**AUTOMATION TEST PLAN**

1. **INTRODUCTION**

This document is to outline the test coverage plan of all the functionalities to be automated in **Computers database App**. It includes the objectives, test responsibilities, entry and exit criteria, scope, schedule major milestones, entry and exit criteria and approach. This document has clearly identified what the test deliverable will be, and what is deemed in and out of scope.

1. **QUALITY OBJECTIVES**

* **Primary Objectives**
* A primary objective of testing is to: assure that the system meets the full requirements, including quality requirements (functional and non-functional requirements) and fit metrics for each quality requirement and satisfies the use case scenarios and maintain the quality of the product. At the end of the project development cycle, the user should find that the project has met or exceeded all their expectations as detailed in the requirements. Any changes, additions, or deletions to the requirements document, Functional Specification, or Design Specification will be documented and tested at the highest level of quality allowed within the remaining time of the project and within the ability of the test team.

* **Secondary Objectives**
* The secondary objectives of testing will be to: identify and expose all issues and associated risks, communicate all known issues to the project team, and ensure that all issues are addressed in an appropriate matter before release. As an objective, this requires careful and methodical testing of the application to first ensure all areas of the system are scrutinized and, consequently, all issues (bugs) found are dealt with appropriately.

1. **TEST APPROACH**

The approach, that used, is Analytical therefore, in accordance to requirements-based strategy, where an analysis of the requirements specification forms the basis for planning, estimating and designing tests. Test cases will be created during exploratory testing. All test types are determined in Test Strategy. Team also must use experience-based.

Computers database High-level Document

Testing and error guessing utilize testers' skills and intuition, along with their experience with similar applications or technologies. The project is using an agile approach, with weekly iterations. At the end of each sprint the requirements identified for that iteration will be delivered to the team and will be tested.

1. **Test Automation**

Automated unit tests are part of the development process, and UI end-to-end testing will be performanced.

**5. ROLE AND RESPONSIBILITIES**

|  |  |  |
| --- | --- | --- |
| **Role** | **Staff Member** | **Responsibilities** |
| **Malaga office´s Lead Link** | Alberto Quiles | 1. Acts as a primary contact for development and QA team. 2. Responsible for Project schedule and the overall success of the project |
| **Software Tester** | Marcus Cruz | 1. Participation in the project plan creation/update process.  2.Planning and organization of the test process for the release.  3.Coordinate with QA analysts/engineers on any issues/problems encountered during testing.  4.Report progress on work assignments to the PM  5. Understand requirements  6. Writing and executing Test cases  7. Preparing RTM  8. Reviewing Test cases  9. Defect reporting and tracking  10. Retesting and regression testing  11. Bug Review meeting  12. Preparation of Test Data  13. Responsible for the automation process |

|  |  |  |
| --- | --- | --- |
| **QA Engineer** | Suchandra Se | 1. Participation in the project plan creation/update process.  2.Planning and organization of the test process for the release.  3.Coordinate with QA analysts/engineers on any issues/problems encountered during testing.  4.Report progress on work assignments to the PM  5. Understand requirements  6. Writing and executing Test cases  7. Preparing RTM  8. Reviewing Test cases  9. Defect reporting and tracking  10. Retesting and regression testing  11. Bug Review meeting  12. Preparation of Test Data |

**Entry Criteria**

* All test hardware platforms must have been successfully installed, configured, and functioning properly.
* All the necessary documentation, design, and requirements information should be available that will allow testers to operate the system and judge the correct behavior.
* All the standard software tools including the testing tools must have been successfully installed and functioning properly.
* Proper test data is available.
* The test environment such as, lab, hardware, software, and system administration support should be ready.
* Tester resources have completely understood the requirements
* Tester resources have sound knowledge of functionality
* Reviewed test scenarios and test cases

**Exit Criteria**

* A certain level of requirements coverage has been achieved.
* No high priority or severe bugs are left outstanding.
* All high-risk areas have been fully tested, with only minor residual risks left outstanding.
* The schedule has been achieved.

**TEST STRATEGY**

**QA role in test process**

- Understanding Requirements:

• Requirement specifications will be sent by the client.

• Understanding of requirements will be done by QA

**- Preparing Test Cases:**

Software Tester will be preparing test cases on the exploratory. This will cover all scenario for

requirement.

- **Preparing Test Matrix:**

Software Tester will prepare a test matrix which maps test cases to respective requirements.

This will ensure the cover.

**- Reviewing test cases and matrix:**

. Peer review will be conducted for test cases and test matrix by Tester

. Any comments or suggestions on test cases and test coverage will be provided by reviewer

respective author of the Test Case and Test Matrix.

. Suggestions or improvements will be re-worked by author and will be send for approval

. Re-worked improvements will be reviewed and approved by the reviewer.

. Creating Test Data: Test data will be created by respective Tester on client's

developments/test site based on scenarios and Test cases.

**- Executing Test Cases:**

Test cases will be executed by respective Tester on client's development/test

site based on designed scenarios, test cases and Test data.

Test result (Actual Result, Pass/Fail) will updated in test case document

Defect Logging and Reporting:

Tester will be logging the defect/bugs in Word document, found during

execution of test cases. After this, Tester will inform respective developer

about the defect/bugs.

- **Retesting and Regression Testing**:

Retesting for fixed bugs will be done by respective Tester once it is resolved by

respective developer and bug/defect status will be updated accordingly.

In certain cases, regression testing will be done if required.

**- Deployment/Delivery:**

Once all bugs/defect reported after complete testing is fixed and no other

bugs are found, the report will be deployed to the client's test site by **PM**.

Once round of testing will be done by Tester on client’s test site if required

Report will be delivered along with sample output by email to respective lead

and Report group.

Tester will be submitting the filled hard copy of the delivery slip to respective developer.

Once lead gets the hard copy of delivery slip filled by Tester and developer,

he will send the report delivery email to the client.

**Testing types**

**Black-box**

It is some time called behavioral testing or Partition testing. This kind of testing focuses on the

functional. requirements of the software. It enables one to derive sets of input conditions that

Will fully exercise all functional requirements for a program.

**End-to-End Testing**

Testing will include testing the UI part of the report. It covers users.

Report format, look and feel, error messages, spelling mistakes, **GUI** guideline violations.

**Functional Testing**:

Functional testing is carried out in order to find out unexpected behavior of the report.

The characteristic of functional testing is to provide correctness, reliability, test-ability and

accuracy of the report output/data

**System Testing:**

System testing of software is testing conducted on a complete, integrated

system to evaluate the system's compliance with its specified requirements.

**Performance Testing:**

- Check the optimal time the page is loaded

- Check the operation of the system under load

**User acceptance testing:**

The purpose behind user acceptance testing is to confirm that system is

developed according to the specified user requirements and is ready for

operational use. Acceptance testing is carried out at two levels - Alpha and

Beta Testing. User acceptance testing (UAT) will be done at the Client. Alpha

testing: The alpha test is conducted at the developer's site by the client.

* **6 Bug Severity and Priority Definition**

Bug Severity and Priority fields are both very important for categorizing bugs

and prioritizing when the bugs will be fixed. Testing will assign a severity level to all bugs.

The Test Lead will be responsible to see that a correct severity level is assigned to each bug.

The Tester, Development Lead and Project Manager will participate in bug

review meetings to assign the priority of all currently active bugs.

* **7 Test Environment**

Linux Ubuntu: Chrome (latest)

* **8 Tasks**

|  |
| --- |
| 1. **Initial state** |
| 1. **Filter by Name** |
| 1. **Add a new computer** |
| **Edit a computer** |
| 1. **Delete a computer** |
| 1. **Navigation through different pages** |
| 1. **Add a computer with wrong name and wrong dates** |
| 1. **Add a new computer cancelation** |
|  |
|  |

* **9 SCOPE**

**General**

**Initial state**

A user to be able to use this system, he/she has to be able to access

the url of the application [**http://computer-database.herokuapp.com/computers**](http://computer-database.herokuapp.com/computers)

He/she is presented with a list of maximum ten computer configurations, from all existing. The available functionalities from the initial state of the system are as follow:

* The user is able to view and navigate to different pages with computer lists.
* The user is able to filter the computer list by name.
* The user is able to add a new computer.
* The user is able to select a computer by clicking on its name.
* The user is able to see the numbers of computer

**Filter by name**

Providing a search function that searches your computers is a design strategy that users find a way to find content. Users can locate content by searching for a specific computer name, without to understand or navigate through the structure of the Web site. This can be a quicker or easier way to find content, particularly computer name.

The search functionality is added by either including a simple form on the Web page, usually a text field for the search term and a button to trigger the search or by adding a link to a page that includes a search form. The search form itself must be accessible, of course.

**Add a new computer**

The objective of this functionality is to provide a mechanism that allows users to explicitly request to add a new computer name. Since the intended use of a new computer name button is to generate an HTTP request that submits data entered in a form.

# **Editing a computer**

# This function allows a user to edit a computer name and this functionality also allows users to change the Introduced date, Discontinued date text field and select a Company.

**Deleting a computer**

The objective of this functionality is to provide a mechanism that allows to explicitly delete a computer that has been defined. Be careful not to delete a computer that users are depending on for criteria settings.

**Navigation through different pages**

The purpose of this functionality is to allow users to navigate the system through the pages in the pagination function. This function allows the user to view the computers name inside the table.

**9. Tactics**

The strategy for accomplishing the automation of the planned test cases and scenarios is to be accomplished by the automation **Selenium web driver** and **JavaScript** framework following approach. The tests will be run on a chrome browser.

**10. Feature to be automated**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Scenario** | **Status** | **Estimation time** |
| **1** | Initial state | **DONE** |  |
| **2** | Filter by Name | **DONE** |  |
| **3** | Add a new computer | **DONE** |  |
| **4** | Edit a computer | **DONE** |  |
| **5** | Delete a computer | **DONE** |  |
| **6** | Navigation through different pages |  |  |