Technological Institute of Costa Rica

Computer Engineering School

Project Test Plan

Software Quality Assurance

I-2024

Professor: Erick Hernández Bonilla

Members:

María Paula Bolaños Apú

Marco Herrera González

Jeffrey Leiva Cascante

Content

Table Index

Revision Log

Date	Version	Changes
March 22, 2024.	1.0.	Initial draft released for review.

Overview

Introduction

This document details the test plan for project #1 of the Software Design course, whose objective is the development of a web system for image publications and sales of beauty and skincare products. The objective of the test plan is to specify the strategy, procedures, activities, and tests that will be performed to ensure the attributes of functionality, performance, and maintainability of the project mentioned before. This document is directed to the personnel responsible for executing the test plan, the Project Manager, and any other interested in the quality assurance of the project.

Scope

This document covers testing for all levels of the project, that is, the requirements specification, the software architecture specification (that includes user stories, the system architecture design, and the database design) and the source code. The document covers dynamic testing for just one module of the system, including only unit testing.

Context of Testing

Project

The project consists of the following objects:

- Requirements specification document.
- Software architecture design document.
- Source code of the system.

The system is divided into the following modules:

- User module: a module that provides functions such as user registration, user login, user profile edition, and password recovery.
- Publication gallery module: a module that allows for the view, registration, and edition of image publications.
- Product gallery module: a module that allows for view, registration, and edition of products available for sale.
- Category module: a module that allows for the view, registration, and edition of product categories and subcategories.
- Purchase module: a module that provides functions to manage the user shopping cart and purchases.
- Calendar module: a module that provides functions to manage a calendar, including the view, registration, and edition of events in the calendar.
- Notification module: a module that provides functions to generate and view notifications for the users.

- Orders module: a module that allows for the view, registration, and edition of the client purchase orders.

The requirements specification document and the software architecture document will be referred together as the project documentation. All modules of the system may be referred together as the system.

Test Items

The test plan for this project includes testing of each object of the project as listed in the previous section, testing of each system module listed above, and testing of the functionality, performance, and maintainability of the software system.

Test Scope

The complete documentation of the system will be reviewed. All features of each module will be tested, including the three layers of implementation, that is, the user interface, the backend, and the database. System testing will be focused only on functionality, performance, and maintainability as quality attributes; no other quality attributes will be tested. Static testing will be applied to both documentation and the system. Dynamic testing will only be applied to one module of the system.

Test Basis

The requirements of the system are captured in the software requirements document. The user stories, system architecture design, and database design are captured in the software architecture document. The source code of the system is available in the online repository of the project.

Assumptions and Constraints

CO-1: All tests have to be automated as possible using technological tools for such purpose.

CO-2: Static testing has to be applied to all documentation and the system.

CO-3: Dynamic testing can only be applied to one module of the project.

CO-4: Exploratory testing can be used if properly justified.

AS-1: It is assumed that all team members have had any sort of prior training in the technological tools that will be used for testing.

AS-2: It is assumed that all team members have the necessary tools and equipment for testing.

AS-3: It is assumed that all data and environment requirements will be satisfied by the time testing beings.

Stakeholders

The Product Owner and Project Manager are stakeholders for approval purposes. The Testing Team is responsible for executing tests, preparing the test process documentation, preparing the test results reports, and any other activity related to testing.

Risk Register

The following abbreviations are used in the risk tables:

P: probability of the risk.

C: consequence, impact or effect of the risk if materialized.

S: severity of the risk, calculated by multiplying the probability by the consequence.

The scales for both probability and consequence will be 1 to 10, where 10 is the highest.

Product Risks

Table 1 - Product Risks

Risk ID and Name	P	C	S	Mitigation Activities
PD-1: Test data is of poor quality and insufficient.	5	5	25	Write data requirements as clear as possible. Review data requirements. Establish quality criteria for data requirements.
PD-2: The test environment setup is complicated to use and install.	4	5	20	Review environment requirements with the Testing Team. Consider current technologies and tools for testing environment.
PD-3: System documentation is incomplete, ambiguous, or difficult to understand.	5	6	30	Review all documentation to ensure completeness and understanding.
PD-4: Source code is poorly organized.	4	6	24	Review of the source code organization with the Development Team.
PD-5: The test environment setup is insufficient or complicates the testing objectives.	5	7	35	Review of environment requirements with the Testing Team. Ensure that the test environment is sufficient to verify and validate the test objectives. Suspend testing activities, reformulate the test environment, review the new environment, and resume testing activities.
PD-6: Test strategy and cases are insufficient, incomplete or cannot ensure the quality attributes designed and the overall quality of the project.	5	9	45	Thoroughly review of the test strategy and test cases. Verify that the test strategy and test cases contribute to the verification of the quality attributes designated. Suspend testing activities, reformulate the test strategy and cases where necessary, review the new plan, and resume testing.

Project Risks

Table 2 - Project Risks

Risk ID and Name	P	C	S	Mitigation Activities
PJ-1: The Testing Team has no knowledge to use and understand the technologies used to implement the system.	4	3	12	Specify the technologies used in the project before the Testing Team acquisition. Request team training on desired technologies if necessary.
PJ-2: The Testing Team has no training to use the testing tools that will be used in the system.	4	3	12	Specify the training needed before the Testing Team acquisition. Request team training on testing tools if necessary. Allocate additional time for initial use and testing with the testing tools.
PJ-3: Data requirements are not supplied on time.	4	7	28	Hand data requirements before the stablished deadline in the project. Follow progress of data requirements activities. Specify the consequences of schedule violations.
PJ-4: The Testing Team does not have the necessary tools to install and use the test environment.	3	3	9	Verify the current available tools for testing. Request the Testing Team to be provided with the necessary tools for testing.
PJ-5: Testing activities are complex, complicated and the expected testing time can delay other activities of the project.	5	8	40	Review of the test plan, test cases and time estimations. If time estimations cannot be reduced if requested, report to the Project Manager to evaluate the necessity to reformulate the testing activities. Suspend testing activities to reformulate the test strategy to a strategy that adjust to the project time constraints, while keeping the quality assurance of the product. Then, resume testing activities.
PJ-6: Testing activities are taking 20% more time executing that planned.	5	8	40	Follow test activities closely. Verify the expected progress of the test plan against the actual progress. Evaluate the causes of delaying in the testing activities and report the findings to project management. Request more resources (personnel or tools) if necessary.

Test Strategy

Test Levels and Types

The tests for project documentation will include inspections. The system will include static testing for all modules, specifically code inspections with automated tools, and exploratory testing when justifiable. The purchase module will be the only module that includes dynamic testing, where only unit tests will be applied. All testing activities (except exploratory testing and documentation inspections) will be performed using technological tools for automation.

Test Deliverables

The deliverables will be the following:

- Project test plan document.
- Software requirements document inspection report.
- Software architecture document inspection report.
- Code inspection report.
- Test model specification document for all test levels.
- Test cases and test procedures for dynamic testing.
- Dynamic testing results and automatic static testing results report.
- Package of scripts used for testing.

Test Design Techniques

The design of the test plan is specified as follows:

- Inspections will be the technique used for documentation and code of all software modules. The results of the inspections will be documented via an inspection report.
- To ease code inspections, automation tools have to be used to review the code base in order to find defects and ensure the quality attributes defined for the project.
- The automation tools that have to be used will be defined in the environment requirements section of this document.
- Testing Team members have the responsibility to plan, manage, execute, and document the results of code automated testing.
- The automation of the code base inspections will be performed using technological tools.
- To design test cases for dynamic testing, the techniques that will be used are equivalence partition, boundary value analysis, invalid values, decision tables, and structural tests.
- Both black box and white box tests will be considered for dynamic test cases.

- Functional and performance tests are required for the module that will undergo dynamic testing.
- Testing Team members have the responsibility to plan, design, execute, and document the results of test cases and procedures for dynamic testing.
- Exploratory testing can only be applied in the static tests if there is no way to automate it with technological tools.

Entry and Exit Criteria

Entry criteria is defined as follows:

- All testing activities can start after the test plan has been approved and signed off.
- The software requirements document can undergo inspection as soon as it has been completed and handed to the inspection team.
- The software architecture document can undergo inspection as soon as it has been completed and handed to the inspection team.
- Code inspections can start as soon as the code is available in the repository.
- Dynamic testing can start as soon as the data and environment requirements are prepared and ready for execution.

Exit criteria is defined as follows:

- Inspections are considered complete when all their defined activities have been accomplished and the completion criteria for the inspection is accomplished.
- Documentation testing is complete when all document inspections have been completed and the completion criteria for the inspection is accomplished.
- System testing is complete when code inspections for all modules are finished and approved, and all high priority test cases for dynamic testing have been passed.
- Overall project testing is complete when both documentation and system testing are completed, and all high priority defects found have been corrected.

Test Completion Criteria

The test completion criteria is the same as the exit criteria defined in the previous section. Overall project testing is completed when all deliverables have been completed, approved, and handed to the Project Team.

Degree of Independence

Unit testing will be conducted by the Development Team, who is under the charge of the Project Manager. Documentation and code inspections will be conducted by the team assembled by the Inspection Leader, who is independent of the Project Team. Inspection teams will consist of independent testers, that is, they do not participate in any means in the Project Team.

Metrics to be Collected

No metrics will be collected as part of this test plan.

Test Data Requirements

The following data requirements are needed for testing:

- 1000 user accounts (998 with client role and 2 with administrator role). The passwords for the users have to be generic passwords (preferably the same password for all users), and they have to be available to the Testing and Development Team.
- 100 products available in inventory.

The previous data has to be sanitized and be available in the test environment. The data has to be available through the entire testing period. The data has to be inserted in the MongoDB Atlas database instance used for testing. The responsible for preparing and delivering the data requirements is the Development Team. To report the readiness of data requirements, a member of the Development Team will make such report via an email sent to all members of the Testing Team.

Test Environment Requirements

The test environment will be a setup that mimics the production environment, so React, NodeJS, Express, Typescript and a MongoDB Atlas database are needed. Each member that executes test needs a computer with React, NodeJS, Express, and Typescript installed- The database instance for testing will be available in the cloud. The database instance will be configured by a member of the Testing Team, and it will provide the connection string to all members of the Development Team. The operative system used is Windows 11. The testing environment is needed for the entire testing period. Specific tools requirements are:

- Jest.
- EsLint.
- JSCPD.

To report the readiness of environment requirements, the Testing Team member will inform via an email the readiness of the database instance for testing. The email will include the connection string and all credentials necessary to interact with the instance. Each member of the Development Team will verbally inform that they have the environment installed in their working environments.

Retesting

Modules with high priority defects will have to be retested once such defects have been corrected. Documentation and code reinspection is only necessary if the inspection team concludes so.

Regression Criteria

Regresstion testing will be performed if new changes are introduced in existing requirements or functions that have been tested before.

Suspension and Resumption Criteria

Suspension and resumption cases are specified in the risk register in this document.

Testing Activities and Estimates

The test plan consists of the following activities:

- The Testing Team Leader assembles the Testing Team and delivers the data requirements to the Development Team.
- The Testing Team members are introduced to the organization documentation procedures and undergo training if necessary.
- The Testing Team reviews the test plan, data requirements, and environment requirements.
- The Testing Team members prepare and install the test environment.
- The Development Team deliver the data requirements.
- The Testing Team Leader request an inspection of the software requirements document, the software architecture document, and the code of the software modules.
- An inspection is performed on the software requirements document. A report specifying the results of the inspection is delivered to the Testing Team Leader.
- An inspection is performed on the software architecture document. A report specifying the results of the inspection is delivered to the Testing Team Leader.
- The Testing Team members prepare the test model specification for the static testing of the code base, and the test model specification for the dynamic testing of the purchase module.
- An inspection is performed on the code base using the automation tools defined. A report specifying the results of the inspection is delivered to the Testing Team Leader. The report is included in the same document for the dynamic testing results report.
- The Testing Team members prepare the test cases and procedures for the dynamic testing of the purchase module.
- The Development Team executes the dynamic testing on the purchase module.
- The Testing Team members prepare a document specifying the results of the dynamic testing. This is an Excel document specifying such results.
- Based on the results of inspections and dynamic testing, retesting is performed.

Time estimates are specified in the schedule section of this document.

Staffing

Roles, Activities and Responsibilities

Detailed roles, activities and responsibilities will be specified in the corresponding documentation for each test type and level. Inspections teams, their responsibilities and activities will be specified in the inspection report documentation. Roles, activities, and responsibilities for dynamic testing will be specified in the test model specification.

Hiring Needs

Three experienced testers are required to assemble the Testing Team. The testers will be hired temporarily until all testing activities of the project have been completed, approved, and signed of. While the testing activities are performed, the testers are needed full-time. The testers should have the following skill set:

- Communication skills.
- Ability to follow documented procedures.
- Analytical mindset.
- Excellent time management and prioritization.
- Automation testing.
- Understanding of testing tools, such as Jest and EsLint.
- Understanding of programming languages and frameworks such as React, NodeJS, Express, Javascript and Typescript.
- Understanding of databases such as MongoDB.

Training Needs

Training in the organization documentation procedures will be provided for any tester that do not have existing experience in such procedures. Training in the project testing environment tools will be provided for any tester that does not consider having a proficient understanding and management of such tools.

Schedule

The testing activities schedule is defined in the next table:

Table 3 - Testing Activities Schedule

Activity	Start Date	Completion Date	Total Time
Testing Team selection.	March 14, 2024.	March 14, 2024.	1 day.
Introduction and training activities. Data requirements and environment	March 14, 2024.	March 22, 2024.	7 days.

requirements preparation.			
Software requirements document inspection.	April 2, 2024.	April 5, 2024.	4 days.
Test model specification documents preparation.	April 1, 2024	April 8, 2024.	6 days.
Test cases and procedures preparation.	April 8, 2024	April 12, 2024.	5 days.
Software architecture document inspection.	April 9, 2024.	April 12, 2024.	4 days.
Code inspections.	April 15, 2024.	April 24, 2024.	8 days.
Dynamic tests preparations.	April 15, 2024.	April 25, 2024.	9 days.
Dynamic tests execution.	April 26, 2024.	April 26, 2024.	1 day.
Dynamic testing results and automatic static testing results report.	April 29, 2024.	April 29, 2024.	1 day.

The total estimated time for testing activities is 28 days, starting on March 14, 2024, and completing the activities on April 29, 2024.