



Software Engineering

SUPERMARKET ERP SYSTEM

Team:

Keisi Breshanaj

Dejvi Kocilja

Ifigjenia Sopiqoti

Loard Bejko



INTRODUCTION

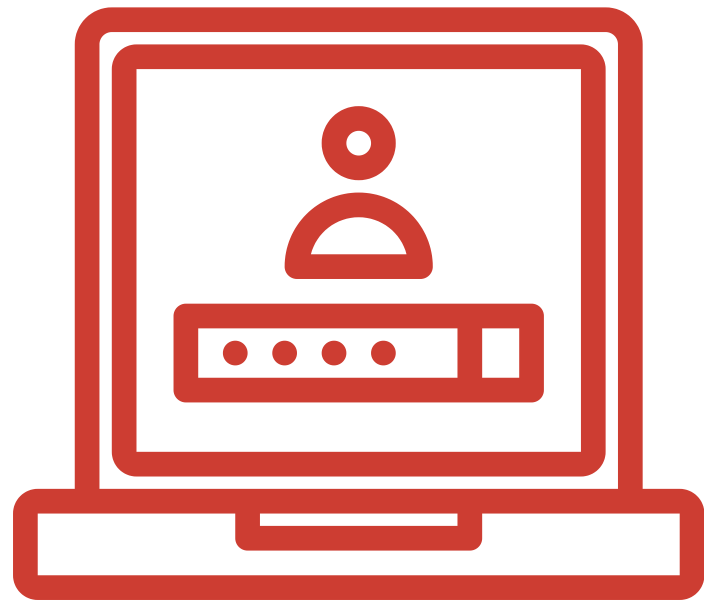
Name: EasyMart

- **Aim:** The aim of our project is to create a user-friendly Supermarket ERP software that simplifies and enhances overall supermarket management.

APPLICATION FEATURES:

- **Seamless User Experience:** EasyMart will boast an easy-to-navigate and user-friendly interface, ensuring that each user be it a cashier, manager, or administrator can effortlessly carry out their tasks.
- **Integration of Modules:** The software integrates distinct modules tailored to the unique responsibilities of each user role.
- **Loyalty Points System:** Recognizing the significance of customer loyalty in the retail landscape, EasyMart implements a loyalty points system. This feature rewards customers based on their purchase history.

OBJECTIVES



Develop a secure login system for Cashiers, Managers, and Administrators.



Implement modules for sales transactions, inventory management, and staff supervision.



Create a loyalty points system to encourage customer retention.



TEAM WORK

For this project, we chose to use Java as the programming language because it's widely supported and known for its reliability. For the front-end design, we turned to JavaFX.

LOARD BEJKO

Selection and design of a suitable development model for EasyMart.

Development of the interfaces for each user of the system and assist on the development of each module.

IFIGJENIA SOPIQOTI

Develop a maintenance plan and software evolution strategy.

Implement the cashier module functionalities.

DEJVI KOCILJA

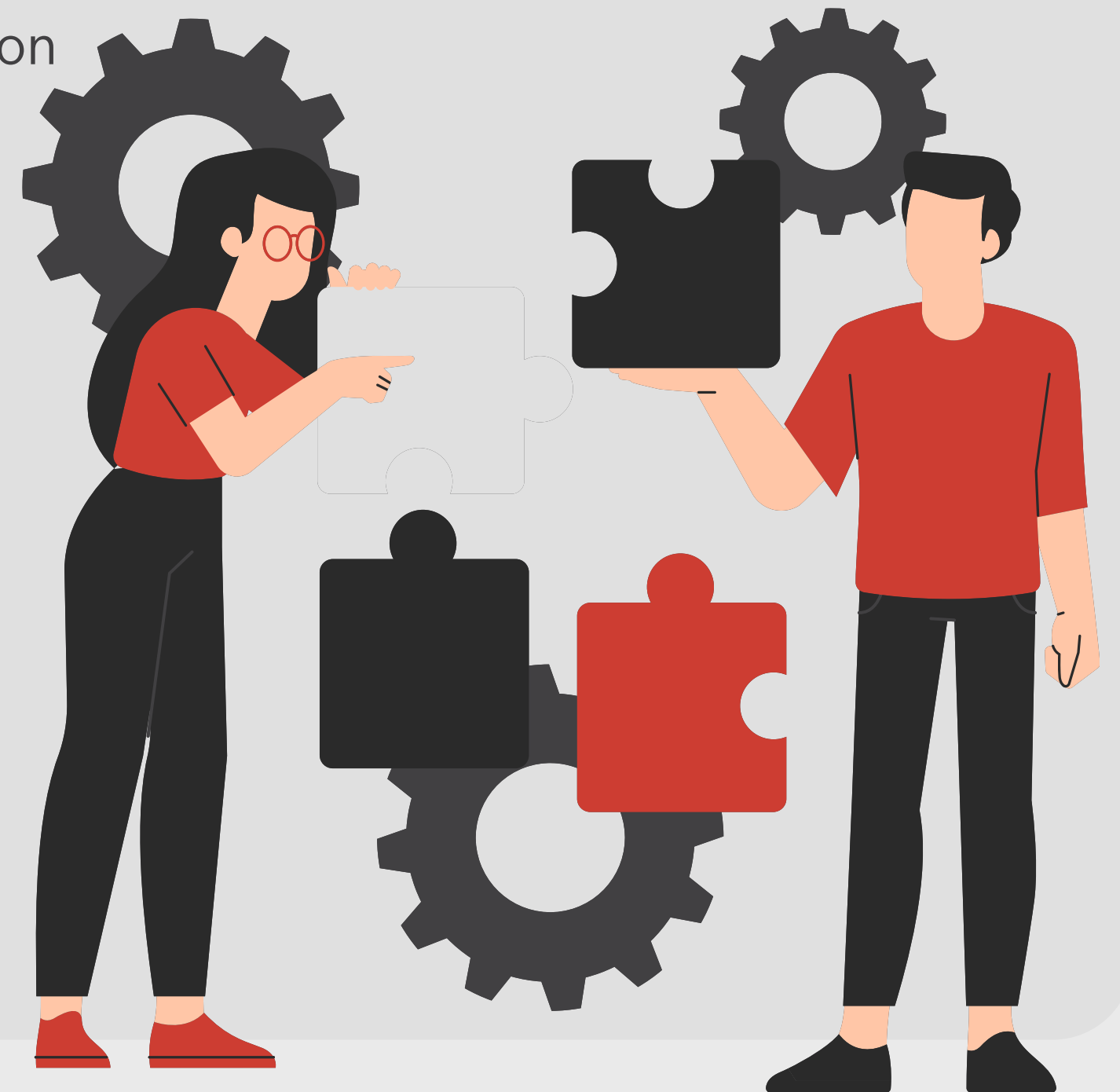
Facilitate the submission of user requirements and application specifications.

Develop the manager module.

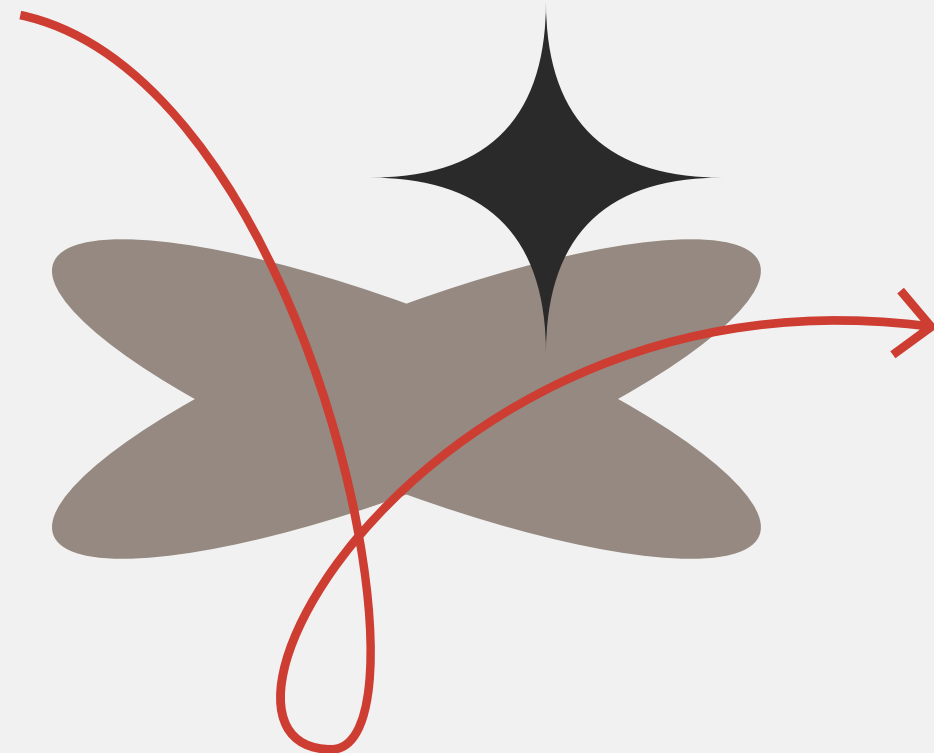
KEISI BRESHANAJ

Work in the description of the software design and modeling.

Implement the administrator module functionalities..



ACCEPTANCE CRITERIA



1

User-friendly interface:

With just three clicks or less, you can access your necessary functionalities.

2

The ability to adapt:

The capacity of the system to handle big transactions without experiencing a drop in performance.

3

Managing Data:

A database will be kept up to date by the system.

4

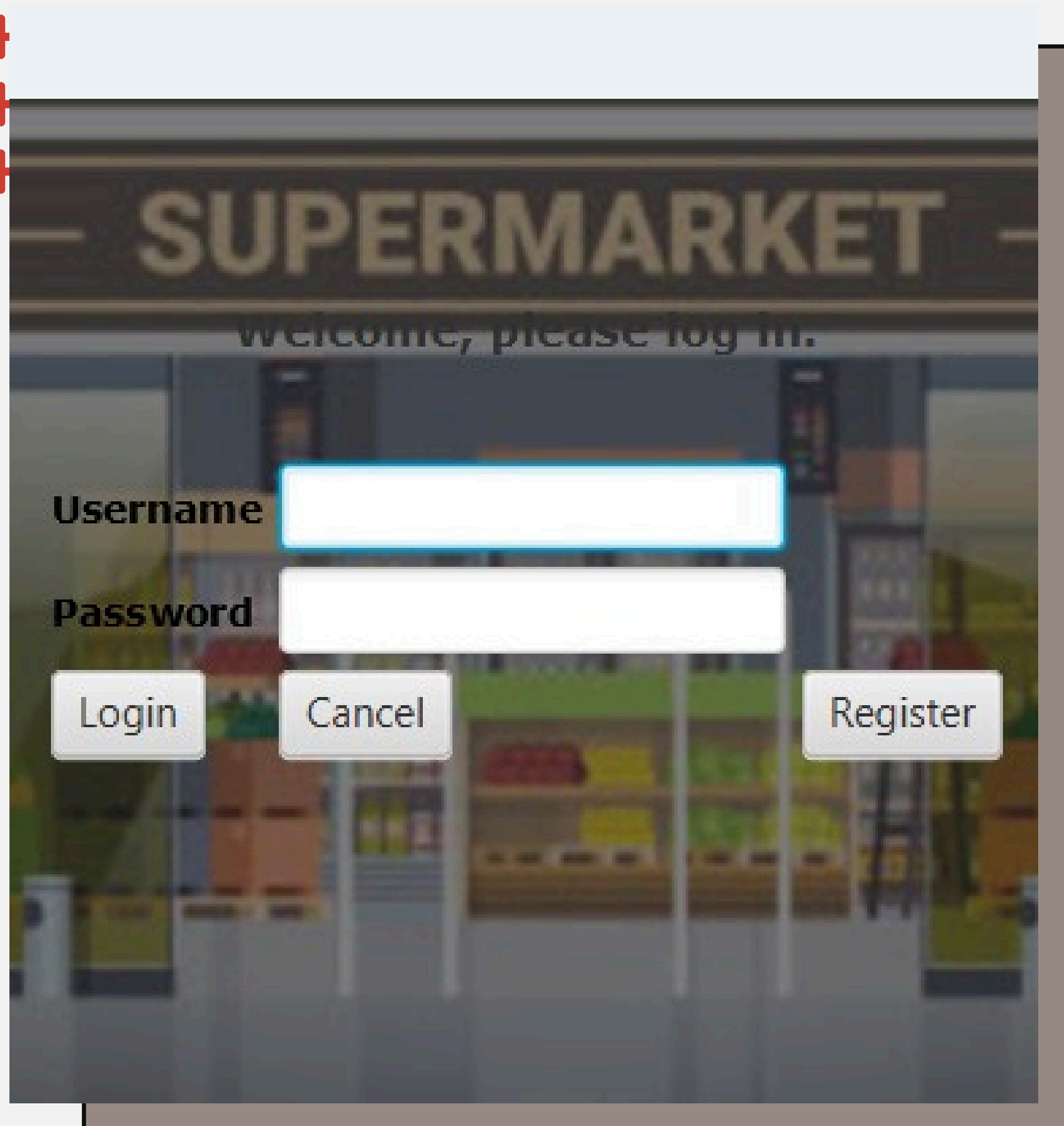
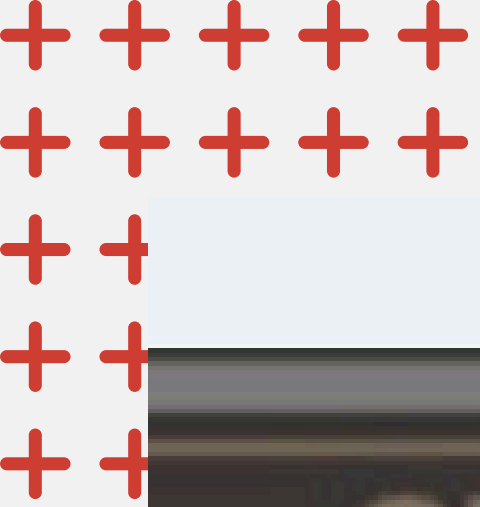
Security:

Only authorized users should be able to access important information.

5

Loyalty Points System:

The total amount of loyalty points is displayed correctly.



LOG IN SYSTEM

The login system requires a username and a password

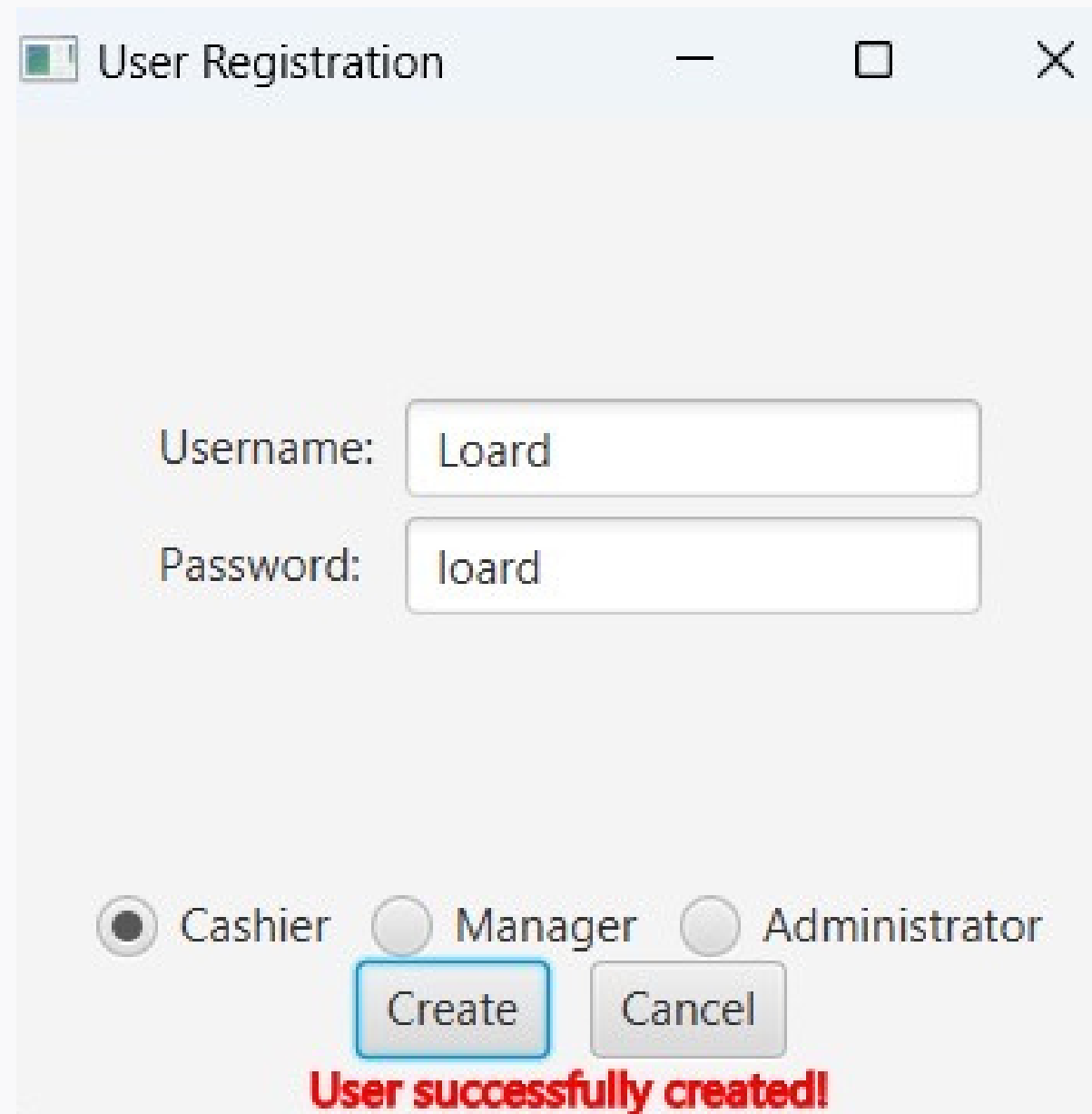
It has three functional buttons:

Login

Cancel

Register

WHAT HAPPENS WHEN YOU CLICK "REGISTER"



A screenshot of a 'User Registration' window. The window has a title bar with a close button (X). Inside, there are two text input fields: 'Username:' with the value 'Loard' and 'Password:' with the value 'loard'. Below these fields are three radio buttons for user roles: 'Cashier' (selected), 'Manager', and 'Administrator'. At the bottom are two buttons: 'Create' (highlighted with a blue border) and 'Cancel'. Below the buttons, the text 'User successfully created!' is displayed in red.

User Registration

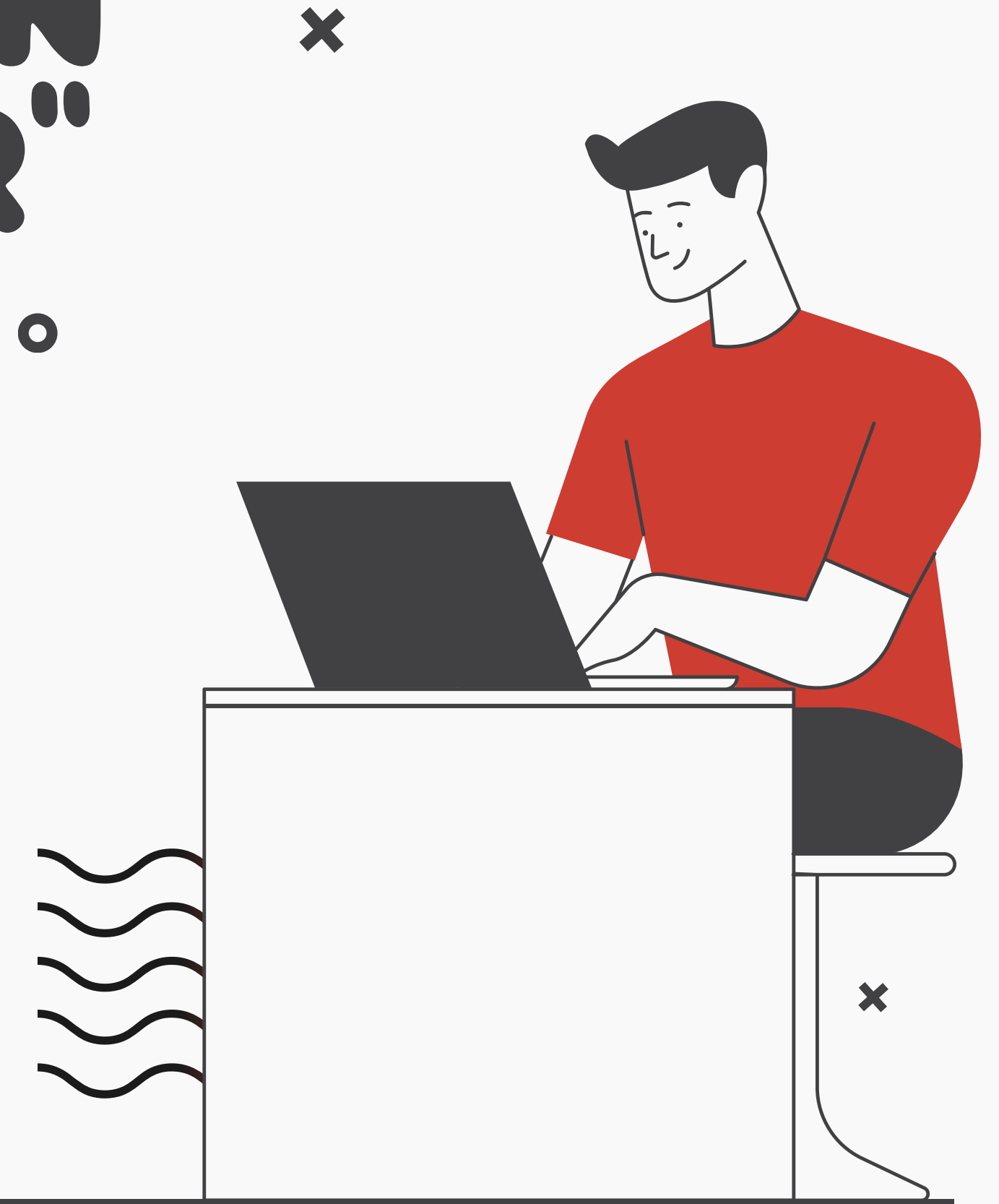
Username: Loard

Password: loard

☒ Cashier ☐ Manager ☐ Administrator

Create Cancel

User successfully created!





CASHIER SYSTEM

When you are logged in as a Cashier, you have the ability to:

- Check Inventory
- Show clients
- Add Clients
- Create Bill

You also have the option to log out.



WHEN YOU ADD CLIENTS

Loyal Client (Phone Number)

Bill Number

Name of product 1

Quantity of product 1

Date of Transaction

Add Product

Create

Welcome Miri
Logged in as an Administrator.

Show Products

Manage Employees

Show Users

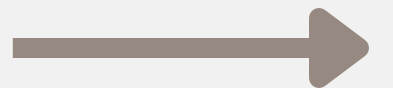
Show Income

LogOut

WHEN YOU ARE A ADMINISTRATOR

An administrator can:

- Check products
- Manage Employees
- Show Users
- Show Income



CHALLENGES



Coding Standards: Ensuring all team members follow the same coding standards and conventions to maintain readability and uniformity.

Misunderstandings: Miscommunication among team members about requirements, design decisions, or implementation details.

Dependency Management: Managing dependencies and ensuring that all parts of the system integrate smoothly.



**THANK
YOU !**

