Test Plan

(IEEE 829-1998 Format)

## Test Plan Identifier

Master-7.8.21-1.00

Contact Information:

Eduardo Valencia

[8025918@gmail.com](mailto:8025918@gmail.com)

## Introduction

The e-commerce application must be tested at the design and application level. This document provides an outline of the testing methodology and describes what should be tested. Specifically, it will describe the test methodology for the following areas related to the software’s project plan: the products list page, product details page, cart page, and checkout functionality.

You can view all project tasks and their requirements at <https://www.notion.so/alex44/CSC318-Project-ecaef5578bf74bfe86709a83349a78eb>.

## Test Items

* Products list page
  + It must display the products.
  + The products should be organized by category.
* Product details page
  + It must display the product’s title, description, images, color options, and size options.
  + It must have a button to add an item to the cart.
* Cart page
  + When the user adds an item to their cart, it must display the items.
  + When users change the item quantities, the cart must display the updated quantities.
  + If the user removes an item from their cart, it must not show the removed item.
* Checkout functionality
  + After the user successfully pays for items, the application must create a new order in Strapi.
  + It must clear the cart after the user checks out.

## Features To Be Tested

* Homepage
  + Users should be able to see the list of products on the homepage (HIGH)
    - They should be able to click on each product to see the product details page.
    - They can see the product’s thumbnail.
  + Users can click on the quantity buttons on each product on the homepage (MEDIUM)
    - It should show the updated quantity.
* Product Details Page
  + Users can see the following:
    - Title (HIGH)
    - Price (HIGH)
    - Color options (MEDIUM)
    - Size options (MEDIUM)
    - Description (LOW)
  + Users can select a size option or color option (MEDIUM)
  + Users can see an “Add to Cart” button and click on it. (HIGH)
  + After clicking on the “Add to Cart” button, users can see a message indicating the item was added. (MEDIUM)
* Cart Page
  + Users can see the list of products in their cart (HIGH)
  + Users can see the following about each product:
    - Title (HIGH)
    - Price (MEDIUM)
    - Thumbnail (MEDIUM)
  + Users can click on each product to navigate to the product details page (LOW)
  + Users can see the total (HIGH)
  + Users can see the subtotal (LOW)
  + Users can see the checkout button and click on it to check out (HIGH)
  + Users can see an error when the payment fails (LOW)

## Features Not to Be Tested

* Checkout design
  + The checkout design is controlled by a third-party library, so it will not be tested.
* Storing the size options and color options
  + We will not test whether the selected size and color options are stored because we do not have a mechanism for shipping out certain sizes or color options. This feature is not part of this release.
* Strapi admin page
  + The admin page is controlled by a third-party library, Strapi, and has been thoroughly tested by its developers.

## Approach

* Tools
  + Web Browser. It should not require special training.
* Metrics
  + The number of errors by page.
* Configurations
  + Browsers
    - Chrome
    - Safari
    - Firefox
  + Resolutions
    - Small (320px)
    - Medium (768px)
    - Large (1440px)
* Regression Testing
  + Considering we are already at the final stage of this application and will begin testing after the application’s completion, we will not use regression testing.
* Clarifying Requirements and Handling Untestable Requirements
  + We will report untestable requirements to the team. If they are untestable, we will define specific criteria for testing them.
  + If requirements do not make sense, the team can ask Eduardo Valencia for clarification.
* Overall Project Testing Approach
  + Design
    - Check the design requirements on Notion.
    - Verify that the design has all components listed in the requirements.
  + Development
    - Check the page requirements on Notion.
    - Validate that the page has all features and components listed in the requirements.
    - Using the design’s resolution, check the page’s design against the existing page.
* Testing Groups
  + Homepage
  + Product Details Page
  + Cart Page
  + Checkout
* Significant Testing Constraints
  + Time
    - We are already in the final stage of this application, so we did not have time to test the prior stages.

## Item Pass/Fail Criteria

* All high-priority requirements were completed.
* At least 90% of requirements with a medium priority were completed.
* At least 80% of low-priority requirements were completed.
* Defects
  + We can measure the number of defects by finding inconsistencies between the design and the application. However, we will only measure **high-priority** defects that can lead to inconsistencies between the project requirements and the application.
  + No high-priority defects should be found in the application.

## Suspension Criteria and Resumption Requirements

* We will pause a group of tests once we determine the pass / fail state for all requirements in that group.
* Once Eduardo Valencia implements a fix or change, he will let the team know what items changed. Once testing resumes, the team will only test the affected items to avoid testing the same requirement multiple times.
* We can stop testing for a group after we meet the aforementioned pass/fail criteria.

## Test Deliverables

The following must be delivered as part of this plan:

* Test plan
* Test requirements (the individual requirements we must test for)
* Test logs
  + Each tester must create a page on Notion describing the results of the test.
  + They must document which test criteria failed.
* Change requests
  + Eduardo Valencia will explain what changes he made to the application after he implements a change or fix.
  + He will list the affected features and components.

## Test Tasks

The following describes the test tasks. You can find each page’s criteria under “Features To Be Tested.” All team members have the skills to complete any of the test tasks.

* Test homepage
  + Depends on homepage completion.
* Test product details page
  + Depends on page completion.
  + Depends on cart functionality.
* Test cart page
  + Depends on the following:
    - Page completion
    - Product details page
    - Homepage
    - Cart functionality
* Test checkout page
  + Depends on the following:
    - Cart page completion

## Environmental Needs

* No special test data must be provided.
* No special hardware will be required.
* Tools
  + Web Browser
  + Polypane (for responsive testing)
  + Lambda Test (for cross-browser tests)
* Communications
  + Web
  + Strapi CMS and Front-End

## Responsibilities

Eduardo Valencia-Project Test Lead

* Responsible for oversight of testing on the project.
* Accountable for the processes used to ensure the quality of the deliverable.

Michael Esan-Testing Manager

* Accountable for conducting quality assurance testing and executing the test plan.

Larry Alexander- Test Approver/Tester

* For reviewing, validating, and approving the test materials created by the Test Designer.
* For reviewing reports from the testers and determining what subsequent actions will be taken.

Treylon Wofford-Reviewer/Test Designer

* Responsible for creating the test scripts, scenarios, test lives, and so on that make up the tests to be performed.
* Responsible for reviewing reports from the testers and determining what subsequent actions will be taken.

## Staffing and Training Needs

Testing should be done by the tester. The tester should conduct testing on each system. The tester assigned should have basic knowledge of the e-commerce platform.

## Schedule

* July 13: Test the homepage.
  + Estimated time: 15 minutes.
  + Team members
    - Michael
    - Larry
* July 13: Test the cart page and checkout functionality.
  + Estimated time: 30 minutes.
  + Team members
    - Eduardo
* July 14: Test the product details page.
  + Estimated time: 15 minutes.
  + Team Members
    - Michael
* July 15: Review the test reports and describe necessary changes.
  + Estimated time: 30 minutes.
  + Team Members
    - Treylon

If we begin to fall behind schedule, Treylon or Eduardo will take over testing for that day.

## Risks and Contingencies

1. We may not have enough time to develop the app.
2. We may not have enough time to develop the checkout points.
3. We may not have to fully develop the cart.
4. Team members may not understand the criteria to test the application.
5. Testers may not test the criteria thoroughly, which may leave defects unnoticed.

|  |  |
| --- | --- |
| **Risk Mitigation Plan** | **Risk Contingency Plan** |
| You identify actions which you will take in advance irrespective of the occurrence of risk | You plan actions, but you monitor certain warning signs. You take these actions only when you see the warning signs. |
| You spend time and money in advance for the given risk condition | You do not spend time or money in advance, but you keep them ready, and invest them when needed |
| We are expected to mitigate the risks which are outside the risk threshold. By applying a mitigation plan, we reduce the probability of impact of the identified risk. | By identifying the contingency plan, we do not change the probability or impact of the current risk, but we plan to control the impact as risk event looks like occurring. |
| This works as the first level of defense for the high exposure risks | This works as a fallback plan for the high exposure |

<https://www.linkedin.com/pulse/know-difference-between-mitigation-plan-contingency-saad-alqurashi>

## Approvals

To approve the master test plan, all team members should approve the application. Although our team members do not have the same technical skills, they should be able to determine whether the application is functional from a user’s perspective.