**Prepared by:**

David Zazulak

Eric Jennings

Alejandro Lara

Amanda Gonzalez

21 February 2022

**OBJECTIVES AND TASKS**

**Objectives**

As a team, we will be adding Unit Testing, Integration Testing, and End to End Testing. We have a stretch goal of adding performance testing using JMeter. We will be using discord to communicate on a daily basis, and a requirements traceability matrix to keep track of our tests.

**Tasks**

Provide some information about each task you’ve defined in your overall testing strategy. This section can include, for instance, details about testing, test result analysis, reporting, etc.

We will use the default reporter for testng to show which tests pass or fail in an html page. We will use the requirements traceability matrix for bug reporting using an Excel spreadsheet.

**SCOPE**

**General**

Describe what exactly you’ll be testing and in what depth. Are you testing all of the units of products, the integration of all units, etc?

We’ll be testing the dao layer, and the service layer to make sure we have unique amount of unit test and as well doing some integration testing for the front end.

**Tactics**

As a team, we will divide the work accurately to make sure everyone gets to participate into the project, and communicate with each other through discord, and emails for any additional files we need to send. We will look at the files and decide who will receive which to create unit test, and integration test.

**TESTING STRATEGY**

Our main goal is to provide at least 40% coverage of all the features. We will be creating multiple happy and sad path tests for each feature, ensuring that they are each thoroughly tested and if we are unable create a working test it will be noted in the traceability matrix. In order to achieve these goals, we will be using the agile methodology conducting daily standups to set goals for the day and keep track of progress. To keep track of code coverage we will be using the built-in feature in IntelliJ that tells us how much of the code is being tested.

**Unit Testing**

Definition:

After checking the code coverage it is clear that the controller methods are in need of testing. After determining that the coverage relieved that controller functions required some testing, we divided the classes to the two participants, and used spring boot testing along with Junit. Running each individual test to either pass or fail will determine how the results come along, but overall run the entire controller test layer to see if they all pass, as well review the coverage once more to determine if there was good amount of testing.

Participants: Amanda Gonzalez/Alejandro Lara

Methodology: Unit Tests Success and Failure

We will be using JUnit to conduct our unit testing along with Spring boot Testing.

Spring boot Testing, basically grabbing the parameters and already having security and enhancing the test cases to be used in our JUnit tests.

**System and Integration Testing**

Definition:

The system and integration testing for this project are the tests to see if the application is fully integrated and working with the dev server using the API calls defined in the controller.

Participants: David Zazulak/Eric Jennings

Methodology: Postman Success and Failure

We will be using Cucumber with Selenium for the E2E tests to confirm that the features are working as intended.

In Postman, create a new user, then test all the functions with that user, then create failure tests for each function.

HARDWARE REQUIREMENTS

We are each running tests on our local machines in Windows.

ENVIRONMENT REQUIREMENTS

We are all running tests that could adversely affect each other if we were running them on the same server at the same time. To avoid conflict, the tests will run on a local implementation of the program. The backend will run on IntelliJ. The frontend will run in VSCode with “npm run start” in the terminal.

TEST SCHEDULE

There is not enough information to estimate the time required for each milestone.

Completed success tests in Selenium.

Completed success tests in Postman.

Completed coverage of success tests in the Controller layer.

Completed sad path tests in Selenium.

Completed sad path tests in Postman.

Completed sad path tests in the Controller layer.

(Stretch) JMeter testing

CONTROL PROCEDURES

Problem Reporting

Our job is to test and not change the code of the original program. Everyone is working mostly individually on their own layer. A bug report document has been added for bugs as we have been developing.

FEATURES TO BE TESTED

* Ability to register as a customer
* Ability to login as an customer or administrator
* Ability to view appropriate interfaces upon login dependent upon level of access (admin or customer)
* Ability to add a product to storefront as an administrator
* Ability to add a discounted product to storefront as an administrator
* Ability to update or remove a product from storefront as an administrator
* Ability to update or remove a discounted product from storefront as an administrator
* Ability to view account information as a user
* Ability to view the storefront as a user (including guest)
* Ability to view the discounted items as a user (including guest)
* Ability to view the product description page as a user
* Ability to add items to cart as a user
* Ability to select quantity of a particular item to add to cart as a user
* Ability to checkout as a user
* Ability to logout as a user
* Ability to add and view customer reviews as a user
* Ability to view orders history as a user

FEATURES NOT TO BE TESTED

We will not be testing unfinished features, we are not entirely sure which features are completed at the moment but we will more focus on making sure the working features are well tested.

RESOURCES/ROLES & RESPONSIBILITIES

Define the roles and responsibilities for this project and specify who will be taking on those roles and responsibilities. Who, for instance, will design the test cases? Who will set up the test environment?

SCHEDULES

Deliverables

* Test Plan
* Requirements Traceability Matrix
* Bug Report

DEPENDENCIES

For integration there are some functions that are not entirely clear. The program is local so it is not signed into an S3 bucket for displaying images.

RISKS/ASSUMPTIONS

We have a 1 week timeline with a program that we have never seen before. We have 4 personnel to write tests. We are splitting along the layers(unit tests, integration tests, and end-to-end tests), so that the work of each person does not interfere with the others and everyone can focus on their own tests.

TOOLS

Which automation tools and bug tracking tools are you using?

Selenium

Postman

IntelliJ Coverage

Requirements Traceability Matrix.

APPROVALS