Test Plan

FashionFinder

Team Members: Maria Alsamaien, Alex Gjeka,

Angjelo Mana & Oloofa Kalid

Introduction

Purpose

The purpose of this document is to provide all the testing information for the FashionFinder website. Ensuring that the platform operates as intended and delivers a smooth, user-friendly experience. This testing plan is created to verify and validate each aspect of FashionFinder, from its core functionality and user interface to its integration with external systems and performance under various conditions.

Objective

The primary aim of the FashionFinder testing plan is to establish a structured approach to testing the website, covering all necessary aspects to ensure that it aligns with the specifications outlined in the Software Requirements Specification (SRS) document. This document serves as a blueprint for the testing process, detailing the types of tests to be conducted and the methods to be used. By following this plan, we aim to identify and resolve any issues that may impact the user experience, functionality, performance, or security of FashionFinder.

Scope

Within the scope of this testing plan, we will thoroughly evaluate every component of the FashionFinder website. This includes testing its core functionalities, user interface elements, integration with external systems, and performance across various scenarios. We will not be testing APIs. We will assess how well the website functions under different conditions to ensure it meets the standards outlined in the Software Requirements Specification (SRS) document.

Intended Audience

This testing plan is intended for the testing team.

Test Objectives

Validating Functional Requirements

- Ensure that all functional requirements outlined in the Software Requirements Specification (SRS) document are thoroughly tested and validated.
- Verify that each feature and functionality of the FashionFinder website performs as expected, meeting the specified user needs and objectives.
- Validate the correctness of user interactions, data processing, and system responses according to predefined criteria.

Verifying System Performance

- Assess the performance of the FashionFinder website under various conditions, including normal usage, peak loads, and stress scenarios.
- Measure and analyze key performance metrics such as page load times, response times for user actions, and system scalability.
- Verify that the website can handle anticipated levels of traffic and data processing without significant degradation in performance.

Ensuring Security and Reliability

- Conduct comprehensive security testing to identify and mitigate potential vulnerabilities in the FashionFinder website, including authentication, authorization, data encryption, and protection against common security threats.
- Validate the reliability of the system by testing its ability to handle errors, recover from failures, and maintain data integrity.
- Ensure compliance with relevant security standards and regulations, safeguarding user data and maintaining trust in the integrity of the FashionFinder platform.

Test Scope

Testing scope refers to the extent and boundaries of the testing process that are defined for our project. Scope outlines what aspects of our software will be tested, and what aspects will not be tested. We will be testing all of the functionalities that we have created a test case for. We have included 34 test cases done below. We will not be testing any of the external functionalities such as FastAPI.

Test Environment

We are not using any testing environment as we are doing manual testing for all levels of testing.

Test Strategy

We are going to be using the Black-Box testing strategy for our project. Testers using the Black-Box strategy do not have knowledge of the internal workings on the system unlike other testing strategies.

Here are some of the reasons why we decided to use the Black-Box strategy:

- Complexity of our system: our project involves multiple user interactions, AI integration, and database management. Black-Box testing allows us to focus on the system's external behavior without needing to dive into its internal complexities.
- User-Centric testing: Black-Box testing aligns well with the user-centric approach required for our project.

- Realistic testing scenarios: Black-Box testing facilitates the creation of realistic testing scenarios that mimic user interactions with the system. This helps us uncover defects and issues that may arise during actual usage of the application.

Functional Testing

1. Unit Testing

Purpose

To test individual units or components of the software separately.

Scope

Focuses on verifying the correctness of small, specific functionalities, mainly functions. Will not be testing interactions between different components.

Tools

Manual testing will be used to unit test.

Approach

The approach to unit testing involves systematically testing individual units or components of the software. This process begins with identifying the units to be tested and writing corresponding test cases. Results are then analyzed to identify any discrepancies or errors, which are debugged and resolved iteratively. Through this process, the reliability and correctness of each unit are ensured, contributing to the overall quality and stability of the software.

Entry/ Exit Criteria

The entry criteria of unit testing is following the completion of prototype 3. The exit criteria of unit testing is the completion of all stated test cases.

Pass/ Fail Criteria

A test would be defined as passed if the expected outcome illustrated by the test case is the result of following said test case's steps. If the result of following the steps for the test case is different than what is expected or raises an error, it will be considered as failed.

Test Suspension and Resumption Criteria

A test will be suspended on a raised error or unexpected outcome. Once the issue has been resolved, testing will be resumed.

Risk Analysis

- **Risk**

  Introducing dependencies and overlooking edge cases when clicking buttons.

- **Method of Mitigation**

  Use a thorough unit testing approach that will manually test each small function to ensure correct functionality.

Items to be tested

Test Case ID	TC-1
Test Case Name	Send Chat Button
Pre-Conditions	1. User has to be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: “angjelomana1@gmail.com” • Password: “Angjelo1” 3. User has to be on the chat page.
Test Steps	1. User to enter “hello, help me find a t-shirt.” 2. Click on the up arrow icon (send chat button) that is on the right-hand side of the input box.

Expected Results	The user's message "Hello, help me find a t-shirt" should show up on the top of the chat screen.
Priority	High

Test Case ID	TC-2
Test Case Name	Save Chat Button
Pre-Conditions	1. User has to be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: "angjelomana1@gmail.com" • Password: "Angjelo1" 3. User has to be on the chat page. 4. User already chatted with the personal designer (even if just one message like "hello, help me find pants").
Test Steps	Click on the "save chat" button on the bottom right side of the screen.
Expected Results	A modal appears asking user to name their chat.
Priority	Mid

Test Case ID	TC-3
Test Case Name	New Chat Button
Pre-Conditions	1. User has to be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: “angjelomana1@gmail.com” • Password: “Angjelo1” 3. User has to be on the chat page. 4. User already chatted with the personal designer (even if just one message).
Test Steps	Click on the new chat button located on the top left side of the chat page.
Expected Results	A new chat page opens up.
Priority	Mid

Test Case ID	TC-4
Test Case Name	Add Outfit to Album Button - Chat Page
Pre-Conditions	1. User has to be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: - Email: “angjelomana1@gmail.com” - Password: “Angjelo1” 3. User has to be on the chat page. 4. User has chatted with the personal designer and matches have been returned.
Test Steps	Click on the “Add” button that’s located at the bottom of each returned match.
Expected Results	Modal appears asking user which album they’d like to add to.
Priority	High

Test Case ID	TC-5
Test Case Name	Go to Link Button
Pre-Conditions	1. User has to be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: “angjelomana1@gmail.com” • Password: “Angjelo1” 3. User has to be on the chat page. 4. User has chatted with the personal designer and matches have been returned.
Test Steps	Click on the “Go” button of any item once items are returned.
Expected Results	New tab will open directing user to link for where to buy item.
Priority	High

Test Case ID	TC-6
Test Case Name	Edit Album Name Button
Pre-Conditions	1. User has to be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: “angjelomana1@gmail.com” • Password: “Angjelo1” 3. User has to be on albums page. 4. An album has already been created with the name “Summer”.
Test Steps	Click the edit button on the album “Summer” which is located on the lower-right part of “Summer” album to edit.
Expected Results	Modal will appear asking user what they would like to change the album name to. If they change the name and press “save” album name will be updated. Else, the album name stays the same.
Priority	High

Test Case ID	TC-7
Test Case Name	Delete Album Button
Pre-Conditions	1. User has to be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: "angjelomana1@gmail.com" • Password: "Angjelo1" 3. User has to be on the albums page. 4. An album has already been created with the name "Summer".
Test Steps	1. Press the red trash can located on the lower-right part of the "Summer" album to delete the album. 2. Modal will appear asking user "do you want to delete this album?"
Expected Results	If the user presses yes, that album will be deleted from the album selection. Else, it will still be in the albums collection.
Priority	High

Test Case ID	TC-8
Test Case Name	Delete Outfit Button
Pre-Conditions	1. User has to be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: “angjelomana1@gmail.com” • Password: “Angjelo1” 3. User has to be on albums page. 4. An album has already been created with the name “Summer” and has at least one outfit in it.
Test Steps	1. User will click that album and be brought to the outfits page. 2. When they hover over their outfit, two options “go” and “delete” will appear underneath the outfit. 3. When pressing delete button, modal will appear asking user “do you want to delete this outfit from this album?”
Expected Results	If the user presses yes, that outfit will be deleted from that specific album. Else, it will still be on that album's outfit page.
Priority	High

Test Case ID	TC-9
Test Case Name	Edit Button - Profile Page
Pre-Conditions	1. User has to be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: "angjelomana1@gmail.com" • Password: "Angjelo1" 3. User has to be on profile page.
Test Steps	1. User will "press edit" button which is located under the "budget" field to edit password, brands, and budget fields. 2. User will press "save" once they have changed a field of their desire.
Expected Results	The new value for these respective fields will be changed to the updated version the user changed them to.
Priority	Mid

Test Case ID	TC-10
Test Case Name	Save Button - Profile Page
Pre-Conditions	1. User has to be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: “angjelomana1@gmail.com” • Password: “Angjelo1” 3. User has to be on profile page.
Test Steps	1. Press the “Edit button” on the profile page. 2. Press “Add Brand”. 3. Press the text field and enter a brand. 4. Go to budget and input a budget. 5. Press the “Save” button.
Expected Results	The modified budget and brand information is successfully saved and updated.
Priority	Mid

Test Case ID	TC-11
Test Case Name	Filter Button - Outfits page
Pre-Conditions	1. User has to be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: “angjelomana1@gmail.com” • Password: “Angjelo1” 3. User has to be on albums page.
Test Steps	1. Select the “Sort By” button. 2. Select a filter option. 3. Verify that the album list is updated according to the selected criteria.
Expected Results	1. The album list will display the albums based on the selected filter. 2. The filtered album list will display without any errors.
Priority	Mid

Test Case ID	TC-12
Test Case Name	Share Button - Albums Page
Pre-Conditions	1. User has to be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: "angjelomana1@gmail.com • Password: "Angjelo1" 3. User has to be on albums page.
Test Steps	1. Select an album. 2. Press on the "share" icon at the bottom right of the album. 3. Copy the link. 4. Send the link. 5. Verify the link takes the user to the shared album page.
Expected Results	The user that clicks the shared album link successfully sees the album.
Priority	Low

2. Integration Testing

Purpose

To test interactions between integrated units or components to ensure they work together as expected.

Scope

Validates the interfaces and communication between different modules or services.

Will not be testing the compatibility between features.

Tools

Manual testing will be used for integration testing.

Approach

The approach to integration testing entails verifying the seamless interaction between integrated units or components within the software. This process starts with identifying the interfaces and communication channels between different modules or services. Test cases are then developed to validate these interactions. Subsequently, the results are analyzed to detect any discrepancies or errors in the integration. Any identified issues are debugged and resolved iteratively. This iterative process ensures that the integrated units work together harmoniously, contributing to the overall functionality and reliability of the software system.

Entry/ Exit Criteria

The entry criteria of integration testing is the completion of functional and non-functional testing. The exit criteria of integration testing is the completion of all stated integration test cases.

Pass/ Fail Criteria

A test would be defined as passed if the expected outcome illustrated by the test case is the result of following said test case's steps. If the result of following the steps for the test case is different than what is expected or raises an error, it will be considered as failed.

Test Suspension and Resumption Criteria

Testing will be suspended if the current test raises an error or unexpected outcome that results in further testing being affected. In this case, we must first fix the issues and once fixed, we may resume testing.

Risk Analysis

- **Risk**

  Bad communication or incompatible interfaces between integrated components may allow for integration problems when pressing and loading different pages.

- **Method of Mitigation**

  To find and fix integration problems quickly, do integration testing often during the development process. Also, to reduce these issues, keep interface documentation and follow communication protocols for integration.

Items to be tested

Test Case ID	TC-13
Test Case Name	Account Creation
Pre-Conditions	User must be on FashionFinder landing page
Test Steps	1. Users will click ‘Register’ button on FashionFinder landing page. 2. Enter in the following fields • First name: “Angjelo” • Last name: “Mana” • Email: “angjelomana1@gmail.com” • Password: “Angjelo1” • Confirm Password: “Angjelo1” 3. Click “register” button.
Expected Results	Users will be prompted with green verification message to verify their email for account creation within 6 min and account will be added to the database.
Priority	High

Test Case ID	TC-14
Test Case Name	Forgot Password
Pre-Conditions	1. User must be on the login page 2. Users must have a verified account created with FashionFinder.
Test Steps	1. Users will click the 'Forgot Password' link under the login field. 2. Enter in the following field • Email: "angjelomana1@gmail.com" 3. Click "Reset Password" button. 4. Users will be prompted with green verification message to reset their password for that associated account. 5. Users will click that verification link and be prompted with a screen asking for a new password. 6. Users will type in a new password and press "reset password" button.
Expected Results	Users will be prompted with a green success message saying their password has been reset for that account. the new password will also be updated in the database.
Priority	High

Test Case ID	TC-15
Test Case Name	Sign in

Pre-Conditions	Users must have successfully registered and verified their account and must be on the login page.
Test Steps	1. Enter in the following field • Email: “angjelomana1@gmail.com” • Password: Angjelo1 2. Click “Next” button to sign into account.
Expected Results	Users will be directed to the “FashionFinder” page.
Priority	High

Test Case ID	TC-16
Test Case Name	User Verification

Pre-Conditions	The user has to have successfully registered their account and received a verification email from AI FashionFinder.
Test Steps	1. Users will click the verification link from AI FashionFinder in their email. 2. User will be prompted with a green conformation message saying their "account has been verified" when they press the link. 3. User will press proceed to login button.
Expected Results	User will be verified and be able to login with that account
Priority	High

Test Case ID	TC-17
Test Case Name	Customer Support

Pre-Conditions	1. User has to be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: "angjelomana1@gmail.com" • Password: Angjelo1 3. User has to be on the "Contact Us" page.
Test Steps	1. Enter a message like "Your website is cool!" in the "Message" input field. 2. Click on the "Send Message" button underneath the message box.
Expected Results	1. Upon clicking "Send Message", the user should get a message saying, "Your message has been sent successfully!" 2. Email will be sent to the AI FashionFinder team. 3. An automated email to be sent to the user.
Priority	Low

Test Case ID	TC-18

Test Case Name	Album Creation
Pre-Conditions	1. User must be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: "angjelomana1@gmail.com" • Password: Angjelo1 3. User must be on the albums page
Test Steps	1. User clicks "Create Album" button on the top middle of the albums page. 2. User enters an album name. 3. User then clicks "Create" button.
Expected Results	1. A new album with the provided name will be created in the user's account. 2. The album will be visible on the user's album page.
Priority	Mid

Test Case ID	TC-19
Test Case Name	Album Deletion
Pre-Conditions	1. User must be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: "angjelomana1@gmail.com" • Password: Angjelo1 3. User is on the albums page. 4. User has at least one album created.
Test Steps	1. Go to an existing album from the user's album list. 2. Click on the red trashcan symbol on the bottom right of the specific album name. 3. After clicking, it will display "Are you sure you want to delete this album?" click "Yes".
Expected Results	1. After clicking "Yes" the selected album should be deleted from the user's account. 2. The albums page will be updated.
Priority	Low

Test Case ID	TC-20
Test Case Name	Album Management - Adding Items
Pre-Conditions	1. User must be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: "angjelomana1@gmail.com" • Password: Angjelo1 3. User is on the chat page. 4. User has at least one album created.
Test Steps	1. User chats with the AI. 2. AI sends clothing images to the user. 3. User hovers over the image and clicks "Add". 4. User chooses which album to add the clothing item too.
Expected Results	1. The selected item(s) should be added to the chosen album. 2. The albums will be updated to include the new item(s).
Priority	Low

Test Case ID	TC-21
Test Case Name	Album Management - Deleting Items
Pre-Conditions	1. User must be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: “angjelomana1@gmail.com” • Password: Angjelo1 3. User is on the albums page 4. User has at least one album with one or more items in it.
Test Steps	1. Select an album from the user’s album list. 2. Select one or more items to delete from the selected album. 3. Hover over the selected item and click delete. 4. After clicking, it will display “Are you sure you want to delete this item?” click “Yes”.
Expected Results	1. After clicking “Yes” the selected item(s) should be deleted from the user's account. 2. The album will be updated.
Priority	Low

Test Case ID	TC-22
Test Case Name	Album Sharing
Pre-Conditions	1. User must be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: - Email: “angjelomana1@gmail.com” - Password: Angjelo1 3. User has to be on the albums page. 4. User has at least one album created.
Test Steps	1. Click on the “share button” icon on the bottom of the album. 2. Copy the link.
Expected Results	1. A link generates which can be copied and sent over to someone. 2. The other person is able to view the album.
Priority	Mid

Test Case ID	TC-23
Test Case Name	Brand Preference
Pre-Conditions	1. User must be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: - Email: “angjelomana1@gmail.com” - Password: Angjelo1 3. User has to be on the profile page.
Test Steps	1. Click on “edit” button on the profile page. 2. Add a brand by clicking “add” next to the brand section” 3. Click the “save” button.
Expected Results	Brand added should show up under the brand section.
Priority	Low

Test Case ID	TC-24
Test Case Name	Budget Input Box
Pre-Conditions	1. User must be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: “angjelomana1@gmail.com” • Password: Angjelo1 3. User has to be on the profile page.
Test Steps	1. Click on “edit” button on the profile page. 2. Add a budget into the budget input box. 3. Click the “save” button.
Expected Results	Budget added should show up under the brand section.
Priority	Low

Test Case ID	TC-25
Test Case Name	Customer Review System
Pre-Conditions	1. User has to be registered as a verified user. 2. User has to be logged into their account. Enter the following credentials to log in: • Email: “angjelomana1@gmail.com” • Password: “Angjelo1” 3. User has to be on the home page. 4. User has started chatting with the AI and provided all 4 keywords (gender, color, clothing type & size). For example, “help me find women's black t-shirts in size small.” 5. Matches have been returned or “Could not find any matches” message shown in the case that there is no match. 6. “Leave us a review” message shows up in a separate message below the matches.
Test Steps	1. Click on “Leave us a review”. 2. Enter 5 stars for rating. 3. Enter “I like this website” in the comments section. 4. Click on the “Submit” button.
Expected Results	1. Review is saved in the database 2. User gets a message saying, “Thank you for your review.”
Priority	Low

Test Case ID	TC-26
Test Case Name	Chat with Personal Designer
Pre-Conditions	1. User has to be registered as a verified user. 2. User has to be logged into their account. Enter the following credentials to log in: • Email: "angjelomana1@gmail.com" • Password: "Angjelo1" 3. User has to be on the chat page.
Test Steps	User enters a message into the chat for example, "Hello, can you help me find pants?"
Expected Results	The AI responds with a message saying something like "Hello [User], yes, I would be more than happy to help you find [user input]."
Priority	High

Test Case ID	TC-27
Test Case Name	Outfit Matching

Pre-Conditions	1. User has to be registered as a verified user. 2. User has to be logged into their account. Enter the following credentials to log in: • Email: "angjelomana1@gmail.com" • Password: "Angjelo1" 3. User has to be on the chat page. 4. User has started chatting with the AI and provided all 4 keywords (gender, color, clothing type & size). 5. AI should take all 4 keywords from user and turn them into a Json format.
Test Steps	1. Take keywords that are in a Json format and match them against what is available in the database. 2. Gather all the ones that match those keywords. 3. Return 5 in an image and a link format.
Expected Results	1. In the case that a match is found, 5 items will be returned to user. 2. In the case that a match is not found, the user will get a message saying, "Could not find any matches."
Priority	High

Test Case ID	TC-28
Test Case Name	Filtering Outfits

Pre-Conditions	1. User must be registered as a verified user. 2. User has to be logged into their account. Use the following credentials to log in: • Email: “angjelomana1@gmail.com” • Password: Angjelo1 3. User must be on the albums page. 4. User has at least one album with one or more items in it.
Test Steps	1. Click on the filter button. 2. Pick one of the filters provided.
Expected Results	Outfits will be filtered based on the selected filters.
Priority	Mid

3. System Testing

Purpose

To test the complete, integrated system to ensure it meets specified requirements and functions correctly.

Scope

Validates end-to-end system behavior, including functional and non-functional aspects such as performance, security, and reliability.

Tools

Manual testing will be used to system test.

Approach

The approach to system testing involves evaluating the entire software system to ensure it meets specified requirements and functions correctly in its intended environment. This entails comprehensive testing of functionality, performance, reliability, security, and more. Through manual testing, the software's quality, reliability, and suitability for its intended purpose are verified, ensuring a robust product for end-users.

Entry/ Exit Criteria

The entry criteria of the following system testing plan is following integration testing, with all test cases receiving the pass status. The exit criteria of the following system testing plan is all tests having been executed.

Pass/ Fail Criteria

A test would be defined as passed if the expected outcome illustrated by the test case is the result of following said test case's steps. If the result of following the steps for the test case is different than what is expected or raises an error, it will be considered as failed.

Test Suspension and Resumption Criteria

Testing will be suspended when a bug, error, or result is found that differs from the expected outcome. Once this issue is resolved, system testing will resume from the beginning.

Risk Analysis

- **Risk**

  Insufficient system testing might lead to unidentified issues with performance, compatibility problems between various environments.

- **Method of Mitigation**

  Create a thorough testing plan for the system that addresses its performance, non-functional, and functional aspects. To run system tests quickly and reliably in a variety of environments and configurations, make use of automated testing tools.

Items to be tested

Test Case ID	TC-29
Test Case Name	Full Application Testing

Pre-Conditions	1. All unit tests passed 2. All integration tests passed 3. User has created and verified their account 4. User has logged into their account 5. User is on the chat page
Test Steps	1. Ask the personal designer to help you find a black shirt for men in a size medium 2. Once matches have been found, press the “Go” button under the first returned item 3. Once redirected, go back to FashionFinder website 4. Press the “Add” button on the first returned item 5. When the Add to Album modal appears, click on the green + button 6. Once the Create Album modal appears, enter in “Black shirts” as the album name and press “Create” 7. Click “Add” button on second returned item 8. Once Add to Album modal appears, select “Black shirts” album and press “Save” 9. Press “Albums” link in navigation bar 10. Once redirected to Albums page, click the blue edit album name button 11. Once Edit Album Name modal appears, in the text box type in “Black clothes” and press “Save”

	12. Click the “Black clothes” album 13. Delete the second item shown within the album by clicking the “Delete” button under the item 14. Once the Confirm Delete Outfit modal appears, press the “Confirm” button 15. Press the Albums link in the navigation bar 16. Press the green Share album button on the “Black clothes” album 17. Logout by clicking the logout button 18. Paste the copied link into your browser
Expected Results	User will be able to successfully: 1. Find black shirts for men in a size medium 2. Go to website selling individual clothing item 3. Create album from chat page 4. Save clothing items to album 5. Edit album name and delete outfits from album 6. Share albums
Priority	High

4. Acceptance Testing

Purpose

To validate whether the software meets the acceptance criteria and satisfies stakeholders' requirements.

Scope

Focuses on user acceptance, usability, and alignment with business goals.

Tools

Manual testing will be used to test acceptance.

Approach

The approach to acceptance testing ensures that the software meets stakeholders' expectations and is ready for deployment. Test cases simulate real-world scenarios to validate requirements and user needs. Testing is conducted by end-users or client representatives, with results compared against acceptance criteria. This process ensures confidence in the software's readiness for deployment and satisfaction with the final product.

Entry/ Exit Criteria

The entry criteria of acceptance testing is following the completion of system testing. The exit criteria of acceptance testing is the completion of all stated test cases.

Pass/ Fail Criteria

For an acceptance test to be considered as passed, the client must approve that each of the requirements stated within the FashionFinder Requirements Specification document have been integrated and meets expectations.

Risk Analysis

- **Risk**

  The provided system may be rejected or cause stakeholders to become unhappy if it does not satisfy their acceptance requirements. Also, application risks not meeting requirements when finalizing it for production.

- **Method of Mitigation**

  Engage stakeholders early in the project regularly to get their input, confirm the specifications, and improve the acceptance standards. Also, meet and confirm requirements in beginning to meet acceptance standards.

Non-Functional Testing

Purpose

Evaluate aspects beyond core functionality such as performance, usability, security, and scalability.

Scope

Assess non-functional requirements to ensure software meets specified standards.

Tools

Manual testing will be used for non-functional testing.

Approach

Systematically test each non-functional aspect according to predefined criteria and methods.

Entry/ Exit Criteria

The entry criteria of non-functional testing is following the completion of functional testing. The exit criteria of non-functional testing is the completion of all stated test cases.

Pass/ Fail Criteria

Tests pass if the system meets specified non-functional criteria; failures indicate areas needing improvement.

Test Suspension and Resumption Criteria

Testing halts if issues arise, requiring resolution before continuing.

Risk Analysis

- **Risk**

  Identify potential issues like performance bottlenecks or security vulnerabilities.

- **Method of Mitigation**

  Employ appropriate testing strategies, adhere to best practices, and address identified risks promptly.

Items to be tested

Test Case ID	TC-30
Test Case Name	Performance - load time
Pre-Conditions	1. The software system is operational. 2. User is logged into account. Enter the following credentials to log in: • Email: “angjelomana1@gmail.com” • Password: “Angjelo1”
Test Steps	1. Interact with the software system by performing various tasks such as

	navigating from a page to another. Click on "about" page, then click on "albums" page, then go back to "FashionFinder" page. 2. Measure the time it took to load the page once clicked. 3. Calculate the average response time.
Expected Results	The average response time should be ≤ 3 seconds.
Priority	Low

Test Case ID	TC-31
Test Case Name	Availability
Pre-Conditions	1. The software system is deployed and operational. 2. User is logged into account. Enter the following credentials to log in: - Email: "angjelomana1@gmail.com" - Password: "Angjelo1"
Test Steps	Monitor the software system for its availability over a specified period (5 minutes) thorough doing various tasks such as chatting, adding items to albums, leaving a review, looking at about page, adding brand and budget in profile.
Expected Results	The system should achieve an availability of ≥ 85%. Meaning for that time period (5 minutes), the system should have respond correctly with no errors for at least 85%.
Priority	Low

Test Case ID	TC-32
Test Case Name	Security
Pre-Conditions	1. The software system is deployed and accessible.

	2. User is logged into account. Enter the following credentials to log in: - Email: “angjelomana1@gmail.com” - Password: “Angjelo1”
Test Steps	1. Attempt unauthorized access to the system. 2. Check for encryption, access controls, and other security features.
Expected Results	The system should protect against unauthorized access and data breaches.
Priority	Low
Test Case ID	TC-33
Test Case Name	Maintainability
Pre-Conditions	Access to the software system's codebase.
Test Steps	1. Implement a minor change to the software system. For example, add another feature under profile and name it “add your measurements.” 2. Save code and run both front-end and back-end servers.
Expected Results	The system should not crash and should not show any errors. “Add your measurements” section will show up on the profile page.
Priority	Low

Test Case ID	TC-34
Test Case Name	Portability
Pre-Conditions	The software system is deployed on various operating systems.
Test Steps	Install and run the software system on different operating systems like Windows and macOS.
Expected Results	The system should function seamlessly on multiple operating systems without modifications.
Priority	Low

Test Design and Execution

We will be using manual testing for the testing process of FashionFinder. Manual tests will be used to test the AI response received through the OpenAI API as user inputs should be as natural as possible to account for the different possible inputs from the user.

The tester will follow the steps shown within the test case tables and if through their test, their outcome is the same as what is mentioned for that specific test case, it will be considered as passed, otherwise it is failed.

Test Data & Defect Management

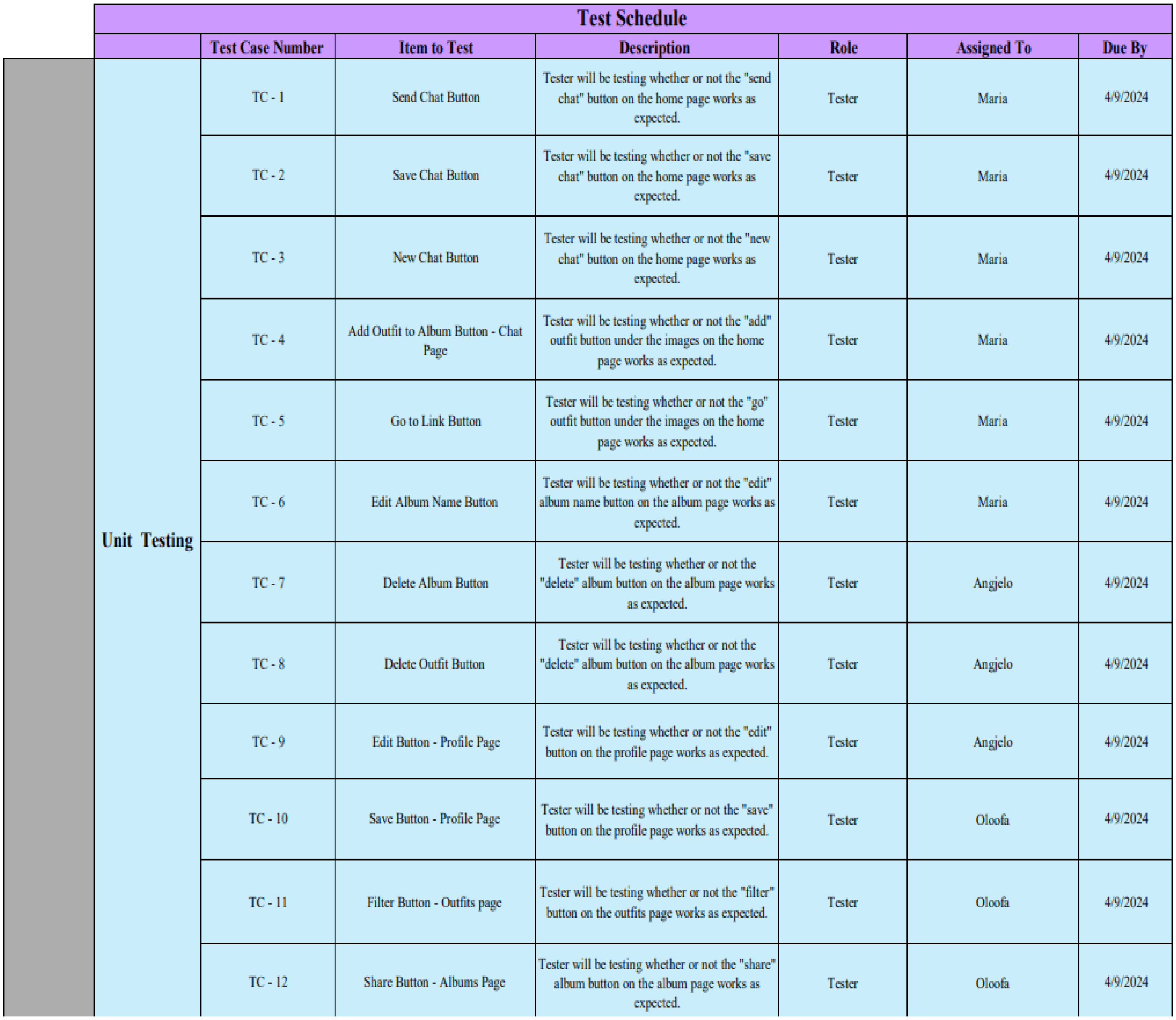
Test Data Management

Test data management involves preparing and maintaining data sets used for testing the FashionFinder system. Test data include user profiles, clothing items, and image datasets. Since the main component of this project requires the AI system to interact with user-provided data such as clothing preferences and style choices, ensuring the privacy and security of sensitive information is crucial. Thus, the test data must be kept private to protect user privacy and follow data protection rules.

Defect Management Process

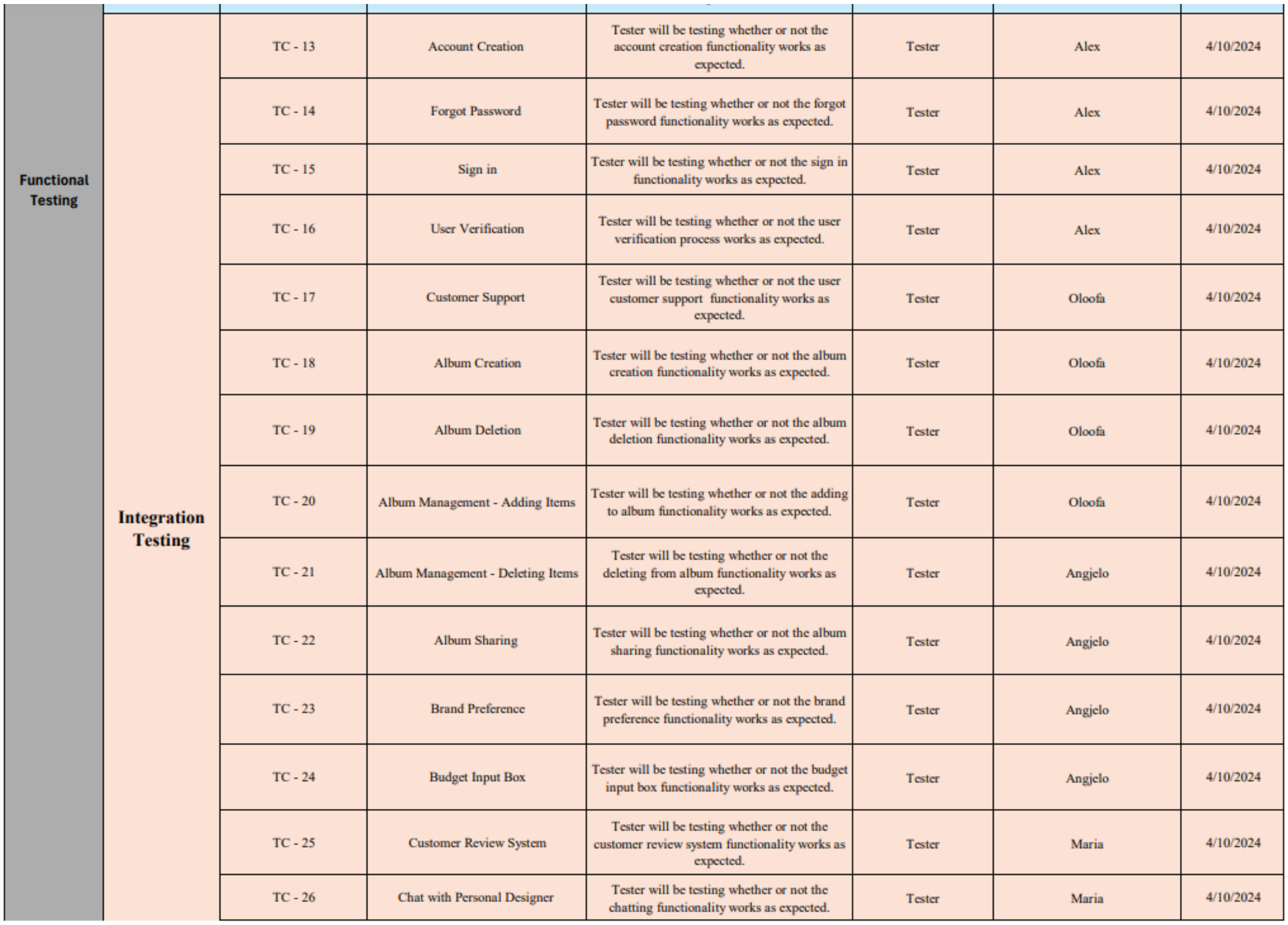
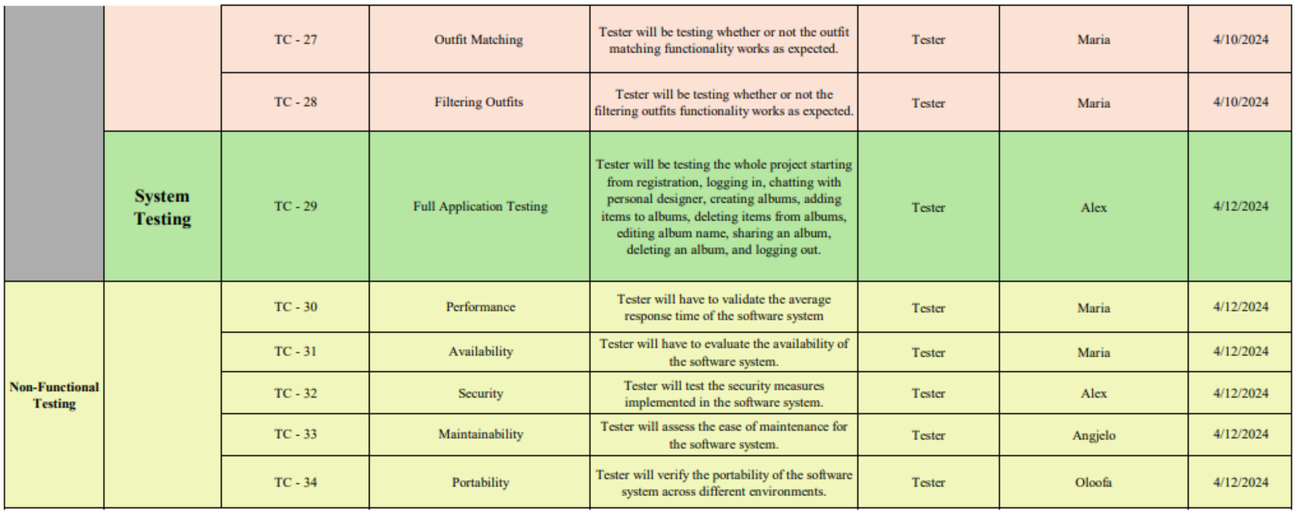
Defect management includes identification, logging, resolution, and verification of issues found during testing. In the FashionFinder project, defects may include inaccuracies in the AI-generated clothing recommendations, user interface glitches, or errors in data processing. The defect management process involves documenting reported issues, categorizing them based on severity and importance, and assigning them to appropriate team members for fixing. Also, the defect resolution process includes thorough testing to confirm that fixes effectively resolve reported issues without introducing other issues.

Test Schedule and Roles & Responsibilities



Test Schedule

	Test Case Number	Item to Test	Description	Role	Assigned To	Due By
Unit Testing	TC - 1	Send Chat Button	Tester will be testing whether or not the "send chat" button on the home page works as expected.	Tester	Maria	4/9/2024
	TC - 2	Save Chat Button	Tester will be testing whether or not the "save chat" button on the home page works as expected.	Tester	Maria	4/9/2024
	TC - 3	New Chat Button	Tester will be testing whether or not the "new chat" button on the home page works as expected.	Tester	Maria	4/9/2024
	TC - 4	Add Outfit to Album Button - Chat Page	Tester will be testing whether or not the "add" outfit button under the images on the home page works as expected.	Tester	Maria	4/9/2024
	TC - 5	Go to Link Button	Tester will be testing whether or not the "go" outfit button under the images on the home page works as expected.	Tester	Maria	4/9/2024
	TC - 6	Edit Album Name Button	Tester will be testing whether or not the "edit" album name button on the album page works as expected.	Tester	Maria	4/9/2024
	TC - 7	Delete Album Button	Tester will be testing whether or not the "delete" album button on the album page works as expected.	Tester	Angjelo	4/9/2024
	TC - 8	Delete Outfit Button	Tester will be testing whether or not the "delete" album button on the album page works as expected.	Tester	Angjelo	4/9/2024
	TC - 9	Edit Button - Profile Page	Tester will be testing whether or not the "edit" button on the profile page works as expected.	Tester	Angjelo	4/9/2024
	TC - 10	Save Button - Profile Page	Tester will be testing whether or not the "save" button on the profile page works as expected.	Tester	Oloofa	4/9/2024
	TC - 11	Filter Button - Outfits page	Tester will be testing whether or not the "filter" button on the outfits page works as expected.	Tester	Oloofa	4/9/2024
	TC - 12	Share Button - Albums Page	Tester will be testing whether or not the "share" album button on the album page works as expected.	Tester	Oloofa	4/9/2024

Functional Testing	Integration Testing	TC - 13	Account Creation	Tester will be testing whether or not the account creation functionality works as expected.	Tester	Alex	4/10/2024
		TC - 14	Forgot Password	Tester will be testing whether or not the forgot password functionality works as expected.	Tester	Alex	4/10/2024
		TC - 15	Sign in	Tester will be testing whether or not the sign in functionality works as expected.	Tester	Alex	4/10/2024
		TC - 16	User Verification	Tester will be testing whether or not the user verification process works as expected.	Tester	Alex	4/10/2024
		TC - 17	Customer Support	Tester will be testing whether or not the user customer support functionality works as expected.	Tester	Oloofa	4/10/2024
		TC - 18	Album Creation	Tester will be testing whether or not the album creation functionality works as expected.	Tester	Oloofa	4/10/2024
		TC - 19	Album Deletion	Tester will be testing whether or not the album deletion functionality works as expected.	Tester	Oloofa	4/10/2024
		TC - 20	Album Management - Adding Items	Tester will be testing whether or not the adding to album functionality works as expected.	Tester	Oloofa	4/10/2024
		TC - 21	Album Management - Deleting Items	Tester will be testing whether or not the deleting from album functionality works as expected.	Tester	Angjelo	4/10/2024
		TC - 22	Album Sharing	Tester will be testing whether or not the album sharing functionality works as expected.	Tester	Angjelo	4/10/2024
		TC - 23	Brand Preference	Tester will be testing whether or not the brand preference functionality works as expected.	Tester	Angjelo	4/10/2024
		TC - 24	Budget Input Box	Tester will be testing whether or not the budget input box functionality works as expected.	Tester	Angjelo	4/10/2024
		TC - 25	Customer Review System	Tester will be testing whether or not the customer review system functionality works as expected.	Tester	Maria	4/10/2024
		TC - 26	Chat with Personal Designer	Tester will be testing whether or not the chatting functionality works as expected.	Tester	Maria	4/10/2024
		TC - 27	Outfit Matching	Tester will be testing whether or not the outfit matching functionality works as expected.	Tester	Maria	4/10/2024
		TC - 28	Filtering Outfits	Tester will be testing whether or not the filtering outfits functionality works as expected.	Tester	Maria	4/10/2024
	System Testing	TC - 29	Full Application Testing	Tester will be testing the whole project starting from registration, logging in, chatting with personal designer, creating albums, adding items to albums, deleting items from albums, editing album name, sharing an album, deleting an album, and logging out.	Tester	Alex	4/12/2024
Non-Functional Testing		TC - 30	Performance	Tester will have to validate the average response time of the software system	Tester	Maria	4/12/2024
		TC - 31	Availability	Tester will have to evaluate the availability of the software system.	Tester	Maria	4/12/2024
		TC - 32	Security	Tester will test the security measures implemented in the software system.	Tester	Alex	4/12/2024
		TC - 33	Maintainability	Tester will assess the ease of maintenance for the software system.	Tester	Angjelo	4/12/2024
		TC - 34	Portability	Tester will verify the portability of the software system across different environments.	Tester	Oloofa	4/12/2024