

FSoftware Test Plan

CSCI-P465/565 (Software Engineering I)

Project Team 16

Suma Priya Davuluri

Yuhan Ke

Shubham Bhagat

Van Thawng

1. Overview

This section provides an overview of the testing approach used to verify the software product.

1.1 Test Objectives

Ensure code works well and returns the correct value.

1.2 Test Environment

The Frontend will be tested by using browsers to inspect element features and then use responsive layouts for different devices. Post deployment, the UI will be testing by accessing the website using different devices. Since java is used for backend code, the backend logic will be tested by writing Junit test cases.

We will also be doing integration testing after integrating different modules.

1.3 Test Personnel

Each groupmate will be responsible for writing test cases to ensure their code works as specified.

1.4 Acceptance Criteria

We will plan our tests so that each test case covers the required functionality. All tests will be expected to pass before the code can be deployed to production. If tests do not pass, the feature will be considered incomplete and will need to be finished.

1.5 Noted Omissions

Our current testing plan is to write both unit and integration tests, so there will not be any major functionality that isn't covered by testing.

2. Test Cases

The test cases are the partitioning of the verification of the software into manageable sections. Often these sections correspond to the set of active use case scenarios but can be organized as the test developer sees fit. Test cases should be in place to cover all the software verification methods. Even non-execution-based testing methods (i.e., inspection/analysis) may be detailed here. The intent is for the test case procedures to provide a repeatable verification of the software specification.

For each test case describe the following:

Number: 1

Name: Detail registration

Description: user can enter details to register

Initial Conditions: Front-end posts user object to API (Servlets)

Input Data: User object

Number: 2

Name: search bar

Description: user can submit a query (keyword)

Initial Conditions: Front-end posts query data to API

Input Data: Query String

Number: 3

Name: user registration

Description: create a new user from received data

Initial Conditions: user data is inserted into the database

Input Data: user data

Number: 4

Name: items creation

Description: create an item profile

Initial Conditions: items data is inserted into the database

Input Data: items data

Number: 5

Name: Edit Product Information

Description: Users should be able to edit product information

Initial Conditions: items data is inserted into the database and product is not rented

Input Data: Product with product id

Number: 6

Name: Delete Product Information

Description: Users should be able to delete products that they no longer want to rent

Initial Conditions: items data is inserted into the database and the product is not rented

Input Data: Product with product id

Number: 7

Name: Google Login

Description: User can login using their google login

Initial Conditions: User has a Gmail account

Input Data: Gmail ID and OAuth authentication token

Number: 8

Name: Forgot Login

Description: User can reset their password

Initial Conditions: User must remember their security question answer

Input Data: Login/Forgot password

Number: 9

Name: Map

Description: User can locate where the products are located at

Initial Conditions: Product must be listed, and location must be verified

Input Data: Product with id and location

Number: 10

Name: Admin Product Approval

Description: Admin can approve products

Initial Conditions: Product must be uploaded by the seller

Input Data: Product with all product details

Number: 11

Name: Product details page

Description: User can view all the details of an available product

Initial Conditions: Product must be approved by the admin and listed

Input Data: Product with all product details

Number: 12

Name: Rent product

Description: User can rent a product

Initial Conditions: Product must be approved by the admin and listed

Input Data: Product with all product details

Number: 13

Name: Chatbot

Description: User can ask their queries to the chatbot

Initial Conditions: Chatbot integrated with the website

Input Data: User query

Number: 14

Name: Speech to text conversion

Description: User can use voice recognition to populate the search field

Initial Conditions: user allows permission to use microphone

Input Data: User's voice command

Number: 15

Name: Message

Description: Seller and renter can have a 1 to 1 conversation about the product

Initial Conditions: Product needs to be listed and both parties need to be registered

Input Data: Renter's information and seller information

Number: 16

Name: Add Review

Description: Renters can add review for a product they have rented

Initial Conditions: Product needs to be listed

Input Data: Renter and product information

Number: 17

Name: Add Complaint

Description: Renters can add complaints about a product they have rented

Initial Conditions: Product needs to be listed

Input Data: Renter and product information

Number: 18

Name: Analytic

Description: Admin can see which products are rented in what region in a chart format

Initial Conditions: Product needs to be listed with appropriate location

Input Data: Product information

Number: 19

Name: Product Filtering

Description: Users can filter products to see products they wish to rent faster

Initial Conditions: Products need to be listed

Input Data: Filtered data of desire state, city, price range, price sort, or rating.