*<Project Name>*

Test Plan Template

**Release *<1>***

***<16.04.2024> - <10.04.2024>***

VERSION HISTORY

*[Provide information on how the development and distribution of the* ***Test Plan****, up to the final point of approval, was controlled and tracked. Use the table below to provide the version number, the author preparing the version, the date of the version, the name of the person approving the version, the date that particular version was approved, and a brief description of the reason for creating the revised version.]*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID & Version #** | **Prepared**  **By** | **Revision**  **Date** | **Approved**  **By** | **Approval**  **Date** | **Reason** |
| unique id | *person name* | *< date>* | *person name* | *<date>* | release number |
|  | Armine Melikyan | 16/05/2024 | Gohar Khachatryan | 17/05/2024 | 01 |

TABLE OF CONTENTS

[**1**](#_heading=h.3znysh7) **Introduction 4**

[1.1](#_heading=h.2et92p0) Purpose of The Test Plan Document 4

[**2**](#_heading=h.tyjcwt) **Test ITEM 4**

[2.1](#_heading=h.3dy6vkm) Project description 4

[2.2](#_heading=h.1t3h5sf) Items to be Tested / Not Tested 4

[2.3](#_heading=h.4d34og8) Items to be excluded 4

[2.4](#_heading=h.2s8eyo1) Test Approach(s) 4

[2.5](#_heading=h.17dp8vu) Test Deliverables 4

[2.6](#_heading=h.3rdcrjn) Staffing / Training Needs 5

[**3**](#_heading=h.26in1rg) **Risk and mitigation 5**

[3.1](#_heading=h.lnxbz9) Test Risks / Issues 5

[**4**](#_heading=h.35nkun2) **Test Environment and infrastructure 5**

[4](#_heading=h.1ksv4uv) Required Infrastructure 5

[**5**](#_heading=h.44sinio) **Roles and responsibilities 5**

[5.1 Roles and assigned responsibilities 5](#_heading=h.2jxsxqh)

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

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

[**6**](#_heading=h.1y810tw) **Test Schedule 8**

[6.1](#_heading=h.4i7ojhp) 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 <Product and Version> 01 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

*Courses web page: Where the user should register, after can fill his/her account’s balance with credit card and buy courses.*

## Items to be Tested / Not to be Tested

*Registration page, Login page, Account balance page,*

|  |  |  |  |
| --- | --- | --- | --- |
| **Item to Test** | **Test Description** | **Test Date** | **Estimation** |
| User Story # 1 | Registration page | <16/05/24> | 3pt |
| User Story # 2 | Develop Login page | <25/05/24> | 1pt |
| User Story #3 | Profile page | <06/06/24> | 5pt |
| User Story #4 | Account Balance page | <16/06/24> | 5pt |
| User Story #5 | Add / Edit / View / Delete Course | <21/07/24> | 5pt |
| User Story #6 | Buy course |  | 3pt |

## Items to Not be tested

|  |  |
| --- | --- |
| **Item Not to Test** | **Comment** |
| Weather | It is integrated via API |
| Home, About us pages | Will not be tested in this release |
|  |  |

## Test Approach(s)

*We are planning to test manually (in the next stage and regression testing we will automate). We will use* ***Static test*** *and review* ***the documentations*** *(Test basis): requirements to understand business and end user requirements and needs, to clarify understandings to avoid misunderstandings omission, inaccuracy and to offer and discuss our team suggestion. After we will use* ***Blackbox*** *testing: mostly* ***Boundary*** *and* ***Equivalent partitioning****, for the parts where is maximum 3 fields (line) we will use* ***Decision Table*** *technic and also will performAPI testing.*

**Automated Testing Approach** – We are planning to automate recreation tests.

**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** –GUI will be tested along with other testing methods for the whole website.

**Integration Testing**  - We will use API testing for integrated parts of project to check integrated parts work together.

**Regular Bug Triages** – Who is responsible for ensure that bugs are being prioritized on a regular basis, which QA, DEV, BA.

**Testing Execution and Bug Tracking** – We will use Jira integrated in it Zephyr for QA, BA, Dev. team communication, for documentation we will use Swagger

**Bug Severity and Priority Setting** – We will do static testing and review in order to minimize ambiguity and ensure everyone has a common understanding. Also, we try to keep communication with BA and Client to understand Business needs and discuss Found bags effect on the project. Bugs will be reported in the Jira management tools and it will allow controlling all the process for all stakeholders.

**Description of the types of testing done and the testing period for this release** – We will do Functional testing, API testing for integration, Performance testing.

## 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 | <16/05/24> |
| *Test Case preparation* | <20/06/24> |
| *Test Case review* | <30/06/24> |
| Manual Testing | <01/07/24> |
| Automation script preparation | <20/07/24> |
| Automation code review | <30/07/24> |
| Regression testing(manual+automation) | <01/08/24> |

## 

## Staffing / Training Needs

*We need training to increase/learn automation testing skills and to learn to execute it. (Selenium, Python, Java)*

# Risk and mitigation

## Test Risks / Issues

For the current project, we as a risk see the overloading of the Server.

# Test Environment and infrastructure

## Required Infrastructure

*We will test the project the educational courses web page in 2 environments: Development and Production. We will use* ***Jira*** *management tool where is integrated* ***Zephy****r tool, we will use Postman for API testing, the system will be Windows 11, the browser will be Google Chrome.*

[*https://qwallity-prod.onrender.com/*](https://qwallity-prod.onrender.com/)

[*https://qwallity-dev.onrender.com/*](https://qwallity-dev.onrender.com/)

# Roles and responsibilities

## Roles and assigned responsibilities

|  |  |
| --- | --- |
| **Role** | **Responsibility** |
| Jounior QA | Test assigned pieces of the project |
| Mid QA | Test assigned pieces of the project, API testing |
| Senior QA | Writing test plan, regulate QA team work |
| BA | Provide information about the project, keep ongoing contact with client, presents business needs |
| PM | Task division, control workflow, time management |
| 1. Backend Developers, (mid, junior, sen.) | Provide the project Front page pieces, dry run |
| 3 Front End Developers (mid, junior, sen.) | Provide the project Front page pieces, dry run |

## 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

*[Describe the describe key milestones, deliverables, efforts, start date and end date]*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Milestone** | **Deliverable** | **Effort(Person Hour)** | **Start Date** | **End Date** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# 