***Go Bear Test plan***

**1 Introduction**

The Test Plan is designed to prescribe the scope, approach, resources, and schedule of all testing activities of the project Go Bear Travel Insurance page.

The plan identifies the items to be tested, the features to be tested, the types of testing to be performed, the resources and schedule required to complete testing, and the risks associated with the plan.

**1.1 Scope**

**1.1.1 In Scope**

After analysis, these are all the features of Travel Insurance page which are needed to be test:

|  |  |  |
| --- | --- | --- |
| Module Name |  | Description |
| Navigation |  | User can navigate to the other pages using the link provided by the pages. (No dead links) |
| Comparison form functions |  | The form submission and reset work well. The form data is up-to-date. |
| Login function |  | User can login to their account using the login functions |
| Review function |  | Need more information from the client |
| Website Security and Performance |  | Check the security and performance of the web |
| User Interfaces (optional) |  | Check the page layout, fonts. |

**1.1.2 Out of Scope**

These feature are not tested in this regression test

* Database logical

**1.2 Quality Objective**

The test objectives are to **verify** the Functionality of Travel Insurance page, the project should focus on testing the **comparison function** to **guarantee** all these operation can work **normally** in real business environment.

**1.3 Roles and Responsibilities**

The project should use **outsource** members as the tester to save the project cost.

|  |  |  |
| --- | --- | --- |
| **No.** | **Member** | **Tasks** |
| **1.** | Test Manager | Manage the whole project Define project directions Acquire appropriate resources |
| **2.** | Tester | Identifying and describing appropriate test techniques/tools/automation architecture Verify and assess the Test Approach Execute the tests, Log results, Report the defects. Implement the test cases, test program, test suite etc. |

**2 Test Methodology**

**2.1 Overview**

**2.2 Test Levels**

In this project, there're 3 types of testing should be conducted.

* **Integration** Testing.
* **API testing:**Test all the APIs create for the software under tested

**2.3 Suspension Criteria and Resumption Requirements**

If the team members report that there are **40%** of test cases **failed**, suspend testing until the development team fixes all the failed cases.

**2.4 Test Completeness**

* Specifies the criteria that denote a **successful** completion of a test phase
* **Run** rate is mandatory to be **100%**unless a clear reason is given.
* **Pass** rate is **80%,**achieving the pass rate is **mandatory**

**2.5 Project task and estimation and schedule**

|  |  |  |
| --- | --- | --- |
| **Task** | **Members** | **Estimate effort** |
| **Create the test specification** | Test Manager | 24 man-hour |
| **Perform Test Execution** | Tester | 24 man-hour |
| **Test Report** | Tester | 24 man-hour |
| **Test Delivery** | Test Manager, Tester | 10 man-hour |
| **Total** |  | **82 man-hour** |

**3 Test Deliverables**

Test deliverables are provided as below

**Before testing phase**

* Test plans document.
* Test cases documents
* Test Design specifications.

**During the testing**

- Test Tools.

- Test Data

- Test Trace-ability Matrix - Error logs and execution logs.

**After the testing cycles is over**

* Test Results/reports
* Defect Report
* Installation/ Test procedures guidelines
* Release notes

**4. Testing Tools**

|  |  |  |
| --- | --- | --- |
| **No.** | **Resources** | **Descriptions** |
| **1.** | Xray & Jira | Test management tools and software project management tools for agile |
| **2.** | Intellij & Selenium | Test tool which can auto generate the test result to the predefined form and automated test script |
| **3.** | Apache JMeter and Google Lighthouse | Performance testing tools. |
| **4.** | Computer | At least 2 computer run Windows 10, Ram 8GB |

**5. Automation test script architecture:**

Behavior Driven Development involves the definition of entities, events, and outputs that the users interact with.

Test script structure:

* **Pages: Page we need to test.**
* **Stepdef: Define the step for the feature we define**
* **Ultilities: Function to support the test**
* **Resource: BDD features file.**