**CS673 Software Engineering** 

**Team 3 - ZicZac**

**Tests Report**

|  |  |  |  |
| --- | --- | --- | --- |
| Team Member | Role(s) | Signature | Date |
| Dinara Tiyekbayeva | Configuration Leader | *DT* | 04/07/2021 |
| Elijah Curme | Team Leader | *EC* | 04/07/2021 |
| Pelin Akbiyik | QA Leader | *PA* | 4/7/2021 |
| Chenghao Feng | Security Leader | *CF* | 04/18/2021 |
| Jay Hwang |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Revision history**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Author** | **Date** | **Change** |
|  |  |  |  |
|  |  |  |  |

[Introduction](#_87t9hln2vjz0)

[Test Summary](#_sm5odwyvuk3j)

[Tests Reports](#_pqso2mbjyzx4)

[Testing Metrics](#_mtfbusfb0eq3)

[References](#_15tmymhipvdv)

[Glossary](#_8n34lvocupub)

# Introduction

This is a document that describes in details test strategy, objectives, schedule and

deliverables of Ziczac project.

Ziczac is a dynamic trading application that is built on python flask backend and

reactjs frontend. It is deployed through heroku.

# Test Summary

For our backend code we used Python framework unittest. Each developer will be

responsible for testing code components/functions.

With the unittest framework we tested features such as login and signup.

In login we ensured that if a user logins with correct credentials, the welcome message will appear.

In signup test we ensured that if certain data such as username, email and password were sent to backend database, that data indeed was recorded in the database and upon querying expected result is returned.

For the system testing a QA leader will have a dedicated spreadsheet we talk about in chapter below.

For a web system testing we are using Selenium.

QA leader will be responsible for the schedule and deliverables of selenium testing.

Regression testing will be done after each refactoring using both unit testing and selenium where applicable.

# Tests Reports

In this section, you will give a detailed description of each test case performed and the result. You shall list what are existing tests developed in the previous semester and what are new tests developed currently.

Here is a sample template that can be used for each test case. For system tests or acceptance tests, you may also include some screenshots.

**UI Test (Manual)** date: 3/05/2021

Test name: Starting the webapp

· New or old: New

· Test items: (what do you test ) Testing if the react app will launch correctly, and the homepage will be displayed once the app is activated.

· Test priority (high/medium/low) High

· Dependencies (to other test case/requirement if any): None

· Preconditions: (if any) None

· input data: Code to run the react app in terminal.

· Test steps:

o Install necessary tools to run react code

o Run npm start command in terminal

· Postconditions:None

· Expected output: ZicZac webapp homepage displayed

· Actual output: Ziczac webapp homepage displayed

· Pass or Fail: Pass

App Login:

● Test case ID: Login

● New or old: NEW

● Test items: An existing user with a registered email and password can successfully login

● Test priority (high/medium/low): HIGH

● Dependencies (to other test case/requirement if any): None

● Preconditions (if any): User is registered in MidasTouch system

● input data: Email, Password

● Test steps:

○ Activate flask app and navigate to home page

○ Click “login”

○ Enter credentials

● Postconditions: None

● Expected output: User Profile page

● Actual output: User Profile page

● Pass or Fail: PASS

New User Signup:

● Test case name: Signup

● New or old: NEW

● Test items: (what do you test ): New user registration

● Test priority (high/medium/low): HIGH

● Dependencies (to other test case/requirement if any): None

● Preconditions: (if any): None

● input data: Email address, username, password, confirm password

● Test steps:

○ Activate Flask app and navigate to home page

○ Click “login”

○ Under “New User”, click “Click here to register”

● Postconditions: None

● Expected output: Flask message indicating successful registration

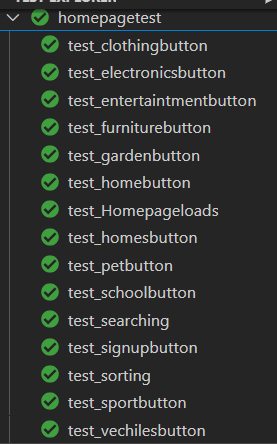
● Actual output: Flask message indicating successful registration

● Pass or Fail: PASS

**Automated UI Testing (Selenium)**  Dates: 3/24-4/7

**Test case:** Testing all the features of home page including, buttons, search function and sorting items by price and viewing recently added items.

Result:





Summary: Recently added items is selected by default in the drop down on the home page, but when the user clicks submit the result does not appear. When the user selects the recently added items from the drop down and clicks submit the result does not appear, however when the user selects price in the drop down and then recently added items and clicks submit, the items appear on the home page. Furthermore, the most recent item does not appear at the beginning of the list.

**Testcase:** Testing if the buttons on account page are functional

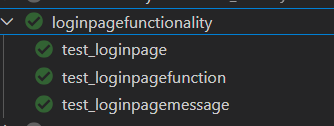
Result:



Summary: All the buttons on account page is functional

**Test case:** Login page

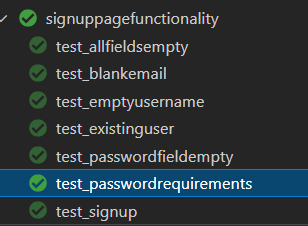
Result:



Summary: No bugs in Login page. Correct error messages are displayed in each scenario. If the username is taken, the user cannot sign up. If one of the fields is empty, user receives an alert saying that they need to enter all the fields. If the password requirements are not met, user receives and error message saying that they do not meet the requirements.

**Test case**: Signup page

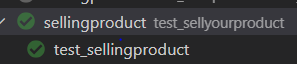
Result:



Summary: All user scenarios were tested and no bug was found.

**Test case:** Sell your product page

Result:



Summary: Test result is a pass. All the fields work correctly and the user is able to post an item to sell. The item appears on the application when searched using the search bar.

**Test case:** Testing if the items appears in the correct category once posted by the seller

Result:

****

Summary: The test result is a fail because the item appears only if there are less than four items in the category. Currently, each category can have only upto 4 items. This is an easy fix but currently affects the testing because if the tester runs multiple amounts of tests for the same category, it fails.

**Summary:**

For Iteration 3, we aim to improve recently added sorting functionality in homepage, increase number of item capacity of each category, make contact us and chat box functional.

Adding user profile page, making rating feature on seller’s profile functional and making like and add to cart buttons functional on product page. Fix the bugs.

**Testing since Iteration 2** Dates:4/07-4/28

**UI Test (Manual)**

**Rating feature**

Test name: Rating Seller date: 4/20/2021

· New or old: New

· Test items: (what do you test ) Testing if the user can rate the seller.

· Test priority (high/medium/low): Low

· Dependencies (to other test case/requirement if any): User having an account and being able to login

· Preconditions: (if any) None

· input data: Code to run the react app in terminal.

· Test steps:

o Login as a user

o Select the number of stars to rate the seller

· Postconditions:None

· Expected output: Seller has a rating

· Actual output: Rating seller feature was not functioning

· Pass or Fail: Fail

**Like feature for the seller**: date: 4/20/2021

Test case ID: Send likes to seller

New or old: NEW

Test items: An existing user with a registered email and password can successfully send a like to seller.

Test priority (high/medium/low): Low

Dependencies (to other test case/requirement if any): User having an account and being able to login

Preconditions (if any): None

input data: None

Test steps:

○ Login as a user

○ select a seller to send a like

Postconditions: None

Expected output: Number of likes that a seller receives increases as a result of clicking like

Actual output: Unable to send a like

Pass or Fail: Fail

**Private chat system**: date: 4/26/2021

Test case name: Private Chat system

New or old: NEW

Test items: (what do you test ): Testing if the buyer and seller communicate through private chat system

Test priority (high/medium/low): HIGH

Dependencies (to other test case/requirement if any): User having an account and being able to login

Preconditions: (if any): None

Input data: Email address, username, password, confirm password

Test steps:

○ Login as buyer

○ Select an item

○ Click chat with button

○ Send message to seller

○ Login as a seller

○ Select My messages under Account drop down

○ View the message from buyer and reply

○ Login as a buyer

○ Select My messages under Account page

○ View the message from the seller

● Postconditions: None

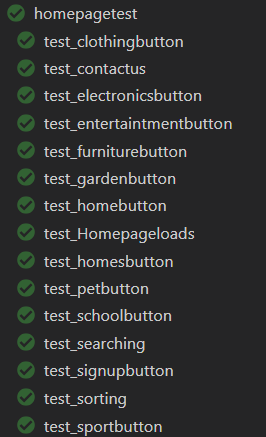
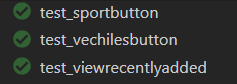
● Expected output: Buyer and seller able to send and receive messages

● Actual output: Buyer and seller able to send and receive messages

● Pass or Fail: PASS

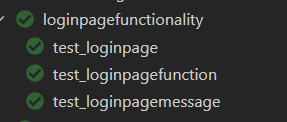
**UI Automated tests** Dates:4/07-4/28

**Test case:** Testing all the features of home page including, buttons, search function and sorting items by price and viewing recently added items.

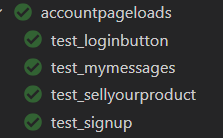
Summary: All test cases are successful. All the features in homepage works as expected.

**Test case:** Testing all the features of Login page. Testing includes username and password match for successful login. Also display of message alerts if the requirements are not met.

****

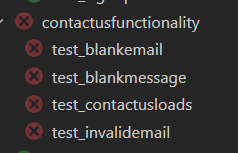
Summary: No bugs in Login page. Correct error messages are displayed in each scenario. If the username is taken, the user cannot sign up. If one of the fields is empty, user receives an alert saying that they need to enter all the fields. If the password requirements are not met, user receives an error message saying that they do not meet the requirements.

**Test case:** Testing the functionality of my account drop down and the buttons under the drop down



Summary: No bugs found. All the buttons are functional under My Account drop down.

**Test case:** Testing the requirements to send a message to customer service



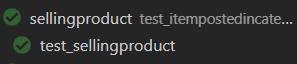
Summary: The requirements to send a message to customer service have not been implemented. The test result is fail.

**Test case:** Testing if the item shows up under recently added display on homepage after the seller posts an item



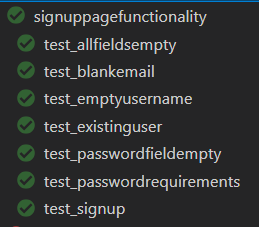
Summary: The test result is fail. The new item does not appear in the recently added items

**Test case:** Testing if the items appear in the appropriate category after being posted.



Summary: The test result is a pass. The new item appears in the correct category.

**Test case:** Testing if the sign up page works. Testing Includes unique username, password requirements, and entering valid data in each field. Also error message display if the requirements are not met.



Summary: The test result is a pass. All the features of the sign up page work as expected.

* Testing Metrics

Iteration 2: Two out of thirty tests failed. This is 93% pass rate.

Iteration 3: 44 test cases have been created. 14 more test cases since iteration2. 10% higher than iteration 2.

Functionality coverage is 100%. Every single feature has been tasted.

Functionality of contact us button, Functionality of chat button, forgot password

in login page, seller rating, Like button for seller have been included in iteration3.

Defect rate:

Including all manual and automated tests, 20% defect rate.

# References

# Glossary