Revision Log

Date	Version	Change
Feb 23, 2026	0.1	Initial outline and content of the document

1. Introduction

1.1 Scope

The purpose of this document is to provide the information and the framework required to plan and perform test processes needed for the testing of the The Internet website made as a demo by Heroku App. This document has been based on the structure provided by ISO/IEC/IEEE 20119-3:2013(E), and based on iSTQB standards.

1.2 References

[TP] Test Plan

[TS] Test Specification

1.3 Abbreviations

NA Not applicable

A/D Archived or Deleted

TBD To be defined, that is it is not yet known what is to be written.

2. Plan Context

2.1. Project

The Internet is a demo website made by Heroku App, an open-source playground made for tests. It contains a variety of functions that allows users to use it as a testing ground, including but not limited to DOM elements, JQueries, JavaScript statements, file interaction, and API calls.

2.2 Requirements (Reverse-Engineering)

The requirements listed in this section for The Internet demo website were written through the use of reverse-engineering. They do not represent what was the intent of the project at the start, but actually what can be the requirements for the current application.

Requirement ID	Description	Test Case ID	Status
RE-01	The system shall allow access to the /secure area only upon entering the valid username and password.	F1-1, F4-1	Draft
RE-02	The system shall display an error message if the username is incorrect.	F1-2, F4-2	Draft
RE-03	The system shall display an error message if the password is incorrect.	F1-2	Draft
RE-04	Successful login should redirect the user to the "Secure Area" page.	F1-1	Draft
RE-05	The UI elements (menus, buttons) must be interactive and visible across Chrome, Firefox, and mobile viewports.	F1-1, F1-2, F2-1, F3-1, F4-1, F4-2, F5-1, F6-1, F7-1, F8-1, F9-1, F9-2	Draft
RE-06	The Dynamic elements must be rendered correctly	F2-1	Draft
RE-07	The API calls should return the expected result with appropriate message	F5-1	Draft
RE-08	All the links should direct to the desired pages without breaking	F7-1	Draft
RE-09	The uploaded file should be stored temporarily	F8-1	Draft

2.2 Test Items(s)

The test for this project includes the manual testing of:

- Basic Authorization functionality
- JQuery UI menu functionality
- Dynamic Content functionality
- Different Screen Size Compatibility
- API Calls
- Browser Compatibility with Chrome, Opera, and Firefox.

2.3 Test Scope

The test scope consists of the functionalities listed above. The test has no objective of being exhaustive to test all the available functionalities, because it has as objective to be a testing practice of some of the main functionalities, focusing on variety instead of exhaustivity due to time constraints.

Non-functional quality factors like performance, security, safety will not be tested in this test project due to time constraints. These functions could be added in a separate Test Plan in the future.

2.4 Assumptions and Constraints

Testing is constrained by available time and resources.

2.5.Stakeholders

The stakeholders are as following:

Project Owner: Heroku Demo Team
Test Lead: Leonardo dos Reis Fernandes
Users: Website Visitors

3. Testing Communication

As this project is conducted by a single tester, no team communication is required during testing. All test results, reports, and documentation will be shared via GitHub, which serves as the primary communication channel for this project.

4. Risk Register

   P = Probability

   I = Impact

   E = Exposure = Probability x Impact

   The scale for both probability and impact will be 1-6, where 6 is the highest.

   4.1. Product Risks

ID	Description	P	I	E	Mitigation Strategies
1	Web app does not open	1	6	6	Test app early Try multiple browsers Check browser console Note errors Report defect immediately
2	Login form doesn't work	3	5	15	Test login early Verify credentials Report defects immediately
3	Dynamic content fails to load	3	3	9	Test multiple times Check browser console Note errors
4	API endpoints return wrong data	3	5	15	Validate with Postman verify data integrity
5	UI Elements not responsive	3	3	9	Test on different screen sizes Log issues
6	Broken links or buttons	2	3	6	Exploratory testing Report issues

   4.2. Project Risks

ID	Description	P	I	E	Mitigation Activities
1	Lack of knowledge / experience in testing	4	6	24	Review ISTQB Follow guides Ask for feedback from peers or mentors
2	No testing tools in budget	3	4	12	Use free tools: Postman, browser dev tools, GitHub, Notion
3	Limited time	4	5	20	Make schedule Focus on 10–20 test cases Prioritize important features
4	Unclear requirements for Heroku demo	2	3	6	Decide scope yourself Document assumptions in Test Plan
5	Technical issues / downtime on Heroku app	2	3	6	Retry testing later Document issues as “environment risk”

5. Test Strategy

5.1. Test Sub-Processes

The test for The Internet includes the following test sub-processes:

- Component testing
- System Testing

5.2. Test Deliverables

The following documents will be produced during testing of The Internet – Heroku Demo App:

- Test Specification: Detailed instructions for executing each test, including expected results and preconditions.

- Test Logs: Documentation of executed tests and defects, including severity, priority, and reproduction steps.

- Test Execution Report: Summary of the testing process, results, metrics, and overall conclusions.

5.3. Test Design Techniques

Each of the test case shall use techniques that are available and relevant, including:

- Black-Box testing
- Decision table testing
- Classification Tree Method
- Boundary Value Method (BVA)
- Use Case testing
- State transition testing

5.4. Test Completion Criteria

The test must achieve 80% requirements coverage, and all test procedures must be executed without failures of severity 1 (High). Minor defects of severity 2 (Medium) or 3 (Low) shall be reported, but will not block completion.

5.5 Metrics to be collected

The following metrics are to be collected during the course of the system test:

- Number of executed test cases;
- Number of incidents per category;
- Number of re-executed test cases;
- Number of solved incidents per category
- Number of hours spent;

5.6. Test Data and Test Environment Requirements

Each of the test cases could require specific environment requirements that are specified in the test cases segment (put reference). This can include, but not limited to:

- Postman (API Testing)
- Internet Browser

5.7. Retesting and Regression Testing

No retesting or regression testing will be performed.

5.8. Suspension and Resumption Criteria

If test completion is impossible due to external causes the completion must be postponed until these have been cleared.

6. Testing Activities and Estimates

   The test work will be broken down into the following main activities:

   1. Definition of an overall structure for the test in the form of feature sets to be tested;
   2. Detailed specification of the test cases and test procedures;
   3. Establishment of the test environment;
   4. Execution cycle of test procedures;
   5. Test completion reporting.

7. Staffing

   7.1. Roles, Activities, and Responsibilities

   The test shall be planned, designed, and executed, and all the necessary reporting shall be written by Leonardo dos Reis Fernandes, acting as test lead, analyst, designer, environment expert, and executor.

8. Schedule

   The overall schedule for the test is shown below, re

Test Schedule	1	2	3	4	5	6	7	8	9	10	11	12
Analyze	■	■										
Design			■	■								
Test Environment				■								
Test Execution					■	■	■	■	■	■	■	
Completion Report												■