**ASSIGNMENT INFORMATION**

**MODULE NAME:** WEB APPLICATIONS AND AI

**MODULE CODE:** 7051CEM

**ASSIGNMENT TITLE:** COURSEWORK

**Table of Contents**

[Introduction 3](#_Toc173366276)

[Application Design 3](#_Toc173366277)

[Design Patterns and ER Diagram 3](#_Toc173366278)

[Development Process 5](#_Toc173366279)

[Key Challenges and Solutions 5](#_Toc173366280)

[Implementation 5](#_Toc173366281)

[Conclusion 10](#_Toc173366282)

[References 12](#_Toc173366283)

# Introduction

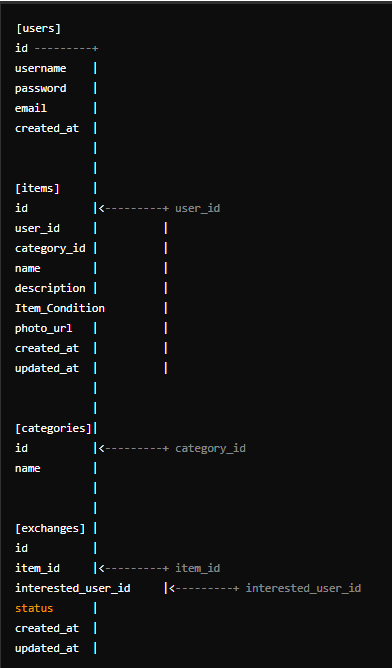
The proposed company is Mary’s Exchange Agency – a new venture for a small business that operates in the Exchange of products and services across the Internet. This project will be an internet-based application for selling and buying goods with the help of gear swapping where a user can create a list of the goods he wants to exchange; view other users’ lists of goods they wish to swap; and arrange the swap. The system combines the enhanced concepts of web elements and relational databases to ensure optimum usability. Comparable sections include platform format, development and deployment of the platform, issues met and addressed correspondingly, and final recommendations.

# Application Design

In the specifics of the application design, there are four goals laid out which are oriented towards creating a strong, scalable, and user-oriented design. It involves more than a few crucial processes to improve the user's perceived quality. First of all, Item Listings enable exchanging personal items by giving descriptions, conditions, and photos of the items that the users want to swap between each other and others. Browsing and Searching functionalities allow users to Browse and Search through the list displayed by other users with options to filter by sub-categories such as electronics, clothing, or furniture to find items that a user is interested in.

## Design Patterns and ER Diagram

MVC (Model-View-Controller): This pattern seems to subdivide the application into three parts that are related to each other, and this makes the level of modularity higher. Model is for the database entities and the view popularizes it to be on JSP whereas the controller for accepting the user inputs is Servlets.



**Figure 1: Design patterns and ERD relationship**

(Source: Acquired from IntelliJ Software)

DAO (Data Access Object): Isolate the code that gets access to the database, enhancing the software’s modular structures.

Singleton: Helps to ensure that there should not be more than one instance of the services such as UserService, and ProductService since all the communication made to any of these services will be restricted to this proxy.

The ER diagram includes key tables: Thus, it is possible to outline the following major tables within the frameworks of the discussed ER diagram:

Users: It saves the name in the string, the password also in the string, the E-mail, and the date of registration of the use.

Categories: Lists item categories.

Items: It maintains the history of an item regarding the user ID, category ID, name, description, condition, and Photo URL of the item.

Exchanges: Exchange transactional information with the ID of the special item and the user who shows interest in the item, the status, and the date/time stamp.

# Development Process

Collecting functional and non-functional requirements and studying them to realize the exact scope of the project. Growing the structures of how something will be built as well as the foundation on which data will be stored and organized and sketching the overall look and feel of what users will see on their screens. Calling it quits on the first, hard coding the backend services, installing the database configurations, and designing the front ends. Carrying out unit tests, integration tests, and user acceptance tests to confirm the efficiency and usability of the created program. Hosting the application in a server environment hence making it runnable by the users.

## Key Challenges and Solutions

***Database Integration:***

Challenge: To make sure that there is harmonious communication between the application and the MYSQL database.

Solution: Using the DAO pattern to implement the access to the database to increase modularity and decrease the amount of code.

***User Authentication:***

Challenge: Beginner: PROTECTING USER CREDENTIALS AND SESSIONS.

Solution: Employing a strong password management system with hashing of passwords as well as applying sound session management mechanism to safeguard users’ confidential information and preserve the session.

***Dynamic Content Rendering:***

Challenge: For effective showing of the dynamic contents such as the list of users or search results.

Solution: For the server side and the part of the application that does not require frequent updates, use JSP and for the part that requires frequent updates incorporate the use of AJAX.

***Data Validation:***

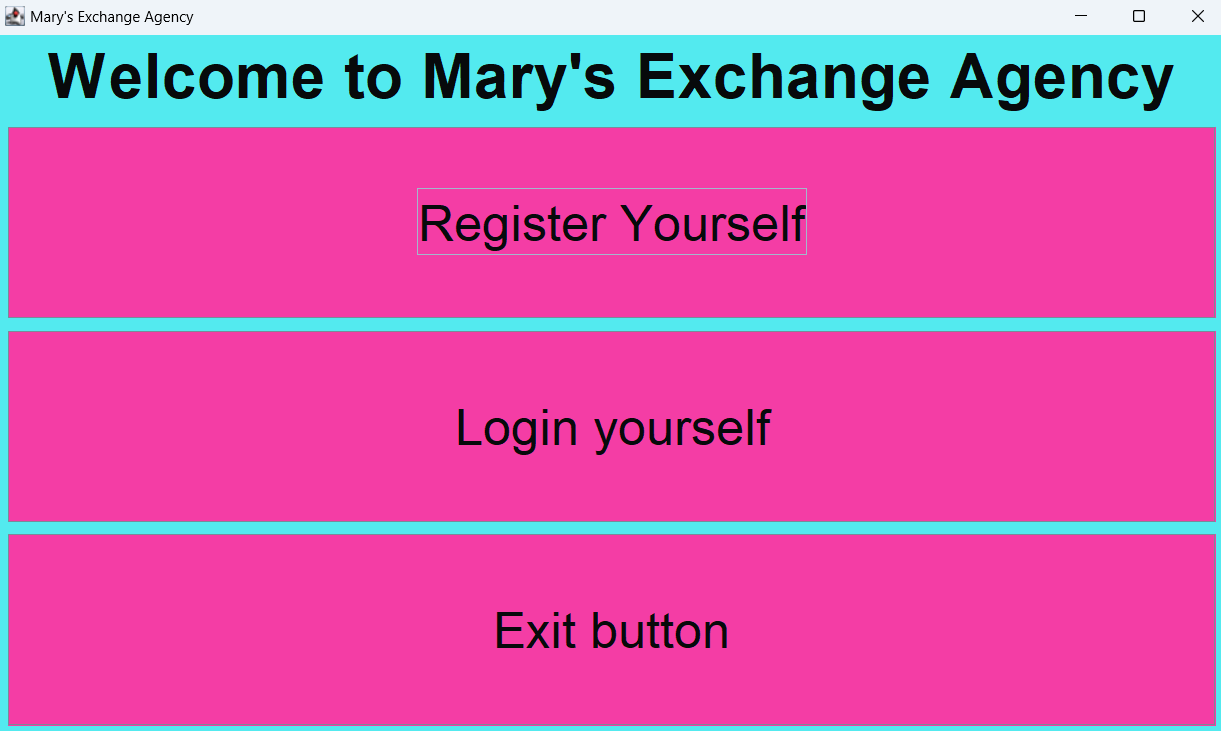
Challenge: Make sure that the data inputs that the users provide are credible and correct.

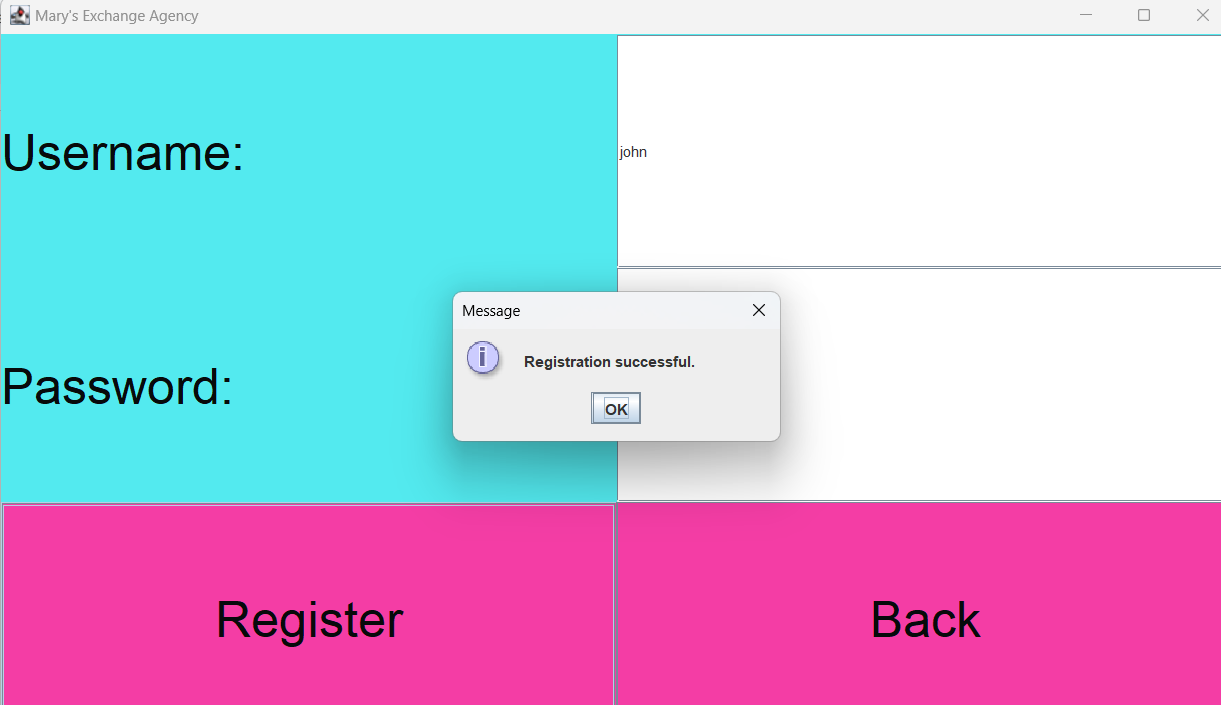
Solution: Applying harsh validation checks on the client and server interfaces to avoid getting the information in the wrong format.

# Implementation

***User Registration***

The user registration process itself is simple and safe to use for every client who plans to use the services of the website.





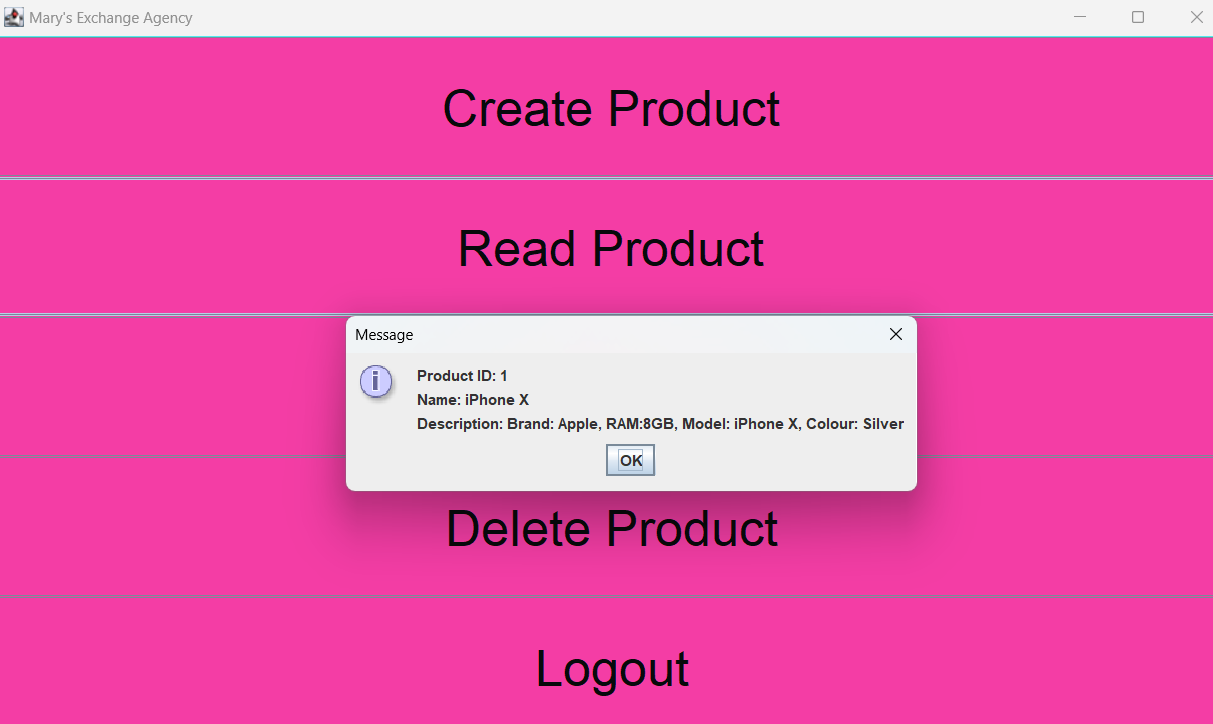
**Figure 2: User Registration**

(Source: Acquired from IntelliJ Software)

The registration form is developed using fashion paradigms JSP (JavaServer Pages) to make the form user-friendly and intelligent. The registration form is performed when a user completes it and submits the form; the Servlet will validate the information input by the user to ensure it is compliant with the necessary standard.

***Item Listing***

The item listing function enables users to type or describe the items to be exchanged systematically.



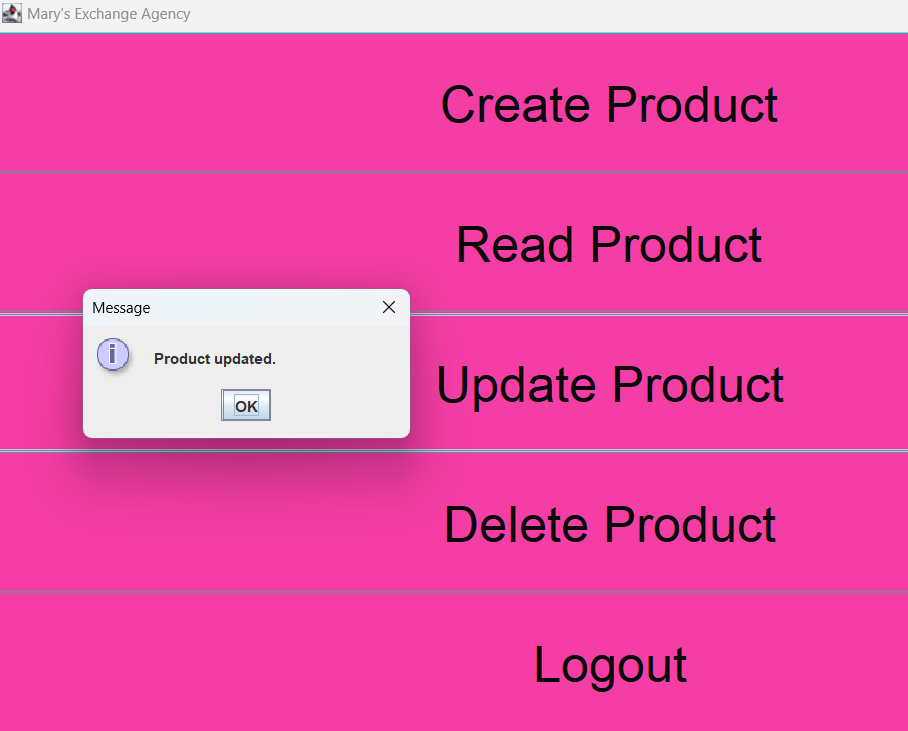
**Figure 3: Item Listing**

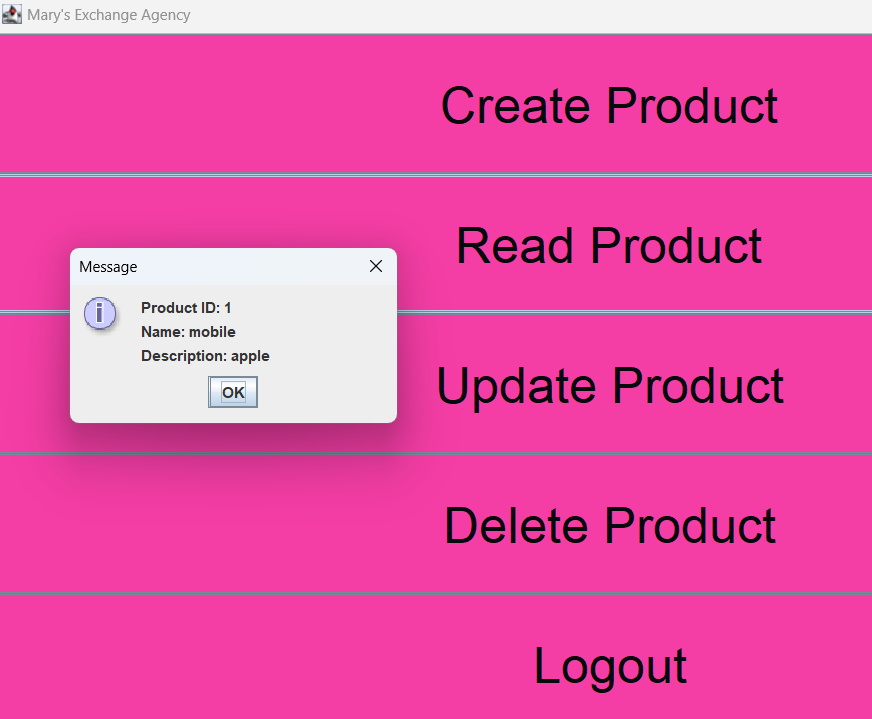
(Source: Acquired from IntelliJ Software)

The item listing form records key information like the item name, item description, its condition as well as an image. This information is important for the other users if in case they have an interest in the specific item. This page offers the form for adding new items and, in addition, includes the list of existing items that belong to the user.

***Browsing and Searching***

Navigation and search of items is the main activity whether through catalogs or by using search engines in the exchange platform.





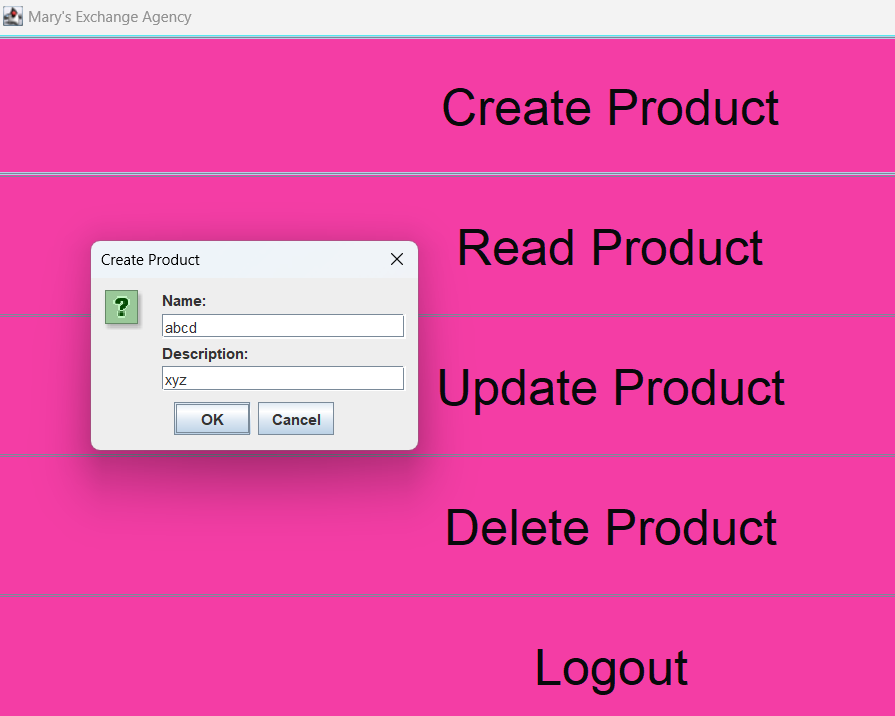
**Figure 4: Browsing and Searching**

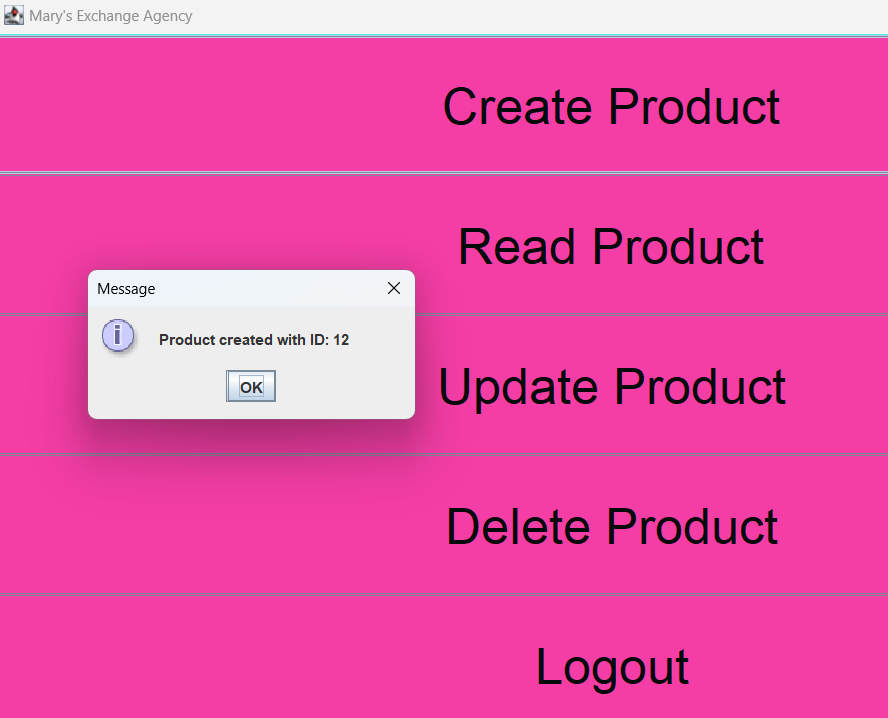
(Source: Acquired from IntelliJ Software)

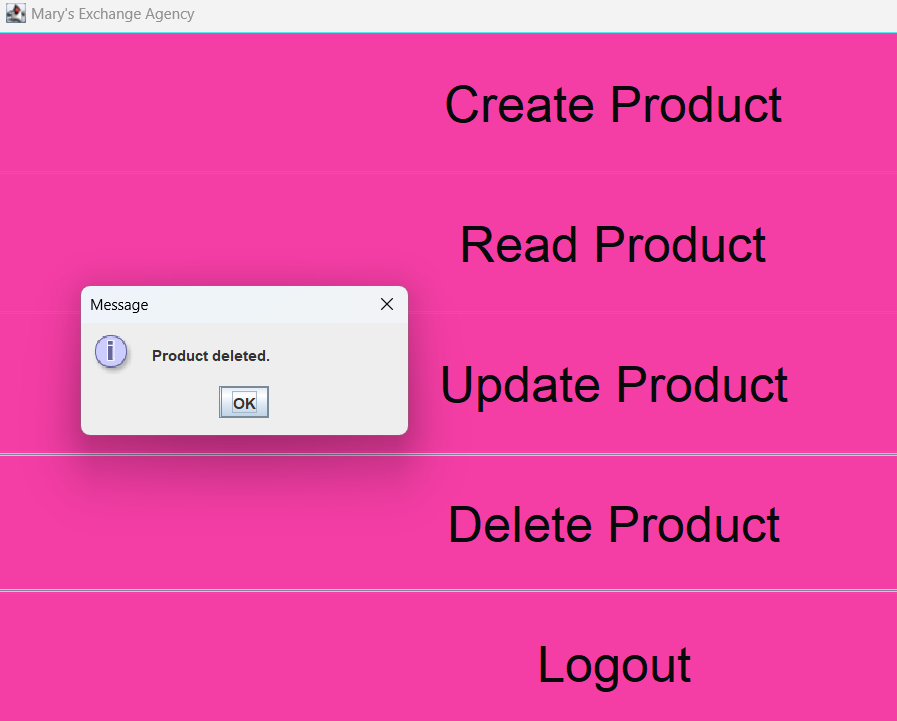
A JSP page is created to present the items while Servlets are used to search for the items and present them on the page. Indeed, users can run through pages of categorized listings and refine the results to their category of interest among other attributes.

***Exchanges Management***

Exchanges are one of the key components of the virtual environment; thus, the platform should allow users to track and finalize exchanges.







**Figure 5: Exchanges Management**

(Source: Acquired from IntelliJ Software)

Servlets deal with exchanges, changing the status, and making agreements between the users. Its interface presents the list of exchanges in its main tab, which indicates both completed and current exchanges for easy access to the users’ exchanges.

# Conclusion

This Exchange Agency Platform is suitable for exchange-based trading and the related functions include item posting, searching, and exchanging, as well as category recommendation. It uses modern technologies available in the market and well-adopted design patterns to achieve scalability, security, and usability goals. The integration of machine learning also simplifies the functionality where new postings are categorized as and when they are listed to make it easier for the users.

# 

# References

CHEN, Y., LU, Y., PAN, Z., CHEN, J., SHI, F., YANG, L. and JIANG, Y., 2024. APIMiner: Identifying Web Application APIs Based on Web Page States Similarity Analysis. Electronics, 13(6), pp. 1112.

CHIEN-HUNG, L., YOU, S.D. and CHIU, Y., 2024. A Reinforcement Learning Approach to Guide Web Crawlers to Explore Web Applications for Improving Code Coverage. Electronics, 13(2), pp. 427.

DAWADI, B.R., ADHIKARI, B. and SRIVASTAVA, D.K., 2023. Deep Learning Technique-Enabled Web Application Firewall for the Detection of Web Attacks. Sensors, 23(4), pp. 2073.

PEARKAO, C., POTISOPHA, W., WONGGOM, P., JUMPAMOOL, A., APIRATWARAKUL, K. and LENGHONG, K., 2023. Outcomes of Emergency Trauma Patients After the Implementation of Web Application Operating Systems. Asian Nursing Research, 17(3), pp. 174-179.

TADHANI, J.R., VEKARIYA, V., SORATHIYA, V., ALSHATHRI, S. and EL-SHAFAI, W., 2024. Securing web applications against XSS and SQLi attacks using a novel deep learning approach. Scientific Reports (Nature Publisher Group), 14(1), pp. 1803.

VAREWYCK, M. and VERBEKE, T., 2024. Web Application for Multi-Drug Resistance. EFSA Supporting Publications, 21(3),.

VERLINDEN, W., VAREWYCK, M. and VERBEKE, T., 2024. Web application for Bayesian Benchmark Dose Modelling. EFSA Supporting Publications, 21(1),.

WON-CHI, J., KIM, J. and PARK, N., 2021. Web-Browsing Application Using Web Scraping Technology in Korean Network Separation Application. Symmetry, 13(8), pp. 1550.

YANG, Z., PRIYATAMA, P., MARJORAM, P. and SIEGMUND, K.D., 2020. iMutSig: a web application to identify the most similar mutational signature using Shiny. F1000Research, 9.

ZHANG, J., QIANG, J. and ZHOU, C., 2024. New Horizons in Web Search, Web Data Mining, and Web-Based Applications. Applied Sciences, 14(2), pp. 530.