

# AIIMS: Agrikultura Information and Inventory Management System

## Software Engineering Project Document

Submitted by:

John Erick Frany

Jan Edgar Tupas

Jimuel Cacayan

Allen Kyle Sabilala

Mark Christian Sayson

Submitted to:

Engr. Antonette D. Gabriel

January 31, 2023

## TABLE OF CONTENTS

I.	Project Description	3
	a. Project Overview	3
	b. Background	3
	c. Purpose and Description of the Project	4
	d. Objectives	6
	e. Scope and Limitations	7
	f. Significance	8
II.	System Requirements Specification and Analysis	10
	a. Use Case Diagrams	10
	b. Use Case Narratives	11
	c. Functional Requirements	16
	d. Non-Functional Requirements	19
III.	System Design	21
	a. Proposed Software System	21
	b. User Interface Design	22
IV.	System Implementation	25
	a. Software Tools	25
	b. Hardware	25
	c. Application Modules	26

## **I. Project Description**

### **Project Overview**

The Agrikultura Information and Inventory Management System (AIIMS) is an inventory monitoring and management system that provides the inventory manager of Agrikultura Market with a user-friendly interface that allows for performing inventory-related tasks. These tasks include displaying and sorting all product records currently in Agrikultura Market's inventory, creating new product records, updating existing product records, and removing product records. AIIMS allows the inventory manager to view all products that are in low supply and to replenish the quantity of these products. AIIMS also allows the inventory manager to view all products that have already expired and to remove the expired products. The inventory manager can also generate inventory reports that contain information of product records in inventory. AIIMS can furnish the inventory manager with a product record report, product replenishing report, product removal report, and sold products report. Tracking products in Agrikultura Market's inventory becomes effective through the use of AIIMS.

### **Background**

Agrikultura Market is a general market that relies on food products and other available commodities to the public. As a food market, it aims to provide consumers with goods and products they need daily. As the Agrikultura market grows, it struggles to track its goods and products effectively. Agrikultura Market's previous way of inventory management is ineffective and inefficient for their growing business. Agrikultura Market manages its goods and assets by manually tracking its inventory using traditional means of written documents and observation. This process is inefficient as it requires significant amounts of time and human labor and is prone to human error. Human error can lead to easily missed and overlooked products or goods, leading to inaccurate inventory recordings and processes. Furthermore, Written documents can be lost, wet, or damaged since they are physical copies of the record. Incorrect or misleading information in the inventory system of the business leads to redoing the inventory management's whole recording, leading to more human labor needed to correct the inventory records.

The traditional pen-and-paper inventory management method used by Agrikultura Market currently wastes time and human power which are vital resources for the business. Errors in inventory management are also prevalent due to the insecure and inefficient method used. Errors in inventory management have significant impacts on how a business operates. Inefficient tracking and management of inventory can lead to the loss of essential resources, products, business capacity, and capabilities. This translates to a loss in sales and revenue, negatively impacting the overall business. The flow of business procedures, from identifying the required quantity of products to be ordered, ordering and acquiring products, storing products, and selling the products, could be disrupted when there are failures in the inventory management procedures.

Customer dissatisfaction becomes prevalent if the inefficient inventory management that Agrikultura Market uses is used continuously and is not replaced by an effective, efficient, and accurate method. Ensuring an accurate inventory management method is necessary for any business translating to smooth business transactions and procedures. Proper inventory management and tracking results in the appropriate record-keeping, acquisition, and sale of products that lead to customer satisfaction which is necessary to keep a business working and successful.

### **Purpose and Description of the Project**

The project is an information and inventory management system designed for the needs of Agrikultura Market. The project's primary function is to provide a computerized manner for the inventory processes of Agrikultura, such as sorting, managing, and tracking products in inventory through a management system and database. The system uses a database to hold records of products and their details within the inventory. The details include the product's name, status, selling price, acquisition price, quantity, acquisition date, and expiry date. The system consists of a login page, a home page, a page for managing inventory, a page for replenishing and removing products, a page for selling products, and a page for generating reports. The system allows the inventory manager to display, create, update, delete, sort, and manage records of products within the inventory in the manage inventory page. The inventory manager can replenish and remove products within the inventory in the replenish and remove

inventory page. Products can be sold in the sell inventory page . Additionally, the system can generate inventory reports for the inventory, which includes the current information regarding the product's record in inventory. AIIMS can generate a product record report, product replenishing report, product removal report, and sold products report. The product record report presents all current products in agrikultura market's inventory along with their record details. The product replenishing report presents all products in inventory that are in low supply and must be replenished. The product removal report presents all products in inventory that have expired and must be removed from inventory. The sold products report presents all products that have been sold by agrikultura market.

The system grants Agrikultura an automated solution for their inventory needs that allows them to effectively and efficiently manage, sort, and track products in their inventory. The records relating to Agrikultura's inventory are essential resources for this product. The use of a database supports the system's reliability. It establishes a robust data storage for the records and ensures that they are secure, can be accessed at all times, and will remain in the computer even after powering it down. Management, sorting, and displaying of records are also simplified through the use of a database. The system manifests ease of use as Agrikultura will significantly save time and human power as the functions of the inventory system simplify the inventory tasks and processes in the traditional method of pen and paper. The inventory system will also lessen and eliminate inventory errors produced in traditional inventory methods. Furthermore, to enhance human-computer interaction and ensure a smooth process between them, the system's graphical user interface encompasses design principles that are practical, consistent, and pleasing to the eyes of the user. Elements that carry out processes are designed for straightforwardness and clarity.

The inventory system caters explicitly to the inventory needs of Agrikultura Market. It is designed to meet the requirements and fulfill the objectives set by Agrikultura. The computerized system is a definite upgrade to the traditional inventory means that Agrikultura uses. It improves the overall inventory process of Agrikultura by providing a simple and straightforward, reliable, and efficient platform for managing,

sorting, and tracking the business' inventory. Today, businesses, corporations, and companies have shifted and utilized technology to increase their capabilities and save significant resources, especially time and human power. The system allows Agrikultura to participate in technological advancements in business, improving its management capabilities and capacity to serve its customers while effectively utilizing the resources they have to the fullest.

## **Objectives**

The project aims to develop a computerized inventory management system for Agrikultura Market that will be used to organize, store, track, and manage products in inventory. Inventory management is a crucial stage in monitoring commodity supply. The fundamental purpose of inventory management is to guarantee that all types of supplies are available whenever the production department requires them, preventing production from being halted or slowed due to a lack of resources. Thus, the objectives of the project encompass the following:

- To generate an effective and efficient inventory method as a solution to the inventory problems of Agrikultura Market.
- To develop a computerized information and inventory management system paired with a database that significantly improves the efficacy and efficiency of the inventory management methods of Agrikultura Market.
- To lessen and eliminate inventory errors produced by the traditional pen and paper means of inventory of Agrikultura Market by replacing it with a computerized information and inventory management system.
- To save the vital business resources of Agrikultura Market, including time and human-power, wasted because of the inefficient inventory management methods utilized.
- To design the inventory system in accordance with the inventory needs of Agrikultura Market.
- To ensure that the inventory system is capable of carrying out functions that will allow the inventory manager to sort, manage, and display records of products in inventory in the database.

- To ensure that the inventory system allows the inventory manager to create, delete, and update records of products in inventory in the database and display the records respectively.
- To ensure that the inventory manager can manage inventory, replenish and remove products in inventory, and sell products in inventory properly using the system
- To ensure that the inventory system can generate reports for the inventory manager that includes all products currently in inventory, products that need to be replenished, products that need to be removed, and products that have been sold.

The computerized inventory management system is necessary for effectively improving the overall inventory management capability and capacity and mitigating and eliminating losses of Agrikultura Market.

### **Scope and Limitations**

The project will be focused on making a computerized inventory management system specifically designed for the needs of the client, Agrikultura Market, and not any other marketplace. The records of products in the system will match the inventory and products sold in Agrikultura Market. A database will be used to store all necessary records. Furthermore, vital product descriptions that are needed for Agrikultura Market employees to know such as name or label, status, price, quantity, date of acquisition, and date of expiry will also be part of the management system. Each entity, relationship, and constraint in the system will be made to suit the products sold by Agrikultura Market. Additionally, updating, adding and removal of products will also be supported by the system to cater to Agrikultura Market's possible changes in products offered.

This will be Agrikultura Market's first time in implementing a computerized inventory management system. As a result, it is essential to make the transition to digital as smooth as possible to mitigate confusion and errors. This will be done by having a straight-forward and easy to understand graphic user interface using JAVA's GUI components. Additionally, employees will undergo a very brief training where they

will be guided and get familiar with the management system and how to utilize it. This training period will be one hour at most.

The group will be responsible for the development of the application, a brief training period for the client's employees, and setting up the initial database. However, the group is not responsible for the system's maintenance. Once the client is happy with the final product, there will be no additional maintenance service that will be offered. Additionally, the cost of maintaining third party servers involved in the database is not included in the group's service. This is due to time constraints and the nature of the project. The group cannot commit to an arbitrary long period of service such as maintenance since the team will dissolve by the end of the semester at the end of January 2023.

## **Significance**

Agrikultura Market needs help to establish an effective method for inventory management. It currently utilizes the traditional pen-and-paper inventory management method, which requires significant amounts of time and human power. The security of records is easily compromised because paper documents can easily be misplaced or destroyed. Thus, inventory management in Agrikultura Market is prone to errors such as inaccuracies in record-keeping, overlooked details, and lost or destroyed documents necessary for the business.

The information and inventory management system will replace the traditional pen-and-paper inventory management method used by Agrikultura Market, translating to an effective and efficient process for managing, sorting, and tracking Agrikultura's inventory. The system will save the business resources manifested in time and human power as a computerized inventory system automates and speeds up physical and logical tasks. Security in the record management and keeping is enforced and enhanced through the database used for the inventory system. Records will remain in the database and be accessible to the inventory manager at all times, ensuring reliability. The accuracy in record keeping is also enhanced because of the capabilities of the computer used for the inventory system.

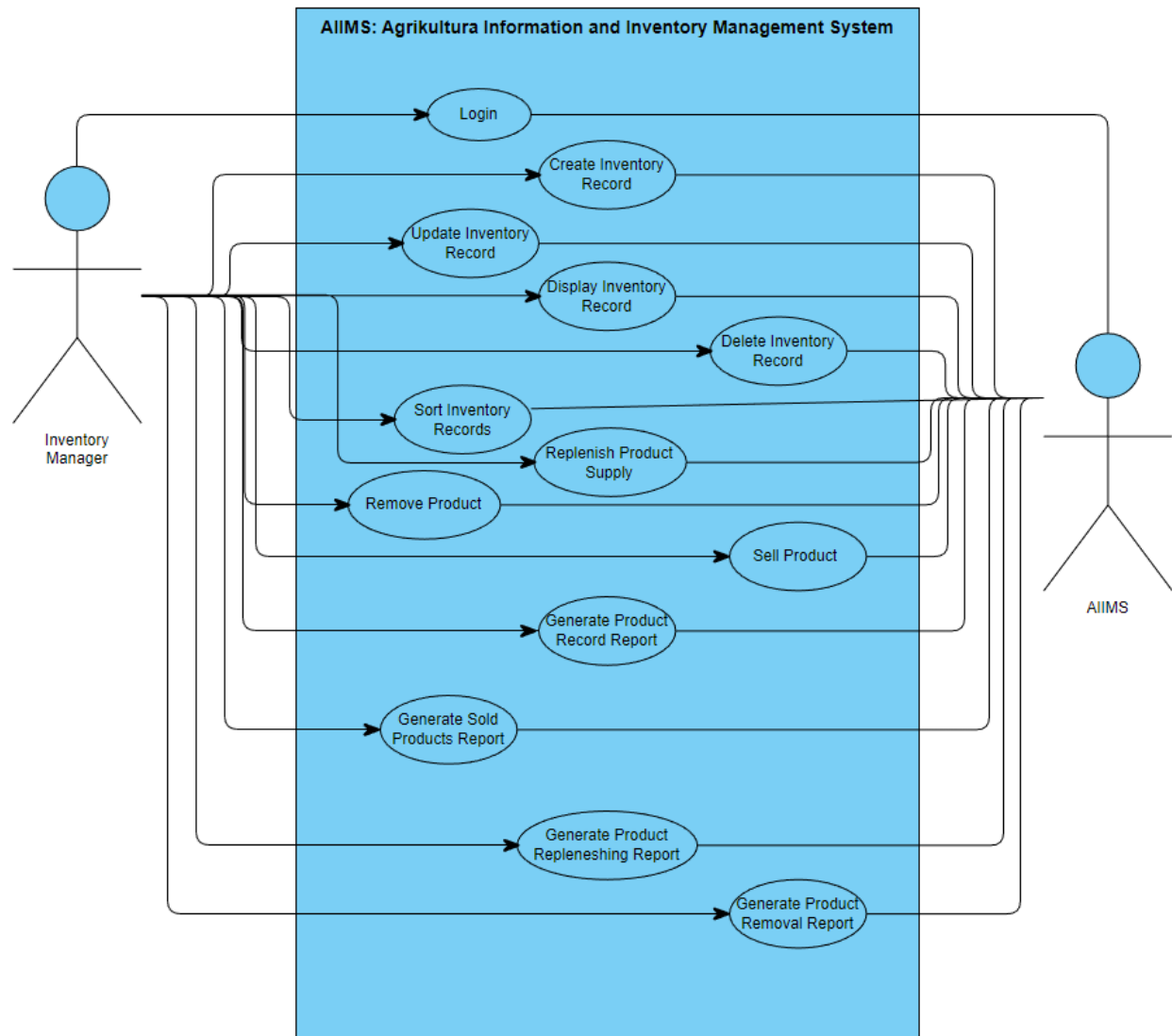


As mentioned, because the traditional inventory management method used by Agrikultura is inefficient, errors are prevalent and continuously occur. The inventory management system will lessen the requirement of human power exerted and simplify the functions of managing, sorting, tracking, adding, updating, deleting, and displaying records. This ensures that the likelihood of errors occurring is significantly lessened or even eliminated. The possibility of records being overlooked is reduced because records are presented in an organized manner on the computer. The consequences of losing physical documents are also minimized as the system can continuously generate inventory reports as needed by the inventory manager. The system's capability to create an inventory report for the inventory manager provides Agrikultura with secure and reliable documents they need to maintain and update their inventory when necessary. This function saves the business time and human power which would be required in excessive amounts if the business' inventory was done through the traditional method.

Despite the numerous capabilities of the inventory management system, the graphical user interface of the system will be designed to be simple and straightforward. The system's functions will be displayed with clarity to ensure that the user can quickly understand how to navigate and utilize the system and carry out the necessary tasks efficiently. This translates to quicker inventory management operations, improving the overall capacity and efficiency of Agrikultura Market. Thus, the information and inventory management system will effectively solve the inventory management problems faced by Agrikultura Market, contributing to its technological advancement in contemporary times.

## II. System Requirements Specification and Analysis

### Use Case Diagram



Use Case Diagram for AIIMS: Agrikultura Information and Inventory Management System Version 0.2

## Use Case Narratives

### Login

Actor Actions	System Responses
<ol style="list-style-type: none"><li>1. Inventory manager enters username</li><li>2. Inventory managers enters password</li><li>3. Inventory manager clicks login</li></ol>	<ol style="list-style-type: none"><li>4. Checks if username and password are correct</li><li>5. Shows prompt saying logging in</li><li>6. Show home page</li></ol>
Alternative Scenarios	
	<ol style="list-style-type: none"><li>5. Shows prompt saying username or password is incorrect</li></ol>

### Display Inventory Records

Actor Actions	System Responses
<ol style="list-style-type: none"><li>1. Inventory manager clicks on "Display Records"</li></ol>	<ol style="list-style-type: none"><li>2. Displays all inventory records from the database in a table</li></ol>

### Create Inventory Record

Actor Actions	System Responses
<ol style="list-style-type: none"><li>1. Inventory manager inputs product name or label</li><li>2. Inventory manager inputs status</li><li>3. Inventory manager inputs selling price</li><li>4. Inventory manager inputs acquisition price</li><li>5. Inventory manager inputs quantity</li><li>6. Inventory manager inputs date of acquisition</li><li>7. Inventory manager inputs date of expiry</li><li>8. Inventory manager clicks "Create Record"</li></ol>	<ol style="list-style-type: none"><li>9. Checks if all input fields are properly filled</li><li>10. Checks if input is correct based on the data type assigned to an attribute</li><li>11. Creates a new inventory record with the specified inputs and generates a numerical ID to it</li><li>12. Adds the new inventory record to the inventory database</li><li>13. Sends a confirmation message stating that the inventory record has been created and added to the inventory database successfully.</li></ol>
Alternative Scenarios	
	<ol style="list-style-type: none"><li>11. Sends an error message alerting that an input field is empty and must be filled</li></ol>

	11. Sends an error message alerting that an input is invalid due to incorrect data type 12. Returns to current page
--	--

#### Update Inventory Record

Actor Actions	System Responses
1. Inventory manager clicks on a record in the table 2. Inventory manager selects the attribute/s field 3. Inventory manager inputs data on the specified attribute/s field 4. Inventory manager clicks "Update Record"	5. Checks if all input fields are properly filled 6. Checks if input is correct based on the data type assigned to an attribute 7. Updates the data on the specified attribute/s in the specified record based on the data inputted 8. Sends a confirmation message stating that the inventory record has been updated.
Alternative Scenarios	
1. Inventory manager inputs the numerical ID of a record on the search field and clicks "Search" 4. Inventory manager selects the attribute/s field 5. Inventory manager inputs data on the specified attribute/s field 6. Inventory manager clicks "Update Record"	2. Searches for the record specified by the inventory manager using the numerical ID 3. Record with matching numerical ID specified is displayed and presented in a table while record attributes are displayed in the given input fields 7. Checks if all input fields are properly filled 8. Checks if input is correct based on the data type assigned to an attribute 9. Updates the data on the specified attribute/s in the specified record based on the data inputted 10. Sends a confirmation message stating that the inventory record has been updated.
	9. Sends an error message alerting that an input field is empty and must be filled

	9. Sends an error message alerting that the input is invalid due to incorrect data type 10. Returns to current page
--	--

#### Delete Inventory Record

Actor Actions	System Responses
1. Inventory manager clicks on a record in the table 2. Inventory manager clicks "Delete Record"	3. Checks if a record is selected 4. Deletes the selected record from the inventory database 5. Sends a confirmation message stating that the inventory record has been deleted from the inventory database successfully
Alternative Scenarios	
1. Inventory manager inputs the numerical ID of a record on the search field and clicks "Search" 4. Inventory manager clicks "Delete Record"	2. Searches for the record specified by the inventory manager using the numerical ID 3. Record with matching numerical ID specified is displayed and presented in a table while record attributes are displayed in the given input fields 5. Checks if a record is selected 6. Deletes the selected record from the inventory database 7. Sends a confirmation message stating that the inventory record has been deleted from the inventory database successfully
	6. Sends an error message stating that no record is selected 7. Returns to current page

## Sort Inventory Records

Actor Actions	System Responses
<ol style="list-style-type: none"> <li>1. Inventory manager clicks on “Sort By ID” button</li> <li>3. Inventory manager clicks on “Sort By ID” button again</li> </ol>	<ol style="list-style-type: none"> <li>2. Records are sorted by ID in ascending order.</li> <li>4. Records are sorted by ID in descending order</li> </ol>
<i>Alternative Scenarios</i>	
<ol style="list-style-type: none"> <li>1. Inventory manager clicks on “Sort By Name” button</li> <li>3. Inventory manager clicks on “Sort By Name” button again</li> </ol>	<ol style="list-style-type: none"> <li>2. Records are sorted by name in alphabetical order.</li> <li>4. Records are sorted by name in reverse alphabetical order</li> </ol>
<ol style="list-style-type: none"> <li>1. Inventory manager clicks on “Sort By Status” button</li> <li>3. Inventory manager clicks on “Sort By Status” button again</li> </ol>	<ol style="list-style-type: none"> <li>2. Records are sorted by status.</li> <li>4. Records are sorted by status in reverse order</li> </ol>
<ol style="list-style-type: none"> <li>1. Inventory manager clicks on “Sort By Selling Price” button</li> <li>3. Inventory manager clicks on “Sort By Selling Price” button again</li> </ol>	<ol style="list-style-type: none"> <li>2. Records are sorted by selling price in ascending order.</li> <li>4. Records are sorted by selling price in descending order</li> </ol>
<ol style="list-style-type: none"> <li>1. Inventory manager clicks on “Sort By Acquisition Price” button</li> <li>3. Inventory manager clicks on “Sort By Acquisition Price” button again</li> </ol>	<ol style="list-style-type: none"> <li>2. Records are sorted by acquisition price in ascending order.</li> <li>4. Records are sorted by acquisition price in descending order</li> </ol>
<ol style="list-style-type: none"> <li>1. Inventory manager clicks on “Sort By Quantity” button</li> <li>3. Inventory manager clicks on “Sort By Quantity” button again</li> </ol>	<ol style="list-style-type: none"> <li>2. Records are sorted by quantity in ascending order.</li> <li>4. Records are sorted by quantity in descending order</li> </ol>
<ol style="list-style-type: none"> <li>1. Inventory manager clicks on “Sort By Date of Acquisition” button</li> <li>3. Inventory manager clicks on “Sort By Date of Acquisition” button again</li> </ol>	<ol style="list-style-type: none"> <li>2. Records are sorted by date of acquisition in ascending order.</li> <li>4. Records are sorted by date of acquisition in descending order</li> </ol>
<ol style="list-style-type: none"> <li>1. Inventory manager clicks on “Sort By Date of Expiry” button</li> <li>3. Inventory manager clicks on “Sort By Date of Expiry” button again</li> </ol>	<ol style="list-style-type: none"> <li>2. Records are sorted by date of expiry in ascending order.</li> <li>4. Records are sorted by date of expiry in descending order</li> </ol>

### Search Inventory Records

Actor Actions	System Responses
<ol style="list-style-type: none"> <li>1. Inventory manager types product ID</li> <li>2. Inventory manager clicks on "Search" button</li> </ol>	<ol style="list-style-type: none"> <li>4. Table gets cleared of all data</li> <li>5. Displays inventory records from the database that matches the product ID typed by the Inventory Manager</li> </ol>
<i>Alternative Scenarios</i>	
	<ol style="list-style-type: none"> <li>4. Sends error message stating that the product ID entered does not exist</li> </ol>

### Replenish Product Supply

Actor Actions	System Responses
<ol style="list-style-type: none"> <li>1. Inventory manager clicks on a record in the table</li> <li>2. Inventory manager specifies quantity</li> <li>3. Inventory manager clicks replenish</li> </ol>	<ol style="list-style-type: none"> <li>4. Replenishes product quantity based on the input</li> <li>5. Displays prompt saying product has been replenished</li> </ol>

### Remove Product

Actor Actions	System Responses
<ol style="list-style-type: none"> <li>1. Inventory manager clicks on a record in the table</li> <li>2. Inventory manager clicks on "Remove" button</li> </ol>	<ol style="list-style-type: none"> <li>3. Removes product quantity based on the input</li> <li>4. Displays prompt saying product has been replenished</li> </ol>

### Sell Product

Actor Actions	System Responses
<ol style="list-style-type: none"> <li>1. Inventory manager clicks on a record in the table</li> <li>2. Inventory manager clicks on process</li> <li>5. Inventory manager clicks cash out</li> </ol>	<ol style="list-style-type: none"> <li>3. Product is added to purchased table</li> <li>4. Amount due is shown</li> <li>6. Displays prompt saying cash out was successful</li> </ol>

#### Generate Product Record Report

Actor Actions	System Responses
1. Inventory manager clicks product record report	2. A report is generated containing all products in inventory and their record details

#### Generate Sold Products Report

Actor Actions	System Responses
1. Inventory manager clicks sold products report	2. A report is generated containing all products that have been sold and relevant record details

#### Generate Product Replenishing Report

Actor Actions	System Responses
1. Inventory manager clicks product replenishing report	2. A report is generated containing all products that are in low supply and require replenishing and relevant record details

#### Generate Product Removal Report

Actor Actions	System Responses
1. Inventory manager clicks product removal report	2. A report is generated containing all products that are expired and must be removed, and relevant record details

### Functional Requirements

Name	Description	Version
Functional Requirements		
Display Current Inventory Records	This function allows the inventory manager to read and show the records currently in inventory of Agrikultura Market as specified in the database. The records encompass the following attributes:	Version 0.1



	<ul style="list-style-type: none"> <li>• Product ID</li> <li>• Name/Label</li> <li>• Status</li> <li>• Selling Price</li> <li>• Acquisition Price</li> <li>• Quantity</li> <li>• Date of Acquisition</li> <li>• Date of Expiry</li> </ul> <p>The inventory manager can review, manage, and analyze Agrikultura's inventory through this function.</p>	
Sort Inventory Records	This function allows the inventory manager to sort records in the inventory database. Records can be sorted according to the product's name, status, selling price, acquisition price, quantity, date of acquisition, and date of expiry. The records can be sorted in ascending, descending, or alphabetical order. The function allows the inventory manager to easily manipulate how information of Agrikultura's inventory is displayed.	Version 0.1
Create/Add Inventory Record	This function allows the inventory manager to create or add new records in the inventory database if there are new products added to Agrikultura's inventory. New records added must contain all the necessary data for every attribute enumerated in the Display Current Inventory Records function, otherwise the record will be invalid.	Version 0.1
Update/Edit Inventory Record	This function allows the inventory manager to update records in the inventory database if needed. Records with attributes that contain errors or are outdated can easily be fixed and updated using this function.	Version 0.1
Delete/Remove Inventory Record	This function allows the inventory manager to delete records in the inventory database. Records of products that are temporarily or	Version 0.1

	permanently removed or discontinued by the market can be deleted using this function.	
Replenish Product Supply	This function allows the inventory manager to replenish the supply of products in inventory when quantity is below 10. Quantity is replenished depending on the given input.	Version 0.2
Remove Product	This function allows the inventory manager to remove products from inventory that are expired.	Version 0.2
Sell Product	This function allows the inventory manager to manage sold products. Available products that are processed are removed from the available product table and moved to the purchased table.	Version 0.2
Generate Product Record Report	<p>This function allows the inventory manager to request the system to generate a report that includes details of Agrikultura's current inventory. The report encompasses the following information:</p> <ul style="list-style-type: none"> <li>• ID</li> <li>• Name</li> <li>• Status</li> <li>• Selling Price</li> <li>• Acquisition Price</li> <li>• Quantity</li> <li>• Acquisition Date</li> <li>• Expiry Date</li> </ul>	Version 0.2
Generate Sold Products Report	<p>This function allows the inventory manager to request the system to generate a report that includes details of products that have been sold. The report encompasses the following information:</p> <ul style="list-style-type: none"> <li>• ID</li> <li>• Name</li> <li>• Selling Price</li> <li>• Quantity</li> </ul>	Version 0.2

Generate Product Replenishing Report	<p>This function allows the inventory manager to request the system to generate a report that includes details of products that are in low supply and require replenishing. The report encompasses the following information:</p> <ul style="list-style-type: none"> <li>• ID</li> <li>• Name</li> <li>• Status</li> <li>• Acquisition Price</li> <li>• Quantity</li> </ul>	Version 0.2
Generate Product Removal Report	<p>This function allows the inventory manager to request the system to generate a report that includes details of products that are expired and must be removed from inventory. The report encompasses the following information:</p> <ul style="list-style-type: none"> <li>• ID</li> <li>• Name</li> <li>• Status</li> <li>• Quantity</li> <li>• Expiry Date</li> </ul>	Version 0.2

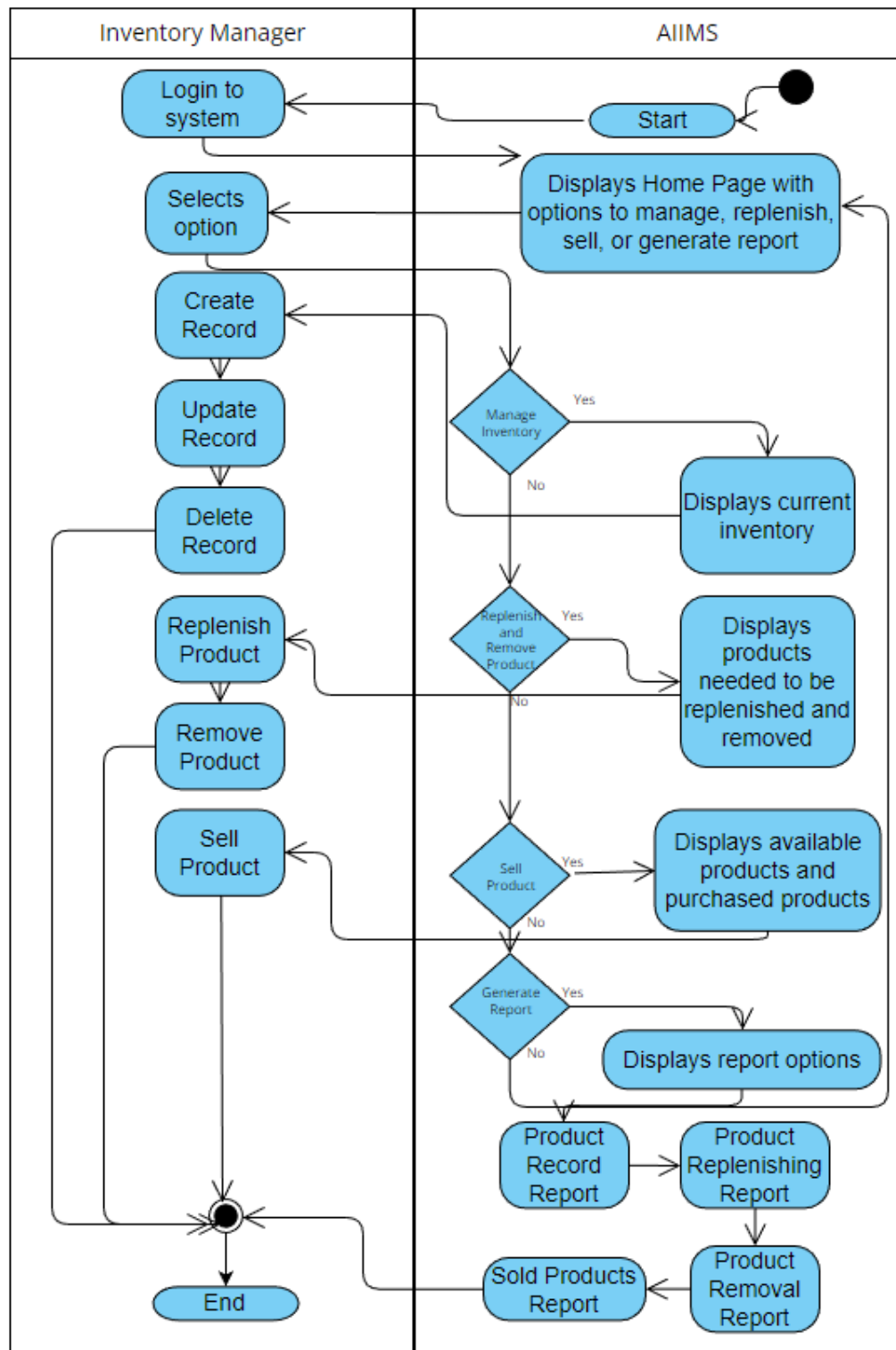
## Non-Functional Requirements

Name	Description	Version
Non-Functional Requirements		
Reliability	<p>This requirement ensures that AIIMS will be available to the inventory manager at all times. This allows managing and tracking of Agrikultura's inventory whenever necessary.</p>	Version 0.1
Ease of Use	<p>This requirement ensures that the system is simple and can quickly and easily be used by the inventory manager. This is necessary to lessen and avoid errors and confusion in using the system. A maximum of one hour of</p>	Version 0.1

	training is sufficient for the inventory manager to effectively utilize the system.	
Appearance	<p>This requirement establishes the graphical user interface (GUI) of the system. It ensures that the GUI is straightforward. The requirement specifies the following:</p> <ul style="list-style-type: none"> <li>• GUI must be equipped with simple designs</li> <li>• Designs must be pleasing and not distracting resulting in the user's ineffective usage of the system</li> <li>• Designs should be consistent</li> <li>• Designs and icons should be familiar to the user</li> <li>• System should provide informative feedback</li> <li>• Buttons that carry out specific functions must be recognizable, direct, and clear</li> </ul>	Version 0.1

### III. System Design

#### Proposed Software System



Activity Diagram for proposed software system of AIIMS

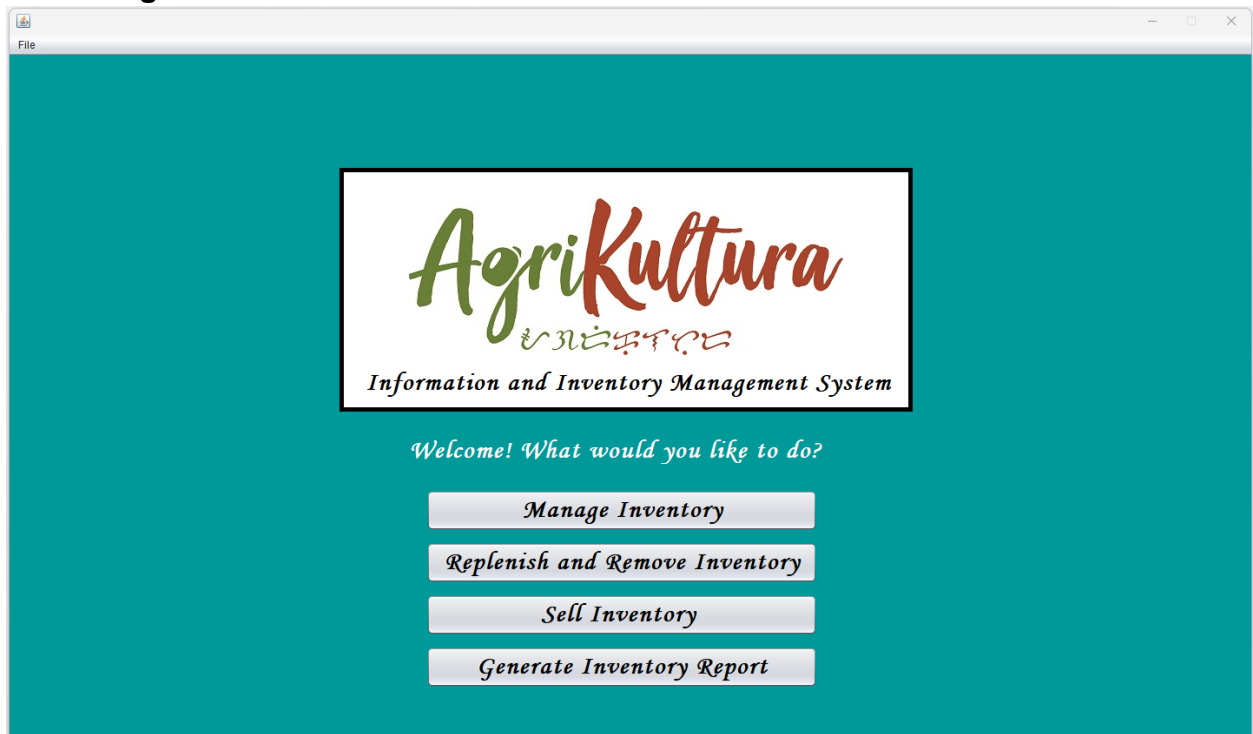
## User Interface Design

### Login Page




A screenshot of a web browser window displaying the login page of the AgriKultura Information and Inventory Management System. The page has a teal background. In the center, there is a white rectangular box containing the system's logo. The logo features the word "AgriKultura" in a stylized font, with "Agri" in green and "Kultura" in brown. Below the name is a small green emblem. Underneath the emblem, the text "Information and Inventory Management System" is written in a smaller, black, serif font. Below the logo box, there are two input fields: one for "Username:" and one for "Password:". Both fields are white with a thin black border. Below the password field is a white button with the text "Log in" in a black, italicized font.

### Home Page



A screenshot of a web browser window displaying the home page of the AgriKultura Information and Inventory Management System. The page has a teal background. In the center, there is a white rectangular box containing the system's logo, which is identical to the one on the login page. Below the logo box, the text "Welcome! What would you like to do?" is displayed in a black, italicized font. Below this text are four white buttons with black borders, each containing a different option in a black, italicized font: "Manage Inventory", "Replenish and Remove Inventory", "Sell Inventory", and "Generate Inventory Report".

## Manage Inventory Page



# Information and Inventory Management System

### Available Products

ID	Name	Status	Selling Price	Acquisition Price	Quantity	Acquisition Date	Expiry Date
686	TALONG	Good	130.00	99.00	20	2023-01-29	2023-02-02
948	BEANS	Good	90.00	68.00	10	2023-01-29	2023-02-03
1169	PAPAYA GREEN	Good	40.00	28.00	10	2023-01-29	2023-02-03
1178	PATATAS	Good	90.00	65.00	15	2023-01-29	2023-02-12
1284	KANGKONG	Good	20.00	13.00	15	2023-01-29	2023-02-02
1306	KIVCHAY	Good	15.00	10.00	10	2023-01-29	2023-02-02
1325	LUMPIA WRAP...	Good	15.00	8.00	25	2023-01-29	2023-02-12
1384	AMPALAYA	Good	100.00	69.00	5	2023-01-29	2023-02-02
1474	TILAPIA	Good	165.00	120.00	10	2023-01-29	2023-02-01
1740	PETSAY	Good	20.00	12.00	10	2023-01-29	2023-02-02
1757	MUSTASA	Good	20.00	13.00	10	2023-01-29	2023-02-02
1799	SAYOTE	Good	45.00	34.00	5	2023-01-29	2023-02-05
1786	OKRA	Good	10.00	5.00	15	2023-01-29	2023-02-02
1793	AGOOT	Good	295.00	215.00	5	2023-01-29	2023-02-01
2590	PECHAY BAG...	Good	60.00	43.00	5	2023-01-29	2023-02-01
2843	PUSO NG SAG...	Good	50.00	35.00	5	2023-01-29	2023-02-03
3208	CARROTS	Good	150.00	112.00	10	2023-01-29	2023-02-03
3618	BELL PEPPER	Good	240.00	180.00	10	2023-01-29	2023-02-03
3680	DALAGA BUKID	Good	295.00	225.00	8	2023-01-29	2023-02-01
3970	SALUYOT	Good	20.00	14.00	15	2023-01-29	2023-02-02
4833	ALUGBATI	Good	20.00	12.00	15	2023-01-29	2023-02-02
5123	SAMARAL	Good	350.00	260.00	3	2023-01-29	2023-02-01
5331	SILI SAGING	Good	10.00	5.00	15	2023-01-29	2023-02-03
5452	PAMPAÑO	Good	370.00	280.00	8	2023-01-29	2023-02-01
5619	MELON	Good	95.00	72.00	10	2023-01-29	2023-02-08
5854	LABANOS	Good	85.00	64.00	10	2023-01-29	2023-02-03
6049	SPRING ONION	Good	15.00	10.00	10	2023-01-29	2023-02-02
6156	SITAW	Good	20.00	13.00	15	2023-01-29	2023-02-02
6324	LUYA	Good	120.00	85.00	15	2023-01-29	2023-02-02
6617	MANGGA	Good	195.00	150.00	12	2023-01-29	2023-02-05
6679	TALBOS CAMO...	Good	20.00	9.00	10	2023-01-29	2023-02-02
6939	HIPON	Good	560.00	400.00	7	2023-01-29	2023-02-01
7018	REPOLYO	Good	60.00	43.00	10	2023-01-29	2023-02-02
7049	TOKWA	Expired	7.00	4.00	7	2023-01-29	2023-01-30
7920	TOGUE	Good	35.00	24.00	15	2023-01-29	2023-02-01
8286	ONION RED	Good	145.00	110.00	15	2023-01-29	2023-02-12
8298	TAHONG	Good	130.00	95.00	5	2023-01-29	2023-02-01
8652	LETTUCE	Good	190.00	150.00	10	2023-01-29	2023-02-02
8995	CALAMANSI	Good	100.00	70.00	20	2023-01-29	2023-02-03
9254	YELLOW FIN	Good	280.00	210.00	5	2023-01-29	2023-02-01

### Manage

Create

Update

Delete

Reset

### Search

ID:

Search

### Record Data

Name:

Status:

Selling Price:

Acquisition Price:

Quantity:

Acquisition Date:

Expiry Date:

Generate Report

### Sort By

ID

Name

Status

Selling Price


Acquisition Price

Quantity

Acquisition Date

Expiry Date

## Replenish and Remove Inventory Page



# Information and Inventory Management System

### Products to be Replenished

ID	Name	Status	Acquisition Price	Quantity
1384	AMPALAYA	Good	69.00	5
1769	SAYOTE	Good	34.00	5
1793	AGOOT	Good	215.00	5
2590	PECHAY BAGUIO	Good	43.00	5
2843	PUSO NG SAGING	Good	35.00	5
3680	DALAGA BUKID	Good	225.00	8
5123	SAMARAL	Good	260.00	3
5452	PAMPAÑO	Good	280.00	8
6939	HIPON	Good	400.00	7
8298	TAHONG	Good	95.00	5
9254	YELLOW FIN	Good	210.00	5

### Replenish Products

ID:

Name:

Quantity:

Replenish

Reset

Generate Report

### Products to be Removed

ID	Name	Status	Quantity	Expiry Date
7049	TOKWA	Expired	7	2023-01-30

### Remove Products

ID:

Name:


Quantity:

Remove

Reset

Generate Report

## Sell Inventory Page



# Information and Inventory Management System

### Available Products

ID	Name	Selling Price	Quantity
686	TALONG	130.00	20
948	BEANS	90.00	10
1169	PAPAYA GREEN	40.00	10
1178	PATATAS	90.00	15
1284	KANGKONG	20.00	15
1306	KINCHAY	15.00	10
1325	LUMPIA WRAPPER	15.00	25
1384	AMPALAYA	100.00	5
1474	TILAPIA	165.00	10
1740	PETSAI	20.00	10
1757	MUSTASA	20.00	10
1769	SAYOTE	45.00	5
1786	OKRA	10.00	15
1793	AGOOT	295.00	5
2590	PECHAY BAGUID	60.00	5

### Add to Cart

Product ID:

Name:

Quantity:

Price:

### Purchased

ID	Name	Selling Price	Quantity
----	------	---------------	----------

### AMOUNT DUE

## Generate Inventory Report Page



# Information and Inventory Management System

Welcome! Which report would you like to be generated?



## **IV. System Implementation**

### **Software Tools**

Java was the programming language used for the creation of AIIMS. Java is a class-based and object-oriented high-level programming language. These aspects make it easy to distribute tasks among the individual group members. Each member can work on an assigned class or form, then have it easily combined together during the finalization of the project. Furthermore, creating a product class with attributes that match the inventory records of Agrikultura made it much easier to execute methods that required the data from the database. Certain objects can simply use the variables in the product class instead of needing to be declared again. Java also supports the implementation of an SQL which was used to connect the Java application to the SQL database. The JasperReport plug-in was used with Java to create and generate the inventory reports of the system.

Netbeans IDE was the integrated design environment used to create the Java program. It was used to create the interface and program the events that will occur when UI components such as buttons and menu bars are interacted with. Netbeans IDE has a drag-and-drop feature where programmers can simply drag and place GUI components onto a form. This makes designing the interface much easier and faster since components can be placed without the need for typing it manually. Furthermore, having a visualization of the UI before it is finalized makes the conceptualization and designing of the interface much easier. Netbeans IDE is also freeware meaning all of our members will not have to pay to use it during the making of the project.

MySQL and MySQL workbench was used to design the database that consisted of the inventory records of Agrikultura Market. MySQL was used due to it having fast query processing, accessibility, and ability to be embedded in Java applications. Furthermore, it is free and can be accessed by all members of the group.

### **Hardware**

The hardware used to create the system were the laptops and personal computers of the team. All IDE's, programming languages, MySQL data, and library files

were collected online and utilized using laptops and personal computers. Testing and debugging the system were also performed using the team's laptops and personal computers.

## **Application Modules**

### **Login Module**

This module is used as a security measure that checks if the user is allowed access to the system if their username and password is registered.

### **Home Module**

This module gives the inventory manager access to the main functions of the system by displaying four buttons that directs the inventory manager to the different pages of the application. From this they can choose to "Manage inventory", "Replenish and Remove Inventory", "Sell Inventory", and "Generate Inventory Report". They will be directed to the respective pages upon button click.

### **Manage Inventory Module**

This module is where the inventory database is displayed and edited. When it is first accessed, the main table will display all of the product information within the current database. The information displayed can be sorted by ID number, name, status, selling price, acquisition price, quantity, acquisition date, or expiry date. Furthermore, this is also where the inventory manager can create, reset, update, and delete (CRUD) the contents of the database. Finally, this form has a search function that allows the inventory manager to track an item in the database through their unique product ID.

### **Replenish and Remove Inventory Module**

This module is used for when the inventory manager wishes to view and update what products need to be replenished due to low amount of quantity and what products should be removed from the inventory due to them being expired. The inventory manager can update the quantity of a product once it has been replenished by using the replenish button. Furthermore, there is also a remove button that is used to update expired products that have already been removed from the inventory. Both the "to be

replenished” and “to be removed” tables can be refreshed through the use of the reset button that updates the display with the changes that the inventory manager has made through the add and remove functions.

### **Sell Inventory Module**

This module is used for updating the database when customers purchase products from the shop. The inventory manager can indicate what product is going to be purchased, the product ID, the quantity, and the price by clicking the product in the table. The product can then be added to cart where the item being sold is removed from the database and onto the customer’s cart. Transaction is then finalized with the cash out button.

### **Generate Inventory Report Module**

This module gives the inventory manager access to generating inventory reports. The inventory managers may request the system to generate a product record report, product replenishing report, product removal report, or sold products report. The product record report generates a report of all products currently in inventory. The product replenishing report generates a report of all products that are in low supply and require replenishing. The product removal report generates a report of all products that are expired and must be removed from inventory. The sold products report generates a report of all products that have been sold. The generated report can be on jasper, pdf etc.

### **MySQL module**

This module connects the SQL database to the java application. It also contains methods used in the application such as the methods responsible for the CRUD functions, displaying tables, and sorting tables.

### **Product Module**

This module contains the attributes of products in the inventory. This is so that the attributes will not need to be declared individually in every module.

## **Sell Product Module**

This module contains the attributes of products in the inventory that have been sold. This is so that the attributes will not need to be declared individually in every module.