 3, pp. 678–708, Jun. 2015.

[9] A. Bahga and V. K. Madisetti, ‘‘A cloud-based approach for interoperable electronic health records (EHRs),’’ IEEE J. Biomed. Health Inform., vol. 17, no. 5, pp. 894–906, Sep. 2013.

[10] E. AbuKhousa, N. Mohamed, and J. Al-Jaroodi, ‘‘e-Health cloud: Opportunities

**APPENDIX-A**

**PSUEDOCODE**

**ADMIN LOGIN**

package com.admin; import java.io.IOException; import java.sql.Connection; import java.sql.ResultSet; import java.sql.Statement; import java.util.Random; import java.util.UUID; import javax.servlet.ServletException; import javax.servlet.annotation.WebServlet; import javax.servlet.http.HttpServlet; import javax.servlet.http.HttpServletRequest; import javax.servlet.http.HttpServletResponse; import javax.servlet.http.HttpSession; import com.connection.DatabaseConnection;

/\*\*

\* Servlet implementation class AdminLogin

\*/

@WebServlet("/AdminLogin") public class AdminLogin extends HttpServlet { protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

try {

String uname = request.getParameter("uname");

System.out.println("Admin Name "+uname);

String pass = request.getParameter("upass");

System.out.println("Admin Password "+uname);

String vercode = request.getParameter("vercode");

String captchaDB = null;

HttpSession hs = request.getSession();

String tokens = UUID.randomUUID().toString();

Random random = new Random();

int newRandomCaptcha = random.nextInt(9000) + 10000;

Connection con = DatabaseConnection.getConnection();

Statement st = con.createStatement();

ResultSet captchResultSet =

DatabaseConnection.getResultFromSqlQuery("select \* from tblcaptcha");

if (captchResultSet.next()) {

captchaDB = captchResultSet.getString(1);

}

if (captchaDB.equals(vercode)) {

ResultSet resultset = st.executeQuery("select \* from tbladmin

where uname='" + uname + "' AND upass='" + pass + "'");

if (resultset.next()) {

hs.setAttribute("uname",resultset.getString("uname"));

hs.setAttribute("email", resultset.getString("email"));

int update =

DatabaseConnection.insertUpdateFromSqlQuery("update tblcaptcha set captcha='"+ newRandomCaptcha + "'");

response.sendRedirect("admin-

dashboard.jsp?\_tokens='" + tokens + "'");

} else {

String message = "You have enter wrong credentials"; hs.setAttribute("credential", message);

response.sendRedirect("admin-login.jsp");

int update =

DatabaseConnection.insertUpdateFromSqlQuery("update tblcaptcha set captcha='"+ newRandomCaptcha + "'");

}

} else {

String message = "You have enter invalid verification code"; hs.setAttribute("verificationCode", message);

response.sendRedirect("admin-login.jsp");

int update =

DatabaseConnection.insertUpdateFromSqlQuery("update tblcaptcha set captcha='"+ newRandomCaptcha + "'");

}

} catch (Exception e) {

// TODO: handle exception

}

}

}

**BUYER ACCOUNT**

package com.buyer; import java.io.IOException; import javax.servlet.ServletException; import javax.servlet.annotation.WebServlet; import javax.servlet.http.HttpServlet; import javax.servlet.http.HttpServletRequest; import javax.servlet.http.HttpServletResponse; import javax.servlet.http.HttpSession; import com.connection.DatabaseConnection;

/\*\*

\* Servlet implementation class BuyerAccount

\*/

@WebServlet("/BuyerAccount") public class BuyerAccount extends HttpServlet { protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException { int buyerId=Integer.parseInt(request.getParameter("buyerId"));

String fullname=request.getParameter("fullname");

String email=request.getParameter("email");

String mobile=request.getParameter("mobile");

String address=request.getParameter("address");

String uname=request.getParameter("uname");

String upass=request.getParameter("upass"); HttpSession hs=request.getSession();

try {

int

addAdmin=DatabaseConnection.insertUpdateFromSqlQuery("insert

into tblbuyer(id,full\_name,email\_id,mobile\_no,address,uname,upass)

values('"+buyerId+"','"+fullname+"','"+email+"','"+mobile+"','"+address+"','"+uname+"','"+u pass+"')");

if(addAdmin>0) {

String message="Buyer account create successfully."; hs.setAttribute("success", message);

response.sendRedirect("buyer-account.jsp");

}

}catch(Exception e) {

e.printStackTrace();

}

}

}

**CUSTOMER / FARMER LOGIN**

package com.customer; import java.io.IOException; import java.sql.Connection; import java.sql.ResultSet; import java.sql.Statement; import java.util.Random;

import java.util.UUID;

import javax.servlet.ServletException; import javax.servlet.annotation.WebServlet; import javax.servlet.http.HttpServlet; import javax.servlet.http.HttpServletRequest; import javax.servlet.http.HttpServletResponse; import javax.servlet.http.HttpSession; import com.connection.DatabaseConnection;

/\*\*

\* Servlet implementation class CustomerLogin

\*/

@WebServlet("/CustomerLogin")

public class CustomerLogin extends HttpServlet {

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

String uname = request.getParameter("uname");

String upass = request.getParameter("upass");

String vercode = request.getParameter("vercode");

String captchaDB = null;

try {

//String tokens = UUID.randomUUID().toString();

Random random = new Random();

int newRandomCaptcha = random.nextInt(9000) + 10000; HttpSession hs = request.getSession();

Connection con = DatabaseConnection.getConnection();

Statement st = con.createStatement();

ResultSet captchResultSet =

DatabaseConnection.getResultFromSqlQuery("select \* from tblcaptcha");

if (captchResultSet.next()) {

captchaDB = captchResultSet.getString(1);

}

if (captchaDB.equals(vercode)) {

ResultSet resultset = st.executeQuery("select \* from

tblcustomer where uname='" + uname+ "' and upass='" + upass + "'");

if (resultset.next()) {

hs.setAttribute("id", resultset.getString("id"));

hs.setAttribute("uname", resultset.getString("uname")); hs.setAttribute("email",

resultset.getString("email\_id"));

int update =

DatabaseConnection.insertUpdateFromSqlQuery("update tblcaptcha set captcha='"+ newRandomCaptcha + "'");

response.sendRedirect("customer-dashboard.jsp");

} else {

String message = "You have enter wrong credentials";

hs.setAttribute("credential", message);

int update =

DatabaseConnection.insertUpdateFromSqlQuery("update tblcaptcha set captcha='" + newRandomCaptcha + "'");

response.sendRedirect("customer-login.jsp");

}

} else {

String message = "You have enter invalid verification code"; hs.setAttribute("verificationCode", message);

int update =

DatabaseConnection.insertUpdateFromSqlQuery("update tblcaptcha set captcha='" + newRandomCaptcha + "'");

response.sendRedirect("customer-login.jsp");

}

} catch (Exception e) {

e.printStackTrace();

}

}

}

**CUSTOMER/FARMER CHANGE PASSWORD**

package com.customer;

import java.io.IOException;

import java.sql.Connection;

import java.sql.ResultSet;

import java.sql.Statement;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

import com.connection.DatabaseConnection;

/\*\*

\* Servlet implementation class CustomerChanagePassword

\*/

@WebServlet("/CustomerChanagePassword")

public class CustomerChanagePassword extends HttpServlet {

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

String cpassword = request.getParameter("cpassword");

String password = request.getParameter("password");

String confpass = request.getParameter("confpass");

String pass = "";

HttpSession session = request.getSession();

try {

Connection con = DatabaseConnection.getConnection();

Statement statement = con.createStatement();

ResultSet resultset = statement.executeQuery("select upass from tblcustomer where upass='" + cpassword+ "' and uname='" + session.getAttribute("uname") + "'");

if (resultset.next()) {

pass = resultset.getString(1);

}

if (password.equals(confpass)) {

if (pass.equals(cpassword)) {

int i = statement.executeUpdate("update tblcustomer set upass='" + password + "' where uname='" + session.getAttribute("uname") + "' ");

String message = "Password change successfully.";

session.setAttribute("password-change-success", message);

response.sendRedirect("change-customer-password.jsp");

} else {

String message = "Old password does not match.";

session.setAttribute("password-change-fail", message);

response.sendRedirect("change-customer-password.jsp");

}

} else {

String message = "New password and confirm password does not match.";

session.setAttribute("password-not-match", message);

response.sendRedirect("change-customer-password.jsp");

}

} catch (Exception e) {

e.printStackTrace();

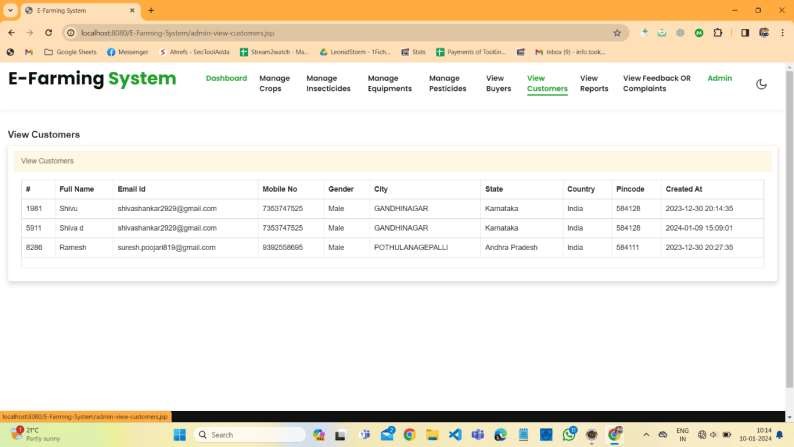
}

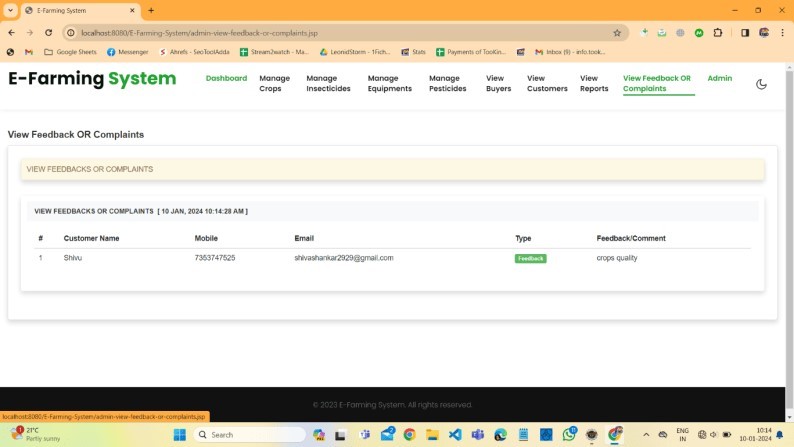
}

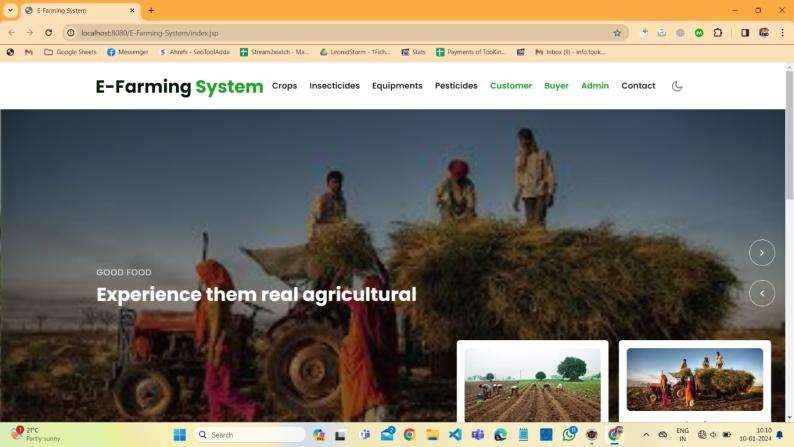
}

**APPENDIX-B**

**SCREENSHOTS**



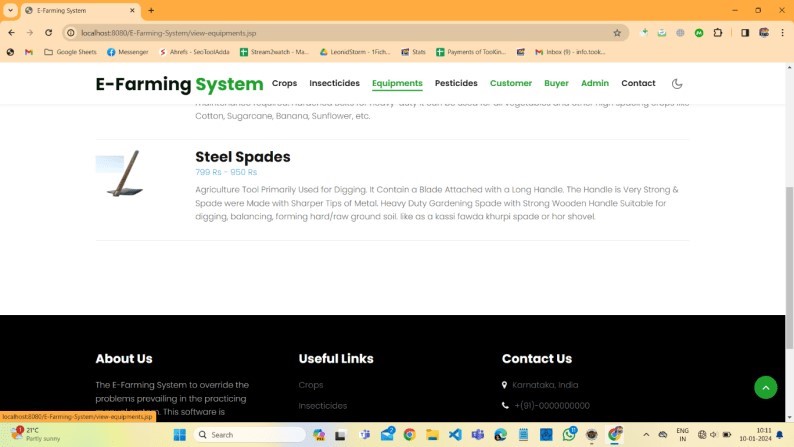
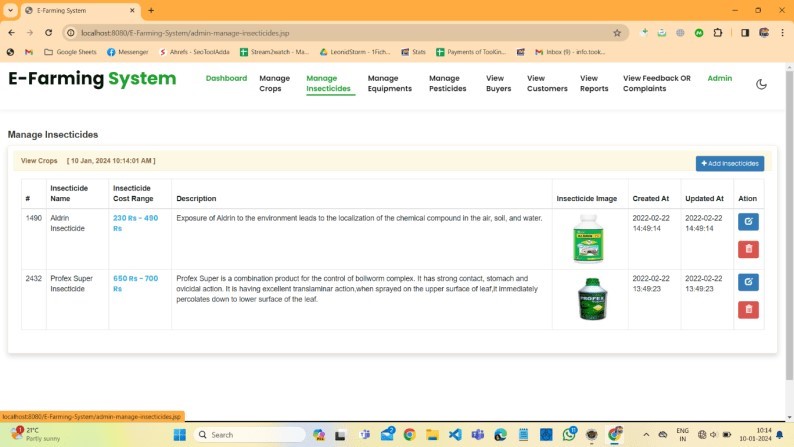
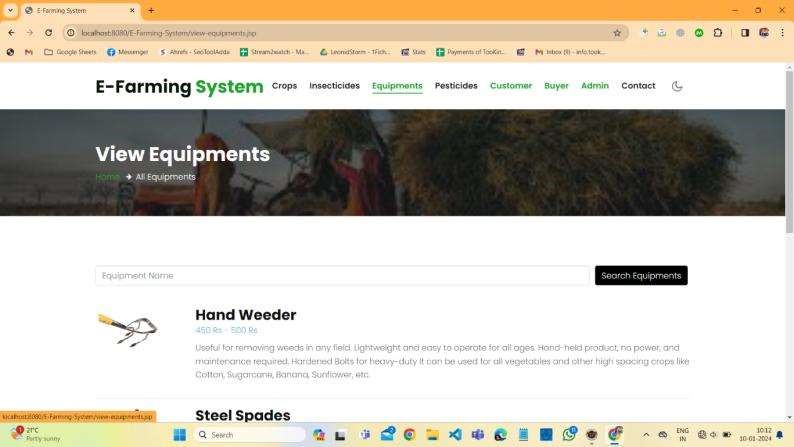
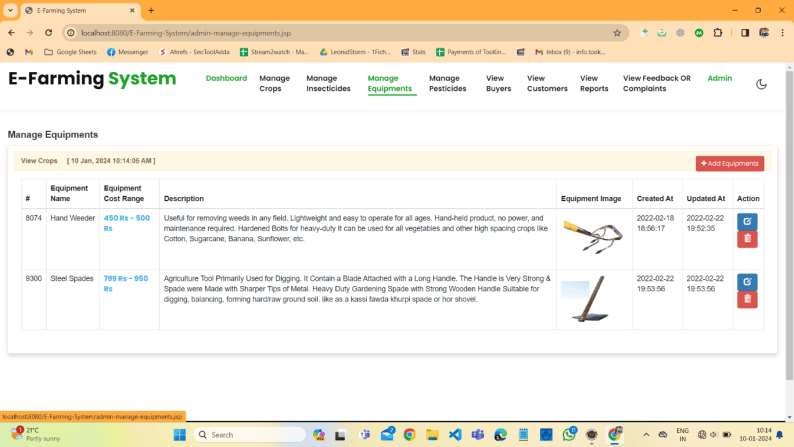
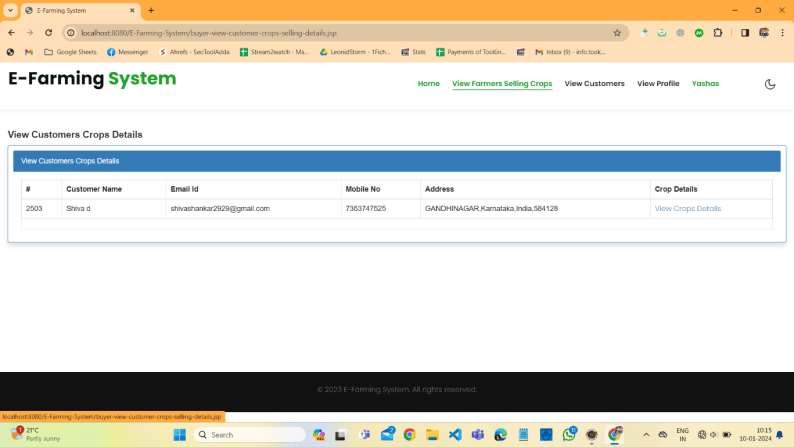
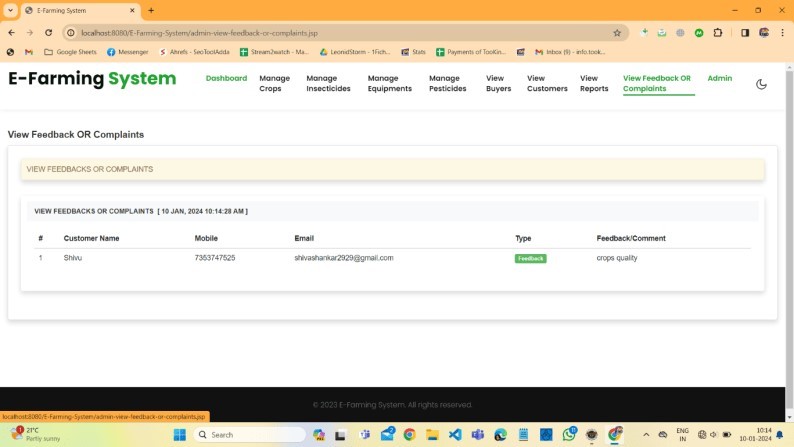




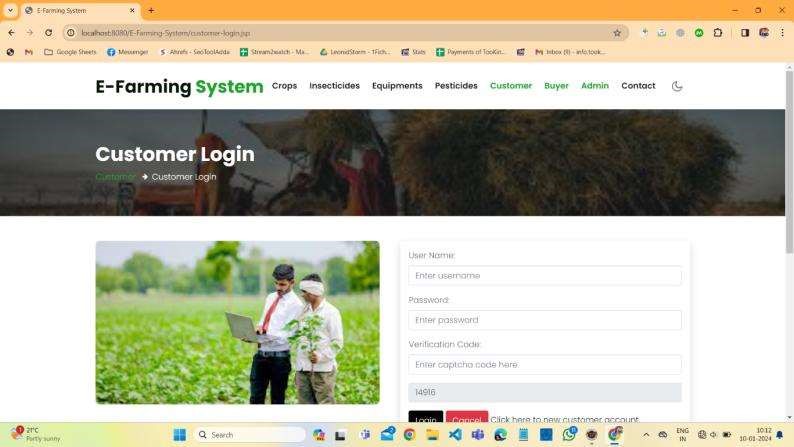
**Admin Login:**



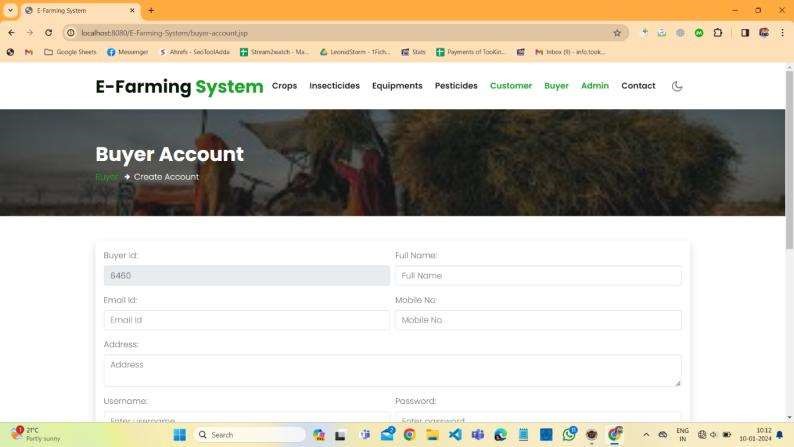
**Customer Account Creating Page:**



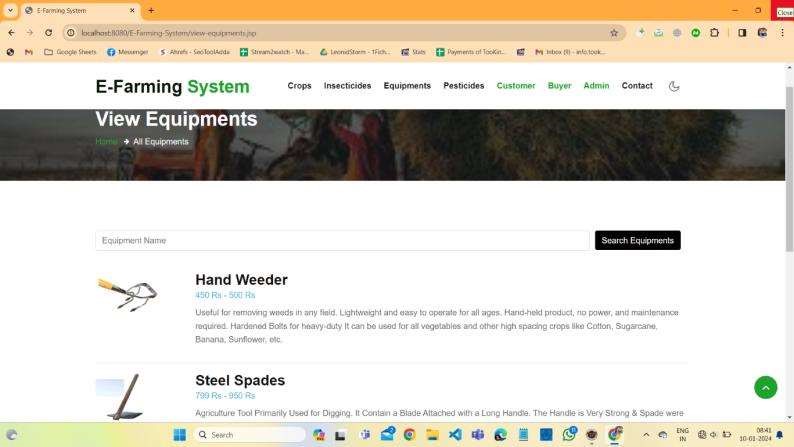
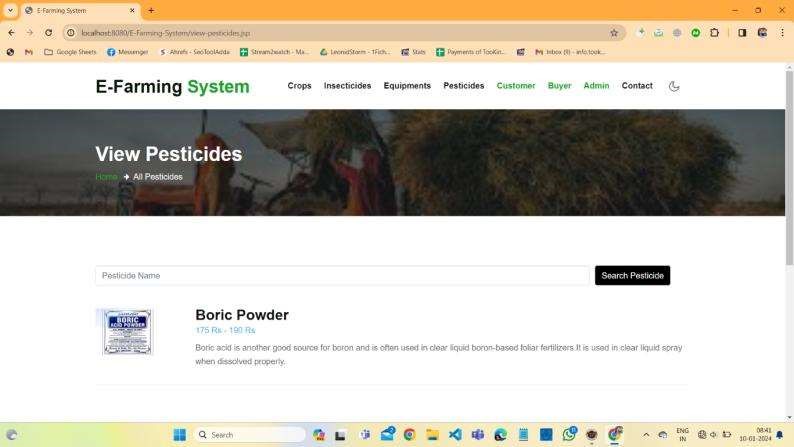
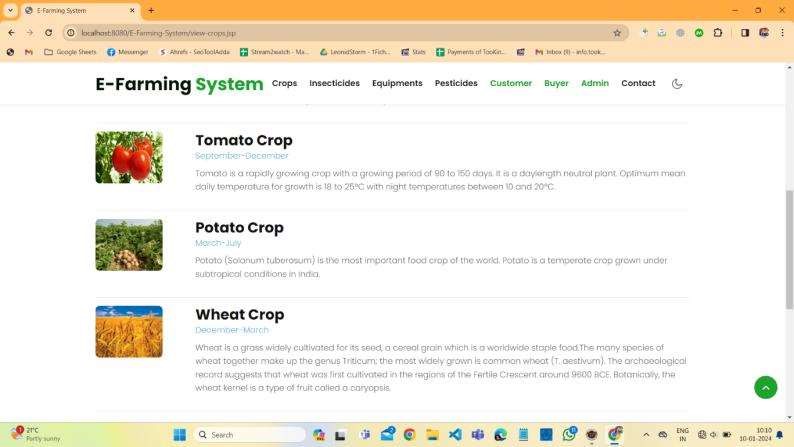
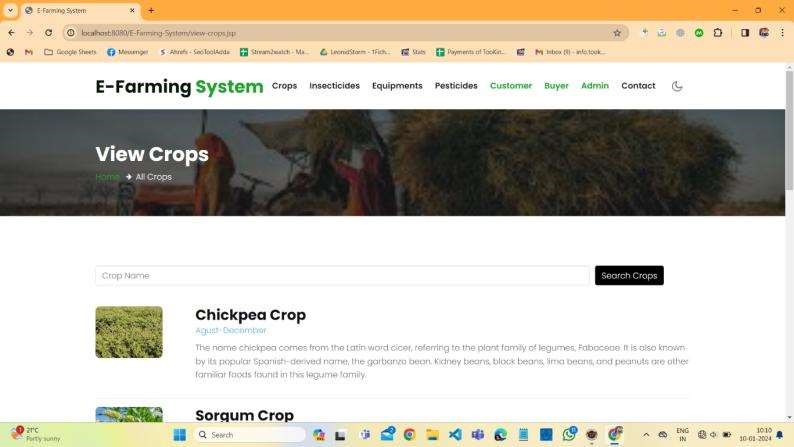
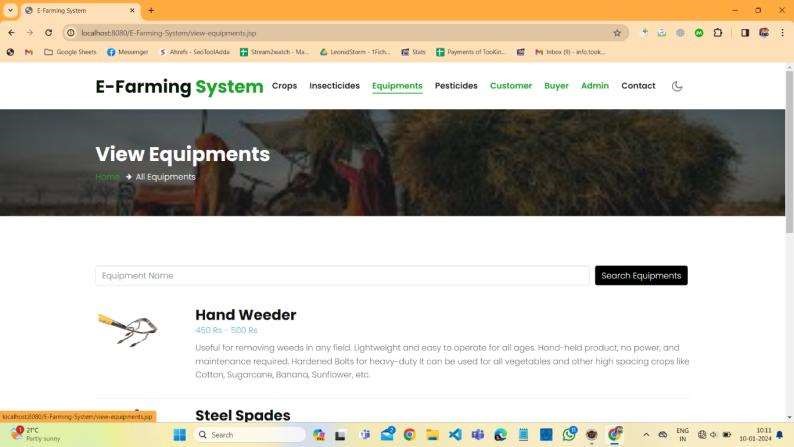
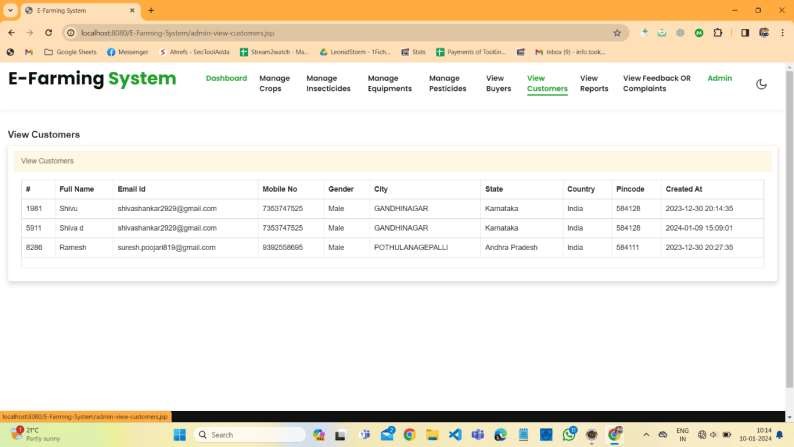
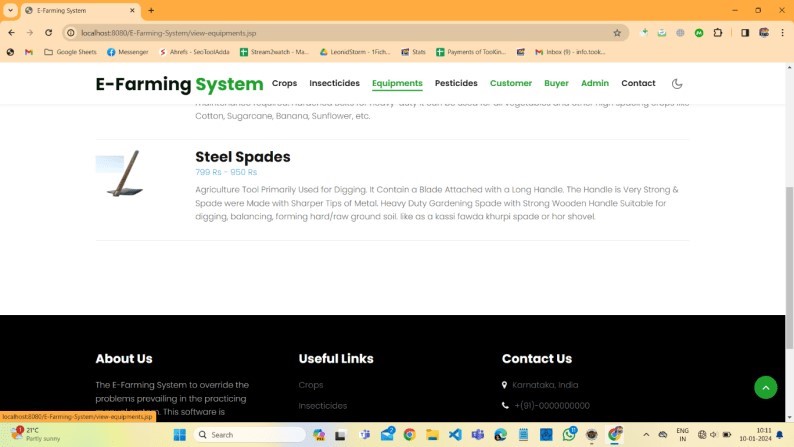
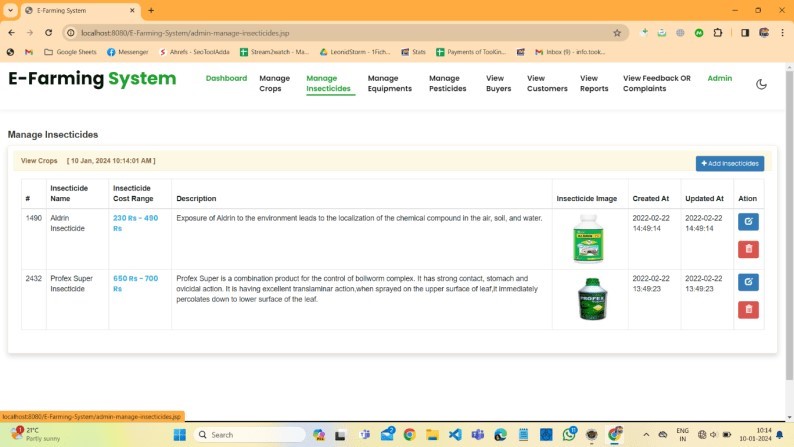
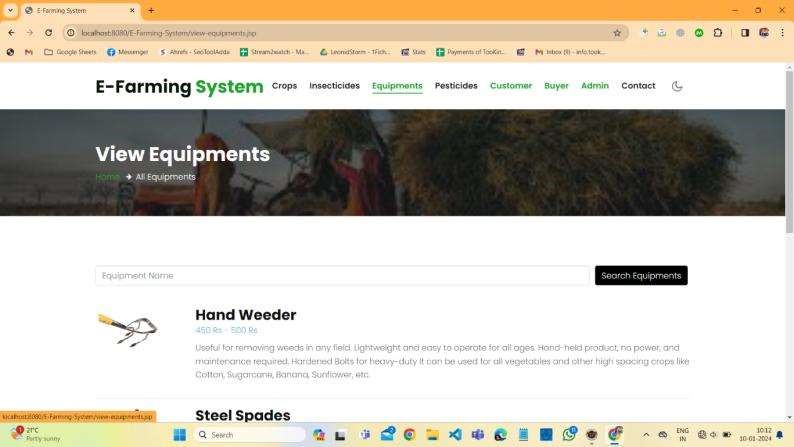
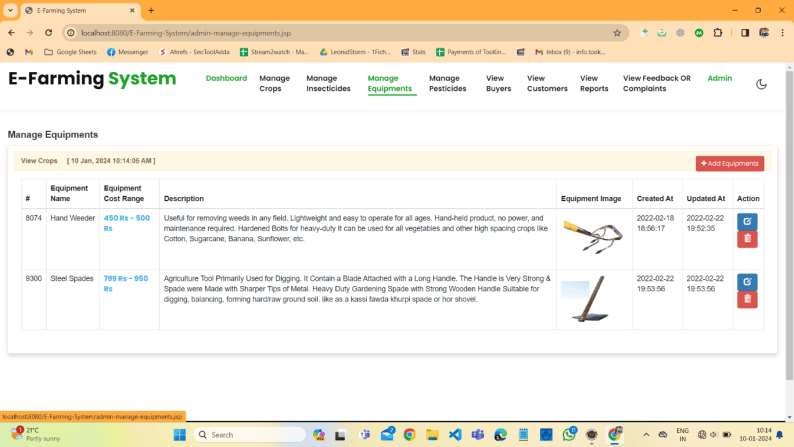
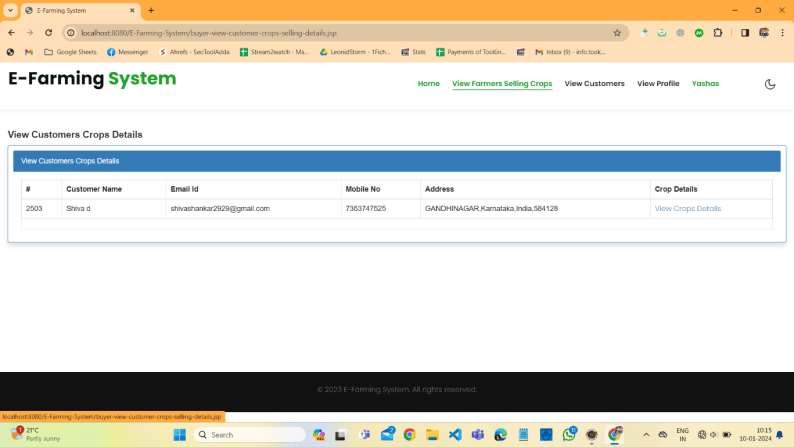
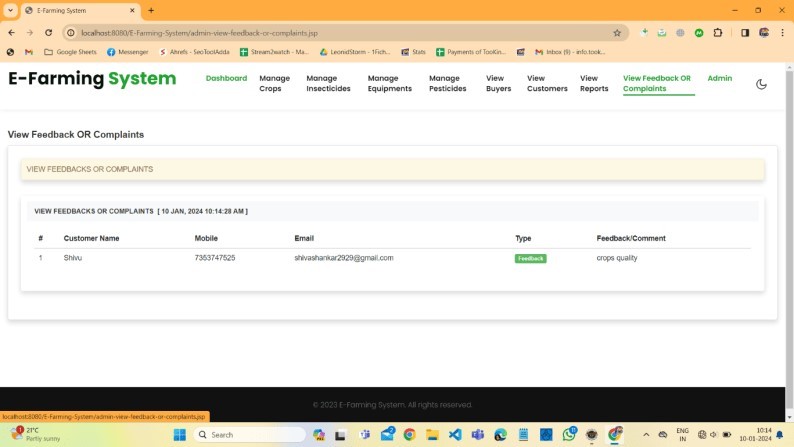
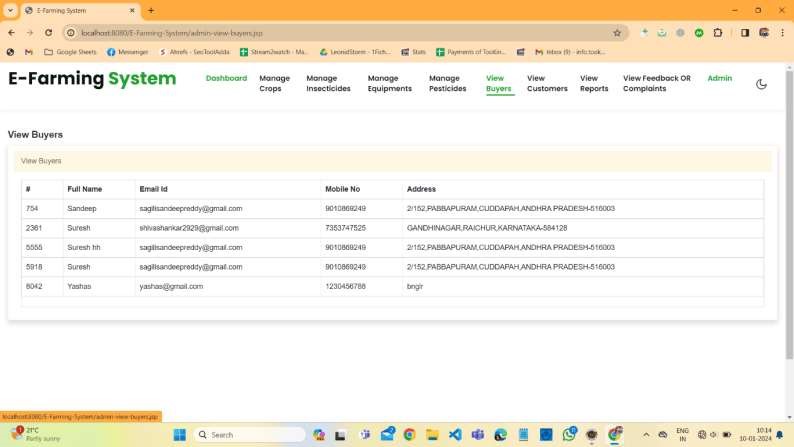
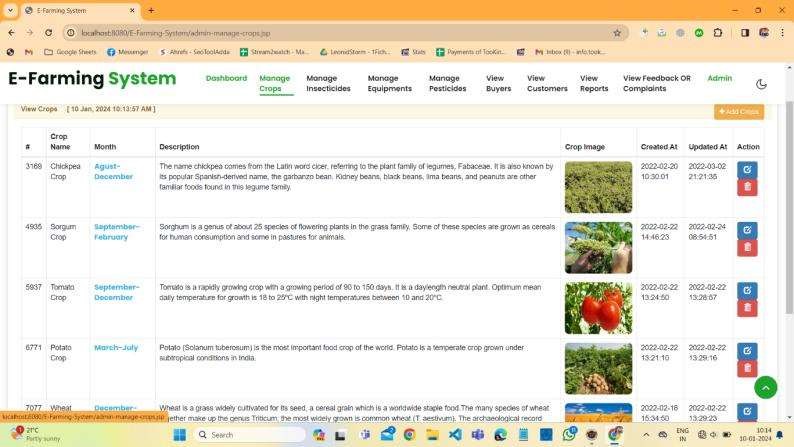
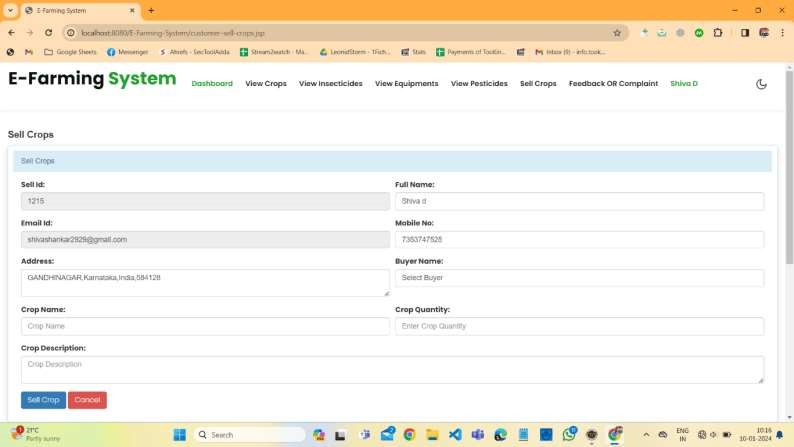
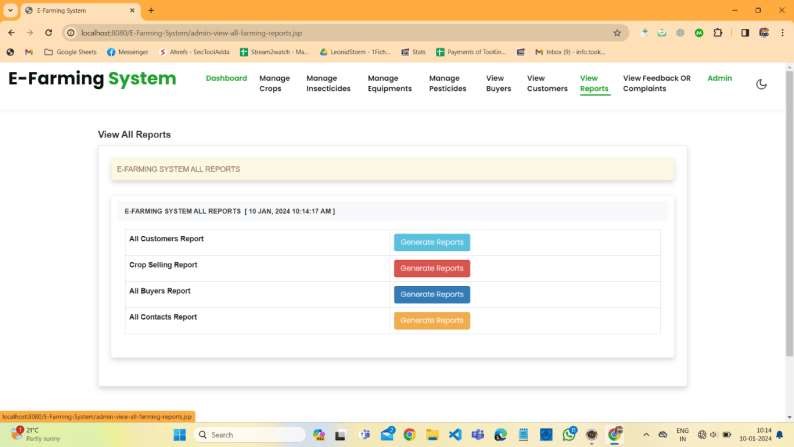
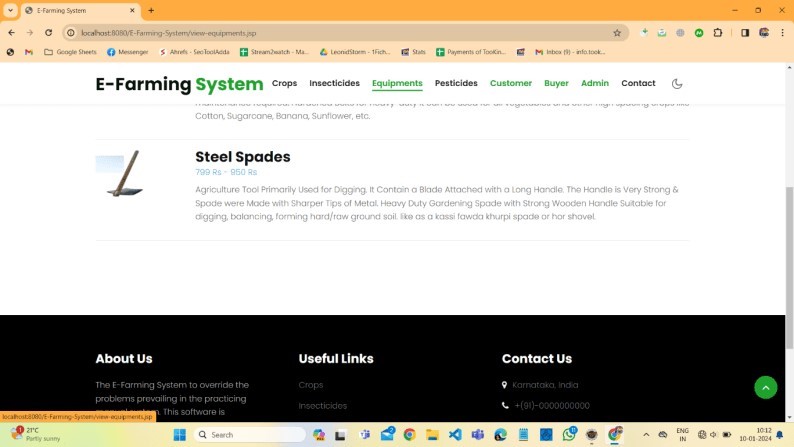
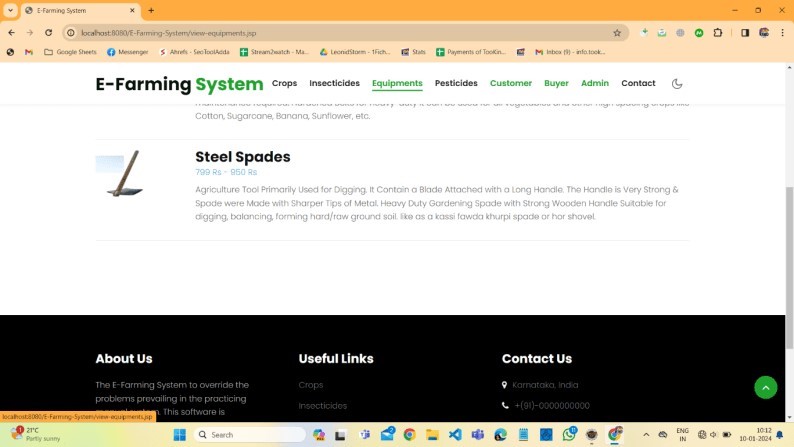
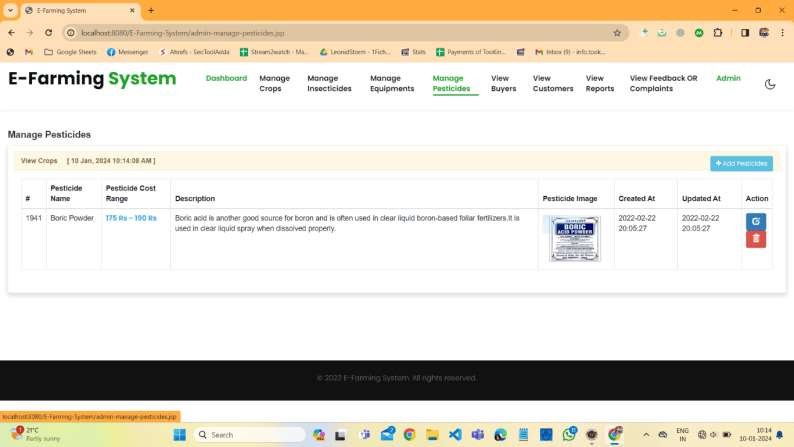
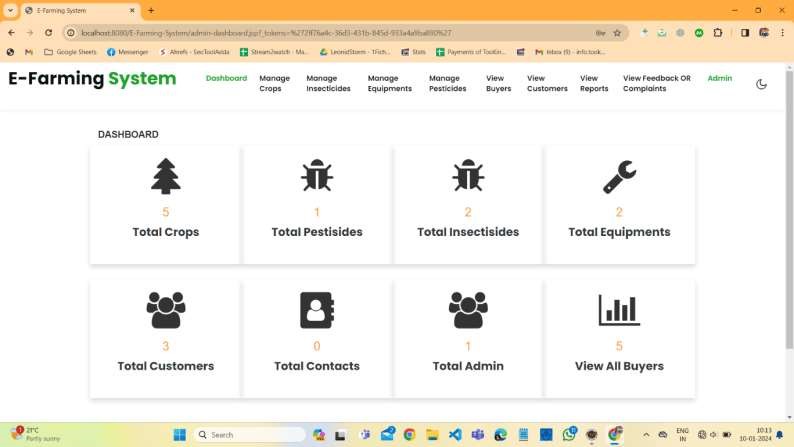
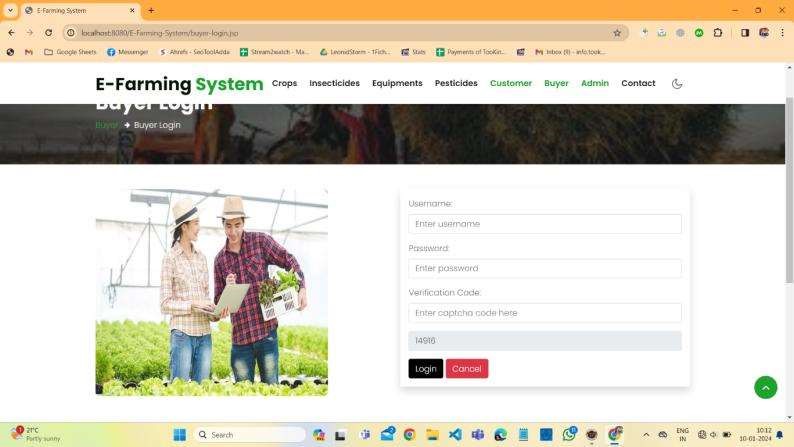
**Customer Login Page:**



**Buyer Account Creation Page:**



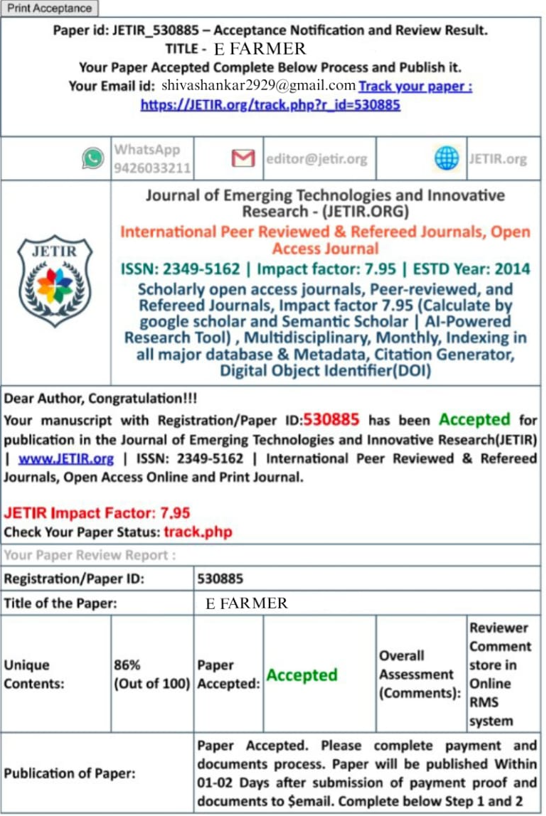
**Buyer Login Page:**



**APPENDIX-C**

**ENCLOSURES**

**ACCEPTANCE CERTIFICATE**

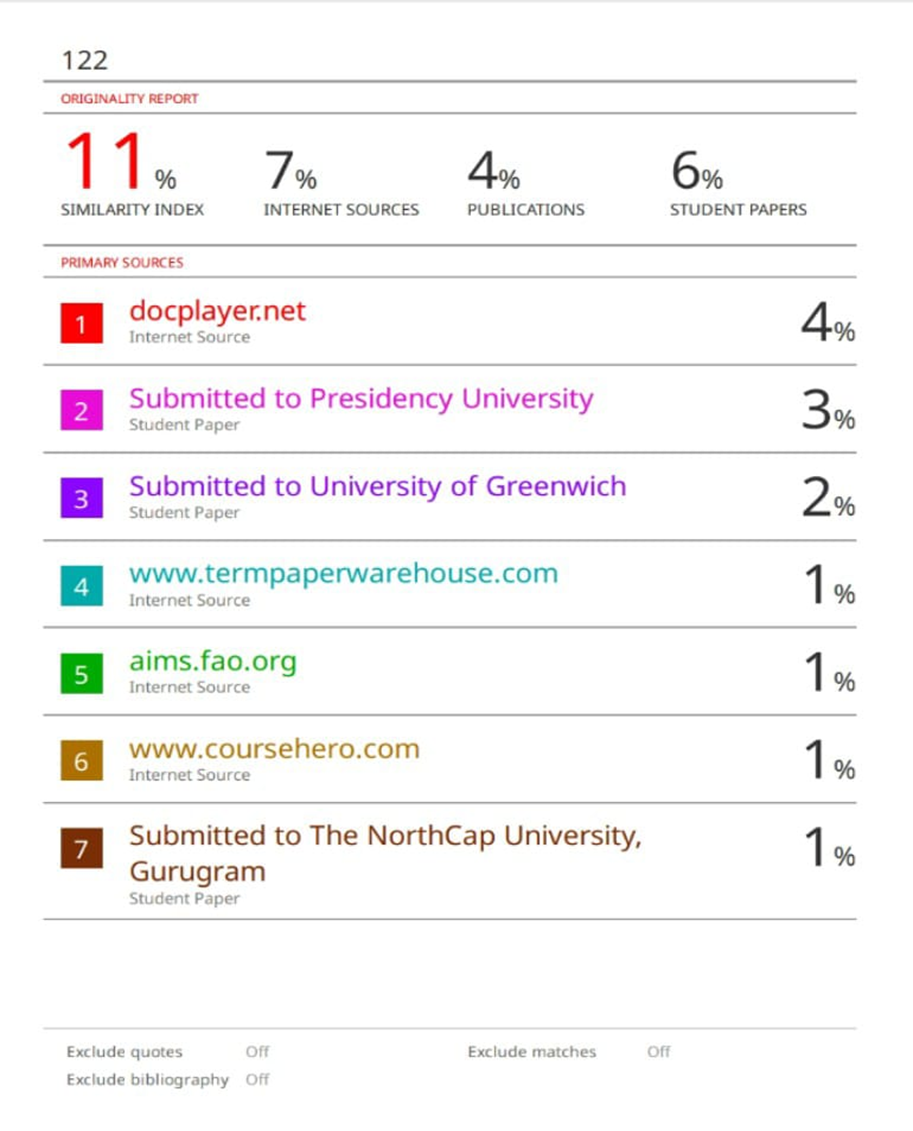
****

****

****

****

**PLAGARISM CHEKED REPORT**





The Project work carried out here is mapped to SDG-12, Responsible Consumption and Production .The E-Farmer project aligns with SDG-12 Responsible Consumption and Production, focusing on Agriculture and Rural Development. it enhances transparency and efficiency in agricultural processes, promoting sustainable practices and contributing to responsible consumption and production goals.