User Guide and Project Documentation

University of Balamand

Faculty of Sciences

Department of Computer Sciences

Java Technology – CSIS231

By David Kharrat

User Manual

Version 1.0

5/11/2024

Table of Contents

1. Introduction

This User Manual delivers a quick explanation about the purpose of the project and provides information necessary for developers to run it and further expand upon it with an efficient and comprehensive manner.

1.1 Purpose:

The primary purpose of this Java tech project is to create a cost-effective and dependable inventory management system for clients. It involves utilizing sophisticated technologies like Spring API, JavaFX, JavaFXML, and 3D Paint to build a robust system that efficiently handles inventory tracking and user interactions. The project's goal is to provide clients with a solution that is not only affordable but also reliable and user-friendly, showcasing our expertise in leveraging cutting-edge technologies for practical business applications..

1.2 Overview:

This project utilizes a combination of technologies to create an efficient inventory management system.

Technologies Used

Backend: Spring API

- Advantages:

- Robust framework for building enterprise-level applications.
- Provides features like dependency injection, MVC architecture, and transaction management.
- Simplifies RESTful API development with annotations and Spring Boot.
-Overall reliable and security heavy backend service

Frontend: JavaFX and JavaFXML

- Advantages:

- JavaFX offers a rich set of UI components for creating modern and interactive user interfaces.
- JavaFXML allows for declarative UI design, separating the presentation layer from the logic.

- Provides flexibility and customization options for designing user-friendly interfaces.

Database: MySQL

-Advantages:

-Reliable and widely-used relational database management system.

-Supports structured query language (SQL) for efficient data manipulation and retrieval.

-Scalable and suitable for handling large volumes of data in an inventory management system.

Design: 3D Paint

- Advantages:

- Enables creating visually appealing designs and graphics for the application.
- Intuitive interface for designing 3D models, icons, and graphical elements.
- Integration with JavaFX allows seamless incorporation of 3D designs into the UI.

2. Getting Started

In this section, We are going to go into detail of the many features that this inventory management system includes

2.1 Features:

-Well Structured Database:

-Accounts: Manage user authentication and access control.

- Inventory: Track and manage inventory items, including stock levels and details.

- OrderDetails: Capture detailed information about orders, such as product quantities and prices.

- Ordering: Stores information about all orders going into the inventory

- Orders: Stores information about all orders going out of the inventory

- Product: Maintain a comprehensive database of products and its details.

-Suppliers: Stores all information about suppliers.

- User Authentication:

-User can create an account, and login to the application to access its features

- Secure login system for authorized access.

- Inventory Management:

-Add and remove orders with their orderdetails

- Add and remove products

- Add and remove suppliers

- Reporting and Analytics:

- Track and manage orders that go in and out of the inventory

-Track current inventory value

-Track inventory in and out values

-Visualise monthly inventory in and out in an interactive graph

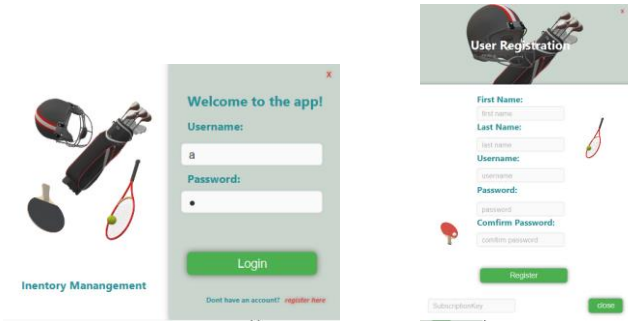
-Track inventory list

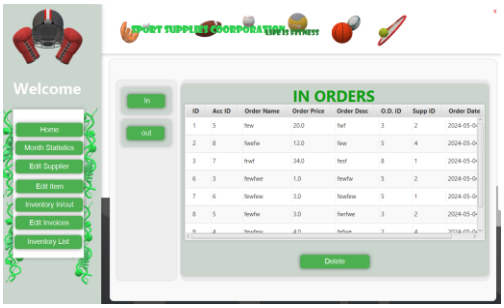
-Track available suppliers

-Track available products

-User Interface:

- An appealing user interface that welcomes the user into the application.





- User friendly and easy to understand interface, specifically designed for users that are less familiar with technology.

-Interactive features that guide users through tasks, ensuring a smooth and enjoyable interaction with the system.

3. Using the System

The following will be a detailed explanation, (a step-by-step guide) on how to run this Java tech project, including the Spring Boot API, JavaFXML frontend, and MySQL database.

3.1 How to run, Implementation

1. Prerequisites:

Before we start, there are some prerequisites that need to be downloaded , these include:

- Java Development Kit (JDK) installed on your system.
- Integrated Development Environment (IDE) like IntelliJ IDEA or Eclipse.
- MySQL Server installed and running.

2. Import the Project into IDE:

- Open your IDE (IntelliJ IDEA or Eclipse).
- Import the project by selecting the option to import from existing sources.
- Choose the project directory that you cloned or downloaded.

3. Set Up MySQL Database:

- Open MySQL Workbench or any MySQL client.
- Create a new database for your project (named: "davidapp").
- Import the database file provided with the project (if available) using the import tool or execute SQL scripts to create tables and populate data.

5. Configure Database Connection:

- In the project, locate the configuration file for database connection
- Update the database connection details such as URL, username, and password to match your MySQL database setup.

(This part might me the most challenging because any mistake in the connection would make the application unusable.)

6. Run the Spring Boot API:

- Open the API-Spirng folder in your IDE
- Locate the main Spring Boot application file (In this case it is named ApiApplicationTests)
- Right-click on the file and select "Run" or "Debug" to start the Spring Boot application.
- Verify that the API is running by checking the console output for any startup messages or errors.

When this is done, Congratulations, you would be done with the Back-End.

7. Launch the JavaFXML Frontend:

- Open the JavaFXML(frontend) file in a second project IDE.
- Navigate to the main class or launcher class for the frontend application.
- Right-click on the file and select "Run" or "Debug" to launch the JavaFXML frontend.
- The frontend application should start, displaying the user interface for the inventory management system.

8. Access the Application:

- Open the JavaFXML application window.

- Log in using the provided credentials or create a new account if required.

9. Test Functionality:

To make sure that all services are correctly connected:

- Once logged in, test the functionality of the application, such as adding, updating, and deleting inventory items.
- Verify that data is being stored and retrieved correctly from the MySQL database.
- Explore different features of the application to ensure everything is functioning as expected.

If everything runs correctly, Congratulations! You now have a functioning inventory management system!

4. Troubleshooting & Support

4.1 Error Messages

-Error messages are logged in the console output of the IDE or application server.

-Common errors include database connection issues, missing dependencies, or configuration errors.

-Refer to the error messages for guidance on resolving specific issues.

4.2 Special Considerations

-Ensure that all dependencies are properly installed and configured.

-Check compatibility between Java versions, Spring Boot versions, and MySQL versions.

-Pay attention to security considerations, such as securing database credentials and API endpoints.

4.3 Support

-For technical assistance, refer to online documentation, forums, or Stack Overflow.

-Reach out to the project's developers or community for support and troubleshooting.

-Consider consulting with experienced Java developers or IT professionals for complex issues.