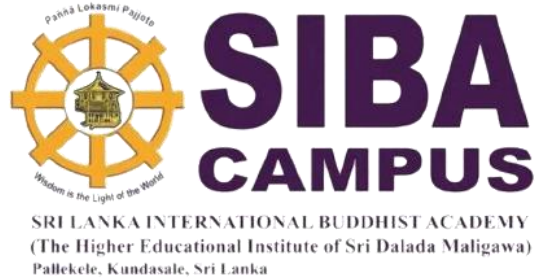# SWA Company Inventory Management System

Project Documentation

Inventory Management System

IT240 - Advanced Programming with Java I

Semester 2025 (I)

SRI LANKA INTERNATIONAL BUDDHIST ACADEMY
(The Higher Educational Institute of Sri Dalada Maligawa)
Pallekele, Kundasale, Sri Lanka

A.M. SANDARU WIMANSA ATHTHANAYAKA

BSC/WD/23/44/05

Submission Date: 11/05/2025

Table of Contents

Introduction

This project aims to develop a simple yet efficient Inventory Management System using Java Swing and a relational database. The system is designed to help users manage inventory by performing basic CRUD operations such as adding, editing, deleting, and viewing inventory items. The application also includes user authentication and basic search functionality, showcasing an understanding of GUI design, event handling, and database integration.

System Overview

The Inventory Management System provides a graphical user interface that allows users to manage stock records in an organized and user-friendly manner. It supports operations such as creating, reading, updating, and deleting inventory records stored in a relational database. The system is ideal for small to medium businesses looking to digitally track and maintain their inventory.

Main Objectives:

- Implement an interactive Java Swing GUI.
- Enable full inventory CRUD functionality.
- Persist data using a database.
- Provide search and validation features.
- Apply the MVC design pattern for clean code separation.

Features Description

Feature	Description
User Authentication	Allows users to register and log in to the system securely.
Add Inventory Item	Users can input new inventory items including item code, name, quantity, price, and supplier.
View Inventory List	Displays all items in a JTable with automatic database fetching.
Update Inventory Details	Allows users to edit existing item records.
Delete Inventory Item	Enables removal of selected inventory items.
Search Functionality	Users can search items using item name or item code.
Input Validation	Checks for required fields and numeric values where necessary.
Error Handling & Tooltips	User-friendly error messages and helpful tooltips for inputs.

Technologies Used

- **Programming Language:** Java
- **Database: MySQL**
- **IDE:** NetBeans IDE 8.2
- **Libraries:** Java Swing, JDBC
- **Design Pattern:** MVC (Model-View-Controller)

Implementation Details

User Authentication

- Registration form stores user credentials into the database.
- Login form verifies user input using prepared SQL statements to prevent SQL injection.

Inventory CRUD

- Add: Collects form data, validates inputs, and inserts into database.
- Update: Pre-fills form with selected record, allows editing, and updates DB.
- Delete: Deletes the selected row from the JTable and database.
- Clear: User can clear text fields.
- Refresh: User can refresh the page.

Search Functionality

- Implements LIKE SQL operator to filter by item name or code.

Validation

- Validates email addresses.
- Password must be in 8 characters.
- Ensures no field is left empty.

Testing

Manual testing was conducted for the following:

- **Login/Registration**: Tested for valid and invalid credentials.
- **Add Inventory**: Confirmed successful entry and error handling for missing fields.
- **Edit & Delete**: Verified correctness of updates and removal of records.
- **Search**: Validated for both full and partial matches.
- **Database Integration**: Ensured correct data flow and updates.

No major bugs were identified after final testing.

Instructions to Run the System

1. Install **JDK** and **NetBeans IDE**.
2. Install and configure **MySQL Server**.
3. Create a database named: **swainventory.**
4. Export the SQL tables (user & inventory) in this folder to the relevant database.
5. Open the project in NetBeans.
6. Run the application via the main class: MainPage.java.
7. Then click the Get Started button.
8. If you don't have an account, click the Register button to create one.
9. Then log in with your username and password.
10. Now you can add, update, and delete items as needed.
11. If you make a mistake while typing an item, you can use the Clear button to clear the text fields.
12. Through the search option, you can select the relevant item from the table by entering the item code or item name.
13. You can refresh the page again via the refresh button.

Screenshots

Main Inventory Window

Login Page

Register Page

Inventory Page

Conclusion

The Inventory Management System successfully demonstrates core programming concepts such as GUI design, event handling, and database interaction using Java Swing and MySQL. It helped enhance understanding of object-oriented programming, MVC structure, and real-world application design. The system is extendable and can serve as a foundation for future feature additions like report generation or multi-user access.