UNIVERSITI TEKNOLOGI MARA

UNIVERSITI TEKNOLOGI MARA (UiTM)

FACULTY OF COMPUTER AND MATHEMATICAL SCIENCES

CSC584 – ENTERPRISE PROGRAMMING REPORT

(Lost & Found Package Tracking & Resolution System)

CLASS: NBCS2404B

SUBMISSION DATE: 27th JULY 2025

PREPARED BY:

NO	STUDENT NAME	STUDENT ID
1	NUR SHUHADAH NAJWA BINTI ZAMZURI	2023250716
2	NUR HANIS AFIFAH BINTI HAMILUDDIN	2023287876
3	BALQIS NURAQILAH BINTI AZUWAN	2023242808
4	NURUL AZZARIAH BINTI MOHAMAD AZIZI	2023869218

PREPARED FOR:

MUHAMAD RIDHWAN BIN MOHAMAD RAZALI

Table of Contents

Introduction

A web-based program called the Lost & Found Package Tracking & Resolution System in E-Commerce Logistics was created to help users deal with the growing difficulties they encounter when packages disappear during delivery. Lost, delayed or incorrectly delivered packages have become more frequent as a result of the quick growth of online shopping and courier services. Consumers frequently struggle to get in touch with courier services or don't have a suitable channel to file complaints. By providing a centralised and user-friendly web portal where people can report lost or found packages and follow the resolution process, this system aims to close that gap.

User registration, report submission, report status updates, parcel recovery tracking and feedback gathering are all supported by the system. It guarantees that every case is resolved effectively by providing open lines of communication between clients and support staff. A smooth transition between the user interface, logic and data management is made possible by the project's MVC (Model-View-Controller) architecture, which makes use of JSP and Servlet technology with Apache Derby as the database. The system can be integrated as an issue-resolution module into the backend of logistics services or e-commerce platforms.

Objectives

This system's primary goal is to give users a dependable and efficient way to report and track problems related to lost or found packages. By using digital tools to streamline the reporting and resolution process, it also seeks to decrease misunderstandings between clients and logistics firms. The system also aims to give users the ability to monitor the status of the cases they have reported.

By providing them with tools to view, update, and validate parcel related issues, this project also seeks to help logistics staff manage reported cases more effectively. Additionally, by allowing users to submit feedback, the system promotes customer involvement and aids service providers in continuously enhancing their delivery and parcel handling procedures.

Scope

The essential features required to report and resolve lost and found packages in the context of e-commerce logistics are covered by this system. It has features like user registration and login, lost and found item reports, the owner's ability to edit or remove reports, and a feedback module where users can leave comments or concerns. Additionally, users can see the status of their cases, encouraging accountability and openness.

Real-time package tracking via third-party APIs from courier services is not, however, covered by the system. Additionally, it lacks integration with real e-commerce or courier platforms as well as a dedicated mobile application.

System Architecture (MVC)

Model-View-Controller (MVC) is a popular software design framework that divides the application into three interrelated components which is Model, View and Controller. It is used in the development of the Lost & Found Package Tracking & Resolution System in E-Commerce Logistics. The application is simpler to develop and administer thanks to this structure, which encourages scalability, maintainability and separation of concerns.

Model

The application's data layer is represented by the Model. It manages the database logic, including communication with the NetBeans IDE's built-in default database, Apache Derby. The DBConnection.java class is in charge of overseeing the connection between the database and the application in this system. It enables data stored in tables like USERS, LOST_ITEMS, FOUND_ITEMS, and FEEDBACK to be accessed, inserted, updated, or retrieved by Servlets.

View

All of the user interface (UI) elements constructed with JavaServer Pages (JSP) make up the View. These pages have a responsive, tidy layout thanks to their HTML design and Bootstrap styling. By completing tasks like registering, logging in, reporting a lost or found item, viewing previously submitted reports, providing feedback, and more, users can engage with the system through the views. JSP page examples include dashboard.jsp, editFoundItem.jsp, editLostItem.jsp, feedback.jsp, forgotPassword.jsp, index.jsp and others. Every page manages user input via form submissions and dynamically displays data pulled from the database.

Controller

Controller is Servlets, which serve as the layer of logic and decision-making between the model and view, make up the Controller. Each servlet uses the DBConnection class to communicate with the database, processes user input from JSP forms and decides which page should be shown next. The following are some instances of the controller classes that were utilised in the project. DeleteFoundItemServlet, DeleteItemServlet, DeleteLostItemServlet, EditFoundItemServlet, EditLostItemServlet, FeedbackServlet and many others. The controller maintains a clear division between the frontend and backend components by controlling requests and routing responses.

Advantages of MVC in This System

Separation of concerns:

Data (DB), logic (Servlets) and user interface (JSP) can all be modularised in development.

Maintainability:

Future UI or business logic modifications can be made on their own.

Reusability:

Design elements and code can be applied to other modules that are similar.

Scalability:

Because of its clear structure, it is simpler to grow with more modules or users.

Database Design (ERD)

In order to store user credentials, lost and found item reports, and user feedback, the Lost & Found Package Tracking & Resolution System makes use of a structured relational database. The NetBeans IDE supports Apache Derby, a lightweight embedded database, which is used to implement the database.

The logical organisation of the system's database and the connections between its main entities are shown in the Entity Relationship Diagram (ERD) below.

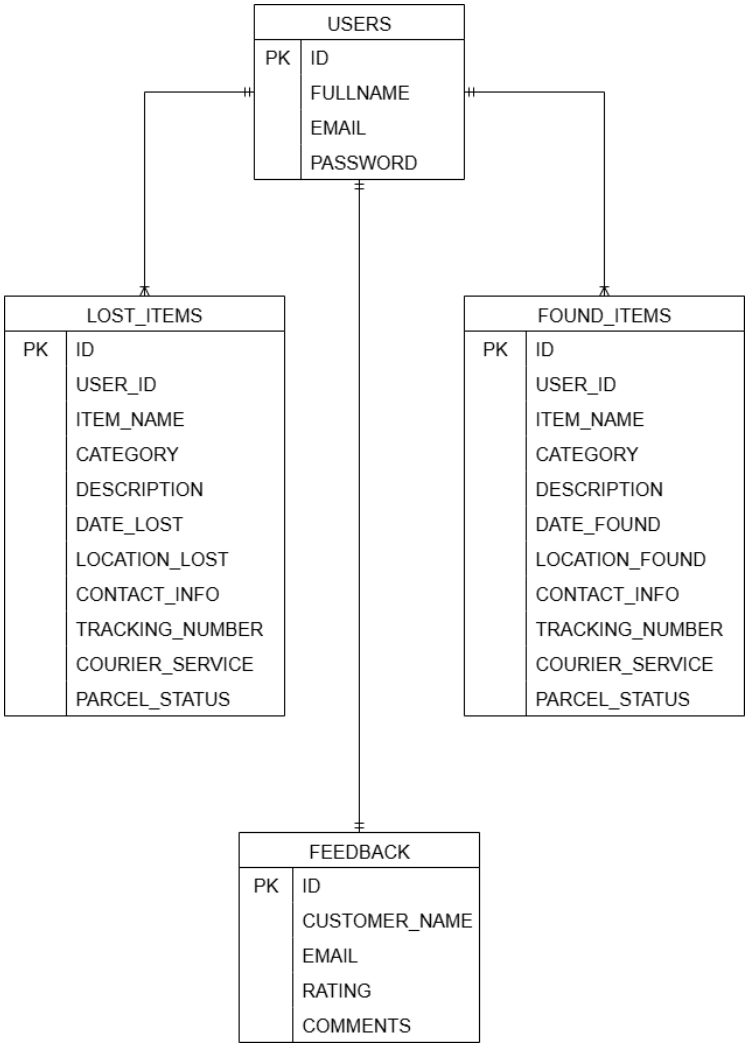

Figure 1 Entity Relationship Diagram

The four primary tables in the system's database are USERS, LOST_ITEMS, FOUND_ITEMS, and FEEDBACK. Every table has a distinct function in the administration of user information, lost and found reports, and user reviews.

Important details about people who have registered in the system are kept in the USERS table. In addition to the user's full name, email address, and password, it contains a distinct user ID as the primary key. To prevent duplicate accounts, the email field must be unique.

All reports related to lost packages are contained in the LOST_ITEMS table. The user ID of the person who filed the report, the item name, category, description, location and unique ID are all included in each lost item record. The table also contains the tracking number for the package, the date it was reported, and the report's current status.

Likewise, all found item reports are kept in the FOUND_ITEMS table. With fields like item name, category, description, tracking number, status and submission date, it is structurally similar to the LOST_ITEMS table. Each found item is linked to the person who filed the report via a foreign key relationship, just like lost items.

User-submitted messages about their experiences or questions about the system are stored in the FEEDBACK table. Every feedback entry includes a unique ID, the user's name and email address, the content of the message and the submission date. By gathering user feedback and resolving issues, this table helps to enhance system performance and user satisfaction.

Storyboard / UI Flow

The Lost & Found Package Tracking & Resolution System's user journey is visually represented by the Storyboard or UI Flow. It describes the main screens that users see and how they navigate between them, showing how the system facilitates administrative oversight, user authentication and reporting of lost and found items.

The application starts with the Landing Page, which gives users access to navigation options for registering or logging in as well as basic system information. After successfully logging in, users are taken to the User Dashboard, which includes quick links to important features and summaries of the lost and found reports they have submitted.

Users can select either Report Lost Item or Report Found Item from the dashboard, which will take them to the appropriate submission form. Users must fill out these forms with specific details like the item name, category, description, tracking number and location. Following submission, the item is entered into the database and assigned a default status.

Additionally, users can view, edit or remove their own submissions by visiting the My Lost Reports and My Found Reports pages. Only when the currently logged-in user is the report's owner does each report entry display important item information and offer edit or delete options.

Users can also view reports submitted by others through the system's All Lost Reports and All Found Reports section. There is a Feedback Form for user communication where users can post queries, recommendations or problems. Users with little technical expertise can easily report issues, navigate the system and get timely updates. The storyboard as a whole follows to the MVC architecture, ensuring that presentation, logic and data management are appropriately separated.

System Features & Functionality

The process of reporting, tracking and resolving lost and found packages is made easier by the user-centric features of the Lost & Found Package Tracking & Resolution System in E-Commerce Logistics. Every feature is essential to helping system everyday users throughout the reported item's lifecycle.

Authentication and User Registration

- By entering their full name, email address and password, new users can register.
- To access the system, registered users must safely log in.
- Only authorised users are able to report or manage items.

Report a Lost Item

- Users can provide comprehensive details regarding a misplaced package, such as:
    i. Item name
    ii. Category
    iii. Description
    iv. Tracking number
    v. Last known location
- To guarantee that all necessary fields are filled out correctly, the form has validation.

Report Found Item

- Users can report found packages by completing a form with the item's details, just like in the lost item module.
- To facilitate item identification, there is an image upload field.
- The database is updated with submitted found reports for community visibility.

View My Lost/Found Reports

- Users who are logged in can see all of the lost or found reports they have submitted.
- The item's details and its current status (such as pending or resolved) are shown in each record.
- Only if the user is the original submitter are the Edit and Delete buttons displayed.

View All Lost/Found Report

- List of all reports submitted by all users that is accessible to the public.
- Allows users to look for matches in other reports, which aids in returning misplaced objects to their rightful owners.
- Read-only access, meaning that unless the user is the owner, they cannot edit or delete anything.

Edit/Delete Capabilities

- Only the reports that users have created can be edited or removed.
- Edit functionality enables status updates or information correction.
- Depending on how it is implemented, deletion can be either soft-deletion or total removal.

Screenshots

A selection of screenshots from the Lost & Found Package Tracking & Resolution System are displayed in this section. A visual summary of how users interact with the system is provided by each screenshot, which represents a significant feature or user interface. These illustrations support the system's functional description and draw attention to the modules ease of use, simplicity and navigation.

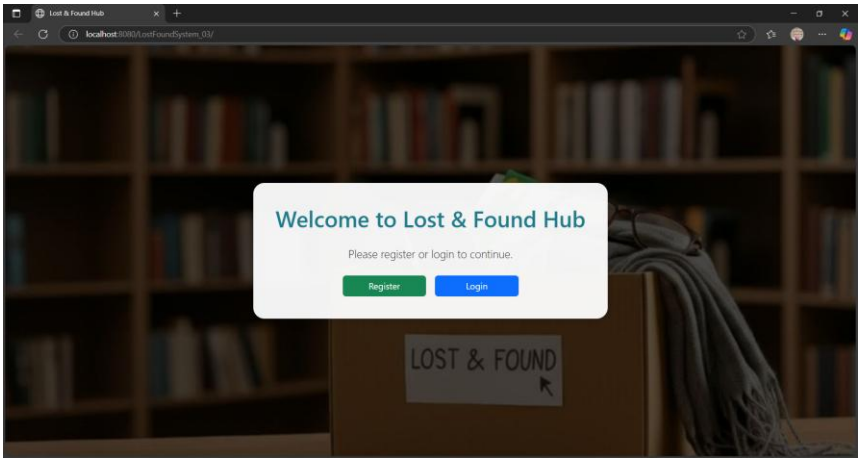

Figure 2 Landing Page

The landing page, which has a simple and intuitive design, acts as the system's welcome interface. Users can register as new users or log in to their existing accounts using its user-friendly navigation buttons and welcome message. This serves as the initial point of contact for all users and clearly explains the goal of the system, assisting users in understanding what the application has to offer from the outset.

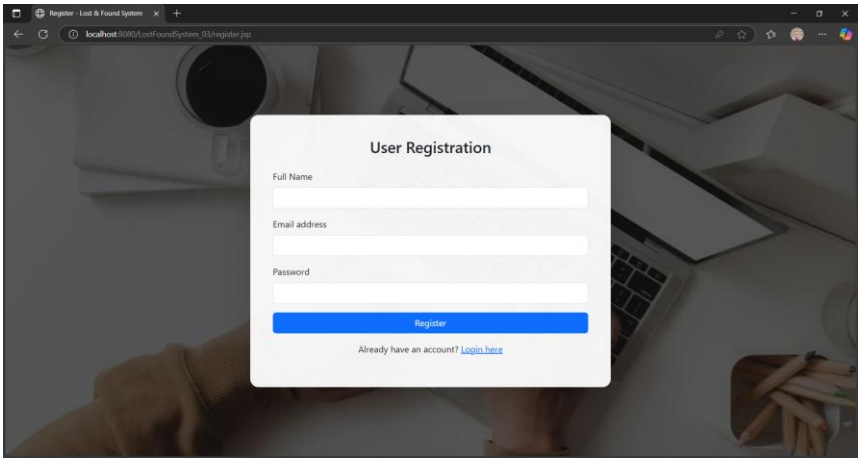

Figure 3 User Registration Form

New users can register for an account on the system using this interface. Users are asked to enter their full name, a working email address and a strong password on the registration form. Before submitting, input validation makes sure the fields are filled out accurately. This is an important first step for new users because after registering, users can log in and access the system's reporting and feedback features.

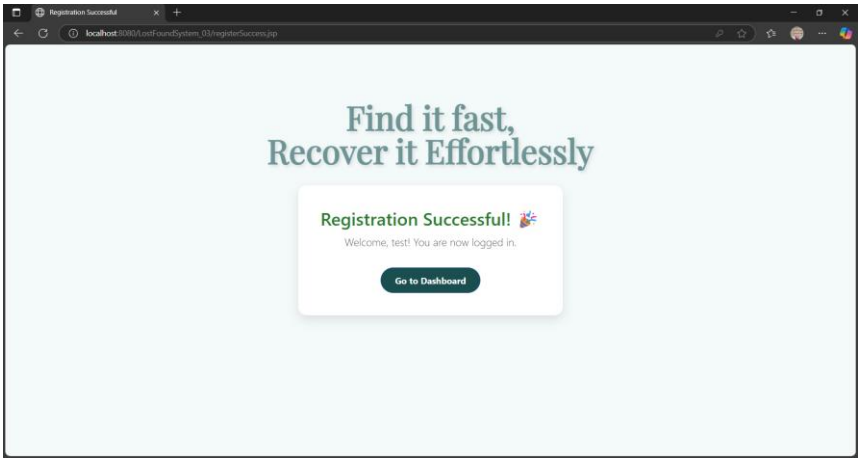

Figure 4 Registration Success Page

Users receive a confirmation message confirming the creation of their account after successfully registering. This encourages the user to continue to the login page by reassuring them that the process has been finished. It is a straightforward but crucial step to improve system confidence and user experience.

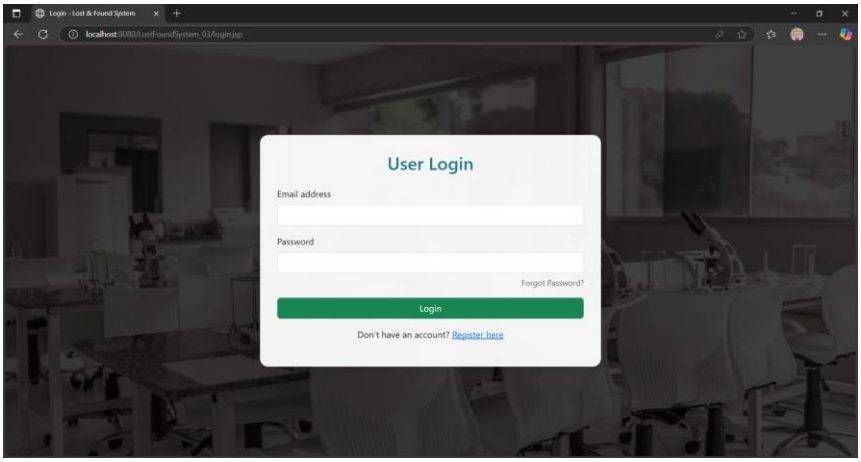

Figure 5 Form for User Login

By entering their registered email address and password, returning users can access their accounts through the login interface. The form incorporates authentication logic to verify user credentials and, if successful, reroute them to their dashboard. An appropriate error message appears if the credentials are entered incorrectly. By limiting system access to registered users only, this screen protects the integrity and privacy of data.

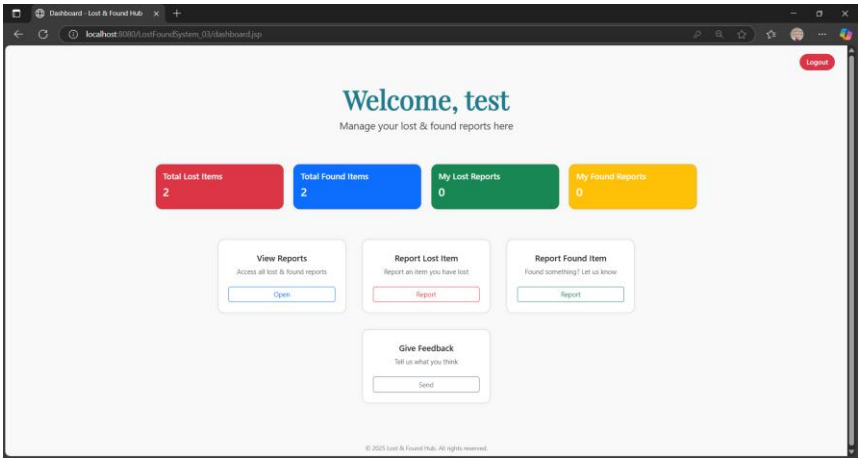

Figure 6 User Dashboard (After Login)

Users are taken to their individual dashboard after logging in. Quick access cards or links to key modules, including viewing personal reports, reporting lost or found items, and getting feedback, are displayed on this interface. For users who must promptly report or handle incidents, it improves efficiency and navigation by centralising all necessary functions.

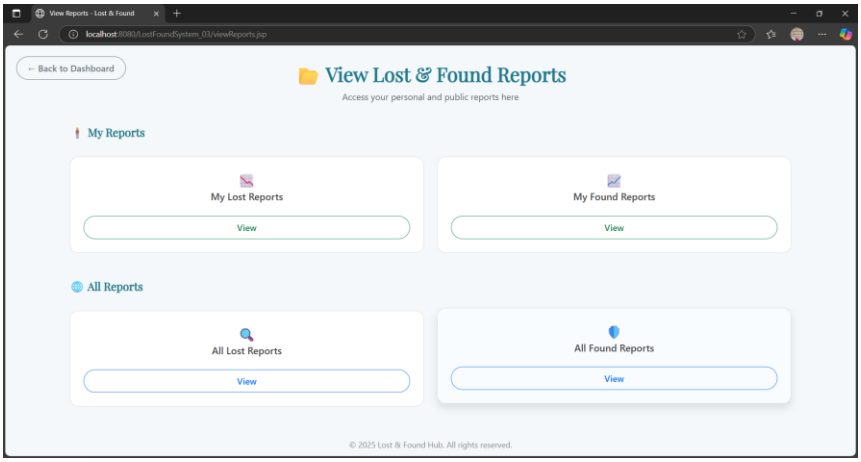

Figure 7 Navigation to All Reports

Users can easily browse through all of the lost and found reports that are stored in the system using this screen. In order to cross-check and help find matching lost or found items, users have the option to view only the reports they have submitted or to access the global list. The feature improves the likelihood of an item being successfully recovered and promotes user collaboration.

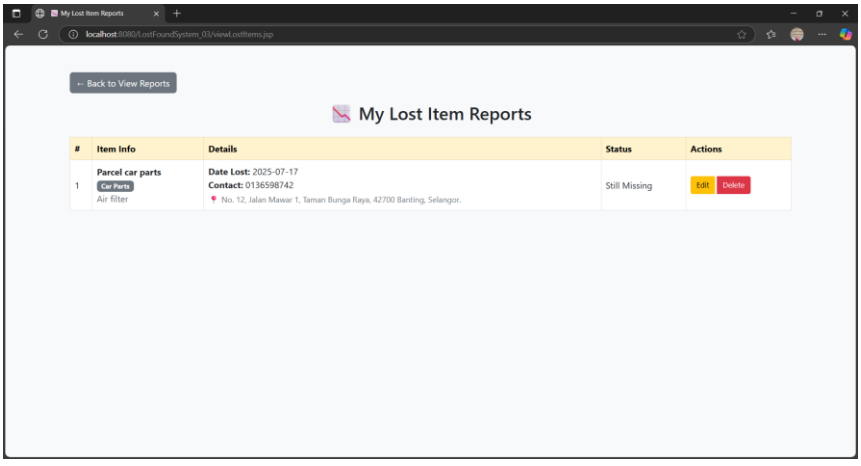

Figure 8 My Lost Item Reports Page.

A table containing every lost item reported by the user currently logged in is displayed on this page. Details like the item name, category, tracking number and submission date are displayed for each entry. Action buttons are provided for user control and accuracy, enabling the user to amend or remove each report as necessary. Users can efficiently manage their individual reports with the aid of this interface.

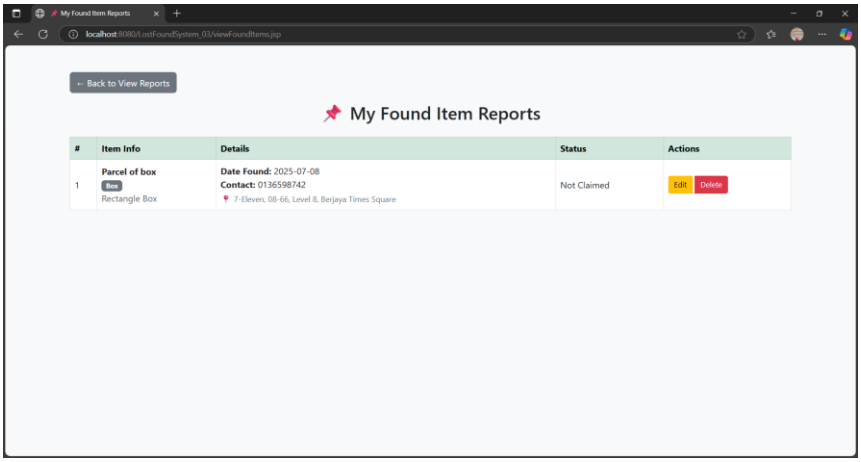

Figure 9 My Found Item Reports Page

This section shows all found items submitted by the logged-in user, much like the lost reports page. It has action buttons for editing or removing each entry along with important information. In addition to improving the matching between reported lost and found items, this guarantees that users can effectively handle found item reports and update any erroneous or out-of-date information.

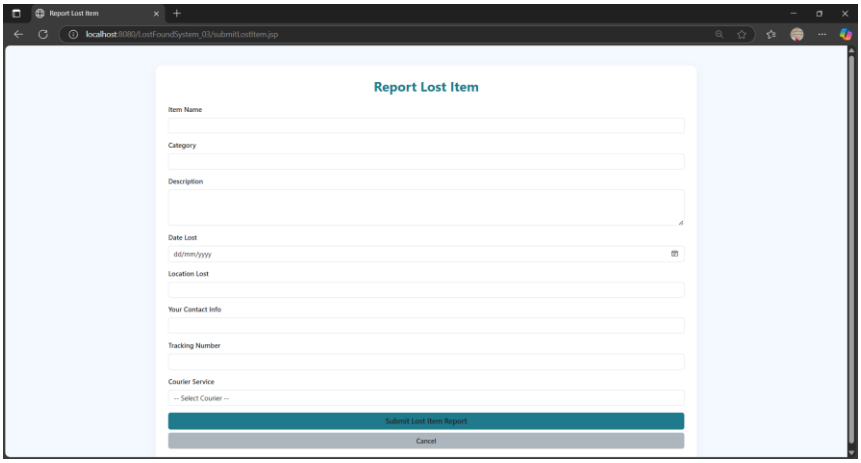

Figure 10 Report Lost Item Form

Users enter details about a package or item they have misplaced on the report lost item form. To provide as much identifying information as possible, it contains fields for the item name, category, tracking number, and description. The likelihood of recovery is increased by this thorough data entry procedure, particularly when users supply special characteristics or tracking codes associated with the misplaced package.

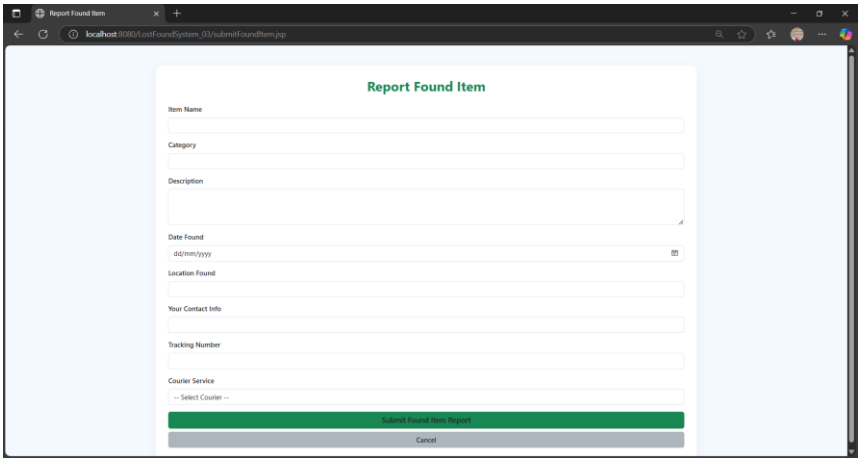

Figure 11 Report Found Item Form

When a user discovers something and wishes to report it in the system, they use this form. It gathers the item's name, category, tracking number, and any other descriptive information that may be useful in identifying the true owner, much like the lost item form does. Matching the report with another person's report of a possible lost item is made easier with more detailed information.

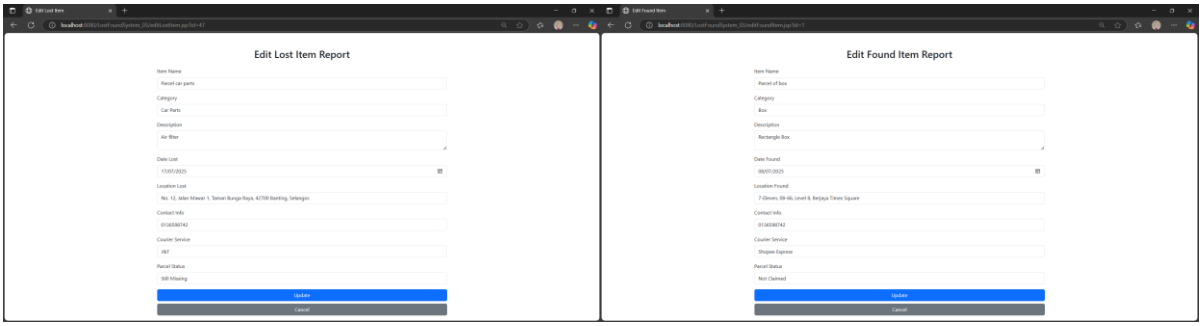

Figure 12 Edit Lost/Found Item Page

Users can update or correct lost or found item reports that have already been submitted using this page. When the user reaches the edit page, the system uses the item ID and user ID to retrieve the pertinent report data, which is then displayed in an editable format. Important fields like item name, category, description, and tracking number are all modifiable by users. This feature is crucial for preserving the system's data accuracy, particularly when users need to correct errors or remember new information. Importantly, data integrity is maintained and unwanted changes to reports are avoided because only the original reporting user is permitted to make edits. This feature helps make the reporting process more reliable and user-focused.

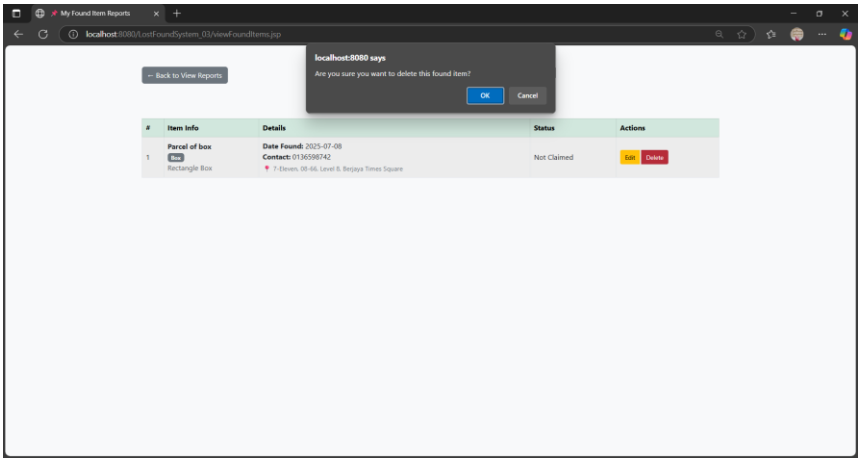

Figure 13 Delete Confirmation Popup

The system asks the user to confirm their action with a confirmation popup before deleting a report. This feature is crucial for preventing unintentional deletions and protecting user-submitted data until the user specifically confirms the action. It supports the system's user-centred and error-preventive design methodology.

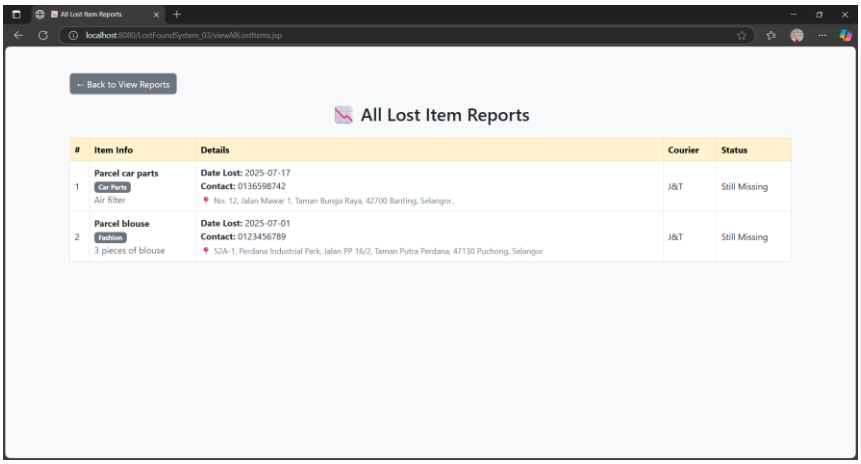

Figure 14 All Lost Item Reports Page

Regardless of who submitted the report, anyone can view all lost item reports in the system on this public page. It encourages openness and facilitates users' scanning for objects they might have recovered or found. This page is crucial in helping people who have lost their possessions get in touch with those who might have located them because it contains all reported items.

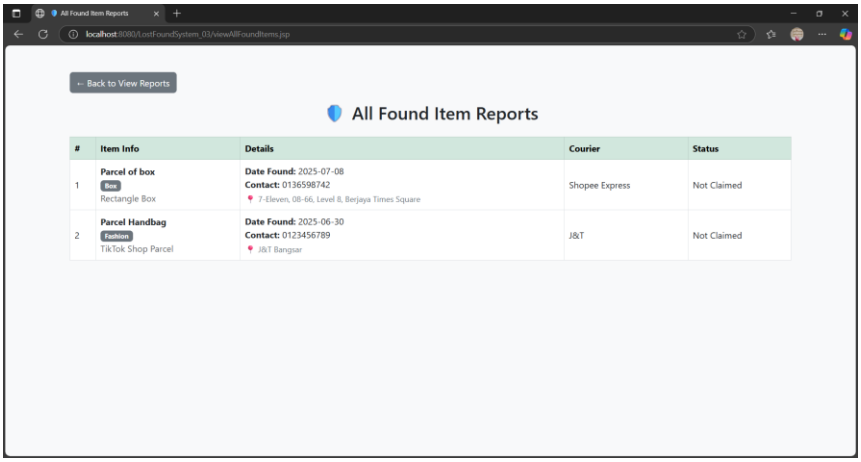

Figure 15 All Found Item Reports Page

All of the items that users have reported finding are listed on this page. It is publicly accessible and contains item details, establishing an open area where owners of misplaced items can peruse and potentially find their possessions. The main goal of item recovery is supported and user participation is encouraged by this system-wide visibility.

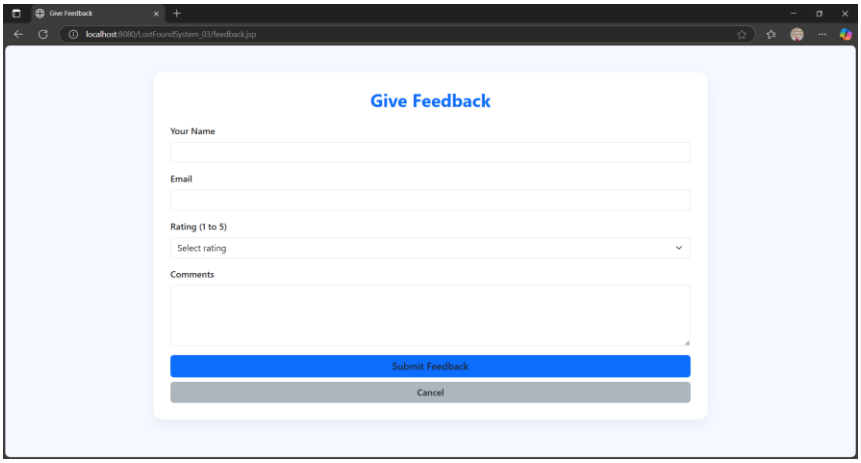

Figure 16 Feedback Form Page

This form allows users to provide feedback while collecting their name, email address, and message. This feature is crucial for getting user feedback, answering questions, and managing recommendations for system enhancements. Additionally, it serves as a direct line of communication between users and the administrator, bolstering the application's overall support framework.

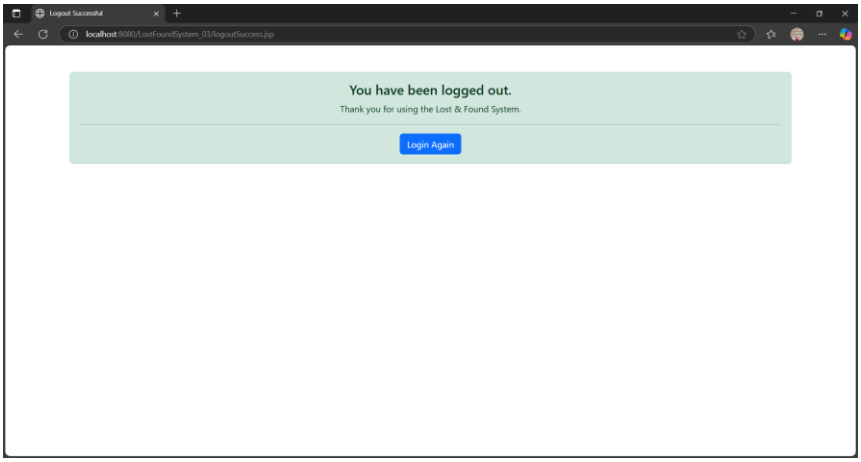

Figure 17 Logout Success Message

Users receive a confirmation message after successfully ending their session. This brings the exchange to a close and verifies that their account is now safe. It is a minor but essential component of the user flow that upholds confidence and supports sound session management techniques.

Challenges & Solutions

User Authentication and Access Control

Challenge:

A key security requirement was to make sure that only registered users could access specific features like editing, deleting and submitting reports. Additionally, we had to limit edit and delete rights to the people who submitted the report in the first place.

Solution:

Using Java Servlets and JSP, we put in place a session-based authentication system. We made sure the system only displayed edit and delete options to the report owner by keeping track of the user's userID during the session. This strategy preserved user information while preserving a customised experience.

Database Connectivity and Query Handling

Challenge:

A number of challenges appeared during the web application's connection to the Apache Derby database, particularly when it came to managing foreign key constraints and multiple table relationships and managing data insertion, retrieval, and updates across multiple modules.

Solution:

We made sure all servlets used PreparedStatement to securely communicate with the database and carefully structured our database with well-defined relationships. This reduced the possibility of SQL injection and guaranteed data consistency between modules such as item reporting, feedback and registration.

Responsive and Consistent UI Design

Challenge:

At first, it was challenging to create an intuitive user interface that functions well on various screen sizes while keeping design elements consistent throughout all pages.

Solution:

To create a simple, responsive design, we incorporated the Bootstrap CSS framework. To guarantee familiarity and usability, reusable elements like tables, cards and navigation bars were styled uniformly throughout all modules.

Team Coordination and Version Control

Challenge:

It became difficult to coordinate code and maintain version control when there were four team members working on different aspects of the system, particularly when combining modules.

Solution:

To monitor changes, settle disputes, and collaborate on various branches, we employed GitHub for version control. To review code integration, assign tasks, and discuss progress, the team met on a regular basis.

Conclusion

One successful attempt to use web-based technology to address a real-world issue is the Lost & Found Package Tracking & Resolution System in E-Commerce Logistics project. This system enhances communication between users and e-commerce logistics providers while encouraging accountability and transparency. It is designed to help users report, track, and resolve issues related to lost parcels.

The team used JSP, Servlets, and the MVC architecture to implement fundamental Java web development concepts throughout the development process. We created user-friendly interfaces for managing and submitting lost or found items, made sure user registration and login were secure and added necessary features like tracking numbers and feedback submission. An efficient backend solution for storing user data, reports and feedback was made possible by the database management system Apache Derby.

The team was successful in creating a fully functional, user-centred system in despite a number of obstacles, such as controlling user sessions, integrating the database, and guaranteeing appropriate access control. This project improved our teamwork, version control procedures, and user experience design skills in addition to our technical proficiency in database administration and web application development.

Future improvements to the system might include adding notification features (like email alerts), extending reporting capabilities through dashboards, and integrating with actual logistics APIs. All things considered, this project shows how technology can be extremely important in the e-commerce and logistics sectors for building trust, resolving problems, and raising customer satisfaction.

References

Dev_Jos4kn. (2023, August 13). *Online Lost & Found and Inventory Management system*. Lostings. https://www.lostings.com/

Lost & Found System (Community). (n.d.). Figma. https://www.figma.com/proto/hX8s9J5bg1UewfrmOpxGhM/Lost---Found-System--Community-?node-id=100-987&t=cH1G1z0snssAPq8m-0&scaling=scale-down&content-scaling=fixed&page-id=0%3A1