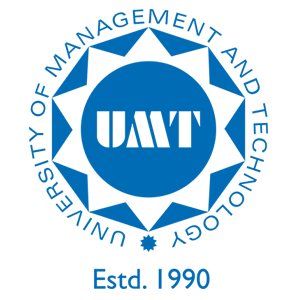
****

**Final Year Project Report**

Giveaway Stuff

**Project Advisor: Abdul Jamil**

**Submitted By:**

**Ali Raza Arshad F2019105036**

**Muhammad Murtaza Arshad F2019065101**

**Muhammad Usman Ghani F2019266154**

**Session**

**2019 - 2023**

**Department of Computer Science**

**School of System and Technology**

**University of Management and Technology**

**C-II Johar Town Lahore Pakistan**

**Dedication**

We take great pride in dedicating this project to our relentless pursuit of knowledge, innovation, and progress. Our unwavering commitment to this transformative journey is evident in every aspect of this project, which stands as a testament to our dedication. We owe our success to the Almighty, whose guidance and blessings have been the cornerstone of our strength. We have surmounted obstacles, gained valuable insights, and attained unparalleled achievements with His grace.

We would also like to express our heartfelt appreciation to our families and friends, whose steadfast support and understanding have been instrumental to our success. Their encouragement, unwavering belief in our abilities, and constant presence have fueled our determination to excel and surpass our limits.

Our project supervisor, Abdul Jamil, deserves special dedication for his unwavering commitment, guidance, and patience, which have driven this project. His invaluable mentorship, technical expertise, and unyielding support have propelled us forward and helped us realize our full potential. With deep gratitude and a sense of purpose, we dedicate this project to all who contributed to its realization. May it resonate, inspire, and serve as a catalyst for future generations, fostering continued growth, innovation, and progress.

**Acknowledgment**

We would like to express our utmost gratitude to everyone who played a pivotal role in the successful completion of this project. Our sincerest thanks go to our project supervisor, Abdul Jamil, for providing us with invaluable guidance, expertise, and unwavering dedication. We also want to extend our appreciation to our esteemed faculty members and mentors for their invaluable contributions to our academic and professional growth. Moreover, we are grateful for the unwavering support and understanding of our families and friends as we embarked on this endeavor. Our heartfelt appreciation also goes to the university, institution, and Information Technology Department at UMT for providing us with all the necessary resources. Finally, we acknowledge the contributions of all individuals who shared their insights, provided feedback, or supported us in any way. Your involvement has been crucial to our journey, and we are truly grateful for your unwavering support.

**Project Title** Giveaway Stuff

**Objective** Create a platform where users can share their unneeded assets with others respectfully and inclusively.

**Undertaken by** Students of the BS (CS) final year.

**Supervised by**  Abdul Jamil

**Starting Date** 30/12/2022

**Completion Date** 25/09/2023

**Tools Used**

*Flutter*

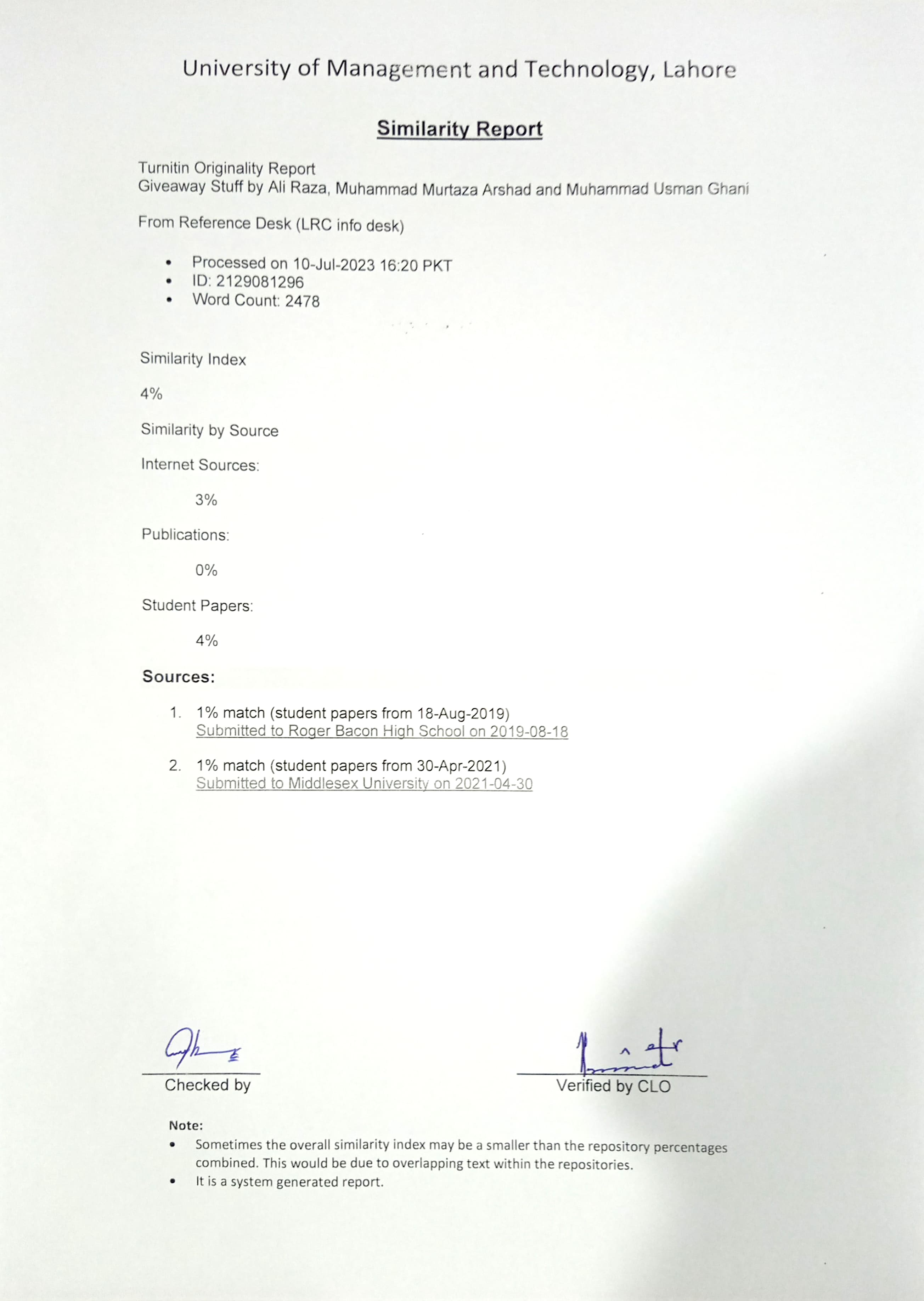
*Android Studio*

**Technologies**

*React JS*

*Firebase DB*

**Operating System**Android

****

**Abstract**

The Giveaway Stuff platform was expertly crafted by final-year Bachelor of Science in Information Technology students. Its objective is to cultivate a warm and respectful community where users can freely share their unneeded items. The platform actively promotes repurposing without resorting to the term "charity" to ensure that all users are recognized and appreciated. Giving and receiving items is incredibly easy, with no location or time restrictions. In the past, obtaining free possessions required considerable physical exertion and time. However, with our platform, giving items is as simple as a click, whether at home, in the office, or anywhere else. Similarly, our platform is the perfect solution for users struggling to find affordable items. It helps them locate high-quality, low-cost items that fit their budget.

Revision Chart

|  |  |  |
| --- | --- | --- |
| **Primary Author(s)** | **Responsibility** | **Date Completed** |
| Ali Raza Arshad | Chapters 1, 4, 7, 9, 12 | 25-09-2023 |
| Muhammad Murtaza Arshad | Chapters 3, 5, 6, 10 | 18-09-2023 |
| Muhammad Usman Ghani | Chapters 2, 5, 8, 11 | 14-09-2023 |

# Contents

[Contents 1](#_Toc146705395)

[1. Introduction 4](#_Toc146705396)

[1.1 Motivations 4](#_Toc146705397)

[1.2 Project Overview 4](#_Toc146705398)

[1.3 Problem Statement 4](#_Toc146705399)

[1.4 Objectives 4](#_Toc146705400)

[2. Domain Analysis 5](#_Toc146705401)

[2.1 User 5](#_Toc146705402)

[2.2 Stakeholders 5](#_Toc146705403)

[2.3 Affected Groups with social or economic impact 5](#_Toc146705404)

[2.4 Dependencies/ External Systems 6](#_Toc146705405)

[2.5 Reference Documents 6](#_Toc146705406)

[3. Requirements analysis 7](#_Toc146705407)

[3.1 Requirements 7](#_Toc146705408)

[3.2 List of Actors 8](#_Toc146705409)

[3.3 List of use cases 9](#_Toc146705410)

[3.4 System use case diagram 10](#_Toc146705411)

[3.5 Extended use cases 11](#_Toc146705412)

[3.6 User interfaces (mock screens) 24](#_Toc146705413)

[4. Data flow diagram 28](#_Toc146705414)

[4.1 Data Flow Diagram Level 0 28](#_Toc146705415)

[5. System Design 29](#_Toc146705416)

[5.1 System Architecture Diagram 29](#_Toc146705417)

[5.2 Class Diagram 30](#_Toc146705418)

[5.3 Sequence Diagrams 31](#_Toc146705419)

[5.4 Collaboration Diagrams 31](#_Toc146705420)

[5.5 ERD 32](#_Toc146705421)

[5.6 Data Dictionary 33](#_Toc146705422)

[6. Implementation details 34](#_Toc146705423)

[6.1 Development Setup 34](#_Toc146705424)

[6.2 Deployment setup 34](#_Toc146705425)

[6.3 Algorithms 34](#_Toc146705426)

[6.4 Constraints 34](#_Toc146705427)

[6.4.1 Assumptions 34](#_Toc146705428)

[6.4.2 System constraints 35](#_Toc146705429)

[6.4.3 Restrictions 35](#_Toc146705430)

[6.4.4 Limitations 35](#_Toc146705431)

[7. Testing 36](#_Toc146705432)

[7.1 Extended Test Cases 36](#_Toc146705433)

[7.2 Decision Table 42](#_Toc146705434)

[7.2.1 Code snippet 42](#_Toc146705435)

[7.2.2 Decision coverage table 43](#_Toc146705436)

[7.3 Traceability Matrix 44](#_Toc146705437)

[7.3.1 RID vs UCID 44](#_Toc146705438)

[7.3.2 Prototypes (RID vs PID) 45](#_Toc146705439)

[7.3.3 Test Cases (RID vs TID) 46](#_Toc146705440)

[8. Results/Output/Statistics 47](#_Toc146705441)

[8.1 %completion 47](#_Toc146705442)

[8.2 %accuracy 47](#_Toc146705443)

[8.3 %correctness 47](#_Toc146705444)

[9. Conclusion 48](#_Toc146705445)

[10. Future work 49](#_Toc146705446)

[11. Bibliography 50](#_Toc146705447)

[11.1 Books/Articles 50](#_Toc146705448)

[12. Appendix 51](#_Toc146705449)

[12.1 Glossary of terms 51](#_Toc146705450)

[12.2 Pre-requisites 52](#_Toc146705451)

Definitions and Acronyms

|  |  |
| --- | --- |
| **Acronym** | **Definition** |
| **UC** | Use Case |
| **UI** | User Interface |
| **GAS** | Giveaway Stuff |
| **Web** | Website |
| **App** | Application |

# Introduction

## Motivations

Giveaway Stuff aims to address the issue of excess assets that many individuals accumulate over time, which often go unused. By providing a platform for users to share their unneeded belongings, the project seeks to reduce unnecessary stuff. The motivations behind the Giveaway Stuff platform encompass community building and promoting a culture of sharing.

## Project Overview

The Giveaway Stuff platform is an Android application developed to facilitate the sharing of unneeded stuff among users. The project provides a user-friendly interface where individuals can offer their unneeded stuff to others. The platform focuses on promoting a culture of sharing. Through the platform, users can create listings for their available Stuff, browse through the offerings of others, and engage in direct communication to arrange pickups or deliveries. The project aims to create an inclusive and respectful environment.

## Problem Statement

The Giveaway Stuff platform addresses the problem of accumulated stuff that often goes unused. Many individuals find themselves with surplus stuff that they no longer need or want, but they lack an efficient and convenient way to pass it on to others. This leads to unnecessary waste, as these belongings are often discarded. Additionally, the absence of a dedicated platform for sharing unneeded stuff creates missed opportunities for individuals to connect, build a sense of community, and support each other. The problem statement focuses on the need for a platform that facilitates the sharing of belongings and fosters a culture of giving.

## Objectives

* Create a user-friendly platform enabling individuals to share their unneeded possessions.
* Foster a sense of community and connection among users through the act of giving and receiving.
* Provide a convenient and efficient way for users to browse, search, and communicate with each other to arrange pickups or deliveries.
* Cultivate a culture of sharing and generosity where individuals feel empowered to contribute to the well-being of others.
* Ensure a respectful and inclusive environment where all users are treated with dignity and gratitude for their participation.

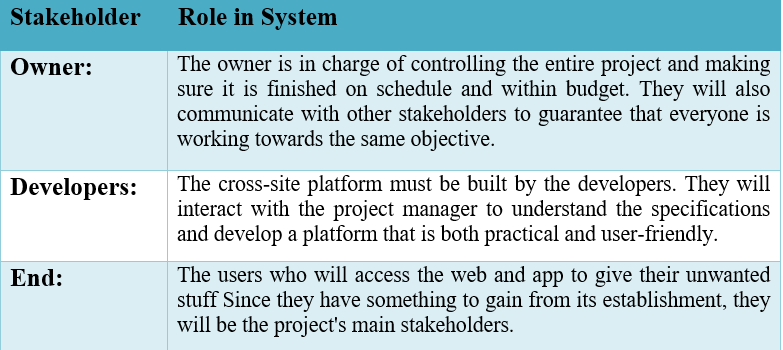
# Domain Analysis

## User

* **Login and register as a sender**. The user can upload stuff they no longer need or want to give away and include information about them such as their condition, and any other pertinent characteristics.
* **Login and register as a receiver**. The user has the option to explore the stuff up for grabs, conduct a search for certain stuff, and make requests for stuff they are interested in. To arrange for pick-up or delivery of the package, they could simply get in touch with the sender.

## Stakeholders

Table : Role of Stakeholders



## Affected Groups with social or economic impact

* **Developer**

The development of the app will have a social impact on the developer, and the chat between the user and UI will be user-friendly for all users.

* **Sender**

The sender will experience social effects as a result of being able to send his stuff while resting at home and using our platform.

* **Receiver**

By using our platform, the receiver will benefit from free stuff, which will impact them socially and financially.

## Dependencies/ External Systems

PayPal/Stripe/Cash on Delivery

## Reference Documents

* Ewraniwego, “Simple Giveaways – Grow your business, email lists and traffic with contests,”.
* T. Brower Ph.D., “Giving Is Good—For Others, But Also for You,” *Forbes*, Nov. 28, 2021. [Online].

# Requirements analysis

## Requirements

Table : Functional Requirements

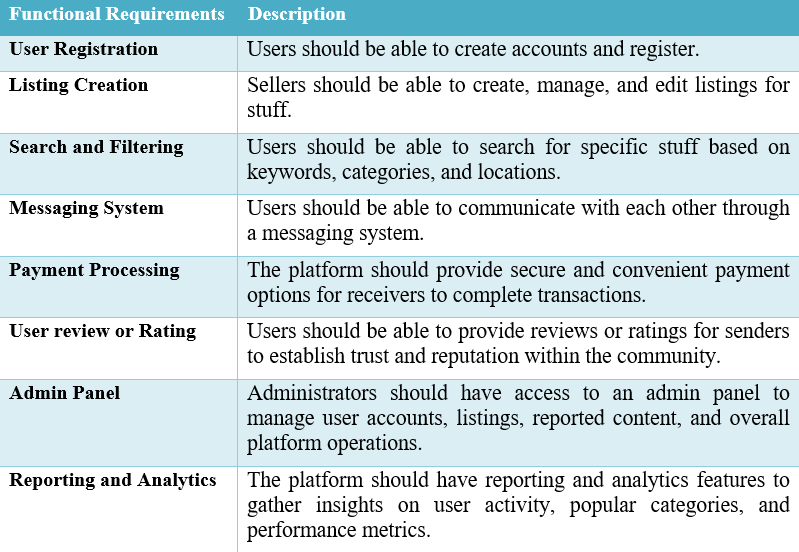
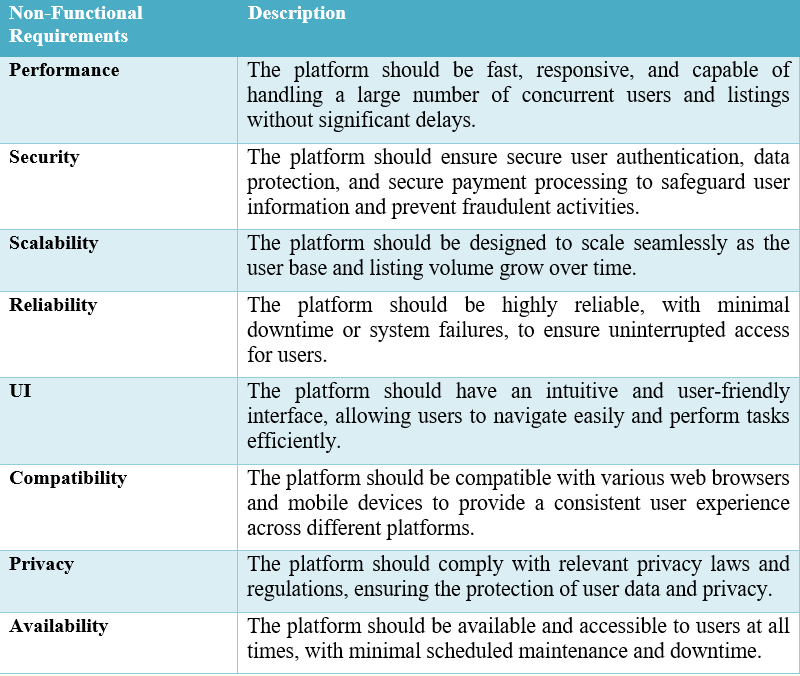


Table : Non-Functional Requirements



## List of Actors

* **User (Sender)**: The user who shares their unneeded possessions with others through the platform. They create listings, communicate with potential receivers, and manage the sharing process.
* **User (Receiver)**: The user who receives shared possessions from senders through the platform. They browse listings, express interest in Stuff, and communicate with senders to arrange pickups or deliveries.
* **Administrator**: The system administrator who oversees the platform's operations, manages user accounts, resolves disputes, and ensures compliance with platform policies and guidelines
* **Developer**: The software developer or development team responsible for designing, building, and maintaining the giveaway stuff platform.
* **Payment Gateway Provider**: The third-party service provider responsible for facilitating secure online payments between receivers and senders on the giveaway stuff platform

## List of use cases

1. **Sign Up**: Enable users to register by providing their credentials, such as to create an account in the system.
2. **Login**: Allow users to authenticate themselves by entering the correct credentials to access their account and the system's features.
3. **Select Stuff**: Users can browse and select deed stuff from the available listings.
4. **View Delivery Fee**: Display the delivery fee associated with the selected stuff.
5. **Update Delivery Fee**: Enable the admin to update the delivery fees of different Stuff based on various factors, ensuring accurate and up-to-date pricing information.
6. **Add New Stuff**: The user to add new Stuff to the system, including details such as title, description, category, and associated images.
7. **Delete Stuff**: Allow the admin to remove Stuff from the system.
8. **Give Ratings or Reviews**: Enable users to provide ratings or reviews based on their experience with the received stuff.
9. **Add to the Like List**: Allow users to add deed stuff to their like list for future reference.
10. **Make Payment**: Facilitate users' secure payments for the selected stuff using various payment methods, including debit cards, easy paisa, and jazz cash.
11. **Manage Users**: Grant the admin the capability to manage user accounts, including creating new accounts, modifying existing accounts, deleting accounts, and viewing user details.
12. **Manage Categories**: Enable the admin to manage categories within the system, including creating new categories, modifying existing categories, deleting categories, and viewing category information.
13. **Search Stuff**: Allow users to search for specific stuff by name or keywords, providing a convenient way to find deed Stuff quickly and efficiently within the system

## System use case diagram

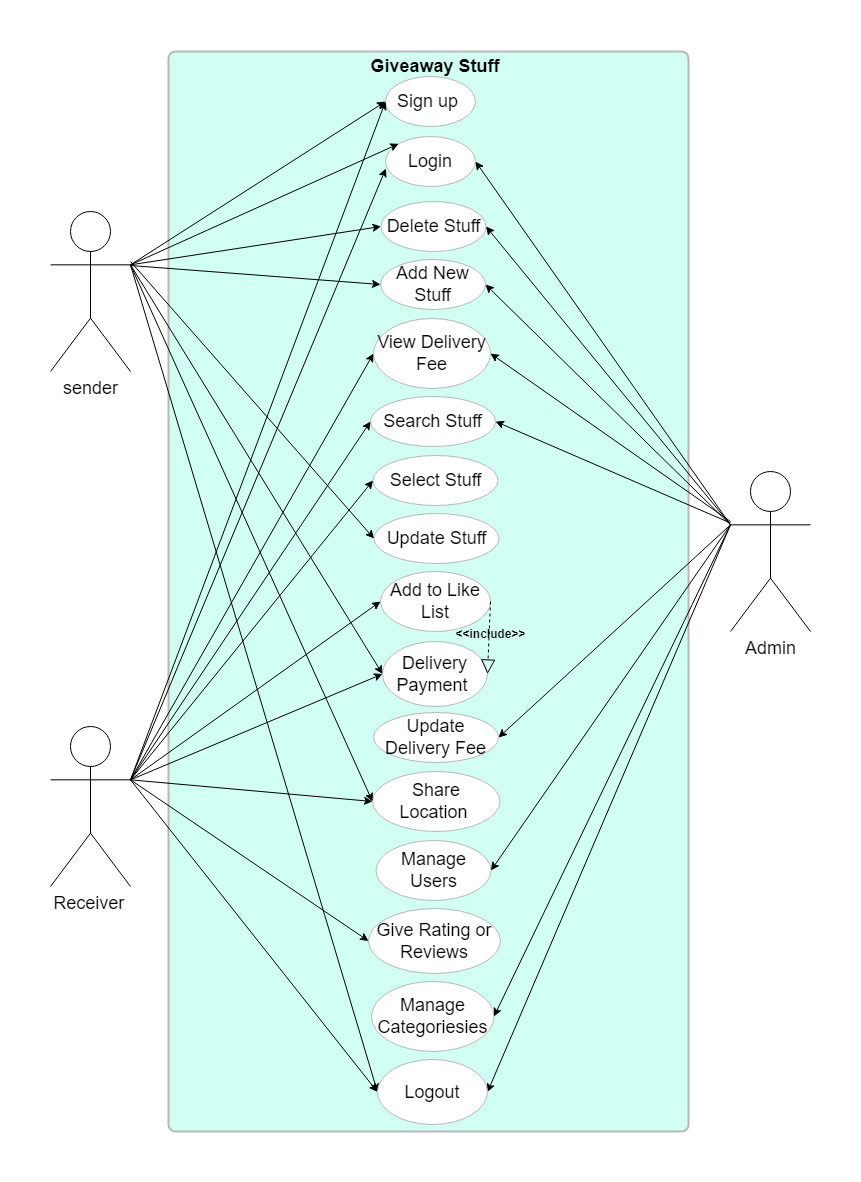


Figure : System Use Case Diagram

Explanation: Giveaway Stuff has three types of users: admin, sender, and receiver. Admin has Privileged access to things like management, maintenance, etc. And Sender and receiver have user access like view, select, etc.

## Extended use cases

Table : Sign up Use Case

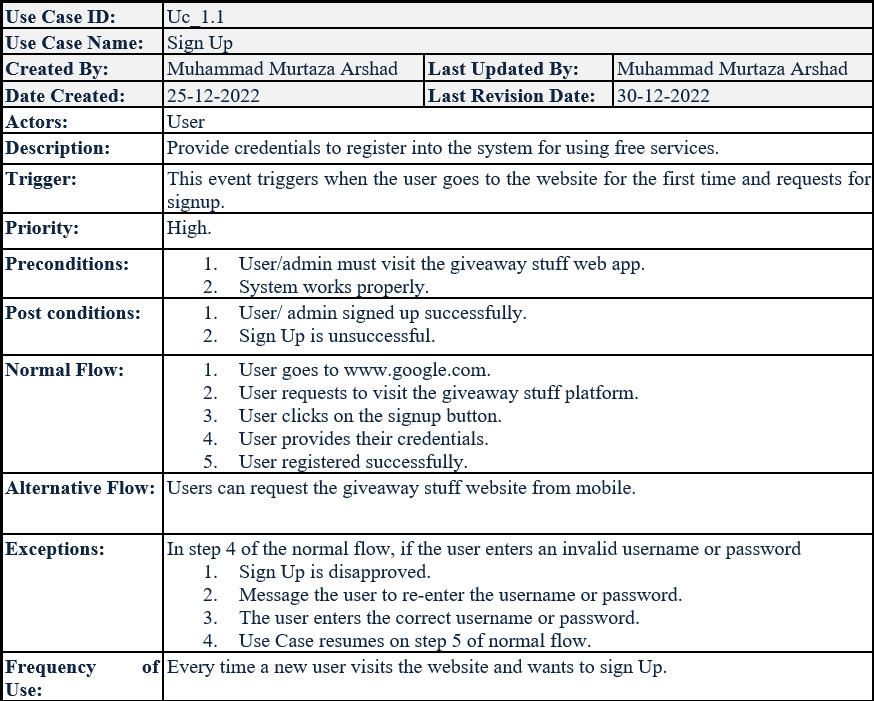


Table : Login Use Case



Table : Select Stuff Use Case

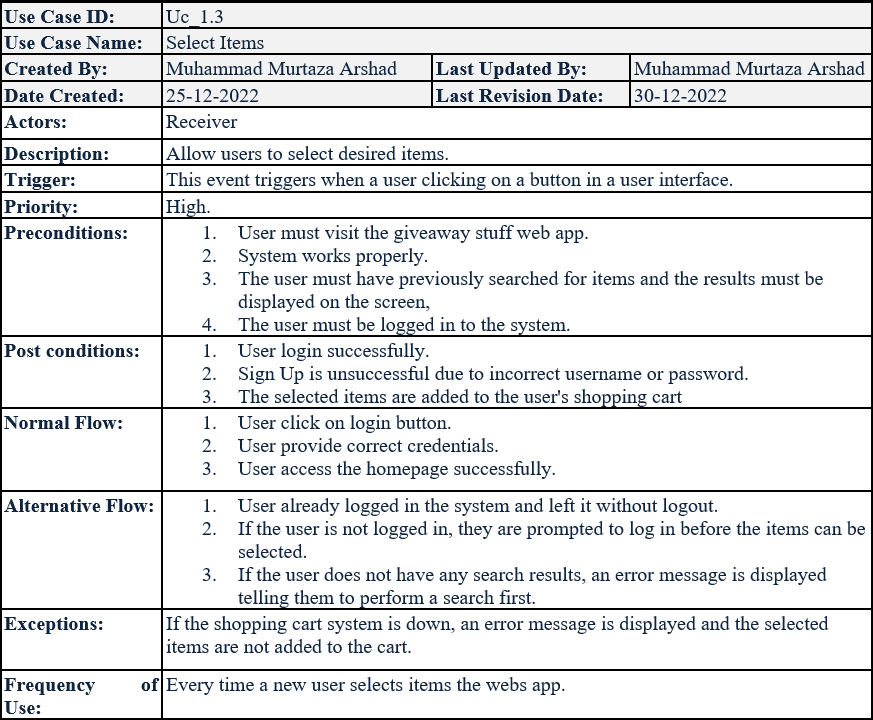


Table : View the Price Of Stuff Use Case



Table : Update The Price Of Stuff Use Case

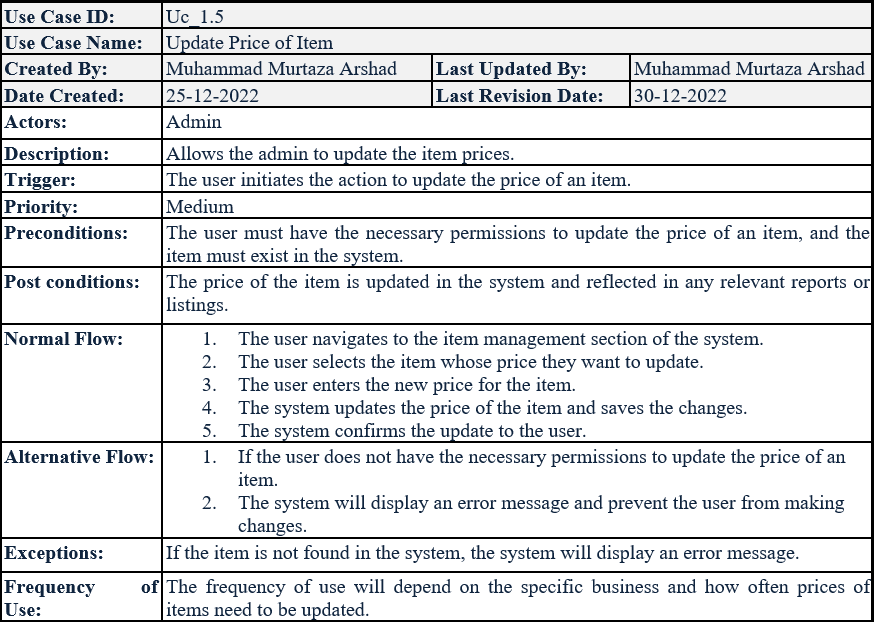


Table : Add New Stuff Use Case

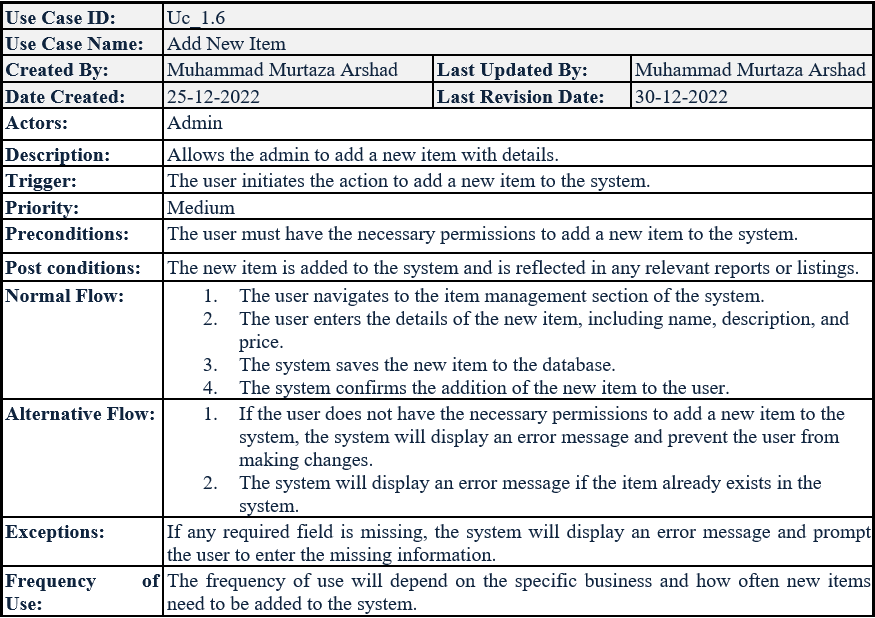


Table : Delete Stuff Use Case

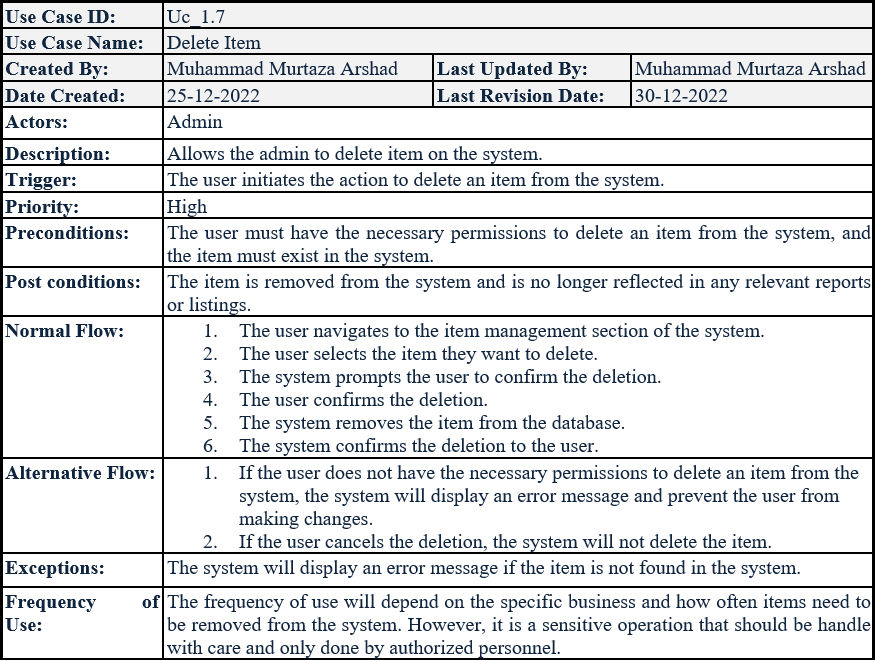


Table : Give Ratings And Reviews Use Case

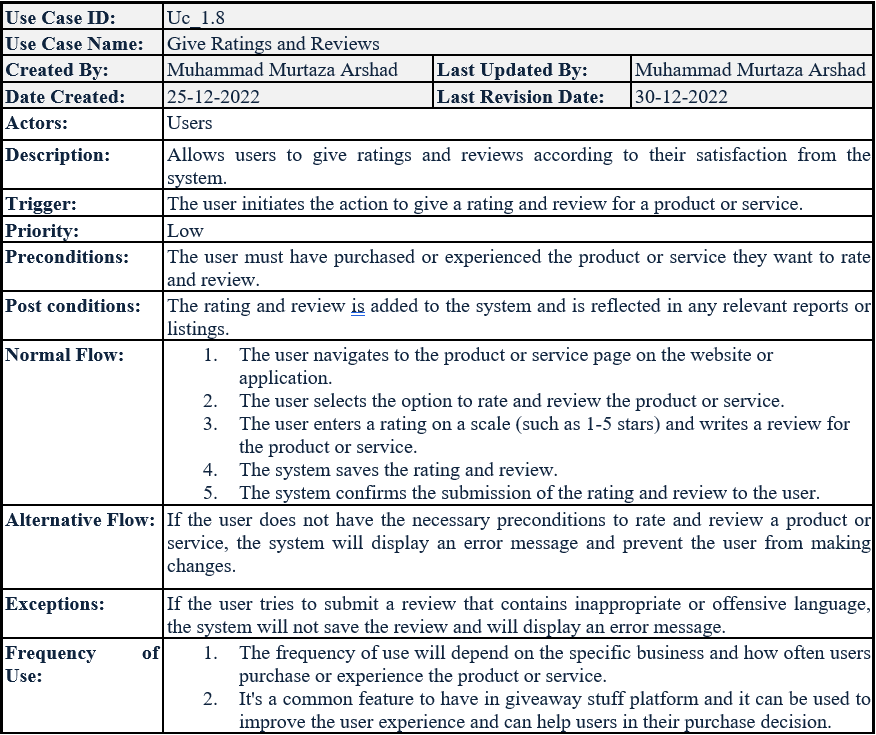


Table : Add to The Like List Use Case

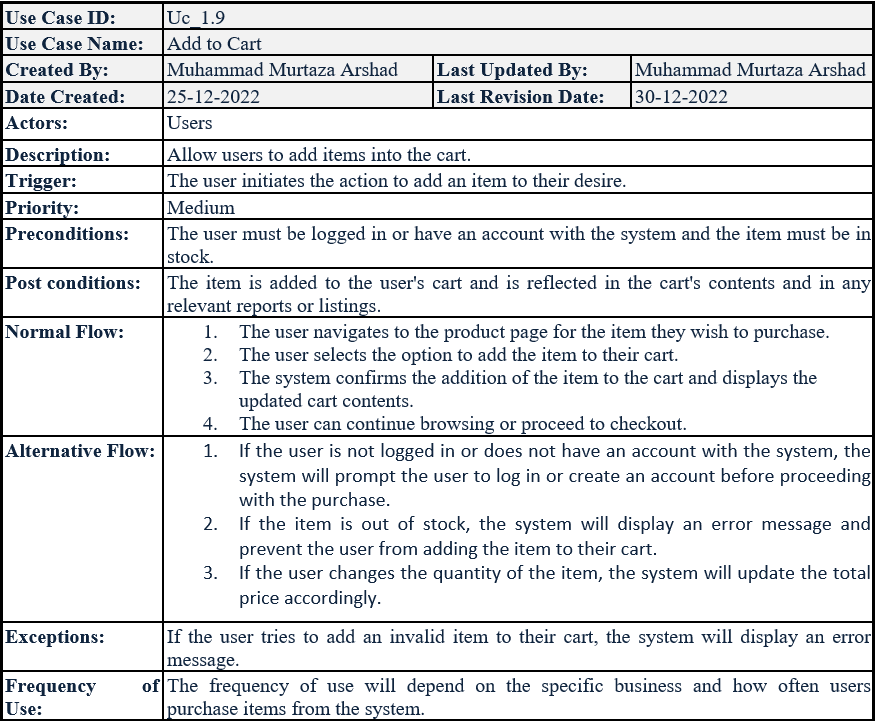


Table : Make Payment Use Case

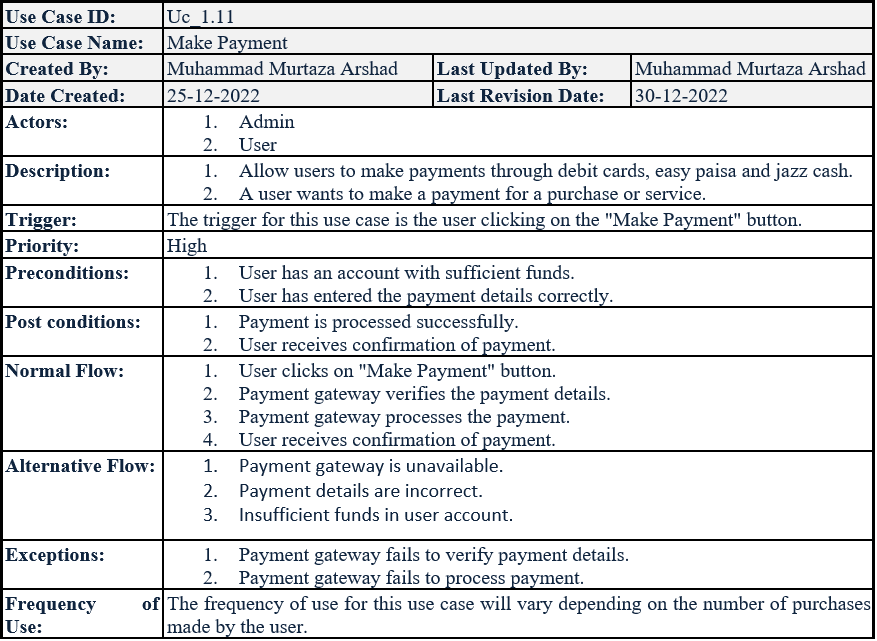


Table : Manage User Use Case

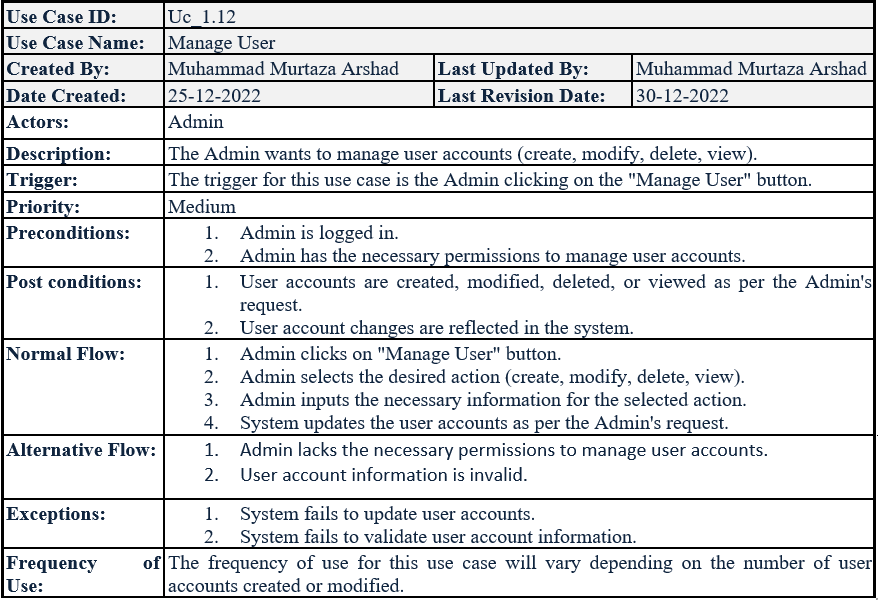


Table : Manage Categories Use Case

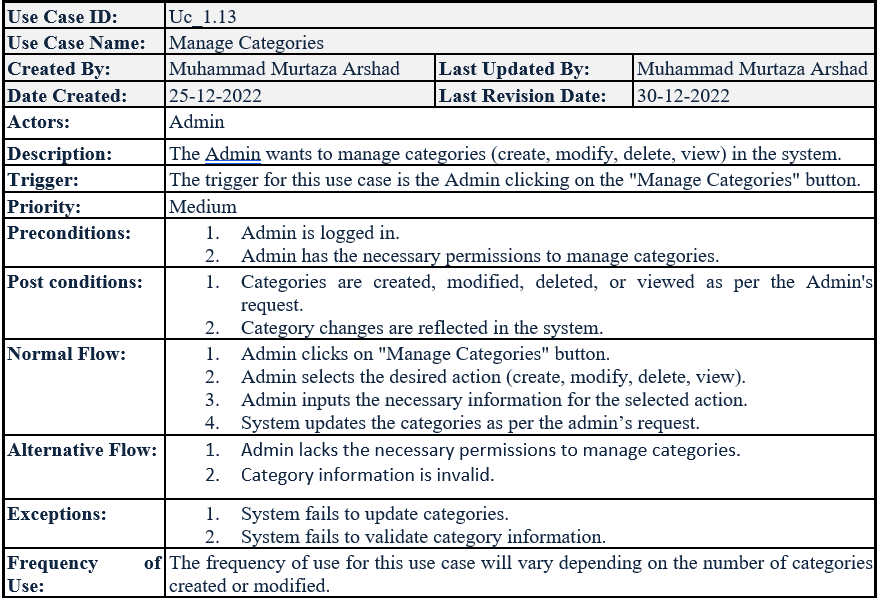
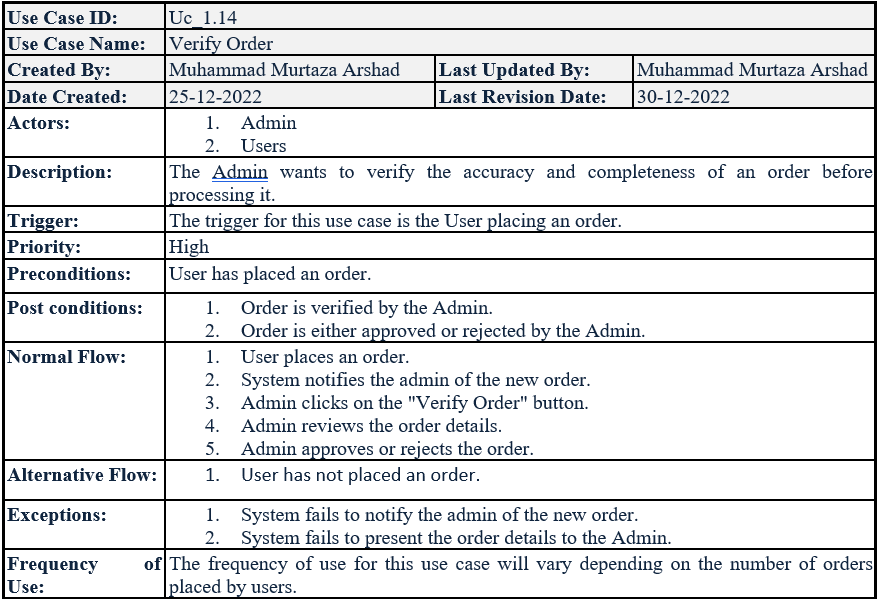
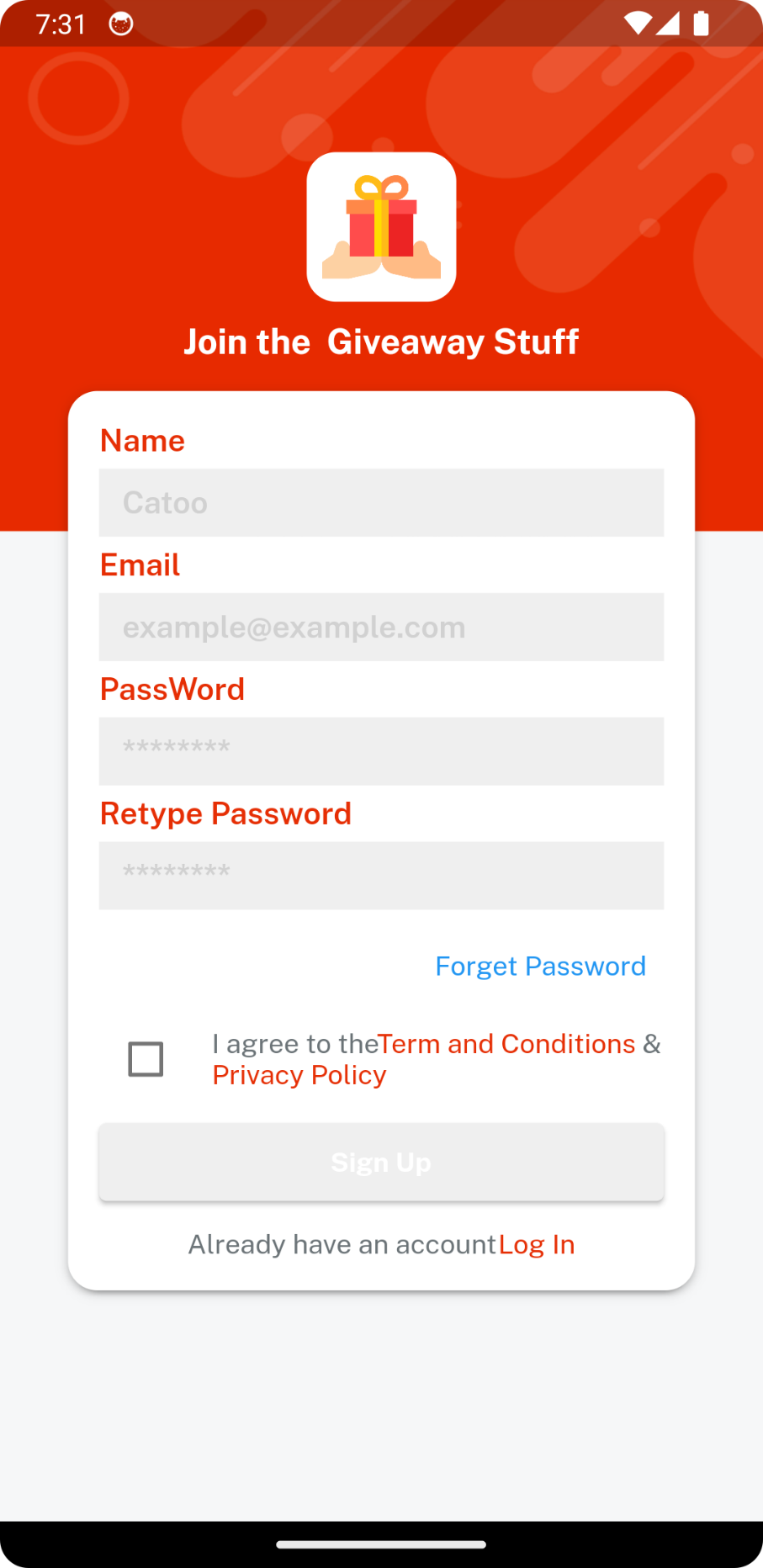


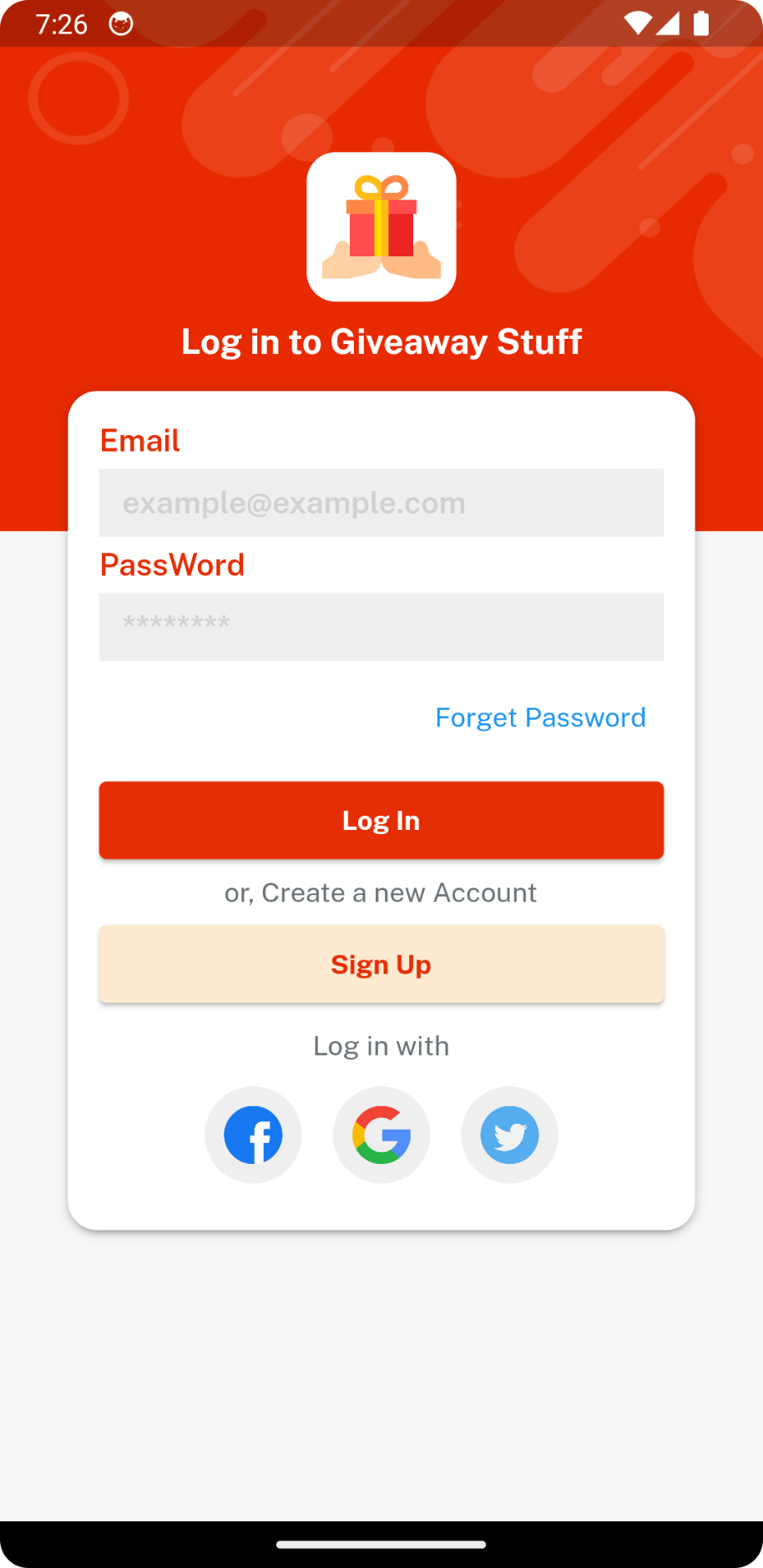
Table : Verify Request Use Case



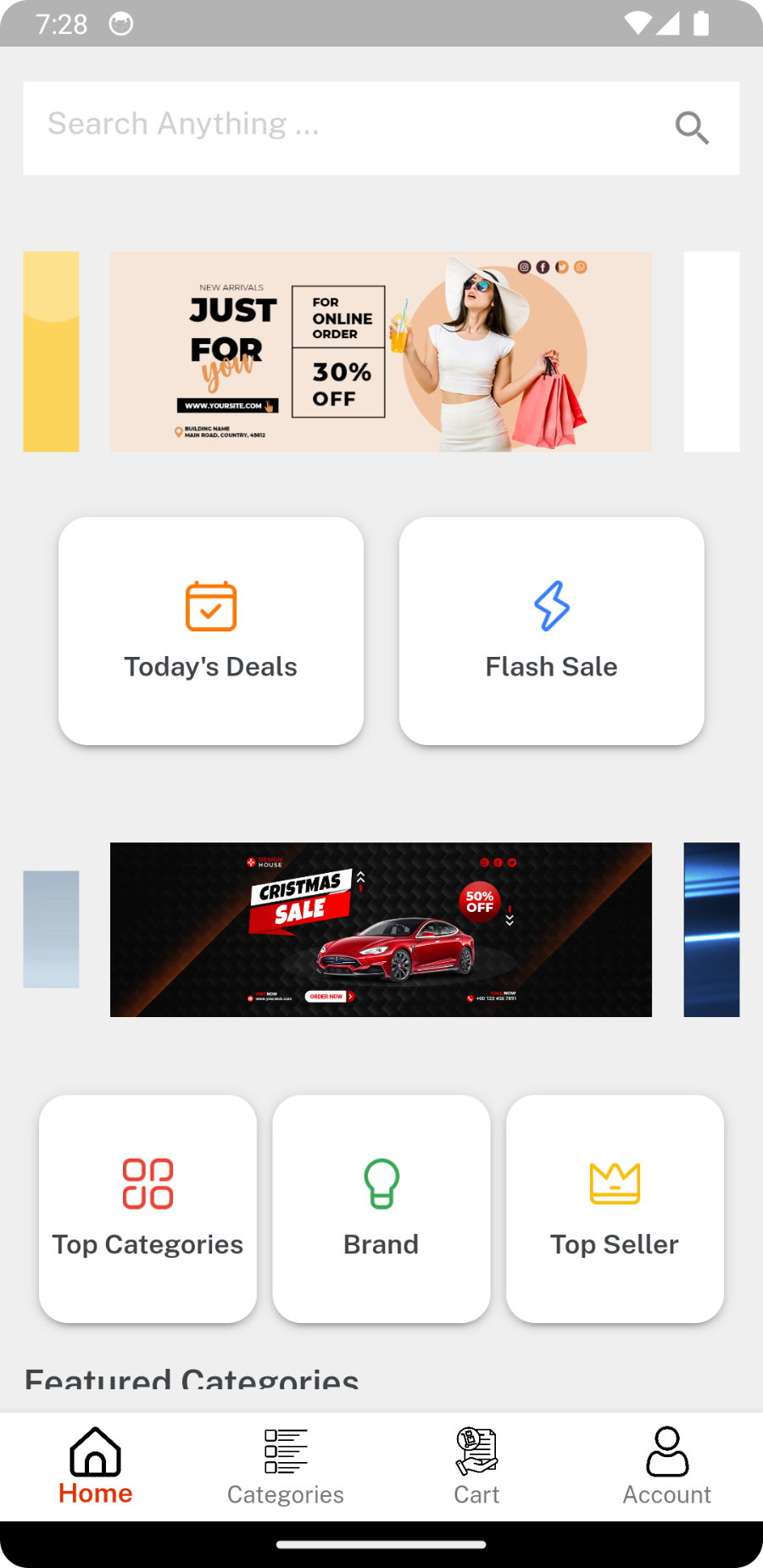
## User interfaces (mock screens)



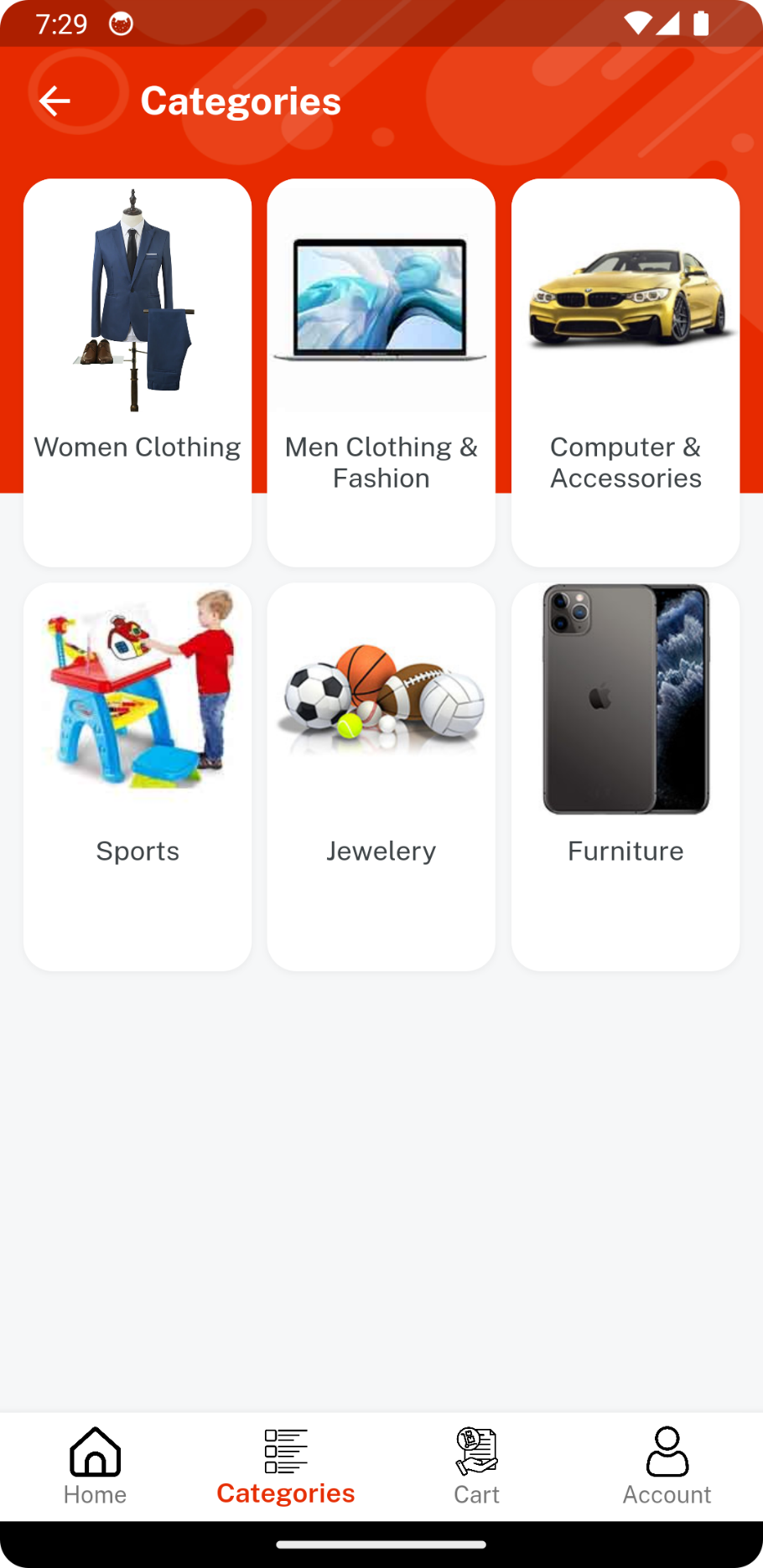
Prototype : Register A New Member



Prototype 2: Log in As A Member



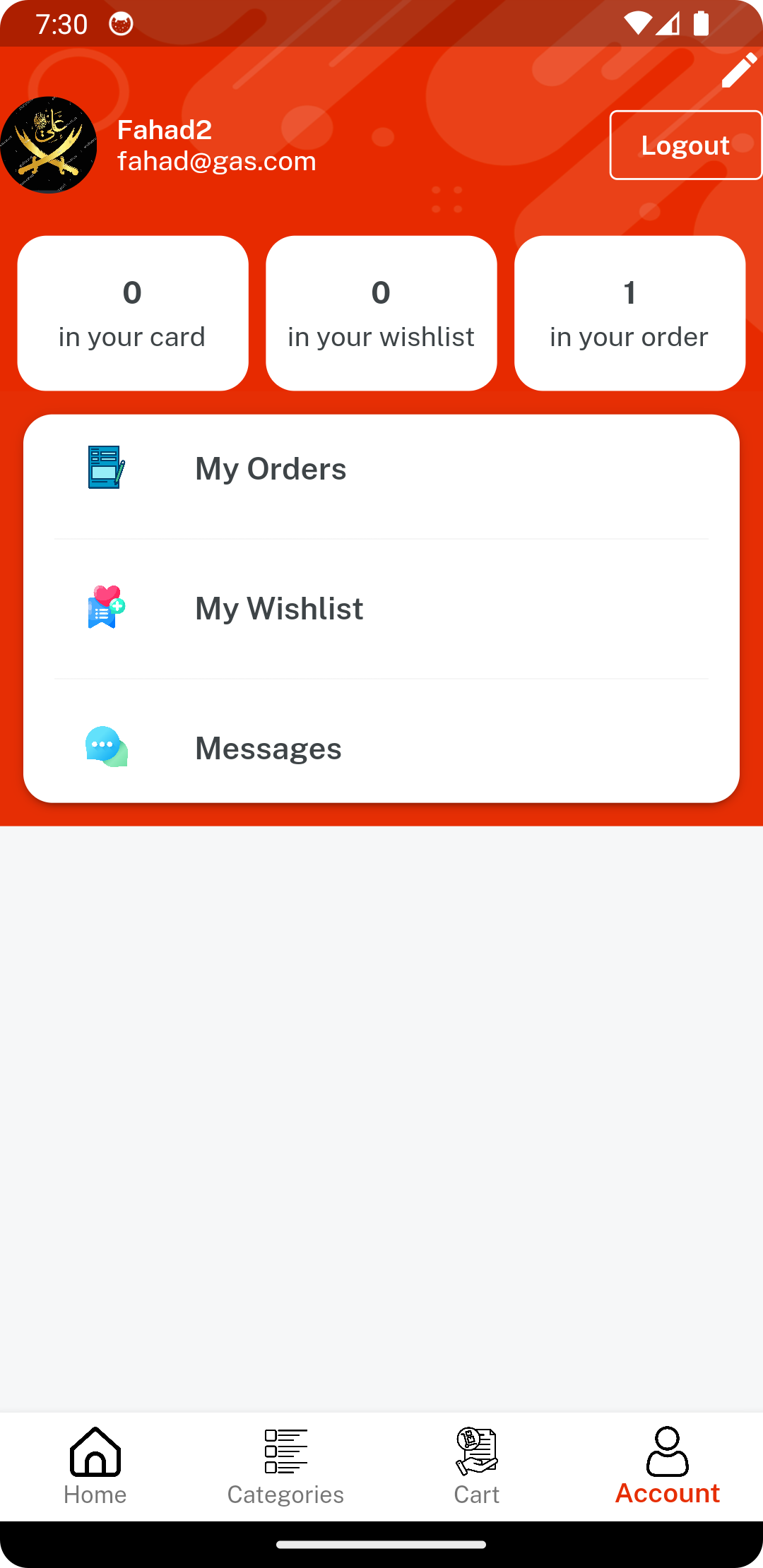
Prototype 3: Home



Prototype 4: Categories



Prototype 5: Cart



Prototype 6: Account

# Data flow diagram

## Data Flow Diagram Level 0

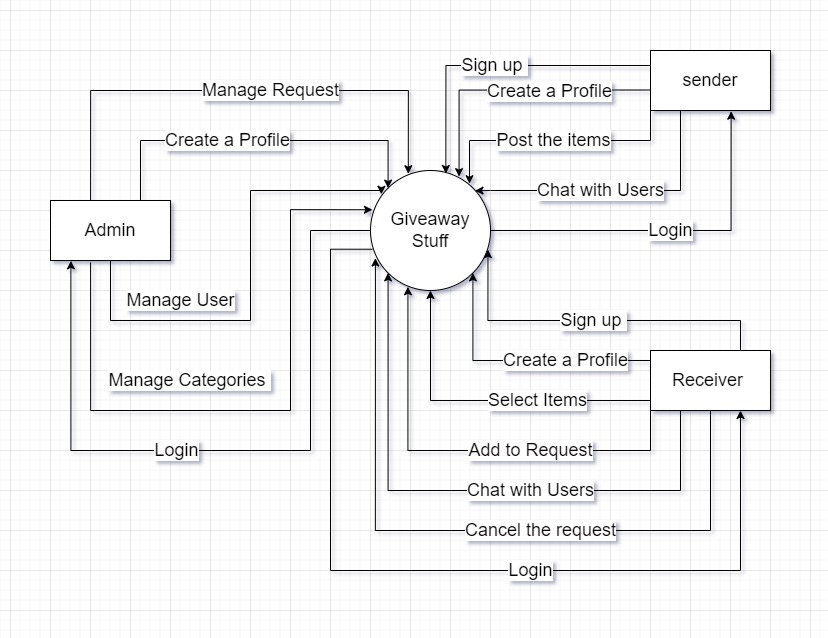


Figure : Data Flow Diagram Level 0

# System Design

## System Architecture Diagram

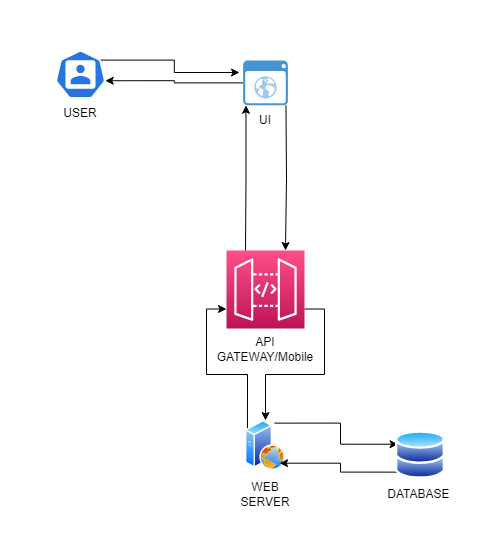


Figure : System Architecture Diagram

## Class Diagram

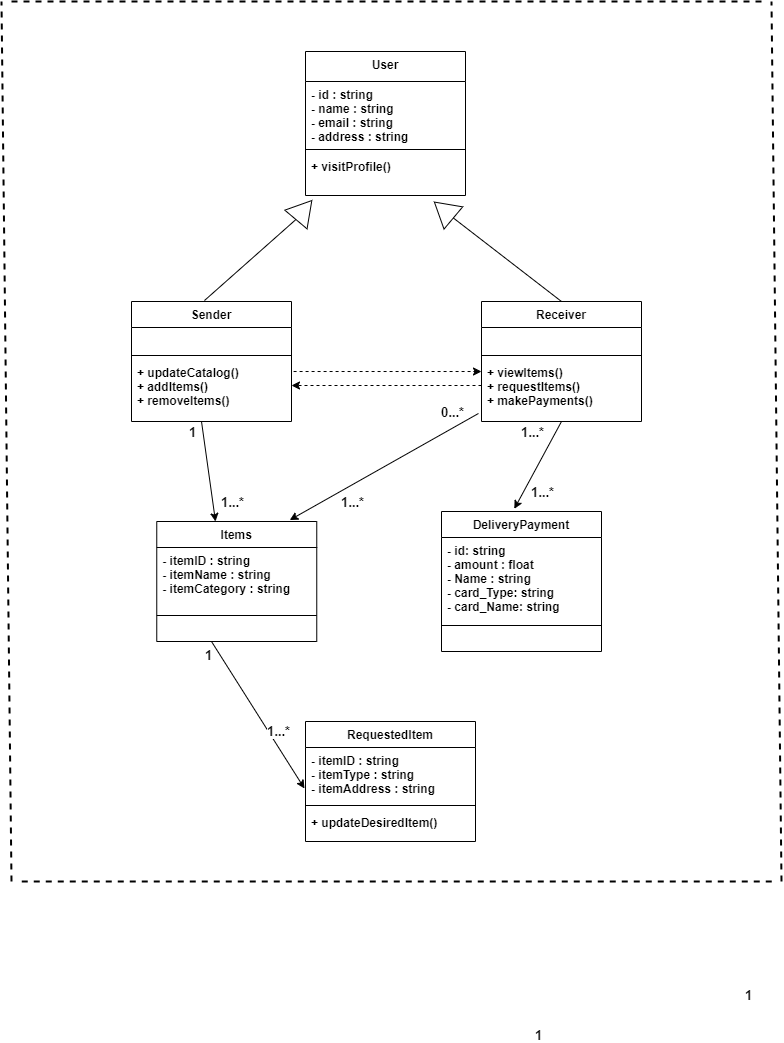


Figure : Class Diagram

## Sequence Diagrams

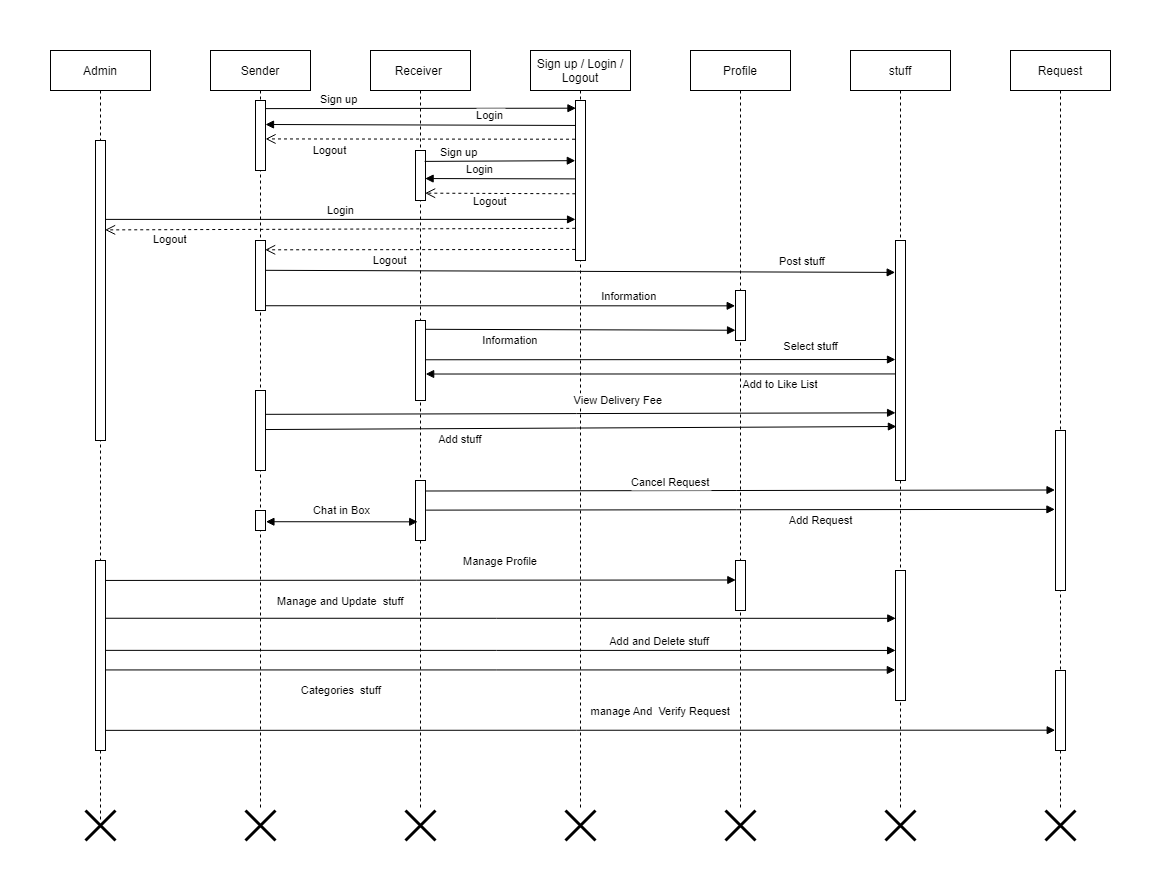


Figure : Sequence Diagrams

## Collaboration Diagrams

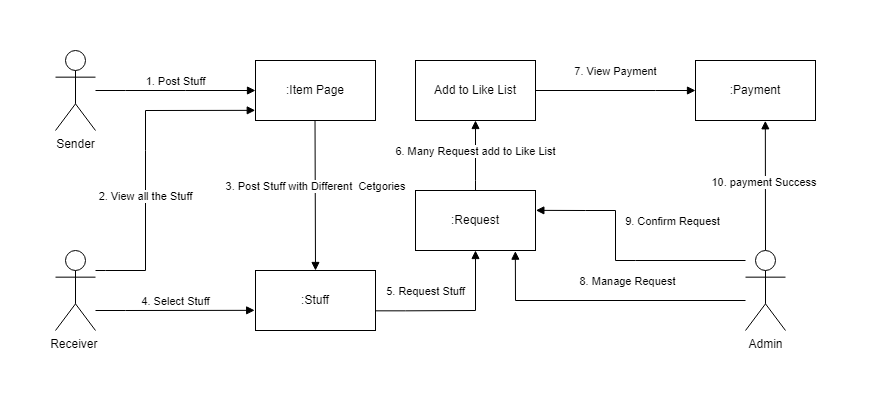


Figure : Sequence Diagrams

## ERD

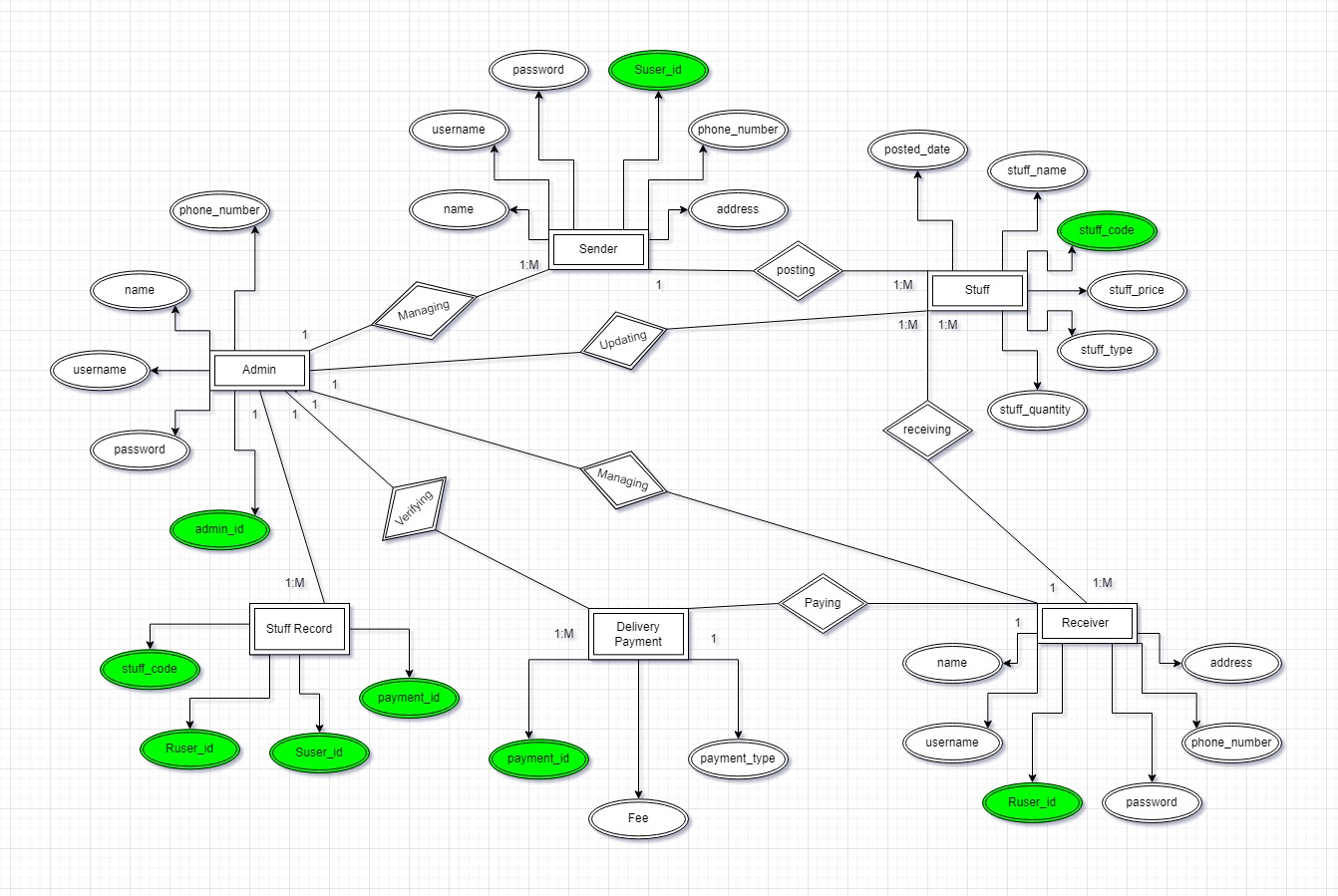
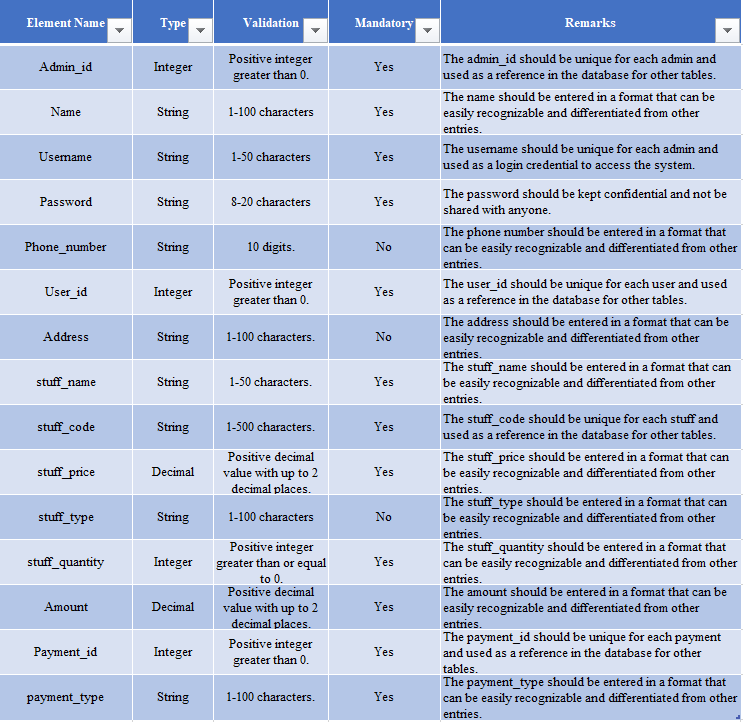


Figure : ERD

## Data Dictionary

Table : Data Dictionary



# Implementation details

## Development Setup

* **React.js** (for front-end development)
* **Android Studio** (for IDE)
* **Firebase DB** (for back-end development)
* **Flutter** (for cross-platforms)

## Deployment setup

To ensure scalability and accessibility, we opted to implement our Giveaway Stuff platform using a Firebase database. However, during the deployment process, we encountered several challenges related to user interface and database integration. To address these issues, we sought guidance from the project advisor, conducted extensive testing, and made necessary modifications to optimize performance and overcome the challenges at hand. Through this collaborative effort and iterative approach, we were able to resolve and fine-tune the user interface for enhanced usability and successfully integrate the Firebase database into our platform.

## Algorithms

* **Recommender System:** A recommender system algorithm is employed to suggest relevant stuff to users based on their preferences and browsing history.
* **Search algorithm:** The search algorithm enables users to search for specific stuff on the platform.
* **Authentication Protocol:** An authentication protocol algorithm is implemented to verify user credentials and prevent unauthorized access to the system.
* **Data Filtering:** The data filtering algorithm is utilized to filter stuff based on specific criteria, such as delivery fee, location, and category.

## Constraints

### Assumptions

* Users will adhere to the terms and conditions of the platform, ensuring a safe and fair environment.
* We assume that the client will provide authentication credentials.
* The internet will be available and reliable.
* The system will be thoroughly tested before its release.
* The database will be maintained by the admin.
* Users will provide feedback on their experience using the platform.
* Senders will accurately describe the stuff they are sharing.

### System constraints

* The system must ensure the secure protection of users' data.
* The system should accommodate a diverse range of stuff categories.
* The system must have a responsive and user-friendly interface.
* The platform should be compatible with different web browsers and mobile devices.
* The system must support multiple payment methods.
* The platform should provide efficient search and filtering functionalities for users to find relevant stuff.

### Restrictions

* Giveaway Stuff only allows users to list stuff in certain categories.
* Giveaway Stuff limits the amount of text that users can enter in the stuff description.
* Giveaway Stuff only allows users to upload images that are a certain size and format.
* Giveaway Stuff only allows users to list stuff for sale in their local area.
* The restricted access to specific features of the Giveaway Stuff platform for clients.

### Limitations

* **No Receiver Protection**: Giveaway Stuff does not provide receiver protection, meaning receivers are not guaranteed to receive their chosen stuff if unsatisfied. Users should exercise caution, thoroughly communicate with senders, and make informed decisions when using the platform.
* **No Sender Protection**: Giveaway Stuff does not offer sender protection, indicating that senders are not guaranteed the delivery of their stuff. Senders should exercise caution, ensure reliable communication with receivers, and establish mutually agreed-upon arrangements for the exchange.
* **Verification of Listing Accuracy**: The platform may not be able to independently verify the accuracy or condition of the stuff listed for giveaway. Users should rely on their judgment, carefully review item descriptions and images, and communicate with senders to ensure the legitimacy and quality of the stuff.
* **Physical Inspections or Guarantees**: The platform may not provide services for physically inspecting or guaranteeing the condition of the listed stuff. Users are responsible for conducting their inspections, negotiating terms, and assessing the suitability and condition of the items before finalizing any exchanges.

# Testing

## Extended Test Cases

Table : Login Test Case

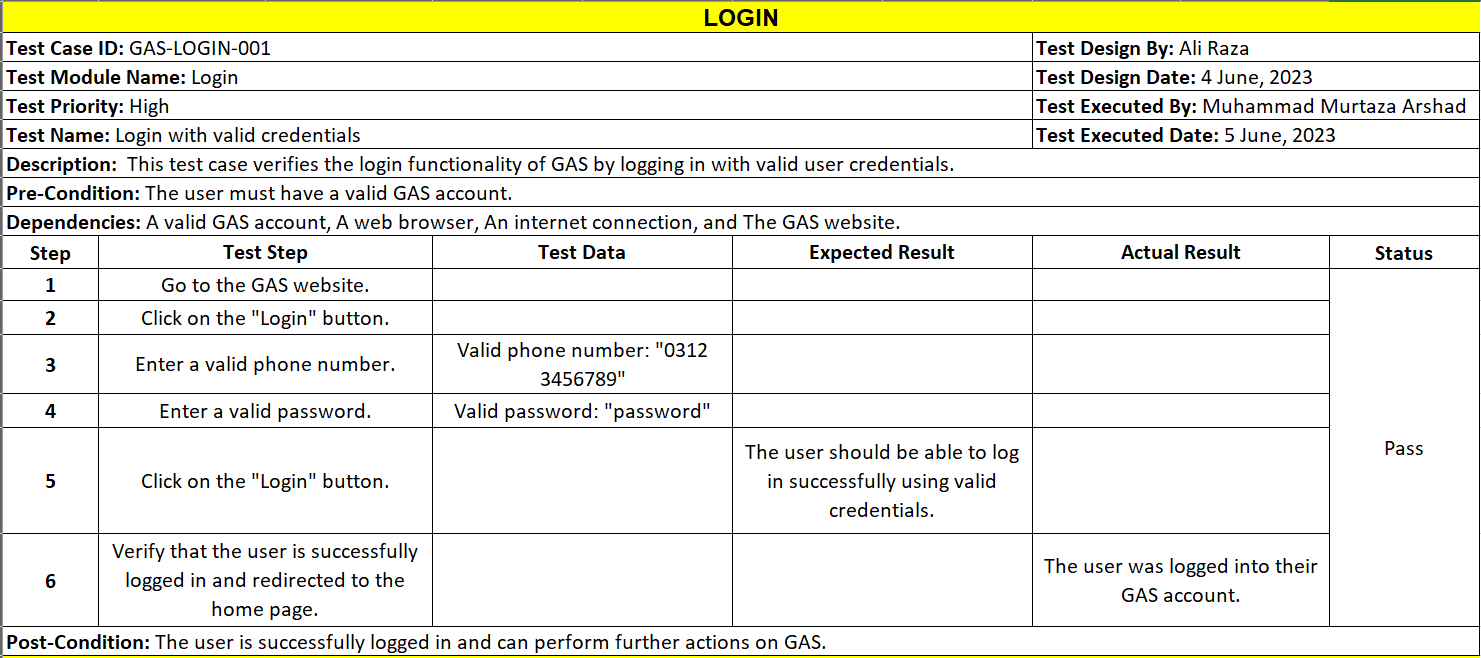
****

Table : Sign up Test Case

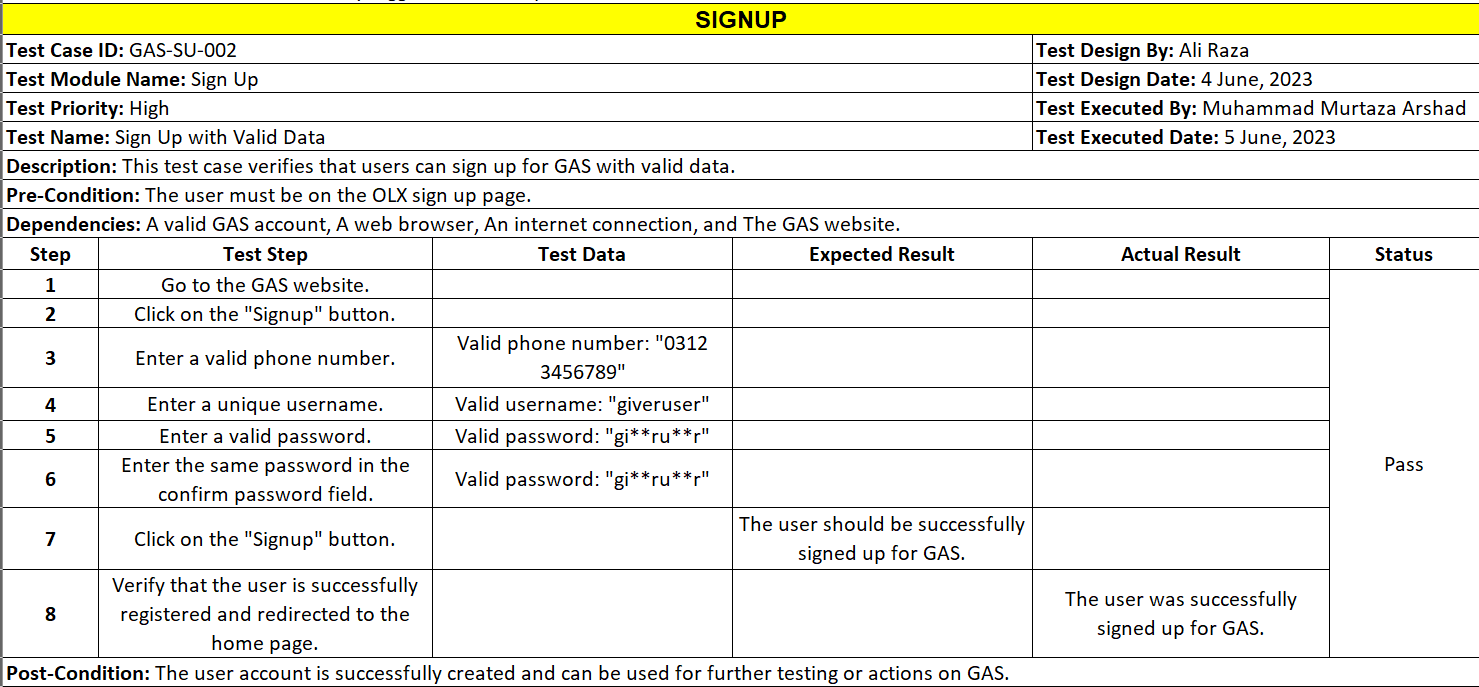
****

Table : Give Ratings and Reviews Test Case

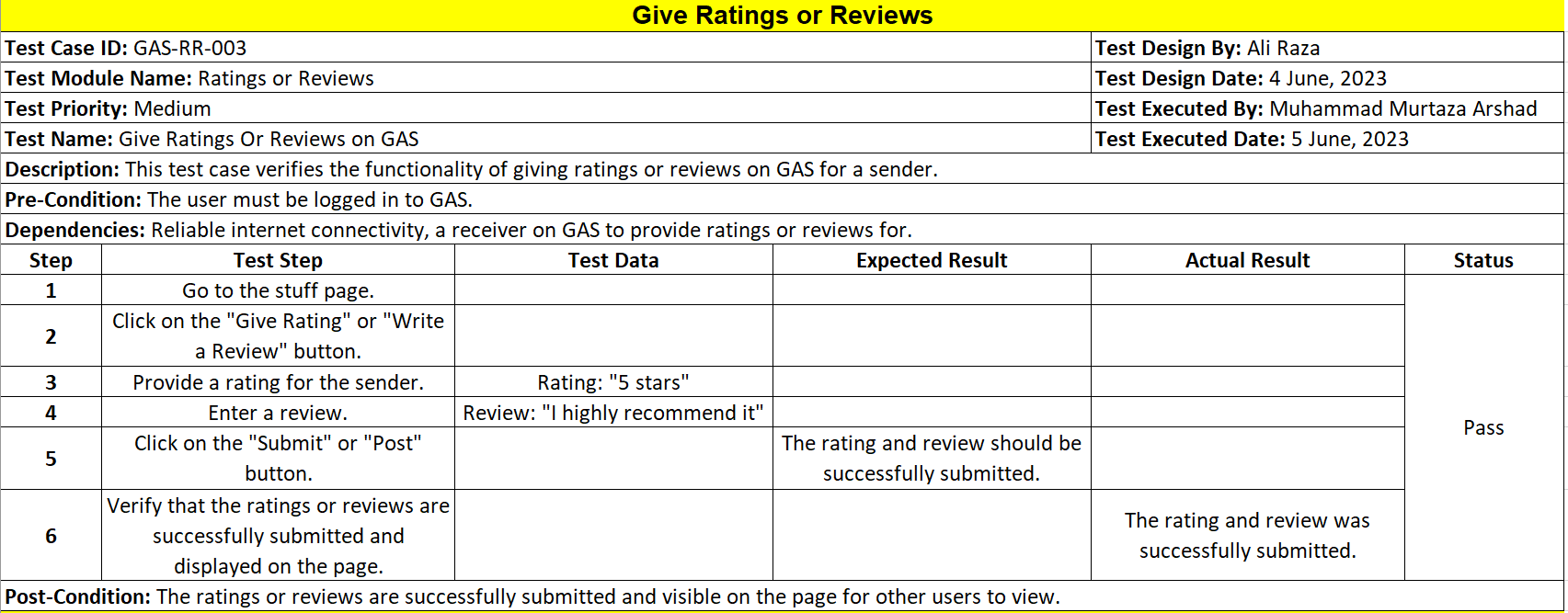
****

Table : Add to Like List Test Case

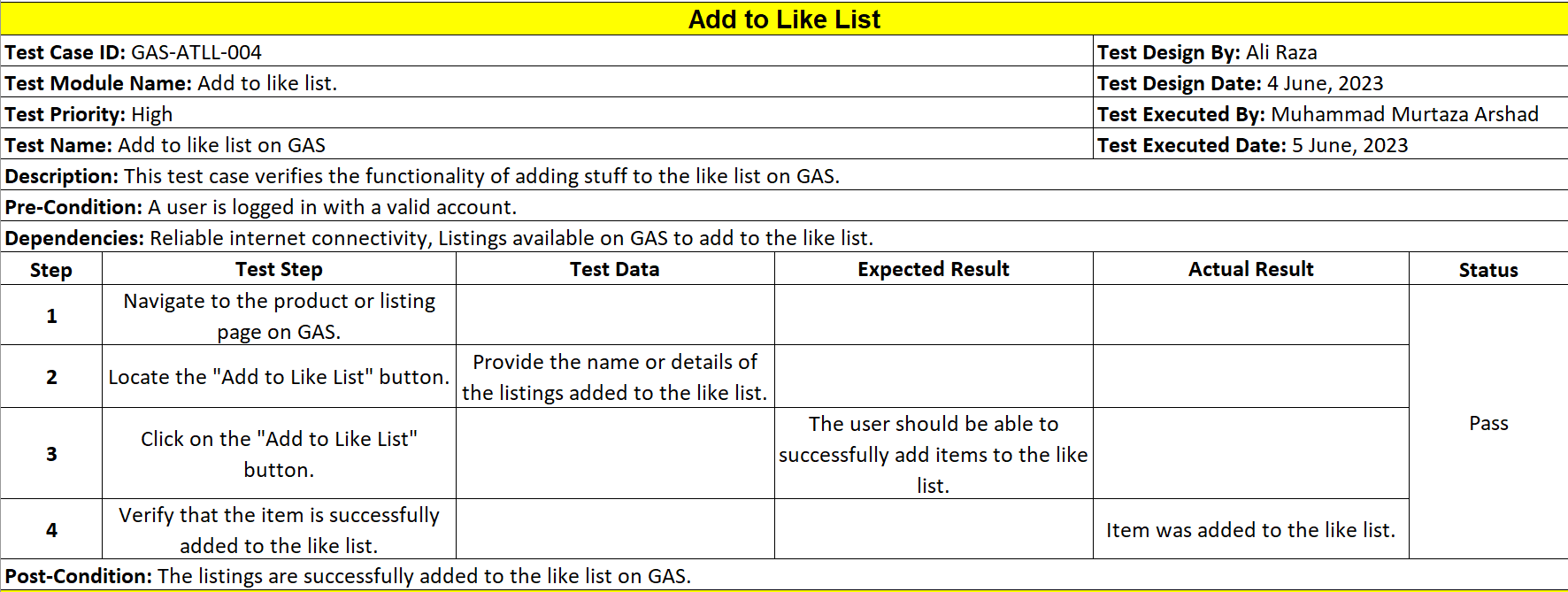
****

Table : Manage Categories Test Case

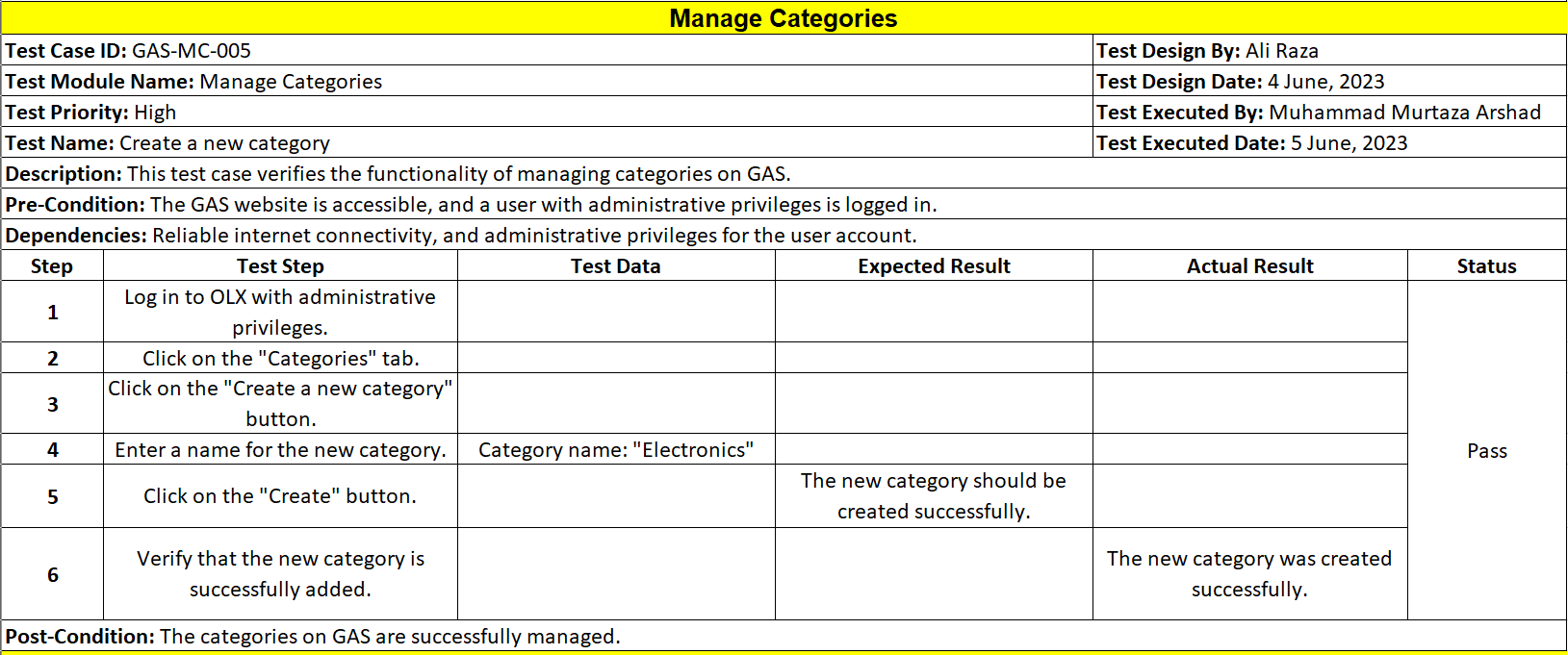
****

Table : Select Stuff Test Case

****

Table : Delivery Fee Test Case

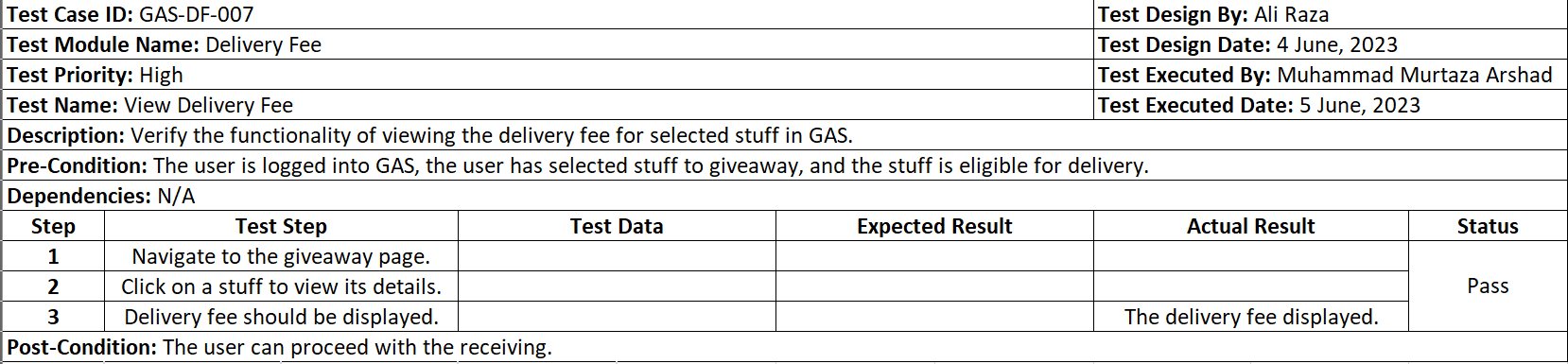
****

Table : Delivery Fee Update Test Case

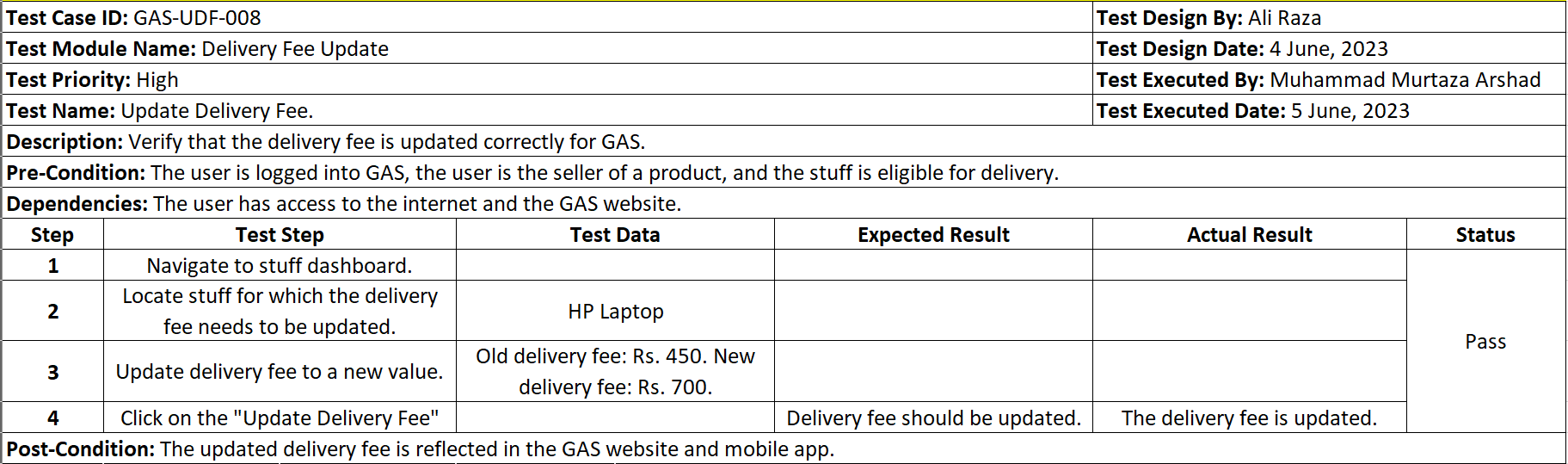
****

Table Add New Stuff Test Case

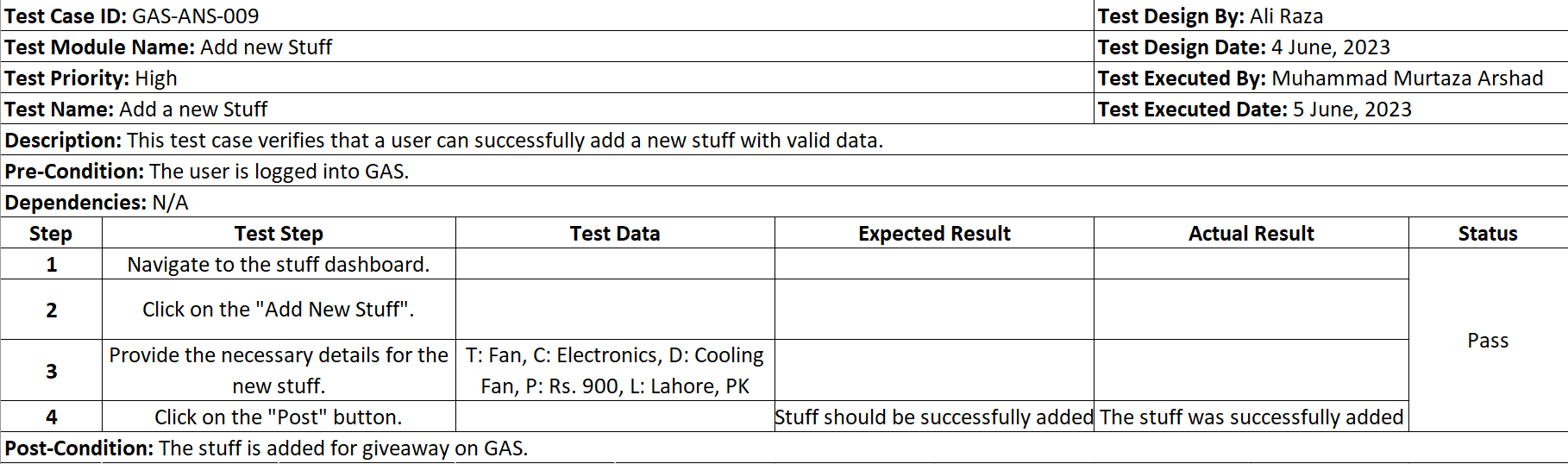
****

Table : Delete Stuff Test Case

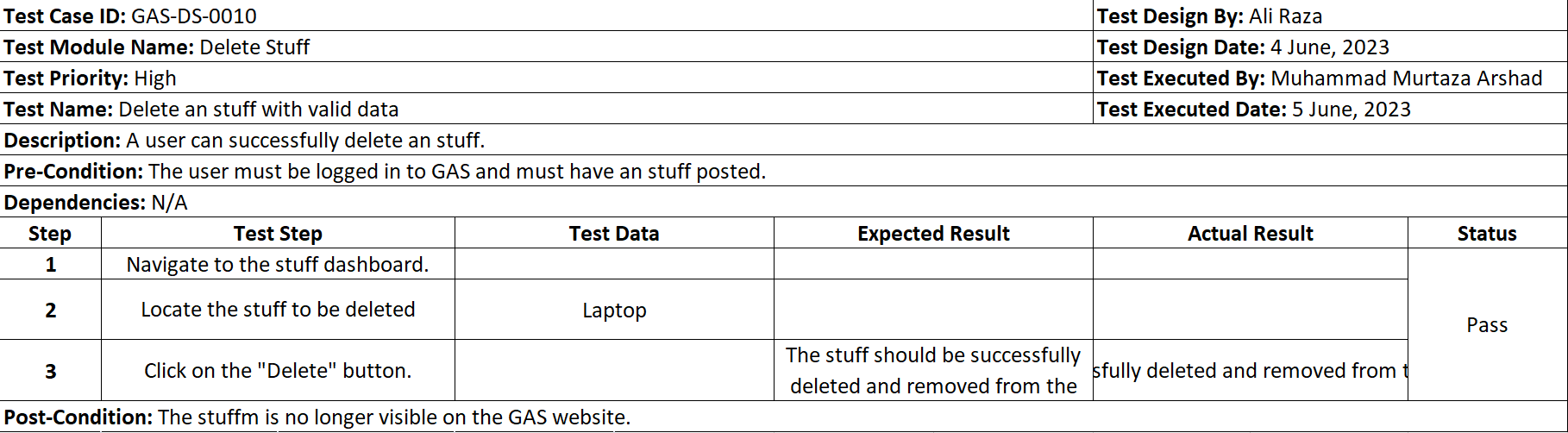
****

Table : Make Payment Test Case

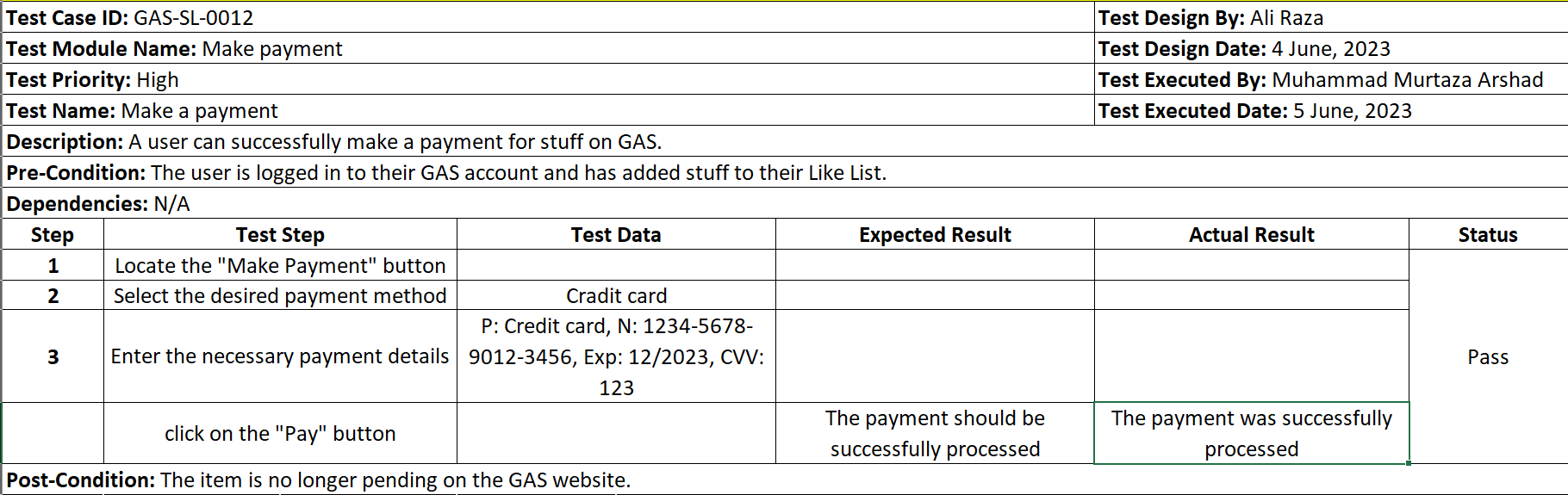
****

Table : Manage User Test Case

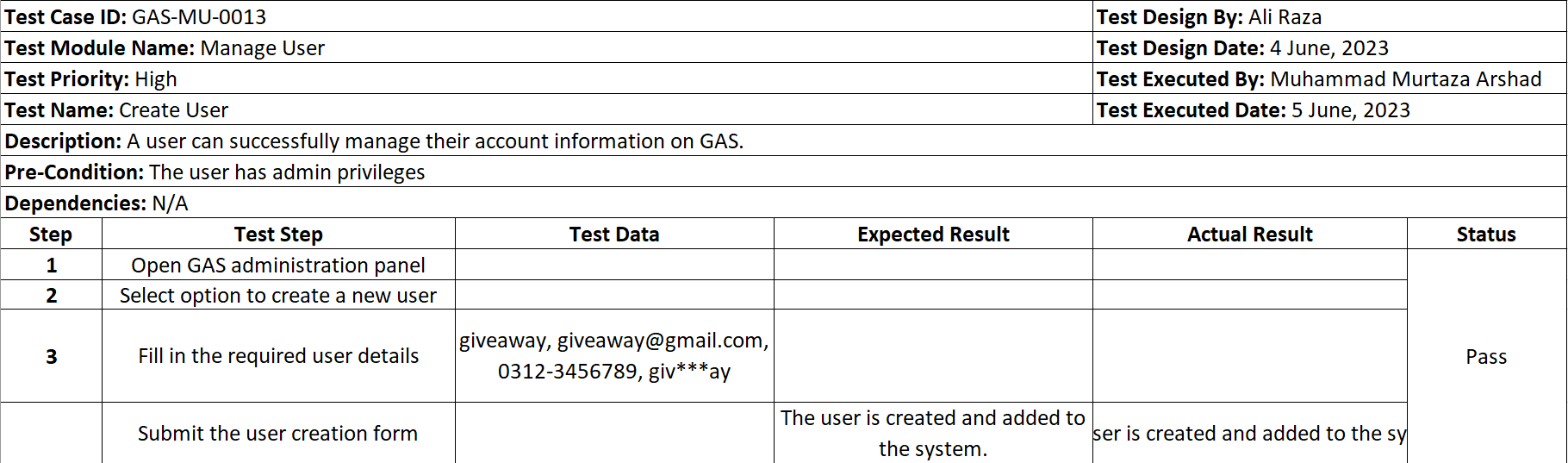
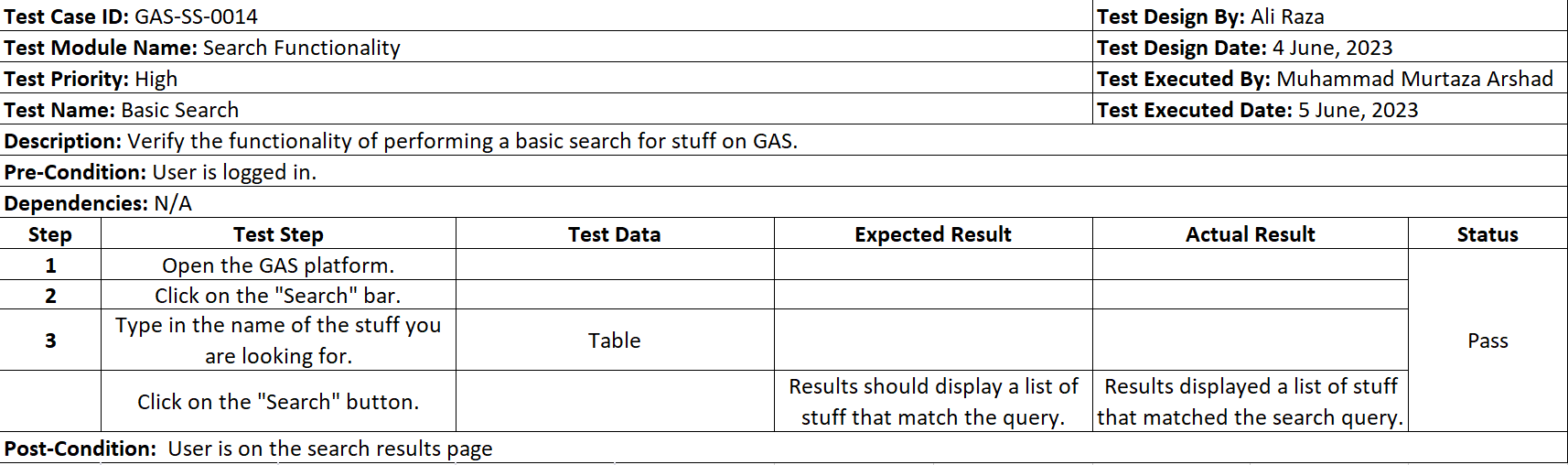
****

Table : Search Test Case

****

## Decision Table

### Code snippet



Figure : Authentication Protocol Code Snippet

### Decision coverage table

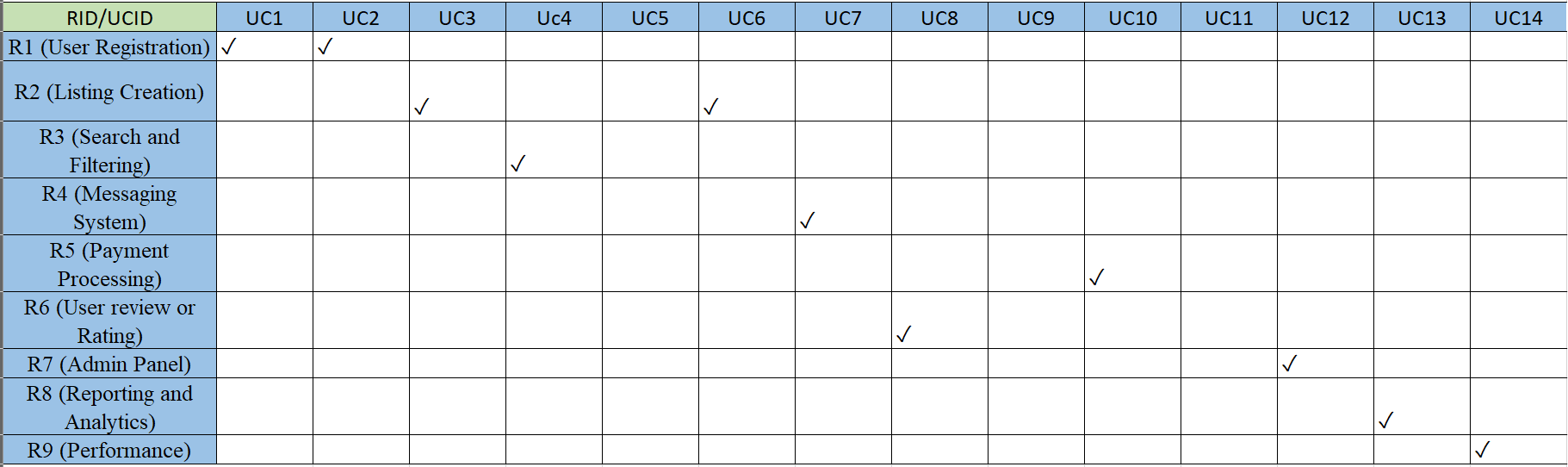
Table : Decision coverage table



## Traceability Matrix

### RID vs UCID

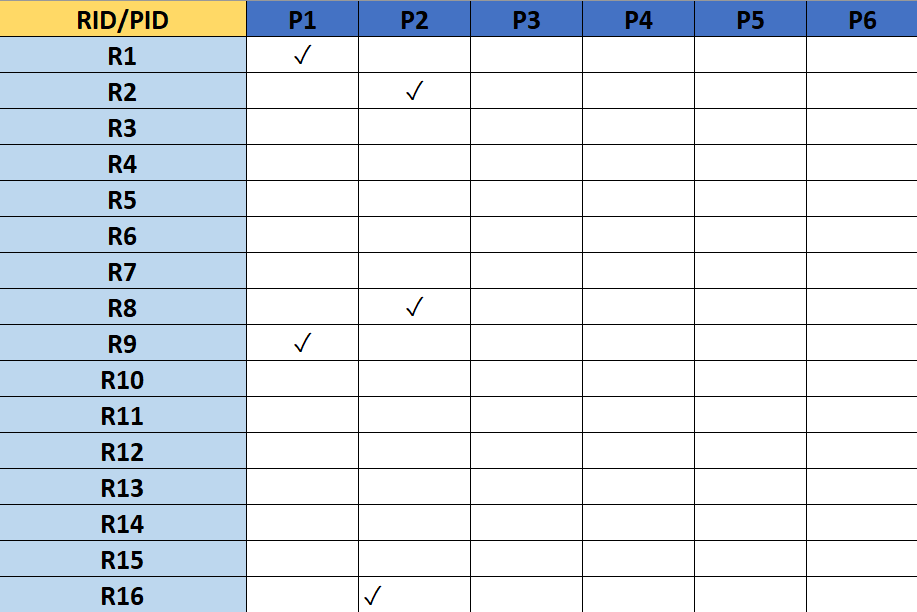
Table : RID vs UCID Traceability Matrix



The tick mark (✓) indicates the relationship between specific requirements (R) and corresponding use cases (UC). Each requirement is associated with a particular use case that addresses the functionality described by the requirement. The matrix assumes a one-to-one relationship, meaning each requirement is fulfilled by one use case, and vice versa.

### Prototypes (RID vs PID)

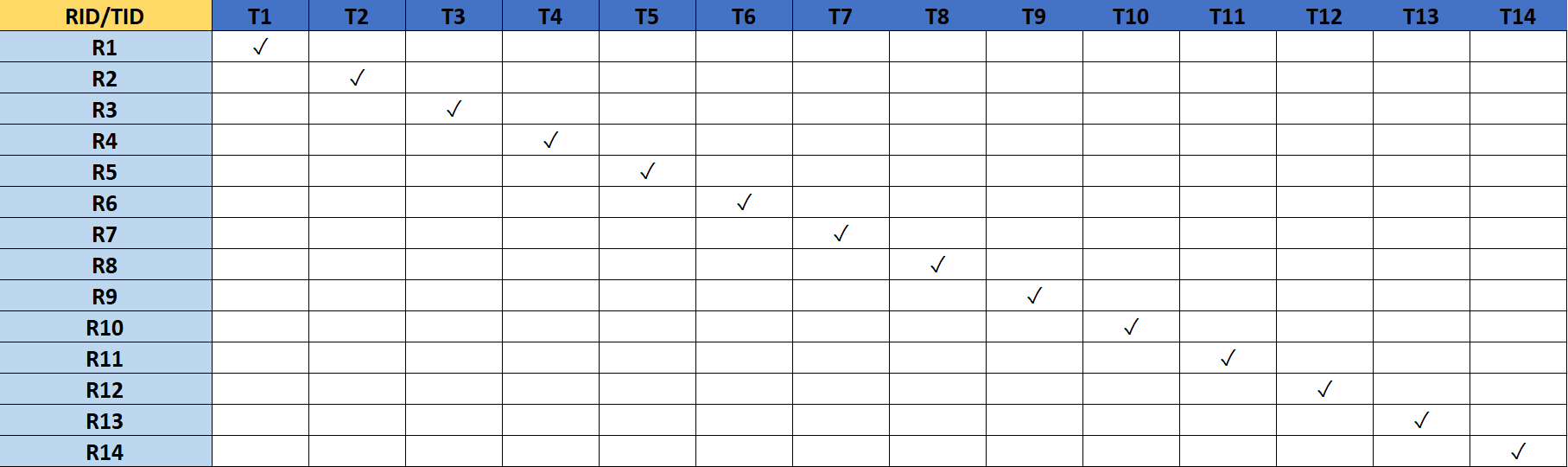
Table : Prototypes (RID vs PID) Traceability Matrix



The tick mark (✓) indicates the relationship between specific requirements (R) and corresponding prototypes (P). Each requirement is associated with a particular prototype that addresses the functionality described by the requirement. The matrix assumes a one-to-one relationship, meaning each requirement is fulfilled by one prototype, and vice versa. The traceability matrix helps ensure that the prototypes adequately cover all requirements during the development process, providing a comprehensive view of the requirements-to-prototypes mapping.

### Test Cases (RID vs TID)

Table : Test Case (RID vs TID) Traceability Matrix



The tick mark (✓) indicates the relationship between requirements (R) and their corresponding test cases (T). Each requirement is associated with at least one test case that verifies its implementation. The traceability matrix helps track and ensure the coverage between requirements and test cases, ensuring that each requirement has an associated test case for verification and validation purposes during the testing process.

# Results/Output/Statistics

## %completion

Total Requirements: 16

Requirements Fulfilled: 13

Completion Percentage: (13 / 16) \* 100 = 81%

## %accuracy

Total Test Cases: 14

Requirements Fulfilled: 12

Accuracy Percentage: (12 / 14) \* 100 = 86%

## %correctness

Total Test Cases: 14

Coverage Fulfilled: 12

Correctness Percentage: (12 / 14) \* 100 = 86%

# Conclusion

The Giveaway Stuff platform has been successfully developed and implemented, providing users with a convenient way to share unneeded items with others. The platform's features, including user registration, stuff selection, and rating functionality, offer a seamless experience for senders and receivers. While there are certain limitations, such as the absence of receiver and sender protection, users are advised to exercise caution and make informed decisions. Going forward, we will continue to enhance the platform based on user feedback and strive to create a sustainable and engaging sharing community. We express our gratitude to all stakeholders involved in the project's success.

# Future work

In the future, We build Admin Dashboard App. Giveaway Stuff aims to improve the user experience and cater to emerging needs through several initiatives. These include enhancing search and recommendation algorithms to deliver more accurate and personalized results for users. Strengthening security measures and implementing advanced fraud detection mechanisms will ensure a safe and trustworthy platform. Expanding to new regions and languages will allow for a broader user base and increased participation. Introducing social features and community-building elements will foster engagement and build trust among users. Incorporating user feedback and continuously refining the platform based on user needs will drive its evolution. Lastly, ensuring accessibility and inclusivity will promote equal opportunities and provide a seamless experience for all users. These future endeavors will contribute to the growth and development of Giveaway Stuff, enabling it to thrive in the dynamic online space and meet the evolving demands of its users.

# Bibliography

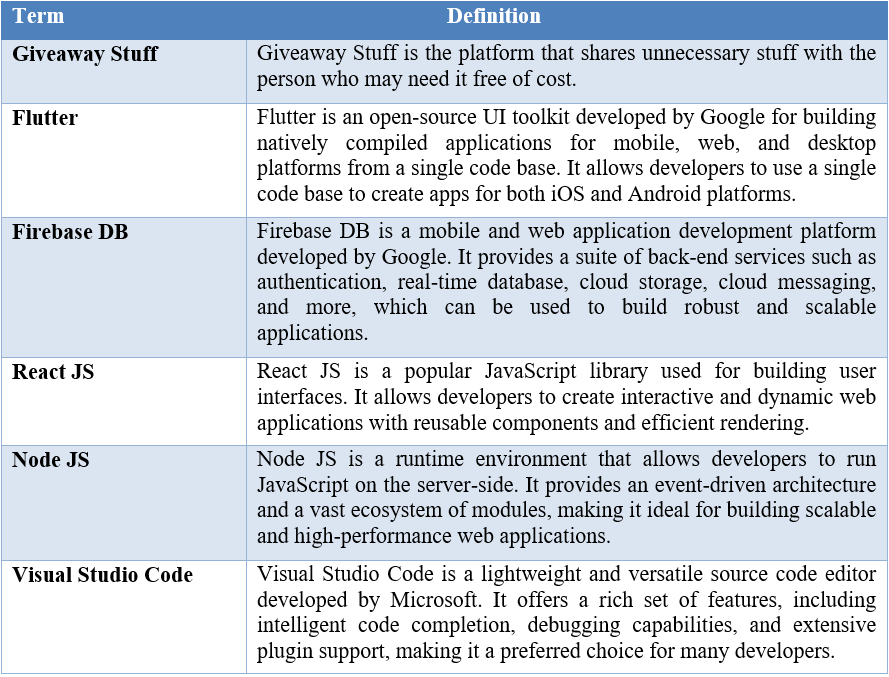
## Books/Articles

* Jamal and H. McKinnon, *The Power of Giving: How Giving Back Enriches Us All*. Penguin, 2008.
* J. Santi, *The Giving Way to Happiness: Stories and Science Behind the Life-Changing Power of Giving*. Penguin, 2015.
* Asada and A. Arai, “*Post-Event Effects: A New Perspective on Giveaway Stuff Distributed at Sporting Events*,” *Journal of Global Sport Management*, vol. 8, no. 1, pp. 281–301, Sep. 2020, doi: 10.1080/24704067.2020.1819618.

# Appendix

## Glossary of terms

Table : Glossary Of Terms



## Pre-requisites

1. **Development/Deployment Setup:**

* **Flutter SDK**: The Flutter software development kit (SDK) needs to be installed on your development machine. Visit the official Flutter website for installation instructions tailored to your operating system.
* **Integrated Development Environment (IDE)**: Choose an IDE for Flutter development, such as Android Studio, and ensure that you have installed the necessary Flutter and Dart plugins to enable a seamless development experience.
* **Firebase Account**: Create an account on Firebase by visiting <https://firebase.google.com/> and setting up a new project specifically for your Giveaway Stuff application. Firebase provides various services that can enhance the functionality and backend of your application.

**Giveaway Stuff Account**: Create an account on the Giveaway Stuff platform. This account will grant you access to their developer resources and APIs, enabling you to utilize their platform-specific features and functionality for your application.

1. **External System Dependencies:**

* **Flutter Packages**: Enhance your application's functionality by leveraging relevant Flutter packages. These packages provide additional features and utilities that can be easily integrated into your Flutter project. Depending on your app's requirements, you may need packages for image loading, state management, networking, UI components, and more.
* **APIs**: Familiarize yourself with the giveaway stuff APIs and documentation to understand how to interact with their platform. This includes authentication mechanisms, data retrieval, listing management, and other API endpoints provided by giveaway stuff specifically for developers.
* **Google Play Services (for Android)**: Ensure that your Flutter project is set up to use Google Play Services, which may be required for certain Firebase functionalities, such as push notifications or device registration.