CST8288 FINAL PROJECT

Research Assignment 2

By

Ahmad Al-Jabbouri

041068196

2024-03-18

Table of Contents

Version History

Version	Author	Date
1.0	Ahmad Al-Jabbouri	2024-03-18

Introduction

The high-level design document outlines the architecture, technologies, and functionalities of the Food Waste Reduction Platform (FWRP). This document provides a comprehensive overview of the system's design, including its solution architecture, UML diagrams, database model, deployment model, and system functionalities.

This document serves as a guide for stakeholders, developers, and other interested parties to understand the design and functionality of the FWRP system, enabling effective implementation and maintenance.

Targeted Audience

This document is primarily aimed at individuals involved in the development, deployment, and maintenance of the Food Waste Reduction Platform (FWRP). The intended audience includes:

Project Managers
Developers
Quality Assurance (QA) Engineers
Database Administrators (DBAs)
System Administrators
Business Analysts
Stakeholders
These stakeholders rely on this document to gain insights into the system's design, functionalities, and requirements, enabling them to contribute effectively to the project's success.

Scope

In Scope:

1. Overview of the Food Waste Reduction Platform (FWRP) system architecture.
2. Description of technologies utilized in the development of the FWRP.
3. Detailed UML diagrams representing the system's structure and behavior.
4. Database model outlining the schema and relationships between entities.
5. Deployment model illustrating how the FWRP will be deployed and hosted.
6. System functionalities, including user registration, inventory management, purchasing, etc.

7. High-level design considerations and decisions made during the planning phase.

Out of Scope:

1. Detailed implementation code or code-level explanations.
2. Low-level technical specifications of individual components.
3. Performance benchmarks or scalability assessments.
4. User interface design specifics, such as layout and styling.
5. Legal or regulatory compliance details.
6. Business-specific policies or procedures not directly related to system design.
7. Future enhancements or feature requests beyond the current scope of the project.

Application Architecture

The system comprises several main components working together to connect food retailers, consumers, and charitable organizations. Below is an overview of the key components:

1. Presentation Layer:
   - Responsible for the user interface, including web pages and user interaction.
   - Utilizes technologies such as JSP (JavaServer Pages) for dynamic content generation.
   - Handles user authentication, registration, and navigation.

2. Business Layer:
   - Contains the core business logic and functionalities of the system.
   - Implements use cases such as inventory management, purchasing, donation claiming, etc.
   - Orchestrates interactions between different system components.

3. Data Layer:
   - Manages the persistence of data used by the system.
   - Stores user information, inventory data, transaction records, etc.
   - Relational database management system (RDBMS) like MySQL is used for data storage.

Presentation Layer

The presentation layer is the front-end component responsible for interacting with users. It includes web pages, forms, and other UI elements presented to the user via a web browser. Key functionalities of the presentation layer include:

- Login/Registration Pages: Users can create accounts or log in to access the platform.

- User Dashboards: Different dashboards for retailers, consumers, and charitable organizations to manage their activities.

- Forms and Input Fields: Allows users to input data, such as inventory details, subscription preferences, etc.

- Error Handling and Feedback: Provides feedback to users on their actions and handles errors gracefully.

Business Layer

The business layer contains the core logic and functionalities of the FWRP system. It processes user requests, enforces business rules, and coordinates interactions between different components. Key functionalities of the business layer include:

- User Management: Handles user authentication, registration, and profile management.
- Inventory Management: Allows retailers to add, update, and manage their inventory of food items.
- Transaction Processing: Facilitates purchasing by consumers, claiming of donated items by charitable organizations, and updates inventory accordingly.
- Subscription Management: Enables users to subscribe to surplus food alerts based on their preferences.

This layered architecture ensures modularity, scalability, and maintainability of the FWRP system, allowing for easy updates and expansions in the future.

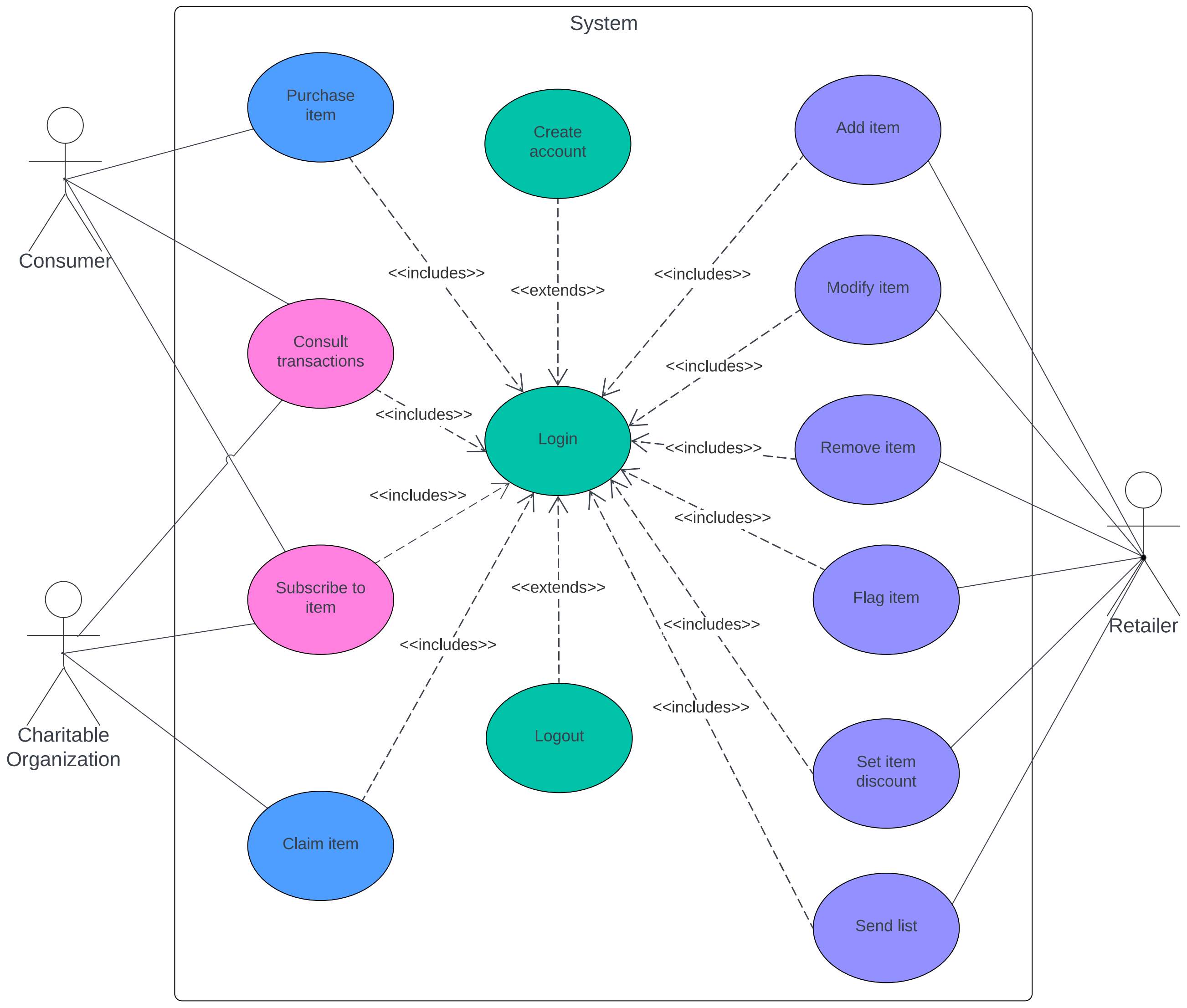
Data Layer

The data layer of the FWRP is responsible for managing the persistence of data used by the system. It stores various types of information, including user profiles, inventory details, transaction records, and subscription preferences. Key aspects of the data layer include:

- Database Management System: Using MySQL for data storage.
- Data Access: Provides mechanisms for accessing and manipulating data through SQL queries stored as prepared .sql files.

Overall, the data layer serves as the backbone of the FWRP, storing and managing the information necessary for the system's operation and facilitating efficient data access and manipulation.

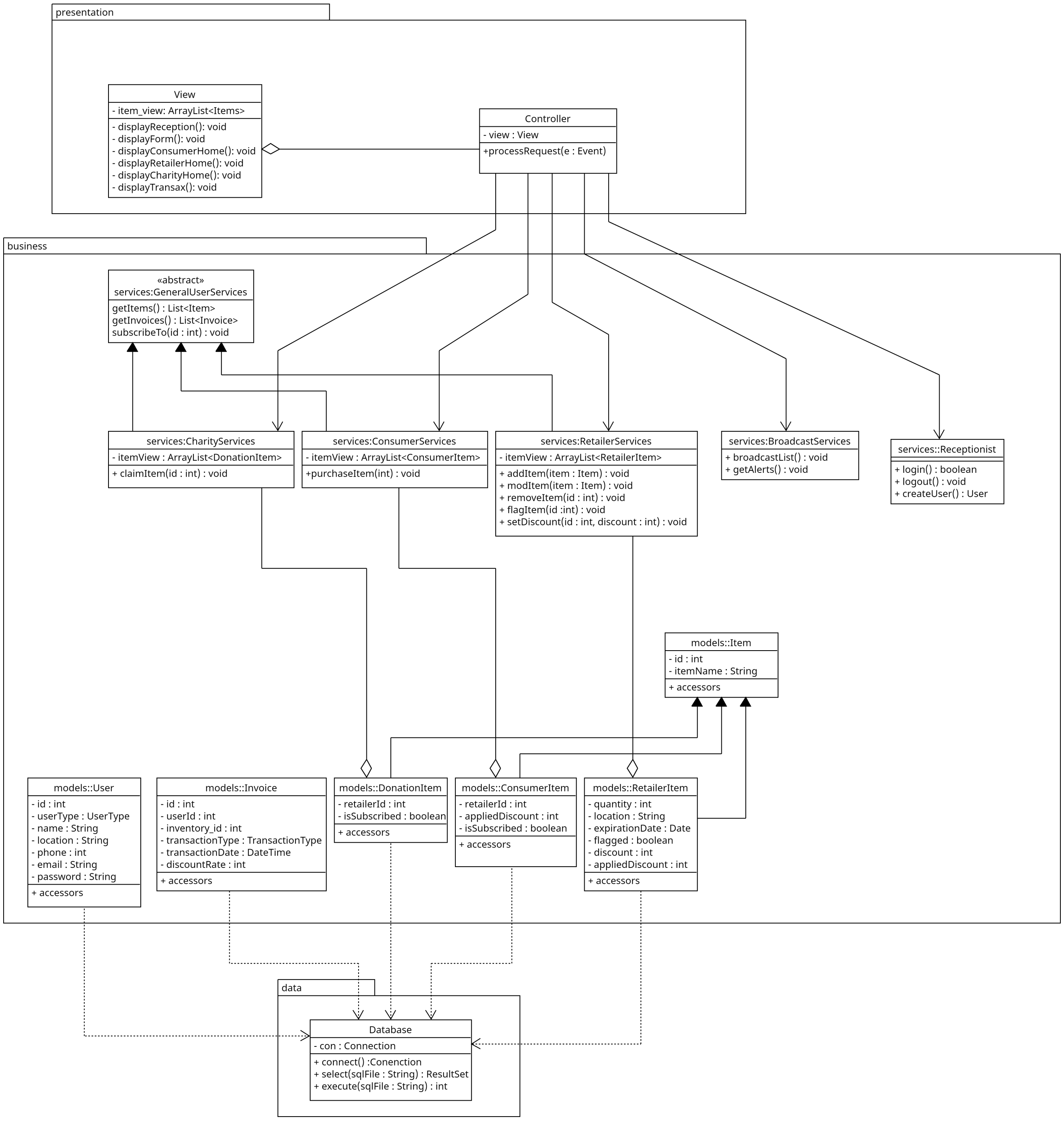
Business Architecture



This use case diagram provide a visual representation of how different actors (users) interact with the system to accomplish specific tasks. The use cases are categorized based on the roles of the actors involved, including retailers, consumers, and charitable organizations.

Detailed Design

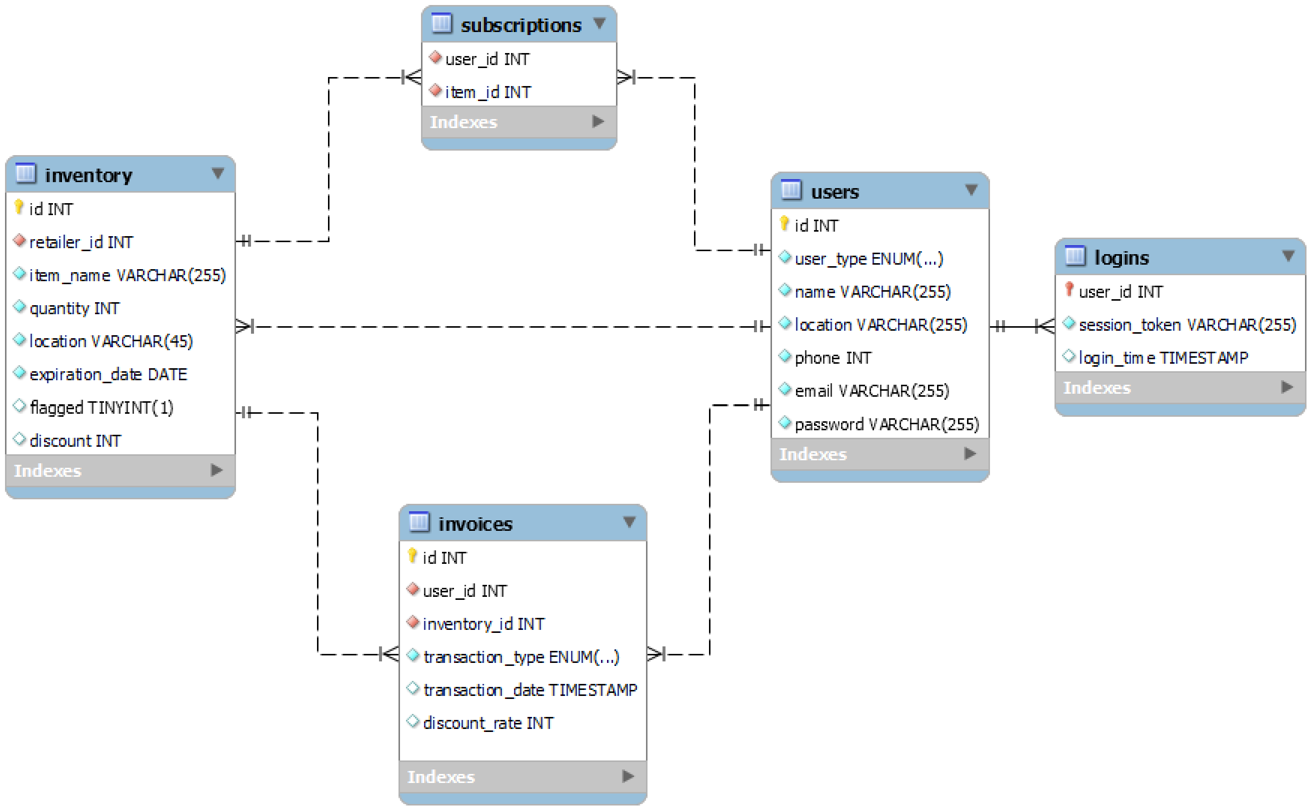
These diagrams provide a visual representation of the system's components and their interactions, facilitating a deeper understanding of the system's design.



Data Architecture

The Entity-Relationship Diagram (ERD) provides a visual representation of the database structure and the relationships between different entities. It illustrates how data entities are related to each other and helps in understanding the database schema.

The ERD for the Food Waste Reduction Platform includes entities such as Users, Inventory, Invoices, and Subscriptions, along with their attributes and the relationships between them. It serves as a blueprint for designing the database schema and guides the implementation of the data model.



For the sake of simplification, only one entity is designed to represent all types of users: Consumer, Retailer, and Charitable Organization; they are distinguished by the "user_type" ENUM attribute. The implication is that, an item in the inventory is never associated to a user of a type other than "retailer".

Security Architecture

SQL Injection Prevention

SQL injection is a common security vulnerability that occurs when an attacker inserts malicious SQL code into input fields or parameters in an application's SQL query. This can lead to unauthorized access to the database, data manipulation, and potentially the execution of arbitrary commands on the database server.

To prevent SQL injection in the Food Waste Reduction Platform, the following measures are implemented:

1. Prepared Statements: All SQL queries executed in the application use prepared statements with parameterized queries. This ensures that user input is treated as data

rather than executable code, preventing SQL injection attacks. Additionally, the prepared statements are stored in separate SQL files, contributing to code organization and maintenance efficiency.

2. Input Validation: All user input is validated and sanitized before being used in SQL queries. This includes validating input length, format, and type, as well as using whitelisting or blacklisting techniques to filter out potentially malicious characters.

3. Encoding: Special characters in user input are encoded using appropriate encoding techniques such as UTF-8 encoding to prevent them from being interpreted as part of an SQL query.

4. Error Handling: Proper error handling mechanisms are implemented to provide meaningful error messages to users without revealing sensitive information about the underlying database structure or query execution.

By implementing these measures, the Food Waste Reduction Platform mitigates the risk of SQL injection attacks and ensures the security of the database and user data.

Authentication and Authorization

Authentication and authorization are essential aspects of security in the Food Waste Reduction Platform, ensuring that only authorized users have access to specific resources and functionalities. Users are required to authenticate themselves using credentials such as username and password before accessing the platform. By implementing robust authentication and authorization mechanisms, the Food Waste Reduction Platform ensures the security and integrity of user accounts and sensitive data.

Deployment Architecture

On-Premises Deployment

For organizations preferring full control over their infrastructure, the on-premises deployment option allows hosting the Food Waste Reduction Platform within their premises. This setup involves installing and configuring the required software on hardware owned and maintained by the organization. While offering high control and customization, it requires ongoing maintenance and monitoring to ensure optimal performance and security.

Testing Model

Unit Testing

Unit testing is performed using JUnit, a popular testing framework for Java applications. It involves testing individual units or components of the application, such as classes and methods, in isolation to ensure they function correctly. Unit tests are automated and cover various scenarios to validate the behavior of the code under different conditions.

References

https://www.lucidchart.com/pages/uml-use-case-diagram

https://online.visual-paradigm.com/diagrams/tutorials/use-case-diagram-tutorial/

https://stackoverflow.com/questions/1696927/whats-is-the-difference-between-include-and-extend-in-use-case-diagram

https://www.codejava.net/java-ee/jsp/jsp-api-overview-uml-class-diagram

https://www.digitalocean.com/community/tutorials/java-web-application-tutorial-for-beginners#web-server-client

https://coderanch.com/t/573389/java/event-listeners-jsp

https://www.tutorialspoint.com/struts_2/basic_mvc_architecture.htm

https://www.reddit.com/r/explainlikeimfive/comments/5qz8hc/eli5_what_does_it_mean_to_build_package_and/

https://www.ibm.com/docs/en/was/8.5.5?topic=start-java-web-architecture-deploying-application-clients

Acronyms/Abbreviation

- **FWRP**: Food Waste Reduction Platform
- **JSP**: JavaServer Pages
- **RDBMS**: Relational Database Management System
- **SQL**: Structured Query Language
- **UML**: Unified Modeling Language

List of Figures