

**BEACONHOUSE NATIONAL UNIVERSITY**

**PRJ-F23/337**

**THRIFTN’:**

**A marketplace for pre-loved items**

**EXTERNAL SUPERVISOR**

Muhammad Danyal

**INTERNAL SUPERVISOR**

Ms. Huda Sarfraz

**GROUP MEMBERS**

**Aiman Zahoor F2020-550**

**Zainab Usama F2020-372  
  
  
SCHOOL OF COMPUTER & INFORMATION TECHNOLOGY**

Project Title **Thriftn’**

|  |  |
| --- | --- |
| Project ID | **PRJ-F23/337** |
| Project Supervisors  Group Members | **Huda Sarfraz**  **Muhammad Danyal**  **Aiman** **Zahoor F2020-550**  **Zainab Usama F2020-372** |
| Academic Session | **2023-24 (Sept 2023 to June 2024)** |
| Credit Hours | **6** |

**PROJECT APPROVAL**

This Project is approved in partial fulfillment of the requirements of BSc (Hons.) in computer science degree conducted by the School of Computer and IT, Beaconhouse National University, Lahore.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Muhammad Danyal Prof. Dr. Shafaat Ahmed Bazaz  
External Supervisor Dean School of Computer and IT**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Huda Sarfraz   
Internal Supervisor**

**Acknowledgement**

We express our deepest gratitude to everyone who supported us throughout our Final Year Project. We are particularly thankful to our project supervisors Ms. Huda Sarfraz, Ms. Shazia Rizwan and Mr. Muhammad Danyal for their continuous support, patience, and insightful guidance.

We also thank the faculty and staff of School of Computer and IT at Beaconhouse National University for providing a conducive environment for learning and research. Our heartfelt thanks go to our fellow students and project team members for their collaboration and moral support.

Lastly, we extend our sincere appreciation to our families and friends for their unwavering encouragement during this journey. Your belief in us has been a source of strength and motivation.

Thank you all for your invaluable support and contributions.

Aiman Zahoor

Zainab Usama



**Link for project code and resources**

<https://github.com/aimanzahoor888/finalproject.git>

**Table of Contents**

[1 Introduction to the Project 8](#_Toc172639070)

[1.1 Existing situation and motivation for the project 8](#_Toc172639071)

[1.2 Literature review 9](#_Toc172639072)

[1.3 Additional research 11](#_Toc172639073)

[2 Requirement Analysis 12](#_Toc172639074)

[2.1 Requirement gathering and fact finding 12](#_Toc172639075)

[2.2 Use case analysis 14](#_Toc172639076)

[2.2.1 User roles 14](#_Toc172639077)

[2.2.2 Key user stories 14](#_Toc172639078)

[2.2.3 Use case diagram Key user stories 15](#_Toc172639079)

[2.3 Functional requirements 17](#_Toc172639080)

[2.4 Non-functional requirements 33](#_Toc172639081)

[3 Design 37](#_Toc172639082)

[3.1 Architecture 37](#_Toc172639083)

[3.2 Key sequence diagrams 39](#_Toc172639084)

[3.3 Data design 42](#_Toc172639085)

[4 Implementation 45](#_Toc172639086)

[4.1 Test cases 46](#_Toc172639087)

[4.2 Key implementation details 53](#_Toc172639088)

[5 Conclusion 62](#_Toc172639089)

[6 References 62](#_Toc172639090)

**Table of Figures**

[Figure 1.1: Factors promoting thrift stores 9](#_Toc171454112)

[Figure 2.1: Preference of Items 13](#_Toc171454113)

[Figure 2.2: Preference of Features 13](#_Toc171454114)

[Figure 2.3: Use case diagram 16](#_Toc171454115)

[Figure 3.1: Mern Stack Structure 37](#_Toc171454116)

[Figure 3.2: System Context Diagram 38](#_Toc171454117)

[Figure 3.3: Container Diagram 38](#_Toc171454118)

[Figure 3.4: High Level Architecture Diagram 39](#_Toc171454119)

[Figure 3.5: Key Sequence diagram 40](#_Toc171454120)

[Figure 3.6: Key Sequence Diagram 41](#_Toc171454121)

[Figure 3.7: Order Schema 42](#_Toc171454122)

[Figure 3.8: Negotiation Schema 42](#_Toc171454123)

[Figure 3.9: Product Schema 43](#_Toc171454124)

[Figure 3.10: User Schema 43](#_Toc171454125)

[Figure 3.11: Data Design Diagram 44](#_Toc171454126)

[Figure 4.1: Code Snippet for Trusted Seller Badge 53](#_Toc171454127)

[Figure 4.2: Mobile Screenshot for Trusted Seller Badge 54](#_Toc171454128)

[Figure 4.3: Code Snippet for Seller Side Negotiation 54](#_Toc171454129)

[Figure 4.4: Mobile Screenshot of Seller Side Negotiation 55](#_Toc171454130)

[Figure 4.5: Code Snippet for Search 55](#_Toc171454131)

[Figure 4.6: Mobile Screenshot for Search 56](#_Toc171454132)

[Figure 4.7: Code Snippet for Buyer Side Negotiation 56](#_Toc171454133)

[Figure 4.8: Mobile Screenshot for Buyer Negotiation 57](#_Toc171454134)

[Figure 4.9: Code Snippet for search Filters 57](#_Toc171454135)

[Figure 4.10: Mobile Screenshot for Search Filters 58](#_Toc171454136)

[Figure 4.11: Code Snippet for Review 58](#_Toc171454137)

[Figure 4.12: Mobile Screenshot for Reviews 59](#_Toc171454138)

[Figure 4.13: Code Snippet for Image Quality Detector 59](#_Toc171454139)

[Figure 4.14: Mobile Screenshot for Image Quality Detector 60](#_Toc171454140)

[Figure 4.15: Code Snippet for Product Listing Form 60](#_Toc171454141)

[Figure 4.16: Mobile Screenshot for Product Listing Form 61](#_Toc171454142)

# Introduction to the Project

In Pakistan, economic challenges are tightening wallets, making it difficult for many to afford new items. Simultaneously, our environment faces increasing pressure as we generate significant waste from discarded goods. Often, people purchase items they rarely or never use, or hold onto things that don't fit, rather than addressing the issue. This contributes to a cycle of wastefulness. Additionally, consumers are frequently misled into paying exorbitant prices, especially in the fashion industry, where brand names overshadow the value of the actual product. Some existing platforms complicate the process of buying and selling pre-used items, exacerbating these issues. As a result, we're caught in a cycle of financial strain, excessive waste, and stagnant progress.

Our web application aims to break this cycle by facilitating the buying and selling of affordable second-hand items. This project modernizes the cultural tradition of passing on items to family, friends, or those in need, offering a convenient platform for sharing. Our goal is to create a community where helping others and reducing waste go hand in hand. Sellers can offer their items at reasonable prices, making them accessible to a wider audience.

Moreover, our platform provides an option for sellers to donate items to rep utable charities. This initiative not only addresses financial strain and excessive consumption but also promotes a more sustainable future for our country.

## Existing situation and motivation for the project

Thrift stores mostly in Pakistan are facing a big challenge in reaching the people they want to help, and this makes it tough for customers to find what they're looking for. These online stores often show up on social media like Facebook and Instagram, but there's a problem. Many of them focus only on selling clothes and forget about making sure the things they sell are good quality. The online platforms where these stores are found also have issues. They don't always have trusted sellers. Imagine wanting to buy something important, but you're not sure if the person selling it is reliable. An online store on Facebook named as Thrift store works in the same way causing hassle for the users.[1]

A lot of these platforms seem like crowded markets where everyone is competing for your attention. It's confused and unclear. Most stores don’t allow direct sales and the ones that do are not well-managed. YBMB or Mega Thrift store are an example for such a store as it does not give the option for users to sell directly on the platform.[2]

There is a significant issue with scams. It's risky when platforms don't thoroughly check the items that users are selling. It's like letting scammers who want to take advantage of others in. There must be a better way for people to interact with thrift shops and for people to find what they need without being concerned about cons or low-quality items. OLX is an online store that allows selling and buying a variety of used or new items but it is also well-known for its scams and C2C model.[3]

## Literature review

Thrift stores have been on the rise amongst the youth who are looking for items that are affordable. There are many people who don’t want to carry the enormous amount of clothing’s or valuables. Hence, there requires a market which allows the system of thrift. Although, thrift stores are on the rise, the condition of the stores and the market as a whole has not been advancing much. Some studies have been conducted regarding objectives like the various factors promoting the use of thrift store products.[4]

According to the survey some various factors promoting the use of thrift stores are some of the following and the table shows the number of respondents regarding the statements [4]:

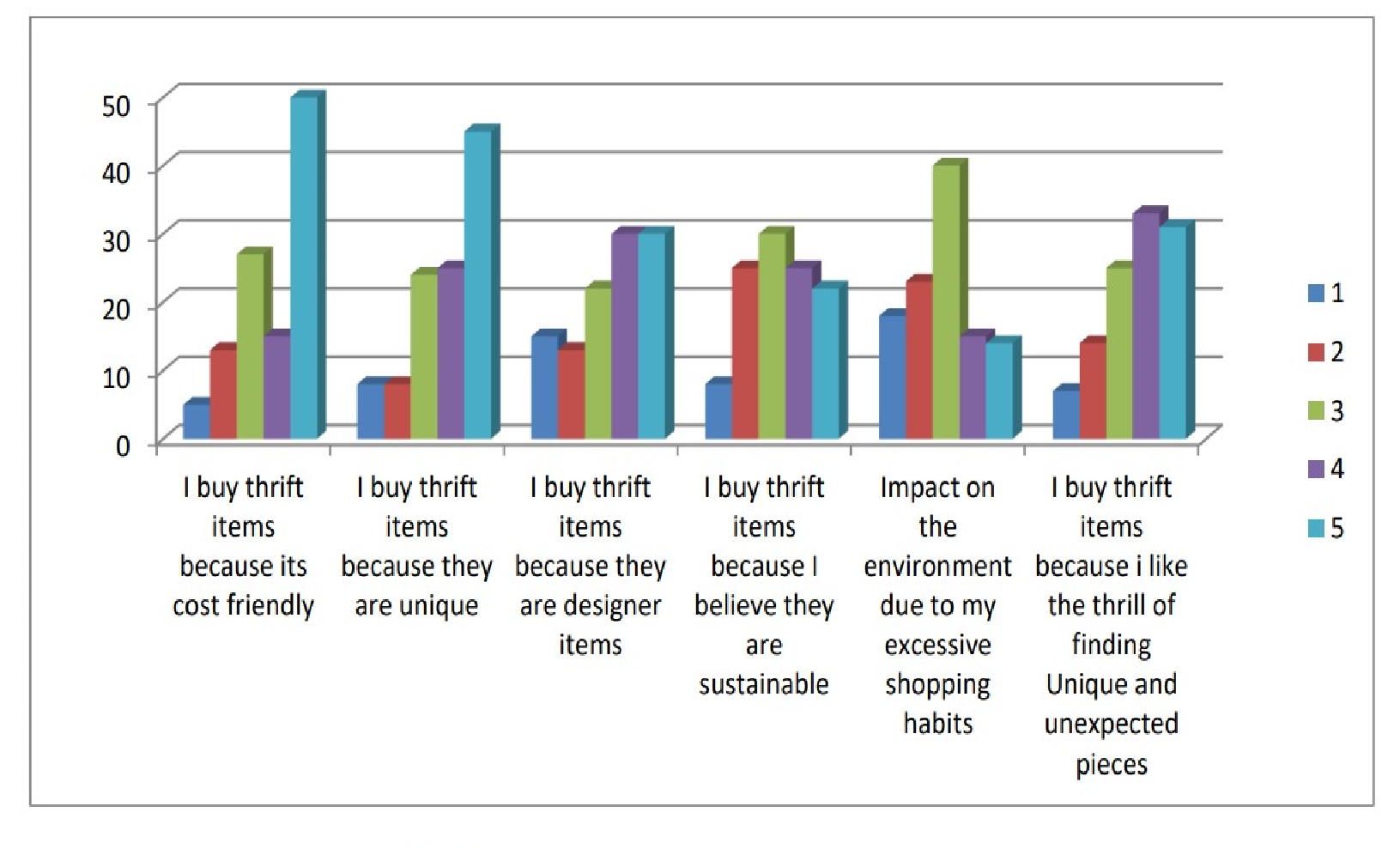


Figure 1.1: Factors promoting thrift stores

Some findings from the research include [4]:

1. The study revealed that people buy thrift items mostly because of budget friendly price and uniqueness.
2. Through this study it has been found that the sellers have lack of funds for expansion and upgrading the stores.
3. From the study it has been found that most of customers of thrift stores are between the age group 15-25 years.
4. The study revealed that most of the thrift sellers started thrift stores for income.
5. The study revealed that sales of thrift products have gone up after the pandemic.
6. Through this Study it has been found that Thrift stores help in promoting Green Consumerism by reducing the waste produced by clothes due to fast fashion.

eBay:

eBay is an online shopping platform that connects buyers and sellers from around the world. It's like a huge virtual marketplace where users can find almost anything they’re looking for. It allows users to buy or sell a wide range of items from brandnew products to pre-owned and vintage items etc. The eBay auction system is one of its unique features.

Sellers can set up auctions for their items, and buyers can bid on them. The highest bidder wins the item. Alternatively, some items have a "Buy It Now" option, where you can purchase the item immediately at a set price. There are many different categories on eBay. Fashion, electronics, home decor, collectibles, and other items are all available. It is a flexible platform for buyers and sellers with a range of interests because of its diversity.

Globally, eBay connects people. Items can be listed for sale by sellers, and buyers can make purchases almost anywhere. Both buyers and sellers on eBay can leave ratings and reviews after a transaction. This feedback system helps build trust within the community. Sellers with good ratings are often considered more reliable. eBay aligns with the concept of sustainable and economic shopping. It provides a virtual space where individuals can explore, bid on, and purchase pre-owned items, echoing the essence of thrift store culture in a digital environment.[5]

Offer Up:

Offer Up is a user-friendly mobile app designed to connect people locally for buying and selling items. It's like having a virtual garage sale right in your pocket. It concentrates on bringing people together locally, as opposed to some online marketplaces that are global in scope. Because of this local flavor, users can buy goods made by locals. Sellers can quickly list items for sale by taking pictures with their phone, adding a description, and setting a price. It's a straightforward process that doesn't require any advanced tech skills.

On Offer Up, users can find everything from furniture and electronics to clothes and accessories. Buyers and sellers can communicate directly through the app. If user is interested in an item, they can ask questions or negotiate the price before making a purchase. It has safety measures in place to make in-person transactions more secure. It provides guidelines for meeting in public spaces and encourages users to be cautious and aware during exchanges.

It also includes a rating system where buyers and sellers can leave feedback based on their experiences. Offer Up promotes the idea of reusing items, contributing to a more sustainable and eco-friendlier lifestyle. Additionally, buyers can often find items at lower prices compared to buying new, aligning with the principles often associated with thrift stores. It represents a modern and localized approach to second-hand commerce, where people within a community can easily exchange items, fostering a sense of shared economy and sustainability.[6]

Limitations:

1. User Interface and Experience: Both platforms could benefit from more intuitive designs that enhance user interaction and ease of use.
2. Localized Features: While eBay focuses globally, it might lack specific local features that could enhance user experiences in particular regions. Offer Up, although local, could expand to include more diverse regional options.
3. Advanced Search Filters: There is a need for more sophisticated search capabilities to help users find specific items more efficiently.
4. Administrative Panels: Neither platform has a robust admin panel for sellers to manage listings and transactions comprehensively.
5. Real-Time Customer Support: Limited real-time support can hinder immediate resolution of issues that arise during transactions.
6. Personalization: Both platforms could improve by offering more personalized user experiences based on shopping habits and preferences.
7. Security Features: Enhanced security measures are needed, especially for in-person transactions facilitated by Offer Up.
8. Sustainability Practices: While both platforms promote second-hand purchases, additional features or programs to further support sustainability could be implemented.

## Additional research

According to some more research, the economy is in poor shape, and a lot of people are looking for affordable options. Giving things to family members or making donations is common in our culture. Both the younger and older generations continue to practice this tradition. Some people do it online, but there is a growing trend of doing it through personal networks. Our platform aims to better organize this process.

The way we handle buying and selling stuff also affects nature. The more we can reuse and share, the less new stuff we need to make, which is good for the planet. Online thrift stores are becoming more popular since COVID. Tech initiatives to promote similar platforms abroad are common. Talking to people, we found out that some folks hesitate to buy used things in person as well as online.

On the other hand, buying new things is a struggle. When we hesitate to buy used things, we end up making more new things, which takes more resources and energy. Here’s why it's a big deal: Making a simple cotton shirt needs a lot of water—about 400 gallons for just one! This water use harms the environment. Apart from this, People also think such stores saves them money and also consider it a green alternative, which is definitely true.[7]

# Requirement Analysis

The Analysis Document serves as a roadmap for our project, which aims to create an eco-friendly, cost effective and socially responsible way to find unique and affordable items. We draw attention to Pakistan's economic hardships and the problems caused by excessive waste in the environment. The purpose of the Analysis Document is to review the current system and highlight problems with online platforms, such as quality control issues, a lack of direct selling options, and susceptibility to scams.

## Requirement gathering and fact finding

For effective requirement gathering and fact finding, a comprehensive approach involving multiple techniques was employed. The goal was to understand the needs, preferences, and challenges of potential users, ensuring the development of a web application aligned with their expectations.

Data Collection Techniques:

1. Discussions: Some discussions were conducted with a set of predetermined questions to gather specific information. It allowed open-ended discussions, encouraging participants to express their thoughts freely about what issues they face. Interviews were conducted with users, including both buyers and sellers, to capture diverse perspectives.
2. Observations: To identify user interactions and pain points, in-person observations of online platforms were made. This method assisted in pinpointing particular problems users encountered with the current systems.
3. Literature Search: A review was conducted of previous research studies and articles concerning thrift stores, online marketplaces, and consumer behavior. This review of the literature provided information about user behavior and global trends that may be important to the project.
4. Online Survey: A larger audience around 35-40 users were reached through online surveys, which offered participants a quick and easy way to share their opinions. Questions about preferences, difficulties, and expectations with regard to purchasing and selling used goods were asked in surveys. Some of the results from the responses were as follows (regarding some specific questions):
   * 1. What types of items would you prefer to buy second-hand?

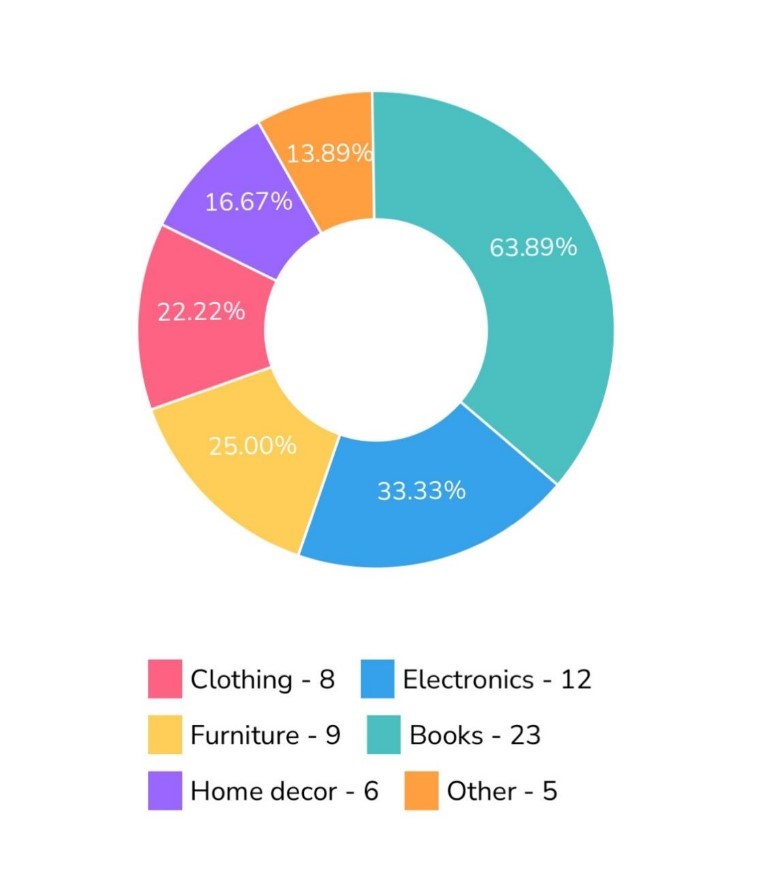


Figure 2.1: Preference of Items

* + 1. What features do you find most important in an online platform for second-hand items?

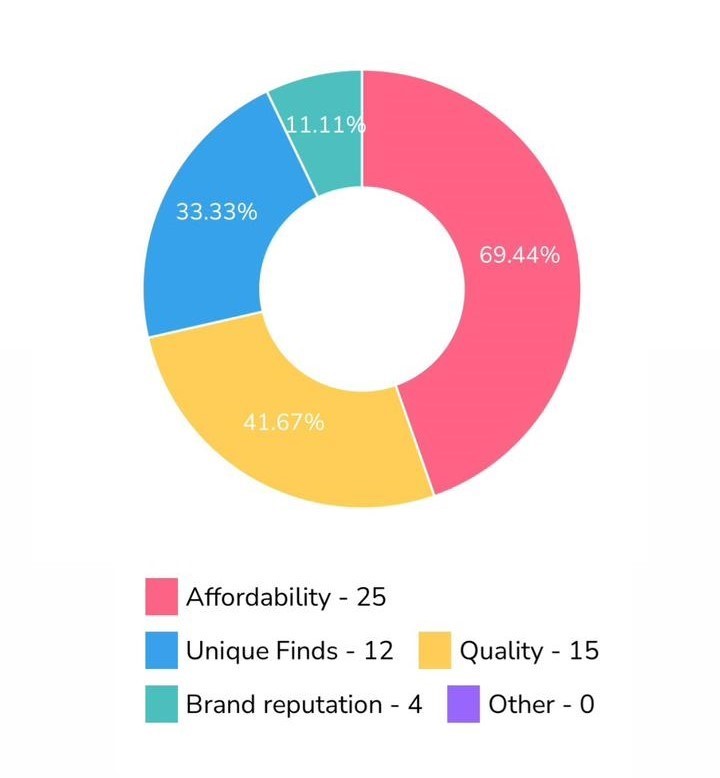


Figure 2.2: Preference of Features

Out of 36 participants, 23 indicated that they preferred to buy books from Thrift stores. Secondly, when asked about their priorities in buying items, 25 respondents highlighted affordability as an important consideration. This highlights a noteworthy trend in which customers prioritize cost-effectiveness when making overall buying decisions and prefer thrift stores for book purchases.

Link to the survey form:

<https://surveyheart.com/form/653a0d1e7e4a390831216110>

## Use case analysis

The use case analysis for ThriftN encompasses the interactions between various user roles and the system. It details how buyers, sellers, admins, NGOs, and delivery personnel engage with the platform to accomplish their specific goals. This section provides a comprehensive view of user stories and a use case diagram, illustrating key functionalities and ensuring that all user requirements are addressed effectively. The analysis aims to highlight the essential tasks each user role can perform, thereby facilitating a user-centered design approach for the ThriftN platform.

### User roles

1. Sellers: Users who list and sell products on the platform.
2. Buyers: Users who search for and purchase products on the platform.
3. Owners/Admin: System administrators responsible for managing categories, approvals, and policies.
4. NGO (Non-Governmental Organization): Organizations receiving donated items for distribution.
5. Delivery Man: Individuals responsible for picking up and delivering products.

### Key user stories

**User Story for a Buyer: Searching for and Buying Products**

As a buyer, I want to search for and purchase products on the platform, so that I can meet my needs efficiently.

1. **Search for Products:** **As a buyer,** I want to use a search bar to find products by name or category, **so that** I can easily locate items of interest.
2. **View Product Listings: As a buyer,** I want to see a list of products matching my search criteria, **so that** I can choose from available options.
3. **Product Details: As a buyer,** I want each product listing to include details like name, description, images, seller information, and pricing, **so that** I can make informed purchase decisions.
4. **Filter Options: As a buyer,** I want to filter products by price range, categories, and trusted sellers, **so that** I can find products that match my preferences.
5. **Purchase Options:** As a buyer, I want to click on a product to view more details and have the options to "Add to Cart," "Buy Now," or "Negotiate," so that I can either save the item for later, purchase it immediately, or try to get a better price.
6. **Immediate Purchase: As a buyer,** I want to select my delivery location, choose a payment method, and confirm my order when I choose "Buy Now," **so that** I can complete my purchase quickly.

**User Story for a Seller: Conveniently sell items:**

As a seller, I want to list my products for sale on the platform, so that I can reach potential buyers easily.

1. Access Selling Features: As a seller**,** I want to click on the "Sell" button when I log into the web app, so that I can start listing my products.
2. Select Product Category:As a seller**,** I want to select the category for my product, so that buyers can find it in the appropriate section.
3. Add Photos: As a seller**,** I want to add photos of my product, so that potential buyers can see what I am selling.
4. Complete Listing Form: As a seller**,** I want to fill in a product listing form with all necessary details, so that my product information is clear and comprehensive.
5. Submit Listing: As a seller**,** I want to click the submit button after completing the form, so that my product gets listed on the platform for buyers to view.

### Use case diagram

The use case diagram below effectively captures the complex interactions within the ThriftN system, highlighting the collaborative dynamics between Buyers, Sellers, Admin, and NGOs. Each of these actors plays a crucial role in ensuring the system operates seamlessly, supporting a diverse range of functionalities that are integral to the user experience.

Buyers and Sellers, in particular, engage with the system daily, contributing to its active and vibrant marketplace. The Admin, meanwhile, ensures that all processes run smoothly, stepping in when necessary to resolve issues and maintain order. NGOs benefit from the system’s integrated donation feature, allowing them to receive contributions efficiently and transparently.

These interactions underscore the system’s ability to cater to various needs while maintaining a user-centric approach. The diagram on the next page illustrates these relationships, providing a clear and concise overview of how each actor contributes to the overall functionality of ThriftN.

The robust nature of these interactions ensures that ThriftN remains a reliable and efficient platform, fostering trust among its users. As the system continues to grow, these foundational relationships will be critical to its ongoing success and adaptability.

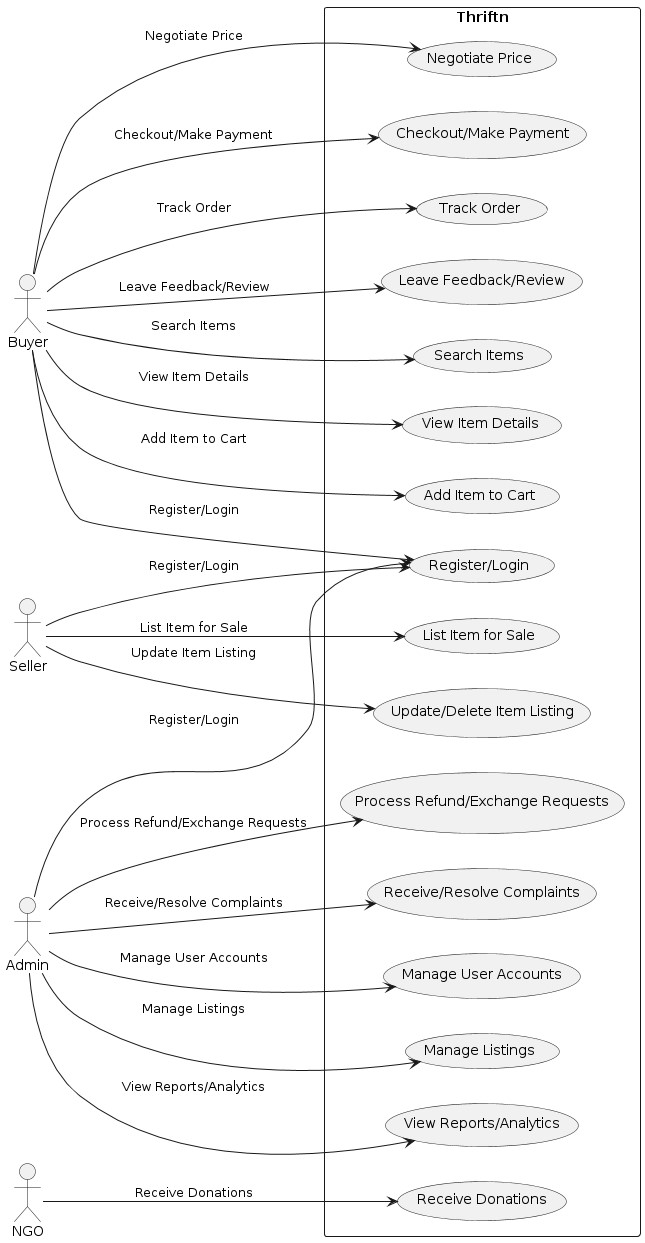


Figure 2.3: Use case diagram

## Functional requirements

## 

Table 2.1: Admin Login

|  |  |
| --- | --- |
| **Req. No.** | **Functional Requirements** |
| **FR1-01** | The system should allow registered admins to log in using their emails and passwords. |
| **FR1-02** | The system should allow registered admins to log out from the account. |

## 

Use case description 01

* + 1. Description: Registered admins can log in using their email and password.
    2. Actor: Admins
    3. Precondition: Admin is not logged in the system.
    4. Postcondition: Admin is logged in and gains access to the admin dashboard.
    5. Details:

1. Admin provide a valid email address and password.
2. The system validates the email format and ensures it is unique.
3. Upon successful validation, the admin is granted access to the admin dashboard.

Use case description 02

* + 1. Description: The system should provide a secure logout option for registered administrators to exit their accounts safely.
    2. Actor: Admins
    3. Precondition: Admin is logged into the admin account.
    4. Postcondition: Admin is logged out of the system.
    5. Details:
    6. A "Logout" button or link is prominently displayed on the admin dashboard or in a consistent location across all admin pages.
    7. Clicking the "Logout" option securely terminates the admin's session and redirects them to the login page or homepage.
    8. The system ensures that after logout, no residual data or session details are accessible without re-authentication.

Table 2.2: User Login

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR2-01** | The system should allow users to create accounts by using their email and passwords. |
| **FR2-02** | The system should allow users to log in using their email and password. |
| **FR2-03** | The system should allow users to log out from the account. |

## 

Use case description 01

* + 1. Description: Registered users can log in using their email and password.
    2. Actor: Sellers, Buyers, Owners, Delivery man
    3. Precondition: User is not logged in.
    4. Postcondition: User is logged in and gains access to their profile.
    5. Details:

1. Admin provide a valid email address and password.
2. The system validates the email format and ensures it is unique.
3. Upon successful validation, the admin is granted access to the admin dashboard.

Use case description 02

* + 1. Description: The system should allow users to log in using their registered email and password, providing secure access to their accounts.
    2. Actor: Users
    3. Precondition: User has a registered account with a verified email and password.
    4. Postcondition: User is logged into their account.
    5. Details:
       1. The login page includes fields for entering email and password.
       2. Users enter their credentials and submit them via a "Login" button.
       3. The system verifies the credentials against stored data; if correct, access is granted to the user's account.
       4. If credentials are incorrect, an error message is displayed prompting re-entry or password recovery.

Use case description 03

* + 1. Description: The system should provide a secure logout option for users to exit their accounts safely.
    2. Actor: Users
    3. Precondition: User is logged into their account.
    4. Postcondition: User is logged out of the system.
    5. Details:
    6. A "Logout" button or link is available on user dashboard or in a consistent location across all user-accessible pages.
    7. Clicking the "Logout" option securely terminates the user's session and redirects them to the login page or homepage.
    8. The system ensures that after logout, no residual data or session details are accessible without re-authentication.

Table 2.3: Seller Side

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR3-01** | Sellers can list product for sale. |
| **FR3-02** | Seller must be able to upload photos of their product during the listing process. |
| **FR3-03** | Seller must complete the product listing form when listing the product for sale. |
| **FR3-04** | The system should allow seller to also buy a product. |

## 

Use case description 01

* + 1. Description: Sellers should be able to list products for sale on the platform by clicking the sell now button.
    2. Actor: Sellers
    3. Precondition: Seller is logged in.
    4. Postcondition: The product is listed for sale.
    5. Details:

1. Sellers fills the product listing form.

Use case description 02

1. Description: Sellers can upload photos of their products during the listing process, with tips to upload photos including images of both the front and back of the product. Sellers can also add more images to showcase the item from various angles.
2. Actor: Sellers
3. Precondition: The seller is in the process of listing a product for sale.
4. Postcondition: The photos are uploaded.
5. Details:
6. Sellers can upload multiple images per product to showcase the item from various angles.

Use case description 03

* + 1. Description: Sellers must complete a product listing form when adding a new product for sale on the platform.
    2. Actor: Sellers
    3. Precondition: Seller is listing a product for sale.
    4. Postcondition: Sellers have successfully completed and submitted the product listing form and product is listed on the platform.
    5. Details:
    6. Sellers are prompted to select a category for their product
    7. Sellers need to Add photo of the product.
    8. Sellers proceed to fill in the following product details:
    9. Product Title: A descriptive title for the product.
    10. Price: The price at which the product will be listed.
    11. Product Description(optional): A brief description of the product
    12. After reviewing the form, the seller submits it.

Use case description 04

* + 1. Description: The homepage should have a section displaying new items available in the section for the specified duration defined by the admin.
    2. Actor: Buyers, Sellers and Admin
    3. Precondition: Users visit the homepage of the platform.
    4. Postcondition: The new items section is visible on the homepage.
    5. Details:

1. The New Items section would have products that have been recently listed by sellers and remain there for a specific duration.

Table 2.4: Categories

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR4-01** | The system must have predefined categories. |

## 

## 

Use case description 01

* + 1. Description: The system should come preloaded with a set of predefined product categories such as clothes, books, furniture, electronics, etc.
    2. Actor: System
    3. Precondition: The web app is initialized or set up.
    4. Postcondition: Predefined product categories are available for use in the web app.
    5. Details:
    6. The system comes with a predefined set of product categories.
    7. Categories include but are not limited to clothes, books, furniture, electronics, etc.
    8. Users can interact with the categories, such as selecting a category for listing products or searching for items within a specific category.

Table 2.5: Homepage and Content Management

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR5-01** | The homepage should include a filter for trusted sellers. |
| **FR5-02** | The homepage should include a section for newly added items. |
| **FR5-03** | The homepage should include a section for All items. |
| **FR5-04** | The homepage should include a section for recommended items |

## 

Use case description 01

* + 1. Description: The homepage should prominently feature a section dedicated to showcasing trusted sellers on the platform. Trusted sellers are designated based on achieving the following criteria: accumulating positive reviews and ratings, successfully completing a minimum of 10 sales, and unwavering adherence to platform policies.
    2. Actor: Buyers, Sellers and Admin
    3. Precondition: Users visit the homepage of the platform.
    4. Postcondition: The trusted seller’s section is visible on the homepage.
    5. Details:
    6. The homepage prominently displays a section that highlights trusted sellers, enabling users to easily identify reliable sellers for their purchases.
    7. Trusted sellers are recognized for consistently providing excellent products and service, maintaining high ratings, receiving positive reviews, and having a history of successful transactions.
    8. The admin has access to the Trusted Seller management section within the admin dashboard.
    9. Admin establishes specific criteria for achieving the trusted seller status, including prerequisites: completing a minimum of 10 successful transactions, maintaining positive reviews and ratings, and demonstrating unwavering adherence to platform policies.
    10. Sellers who meet or exceed the defined criteria are automatically designated as trusted sellers
    11. Users can readily identify trusted sellers through badges displayed on their profiles and product listings.

Use case description 02

* + 1. Description: The homepage should display a section suggesting items recommended for users based on factors, including previous searches, purchases, product views, preferred product categories, seasonal trends, and complementary products.
    2. Actor: Buyers
    3. Precondition: Users visit the homepage of the platform.
    4. Postcondition: The recommended items section is visible on the homepage.
    5. Details:
    6. The system gathers a comprehensive dataset of buyer interactions.
    7. This dataset encompasses product views, searches, purchases etc. Additionally, it records each buyer’s preferred product categories.
    8. Utilizing advanced machine learning algorithms and collaborative filtering techniques, the system analyzes the collected data. These algorithms take into account a buyer’s historical behavior,
    9. identifying patterns and preferences. For instance, if a buyer frequently searches for and buys clothes, the system learns that clothes are a preferred category.
    10. The recommendation system incorporates seasonal trends to reflect time-based patterns and preferences. For instance, during the holiday season, it considers the heightened demand for festive items and adjusts recommendations accordingly.
    11. The system identifies complementary products by evaluating user interactions. If a buyer adds a shirt to their cart, the system may suggest related accessories such as pants, jeans, or skirts based on the purchase histories and patterns of other users.
    12. Buyers receive tailored product suggestions on the homepage.

Table 2.6: Policy

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR6-01** | The system should have a policy page on the homepage |

## 

## 

Use case description 01

* + 1. Description: The system should provide a policy page accessible from the homepage, containing important platform operational guidelines including terms of use, privacy policy, and return and refund policies.
    2. Actor: System, User
    3. Precondition: Web app is initialized and operational.
    4. Postcondition: Policy page is accessible from the homepage.
    5. Details:
    6. The homepage features a clearly visible link to "Policies".
    7. Clicking the link navigates users to a dedicated policy page with sections on Privacy Policy, Terms of Use, Return and Restricted Items Policy.

Table 2.7: About Us

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR7-01** | The system should have an about us page on the home page. |

Use case description 01

* + 1. Description: The system should feature an "About Us" page accessible directly from the homepage, providing information about the platform's mission, history, team, and objectives.
    2. Actor: System, Users
    3. Precondition: Web app is initialized and operational.
    4. Postcondition: "About Us" page is accessible from the homepage.
    5. Details:
    6. The homepage includes a prominently displayed link or button labeled "About Us".
    7. Clicking this link directs users to the "About Us" page.
    8. The page contains sections detailing the platform's mission, history, team, and core objectives.
    9. Access to this page does not require user login, facilitating easy and immediate access for all visitors.

Table 2.8: Customer Services

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR8-01** | The system should have customer service information which includes helpline and email address of the organization on the homepage. |

## 

Use case description 01

* + 1. Description: The system should feature customer service information on the homepage, including the helpline number and email address of the organization.
    2. Actor: System, Users
    3. Precondition: Web app is initialized and operational.
    4. Postcondition: Customer service information is prominently displayed on the homepage.
    5. Details:
    6. The homepage includes visible sections for both the helpline number and the organization's email address.
    7. This information is displayed in an easily accessible location on the homepage, ensuring immediate visibility.
    8. Access to view this contact information does not require user login, allowing all visitors immediate access.

Table 2.9: Product Search Viewing and Browsing

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR9-01** | The system should provide search bar for buyers to search for products. |
| **FR9-02** | Buyers should be able to see detailed product information after clicking a product. |

## 

Use case description 01

* + 1. Description: The system should provide a search bar on the homepage, allowing buyers to search for products by keywords.
    2. Actor: Buyers
    3. Precondition: Web app is initialized and operational.
    4. Postcondition: Search bar is available on the homepage for product searches.
    5. Details:
    6. A search bar is prominently displayed on the homepage.
    7. Buyers can enter keywords into the search bar to find products.
    8. Search results are displayed based on the keywords entered, showing relevant products.

Use case description 02

* + 1. Description: Buyers should be able to view detailed information about a product by clicking on the product from the search results or product lists.
    2. Actor: Buyers
    3. Precondition: Buyer has accessed the search results or product lists.
    4. Postcondition: Detailed product information is displayed when a product is clicked.
    5. Details:
    6. Each product in search results or lists features a clickable area or button.
    7. Clicking on a product navigates to a detailed product page.
    8. The product page includes information such as price, description, photos, seller details, and availability.

Table 2.10: Filters

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR10-01** | The system should allow buyers to use’ Price' filter to refine search results. |
| **FR10-02** | The system should allow buyers to use’ Categories' filter to refine search results. |

## 

Use case description 01

* + 1. Description: The system should allow buyers to use a 'Price' filter to refine search results on the platform, enabling them to select a price range for displayed products.
    2. Actor: Buyers
    3. Precondition: Web app is initialized and operational, and search results are displayed.
    4. Postcondition: Buyers can refine search results using the 'Price' filter.
    5. Details:
    6. A 'Price' filter option is available on the page where search results are displayed.
    7. Buyers can select or enter a specific price range (e.g., minimum and maximum price).

Use case description 02

* + 1. Description: The system should allow buyers to use a 'Categories' filter to refine search results, enabling them to select specific product categories to view.
    2. Actor: Buyers
    3. Precondition: Web app is initialized and operational, and search results are displayed.
    4. Postcondition: Buyers can refine search results using the 'Categories' filter.
    5. Details:
    6. A 'Categories' filter option is available on the page where search results are displayed.
    7. Buyers can select one or multiple categories from the predefined list (e.g., clothes, books)

Table 2.11: Shopping and Transaction

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR11-01** | Buyers should be able to add products to a shopping cart for purchase. |
| **FR11-02** | Buyers can initiate buying process, enter delivery details and select payment method. |

Use case description 01

* + 1. Description: The system should allow buyers to add products to a virtual shopping cart for later purchase, facilitating a seamless shopping experience.
    2. Actor: Buyers
    3. Precondition: Web app is initialized and operational, and the buyer is viewing product details.
    4. Postcondition: Products selected by the buyer are added to the shopping cart.
    5. Details:
    6. Each product page includes an "Add to Cart" button.
    7. Clicking the "Add to Cart" button adds the product to the buyer's shopping cart.
    8. The shopping cart updates in real time, reflecting the addition of the product.

Use case description 02

* + 1. Description: Buyers should be able to initiate the buying process from their shopping cart, which includes entering delivery details and selecting a payment method.
    2. Actor: Buyers
    3. Precondition: Buyer has products in their shopping cart.
    4. Postcondition: Buyer has entered delivery details, selected a payment method, and is ready to finalize the purchase.
    5. Details:
    6. The shopping cart includes an "Proceed to Checkout" button.
    7. Clicking this button takes the buyer to a checkout page where they can enter delivery details (such as address and contact information).
    8. The checkout page includes options for selecting a payment method (e.g., cash on delivery).
    9. The buyer can review all details before finalizing the purchase.

Table 2.12: Order Tracking

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR12-01** | The system should provide an order tracking feature that allows users to monitor the status of their order. |

Use case description 01

* + 1. Description: The system should provide an order tracking feature that allows users to monitor the status of their orders, from processing to delivery.
    2. Actor: Buyers
    3. Precondition: Buyer has placed an order.
    4. Postcondition: Buyer can access real-time status updates of their order.
    5. Details:
    6. The system provides a tracking option in the user's account page or order confirmation page.
    7. Users can click on the tracking link to view detailed status updates of their order, including processing, shipping, and delivery stages.
    8. The tracking feature updates automatically as the order status changes.

Table 2.13: Negotiation Option

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR13-01** | The system should allow buyers to negotiate the price and seller can accept or reject. |
| **FR13-02** | The system should allow buyers to negotiate the price to seller more than once. |

## 

Use case description 01

* + 1. Description: The system should allow buyers to negotiate the price of a product with the seller, who can then accept or reject the proposed price.
    2. Actor: Buyers, Sellers
    3. Precondition: Buyer is viewing a product.
    4. Postcondition: Buyer has submitted a price offer; seller can accept or reject it.
    5. Details:
    6. Products eligible for negotiation display a "Negotiate Price" option.
    7. Buyers can submit a price offer through a specified interface.
    8. Sellers receive notifications of price offers and can accept or reject them within a designated timeframe.

Use case description 02

* + 1. Description: The system should allow buyers to negotiate the price with the seller more than once, depending on the seller’s response and negotiation terms.
    2. Actor: Buyers, Sellers
    3. Precondition: Seller has responded to a previous negotiation attempt.
    4. Postcondition: Buyers can re-negotiate based on the seller's feedback.
    5. Details:
    6. If a seller rejects an initial offer, the buyer can submit another offer.
    7. The system should allow multiple rounds of negotiation until an agreement is reached or the negotiation window expires.

Table 2.14: Donation

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR14-01** | The system should allow users to donate items to an NGO that is linked to the organization. |

## 

## 

Use case description 01

* + 1. Description: The system should allow users to donate items directly to an NGO that is partnered with the organization, facilitating charitable contributions.
    2. Actor: Users
    3. Precondition: User is logged in and has items to donate.
    4. Postcondition: Items have been marked for donation to the NGO.
    5. Details:
    6. On the product listing or user dashboard, there is an option to "Donate".
    7. Users can select items and email to the given admin email.

Table 2.15: Policy

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR15-01** | The system should have a return policy section to define it. |

Use case description 01

* + 1. Description: The system should include a dedicated return policy page that clearly defines the terms and conditions for returning products purchased through the platform.
    2. Actor: System, Users
    3. Precondition: Web app is initialized and operational.
    4. Postcondition: Return policy page is accessible and provides comprehensive information on return procedures.
    5. Details:
    6. The homepage includes a clearly visible link or section labeled "Return Policy".
    7. Clicking on Policy link navigates users to the return policy section.
    8. The page details the conditions under which items can be returned, the timeframe for returns, the refund process, and any exceptions to the standard policy.
    9. This information is presented in a clear and user-friendly format to ensure users understand their rights and obligations.

Table 2.16: 24/7 Chat Bot

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR16-01** | A 24/7 Chat Bot should be available to assist users with questions and provide support. |

## 

## 

Use case description 01

* + 1. Description: The system should provide a 24/7 chatbot accessible from the website to assist users with questions and provide support at any time.
    2. Actor: Users
    3. Precondition: Web app is initialized and operational.
    4. Postcondition: Chatbot is available to assist users on the platform.
    5. Details:
    6. The chatbot is accessible via a chat interface on the homepage.
    7. Users can interact with the chatbot by typing questions; the chatbot responds in real-time.
    8. The chatbot can handle queries related to navigation, policies, order status, and troubleshooting.

Table 2.17: Adding Product Ratings and Reviews

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR17-01** | The system should allow buyers to only rate and review the product after they have purchased it. |

## 

Use case description 01

* + 1. Description: The system should allow buyers to write reviews for products only after they have made a purchase.
    2. Actor: Buyers
    3. Precondition: Buyer has completed a purchase.
    4. Postcondition: Buyer is able to submit a rate and review for the purchased product.
    5. Details:
    6. The system checks the purchase history before allowing access to the review submission page.
    7. A "Rate and Write a Review" option appears on the product page or in the purchase history after the sale is confirmed.
    8. Buyers can submit ratings and reviews that are then displayed on the product page for reference by future buyers.

Table 2.18: Sale and Product Statistics

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR18-01** | The system should allow admin to check the sale and product statistics on admin account. |

## 

Use case description 01

* + 1. Description: The system should allow administrators to access sales and product statistics through their admin accounts.
    2. Actor: Admin
    3. Precondition: Admin is logged into the admin account.
    4. Postcondition: Admin can view and analyze sales and product statistics.
    5. Details:
    6. The admin dashboard includes an option for "Sales Statistics and Product Statistics".
    7. The feature provides data on sales volume, revenue, most popular products, and trends over selectable time periods.
    8. This information helps in strategic planning and performance assessment.

Table 2.19: List an Item (Admin)

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR19-01** | The system should allow admin to list an item. |

## 

Use case description 01

* + 1. Description: The system should enable administrators to list items on the platform, facilitating inventory management and special listings.
    2. Actor: Admin
    3. Precondition: Admin is logged into the admin account.
    4. Postcondition: Item is listed on the platform by the admin.
    5. Details:
    6. The admin dashboard includes an option to "List an Item".
    7. Admins can enter item details such as name, description, price, and category, and upload images.
    8. Upon submission, the item is immediately listed and available for viewing and purchase by users.

Table 2.20: Image Quality Detector

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR20-01** | The system should allow user to check the quality of image after uploading. (The image quality detector shows percentage of quality after photo is uploaded before submitting the product listing form) |

Use case description 01

* + 1. Description: Users should be able to check the quality of images they upload during the product listing process, with a system-provided quality percentage shown after each upload.
    2. Actor: Sellers
    3. Precondition: Seller is logged in and in the process of listing a product for sale.
    4. Postcondition: The quality of the uploaded image is assessed, and a quality percentage is displayed.
    5. Details:
       1. Seller navigates to the product listing form on the platform.
       2. Seller uploads one or more images of the product.
       3. The system processes each uploaded image to assess its quality.
       4. A quality percentage is displayed next to each image immediately after processing.
       5. Sellers can decide to keep the image, replace it, or adjust the quality based on the feedback provided.

Table 2.21: Disable User Account

|  |  |
| --- | --- |
| **Req. No** | **Functional Requirements** |
| **FR21-01** | The system should allow an admin to disable a user's account by clicking the "Disable" button in the admin panel. |

Use case description 01

1. Description: Admins should be able to disable user accounts by clicking the "Disable" button in the admin panel.
2. Actor: Admin
3. Precondition: Admin is logged in and has navigated to the user management section of the admin panel.
4. Postcondition: The selected user account is disabled, and the user is notified of the account status change.
5. Details:
   * 1. Admin logs into the admin panel.
     2. Admin navigates to the user management section.
     3. Admin locates the user account to be disabled.
     4. Admin clicks the "Disable" button next to the selected user account.
     5. The system processes the request and disables the user account.
     6. A confirmation message is displayed to the admin.

## Non-functional requirements

Table 2.22: System Monitoring

|  |  |
| --- | --- |
| **Req. No** | **Non-Functional Requirements** |
| **NFR1-01** | Implement real-time system monitoring and altering to identify and address issues promptly. |

Table 2.23: Browser Compatibility

|  |  |
| --- | --- |
| **Req. No** | **Non-Functional Requirements** |
| **NFR2-01** | The system should work seamlessly on various web browsers to accommodate a broad user base. |

Table 2.24: Accessibility

|  |  |
| --- | --- |
| **Req. No** | **Non-Functional Requirements** |
| **NFR3-01** | Make the platform accessible to user by following accessibility standards. |

Table 2.25: Regulatory Compliance

|  |  |
| --- | --- |
| **Req. No** | **Non-Functional Requirements** |
| **NFR4-01** | Ensure compliance with relevant legal and industry specific regulations such as customer protection laws. |

Table 2.26: Data Backup and Recovery

|  |  |
| --- | --- |
| **Req. No** | **Non-Functional Requirements** |
| **NFR5-01** | Regularly backup data and establish data recovery process in case of failures. |

Table 2.27: Transaction Speed

|  |  |
| --- | --- |
| **Req. No** | **Non-Functional Requirements** |
| **NFR6-01** | The system should process transactions including order placements and payments without delays. |

Table 2.28: System Uptime

|  |  |
| --- | --- |
| **Req. No** | **Non-Functional Requirements** |
| **NFR7-01** | Aim for high availability with minimal downtime for maintenance or unexpected issues. |

Table 2.29: Payment Security

|  |  |
| --- | --- |
| **Req. No** | **Non-Functional Requirements** |
| **NFR8-01** | Ensure secure payment processing by complying with industry standards and utilizing encryption for financial transactions. |

Table 2.30: Authentication and Authorization Security

|  |  |
| --- | --- |
| **Req. No** | **Non-Functional Requirements** |
| **NFR9-01** | Implement robust user authentication and authorization mechanisms to safeguard against unauthorized access. |

Table 2.31: Page Load Times

|  |  |
| --- | --- |
| **Req. No** | **Non-Functional Requirements** |
| **NFR10-01** | Web Page should load within 3 seconds |
|  | or less to ensure a responsive user experience. |

Table 2.32: Scalability Planning

|  |  |
| --- | --- |
| **Req. No** | **Non-Functional Requirements** |
| **NFR11-01** | The system should be designed to accommodate a growing number of users and products. It should be capable of scaling up with increased demand. |

Table 2.33: User Data Protection

|  |  |
| --- | --- |
| **Req. No** | **Non-Functional Requirements** |
| **NFR12-01** | The system should employ strong data measures to protect user data, including encryption for sensitive information. |

Table 2.34: User-Friendly Interface

|  |  |
| --- | --- |
| **Req. No** | **Non-Functional Requirements** |
| **NFR13-01** | The web app should have an intuitive and user-friendly interface, making it easy for users to navigate and use the platform. |

Table 2.35: Consistency

|  |  |
| --- | --- |
| **Req. No** | **Non-Functional Requirements** |
| **NFR14-01** | The user interface design, layout and navigation should be consistent across all pages to provide a uniform experience. |

# Design

The MERN stack is a set of technologies for building websites: MongoDB (a database), Express.js (handles server requests), React.js (creates user interfaces), and Node.js (runs JavaScript on the server). Together, they help make web development easier.

By choosing the MERN stack, we are aligning our project with modern industry standards, which is crucial in today's fast-paced technology environment. This choice offers us the chance to not only develop a robust, full-stack application but also to gain hands-on experience with tools that are widely used in the field.

Working with the MERN stack allows us to approach development in a more streamlined and cohesive manner. The use of JavaScript across all components of the stack reduces the complexity typically associated with using multiple programming languages, making the development process more efficient. This unified language environment facilitates better collaboration among team members, as everyone can contribute to both front-end and back-end development without needing to switch contexts.

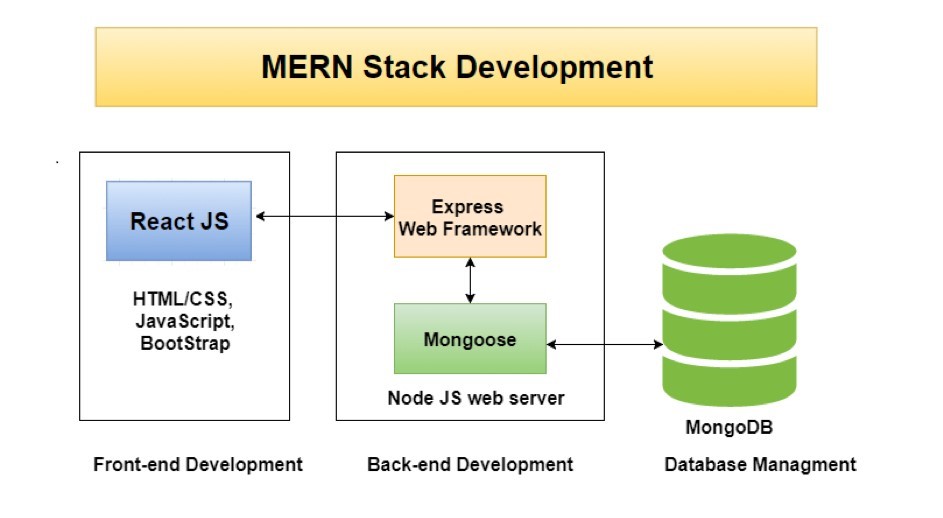


Figure 3.1: Mern Stack Structure

## Architecture

**System Context Diagram:**

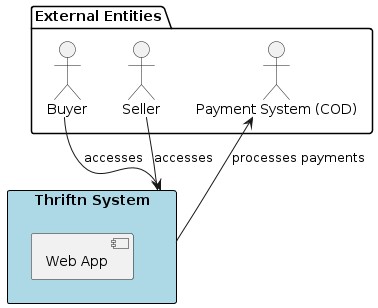


Figure 3.2: System Context Diagram

**Container Diagram:**

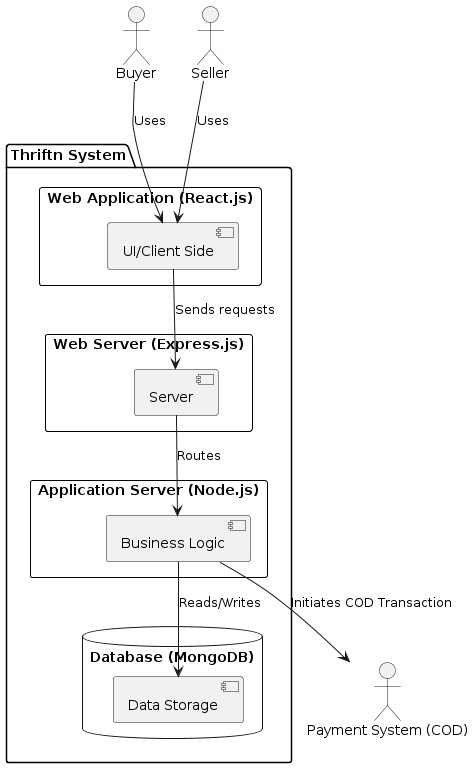


Figure 3.3: Container Diagram

**High Level Architecture Diagram:**

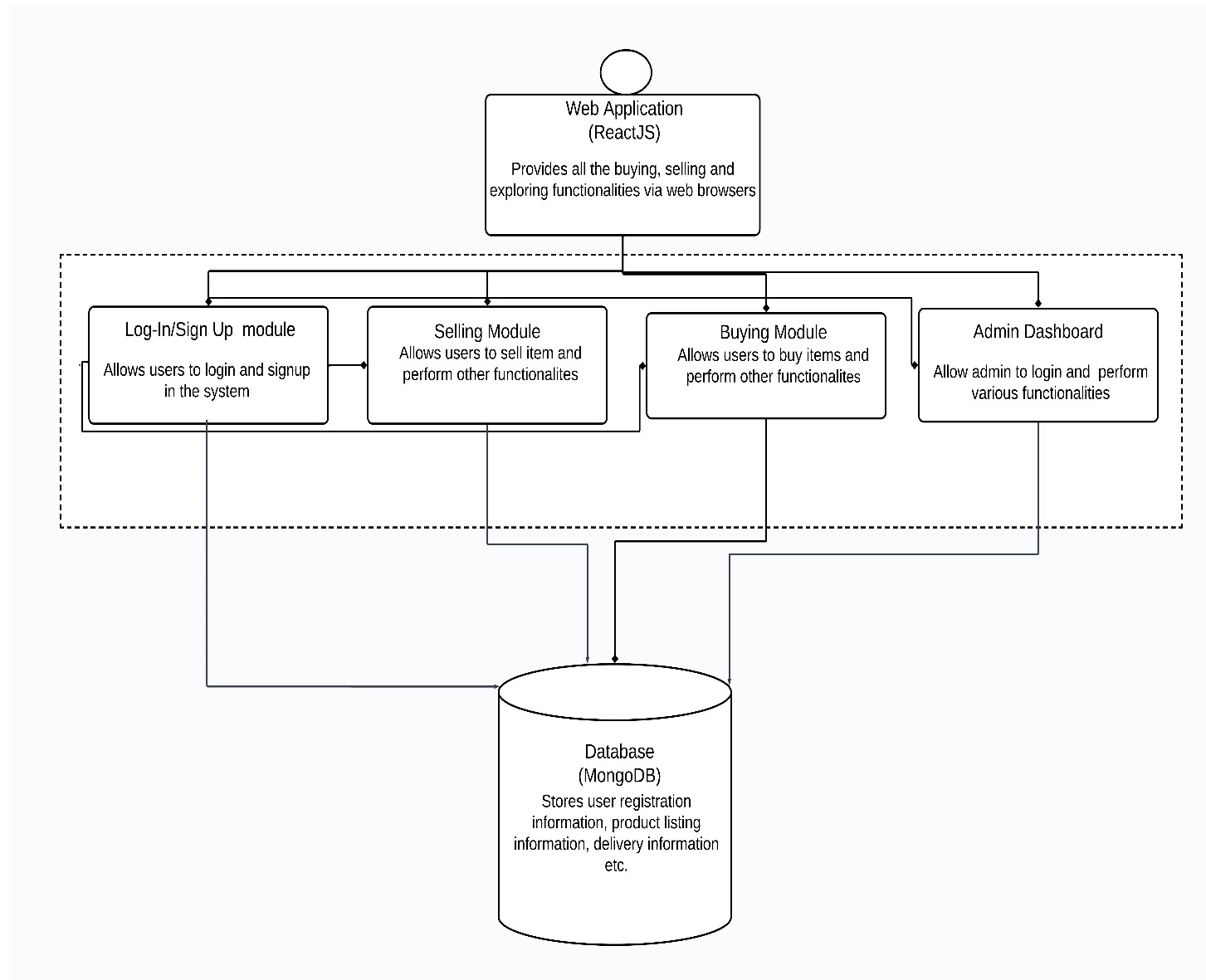


Figure 3.4: High Level Architecture Diagram

## Key sequence diagrams

**Note: User is equal to seller, buyer, owner/admin, NGO**

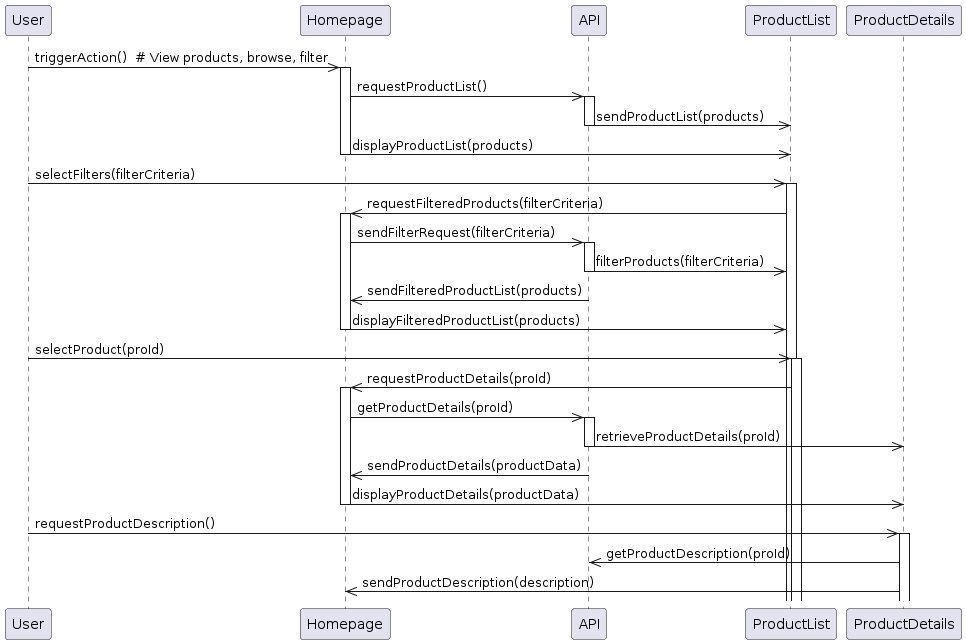


Figure 3.5: Key Sequence diagram

This diagram combines user interactions for product browsing, filtering, viewing details, and adding items to the cart. It incorporates filtering functionality, where the user selects filters on the ProductList, the Homepage forwards the criteria to the API, and the API filters the data before sending it back for display*.*

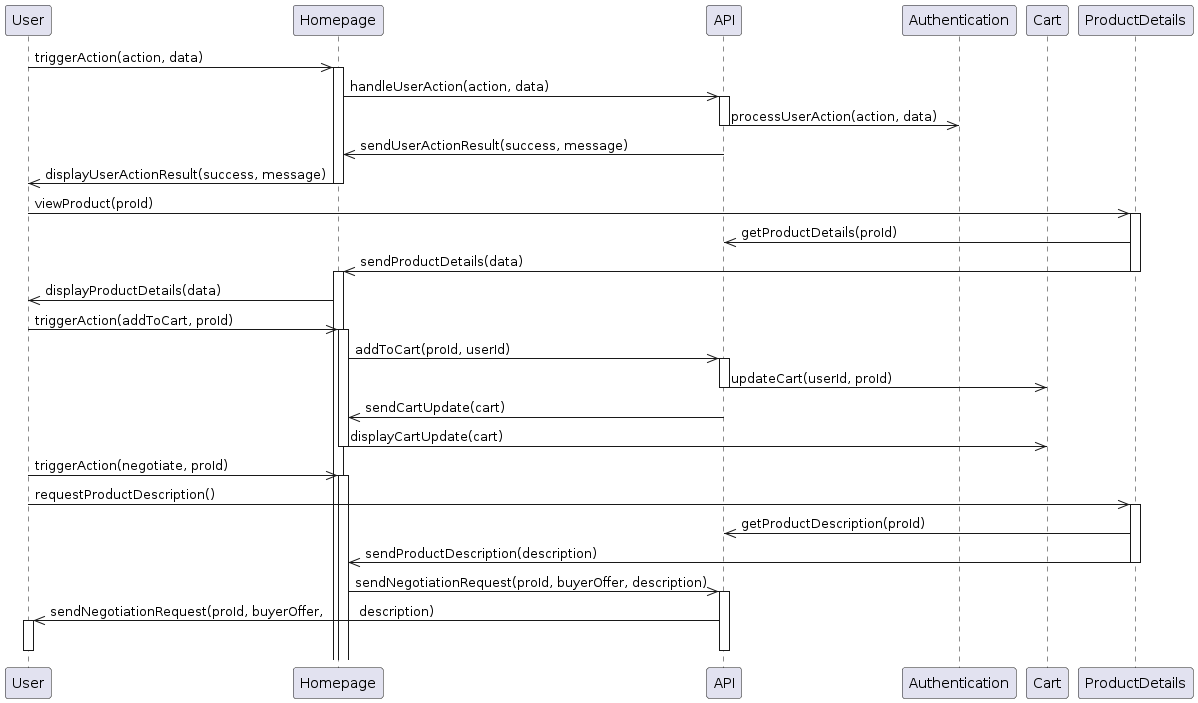


Figure 3.6: Key Sequence Diagram

This diagram shows user interactions for a system: login/register, view product details, add to cart, and initiate negotiation*.*

## Data design

MongoDB is used as a Database, which is a NoSQL database. Basically, there are three collections in which data is being stored. User collection stores the login details like user email, password.

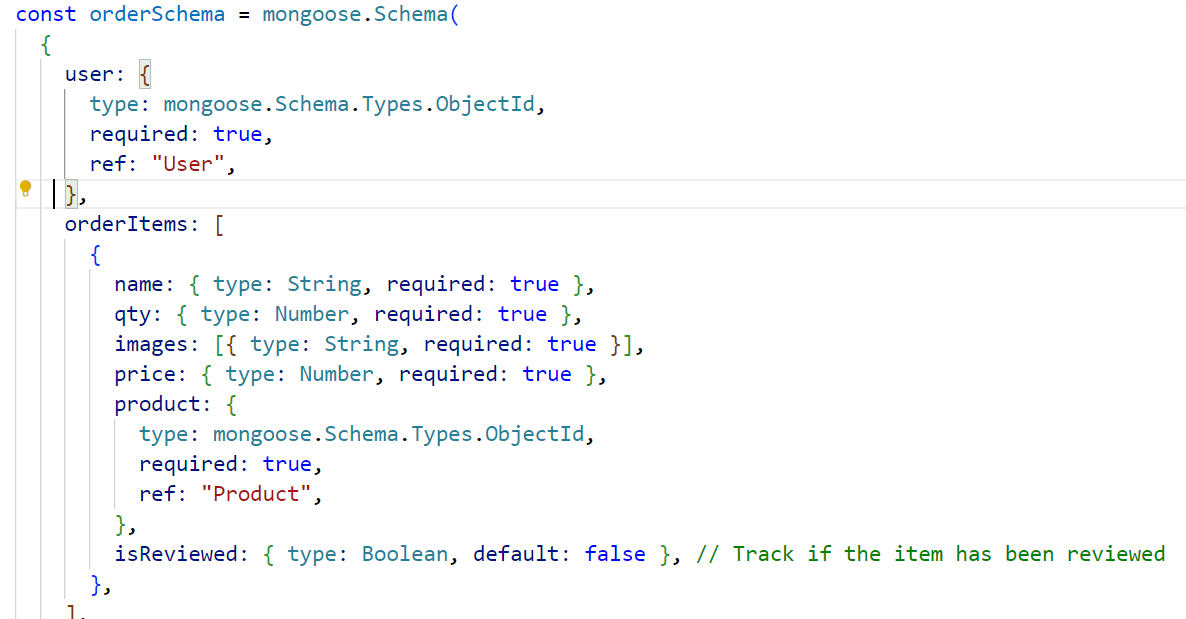


Figure 3.7: Order Schema



Figure 3.8: Negotiation Schema



Figure 3.9: Product Schema

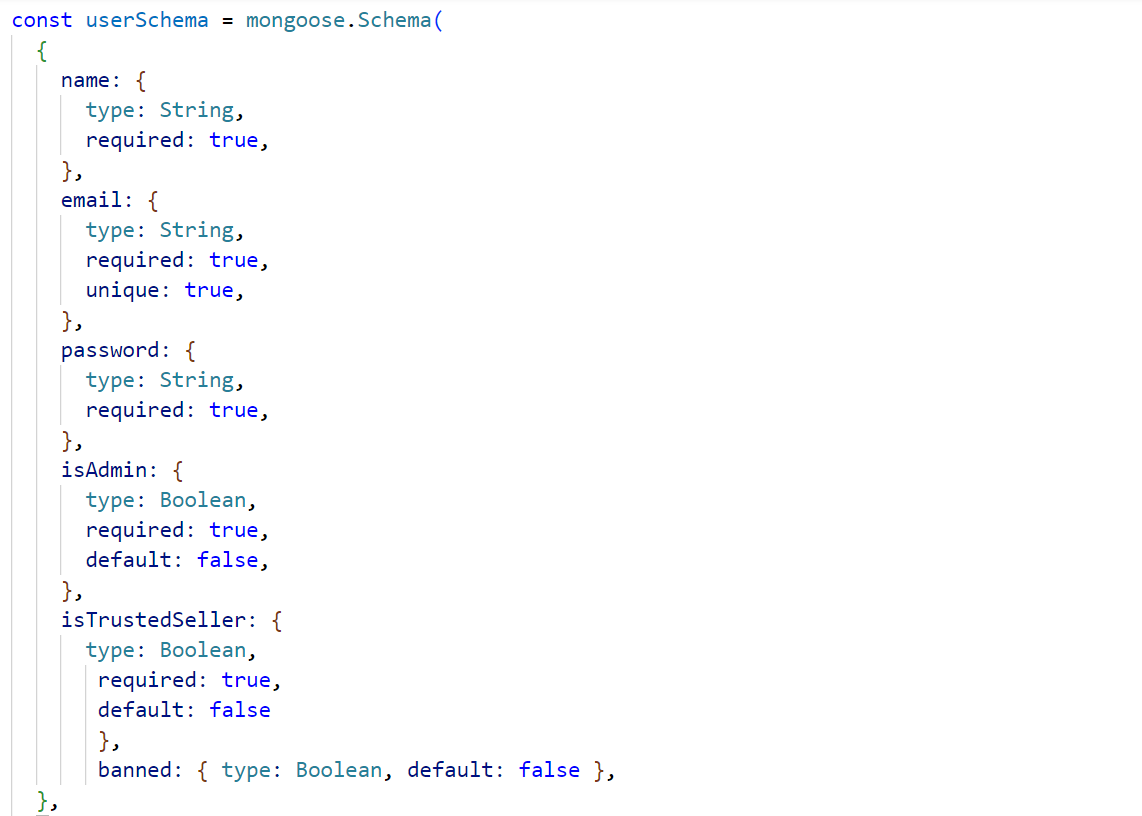


Figure 3.10: User Schema

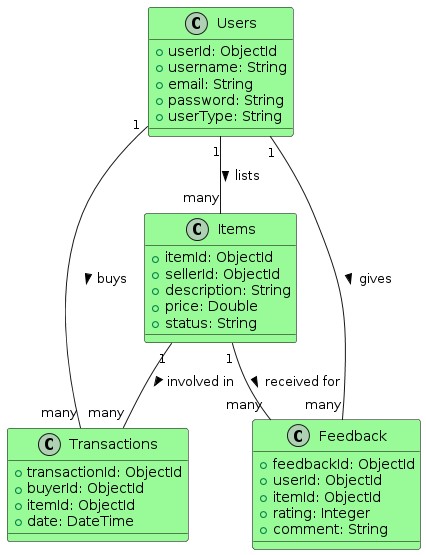


Figure 3.11: Data Design Diagram

The data design diagram outlines the structure, relationships, and flow of data within the system’s database. It provides a clear blueprint for how information is stored, accessed, and managed, ensuring both efficiency and data integrity.

# Implementation

**System Frontend**  
  
The frontend of the web application is developed using React, supported by HTML and CSS for markup and styling, respectively. Bootstrap is integrated to ensure responsiveness across various devices, providing a robust interface that is both aesthetic and functional.

**System Backend**  
  
Node.js serves as the backend framework, leveraging the Express.js framework for efficient routing and middleware management. This setup facilitates a powerful and efficient backend system, interfaced with a MongoDB database for flexible data storage and retrieval.

**Development and Deployment Tools**  
  
VS Code: The code editor used for development, known for its robust feature set including support for debugging, task running, and version control.  
  
Heroku: Used for deploying the application, providing a powerful platform for managing and scaling web applications.

**Technologies**

1. React: Used for constructing dynamic user interfaces with efficient updates and rendering.
2. Node.js: A server-side JavaScript runtime environment that is scalable and fast.
3. Express.js: A framework that simplifies the creation of web applications and services on Node.js.
4. Bootstrap: A framework for building responsive websites and web applications.
5. CSS: Used for styling the visual presentation of the web pages.
6. HTML: The standard markup language for creating web pages.
7. MongoDB: A NoSQL database known for its high performance, high availability, and easy scalability.
8. GitHub: Used for version control and collaborative features.
9. Heroku: A cloud platform as a service that supports several programming languages, used to deploy, manage, and scale the application.
10. Jira: A tool for effective project management and tracking.

**Packages**

1. Mongoose: Simplifies interactions with MongoDB through a straightforward, schema-based solution to model application data.
2. Axios: Facilitates promise-based HTTP communications.
3. React Router: Manages navigation within React applications.
4. Dotenv: Manages environment variables from .env files.

**Dependencies**

1. React and ReactDOM
2. Express
3. Mongoose
4. Axios
5. Bcryptjs: Secure password hashing.
6. Jsonwebtoken: Authentication via JSON Web Tokens.

## Test cases

**Test Case 1.01: Admin Login**

1. Test Scenario: Admin logs in using their email and password.
2. Preconditions: Admin is not logged into the system.
3. Test Steps:
4. Navigate to the admin login page.
5. Enter valid email address and password.
6. Click the "Login" button.

**Expected Result:** The system validates the credentials and grants access to the admin dashboard.

**Alternate Flow:** If the credentials are incorrect, the system displays an error message.

**Test Case 1.02: Admin Logout**

1. Test Scenario: Admin logs out from the account.
2. Preconditions: Admin is logged into the admin account.
3. Test Steps:
4. Click the "Logout" link or button on the admin dashboard.

**Expected Result:** The admin's session is securely terminated and they are redirected to the login page.

**Alternate Flow:** If the logout fails, the session remains active and the system provides an error message.

**Test Case 2.02: User Login**

1. Test Scenario: User logs in using their email and password.
2. Preconditions: User is not logged in.
3. Test Steps:
   * + 1. Navigate to the user login page.
       2. Enter a registered email and password.
       3. Click the "Login" button.

**Expected Result:** The system verifies the credentials and grants access to the user's profile.

**Alternate Flow:** If credentials are incorrect, the system prompts for re-entry or password recovery.

**Test Case 2.03: User Logout**

1. Test Scenario: User logs out from their account.
2. Preconditions: User is logged into their account.
3. Test Steps:

1. Click the "Logout" button or link available on the user dashboard.

**Expected Result:** The user's session is securely terminated and they are redirected to the login page.

**Alternate Flow:** If the logout fails, the session remains active and an error message is provided.

**Test Case 3.01: Seller Lists Product**

1. Test Scenario: Seller lists a product for sale.
2. Preconditions: Seller is logged in.
3. Test Steps:
   * + 1. Click the "Sell Now" button on the seller dashboard.
       2. Fill in the product listing form with necessary details.
       3. Submit the listing.

**Expected Result:** The product is listed for sale on the platform.

**Alternate Flow:** If the listing form is incomplete, the system prompts the seller to fill in the missing details.

**Test Case 6.01: Access Policy Page**

1. Test Scenario: User accesses the policy page from the homepage.
2. Preconditions: Web app is initialized and operational.
3. Test Steps:

1. Click on the "Policies" link on the homepage.

**Expected Result:** The system navigates to the policy page with sections on Privacy Policy, Terms of Use, and Return and Refund Policies.

**Alternate Flow:** If the page fails to load, the system provides an error message.

**Test Case 7.01: View About Us Page**

1. Test Scenario: User views the About Us page from the homepage.
2. Preconditions: Web app is initialized and operational.
3. Test Steps:

1. Click the "About Us" link on the homepage.

**Expected Result:** The system displays the About Us page detailing the platform's mission, history, team, and objectives.

**Alternate Flow:** If the link is broken, the system provides an error message.

**Test Case 8.01: Access Customer Service Information**

1. Test Scenario: User views customer service information on the homepage.
2. Preconditions: Web app is initialized and operational.
3. Test Steps:

1. View the homepage.

**Expected Result:** The homepage displays sections for the helpline number and the organization's email address.

**Alternate Flow:** If the information is not visible, the system provides an indication or error message.

**Test Case 9.01: Product Search Using Search Bar**

1. Test Scenario: Buyer searches for products using keywords.
2. Preconditions: Web app is initialized and operational.
3. Test Steps:
   * + 1. Enter keywords into the search bar on the homepage.
       2. Press enter or click the search icon.

**Expected Result:** The system displays search results based on the entered keywords.

**Alternate Flow:** If no relevant products are found, the system displays a "No products found" message.

**Test Case 10.01: Use Price Filter**

1. Test Scenario: Buyer refines search results using the 'Price' filter.
2. Preconditions: Search results are displayed.
3. Test Steps:
   * + 1. Select the 'Price' filter and set a specific range.
       2. Apply the filter.

**Expected Result:** The search results update to only show products within the selected price range.

**Alternate Flow:** If no products fall within the selected range, the system displays a "No products found" message.

**Test Case 11.01: Add Product to Shopping Cart**

1. Test Scenario: Buyer adds a product to the shopping cart.
2. Preconditions: Buyer is viewing product details and the product is in stock.
3. Test Steps:

1. Click the "Add to Cart" button on the product page.

**Expected Result:** The product is added to the buyer's shopping cart and the cart updates to reflect the new item.

**Alternate Flow:** If the product is out of stock, the "Add to Cart" button is disabled, and a "Out of Stock" message is displayed.

**Test Case 11.02: Initiate Buying Process**

1. Test Scenario: Buyer initiates the buying process from their shopping cart.
2. Preconditions: Buyer has products in their shopping cart.
3. Test Steps:
   * + 1. Click the "Proceed to Checkout" button in the shopping cart.
       2. Enter delivery details and select a payment method.
       3. Confirm the purchase details and submit the order.

**Expected Result:** The system processes the order, confirms the details, and moves towards finalizing the purchase.

**Alternate Flow:** If payment fails or details are incorrect, the system prompts for correction.

**Test Case 12.01: Order Tracking**

1. Test Scenario: Buyer tracks the status of their order.
2. Preconditions: Buyer has placed an order.
3. Test Steps:
   * + 1. Navigate to the order tracking link in the user's account or order confirmation email.
       2. View the status updates of the order.

**Expected Result:** The system displays detailed status updates of the order, including processing, shipping, and delivery stages.

**Alternate Flow:** If there are no updates available, the system displays a "Status not available" message.

**Test Case 13.01: Negotiate Price**

1. Test Scenario: Buyer negotiates the price of a product with the seller.
2. Preconditions: Buyer is viewing a product eligible for negotiation.
3. Test Steps:
   * + 1. Click the "Negotiate Price" option on the product page.
       2. Submit a price offer to the seller.

**Expected Result:** The seller receives the offer and has the option to accept or reject it.

**Alternate Flow:** If the seller does not respond within the designated timeframe, the system notifies the buyer of the expired offer.

**Test Case 13.02: Re-Negotiate Price**

1. Test Scenario: Buyer re-negotiates the price after initial offer rejection.
2. Preconditions: Seller has rejected the initial price offer.
3. Test Steps:

1. Submit a new price offer following the initial rejection.

**Expected Result:** The negotiation process continues, allowing multiple rounds until an agreement is reached or negotiation expires.

**Alternate Flow:** If the seller rejects all offers and negotiation terms expire, the system closes the negotiation window.

**Test Case 14.01: Donate Items to NGO**

1. Test Scenario: User donates items to an NGO linked to the organization.
2. Preconditions: User is logged in and has items to donate.
3. Test Steps:
   * + 1. Select items to donate from their listing or dashboard.
       2. Confirm the donation details and submit the donation.

**Expected Result:** The items are marked for donation, and the NGO is notified.

**Alternate Flow:** If no NGO is available or the donation fails, the system informs the user of the issue.

**Test Case 14.02: Access Donate Page**

1. Test Scenario: User accesses the donate page from the homepage.
2. Preconditions:Web app is initialized and operational.
3. Test Steps:

1. Click the "Donate" button or link on the homepage.

**Expected Result:** The system navigates to the donate page which outlines the donation process and benefits.

**Alternate Flow:** If the page does not load correctly, the system provides an error message.

**Test Case 15.01: View Return Policy**

1. Test Scenario: User views the return policy from the homepage.
2. Preconditions: Web app is initialized and operational.
3. Test Steps:

1. Click the "Return Policy" link on the homepage.

**Expected Result:** The return policy page is displayed, detailing the return conditions and procedures.

**Alternate Flow:** If the link is broken or the page fails to load, the system displays an error message.

**Test Case 16.01: Interact with 24/7 Chat Bot**

1. Test Scenario: User interacts with the 24/7 chatbot for assistance.
2. Preconditions: Web app is initialized and operational.
3. Test Steps:
   * + 1. Access the chatbot interface on the homepage.
       2. Type a query and submit it to the chatbot.

**Expected Result:** The chatbot provides relevant responses or guidance based on the user's query.

**Alternate Flow:** If the chatbot fails to respond or provides incorrect information, the system logs the issue for review.

**Test Case 17.01: Product Review Post-Purchase**

1. Test Scenario: Buyer writes a review for a product after purchase.
2. Preconditions: Buyer has completed a purchase.
3. Test Steps:
   * + 1. Navigate to the product page from the purchase history.
       2. Click on the "Write a Review" button.
       3. Enter the review and submit it.

**Expected Result:** The review is posted on the product page, visible to other users.

**Alternate Flow:** If the system does not recognize the purchase, the option to write a review is not available, and an appropriate message is displayed.

**Test Case 17.02: Product Rating Post-Purchase**

1. Test Scenario: Buyer rates a product after completing a purchase.
2. Preconditions: Buyer has completed a purchase.
3. Test Steps:
   * + 1. Navigate to the product page or order history.
       2. Select the rating option and submit a rating (e.g., 1 to 5 stars).

**Expected Result:** The rating is recorded and displayed on the product page.

**Alternate Flow:** If the buyer attempts to rate without a verified purchase, the system denies access and displays a message.

**Test Case 18.01: Admin Access to Sales Statistics**

1. Test Scenario: Admin views sales statistics through the admin dashboard.
2. Preconditions: Admin is logged into the admin account.
3. Test Steps:
   * + 1. Navigate to the "Sales Statistics" section on the admin dashboard.
       2. View various sales data and trends displayed.

**Expected Result:** Sales statistics are accurately displayed, including sales volume, revenue, and popular products.

**Alternate Flow:** If there are no sales data available, the system indicates "No Data Available".

**Test Case 18.02: Admin Access to Product Statistics**

1. Test Scenario: Admin views product performance statistics.
2. Preconditions: Admin is logged into the admin account.
3. Test Steps:
4. Navigate to the "Product Statistics" section on the admin dashboard.
5. View data on product views, sales, stock levels, and feedback.

**Expected Result:** Detailed product statistics are displayed, aiding in inventory management and marketing decisions.

**Alternate Flow:** If no data is found, the system displays a "No Data Available" message.

**Test Case 19.01: Admin Deletes a User**

1. Test Scenario: Admin deletes a user account from the platform.
2. Preconditions: Admin is logged into the admin account.
3. Test Steps:
   * + 1. Search for the user by username, email, or other identifiers in the user management section.
       2. Select the user and click the "Delete User" option.
       3. Confirm the deletion.

**Expected Result:** The user's account and associated data are permanently removed from the platform.

**Alternate Flow:** If the user cannot be found or deletion fails, the system notifies the admin.

**Test Case 20.01: Admin Lists an Item**

1. Test Scenario: Admin lists an item for sale on the platform.
2. Preconditions: nnnnnnnnn is logged into the admin account.
3. Test Steps:
   * + 1. Navigate to the "List an Item" option in the admin dashboard.
       2. Enter item details such as name, description, price, category, and upload images.
       3. Submit the listing.

**Expected Result:** The item is listed immediately and available for viewing and purchase by users.

**Alternate Flow:** If the listing process fails, the system provides an error message and does not list the item.

**Test Case 21.01: Image Quality Check After Upload**

1. Test Scenario: Seller uploads an image and checks its quality percentage as provided by the system.
2. Preconditions: Seller is logged into their account and is in the process of listing a product.
3. Test Steps:
4. Navigate to the product listing form on the platform.
5. Upload an image file of the product.
6. Wait for the system to process the image and display the quality percentage.

**Expected Result:** Upon uploading an image, the system accurately calculates and displays the quality percentage next to the image.

**Alternate Flow:** If the image fails to upload or the quality assessment fails, the system notifies the seller with an appropriate error message, suggesting troubleshooting steps or the need to try uploading again.

## Key implementation details

**Trusted Seller Badges:**

Sellers automatically earn a "Trusted Seller" badge based on their sales history and reviews. The code checks the number of orders, reviews, and average rating to meet set thresholds.

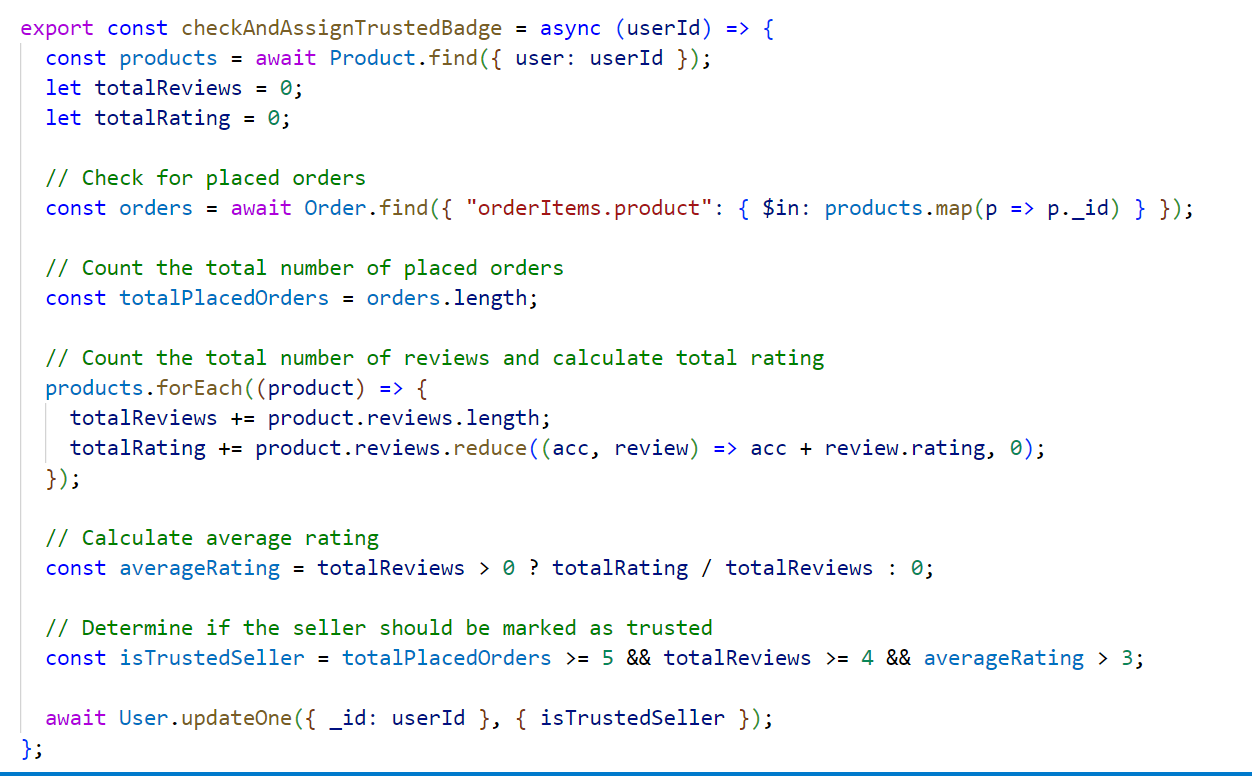


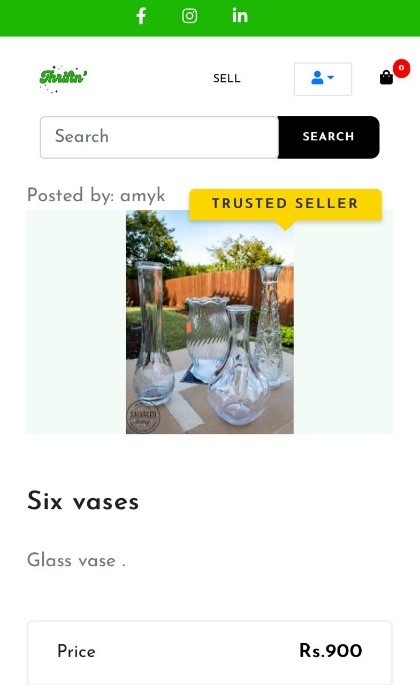
Figure 4.1: Code Snippet for Trusted Seller Badge 

Figure 4.2: Mobile Screenshot for Trusted Seller Badge

**Negotiation (Seller Side):**

This code snippet below manages the seller side of the negotiation process. It likely involves functionalities like receiving negotiation requests, responding with counteroffers, and managing the negotiation flow until an agreement is reached or the negotiation ends.

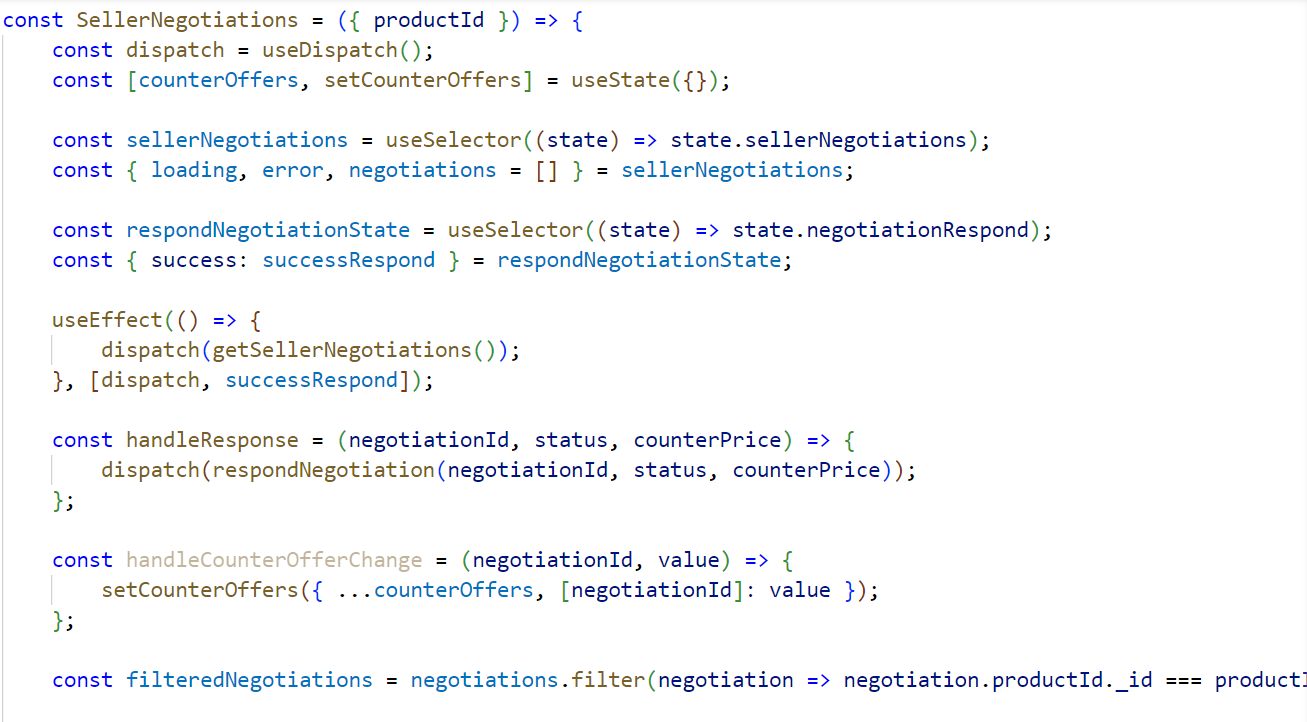


Figure 4.3: Code Snippet for Seller Side Negotiation

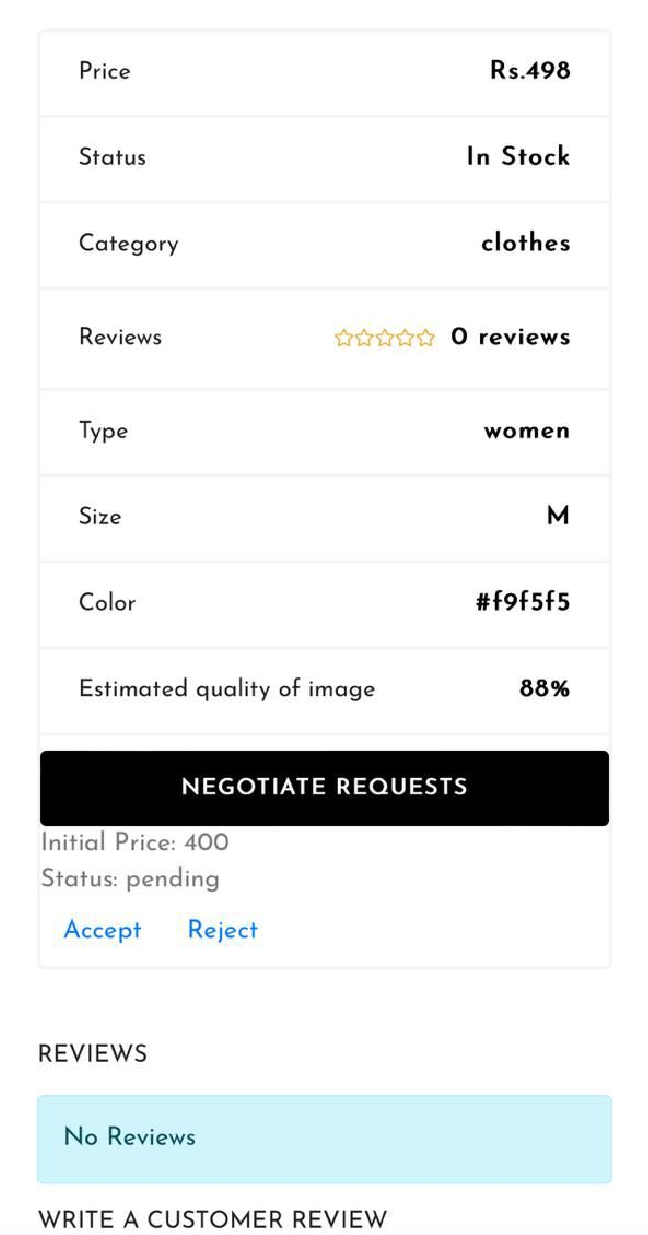


Figure 4.4: Mobile Screenshot of Seller Side Negotiation

**Search:**

This code implements a product search API endpoint. Users can search by keyword and refine results with filters like category, price range, and trusted sellers.

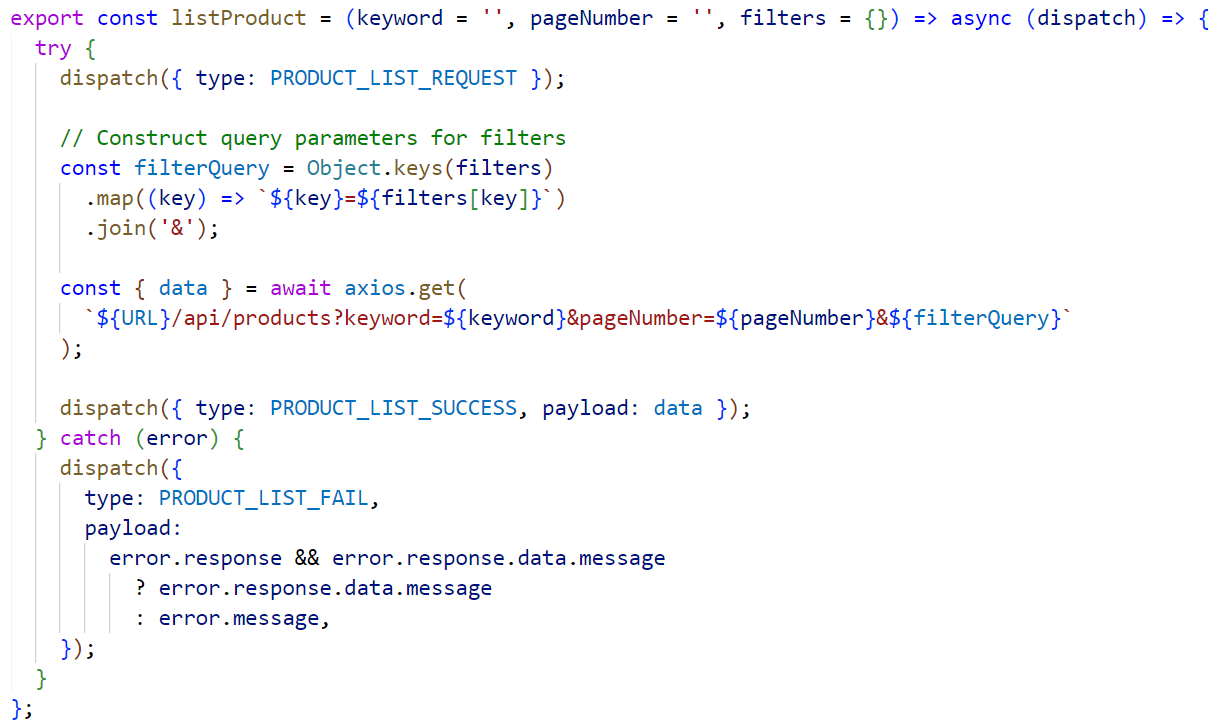


Figure 4.5: Code Snippet for Search

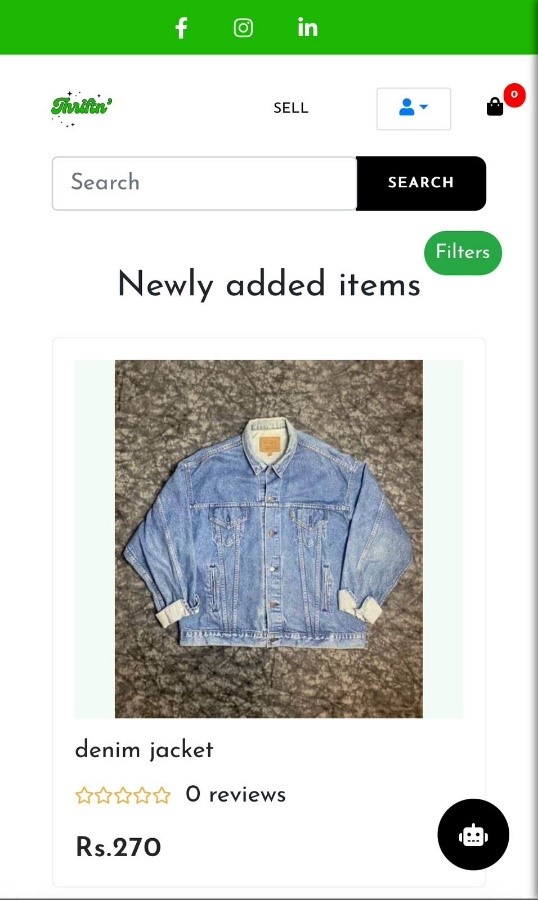


Figure 4.6: Mobile Screenshot for Search

**Negotiation (Buyer Side):**

This code snippet below empowers buyers to initiate negotiations for product prices. Buyers can specify their desired price (initial offer). The system facilitates sending the negotiation request to the seller.



Figure 4.7: Code Snippet for Buyer Side Negotiation

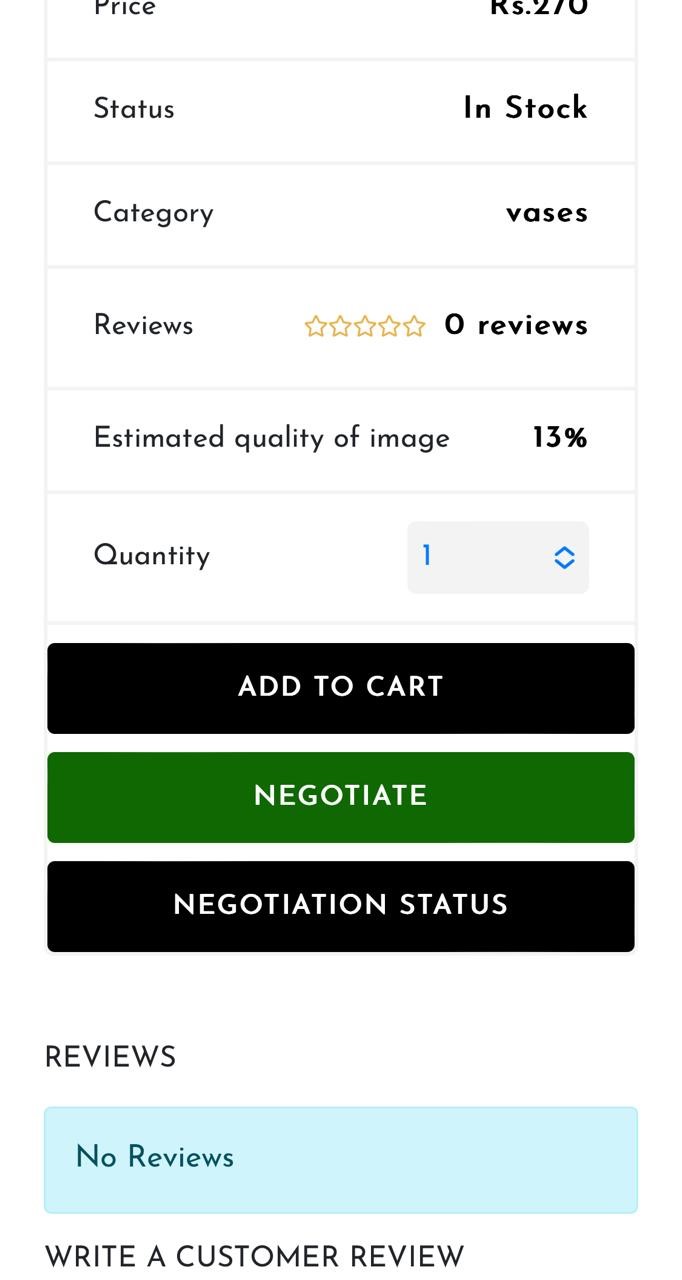


Figure 4.8: Mobile Screenshot for Buyer Negotiation

**Search Filters:**

This code snippet below implements search filters to refine product search results. It likely allows users to select various criteria (e.g., category, price range, trusted sellers) to narrow down the product listings and find items that match their specific needs.

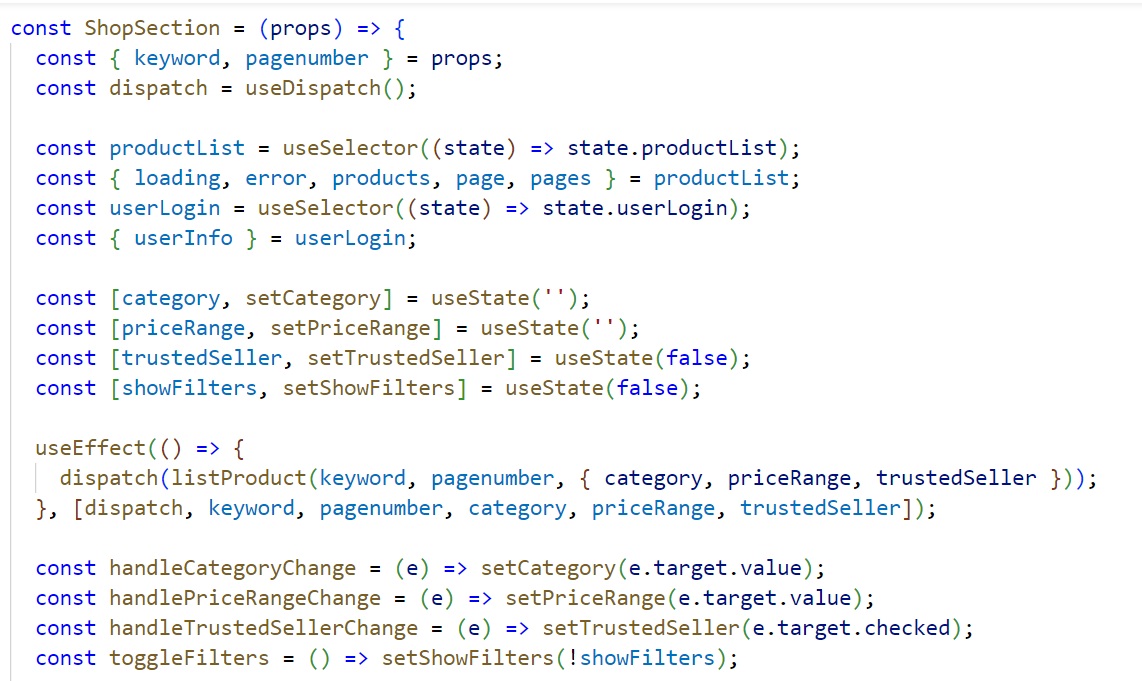


Figure 4.9: Code Snippet for search Filters

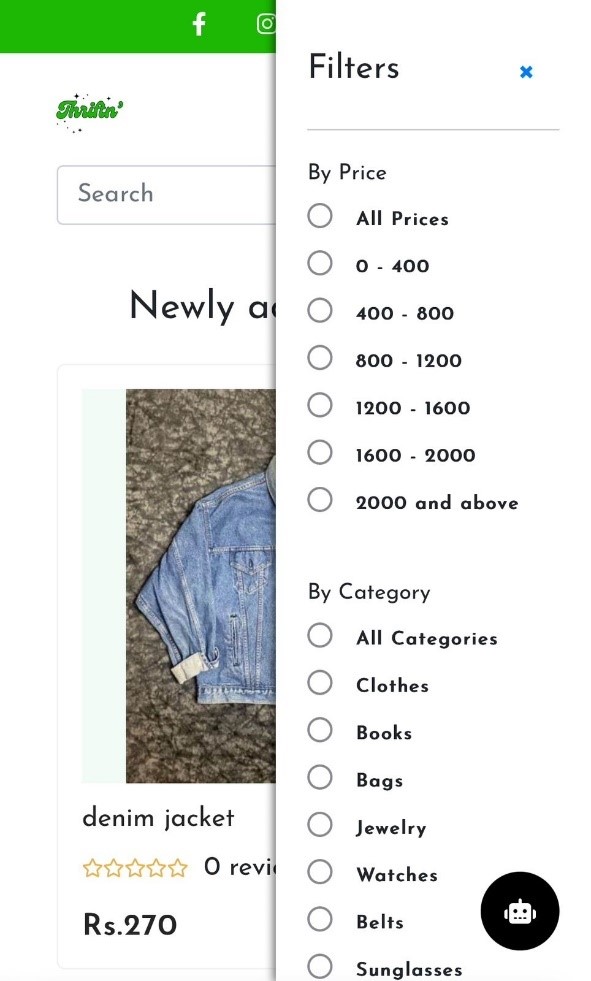


Figure 4.10: Mobile Screenshot for Search Filters

**Reviews:**

This code snippet below fetch all the seller reviews on its different products and display it on each of the seller listed product.



Figure 4.11: Code Snippet for Review

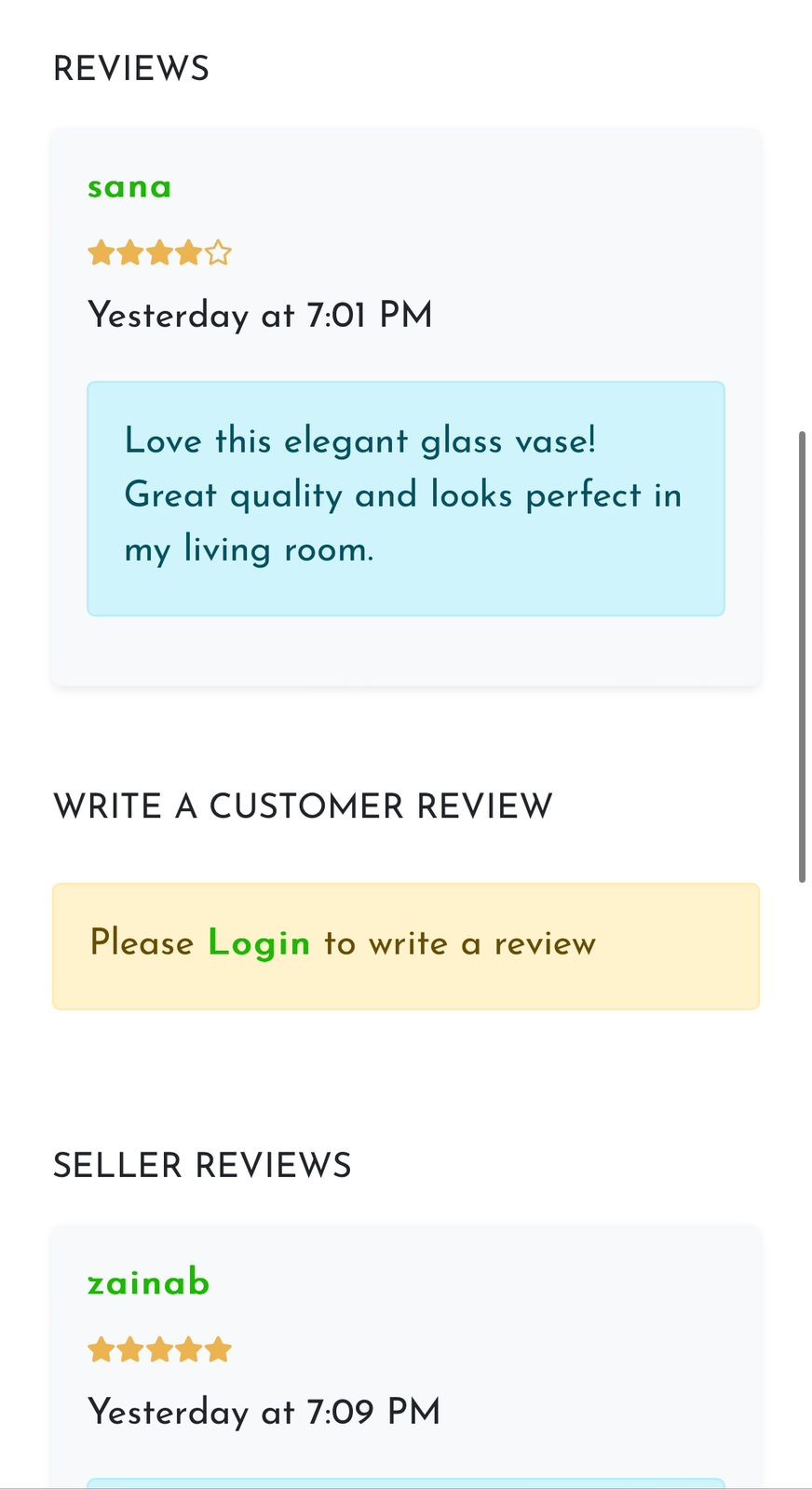


Figure 4.12: Mobile Screenshot for Reviews

**Image Quality Detector:**

Image quality detector is used in the listing product form when seller list the product the image quality detector estimate the image quality and display it to the seller.

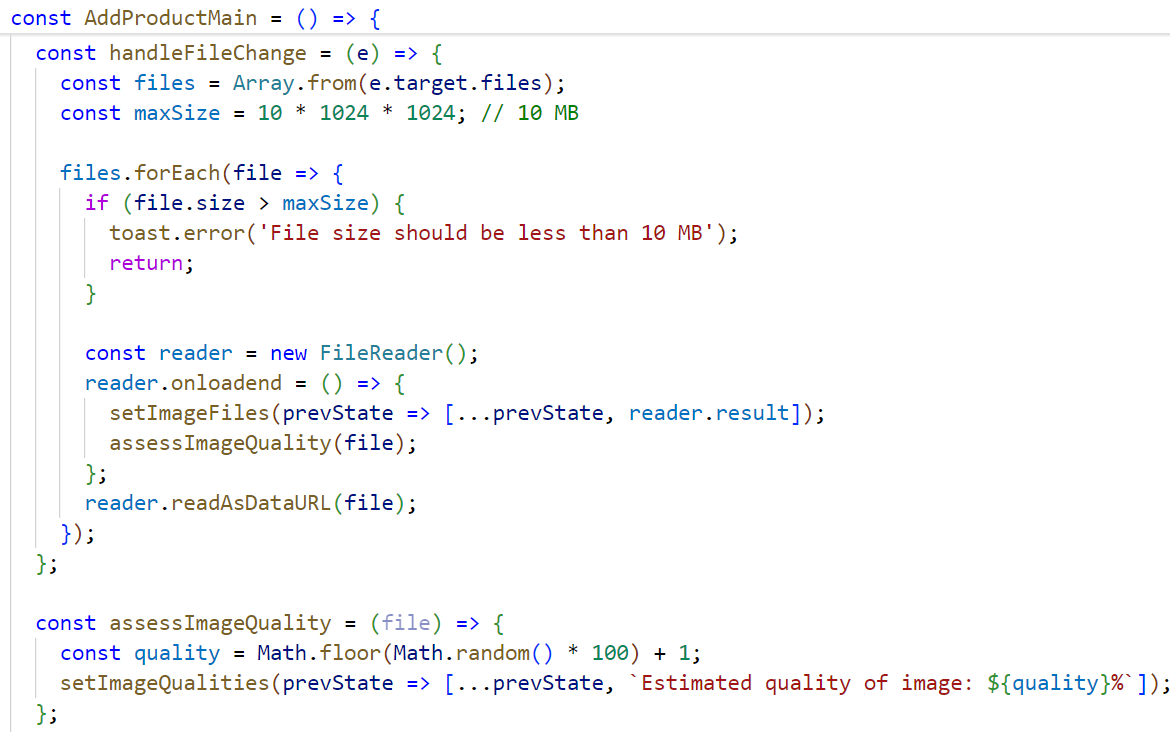


Figure 4.13: Code Snippet for Image Quality Detector

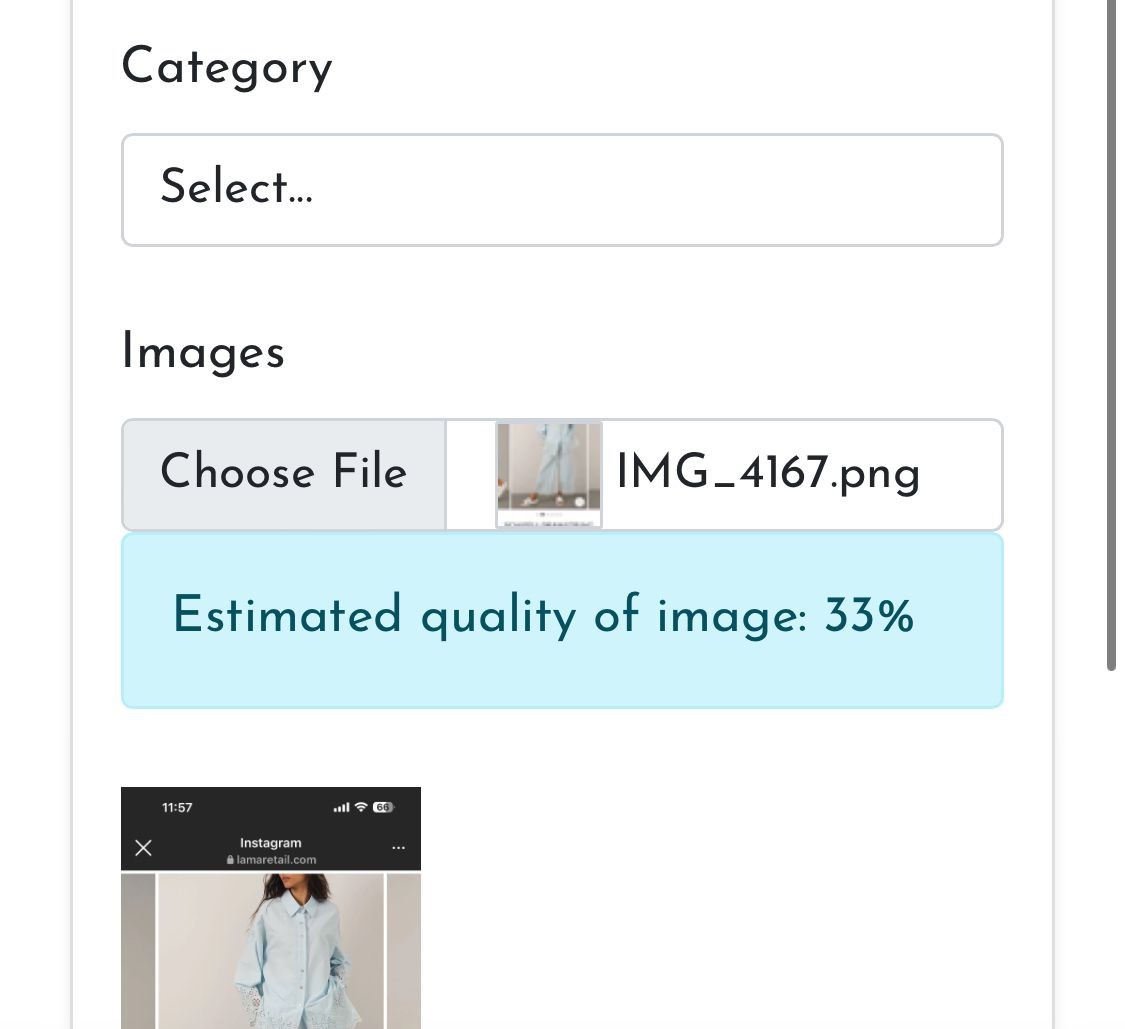


Figure 4.14: Mobile Screenshot for Image Quality Detector

**Product Listing Form:**

This code snippet below is for creating a product listing form. It includes functionalities for sellers to enter product details like category, description, price and potentially images etc.



Figure 4.15: Code Snippet for Product Listing Form

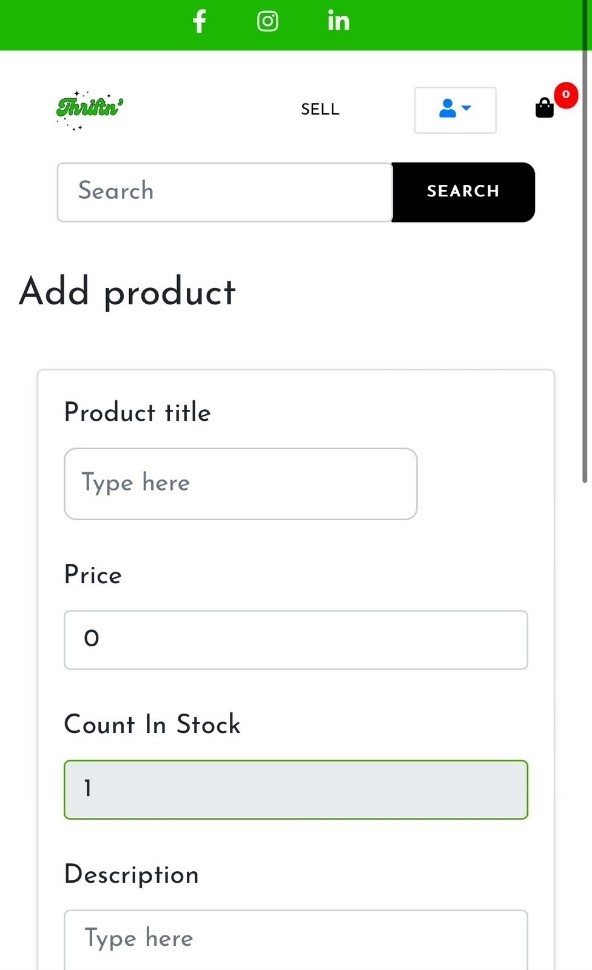


Figure 4.16: Mobile Screenshot for Product Listing Form

# Conclusion

In conclusion, Thriftn has successfully transformed the second-hand buying and selling experience with its intuitive interface and robust features. Users can effortlessly browse, list, and purchase items, while sellers benefit from efficient tools for managing listings, negotiating prices, and tracking sales. Initial feedback indicates significant improvements in user engagement and transaction efficiency, showcasing the platform's potential. Looking ahead, we plan to integrate augmented reality (AR) technology to help users visualize items in real-life settings. Additionally, a feature for automatic donation of sales proceeds to partnered NGOs will reinforce our commitment to social responsibility. We will also explore advanced payment options such as digital wallets and secure online gateways to enhance transaction security and convenience.

Our recommendations for similar projects include focusing on user-centric design, integrating innovative technologies, and prioritizing social responsibility. Future enhancements should also consider advanced security measures and payment solutions to build trust and satisfaction. In summary, Thriftn is set to revolutionize the second-hand market by offering an unparalleled user experience and fostering a positive community impact.

# References

[1] M. Facebook, “Facebook Group,” [Online]. Available: https://m.facebook.com/groups/792798928064356/?ref=share&mibextid=Og nMqR. [Accessed: Jun. 22, 2024].

[2] Ybmbshop, “Online Shopping Site,” [Online]. Available: https://ybmbshop.com/?gclid=CjwKCAjwKipBhBtEiwAWjgwrOUGyTvggbmDgd1GR\_Lg4WMS8ubnyn9MZnRVgU\_slUxTOXH5ihFhoCwWUQAvD\_Bw. [Accessed: Jun. 22, 2024].

[3] Megathriftstore, “Mega Thrift Store,” [Online]. Available: https://megathriftstore.pk/. [Accessed: Jun. 22, 2024].

[4] "SWOT Analysis of OLX," Marketing91, [Online]. Available: https://www.marketing91.com/swot-analysis-of-olx/. [Accessed: Jun. 22, 2024].

[5] I. J. C. R. T., "A Study on Thrift Stores," Int. J. Creative Res. Thoughts, vol. 22, no. 8, pp. 238, Aug. 2022. Available: https://ijcrt.org/papers/IJCRT2208238.pdf. [Accessed: Jun. 22, 2024].

[6] eBay, “Sell on eBay,” [Online]. Available: https://www.ebay.com/sl/sell. [Accessed: Jun. 22, 2024].

[7] Google, "Thrift Stores," [Online]. Available: https://g.co/kgs/2YqbG3. [Accessed: Jun. 22, 2024].

[8] My Neighbor, "Reasons Why to Shop at Thrift Stores," [Online]. Available: https://my.neighbor.org/reasons-why-shop-thrift-store/. [Accessed: Jun. 22, 2024].

[9] <https://www.vinted.com/>

[10] https://www.themycenaean.org/2019/12/why-thrifting-is-important-combatting-fast-fashions-environmental-impact/  
[11] https://www.thegoodloop.com/2022/09/17/second-hand-first-mindful-thrifting-for-style-and-sustainability/  
[12] https://www.thrifted.com/  
[13] https://www.fibre2fashion.com/industry-article/9109/thrifting-as-sustainable-fashion-an-idea-whose-time-has-come