TEST PLAN

SHIVAJI DHUMAL
2023

Version History

Version	Revised By	Summary	Approval	Date

Table of Contents

INTRODUCTION

The Test Plan outlines the scope, approach, resources, and schedule of all testing activities. It identifies the items and features to be tested; types of testing. It contains a detailed and executable strategy for conducting. It defines the detailed testing objective. specific to a particular system, the testing approach, test environment, test conditions, and the test plan.

SCOPE

The scope of this test plan is to ensure websites meets all of its technical, functional and business requirements. The purpose of this document is to describe the overall test plan and strategy for evaluating the website. The approach described in this document provides the framework for all the testing related to websites.

This document will also be updated as required with the requirement updates. We also need to make sure that all the expected results are achieved.

OUT OF SCOPE

TEST OBJECTIVES

The general test objectives are to test the correctness of the generation of the interface data file, the content of the interface data file, and any error conditions. The quality objectives of testing the website are to ensure complete validation of the business and software requirements:

- Verify software requirements are complete and accurate.
- Perform detailed test planning.
- Identify testing standards and procedures that will be used while testing the website.
- Prepare and document test scenarios and test cases.
- Manage defect tracking process.
- Finalize the project for release.

REFERENCE DOCUMENTS

DETAILED TEST APPROACH

Detailed testing phases and methodologies are mentioned below. We will follow the protocols of each phase and achieve the highest results.

- Requirement Analysis
- Design Testing
- Functionality Testing
  - Verify each functionality of the system works as per requirement.
  - Testing the links
  - Testing the forms
  - Cookies Testing
  - Validation (HTML/CSS/PHP)
  - Database Testing
- Integration Test Specification
- API Testing
- Usability Testing
- Compatibility Testing
  - Browser Compatibility
  - OS compatibility
  - Mobile browsing
- Performance Testing
  - Load Testing
  - Stress Testing
- Security Testing
- Automation Testing
- Smoke Testing
- Beta Testing

REQUIREMENT ANALYSIS

Requirements analysis is critical to the success or failure of a systems or software project so we have to verify, validate and confirm each requirement. Requirements must be validated on the basis of User Experience, User Interface. How to test the requirements. Requirements are up to date. Make sure the major scenarios and requirements are mentioned in the document. If there is something missing, highlight the missing requirements and also suggest improvements if there are any.

DESIGN TESTING

Test all the designs and verify all the designs must be correct as per the requirements. And also make sure the designs for all the specified languages and dark themes.

FUNCTIONALITY TESTING

Test the functional requirements and determine every function of the software is acting in accordance with the pre-determined requirements and tasks. At website performed testing of all the links in web pages, checking the database connections, forms used in the web pages for submitting or getting information from user & Cookie testing. Functional testing is extended to the types given below.

o Testing all the Links

- Test the outgoing links from all the pages from a specific domain under test.
- Test all internal links.
- Test links jumping on the same pages.
- Test links used to send the email to admin or other users from web pages.
- Test to check if there are any orphan pages.
- Lastly in link checking, check for broken links in all above-mentioned links.

o Testing of the forms on the web pages

Forms are the essential and integral parts of websites. Forms are used to get information from users and to keep interaction with them.

The following should be checked on the forms:

- Check all the validations on each field.
- Check for the default values of fields.
- Wrong inputs to the fields in the forms.
- Options to create forms, if any, form delete, view or modify the forms.

- Check that no empty forms are created.
- There are different field validations like email-id's, user financial information, date, etc.
- All the above validations should be checked in a manual or automated way.

o Cookie Testing

Cookies are small files stored on a user machine that are basically used to maintain the sessions such as the 'login sessions. Test the website to verify.

- Test the application by enabling or disabling the cookies in your browser options.
- Test if the cookies are encrypted before writing to user machine.
- During the test for the session cookies (i.e., cookies expire after the sessions end) check for login sessions and user status after session end.
- Check effect on application security by deleting the cookies.

o Validation (HTML/CSS)

HTML/CSS validation is especially important for optimizing the website for search engines.

- The site has full and correct Doctype
- The site uses a character set.
- The site uses valid XHTML.
- The site uses valid CSS
- The site has no unnecessary ids or classes.
- The site uses well-structured code.
- The site has no broken links.
- The site has clearly defined links.

o Database Testing

- Check for data integrity and errors while you edit, delete, modify the forms or do any database related functionality.
- Check if all the database queries are executing correctly and data is retrieved correctly and updated correctly.
- Also, we need to validate the Database by executing the queries.

AUTOMATION TESTING

Automate all the implemented functionalities of the website. Creation of the automation test cases. Details of Automation testing will be discussed and covered inside the Test Plan of
automation testing.

Write the test cases in a test rail for specific features. Analysis of test cases which are possible to be automated. Plan the test cases and how to execute all the test cases.
Write the test cases w.r.t environment.
write the logic for how to validate the test.
Write the pass/fail criteria against each test case. Run and actually verify the test.
Integrate all the test cases of each feature/module.

SMOKE TESTING

Smoke Testing is a software testing process that determines whether the deployed
software build is stable or not. Inside the smoke testing QA Engineer will make sure all the critical functionalities are working fine. We will create a checklist for smoke testing.
Smoke testing will be performed at two stages. Once new features are added, the other is before finalizing the build for Production/live. Create a checklist for smoke testing.

BETA TESTING

Beta testing is basically a release for specific users to use a product in a production environment to uncover any bugs or issues before a general release. Beta testing is the final round of testing before releasing a product to a wide audience. The objective is to uncover as many bugs or usability issues as possible in this controlled setting. QA will also perform the beta Testing.

TEST STRATEGY

The overall strategy of this testing initiative is manual, black box testing. We are testing the data, interface part and implemented system in detail. The testing at the SAP end of the interface will be covered by the SAP functional testing. Follow the testing phases and techniques mentioned inside "Detailed Test Approach".

All type of testing is covered in this document. Some of the test specifications use test data which needs to be set-up in the test environment prior to executing the test cases.

For each level of testing, a separate test plan is prepared with the following set of deliverables:

- o Test Cases/Test Scenarios
- o Features to be tested.
- o Items to be tested.
- o Pass / Fail criteria.
- o Bugs cycle.
- o Automation.
- o Expected Results.
- o Actual Results.

TEST SCHEDULE

The test schedule is the timeline of acceptance testing activities and deliverable dates. Testing activities are mentioned below.

- o Requirement Analysis.
- o Design Testing.
- o Develop test scenarios.
- o Develop test cases.
- o Review scenarios/test cases for accuracy, completeness and sequence.
- o Integration testing.
- o API Testing.
- o Regression Testing.
- o Functionality Testing.
- o Database Testing.
- o Integration Test Specification.
- o Usability Testing.
- o Compatibility Testing.
- o Performance Testing.
- o Security Testing.
- o UAT Testing.
- o Automation Testing.
- o Smoke Testing.
- o Beta Testing

PROBLEM/BUG SEVERITY CLASSIFICATION

The identified severity for each problem implies a general reward for resolving it, and a general risk for not addressing it, in the current release.

Severity 1 - Crash or High impact problems that often prevent a user/host from correctly completing an experience/booking.

Severity 2 - Moderate to high frequency problems with the functionality/UI or UX impact

Severity 3 - Either moderate problems with low frequency or low problems with moderate Frequency; these are minor annoyance problems faced by a number of participants.

Severity 4 - Low impact problems faced by few participants; there is low risk of not resolving these problems. Reward for resolution is typically exhibited in increased user satisfaction.

TEST RESOURCES

Here is the list of resources with the roles those will work on websites.
1.
2.
3.

PASS/FAIL CRITERIA

Create the test cases and mention the expected results/pass criteria against each test case.

ENVIRONMENT

Start testing on a staging server once a certain level is achieved, then move to Production and give the final approval at Production. All the experiments should be performed at staging. Testing data must be private at Production.

TEST CASES AND TEST SCENARIOS

Write down the detailed test cases on the basis of requirement, technical document and test plan. For testcases use Google sheet and use the Jira for bug, suggestion reporting.

TOOLS AND DEFECT TRACKING

Jira will be used for defect reporting and issue bugs/defects management and traceability.

FINAL TEST REPORT

Test closure reports shall be generated for each testing phase as the testing phase gets completed.

EXIT CRITERIA

All the test cases and test scenarios must be passed. Every user must get the music recommendation as per their interests.