

Academic year 2022 – 2023



Abbas Al Shies #1 + 201762530#1

Ali Alsayafi #2 + 201845580

Mohammed Anwar Almohammedsaleh#3 + 201814920#3

Ali Alaqi #4 + 201839620#4

Ali Albannay #5 + 201848300

Emad Ali Alomran 201757930

History

| Version | Date | Author | Changes |
|---------|-------------|-----------------|------------------------------|
| 1.0 | Jan 27,2023 | All the members | Test Cases set up |
| 1.1 | Jan 28,2023 | All the members | Rest of the document section |

Table of Contents

| | | |
|--------|---------------------------------|---|
| 1. | Introduction..... | 4 |
| 1.1. | The Project..... | 4 |
| 1.2. | Overview..... | 4 |
| 1.2.1. | Test Organization..... | 4 |
| 1.2.2. | System Under test (SUT)..... | 4 |
| 1.2.3. | Testing types..... | 4 |
| 1.2.4. | Features to be tested..... | 4 |
| 1.2.5. | Features not to be tested..... | 5 |
| 1.3. | Test Environment and Tools..... | 5 |
| 1.4. | Document Structure..... | 5 |
| 2. | System Test..... | 5 |
| 2.1. | System Requirements..... | 5 |
| 2.2. | System Test Cases..... | 8 |

1. Introduction

1.1. The Project

project aims to establish a relationship between the user and the business owner in which the user can earn points by interacting with the system and converting them into coupons, receive discounts and rewards for their daily spending, and receive a reward for analyzing their data. Moreover, it will make it possible for business owners to quickly construct a loyalty reward system in order to employ a pull and push marketing campaign. In addition, it will enable business owners to establish a richness of reaching customers in order to provide them with products and services based on their requirements.

1.2. Overview

1.2.1. Test Organization

Teams are composed of four members each. Provide the assigned tasks for each team member.

Phase 1: Mohammed Almohammedsaleh

Phase 2: Ali Alsayafi

Phase 3: Ali Albannay

Phase 4: Ali Alaql

1.2.2. System Under test (SUT)

The SUT is on the source code of MyRewards.

The source code consists of both front-end and back-end and has two main prospective view which are customer and business owner.

1.2.3. Testing types

Black Box Testing

White Box Testing

Alpha Testing

Beta Testing

1.2.4. Features to be tested

Briefly describe what aspects of the software will be tested.

R2, R4, R5, R9, R11, R12, R16, R17, R18, R22, R24, R27, R28, R30, R32, R35, R36, R37, R39, R40, R45, R46, R47, R48, R49, R450

1.2.5. Features not to be tested

R1, R3, R6, R7, R8, R10, R13, R14, R15, R19, R20, R21, R23, R25, R26, R31, R33, R34, R38, R41, R42, R43, R44

1.3. Test Environment and Tools

Since the project is split into two platforms: Flutter and Angular. Each test case will be tested via the Environment associated with the used Platform for the requirement.

The Test Environment that will be used is as follows:

| Platform | Test Environment |
|----------|---------------------------|
| Flutter | Flutter test environment |
| Angular | Jasmine testing framework |

1.4. Document Structure

This document follows (not strictly) the template for test plans in IEEE standard for software test documentation [IEEE829].

2. System Test

2.1. System Requirements

All requirements extracted from the SRS document are listed in the following table:

| Requirement number | Description |
|--------------------|---|
| R1 | The system shall provide two-point systems (Coins & XP). |
| R2 | The system shall allow the user to view wallet balance. |
| R3 | The system shall enable the user to exchange wallet coins with offers that are predefined by the system. |
| R4 | The system shall enable the user to gain points (Coins & XP) for each interaction with the system. |
| R5 | The system shall categorize the users based on the accumulated XP. |
| R6 | Wallet coins shall expire within a year. |
| R7 | The system shall display a dashboard with users' rankings (XP points). |
| R8 | The system shall define a classification structure based on users' XP points |
| R9 | The system shall enable the BO to add a new offer. |
| R10 | The system shall enable BO to modify/delete the offer within x time (less hours) -> before the offer go public... (Once the offer is public it cannot be modified |
| R11 | The BO should be able to register its store through the website |
| R12 | The BO should be able to register its store through the website |

| | |
|-------|--|
| R13 | The system should be able to validate the store using its commercial record through API of ministry of commerce. |
| R14 | The system shall allow BOs that has their own promo code system to provide a set of promo codes |
| R15 | The system shall send the users promo codes form the provided set. |
| R16 | The system shall generate promo code for small/medium BOs. |
| R17 | The system shall send users the generated promo code. |
| R18 | The system shall enable BOs to validate promo code through the system. |
| R19 | The system shall enable users to send an invitation code (download link and code) to other users or non-users. |
| R20 | The system shall give points (XP + coins) to the link owner when his code got used by another user who met the criteria. |
| R21 | The system shall give the link sender credit points silently (should not notify him), the notification should be with the next time the user opens the app (you received x points) |
| R22 | The system shall give points to users when they use an invitation code |
| R23 | Invitation code users shall receive less points than the code host |
| R24 | The system shall generate a link once per user per store |
| R25 | The link shall include the Store Id and User Id |
| R26 | The system shall generate the code with the first invitation request, then it can be copied (users can copy and send the code but the only beneficiary is the owner of the code) or sending the same one |
| R27 | The system shall revoke all links associated with a store when the store close (deleted) |
| R28 | The system shall redirect the link user to his application store if he does not have the application. |
| R29 | The system shall activate a trigger to listen for a transaction from the required store, in the link user device. |
| R30-- | The system shall collect and analyze incoming texts to user mobile and check if they are from a bank and it is a purchase transaction, extract the store name, amount, and date of transaction. |
| R31 | The system shall store data collected from the user's SMS message. |

| | |
|-------|--|
| R32 | The system shall display the extracted transaction's details with a notification and enable the user to modify the information, if no modification is made the system will process the data as is. |
| R33 | The system shall prompt them to share their review of their experience with the store and remind them of the rewards of the review. |
| R34 | The system shall process users' reviews using NLP and extract |
| R35-- | Users shall be able to view their monthly spending |
| R36 | User shall be able to see his spending in each which category (entertainment, coffee,). |
| R37 | Users should be able to view how much they saved (based on the offer) |
| R38 | User shall be categorized based on his saving achievements |
| R39 | The system shall allow users to add a new purchase detail. |
| R40-- | The system shall prompt(notify) the user to make a review of store only and only if a transaction is made from that store. |
| R41 | The system shall only accept reviews with a minimal of 100 characters |
| R42 | The system shall validate the review using NLP that the review is legit (useful) |
| R43 | The system shall store all reviews in a database. |
| R44 | The system shall store processed reviews in the database. |
| R45-- | The system shall provide a public forum for sharing inquiries (Jodel like) |
| R46 | The system shall enable users to submit an inquiry. |
| R47 | The system shall enable other users to answer the inquiry (answers will be public) |
| R48 | The system shall enable the inquiry submitter to choose(vote) the most helpful answer |
| R49 | The system shall reward both the user with the answer and the inquiry submitter (for selecting answer) |
| R50 | The user should be able to vote for helpful responses. |
| R51 | The user shall be able to register by their phone number and username. |
| R52 | The system shall prompt (optional) the user for birth date, sex. |
| R53 | The system shall give each account an associated points wallet |
| R54 | The system shall enable the user to change their phone number. |
| R55 | The BO shall register by their email and password. |
| R56 | The BO shall be able to change their password. |

| | |
|-----|--|
| R57 | The system shall authenticate the SMS sender. |
| R58 | The system phone number by OTP |
| R59 | The system shall authenticate BO MOC identity. |

2.2. System Test Cases

Design your test cases according to the following test case template

Here are the **requirements that can be tested.**

| | |
|--|--|
| Test Case Title | View Wallet Balance |
| Test Case ID | TC-01 |
| Requirement IDs | R2 |
| Author | Ali Alsayafi |
| Description | The system shall allow the user to view wallet balance. |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | The balance is 0. |
| Test steps (inputs, actions, etc.) | The user opens the application. Taps on the wallet. Shows a page with the balance of the user. |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | Zero if the user has not made any points. Otherwise, the points that he/she claimed. |
| Priority (High/Medium/Low) | High |
| Test Type (Functional, Performance, stress, etc.) | Functional. |
| Platform | Flutter |

| | |
|--|--|
| Test Case Title | Gain points and Xp |
| Test Case ID | TC-02 |
| Requirement IDs | R4 |
| Author | Ali Alsayafi |
| Description | Gain points after each interaction with the system |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | The user have valid account |
| Test steps (inputs, actions, etc.) | The user opens the application. Tabs on community page Answer an inquiry |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | His points should be incremented |
| Priority (High/Medium/Low) | Medium |
| Test Type (Functional, Performance, stress, etc.) | Functional. |
| Platform | Flutter |

| | |
|--|--|
| Test Case Title | Categorize user |
| Test Case ID | TC-03 |
| Requirement IDs | R5 |
| Author | Ali Alaql |
| Description | The system ranks the user based on his points and redeemed offers. |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | Base rank. |

| | |
|---|---|
| Test steps (inputs, actions, etc.) | The user opens the application. Goes to profile. Sees his/her rank. |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | The rank label. |
| Priority (High/Medium/Low) | Low. |
| Test Type (Functional, Performance, stress, etc.) | Functional. |
| Platform | Flutter |

| | |
|--|--|
| Test Case Title | Add offer |
| Test Case ID | TC-04 |
| Requirement IDs | R9 |
| Author | Ali Alaql |
| Description | BO can add new offer to his store |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | The user has an account as BO |
| Test steps (inputs, actions, etc.) | The user opens the web application. The user logs in Tabs on add offer |

| | |
|---|------------------------------------|
| | Enter offer's details |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | The offer is added to the database |
| Priority (High/Medium/Low) | High |
| Test Type (Functional, Performance, stress, etc.) | Functional. |
| Platform | Angular |

| | |
|--|--|
| Test Case Title | modify/delete the offer |
| Test Case ID | TC-05 |
| Requirement IDs | R11 |
| Author | Ali Alsayafi |
| Description | The system shall enable BO to modify/delete the offer within x time (less hours) -> before the offer go public... (Once the offer is public it cannot be modified) |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | 1. The user must be logged in with an active BO account. There must be an offer in private state |
| Test steps (inputs, actions, etc.) | 1. BO enters his dashboard thru website 2. BO presses on existing offers in the dashboard 3. The Application displays list of offers 4. BO selects an offer that is in private state 5. The Application prompts the user to edit it Store the modification of the offer |

| | |
|---|----------------------|
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | A new offer modified |
| Priority (High/Medium/Low) | Medium |
| Test Type (Functional, Performance, stress, etc.) | Functional. |
| Platform | Angular |

| | |
|--|--|
| Test Case Title | BO register |
| Test Case ID | TC-6 |
| Requirement IDs | R12 |
| Author | Ali Alsayafi |
| Description | BO registers via the website. |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | The user has no account |
| Test steps (inputs, actions, etc.) | The user opens the website. Clicks on sign up. Fill his/her information. |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | He/she has an account in myRewards. |
| Priority (High/Medium/Low) | High. |
| Test Type (Functional, Performance, stress, etc.) | Functional. |
| Platform | Angular |

| | |
|--|--|
| Test Case Title | Claim Offer Code |
| Test Case ID | TC-7 |
| Requirement IDs | R16, R17 |
| Author | Ali Alsayafi |
| Description | Once the user claim the offer, he/she will receive a promo code |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | No codes are provided. |
| Test steps (inputs, actions, etc.) | User login User clicks the offer User claim the offer Code is shown to user |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | Codes are available for the users "Clients" |
| Priority (High/Medium/Low) | Medium |
| Platform | Flutter |

| | |
|--|--|
| Test Case Title | Validate Promo code |
| Test Case ID | TC-8 |
| Requirement IDs | R18 |
| Author | ABBAS AL SHIEF |
| Description | The system validates the code if it is useable for the clients or not. |
| Initial State (based on Pre-conditions and other data | There is code to validate. |

| | |
|---|---|
| that have available to conduct the test.) | |
| Test steps (inputs, actions, etc.) | Types the code in the field Clicks on validate |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | The status of the code |
| Priority (High/Medium/Low) | Medium |
| Platform | Angular |

| | |
|--|--|
| Test Case Title | giving the link sender credit points |
| Test Case ID | TC-9 |
| Requirement IDs | R22 |
| Author | ABBAS AL SHIEF |
| Description | The system shall give the link sender credit points silently (should not notify him), the notification should be with the next time the user opens the app (you received x points) |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | The user has an account |
| Test steps (inputs, actions, etc.) | The user should send the links to get the points. After opening the link, the system gives points to the sender. |

| | |
|---|--------------------------------|
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | The sender will get the points |
| Priority (High/Medium/Low) | Medium |
| Platform | Flutter |

| | |
|--|---|
| Test Case Title | Generate Invitation Code |
| Test Case ID | TC-11 |
| Requirement IDs | R24 |
| Author | ABBAS AL SHIEF |
| Description | The user should be able to generate an invitation code for each store. |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | A user has an account |
| Test steps (inputs, actions, etc.) | Opens the application Select a store from stores list Clicks generate invitation code |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | A link is generated with a unique id that links user with the store |
| Priority (High/Medium/Low) | Low |
| Platform | Flutter |

| | |
|------------------------|----------------------------|
| Test Case Title | Delete deleted store links |
| | |

| | |
|--|---|
| Test Case ID | TC-10 |
| Requirement IDs | R27 |
| Author | Ali Alaql |
| Description | The system deletes the links of the deleted store. |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | The store is deleted. |
| Test steps (inputs, actions, etc.) | Try the link. |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | Page states that the store is no longer in the application. |
| Priority (High/Medium/Low) | Low |
| Platform | Flutter |

| | |
|--|--|
| Test Case Title | display the extracted transaction's details |
| Test Case ID | TC-11 |
| Requirement IDs | R32 |
| Author | Ali Alaql |
| Description | Extracting data from the transaction SMS message |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | The application is on and can listen to incoming messages to the mobile |
| Test steps (inputs, actions, etc.) | <ol style="list-style-type: none"> 1. The system detects an incoming message. 2. The system checks if the sender is a bank and message is transaction SMS. 3. The system extract Date, Amount, and Store name from the message 4. The system displays the extracted data to user and prompts confirmation. 5. User confirms. 6. The system stores the data in the database |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | Date, Amount, and Store is extracted in from the message and stored in the database |
| Priority (High/Medium/Low) | Low |
| Platform | angular |

| | |
|--|---|
| Test Case Title | SMS Extraction |
| Test Case ID | TC-12 |
| Requirement IDs | R30, R40 |
| Author | Ali Alaql |
| Description | The application extracts the message and analyze it. |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | There is no purchase. |
| Test steps (inputs, actions, etc.) | The user makes a transaction. Receives a message from the bank. The application extracts it and analyze it. |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | The transaction is analyzed by the system and the user can review the system |
| Priority (High/Medium/Low) | High |
| Platform | Flutter |

| | |
|--|---|
| Test Case Title | revoke all links associated with a store |
| Test Case ID | TC-13 |
| Requirement IDs | R28 |
| Author | ABBAS AL SHIEF |
| Description | The system shall revoke all links associated with a store when the store close (deleted) |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | The store should be closed |
| Test steps (inputs, actions, etc.) | The system checks the state of store. The system will revoke all links associated with a store when the store close. |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | Closing all links |
| Priority (High/Medium/Low) | Medium |
| Platform | Flutter |

| | |
|------------------------|---|
| Test Case Title | View monthly spending |
| Test Case ID | TC-15 |
| Requirement IDs | R35, R36 |
| Author | Ali Albannay |
| Description | The user views his/her monthly spending |

| | |
|--|--|
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | There are spending that user has. |
| Test steps (inputs, actions, etc.) | Opens application. Clicks on spending |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | Chart with spendings and the categories |
| Priority (High/Medium/Low) | Medium |
| Platform | Flutter |

| | |
|--|--|
| Test Case Title | viewing their monthly spending |
| Test Case ID | TC-16 |
| Requirement IDs | R37 |
| Author | ABBAS AL SHIEF |
| Description | Users shall be able to view their monthly spending |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | The user should have an account. The user should use the application |
| Test steps (inputs, actions, etc.) | <ol style="list-style-type: none"> 1. The user clicks on view monthly spending. 2. The system will display the hole spending during the month. |
| Expected Output (based on post-conditions and other | The user should see his/her spending details |

| | |
|---|---------|
| outcomes specified in the flow of events.) | |
| Priority (High/Medium/Low) | Low |
| Platform | angular |

| | |
|--|--|
| Test Case Title | Add New Purchase details |
| Test Case ID | TC-17 |
| Requirement IDs | R39 |
| Author | Ali Albannay |
| Description | To add purchase manually |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | The user has a valid account |
| Test steps (inputs, actions, etc.) | <p>The user opens the app</p> <p>Go to purchases page</p> <p>Select a purchase</p> <p>Enter purchase details</p> |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | The system stores new details |

| | |
|-----------------------------------|---------|
| Priority (High/Medium/Low) | Medium |
| Platform | Flutter |

| | |
|--|---|
| Test Case Title | Community |
| Test Case ID | TC-18 |
| Requirement IDs | R45 to R50 |
| Author | Ali Albannay |
| Description | A community for the users to interact within. |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | The user is registered in the application. |
| Test steps (inputs, actions, etc.) | Opens application clicks on community and post inquires. Votes for helpful response. Reply on other users |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | Get successful feedback on every action |
| Priority (High/Medium/Low) | Low |
| Platform | Flutter |

| | |
|--|---|
| Test Case Title | Register by phone number and username |
| Test Case ID | |
| Requirement IDs | 51 |
| Author | Emad Alomran |
| Description | The system should provide an option for the user to register by phone number and username |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | |
| Test steps (inputs, actions, etc.) | 1- Click on register text button. 2- Chose to register by phone number. 3- Write a phone number. 4- Enter the received OTP. 5- Write username |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | New registered account |
| Priority (High/Medium/Low) | Medium |
| Test Type (Functional, Performance, stress, etc.) | Functional |
| Platform | Flutter |

| | |
|--|---|
| Test Case Title | Optional Information is provided |
| Test Case ID | |
| Requirement IDs | 52 |
| Author | Emad Alomran |
| Description | The system should provide birth date, sex and other information in the form to be optionally filled |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | |
| Test steps (inputs, actions, etc.) | 1- Click on register text button. |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | The user should see birth date and sex in the form |
| Priority (High/Medium/Low) | Low |
| Test Type (Functional, Performance, stress, etc.) | Non-Functional |

| | |
|-----------------|---------|
| Platform | Flutter |
|-----------------|---------|

| | |
|--|---|
| Test Case Title | Point wallet association |
| Test Case ID | |
| Requirement IDs | 53 |
| Author | Mohammed Almohammedsaleh |
| Description | The system should associate a point wallet for each account |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | - Registered account. |
| Test steps (inputs, actions, etc.) | 1- Create new account. 2- Access the points wallet |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | The user should have seen a points wallet |
| Priority (High/Medium/Low) | High |
| Test Type (Functional, Performance, stress, etc.) | Non-Functional |
| Platform | Flutter |

| | |
|--|---|
| Test Case Title | |
| Test Case ID | |
| Requirement IDs | 54 |
| Author | Emad Alomran |
| Description | The user should be able to change his/her registered phone number |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | - The user should have a registered phone number |
| Test steps (inputs, actions, etc.) | 1- Enter the settings page. 2- Enter the account information page. 3- Click on change phone number. 4- Enter the OTP sent to the old phone number. 5- Write the new phone number. 6- Enter the OTP sent to the new phone number. |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | A confirmation pop-up to tell the user that his number has changed |
| Priority (High/Medium/Low) | Medium |

| | |
|--|------------|
| Test Type (Functional, Performance, stress, etc.) | Functional |
| Platform | Flutter |

| | |
|--|---|
| Test Case Title | |
| Test Case ID | |
| Requirement IDs | 55 |
| Author | Mohammed Almohammedsaleh |
| Description | The BO should be able to register by email and password |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | - The BO should have a real email to confirm it after registration process |
| Test steps (inputs, actions, etc.) | 1- Clicks on register text button. 2- Choses to register using email. 3- Fills the form. 4- Clicks on register button. |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | 1- Registered account pop-up. 2- Home page |
| Priority (High/Medium/Low) | High |
| Test Type (Functional, Performance, stress, etc.) | Functional |
| Platform | Angular |

| | |
|--|---|
| Test Case Title | authenticate BO MOC identity. |
| Test Case ID | TC-13 |
| Requirement IDs | R59 |
| Author | ABBAS AL SHIEF |
| Description | The system shall authenticate BO MOC identity. |
| Initial State (based on Pre-conditions and other data that have available to conduct the test.) | The user should have |
| Test steps (inputs, actions, etc.) | The system checks the state of store. The system will revoke all links associated with a store when the store close. |
| Expected Output (based on post-conditions and other outcomes specified in the flow of events.) | Closing all links |
| Priority (High/Medium/Low) | Medium |
| Platform | Flutter |

2.2 Beta testing for mobile app

| | | | | | | | | |
|---|---------|---------|---------|---------|---------|---------|---------|---------|
| Can you register in the app? / هل تستطيع التسجيل في التطبيق ؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |
| Can you login in the application? / هل تستطيع تسجيل الدخول في التطبيق ؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |
| Can you logout from your account ? / هل تستطيع تسجيل الخروج من حسابك ؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |
| Can you change your personal information from the settings ? / هل تستطيع تعديل بياناتك الشخصية من الإعدادات ؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |
| Can you see the available stores in the app? / هل تستطيع رؤية المتاجر المتوفرة ؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |

| | | | | | | | | |
|---|---------|---------|---------|---------|---------|---------|---------|---------------------|
| في التطبيق؟ | | | | | | | | |
| Can you use the search bar to find a specific store? / هل تستطيع استخدام شريط البحث للعثور على متجر معين؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |
| Can the app recognize your purchase in the Statistics page? / هل يمكن للتطبيق التعرف على مشترياتك في صفحة الإحصائيات؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Somewhat/ الى حد ما |
| Can you see your financial statistics? / هل تستطيع رؤية إحصاءاتك المالية؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |
| Can you view your previous purchase transactions? / هل يمكنك عرض معاملات الشراء السابقة الخاصة بك؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |
| Can you see your categorized purchase transactions? / هل يمكنك رؤية معاملات الشراء المصنفة الخاصة بك؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |

2.3 Beta testing for website

| | | | | | | | | |
|--|---------|---------|---------|---------|---------|---------|---------|---------|
| Can you register in the website? / هل تستطيع التسجيل في الموقع؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |
| Can you log in to the website? / هل يمكنك تسجيل الدخول إلى الموقع؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |
| Can you logout from your account? / هل تستطيع تسجيل الخروج من حسابك؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |
| Can you add an offer to your store? / هل يمكنك إضافة عرض لمتجرك؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |
| Can you see the offers in your store? / هل يمكنك رؤية العروض في متجرك؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |
| Can you see your store's statistics dashboard? / هل يمكنك رؤية لوحة معلومات إحصائيات متجرك؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |
| Can you delete an offer from your store? / هل يمكنك حذف عرض من متجرك؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |
| Can you modify an offer from your store? / هل يمكنك التعديل على عرض من متجرك؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |
| Can you redeem an offer? / هل يمكنك استرداد العرض؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |
| Can you see the offer's details before redeeming it? / هل يمكنك الاطلاع على تفاصيل العرض قبل استرداده؟ | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم | Yes/نعم |

2.4 White box testing

| Function | Case | Values | Pass/Fail |
|---------------------|--|--|-----------|
| cashierLoginWithOTP | Incorrect OTP | email: "test@example.com", otp: "123456" | Pass |
| cashierLoginWithOTP | Expired OTP | email: "user@example.com", otp: "987654" | Pass |
| cashierLoginWithOTP | Valid OTP and email | email: "admin@example.com", otp: "000000" | Pass |
| createStore | Successful store creation | uid: "exampleUid", storeName: "Example Store" | Pass |
| createStore | Failed store creation | uid: "", storeName: "Invalid Store" | Fail |
| customCashierLogin | Successful login (existing owner and cashier) | ownerEmail: "example@example.com" " | Pass |

| | | | |
|--------------------|--|---|------|
| customCashierLogin | Successful login (existing owner, create new cashier) | ownerEmail: " example@example.com " | Pass |
| customCashierLogin | Failed login (owner does not exist) | ownerEmail: " nonexistent@example.com " | Fail |
| customCashierLogin | Failed login (user not found error) | ownerEmail: "" (empty owner email) | Fail |
| registerCashier | Successful registration | ownerUid: "abc123" | Pass |
| registerCashier | Failed registration (missing owner UID) | ownerUid: "" | Fail |
| generateCashierOTP | New user | email: " test@example.com " | Pass |
| generateCashierOTP | Existing user | email: " user@example.com " | Pass |
| generateCashierOTP | Empty email | email: "" | Pass |

| | | | |
|--------------------|-------------------------|---|------|
| registerOwner | Successful registration | email: "test@example.com", password: "123456", storeName: "Test Store" | Pass |
| registerOwner | Empty email | email: "", password: "123456", storeName: "Test Store" | Pass |
| registerOwner | Weak password | email: "user@example.com", password: "123", storeName: "Test Store" | Pass |
| convertRialToToken | Positive Rial | rial: 100 | Pass |
| convertRialToToken | Zero Rial | rial: 0 | Pass |
| convertRialToToken | Negative Rial | rial: -50 | Pass |
| claimOffer | Valid data | data: { offer_id: "offer123", store_id: "store456" } | Pass |
| claimOffer | Missing data | data: {} | Pass |

| | | | |
|---------------------|---|--|------|
| onCreateTransaction | Store found, add points | snapshot: { data: { transaction data } }, context: { params: { userId: "user123" } } | Pass |
| onCreateTransaction | Store found, update store | snapshot: { data: { transaction data } }, context: { params: { userId: "user123" } } | Pass |
| onCreateTransaction | Store not found, store name match | snapshot: { data: { transaction data } }, context: { params: { userId: "user123" } } | Pass |
| onCreateTransaction | Store not found, store name mismatch | snapshot: { data: { transaction data } }, context: { params: { userId: "user123" } } | Pass |
| updateTopStores | Update user points (more than 5 stores) | change: { after: { data: { points: { store1: 100, store2: 200, store3: 150, store4: 300, store5: 250, store6: 50 } } } } | Pass |
| updateTopStores | Update user points (less than 5 stores) | change: { after: { data: { points: { store1: 100, store2: 200 } } } } | Pass |
| updateTopStores | Update user points | change: { after: { data: { | Pass |

| | | | |
|-------------------|---------------------------------------|--|------|
| | (empty points) | points: {}}} | |
| updateTopStores | Update user points (same points) | change: { after: { data: { points: { store1: 100, store2: 100, store3: 100, store4: 100, store5: 100 } } } } | Pass |
| updateTotalPoints | Update user points (new store added) | change: { after: { data: { points: { store1: 100, store2: 200, store3: 150, ... } } } } | Pass |
| updateTotalPoints | Update user points (store removed) | change: { after: { data: { points: { store1: 100, store2: 200, ... } } } } | Pass |
| updateTotalPoints | Update user points (points reduced) | change: { after: { data: { points: { store1: 50, store2: 150, ... } } } } | Pass |
| updateTotalPoints | Update user points (points increased) | change: { after: { data: { points: { store1: 200, store2: 300, ... } } } } | Pass |