

**INTEGRATED INVENTORY AND ANIMAL VACCINATION RECORD
MANAGEMENT SYSTEM FOR THE DEPARTMENT OF AGRICULTURE -
LILIW OFFICE**

A Final Project
Presented to the
Faculty of College of Computer Studies
Laguna State Polytechnic University
Santa Cruz Campus

**In partial Fulfilment of the Requirements for the course
CLIENT SERVER TECHNOLOGIES**

Submitted by:

Godffrey M. Bonza
Lean Jay L. Javier
John Railey Raymundo
Daisy M. Violanta

Submitted to:

Harlene Gabrielle E. Origines
Course Instructor

December 2025

CHAPTER I

INTRODUCTION

Today, technology is changing how organizations handle their operations, especially in inventory control and record management. Manual recordkeeping, once the main way to track data, is now being replaced by digital systems that are more efficient, accurate, and accessible. Automated systems reduce mistakes, save time, and help people make better decisions with real-time, organized data.

This study looks at creating an Integrated Inventory and Animal Vaccination Record Management System for the Department of Agriculture in Liliw. The current manual process for recording supplies, vaccines, and vaccination activities takes a lot of time and is often inaccurate. These problems can cause delays, incorrect reports, and make it hard to track vaccine stocks and animal health records. By using a digital system, the office can manage inventory and vaccination records more efficiently and accurately, which will help provide better service to local farmers and livestock owners.

Background of the Study

Client-server technology enables multiple users to efficiently access, store, and manage shared information through a centralized system. This technology allows real-time data storage, retrieval, and updating, making organizational operations faster, more secure, and more reliable. For the Department of Agriculture - Liliw Office, adopting this technology is essential in

improving the management of agricultural inventory and animal vaccination records.

By implementing an Integrated Inventory and Animal Vaccination Record Management System, the office can modernize its recordkeeping processes, enhance coordination among staff, ensure accurate tracking of supplies and vaccines, and maintain reliable vaccination histories for animals. This system will promote transparency, accountability, and efficiency in the distribution of agricultural resources and delivery of veterinary services, ultimately supporting the office's mission to improve agricultural productivity and animal health within the municipality of Liliw.

Research Problem

The main issue in this study is the inefficiency of the current manual inventory and animal vaccination record management used by the Department of Agriculture, Liliw Office. The existing system depends on handwritten logs and spreadsheets to track agricultural supplies, vaccines, and vaccination histories. This method is susceptible to human error, inconsistent data, delayed reporting, and a lack of real-time monitoring. These problems hinder effective service delivery, planning, and decision-making.

The specific problems include:

1. How can the system reduce errors in recording and updating inventory and vaccination data?

2. How can the system provide real-time access to inventory levels and vaccination information?
3. How can the system improve the efficiency and speed of generating reports for inventory and vaccinations?
4. How can the system help staff accurately monitor and track agricultural supplies, vaccines, and vaccinated animals?
5. How can the system ensure organized and reliable vaccination histories for animals under the office's care?

These issues arise because of the ongoing use of manual documentation and disconnected digital files. These methods are hard to update, retrieve, and verify. Delays in recording inventory use and vaccination activities often lead to inaccurate stock levels, missed vaccination schedules, and incomplete health records for animals. Evidence of these problems includes frequent mismatches between actual inventory and recorded data, as well as incomplete or lost vaccination records. This results in inefficiencies in agricultural and veterinary support programs. To tackle these challenges, the researchers suggest creating an Integrated Inventory and Animal Vaccination Record Management System. This system would automate the recording, updating, monitoring, and reporting of inventory and vaccination data through a centralized digital platform.

Project Objectives

The overall goal of this study is to design and develop an Integrated Inventory and Animal Vaccination Record Management System. This system

aims to improve the efficiency, accuracy, and accessibility of inventory and vaccination record management for the Department of Agriculture - Liliw Office.

The system specifically aims to:

1. Create a centralized digital platform for managing agricultural inventory and animal vaccination records.
2. Offer real-time updates and reports on inventory levels, vaccine usage, and vaccination status.
3. Minimize human errors and data duplication through automated data entry and validation.
4. Enhance accessibility, coordination, and monitoring among staff managing inventory and vaccination activities.
5. Keep accurate and current vaccination histories to support animal health monitoring and program planning.
6. Improve decision-making with reliable, organized, and timely inventory and vaccination data.

Scope and Limitations of the Research

This study focuses on designing and developing the Integrated Inventory and Animal Vaccination Record Management System for the Department of Agriculture - Liliw Office. It includes the digital recording, updating, monitoring, and reporting of agricultural supplies, equipment, vaccines, and vaccination records handled by the office. Key functions covered include item and vaccine registration, stock monitoring, animal profiling, vaccination scheduling, and automatic generation of inventory and vaccination reports within the office.

Limitations of the study involve factors outside the researchers' control, such as internet connectivity problems, user training and adaptation, and system maintenance post-implementation. The system is primarily designed for internal use and may not initially support large-scale deployment, offline functionality, or integration with other municipal or regional offices.

Significance of the Study

The proposed system will greatly benefit the Department of Agriculture - Liliw Office by providing a faster, more reliable, and more accurate method of managing inventory records. Through automation, the system will minimize manual processes, reduce human error, and streamline daily operations. Office staff will experience a lighter administrative workload and improved efficiency in tracking supplies, equipment, and agricultural inputs. Management will also gain access to real-time, data-driven insights, enabling more informed decisionmaking, better resource allocation, and timely responses to operational needs.

Furthermore, farmers and the local community will indirectly benefit from the improved services offered by the department. With a more organized and transparent inventory system, the distribution of agricultural resources such as seeds, fertilizers, and equipment can become more efficient and equitable. This ultimately supports increased agricultural productivity, ensures timely assistance, and contributes to the overall development of the municipality's agricultural sector.

- Department of Agriculture - Liliw Office

- Office Staff
- Department Management/Administrators
- Local Farmers
- Local Community / Beneficiaries of Agricultural Programs

CHAPTER II

SYSTEM ANALYSIS AND DESIGN

Overview of Client-Server Model

The proposed Integrated Digital Inventory System uses a client-server architecture, which divides the system into two main parts: the client side and the server side. The client side refers to the user interface that employees use to record, update, and view inventory data through a web browser. The server side handles the storage, processing, and management of all inventory records in a centralized database.

In this setup, users send requests (such as adding or viewing items) from their computers to the server. The server processes the request, retrieves or updates the data, and then sends back the results to the client. This model allows multiple users to access the system simultaneously without interfering with each other's work. The client-server model improves data security, efficiency, and accessibility, making it ideal for managing agricultural resources in the Department of Agriculture - Liliw Office.

System Requirements

To ensure that the proposed system operates efficiently, securely, and reliably, the following hardware, software, and network requirements are recommended. These components work together to support smooth performance, data accuracy, stable connectivity, and ease of access for users.

Hardware Requirements

1. Server

The server hosts the main application, database, and system resources; therefore, it must have enough processing power, memory, and storage.

- Processor: Intel Core i5 or higher**

A multi-core processor provides faster computing performance, enabling the server to handle multiple simultaneous requests, process large amounts of data, and support several users at once without lag.

- RAM: 8GB or higher**

Adequate memory ensures the server can process tasks smoothly. Higher RAM allows faster data retrieval, improves database performance, and prevents system slowdowns, especially when handling large inventory records.

- Storage: 500GB or more**

Sufficient storage is required for saving inventory data, logs, backups, system files, and any future expansion of the database.

- Operating System: Windows 10 / Linux Ubuntu**

These operating systems provide stable environments for running Apache, MySQL, and PHP. They are widely supported, secure, and compatible with most server tools.

2. Client Computers

Client machines access the system through a browser; therefore, they must meet minimum specifications to ensure smooth usage.

- **Processor: Intel Core i3 or higher**

Ensures client computers can handle web-based operations and load system pages quickly.

- **RAM: 4GB minimum**

Supports smooth browser operation and multitasking without lag.

- **Storage: 100GB free space**

Allows installation of operating systems, browsers, and temporary cache storage needed by the system.

- **Browser: Google Chrome or Mozilla Firefox**

Modern browsers ensure compatibility with web technologies like JavaScript, CSS, and responsive interfaces, ensuring a consistent user experience.

Software Requirements

1. Programming Language: PHP 8.0 or later

Modern versions of PHP are faster, more secure, and offer improved performance. PHP 8 introduces better error handling and optimization, which ensures the system runs smoothly and safely.

2. Web Server: Apache (via XAMPP or WAMP)

Apache serves as the backbone of the web application, handling client requests and delivering the system interface. XAMPP or WAMP makes local deployment and testing easier.

3. Database: MySQL

MySQL is efficient, lightweight, and highly compatible with PHP. It allows structured storage of inventory records and supports fast and reliable retrieval of data needed for daily operations.

4. Front-End: HTML, CSS, JavaScript, Bootstrap

These technologies create the user interface:

- **HTML** structures content
- **CSS** improves appearance
- **JavaScript** adds interactivity and dynamic functions
- **Bootstrap** ensures a responsive, mobile-ready layout

This combination ensures users have a clean, intuitive, and accessible system interface.

5. API (if applicable): REST API

A REST API allows future integration with external systems such as weather data, supply deliveries, or mobile applications. This expands the system's capabilities and supports modernization.

6. Operating System: Windows or Linux

Both OS options provide compatibility with PHP, MySQL, and Apache. Linux is preferred for servers due to security and stability, while Windows is ideal for client use.

Network Requirements

1. Local Area Network (LAN) or Internet Connection

A LAN allows the system to be accessed by multiple users within the office.

An internet connection supports:

- Remote access (if enabled)
- Real-time synchronization
- System updates
- Cloud or remote database hosting

2. Router or Switch

These ensure proper network distribution, enabling client computers to connect to the server efficiently. A stable internal network reduces delays and improves system performance.

3. Stable Internet Connection Needed

for:

- API calls (e.g., weather updates or supply status)
- Remote backup
- Future online expansion
- Updating system components

A reliable connection ensures the system remains functional and up-to-date.

Overall Importance of These Requirements

These specifications ensure that the proposed system will:

- Operate smoothly with minimal downtime
- Accommodate multiple users simultaneously
- Handle growing data without slowing down
- Provide fast and accurate data access
- Support security, future expansion, and system stability

By meeting these requirements, the Department of Agriculture - Liliw Office will have a dependable, efficient, and scalable inventory management system that serves both staff and farmers effectively.

System Design

The system design represents how data and processes flow within the system. It shows how different modules interact to perform specific functions such as adding items, updating inventory, and generating reports.

1. System Flowchart

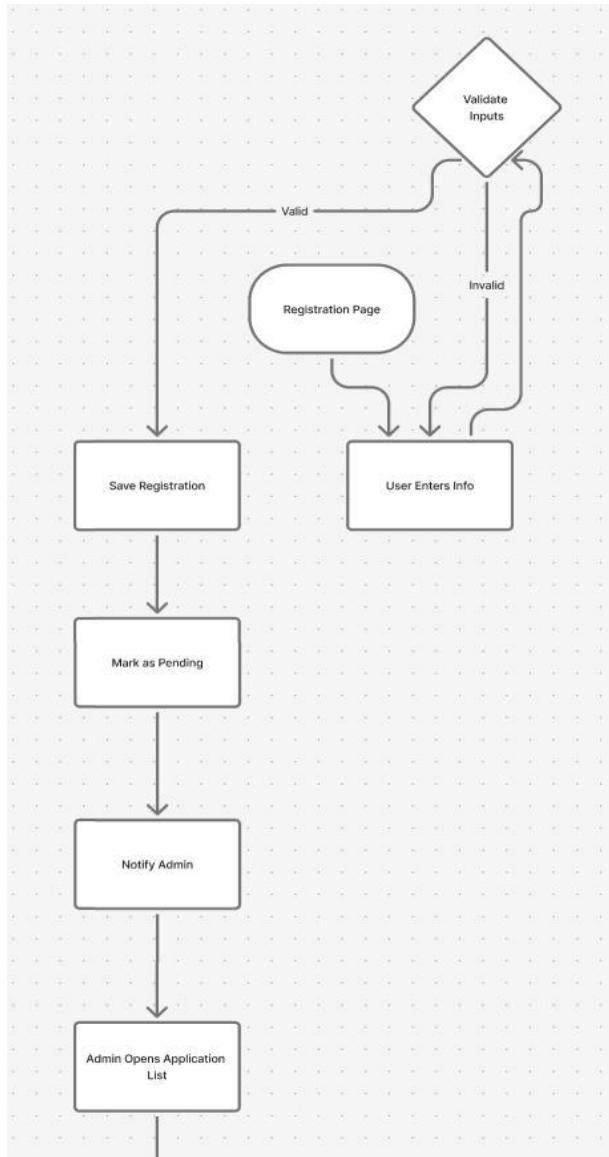


Figure 1. User Registration (Staff)

The system flow begins with user registration, where a user/staff member creates an account to access the inventory management system in Liliw. Users provide basic information such as their name, email, and password. Once registered, they can log in to the system to view and manage items, monitor stock, and perform transactions efficiently.

The data entered by the user is sent to the server, where it is validated and stored in the MySQL database. The server then sends a confirmation or result back to the client.

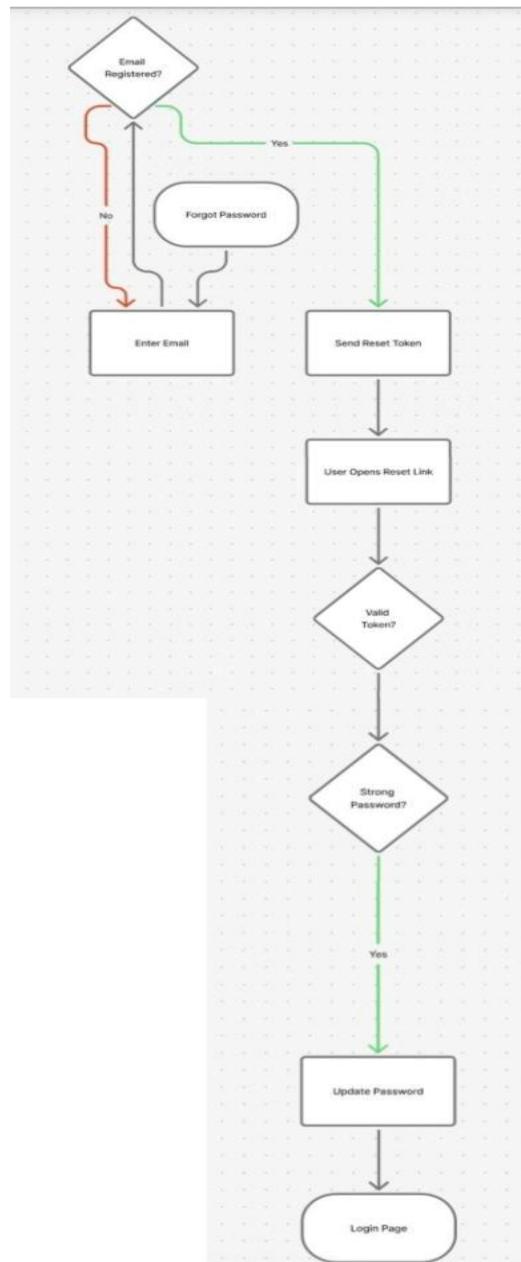


Figure 2. Email Verification

The user first checks if their email is registered in the system. If not, they enter their email, if yes, a reset token is sent. After the user opens the reset link, the token is verified. If valid, the user creates a new password, if it meets

strength requirements, the password is updated, and the user is directed to the login page.

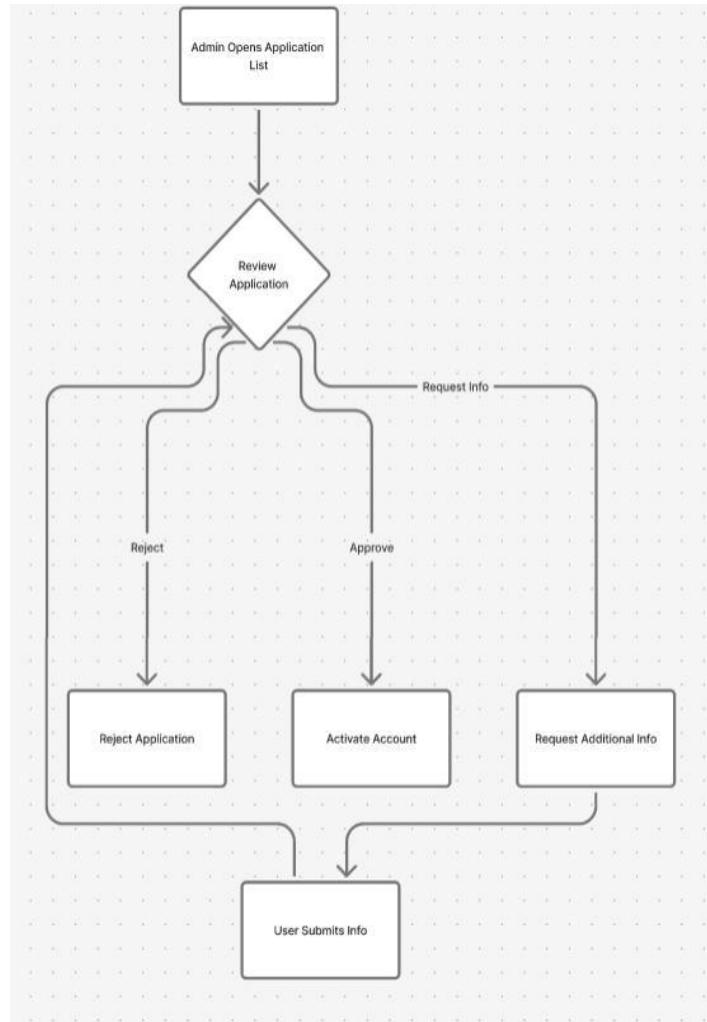


Figure 3. User/Staff Application Evaluation (Admin)

The process begins when an admin opens the application list. Next, the admin reviews each application, with three possible outcomes:

- If rejected, the application is formally declined.
- If approved, the user's account is activated.
- If more details are needed, additional information is requested from the user.

In all cases, the process concludes once the user submits any required information or the application is finalized.

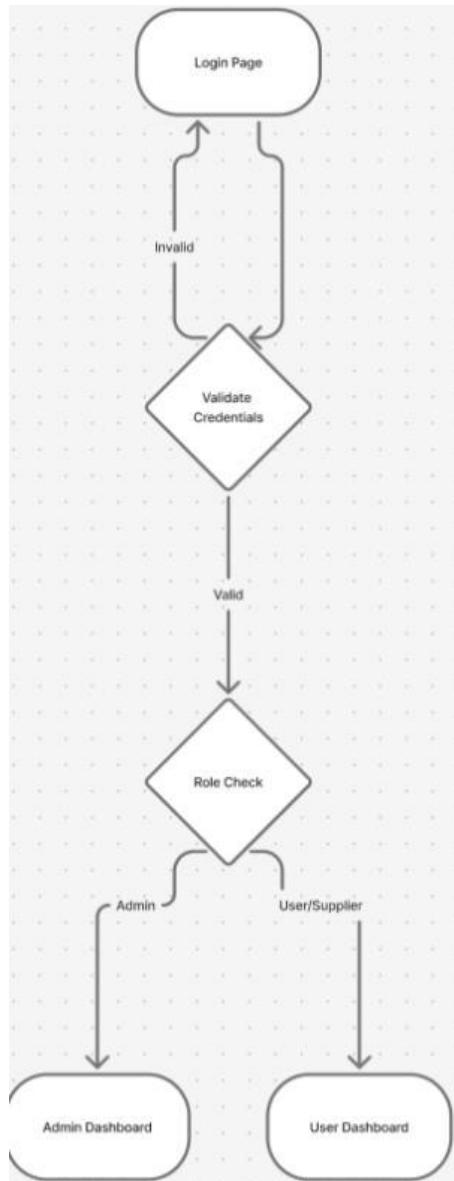


Figure 4. User login

Users begin by accessing the Login Page, where their credentials are entered and then validated. If credentials are invalid, the user is returned to the Login Page to try again. If valid, a role check is performed, admins are directed to the Admin Dashboard, while users or staff member are taken to the User Dashboard.

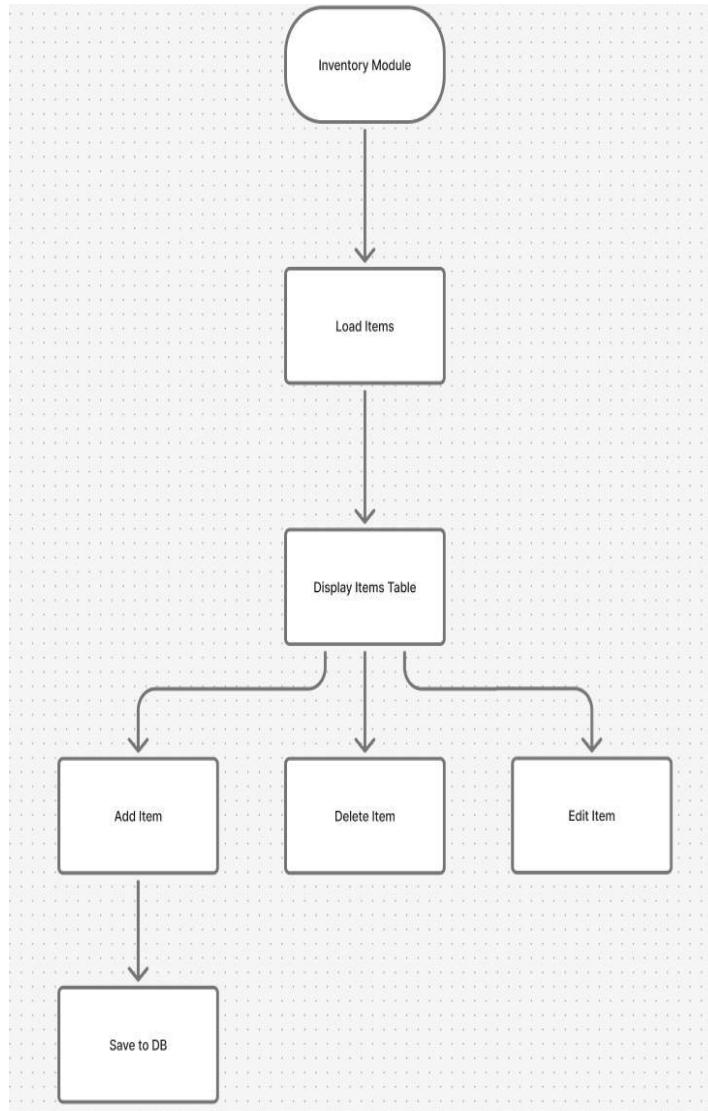


Figure 5. Inventory Module

The Inventory Module initiates by loading items from the system, which are then displayed in an items table. From here, users can perform three core actions: add an item (with new entries saved to the database), delete an item, or edit an item.

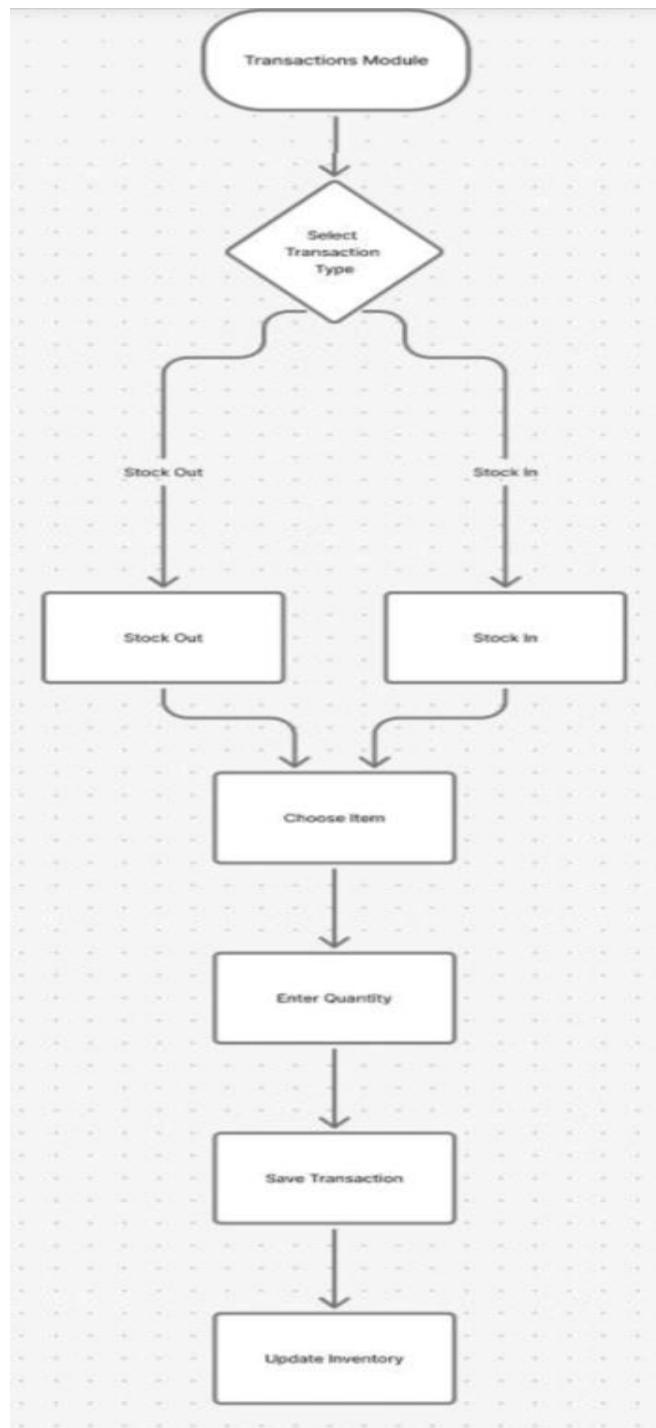


Figure 6. Transaction Module

The Transactions Module starts by prompting the user to select a transaction type either Stock Out or Stock In. Once a type is chosen, the user proceeds to select the relevant item, enters the transaction quantity, saves the transaction details, and finally the system updates the inventory records accordingly.

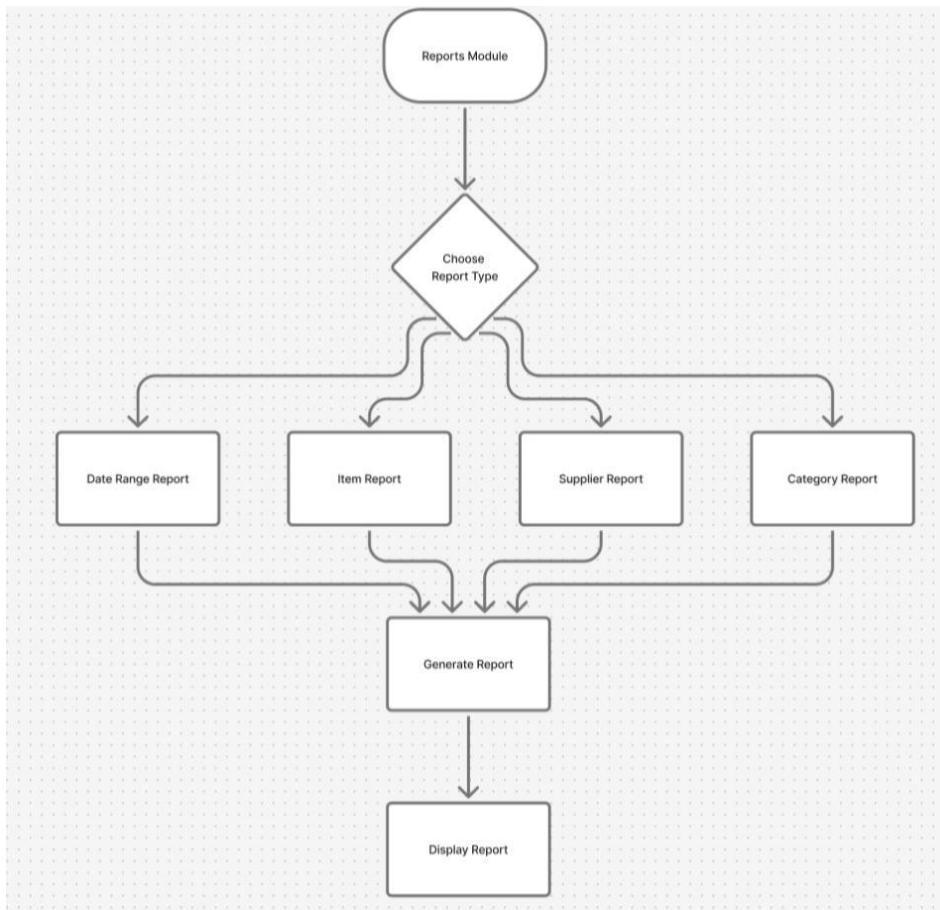


Figure 7. Reports Module

The Reports Module is viewed by having the user select a report type from four options: Date Range Report, Item Report, (exclude supplier) or Category Report. After making a selection, the system generates the chosen report and then displays it for the user to view.

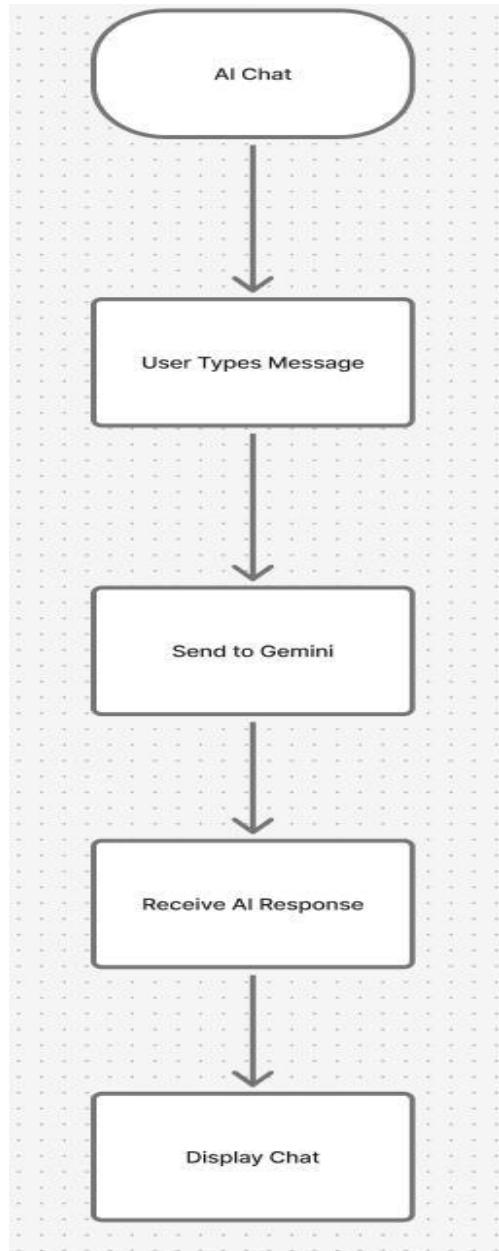


Figure 8. Help & Support Chat

The AI Chat feature is accessible via a floating bubble on the website. When users click it to open the chat interface, they type their message, which is sent to Google's Gemini AI model. After receiving the AI-generated response, the full conversation including the user's input and the AI's reply is displayed within the chat window.

2. Entity-Relationship (ER) Diagram

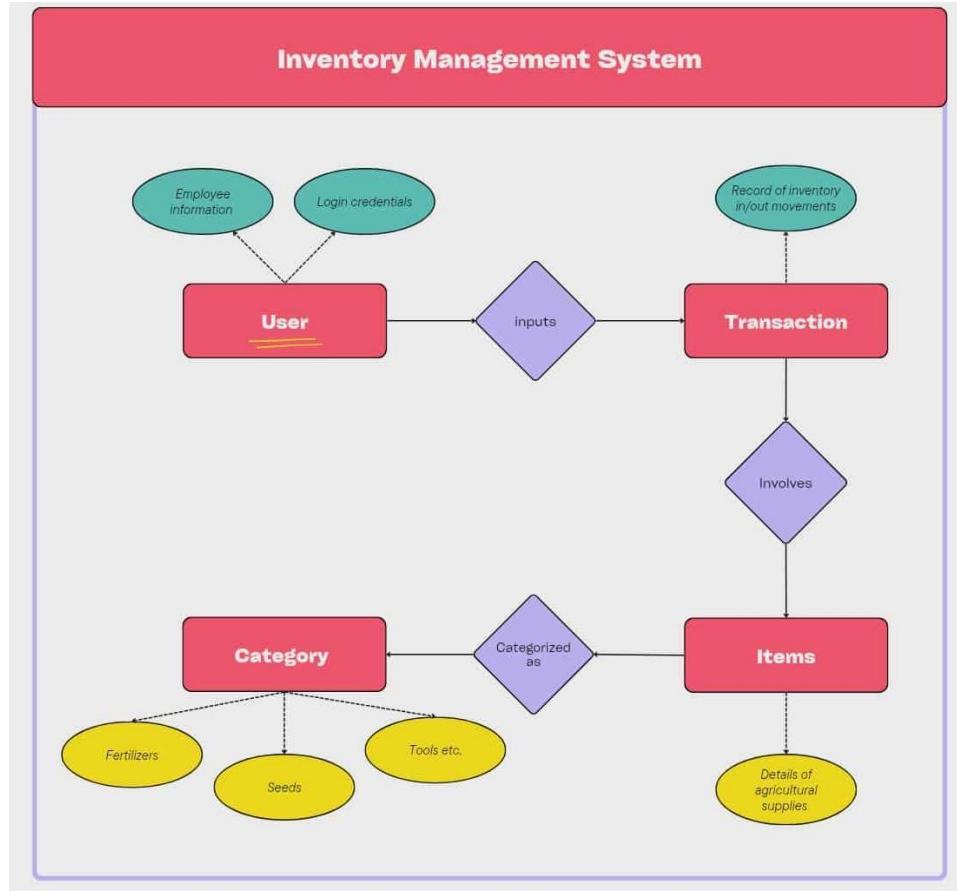


Figure 9. ER Diagram

The ER diagram shows the main entities of the system:

User (Employee information and login credentials)

Item (Details of agricultural supplies)

Category (Grouping of items like fertilizers, seeds, tools)

Transaction (Record of inventory in/out movements)

These entities are linked through relationships, ensuring that every transaction and item is connected to a valid user and category.

3. UML Use Case Diagram

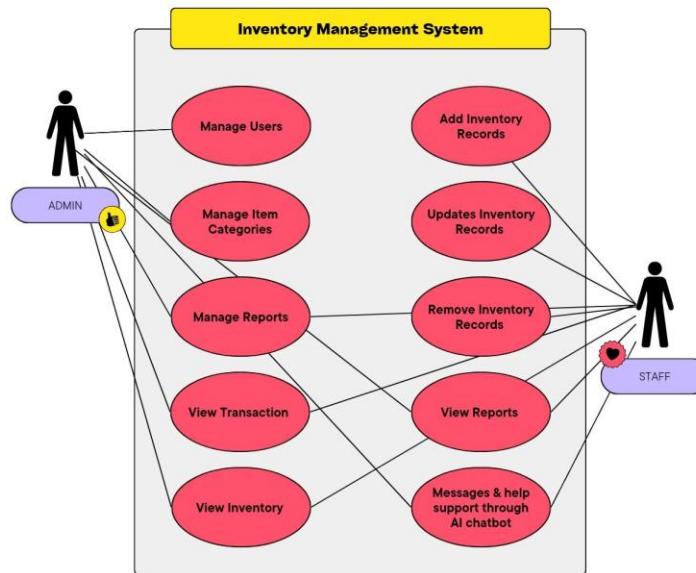


Figure 10. Use Case Diagram

The Inventory Management System provides structured access to two user roles: Admin and Staff.

Admins have full privileges, including managing users, item categories, and reports; viewing transactions and inventory; adding/updating/removing inventory records; and using the AI chatbot for support.

Staff users focus on operational tasks, such as maintaining inventory records, viewing inventory data and reports, and accessing the AI chatbot.

Key features managing inventory records, viewing reports/inventory, and AI chatbot access are shared across both roles to ensure coordinated workflows and resource accessibility.

API Integration

The Digital Inventory System integrates several external APIs to enhance its functionality and improve user experience. These APIs allow the system to provide real-time information, communication capabilities, and location-based services. The main APIs used are:

1. **News API** - for displaying agriculture-related news updates.

Purpose:

To display the latest agriculture-related news and updates relevant to the Department of Agriculture's operations.

APIType:

RESTful API (e.g., [NewsAPI.org](#)) **Endpoints**

Used:

- GET /v2/everything?q=agriculture&language=en - fetches the latest agriculture-related news articles.

RequestType:

GET

Authentication:

Requires an API key passed via request headers (e.g., Authorization:

Bearer <API_KEY>) **Data**

Exchanged:

- **Input:** Keyword or category (e.g., “agriculture”, “farming”)

- **Output:** JSON containing article title, description, URL, and publication date.

2. **Location/Mapping API** - for tracking and visualizing item locations, farm areas, and delivery routes.

Purpose:

To visualize locations of farms, storage facilities, and distribution routes within Liliw.

APIType:

REST API (e.g., Google Maps API or OpenStreetMap) **Endpoints**

Used:

- GET /maps/api/geocode/json?address=<location> - converts an address to geographic coordinates.
- GET
/maps/api/directions/json?origin=<origin>&destination=<destination> - generates route data for deliveries.

RequestType:

GET

Authentication:

Requires API key (added to URL or request headers) **Data**

Exchanged:

- **Input:** Address, coordinates, or route parameters

- **Output:** JSON containing latitude, longitude, map coordinates, or route details.

3. **Email & SMS Notification API** - for sending automated notifications to users regarding inventory status, requests, and updates.

Purpose:

To send automated notifications to staff and suppliers regarding stock alerts, approval updates, and delivery confirmations.

APIType:

REST API (e.g., Twilio for SMS, SendGrid for email) **Endpoints**

Used:

- **Email:** POST /v3/mail/send -sends email messages.
- **SMS:** POST /2010-04-01/Accounts/{AccountSID}/Messages.json - sends SMS notifications.

RequestType:

POST

Authentication:

Requires API credentials (Account SID, Auth Token, or API key) **Data**

Exchanged:

- **Input:** Recipient information, message subject/body
- **Output:** Response status confirming message delivery.

Data Flow and User Interface

The data flow begins when a user logs in and enters data (e.g., new stock details). This information is sent from the client to the server, where it is processed and stored in the MySQL database. When the user requests a report or inventory list, the system retrieves the necessary data from the database and displays it on the client side.

A typical user journey includes:

1. Logging into the system
2. Adding or updating inventory items
3. Viewing available stock and reports
4. Logging out

User Interface Design

The system interface is simple and user-friendly. It consists of:

Login Page: Secure access for staff and admins

Dashboard: Displays summary of items and notifications

Inventory Page: Add, edit, or delete stock records

Reports Page: Generate and print inventory summaries

CHAPTER III

IMPLEMENTATION

Technology Stack

The Digital Inventory System was developed using a combination of modern web technologies that ensure efficiency, reliability, and scalability. These technologies were carefully selected to make the system easy to maintain and accessible on multiple devices.

Technology/Tool	Purpose	Version
PHP	Backend scripting language	8.0
MySQL	Database server	8.0
HTML/CSS/JS	Frontend interface	Latest
Bootstrap	Responsive design	5.3
Apache	Web server	3.3.0
phpMyAdmin	Database management	5.2
VS Code	Code editor	Latest
Xdebug	Debugging tool	3.2

Table 1. Technology / Tool Description / Purpose Version

System Modules

The system was divided into several functional modules, each responsible for a specific part of the inventory process. This modular design allows easy updates and maintenance.

Module	Description / Responsibilities
User Management Module	Handles user registration, login, authentication, and role assignment (Admin or Staff)
Inventory Management Module	Core module responsible for adding, updating, viewing, and deleting agricultural items and supplies
Category Management Module	Organizes items into categories such as seeds, fertilizers, and tools for easier tracking
Transaction Module	Records and monitors stock movements (incoming and outgoing items)
Reporting Module	Generates printable inventory reports, summaries, and charts for decision-making
Dashboard Module	Displays real-time statistics and summaries of inventory levels and user activities
Backup and Restore Module	Allows secure saving and restoring of database files to prevent data loss

Table 2. Module, Description / Responsibilities

Testing and Debugging

Testing was an important step to ensure the system works correctly before deployment. The following testing methods were applied:

1. Unit Testing

Each module (e.g., login, add item, report generation) was tested individually to verify that it performs as expected.

Example: Checking if the login form correctly validates user credentials.

2. Integration Testing

After unit testing, modules were connected and tested together to ensure smooth data flow between components.

Example: Verifying that adding a new item automatically updates the report section.

3. User Acceptance Testing (UAT)

The system was tested by selected users from the Department of Agriculture–Liliw Office to ensure that it meets their needs and is easy to use.

Example: Testing how staff record new supplies and generate daily reports.

4. Debugging Tools Used

Xdebug: Used to trace errors and identify code issues in PHP. Browser

Developer Tools: Used to fix layout issues and JavaScript errors.

phpMyAdmin: Used to verify that database queries store and retrieve correct data.

Test Case	Expected Result	Actual Result
		Status
User logs in with correct credentials	System redirects to dashboard	Redirect successful Passed

Add new inventory item	Item appears in inventory list	Item successfully added Passed
Delete item	Item removed from list	Deleted successfully Passed
Generate report	PDF report displays correct data	Report generated accurately Passed

Table 3. Sample Test Case

Challenges and Solutions

During development, several challenges were encountered. Below are some of the major issues and how they were resolved:

Challenge	Description	Solution
Database Connection Error	Failed to connect due to wrong credentials	Updated config.php and reconnected
Form Validation	Invalid data stored	Added front-end and back-end validation
Slow Loading	Report generation delays	Optimized SQL queries and added pagination
UI Bugs	Misalignment on mobile screens	Used Bootstrap responsive grid

Table 4. Challenges & Solutions

APPENDICES

**Appendix A
Approved Letter**

Group 6

Bachelor of Science in Information Technology

3A-WAMD

Laguna State Polytechnic University-Sta. Cruz Campus, Laguna

October 13, 2025

Ms. Eunice Decena

Municipal Agriculturist

Department of Agriculture – Liliw Office

Liliw, Laguna

Subject: Request for Approval to Conduct a Project: Digital Inventory and Management System

Dear Mrs. Decena,

Good day!

We, **Group 6**, third-year students from the **Bachelor of Science in Information Technology** program at **Laguna State Polytechnic University – Sta. Cruz Campus**, are currently undertaking our project as part of our academic requirement in Client-Server Technologies.

We would like to respectfully request your approval and partnership in developing a **Digital Inventory and Management System** for the **Department of Agriculture – Liliw Office**. This system aims to assist your office in improving the management of agricultural records, streamlining the tracking of inventory, and enhancing the efficiency of day-to-day operations.

The proposed system will be designed with features tailored to your current processes and will be developed at no cost to your office. Our team will coordinate closely with you and your staff to gather the necessary data and feedback throughout the development stages, ensuring that the final system meets your expectations and operational needs.

We hope for your favorable response and approval to proceed with this project. We would be honored to work with your office and contribute to improving the digital processes of your department.

Thank you very much for considering our request. We look forward to your support and guidance throughout the project.

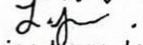
Respectfully yours,

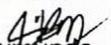
Group 6 – BSIT 3rd Year

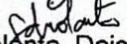
Laguna State Polytechnic University – Sta. Cruz Campus

Members:


Bonza, Godfrey


Javier, Lean Jay L.


Raymundo, John Railey


Violenta, Daisy M.

Client's Approval Section

I, **Ms. Eunice Decena**, Municipal Agriculturist of the Department of Agriculture – Liliw Office, hereby grant approval to Group 6 from LSPU – Sta. Cruz Campus to conduct and develop their Capstone Project entitled "**Digital Inventory and Management System**" in partnership with our office.

Approved by:

Eunice
Ms. Eunice Decena

Municipal Agriculturist

Date: 11 / 24 / 25

Appendix B
Photo Documentation

Action Photo



Formal Photo

