INFO 6068 Capstone

Test Plan

Team Pixel

**Team Members:**

**Rutvik Patel (Developer): 1220039**

**Sakshi Modi (Manager): 1179590**

**Parth Patel (Scribe): 1229403**

**Vijul Vyas (Tester): 1227571**

### Amendment History:

|  |  |  |
| --- | --- | --- |
| Version | Date | Amendment History |
| v.1.0 | May 31,2024 | Test Planning Draft |
| v.1.1 | May 31,2024 | Updated Milestone, Scope and Test Environment |
| v.1.2 | June 04,2024 | Final Version Test Plan |
|  |  |  |
|  |  |  |
|  |  |  |

### 

### Reviewers:

This document must be reviewed by the following:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Signature | Title / Responsibility | Date | Version |
| Sakshi Modi | S.M | Project Manager | June 01,2024 | v.1.0 |
| Parth Patel | P.Patel | Scribe | June 01,2024 | v.1.0 |
| Vijul Vyas | Vijul | Tester | June 01,2024 | v.1.0 |
| Rutvik Patel | R.P | Developer | June 01,2024 | v.1.0 |

### Approvals:

This document must be approved by the following:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Signature | Title / Responsibility | Date | Version |
| Shrijy Atodaria |  | Sponsor | June 01,2024 | v.1.0 |

### 

### Distribution:

### Distribution:

|  |  |  |
| --- | --- | --- |
| **Name** | **Title / Responsibility** | **Date** |
| Shradha Gupta | Product Owner | June 01, 2024 |
| Shrijy Atodaria | Project Sponsor | June 01, 2024 |
| Sakshi Modi | Project Manager | June 01, 2024 |
| Rutvik Patel | Developer Team | June 01, 2024 |
| Parth Patel | Scribe | June 01, 2024 |
| Vijul Vyas | QA Team | June 01, 2024 |

### Related Documents:

These documents will provide additional information.

|  |  |  |  |
| --- | --- | --- | --- |
| Ref no | Doc Reference Number | Title | Version |
| R\_01 | D\_01 | Test Plan Document | V.1.0 |
| R\_02 | D\_02 | Meeting Agenda | V.1.0 |
| R\_03 | D\_03 | Meeting Minutes | V.1.0 |
| R\_04 | D\_04 | IAD logs | V.1.0 |
| R\_05 | D\_05 | Weekly Status Report | V.1.0 |
| R\_06 | D\_05 | Test Strategy Document | V.1.0 |

### Glossary of Terms:

List any terms used in this document.

|  |  |  |
| --- | --- | --- |
| **Term** | **Acronym** | **Definition** |
| IAD | Issue, Action, Decision | A document which contains issues, actions and decisions of  any problem occurred. |

Table of Contents

[1. Introduction 5](#_Toc168853654)

[1.1 Objectives 5](#_Toc168853655)

[1.2 Team Members 6](#_Toc168853656)

[2 Scope 6](#_Toc168853657)

[3 Assumptions / Risks 7](#_Toc168853658)

[3.1 Assumptions 7](#_Toc168853659)

[3.2 Risks 8](#_Toc168853660)

[4 Test Approach 9](#_Toc168853661)

[4.1 Test Automation 9](#_Toc168853662)

[5 Test Environment 11](#_Toc168853663)

[6 Milestones / Deliverables 12](#_Toc168853664)

[6.1 Test Schedule 12](#_Toc168853665)

[6.2 Deliverables 14](#_Toc168853666)

# Introduction

* The test plan's purpose is to provide an overview of the extensive testing approach for the Travel app, which is intended to guarantee that the program offers an effortless and successful user experience. The goals—which include confirming functionality, usability, performance, security, and compatibility—are thoroughly outlined in this paper. All essential aspects, including user registration, booking procedures, itinerary management, payment systems, and alerts, are included in the testing scope. Integration with third-party services, such as payment gateways and booking engines, is also included. The testing team, which consists of engineers, automation engineers, test managers, UX specialists, security experts, and testers, will execute toward this strategy in agreement with a planned timeline that covers everything from case generation and test planning to final testing and sign-off. Test cases, scripts, executions outcome, defect logs, and other deliverables will be included.
* This test plan will be implemented with the contribution of Team Pixel, a professional testing group. They will oversee creating thorough test cases, carrying out thorough testing, and recording all results. They will also manage usability testing for assessing user experience, security testing to verify data protection protocols, performance testing under varied load situations, and automated testing for regression. To guarantee smooth operation with outside services, Team Pixel will also carry out integration testing, report and monitor errors, and carry out final verification to make sure all problems have been fixed. With this careful process, we can make sure the Travel App is up to standard and prepared for an error-free launch.

## Objectives

The test plan for the Travel App ensures it meets requirements and delivers a high-quality user experience. It verifies functionality, identifies and resolves bugs, and assesses user design. Security and performance reliability are tested, along with compatibility across devices and integration with external services. Compliance with data security and privacy regulations is ensured. The goal is to meet quality standards before release, providing a reliable travel management solution.

**Key Objectives:**

* **Requirement Verification:** Confirm all features work as intended.
* **Bug Resolution:** Identify and fix flaws quickly.
* **User Experience**: Assess and enhance design and usability.
* **Security Testing:** Protect data and prevent unauthorized access.
* **Performance Testing:** Ensure reliability under various scenarios.
* **Compatibility Testing:** Functionality across devices, OS, and browsers.
* **Integration Testing:** Seamless connectivity with external services.
* **Quality Assurance:** Meet quality standards before release.

|  |  |  |
| --- | --- | --- |
| Name | Role | Responsibilities |
| Sakshi Modi | Project Manager | Project Manager is tasked with coordinating with team members to develop a comprehensive test plan, ensuring adherence to the defined test approach, and monitoring progress across multiple testing stages. Holds the responsibility of overseeing the entire testing process. |
| Parth Patel | Scribe | The Scribe is responsible for documenting all testing activities. They keep track of test plans, results, and any issues encountered during testing. The Scribe ensures that all information is accurately recorded and shared with the team and stakeholders, contributing to the smooth progress of the testing process. |
| Rutvik Patel | Developer | The Developer is tasked with resolving any bugs or issues identified during testing. They work closely with the testing team to understand and address reported issues promptly. Additionally, the Developer ensures that the application's code is updated and optimized to improve performance and functionality based on feedback from testing. |
| Vijul Vyas | Tester | The Tester is responsible for thoroughly examining the application's functionality and usability. They meticulously execute test cases, identify potential issues or bugs, and report their findings to the development team. |

## Team Members

# 

# Scope

Considering that developers have previously carried out unit testing on every module interface, this is the thorough testing approach for the Travel App.

* **In Scope Items**
* **User Registration and Authentication Module:** Verification of user authentication, sign-up, login, and password recovery procedures.
* **Booking Module:** Testing the efficiency of making reservations for accommodation, transport, and other travel-related services.
* **Itinerary Management Module**: ensuring that users can effectively create, view, and manage their trip plans.
* **Payment Module:** Testing the integration of several payment gateways and the complete payment process.
* **User Profile Module:** Verifying features such as travel history, preferences, and personal information pertaining to user profile management.
* **Types of Testing to be Performed:**

* **System tests:** Extensive testing of the system to guarantee that every component works together as it is supposed to.
* **Acceptance Tests:** Verifying that the program satisfies end-user demands and validates against business criteria.
* **Security tests:** Making sure there are no security holes or unapproved access points in the program.
* **Performance tests:** Assessing the application's responsiveness and stability under several load scenarios.
* **Accessibility tests:** Verifying that the program complies with applicable standards and is usable by people with impairments (e.g., WCAG).
* **API Tests:** Ensuring each API endpoint's dependability, security, and functionality.
* **Automation of Tests:** Putting automated test scripts into practice to guarantee consistency in test execution and to enable effective regression testing.
* **Out of Scope Items:**
* **Cost:** Budget and financial aspects of the testing process.
* **Unit Testing:** low-level testing that is intended to be performed by developers on specific parts or features.

# Assumptions / Risks

## Assumptions

* This section lists assumptions that are made specific to this plan.
* The Travel App is only compatible with web browsers.
* Unit and integration testing is done by developers prior to formal testing.
* There is an environment set up specifically for testing that is available.
* Continuous developer assistance is expected during all testing phases.
* The Project Manager's prior approval is required before testing documents can be legally implemented.
* Test runs are performed at each testing stage to guarantee iterative validation.

## Risks

Identify risks and the appropriate action needed to mitigate their impact on the project. The impact (or severity) of the risk is based on how the project would be affected if the risk was triggered. The trigger is what milestone or event would cause the risk to become an issue to be dealt with.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | Risk | | Impact | Trigger | Mitigation Plan |
| 1 | | Hardware Failure | High | Unexpected hardware failure | Regular maintenance, monitoring, and backup of hardware components. Immediate resolution by IT support. |
| 2 | | Software Compatibility Issues | Medium | Incompatibility between browsers, OS, or third-party tools | Thorough compatibility testing, software version updates, and utilization of virtualized testing environments. |
| 3 | | Network Connectivity Problems | Medium | Unstable or unreliable network connections | Implement redundant network connections, use monitoring tools, and establish alternative communication channels. |
| 4 | | Insufficient Test Environment Resources | Medium | Limited resources such as memory or processing power | Monitor resource usage, optimize configurations, and allocate additional resources as necessary. |
| 5 | | Data Security Breaches | High | Inadequate security measures in the test environment | Implement robust security protocols, restrict access to sensitive data, and conduct regular security audits. |
| 6 | | Tool Integration Challenges | Medium | Difficulties in integrating testing tools with other systems | Ensure compatibility, provide training, and establish clear communication channels for tool integration. |
| 7 | | Lack of Test Data | High | Insufficient or inaccurate test data | Develop comprehensive test data sets, use data generation tools, and validate test data integrity. |
| 8 | | Inadequate Test Coverage | High | Incomplete coverage of critical website functionalities | Conduct thorough requirements analysis, prioritize testing efforts, and continuously refine test coverage. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 9 | Human Errors | Medium | Errors in test case design, execution, or data entry | Implement quality assurance processes, conduct peer reviews, and provide training to minimize human errors. |
| 10 | Scope Creep | High | Changes in project scope or requirements | Establish change management processes, document and prioritize changes, and communicate promptly to stakeholders. |
| 11 | Lack of Documentation | Medium | Inadequate documentation of test plans, procedures, or results | Develop documentation standards, provide training, and establish review processes for documentation. |

# Test Approach

In an Agile setting, the test approach for website testing revolves around adaptability, collaboration, and continuous improvement. Testers engage early in the development cycle, collaborating closely with developers, product owners, and stakeholders to understand requirements and provide timely feedback. Testing is iterative, focusing on frequent increments or user stories, with an emphasis on risk-based testing and automation for efficiency. Test activities include requirement analysis, test planning, design, execution, and defect reporting, with regression testing after each iteration. Stakeholder involvement in User Acceptance Testing ensures alignment with expectations. Test tools encompass automation, defect tracking, collaboration, performance, and security testing tools. Continuous improvement is driven by retrospectives, feedback loops, and ongoing skill development. This approach ensures that the website is thoroughly tested, meeting quality standards, and delivering value to end-users in a dynamic and rapidly evolving Agile environment.

## Test Automation

**Test Approach for Travel Application**

The Travel application employs an Agile development model, ensuring regular testing intervals throughout the project cycle. This approach provides continuous feedback to the customer, helping to track the project schedule and constraints effectively. Incremental testing within the Agile model allows us to test every aspect of the application before each release, ensuring defects are identified, fixed, and retested promptly.

**Key Aspects of Test Automation**

1. **Selection of Automation Tools:**
   * Choose appropriate tools that support the technology stack of the website. Popular tools include Selenium, Cypress, and TestComplete.
   * Ensure the tools integrate seamlessly with CI/CD pipelines (e.g., Jenkins, GitLab CI).
2. **Test Planning and Design:**
   * Identify critical test scenarios that benefit most from automation, such as regression tests, smoke tests, and frequently executed test cases.
   * Design modular, reusable, and maintainable test scripts to ensure scalability and ease of updates.
3. **Integration with CI/CD Pipelines:**
   * Automate test execution to trigger on code commits and builds, providing immediate feedback to developers.
   * Utilize continuous integration tools to run automated tests, ensuring quick identification and resolution of defects.
4. **Test Data Management:**
   * Automate the creation, modification, and cleanup of test data to ensure consistency and reliability in test environments.
   * Use data-driven testing to run tests with multiple data sets, enhancing coverage.
5. **Cross-Browser and Cross-Device Testing:**

* Implement automation to validate the website’s functionality across different browsers and devices (e.g., desktops, tablets, smartphones).
* Use tools like BrowserStack or Sauce Labs to facilitate comprehensive cross-browser testing.

1. **Reporting and Monitoring:**
   * Generate detailed test reports that provide insights into test execution results, highlighting pass/fail status, errors, and performance metrics.
   * Implement dashboards for real-time monitoring of automated test results, enabling proactive issue resolution.
2. **Maintenance and Updates:**
   * Regularly update test scripts to align with changes in website features and functionalities.
   * Refactor test code to improve efficiency and accommodate new testing requirements.
3. **Collaboration and Communication:**
   * Foster collaboration between development, testing, and operations teams to ensure seamless integration of test automation within the Agile workflow.
   * Conduct regular meetings and reviews to discuss automation progress, challenges, and improvements.

# Test Environment

* **Physical Devices:**
* Laptop/Desktop
* Additional Monitor (optional)
* Ethernet cables
* RAM: 4GB Processor or higher
* Operating System: x64 bit - Windows 10
* New server
* **Software:**
  + Browsers: Mozilla Firefox, Google Chrome, Edge, Internet Explorer
  + Collaboration: Slack
  + Office Suite: Microsoft Office
  + Virtual Meeting Tools: Google Meet, Zoom
  + Databases: MySQL 5.7.26
  + Server: Apache Tomcat 9.0.19
* **Test Tools:**
  + TestNG
  + Selenium WebDriver
  + Integrated Development Environment: Eclipse IDE
  + Programming Language: Java - JDK
  + Database Management: MySQL
  + API Testing: Postman
  + Performance Testing: Apache JMeter
  + Version Control: GitHub, Git
  + Documentation: Microsoft Word
  + Project Tracking: JIRA
  + Spreadsheet: Microsoft Excel
  + Functional Testing: Selenium WebDriver

# Milestones / Deliverables

## Test Schedule

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task Name** | **Start** | **Finish** | **Effort** | **Comments** |
| Capstone Project | Mon 13/05/24 | Sun 11/08/24 | 73 days |  |
| Test Strategy | Mon 13/05/24 | Sun 26/05/24 | 12 days |  |
| Define objectives and scope | Mon 13/05/24 | Tue 14/05/24 | 2 days |  |
| Identify resources and responsibilities | Wed 15/05/24 | Thu 16/05/24 | 2 days |  |
| Determine test levels and cycles | Fri 17/05/24 | Mon 20/05/24 | 2 days |  |
| Select testing tools and techniques | Tue 21/05/24 | Wed 22/05/24 | 2 days |  |
| Establish risk management procedures | Thu 23/05/24 | Fri 24/05/24 | 2 days |  |
| Review and approve the strategy document | Sat 25/05/24 | Sun 26/05/24 | 2 days |  |
| Test Plan & MS Project | Mon 27/05/24 | Sun 09/06/24 | 11 days |  |
| Create a detailed test plan | Mon 27/05/24 | Tue 28/05/24 | 2 days |  |
| Develop a timeline with milestones in MS Project | Wed 29/05/24 | Wed 29/05/24 | 1 day |  |
| Assign tasks to team members | Thu 30/05/24 | Fri 31/05/24 | 2 days |  |
| Set up resource allocation in MS Project | Mon 03/06/24 | Tue 04/06/24 | 2 days |  |
| Define entry and exit criteria | Wed 05/06/24 | Thu 06/06/24 | 2 days |  |
| Review and finalize the test plan | Fri 07/06/24 | Sun 09/06/24 | 2 days |  |
| Test Scenarios, Use Cases | Mon 10/06/24 | Sun 30/06/24 | 16 days |  |
| Identify test scenarios | Mon 10/06/24 | Thu 13/06/24 | 4 days |  |
| Develop detailed use cases | Fri 14/06/24 | Tue 18/06/24 | 3 days |  |
| Map scenarios to requirements | Wed 19/06/24 | Fri 21/06/24 | 3 days |  |
| Review use cases with stakeholders | Mon 24/06/24 | Wed 26/06/24 | 3 days |  |
| Update scenarios based on feedback | Thu 27/06/24 | Sun 30/06/24 | 3 days |  |
| Test Cases/Scripts | Mon 01/07/24 | Sun 14/07/24 | 11 days |  |
| Design test cases/scripts for each scenario | Mon 01/07/24 | Tue 02/07/24 | 2 days |  |
| Define test data requirements | Wed 03/07/24 | Thu 04/07/24 | 2 days |  |
| Review test cases with the team | Fri 05/07/24 | Tue 09/07/24 | 3 days |  |
| Update test cases based on review | Wed 10/07/24 | Thu 11/07/24 | 2 days |  |
| Organize test cases in a test management tool | Fri 12/07/24 | Sun 14/07/24 | 2 days |  |
| Test Execution | Mon 15/07/24 | Sun 21/07/24 | 6 days | Will be completed Parallelly |
| Prepare the test environment | Mon 15/07/24 | Mon 15/07/24 | 1 day |  |
| Execute test cases/scripts | Fri 26/07/24 | Tue 16/07/24 | 1 day |  |
| Log defects and issues | Mon 29/07/24 | Thu 18/07/24 | 1 day |  |
| Retest resolved defects | Fri 02/08/24 | Fri 19/07/24 | 1 day |  |
| Track test execution progress | Sun 04/08/24 | Sun 21/07/24 | 1 day |  |
| Reporting | Mon 15/07/24 | Sun 21/07/24 | 6 days | Will be completed Parallelly |
| Generate daily/weekly status reports | Mon 15/07/24 | Mon 15/07/24 | 1 day |  |
| Create defect summary reports | Tue 16/07/24 | Wed 17/07/24 | 2 days |  |
| Provide test coverage reports | Thu 18/07/24 | Thu 18/07/24 | 1 day |  |
| Share progress with stakeholders | Fri 19/07/24 | Fri 19/07/24 | 1 day |  |
| Conduct regular test status meetings | Sun 21/07/24 | Sun 21/07/24 | 1 day |  |
| Execution Report | Mon 22/07/24 | Sun 04/08/24 | 11 days |  |
| Compile test execution results | Mon 22/07/24 | Tue 23/07/24 | 2 days |  |
| Analyze test coverage and defect metrics | Wed 24/07/24 | Thu 25/07/24 | 2 days |  |
| Document lessons learned | Fri 26/07/24 | Mon 29/07/24 | 2 days |  |
| Prepare a summary report | Tue 30/07/24 | Wed 31/07/24 | 2 days |  |
| Review the report with stakeholders | Thu 01/08/24 | Sun 04/08/24 | 3 days |  |
| Final Presentation & Project Closure | Mon 05/08/24 | Sun 11/08/24 | 6 days |  |
| Prepare the final test summary presentation | Mon 05/08/24 | Mon 05/08/24 | 1 day |  |
| Present findings to the stakeholders | Tue 06/08/24 | Tue 06/08/24 | 1 day |  |
| Obtain sign-off from stakeholders | Tue 06/08/24 | Wed 07/08/24 | 1 day |  |
| Archive test artifacts and documentation | Thu 08/08/24 | Thu 08/08/24 | 1 day |  |
| Conduct a project retrospective | Fri 09/08/24 | Fri 09/08/24 | 1 day |  |
| Close the project officially | Sun 11/08/24 | Sun 11/08/24 | 1 day |  |

## Deliverables

|  |  |  |
| --- | --- | --- |
| **Deliverable** | **For** | **Date / Milestone** |
| Test Strategy | Stakeholders, testers, developers, project manager | 26thMay 2024/  Milestone 1 |
| Test Plan | Stakeholders, testers, developers, project manager | 9th June 2024/  Milestone 2 |
| Test Scenarios | Testers | 30thJune2024/  Milestone 3 |
| Test Cases | Testers, project manager | 14th July 2024/  Milestone 4 |
| Test Reports | Project managers, stakeholders, developers | 21st July 2024/  Milestone 4 Activity |
| Test design specifications | Stakeholders, testers, developers, project manager | 14th July 2024/  Milestone 4 |
| Defect Reports | Developers | 21st July 2024/ Milestone 4 Activity |
| Test Results | Project managers, developers | 4th August 2024/  Milestone 5 |