

CAMPUS SECOND HAND TRADING PLATFORM

ABSTRACT: This study aims to develop a front-end and back-end separated architecture for a second-hand trading platform. It provides an efficient and user-friendly platform for users to easily post, browse, purchase, and review second-hand items. The front-end utilizes HTML, CSS, and JavaScript, while the back-end employs Java language and the Spring Boot framework. Additionally, Navicat is used as the database management tool for MySQL. The platform encompasses key functional modules including user authentication and management, item posting and browsing, transactional purchasing, reviewing and feedback, reporting functionalities, announcement management, and system configuration. The project adheres to the front-end and back-end separation development model, enabling communication between the two through APIs to ensure system efficiency and flexibility. Project implementation will leverage Maven for project management and construction, with a focus on user-friendly interface design to enhance overall user experience.

Keywords: Spring Boots; Java-web; MVC.

INTRODUCTION

In the digital age, the growth of the second-hand market has been significantly catalyzed by the advent of online trading platforms. These platforms have transformed traditional peer-to-peer transactions, offering users the convenience to buy, sell, and exchange goods via specialized websites and apps. This shift not only promotes sustainable consumption by extending the lifecycle of products but also presents an economical alternative for consumers seeking value in pre-owned items. Despite their rising popularity, existing second-hand trading platforms often grapple with challenges such as inefficient user interfaces, inadequate security measures, and a lack of robust functionalities that can support seamless user interactions and transactions.

The proposed project aims to develop a front-end and back-end separated architecture for a second-hand trading platform, focusing on the unique needs of a university campus environment. The platform seeks to address the gaps observed in existing systems by enhancing usability, security, and performance. By employing cutting-edge web technologies and a comprehensive development approach, the project endeavors to create a user-friendly, secure, and efficient marketplace for second-hand goods.

The platform will incorporate several key features to ensure its effectiveness and reliability:

User Authentication and Management: Robust mechanisms will be implemented to manage user registrations, logins, and profile management, ensuring that all interactions on the platform are secure.

Item Posting and Browsing Interface: An intuitive interface will allow users to easily post items for sale and browse available listings. Detailed descriptions, images, and categorization will enhance the shopping experience and aid in item discovery.

Transactional Support: The platform will facilitate secure transactions, including options for payment processing and order tracking, to ensure the authenticity of transactions and minimize the risk of fraud.

Administrative Dashboard: Administrators will be equipped with powerful tools to oversee platform activities, manage user accounts, moderate listings, and generate reports to maintain the platform's integrity and operational efficiency.

By focusing on these areas, the project will not only provide a practical solution to the challenges faced by current platforms but also enhance the economic and environmental benefits of second-hand trading. This initiative is aligned with global sustainability goals, advocating for reduced waste and promoting the reuse of goods. As such, the platform will not only serve the immediate needs of the university community but also contribute to broader societal change, encouraging more responsible consumption patterns and resource use.

In conclusion, this project represents a significant step forward in the evolution of second-hand trading platforms. By leveraging modern technology and innovative design principles, it aims to deliver a superior user experience, robust security features, and comprehensive functionality that will set a new standard for online second-hand marketplaces.

RELATED WORKS

The rapid advancement of technology has significantly elevated the importance of second-hand trading platform systems today. These platforms offer individuals and businesses a convenient way to buy and sell used goods, promoting resource reuse and supporting sustainable development. Through digitalization and online accessibility, these platforms provide a broad market where users can easily trade items ranging from electronic devices to furniture and clothing. This not only benefits users economically but also contribute to the sustainable use of resources.

The emergence of second-hand trading platform systems can be attributed to the evolving demands of the traditional market and technological advancements. Traditionally, second-hand transactions occurred in physical markets, garage sales, or through advertisements, limiting participant scope and market size. However, the widespread adoption of the internet and mobile devices has created a demand for more convenient and efficient second-hand transactions. This led to the rise of online platforms connecting sellers and buyers, offering a wider selection, more convenience, and greater transparency. These platforms have become integral to everyday life, especially through apps like Idle Fish and Carousell, and social media groups on WhatsApp and Facebook.

Problems with Current Second-Hand Trading Apps:

Quality and Valence Inequality: Buyers often rely solely on seller-provided information, leading to information asymmetry and potential financial losses. Sellers may not provide comprehensive details about the item's condition, origin, usage history, or fair market value.

Overpricing: Sellers control pricing, often leading to overpricing. Buyers adjust their perceived value based on the seller's anchor price, resulting in higher payments than the actual value of items.

Lack of Services: Transactions typically terminate upon item receipt and payment, with no after-sales support. This results in a high level of user dissatisfaction, mainly due to inadequate after-sales services.

Problems with Current Social Media Second-Hand Selling Groups:

Information Security Issues: Information in these groups is complex and not scientifically categorized, leading to difficulties in verifying authenticity and timeliness. This results in information acquisition losses and increased transaction insecurity.

Transaction Security Issues: The lack of formal third-party payment platforms poses risks, as buyers and sellers rely on simple communication and trust. This can lead to financial losses due to non-fulfillment of transaction obligations.

Privacy Security Issues: These groups have low entry thresholds and lack identity verification, leading to privacy risks and difficulties in accountability. Users' personal information can be leaked or maliciously used.

Compare and Contrast:

Developing a second-hand trading platform typically involves modernized approaches like front-end and back-end separation for enhanced efficiency and user experience. In back-end development, developers often choose between the Spring Boot framework and traditional Servlet technology.

Spring Boot:

Advantages: Provides a highly integrated environment, extensive ecosystem, and automatic configuration, enhancing development efficiency.

Weaknesses: Steep learning curve for beginners.

Servlet:

Advantages: Offers more flexibility and control over low-level details, suitable for simple projects.

Weaknesses: Requires more manual configuration, increasing complexity and lacking advanced features.

Comparison of Idle Fish and Carousell:

Feature	Carousell	Idle Fish
User Base	Popular in Asia with a diverse user base	Mainly targets the Chinese market
Payment and Integration	Varies by region with third-party providers	Integrated with Alibaba services like Alipay
Social Interaction	Strong social networking elements	Weaker social interaction, focus on transactions
Product Categories	Wide range, including clothing, electronics, home goods	Diverse categories for various second-hand goods
Interface Design	Clean, user-friendly	Simple, facilitating browsing and shopping

Table 2.1 The comparison of Idle Fish and Carousell

METHODS

The first step in developing the campus second-hand trading platform involved conducting a thorough requirements analysis. This process included understanding user behaviors, transaction types, and payment preferences to ensure the platform's functionalities meet user needs and enhance their experience. The analysis also considered security features such as identity verification and transaction protection to safeguard user information and transactions. Additionally, the user interface design was tailored to be intuitive and user-friendly, aligned with the target users' habits.

Functional Requirements

Registration: Users can register on the platform to access its features. The registration process includes username validation and form completion checks.

Login: Users can log in using their registered credentials to access the main page.

Product Selling: Users can list their second-hand items for sale by providing necessary details on a designated page.

Reporting and Commenting: Logged-in users can report or comment on items, providing reasons for their actions.

Product Requesting: Users can request specific second-hand items by providing relevant information, enabling other users with matching items to contact them.

Non-Functional Requirements

Security: Ensuring the platform's security is paramount, including measures like password encryption to protect user data and prevent unauthorized access.

Performance: The platform should be simple, convenient, and user-friendly, with regular updates to enhance processing capabilities and minimize page loading times.

Scalability and Maintenance: The system should support future enhancements and regular updates based on real-time user feedback to maintain user satisfaction and trust.

System Design

Use Case Diagrams were utilized to depict the interactions between the system and external entities, focusing on the roles of users and administrators. These diagrams provided a high-level overview of system functionalities, aiding in communication with stakeholders and supporting requirements analysis, system design, and testing.

In the context of a campus second-hand trading platform system, the Use Case Diagram is crucial for outlining system functionalities, emphasizing user perspectives, defining system boundaries, facilitating communication with stakeholders, supporting requirements analysis, designing test cases, aiding in system design, and managing changes throughout the development process. It provides a high-level overview of the system's behavior, ensuring a shared understanding among developers and stakeholders, and serving as a foundational tool for subsequent stages of system development, including design, implementation, and testing. Figure 3.1 shows the use case diagram of user. Figure 3.2 shows the use case diagram of admin.

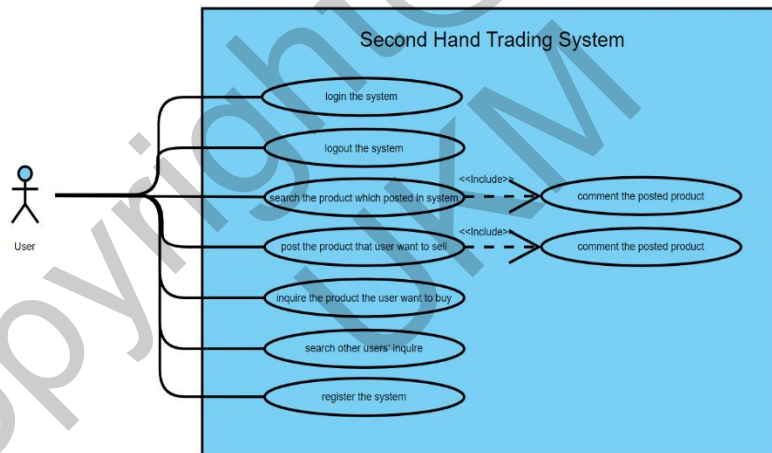


Figure 3.1 Use case diagram for user (buyer and seller).

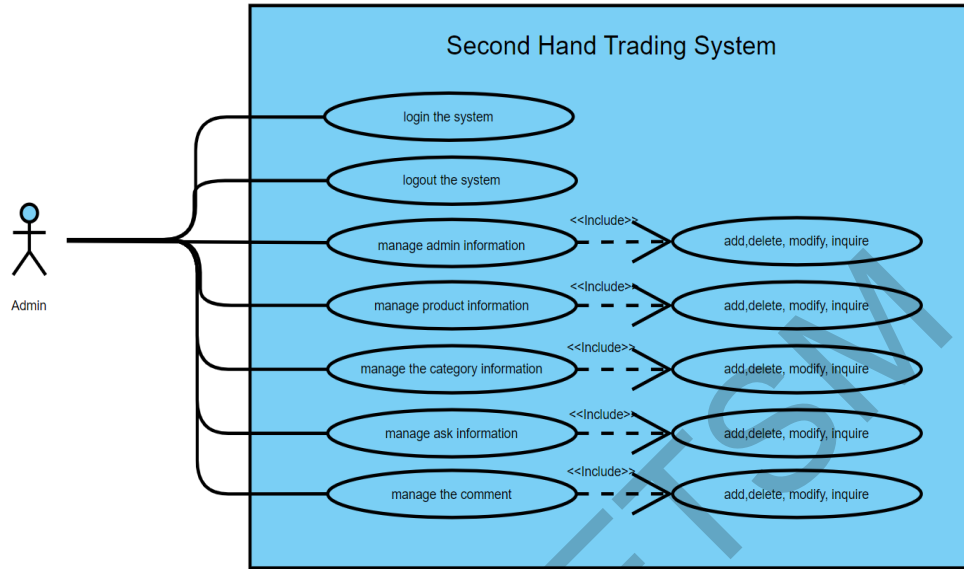


Figure 3.2 Use case diagram for Admin.

Use case specification

The Use Case Specification is a detailed document that outlines specific use cases, User Login providing comprehensive information such as the use case's goals, preconditions, postconditions, basic flow, and alternative Scenarios. In the context of a campus second-hand trading platform system, it plays a crucial role by clearly defining the behavior of each use case, guiding development and testing efforts, emphasizing preconditions and postconditions, supporting system design, reducing errors in development and testing, facilitating communication with stakeholders, and aiding in change management. The Use Case Specification ensures a shared and precise understanding of system functionalities, contributing to the overall quality and maintainability of the system.

User login

Table 3.1 shows the use case specification for user login.

Table 3.1 Use case specification user login.

ID	UCS1
Title	Users Login
Description	Allows a registered user to access the system by entering their valid username and password. Provides a secure authentication process for user accounts.
Primary Actor	User and Admin
Precondition	User has a valid account.
Postcondition	User successfully logs into the system.
Basic flow	User accesses the login page. User enters a valid username and password. System verifies user credentials. If verification is successful, the system allows the user to log in.

Alternative Scenario	If the entered username or password is invalid, the system displays an error message and prompts the user to re-enter.
----------------------	--

User Register

Table 3.2 shows the use case specification for user registration.

Table 3.2 Use case specification user registration.

ID	UCS2
Title	User Register
Description	This use case enables a user to create a new account, providing necessary information for registration.
Primary Actor:	User
Precondition:	User does not have a registered account.
Postcondition	User successfully registers a new account.
Basic flow	User accesses the registration page. User fills in required registration details (username, password, email, etc.). User submits the registration form. System validates the information and creates a new user account.
Alternative Scenarios:	If the chosen username is already taken, the system prompts the user to choose a different one.

User Logout

Table 3.3 shows the use case specification for user logout.

Table 3.3 Use case specification user logout.

ID	UCS3
Title	User Logout
Description	This use case allows a user to securely log out of the system.
Primary Actor:	User, Admin
Precondition:	User is logged into the system.
Postcondition	User is logged into the system.
Basic flow	User accesses the logout option. System logs the user out and redirects to the login page.
Alternative Scenarios:	-

User Search Second-Hand Products

Table 3.4 shows use case specification for user search second-hand products.

Table 3.4 Use case specification user search second-hand products.

ID	UCS4
Title	User Search Second-Hand Products
Description	This use case enables a user to search for second-hand products based on specific criteria.
Primary Actor:	User
Precondition:	User is logged into the system.
Postcondition	User successfully views search results.
Basic flow	User accesses the search page. User enters keywords or selects search criteria. User triggers the search operation. System displays relevant second-hand products based on the user's criteria.
Alternative Scenarios:	If no search criteria are provided, the system may show popular or recommended products.

User Post Second-Hand Product for Sale

Table 3.5 shows use case specification for user post second-hand product for sale.

Table 3.5 Use case specification user post second-hand product for sale.

ID	UCS5
Title	User Post Second-Hand Product for Sale
Description	This use case enables a user to search for second-hand products based on specific criteria.
Primary Actor:	User
Precondition:	User is logged into the system.
Postcondition	User successfully views search results.
Basic flow	User accesses the search page. User enters keywords or selects search criteria. User triggers the search operation. System displays relevant second-hand products based on the user's criteria.
Alternative Scenarios:	If no search criteria are provided, the system may show popular or recommended products.

User Post Wanted items.

Table 3.6 shows use case specification for user post wanted items.

Table 3.6 Use case specification user post wanted items.

ID	UCS6
Title	User Post Wanted items
Description	This use case allows a user to post a wanted item, expressing interest in purchasing specific second-hand products.
Primary Actor:	User
Precondition:	User is logged into the system.
Postcondition	User successfully posts a wanted item.
Basic flow	User accesses the wanted ad posting page. User fills in details about the desired product and any specific requirements. User submits the wanted ad. System displays the posted wanted ad on the platform.
Alternative Scenarios:	If the user's input is incomplete, the system prompts the user to provide all necessary information.

User View Others' Wanted items

Table 3.7 shows use case specification for user view others' wanted items.

Table 3.7 Use case specification user view others' wanted items.

ID	UCS7
Title	User View Others' Wanted items
Description	This use case allows a user to view wanted items posted by other users on the platform.
Primary Actor:	User
Precondition:	User is logged into the system.
Postcondition	User successfully views others' wanted items.
Basic flow	User accesses the wanted items page. System displays a list of wanted items posted by other users.
Alternative Scenarios:	If the user's input is incomplete, the system prompts the user to provide all necessary information.

Admin Manage User Information

Table 3.8 shows use case specification for admin manage user information.

Table 3.8 Use case specification admin manage user information.

ID	UCS8
Title	Admin Manage User Information
Description	This use case allows an administrator to manage user information, including deleting, modifying, and viewing user profiles.
Primary Actor:	Admin
Precondition:	Admin is logged into the admin panel.
Postcondition	Admin successfully manages user information.
Basic flow	Admin accesses the user management section. Admin selects the desired user profile. Admin performs actions such as adding, deleting, or modifying user information.
Alternative Scenarios:	If an error occurs during the management process, the system provides feedback to the admin.

Admin Manage Admin Information

Table 3.9 shows use case specification for admin manage admin information.

Table 3.9 Use case specification admin manage admin information.

ID	UCS9
Title	Admin Manage Admin Information
Description	This use case allows an administrator to manage other administrators' information, including adding, deleting, modifying, and viewing admin profiles.
Primary Actor:	Admin
Precondition:	Admin is logged into the admin panel.
Postcondition	Admin successfully manages user information.
Basic flow	Admin accesses the admin management section. Admin selects the desired admin profile. Admin performs actions such as adding, deleting, or modifying user information.
Alternative Scenarios:	If an error occurs during the management process, the system provides feedback to the admin.

Admin Manage Product Information

Table 3.10 shows use case specification for admin manage product information.

Table 3.10 Use case specification admin manage product information.

ID	UCS10
Title	Admin Manage Product Information
Description	This use case allows an administrator to manage product information, including adding, deleting, modifying, and viewing product details.
Primary Actor:	Admin
Precondition:	Admin is logged into the admin panel.
Postcondition	Admin successfully manages product information.
Basic flow	Admin accesses the product management section. Admin selects the desired product. Admin performs actions such as adding, deleting, or modifying product details.
Alternative Scenarios:	If an error occurs during the management process, the system provides feedback to the admin.

Admin Manage Category Information

Table 3.11 shows use case specification for admin manage category information.

Table 3.11 Use case specification admin manage category information.

ID	UCS11
Title	Admin Manage Category Information
Description	This use case allows an administrator to manage category information, including adding, deleting, modifying, and viewing product categories.
Primary Actor:	Admin
Precondition:	Admin is logged into the admin panel.
Postcondition	Admin successfully manages category information.
Basic flow	Admin accesses the category management section. Admin selects the desired category. Admin performs actions such as adding, deleting, or modifying category details.
Alternative Scenarios:	If an error occurs during the management process, the system provides feedback to the admin.

Admin Manage Wanted Items Information.

Table 3.12 shows use case specification for admin manage wanted items information.

Table 3.12 Use case specification admin manage wanted items information.

ID	UCS12
Title	Admin Manage Wanted Items Information
Description	This use case allows an administrator to manage information related to wanted ads, including adding, deleting, modifying, and viewing wanted ads.
Primary Actor:	Admin
Precondition:	Admin is logged into the admin panel.
Postcondition	Admin successfully manages wanted ad information.
Basic flow	Admin accesses the wanted ad management section. Admin selects the desired wanted ad. Admin performs actions such as adding, deleting, or modifying wanted ad details.
Alternative Scenarios:	If an error occurs during the management process, the system provides feedback to the admin.

Admin Manage Comment Information

Table 3.13 shows use case specification for admin manage comment information.

Table 3.13 Use case specification admin manage comment information.

ID	UCS13
Title	Admin Manage Comment Information
Description	This use case allows an administrator to manage comments, including adding, deleting, modifying, and viewing user comments.
Primary Actor:	Admin
Precondition:	Admin is logged into the admin panel.
Postcondition	Admin successfully manages wanted ad information.
Basic flow	Admin accesses the comment management section. Admin Selects the desired comment. Admin performs actions such as adding, deleting, or modifying comment details.
Alternative Scenarios:	If an error occurs during the management process, the system provides feedback to the admin.

Payment

Table 3.14 shows use case specification for payment.

Table 3.14 Use case specification payment.

ID	UCS14
Title	Payment
Description	The Payment Page use case describes the process where a user makes a direct purchase of an item on the second-hand platform and proceeds with the payment.
Primary Actor	User
Precondition	The user is logged in and has selected the item to purchase. The user has provided valid payment information
Postcondition	The transaction is successfully completed, and the shopper's account is debited.
Basic flow	The user selects the item to purchase on the platform. The user proceeds to the payment page, and the system displays item information and payment options. The user chooses a payment method. The payment system processes the payment request and returns the payment result.
Alternative Scenarios	If the user chooses to cancel the payment during the process, the system cancels the payment and returns to the item selection page without executing the payment operation. If the payment system does not return a payment result within the specified time, the system prompts the user to retry the payment or provides alternative payment options.

Architecture diagram

Figure 3.3 shows the architecture diagram for campus secondhand trading platform. This diagram illustrates the architecture of a campus second-hand trading platform based on a web application. Users interact with the system through the front-end interface, which collects user data and displays results. When users make requests, such as searching for items, the front end sends these requests to the back end. The back end handles the business logic and sends requests to the database to retrieve or store necessary data. The database stores all the essential data needed by the application, such as user information, product listings, and transaction records. Additionally, image storage is used to save images related to product listings or user profiles. Once the back end retrieves data from the database, it generates a response and sends it back to the front end, which then displays the processed results to the user. This architecture ensures that users can efficiently and conveniently utilize the application's functionalities. The front end provides an intuitive user interface, the back end manages business logic and data processing, and the database and image storage ensure that all necessary information is readily available, thereby facilitating smooth system operation and a positive user experience.

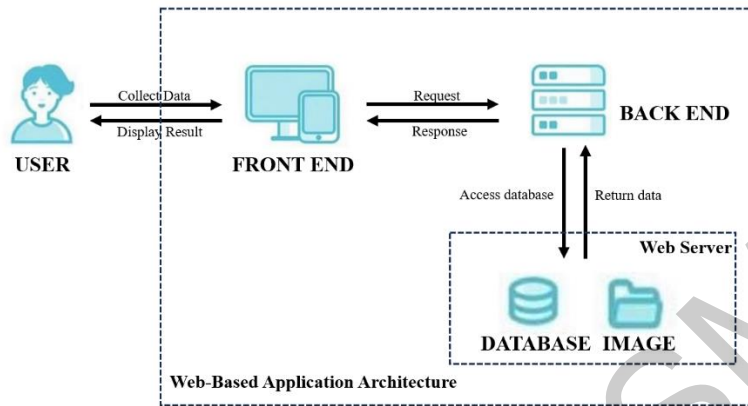


Figure 3.3 Architecture diagram for campus second hand trading platform

Dataset Design

Database design is the combination of object-oriented method and database, which can simply analyze the database and facilitate the communication between database developers and users. Second-hand trading platforms have a few classes name Students, Goods, Report, Wanted_goods, Category, Comment, Admin.

A student can list multiple goods, with each good belonging to a specific category. Reports are associated with students, signifying that a student can initiate multiple reports, and each report is directed towards the reported user. Goods are open to comments, and a single good can have multiple comments, each posted by a student. WantedGoods, posted by students, can be associated with a specific good. The admin entity is responsible for system management, establishing a one-to-many relationship with all entities. These relationships are primarily implemented through foreign keys, establishing connections between entities in the system. This design aims to fulfill the functional requirements of a second-hand platform trading system. The Figure 4.2 shows the ERD of this project.

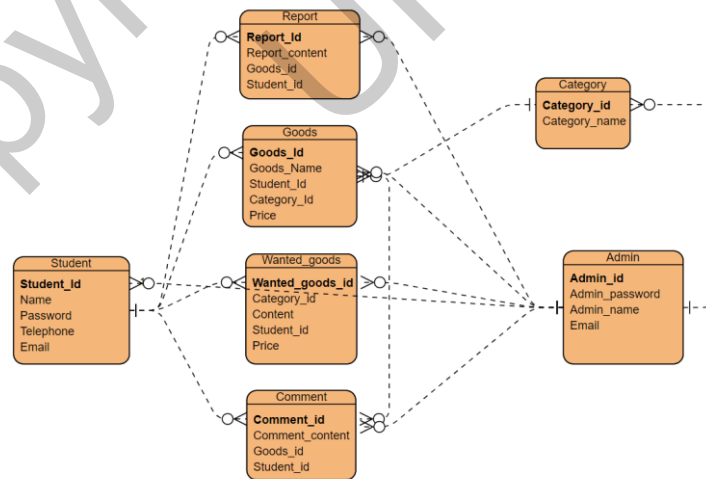


Figure 3.4 ERD for campus second hand trading platform

TESTING

The system testing for the Second-hand Management System aims to identify discrepancies between user expectations and the system's actual performance, uncover hidden defects, and enhance overall system

quality. This process is crucial in ensuring that the system meets both functional and non-functional requirements, providing a reliable and user-friendly platform for second-hand trading.

Testing Plan

The primary focus of the testing plan is to validate the system's functionality against the requirements, identify discrepancies, uncover defects, and ensure system robustness and reliability. The testing scope includes key functionalities for users (login, registration, item posting, wanted item posting, and product search) and administrators (managing user information and items). The testing is based on functional and non-functional requirements, system specifications, and design documents, utilizing black box testing techniques.

Environment Setup

The testing environment includes:

Software: JDK 1.8, MYSQL 8.0, IntelliJ IDEA 2020.1.1, Navicat Premium 16, Maven 3.6.3, Windows 10, Chrome 124.0.6367.208.

Hardware: Internet 500Mbps, CPU Intel(R) Core (TM) i7-10710U @ 1.10GHz 1.61 GHz, GPU NVIDIA 3060.

Execution and Monitoring

Test cases were executed according to the test procedures, recording actual results and comparing them with expected results. Any defects found were logged, and the test execution process was continuously monitored, with updates provided to stakeholders on progress.

CONCLUSION

Overview

This project aimed to develop a robust campus second-hand goods trading platform, based on extensive research and analysis within the current social context. The platform employs a B/S architecture, ensuring front-end and back-end separation, allowing users and administrators to access different web pages with specific account credentials. The user-friendly and aesthetically pleasing front-end interface provides all necessary functionalities, while the feature-rich back-end enables efficient management of user requests and operations by administrators. A key feature is the comments module, which helps users assess seller credibility and product authenticity through reviews, thereby reducing the risk of fraud. The platform promotes the reuse of idle items within the campus, enhancing resource efficiency and environmental protection. Despite meeting the identified user needs, the system requires further optimization and expansion due to evolving technology and user demands. Areas for improvement include verifying product authenticity and preventing the misuse of multiple accounts.

Constraints

The project faces two main constraints:

Payment Integration: Currently, the platform lacks a payment processing system, necessitating that users arrange payments independently outside the system.

Email Verification: The platform cannot verify if a registrant's email address is associated with the school, posing challenges in ensuring that only legitimate campus community members can register and use the platform.

Future Suggestion

To enhance the Second-hand Management System, several improvements can be considered:

Payment Integration: Implementing third-party payment services like PayPal or Stripe, or developing an in-app digital wallet system, to facilitate secure and seamless transactions.

Email Verification: Introducing a school email verification system to ensure that only legitimate campus members can register, potentially through verification links or integrating Single Sign-On (SSO) with the school's authentication system.

Security Enhancements: Adding user verification steps such as two-factor authentication (2FA) to improve account security and implementing automated and manual content moderation tools to prevent fraudulent or inappropriate listings.

User Experience Improvements: Developing a mobile application for both iOS and Android platforms to allow users convenient access on-the-go and conducting regular user feedback sessions to refine the user interface for better usability.

Community Features: Incorporating a comprehensive rating and review system to build trust and reliability and adding a forum and direct messaging feature to facilitate better communication between users, helping resolve disputes and foster a sense of community.

These enhancements aim to improve the functionality, security, and user satisfaction of the platform, ensuring it remains effective and user-friendly in the long term.

REFERENCES

- Ambler, S. W. (2004). "Introduction to UML 2 Use Case Diagrams." Retrieved from Agile Modeling.
- Abrahamsson, P., Warsta, J., Siponen, M. T., & Ronkainen, J. (2003). New directions on agile methods: A comparative analysis. Proceedings of the 25th International Conference on Software Engineering.
- Belk, R. (2014). You are what you can access: Sharing and collaborative consumption online. *Journal of Business Research*, 67(8), 1595-1600.
- Chen, L., & Hu, Y. J. (2013). The impact of website quality on customer satisfaction and purchase intention: perceived playfulness and perceived flow as mediators. *Information Systems Frontiers*, 15(3), 417-434.
- Consumer-to-Consumer Secondhand Marketplace Platform. *Behavioral sciences (Basel, Switzerland)*, 13(3), 242. <https://doi.org/10.3390/bs13030242>
- Felippe, L. (2017). *Learning Spring Boot 2.0: Simplify the development of lightning-fast applications based on microservices architecture*, 2nd Edition. Packt Publishing.
- Geissdoerfer, M., Savaget, P., Bocken, N. M., & Hultink, E. J. (2017). The Circular Economy—A new sustainability paradigm?. *Journal of cleaner production*, 143, 757-768.
- Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (2015). *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley.
- Hai-Tao, Zhang. (2015). Campus Second-Hand Goods Trading Platform Design Based on BS. 261-264. 10.1109/ICICTA.2015.73.
- Hamari, J., Sjöklint, M., & Ukkonen, A. (2016). The sharing economy: Why people participate in collaborative consumption. *Journal of the association for information science and technology*, 67(9), 2047-2059.

- Hunter, J., & Crawford, W. (2014). *Java Servlet & JSP Cookbook*. O'Reilly Media.
- Jang, Y., & Kim, S. (2023). The Factors Influencing Users' Trust in and Loyalty to.
- Hossain, M. A., & Kauranen, I. (2015). Mobile payment adoption in the age of smartphones. *Journal of financial services marketing*, 20(2), 150-166.
- Pressman, R. S. (2014). *Software Engineering: A Practitioner's Approach*. McGraw-Hill Education.
- Pressman, R. S. (2014). *Software Engineering: A Practitioner's Approach (8th ed.)*. McGraw-Hill Education.
- Silberschatz, A., Korth, H. F., & Sudarshan, S. (2010). *Database System Concepts*. McGraw-Hill.
- Sharma, A. (2018). *Spring Boot: Up and Running*. O'Reilly Media.
- Tondreau, B. (2014). *Layout Essentials: 100 Design Principles for Using Grids*. Rockport Publishers.
- Visual Paradigm. (n.d.). What is Use Case Specification? Retrieved from <https://www.visual-paradigm.com/guide/use-case/what-is-use-case-specification/>
- Vermeulen, S. J., Campbell, B. M., & Ingram, J. S. (2012). Climate change and food systems. *Annual Review of Environment and Resources*, 37, 195-222.
- Wall, C., Macy, J., & Dorsey, C. (2003). *Spring: A Developer's Notebook*. O'Reilly Media.
- Wall, C., Macy, J., & Dorsey, C. (2019). *Spring Boot: Up and Running: Building Cloud Native Java and Kotlin Applications*. O'Reilly Media.
- Wikipedia. (n.d.). Use case diagram. Retrieved from https://en.wikipedia.org/wiki/Use_case_diagram
- Wieggers, K. E., & Beatty, J. (2013). *Software Requirements (3rd ed.)*. Microsoft Press.
- Whitten, J. L., Bentley, L. D., & Dittman, K. C. (2007). *Systems Analysis and Design Methods (7th ed.)*. McGraw-Hill Education.
- Wisakha Phoochinda, Saraporn Kriyapak. Electronic Waste Recycling Business: Solution, Choice, Survival[J]. *IJSDP*, 2021, 16(4): 4~20.

