***qwallity app***

**Test Plan**

**Release *1***

***15.10.2022 - 25.10.2022***

**VERSION HISTORY**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID & Version #** | **Prepared**  **By** | **Revision**  **Date** | **Approved**  **By** | **Approval**  **Date** | **Reason** |
| unique id | *Tatev Hakobyan* | *15/10/2022* | *Liana Sedrakyan* | *<date>* | release number |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**TABLE OF CONTENTS**

**1** **Introduction 4**

1.1 Purpose of The Test Plan Document 4

**2** **Test ITEM 4**

2.1 Project description 4

2.2 Items to be Tested / Not Tested 4

2.3 Items to be excluded 4

2.4 Test Approach(s) 4

2.5 Test Deliverables 4

2.6 Staffing / Training Needs 5

**3** **Risk and mitigation 5**

3.1 Test Risks / Issues 5

**4** **Test Environment and infrastructure 5**

4 Required Infrastructure 5

**5** **Roles and responsibilities 5**

5.1 Roles and assigned responsibilities 5

5.2 Test Team Leader………………………………………………………………...7

5.3 Software Tester…………………………………………………………………..8

**6** **Test Schedule 8**

6.1 Milestones and schedule 9

* **Introduction**
* **Purpose of The Test Plan Document**

The purpose of this document is to communicate the testing approach that the QA team will use for the Qwallity, 2.0v release. This document is targeted to the following reader groups:

**The QA Team**- This document will communicate internally the process used and the scope of the testing.

**The Development/Management Teams**- This document will provide a clear understanding of the testing approach to all external teams.

* **Test ITEM**
* **Project description**

*Project enables students to work on real application and to find bugs. Project contains some exercises, courses. Users can login and add, delete or modify courses.*

* **Items to be Tested / Not to be Tested**

|  |  |  |  |
| --- | --- | --- | --- |
| **Item to Test** | **Test Description** | **Test Date** | **Estimation** |
| Login screen | Login functionality, correctness of fields, buttons | 20.10.2022 | 6pt |
| Registration screen | Registration functionality, correctness of fields, buttons | 23.10.2022 | 9pt |
| Courses page | Add/Delete/Edit courses functionality | 28.10.2022 | 10pt |
| Forget/Change password screen | Sending code, Changing password functionality | 04.11.2022 | 5pt |
| Exercises page | Calculator/BlackBox/WhiteBox functionality | 08.11.2022 | 20pt |

* **Items to Not be tested**

|  |  |
| --- | --- |
| **Item Not to Test** | **Comment** |
| Source code quality | Code standards are used or not (SOLID principles) |
| Application security | The application has vulnerability or not |
| Database | The correctness of schema, integrity etc. |

* **Test Approach(s)**

This section describes the methodology used by the QA team. Some examples of subsections are as follows:

**Functional testing** – Mainly involves black box testing, checks User Interface, APIs, Database, Security, Client/Server communication and other functionality of the Application Under Test. (Either manually or automation)

**Security testing** – Uncovers vulnerabilities, threats, risks in a software application and prevents malicious attacks from intruders. The purpose of Security Tests is to identify all possible loopholes and weaknesses of the software system.

**Performance testing –** Testing process is used for testing the speed, response time, stability, reliability, scalability, resource usage of a software app under a particular workload.

**Usability testing –** Mainly focuses on user’s ease of using application, flexibility of application to handle controls and ability of application to meet its objectives.

**Confirmation testing** - Type of software testing technique used by testers to check if the previously posted bugs are rectified or not in the system or its components.

**Automated Testing Approach** – What types of automated tests will be done on this product, what is the automated testing tool to be used, where will the automated tests be documented, etc.

**Testing and Traceability** – How will you ensure that you have implemented a systematic testing approach, what testing documents are traceable back to development documents, etc.

**Scope of GUI Testing** – What parts of the GUI will be tested, will it be automated, etc.

**Integration Testing** - Is there an integration component to the product in question, what are some of the integration challenges that need to be overcome.

**Regular Bug Triages** – Who is responsible for ensure that bugs are being prioritized on a regular basis, which QA, DEV, BA and Document Writer representatives are taking part in the regular bug reviews.

**Testing Execution and Bug Tracking** – How will the testing team keep track of their testing progress?

**Bug Severity and Priority Setting** – where are these defined clearly in order to minimize ambiguity and ensure everyone has a common understanding.

**Description of the types of testing done and the testing period for this release** - This shows what a standard testing process looks like.

* **Test Deliverables**

This QA testing schedule is largely based on the development and technical publications schedules. All dates are subject to change if the development or documentation milestones are moved. Here are the key dates and testing periods.

|  |  |
| --- | --- |
| **Milestone/Project** | **Completion/Execution Dates** |
| Requirements Review/Estimation | <mm/dd/yy> |
| *Test Case preparation* | 17.10.2022 |
| *Test Case review* | 20.10.2022 |
| Manual Testing | <mm/dd/yy> |
| Automation script preparation | <mm/dd/yy> |
| Automation code review | <mm/dd/yy> |
| Regression testing(manual+automation) | <mm/dd/yy> |

* **Staffing / Training Needs**

Requirement needed for the testing to be performed –

* At least basic knowledge of manual testing
* Knowledge of the types of testing techniques
* Analytical skills
* Negotiation skills
* Responsibility
* Training of automation testing
* **Risk and mitigation**
* **Test Risks / Issues**

Include in this section any areas that may impede the progress of testing. Basically this section is an outstanding items list for QA.

* **Test Environment and infrastructure**
* **Required Infrastructure**
* System and applications
* Test data
* Database server
* Front-end running environment
* browser
* network
* documentation required like reference documents/configuration guides/installation guides/ user manuals
* **Roles and responsibilities**
* **Roles and assigned responsibilities**

|  |  |
| --- | --- |
| **Role** | **Responsibility** |
| Junior Tester | Perform tests, communicate with clients, report bugs, work with team members. |
| Senior Tester | Leading the QA team in designing QA test procedures, implementing testing procedures and overseeing the QA procedures, reviewing Quality Assurance reports. Conducting analysis checks on product specifications. Ensuring the successful deployment of products into the market. |
| Project Manager | Plan and develop the project idea, monitor project progress and set deadlines, solve issues that arise, manage the money, ensure stakeholder satisfaction, evaluate project performance. |

* **Test Team Leader/Manager**

The QA Team Leader/Manager is responsible for the following:

**Team Management - Planning Tasks**

* Define detailed Test schedule for team.
* Provide initial test planning for the QA team.
* Define QA Team roles and responsibilities.
* Estimate effort for the various deliverables.
* Identify training requirements.
* Identify support requirements.
* Interview candidates to fulfill the various Software Tester roles.

**Team Management - Daily Tasks**

* Define QA tasks to be performed.
* Resolve management issues involving QA and the development team.
* Track ongoing QA preparation and execution tasks in a schedule tool.
* Manage the QA Team (motivation, assessment, and orientation of new members).

**Team Management - Weekly Tasks**

* Assign tasks to various team members.
* Attend applicable management meetings for the purpose of providing QA's approval of all change requests (when applicable).
* Chairs the team status meetings.

**Team Management – Ad hoc Tasks**

* Manage testing of software fixes during the Beta, Final and Regression phases of Testing.
* Identify potential testing roadblocks.
* Write performance reviews of testers.

**Team Management - Deliverables**

* Ensure quality, timeliness of the various testing deliverables as identified in this Strategy document.
* Provide comments as the internal testing reviewers for the development deliverables (Functional Specs, design docs, etc.).
* **Software Tester**

The software tester reports to the QA Team Leader/Manager and is responsible for writing and executing manual and automated tests. The Software Tester's responsibilities include:

**Test Plan/Matrices and Scripts Preparation**

* Research relevant documentation to become knowledgeable enough to understand how the application was designed for the purpose of writing Test Plans/Matrices and Scripts.
* Write test plans that can be easily reproduced.
* Write test scripts that are easy to maintain.
* Ensure test plans and scripts are Traceable to applicable requirements and functional design documents (Functional Specs, help text, Design Documents, etc.).
* Write test cases (required set-up, procedures and information).
* Attend testing overviews (if available).

**Independent Verification of Test Specs/Matrices and Scripts**

* Provide Test Plans/Matrices and Scripts for review by peers, development and marketing representatives.
* Incorporate review comments into Test Plans/Matrices and Scripts.
* Conduct peer reviews Test Plans/Matrices and Scripts.

**Test Execution**

* Execute the Test Plans and Matrices.
* Run the automated tests.
* Report problems by raising bugs in Siebel.
* Follow up on bugs previously submitted in Siebel.

**Analyze Results**

* Report on successful test spec/matrix completion.
* Report on successful automated test completion.
* Verify successful resolution of bug fixes by verifying the contents of the bug reports and rerunning the test where applicable.
* Identify issues that should be documented in the Readme/Release Notes.
* **Test Schedule**
* **Milestones and schedule\**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Milestone** | **Deliverable** | **Effort(Person Day)** | **Start Date** | **End Date** |
| Prototype Test Specification | Test Cases/Test data | 4 | October 15 | October 19 |
| Prototype Test Execution | Test Results,  Test Defect Reports | 3 | October 20 | October 23 |
| Prototype Test Evaluation | Test Evaluation Report | 1 | October 24 | October 25 |