Strathmore University

School of Computing and Engineering Sciences

**Master of Science in Information Technology**

**MIT8102:** Advanced Distributed Systems

**Assignment:** Fruit Service Engine

**Date:** 6th July, 2025

**GROUP MEMBERS**

| NAME | ADMISSION NUMBER |
|---|---|
| Chepsol Sammy Kiplagat | 224161 |
| Olela Wafula James | 112757 |

## 1. Project Overview

The Fruit Service Engine is a Java-based distributed mobile web based application that allows users to manage fruit prices. It supports operations like adding, updating, deleting, and calculating fruit costs. The system is built using Java Servlets, Java RMI, JDBC, MySQL, and HTML/CSS for User Interfaces (UI).

## 2. Features Implemented

- Add Fruit Price – Adds a new fruit and price to the database
- Update Fruit Price – Updates the price of an existing fruit
- Delete Fruit – Deletes a fruit record from the database
- Calculate Cost– Computes the cost based on fruit and quantity
- Generate Receipt – Generates transaction receipt including cashier info and change due

## 3. Technologies Used

Backend Server: Java RMI

Web Layer: Java Servlets

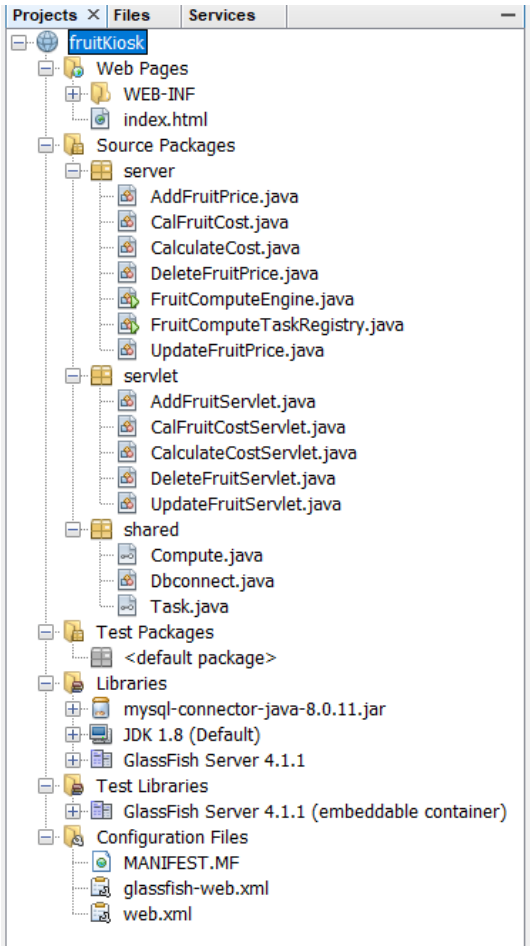
Database:  MySQL

Client Interface: HTML + CSS

IDE: NetBeans 8.2

Web Server: GlassFish Server

## 4. Key Files and Structure




- **fruitKiosk** - this is the project root folder
- **web.xml** – Servlet configuration file
- **index.html** – Main user interface for client interaction via the web browser
- **/shared/** – has common shared task classes i.e. Compute, Task and Database Utility
- **/server/** – has RMI engine and task class implementation classes
- **/servlet/** – Servlet classes for Add, Update, Delete, Calculate Fruit Cost and Calculate Cost
- **/Libraries/** - has libraries that facilitates mysql, jdk and GlassFish server communications

- fruitdb.jdb – javadb table schema for the fruit database

**5. Setup Instructions**

i. Import project into NetBeans

ii. Ensure java database on the netbeasn ide is running with the fruitdb database and table (fruit_price)

iii. Run FruitComputeEngine to Start the RMI registry on port (1099) and register RMI service

iv. Deploy the web app using NetBeans GlassFish Server

v. Access UI at: http://localhost:8080//fruitKiosk/web/index.html

vi. Use the web UI to test all five client tasks

**6. Challenges**

- Any change made required to clean and rebuild for the changes to take effect.

a. RMI issues

- RMI server would return null pointer exception error after timing out which prompted for a restart of the service every time the error occurred in order to complete client requests

**7. Conclusion**

The Fruit Service Engine is a completed and working distributed application with servlet-RMI integration and JavaDB backend that is persistent. All client tasks are executed and the results are as expected.

**References**

Arjun Yonjan / CodeTravel. (2020, October 19). *RMI Tutorial in Java | Java Remote Method Invocation | Java RMI step by step* [Video]. YouTube. https://www.youtube.com/watch?v=NPIRiKxLGr0

Baeldung. (2021, May 26). *Introduction to Java RMI*. https://www.baeldung.com/java-rmi

David Mbugua. (2022, March 24). *Java RMI full tutorial | Remote Method Invocation | Distributed computing | Java advanced* [Video]. YouTube. https://www.youtube.com/watch?v=scio4KeDwt4

GeeksforGeeks. (n.d.). *Remote method invocation (RMI) in Java*. https://www.geeksforgeeks.org/java/remote-method-invocation-in-java/

Java Code Geeks. (n.d.). *Java RMI – Java remote method invocation example*. https://examples.javacodegeeks.com/java-rmi-java-remote-method-invocation-example/

Oracle. (n.d.). *Java RMI tutorial*. https://docs.oracle.com/javase/tutorial/rmi/index.html