

|  |  |
| --- | --- |
| ROQ Main Contact Details: | <<Client>> Main Contact Details |
| * <<Jake Faulkner>> | * <<Name>> |
| * <<Graduate Software Tester>> | * <<Title>> |
| * <<07982099185>> | * <<Telephone No>> |
| * <<jake.faulkner@roq.co.uk>> | * <<Email Address>> |

<<Credersi>>

<<Publishing Date>>

<<ROQExpress CredersiVend>>

Version <<1>>

Contents

[1. Introduction 3](#_Toc256774621)

[1.1. Project Background 3](#_Toc256774622)

[1.2. Purpose 3](#_Toc256774623)

[1.3. Test Objectives 3](#_Toc256774624)

[2. Scope 3](#_Toc256774625)

[2.1. Test Scope – Inclusions 3](#_Toc256774626)

[2.1.1. Systems Under Test 3](#_Toc256774627)

[2.1.2. Features Under Test 3](#_Toc256774628)

[2.2. Test Scope – Exclusions 3](#_Toc256774629)

[3. Approach 3](#_Toc256774630)

[4. Acceptance Criteria 3](#_Toc256774631)

[4.1. Entry Criteria 3](#_Toc256774632)

[4.2. Exit Criteria 3](#_Toc256774633)

[4.3. Suspension Criteria 3](#_Toc256774634)

[5. Tasks and Deliverables 3](#_Toc256774635)

[5.1. Test Project Plan 3](#_Toc256774636)

[5.2. Test Milestones 3](#_Toc256774637)

[5.3. Test Deliverables 3](#_Toc256774638)

[6. Roles and Responsibilities 3](#_Toc256774639)

[7. Test Environment Needs 3](#_Toc256774640)

[8. Test Data Needs 3](#_Toc256774641)

[9. Staffing and Training Needs 3](#_Toc256774642)

[10. Test and Defect Management 3](#_Toc256774643)

[10.1. Test Management 3](#_Toc256774644)

[10.2. Defect Management 3](#_Toc256774645)

[11. Assumptions 3](#_Toc256774646)

[12. Constraints 3](#_Toc256774647)

[13. Risks 3](#_Toc256774648)

[14. Templates 3](#_Toc256774649)

[15. Document Control 3](#_Toc256774650)

[15.1. Document Review 3](#_Toc256774651)

[15.2. Change History 3](#_Toc256774652)

[15.3. Distribution 3](#_Toc256774653)

[15.4. Referenced Documents 3](#_Toc256774654)

[15.5. Document Sign-off 3](#_Toc256774655)

1. Introduction
   1. Project Background

This is explanation text to help a staff member use the template effectively. All text of this type should be removed from the document before submitting it for formal review.

Provide a high level explanation of the project background, and ROQ’s involvement in the project.

The Project that our team at ROQexpress have been Tasked to complete, is the Testing of the CredersiVend Admin Software Created by Credersi. So, within this project we have been asked to test the software to make sure that at least the minimal viable product is functional and working to the standard they want. The Minimal viable product contains the possibility for the user to create Customers/Company Names, then create routes one main route which breaks of into many sub-routes however there can only be one main access point within software.

* 1. Purpose

*The purpose of this document is to describe the scope and high-level approach for the work to be undertaken, along with supporting material on factors that will affect the testing*.

The purpose of this document is to make sure you the reader is able to understand the product we have been tasked to test and make sure that you can understand what we have tested, why we have chosen to test the certain aspects we have selected and also what testing techniques we have used during the process, and why we have chosen these specific testing techniques.

* 1. Test Objectives

State the major objectives of the testing.

* Ensure Confidence in the Product
* Ease of Access for the user
* Ensure product performs as intended and report defects or concerns appropriately

1. Scope
   1. Test Scope – Inclusions
      1. Systems Under Test

Define the system under test. Summarise its constituent testable systems, sub-systems or component parts. Include versions where possible.

| Item | Purpose | Version |
| --- | --- | --- |
| Localhost 80:80 | Process Data from the Database | 1 |
| Routes | The Routes folder is the main source pathway that links the backend and frontend | 1 |
| Backend | To provide frontend with how it’s meant to operate | 1 |
| Frontend | To provide the user a clean and friendly interface | 1 |

* + 1. Features Under Test

Summarise all the major features and combinations of features to be tested, include non-functional aspects and documentation references, e.g. Business Requirements. Include versions where possible.

| Feature | Sub-Feature | Reference Document and Version |
| --- | --- | --- |
| Check Login Function |  |  |
| Check Functionality |  |  |
| Database Functionality |  |  |
| Check Environment Functionality |  |  |
| Entering a new Vending Machine |  |  |
| Check Create Customer |  |  |
| Username Field |  |  |
| Password Field |  |  |

* 1. Test Scope – Exclusions

Identify all items and/or software features excluded from the scope of testing, explain why. Note that this information is often more important than specifying what is in scope.

Due to time constraints non-functional testing such as performance testing and high-level component testing outside the most viable testing of the product

1. Approach

If the plan covers multiple types or phases of testing then this section can be split into an Overall Approach and a separate specific approach for each of the testing types. Where the different types of testing will be handled in a very different way it is better to have multiple test plan documents instead.

Specify the major activities, techniques and tools that are to be used to test the items and features in scope.

The approach should be sufficiently detailed to enable identification of the major tasks and estimation of time to do each one. When developing the approach consider:

The test policy/strategy (if applicable)

Significant constraints on testing, e.g. test resource and/or environment availability, time, etc and the impact on the approach

Risk based testing

How much testing is required: too much is a waste of time and money, as is too little and the wrong testing is the worse for little or no gain

The type of industry

Contractual, legal, regulatory or specific customer requirements

Team experience

Resource availability

Documentation availability

Etc.

Define the test specification document requirements, how many specifications will be required, map these to the items/features identified as being in scope.

State the expected number of test cycles.

Indicate how test coverage and completion will be determined.

Define how the tests will be developed and test results captured. Describe how repeatability is assured, the extent of regression needs and how these are supported.

The approach to the testing will be scrum approach to testing, this will be split into sprints in which the most useful tests will be performed in the 1st sprint. Due to the time constraints on the testing this is going to cause significant issues with how much testing that will be manageable and has an impact on how many tests will be conducted. With this sprint we look to cover all of the test levels to prove the confidence in the product and feed back any defects or concerns at the review stage at the end of the sprint, this will be split into different test case documentation for each test level we are looking to conduct.

The test team is early in their experience and will aid in the development and implementation of tests with the resources that have been provided by credersi vend such as code and read me documentation of how the app works and how to install the application onto the ROQ laptops given. Risks could include issues setting up the environments on the testers system and any issue that may come from this.

1. Acceptance Criteria
   1. Entry Criteria

Detail the entry criteria that must be satisfied prior to commencing the test execution. (Include consideration for the system under test handover requirements - how is its content and status defined and determined, how are known outstanding issues identified?).

Entry Criteria that must be met is that the piece of software must run to the level that the developer wanted, that the complete or partially completed testable code is readily available. Having test Environment availability set up, having a test scope definition and finalisation, the types of testing we are going to carry out on the software, which test cycle we think is appropriate to use and having a concrete schedule/plan in place. Having the required test resources readily available to hand to use, coder has been deployed within test environment and also that the test data is readily available.

* 1. Exit Criteria

Specify the exit criteria that will be used to judge the completeness of the test activity – the comprehensiveness and completeness of testing. This section addresses how the system under test will be deemed fit for purpose; when this point is reached the testing activity should stop.

So, for our exit criteria we hope that the test case and scope have been correctly executed and have passed all our expectations, also that all the known and identified defects have been solved within the software. The Test Cycle has been executed and fully completed before the timeline is complete. Making sure no de-scoping as it were occurring whilst working on the project. Making sure that out gathered test results are available to everyone on the project and the clients most importantly.

* 1. Suspension Criteria

Specify the circumstances and criteria in which all or a portion of the testing activity might be suspended. Specify in what circumstances and how the test activity will be restarted and what tasks must be undertaken.

In cases that we could potentially implement the use of Suspension criteria within our testing process, would only happen if we come across a defect that is highly impacting our testing process, maybe if the correct hardware and software has not been readily available to us this could be a cause for suspension as we will most likely not be able to work without the correct hardware and software. Our testing process could possibly be limited by defects within the main build which could then lead to the suspension of our project.

1. Tasks and Deliverables
   1. Test Project Plan

Define the project plan which governs the test phases and activities, summary Gantt chart style, showing testing tasks, timescales, dependencies and milestones, and resource assignments. Show how this fits with the overall project plan.

The plan for our project is, on Monday our official start day we are starting to create the Test plan and Test strategy documentation for our project, The main aim for the Monday session is to try and get most of this documentation complete, currently at the end of the day on Monday we have successfully started creating the test plan/strategy documentation and then within the last 2 hour session we started to open up and build the CredersiVend application which unfortunately we ran into a very big blocker that the Maven Dependencies plug-in doesn’t seem to be working at all, which we will try to fix on Tuesday 6th.

Tuesday the 6th within this session our plan is to try firstly and mainly to get our Eclipse Workspace up and running hopefully within the morning session which zak had tasked an outside source of Floz to help with this. Meanwhile andy and Junaid started to create some test cases for our project unfortunately trying to make test cases without the actual use of the software has been a challenge. For Jake He has been Writing out the Test Plan and making sure that it will be finished by the end of the day. Creating some smoke tests as well within our test case creation block within the afternoon

Wednesday the 7th The main goal that we had planned for this session was to hopefully have our test plan/strategy documentation fully written down and proofread by one another for the most accurate depiction of what we are creating. But mainly we want to start using the test cases we created on Tuesday to start building some tests around the CredersiVend software. And completing any documentation that remains to be completed within the morning session.

Thursday the 8th, during this session we want to continue creating test cases and writing tests by using Postman API, Junit Tests, Selenium Tests and lastly cucumber and selenium testing. Hopefully we will get a big bulk of these tests completed within the Thursday session as this will be close to the end of the week and primarily our focus is to hopefully have at least half of our test cases written down and running within our test cases.

Friday the 9th last day of the week, do a morning recap go over everything we have covered so far make sure we are on track to finish by Tuesday the 13th next week.

Monday 12th making sure that we have completed all our created test cases and making sure they relate to our user stories. Last overview of the completed documentation making sure its been proofread by floz and maddy.

Tuesday 13th Cut-off point for creating our test cases and making sure that all the documentation is completed, we want to ideally be finished with all the technical documentation by the end of Tuesday this is so that we can make a start on our project presentation on Wednesday.

Here is a basic overview of our timetable.

Table

Description automatically generated

Test Milestones

Document all the milestones associated with this test plan.

| Task | Milestone | Planning Date |
| --- | --- | --- |
| High level task, e.g. Test Build | The milestone being met, for example build complete | The initial planned date to hit this milestone |
| Create Test strategy | Test Strategy Complete | Tuesday 6th |
| Create user stories | User Stories complete | Tuesday 6th |
| Create test cases | Test Cases created | Tuesday 6th/ Wednesday7th |
| Create test scripts | Test scripts created | Wednesday7th/Friday 9th |
| Create Presentation | Presentation created | Tuesday/Wednesday13th |

* 1. Test Deliverables

Document all the deliverables associated with this test plan.

| Deliverable | Description | Task |
| --- | --- | --- |
| The physical item to be delivered, e.g. a Test Specification | Describe the deliverable in the context of the planned work, e.g. This document records the testable requirements for system x | The associated high level task, e.g. Test Analysis |
| Github | This will include the commits and the final code from the automated tests and the test documentation.  Defect reports will also be included as a deliverable to show the finding of defects to credersi. |  |
| Trello | Trello will include the user stories and what user stories we will be undertaking within the current sprint and the progress of what these tasks are at. |  |
| Eclipse Automated tests | We will be able to deliver automated test scripts for the backend and routes components. |  |
| Visual studio Automated tests | We will be able to deliver automated test scripts on the front end component. |  |
|  |  |  |

1. Roles and Responsibilities

Define the roles which need to be undertaken to fulfil the plan, and list the responsibilities of that role. Also record who will undertake each role. Remember that each role can potentially be carried out by more than one person, and a single person can potentially undertake multiple roles.

| Role | Responsibilities | Person(s) |
| --- | --- | --- |
| Tester | Involved in the whole project | Andy heskett |
| Tester | Involved in the whole project | Jake Faulkner |
| Tester | Involved in the whole project | Junaid raja |
| Tester | Involved in the whole project | Zak Hilton |

Please note that a single member of staff may be able to undertake more than one role, subject to time constraints.

1. Test Environment Needs\*

Where testing requires several environments with different attributes then it is advisable to split these out into separate sub headings.

Define the test environment and the support and controls needed. Identify any aspects which will need construction and build (e.g. harnesses or simulators).

Include non-computer system requirements e.g. rooms and other facilities.

Identify what exists and what needs to be procured or commissioned.

Make clear how peripheral devices might be allocated, shared and used.

Make clear how external interfaces to other systems or outside the organisation are supported.

Summarise any key administrative requirements and functions (e.g. backup/restore, batch execution and schedule management, printer management).

Requirements for the test environment will be meeting rooms so that we can conduct meeting as well as a place to work with the members of the test team to develop test scripts. Booking out the meeting rooms at the ROQ head office will be put in place so that there is no risk to the development of the tests during the sprints.

Eclipse environment – within the eclipse environment we want to be creating Junit tests

Vsstudio – Within the Visual studio we are going to create a selenium and cucumber environment to create our test cases for them

Postman – postman could potentially be an add on but due to the time constraints of the project this could be delayed to the next sprint.

1. Test Data Needs

Describe the data strategy, how it supports the testing defined in this plan and how data will be built or generated.

Make clear how data is aged if this is required in the test approach.

Make clear how production data is sourced, and any timing or data protection issues associated with production data use.

* Use of the test data already provided by credersi vend
* Adding our own test data
* Building 3 different test environments for each test levels
* Creating test automated scripts to repeat tests

1. Staffing and Training Needs

There are no staffing needs for this project, Training on software’s such as postman for the test scripts will be needed to conduct API tests to provide quality tests on the product.

1. Test and Defect Management
   1. Test Management

The work will be split up, as we have 4 members, we can equally split of within 2 separate groups of 2 people and work on certain aspects of the project whilst also updating each other on what we are currently up to and completing throughout the day and making sure that we are up to date. Making sure all the work is planned everyday within the morning session our 15-minute standup from 9 – 9:15 so we know what we’re doing on the daily and when to convene with each other throughout the day and making sure that we are on track for the day.

* 1. Defect Management

If we come across defects, they will be categorised by threat levels, with 5 being the lowest and 1 being the highest. If we come across a priority level 1 defect this will be the issue to be solved first and focussed on and then as the priority lowers, they will be completed but there not on the top of the list when it comes to defects.

1. Assumptions

Record any assumptions used during the preparation of this plan. Assumptions are typically positive things which enable project work, but cannot currently be demonstrated or proved to be true. Also explicitly state what the impact to the plan would be if the assumption proved to be false or incorrect.

All assumptions need to be agreed by an appropriate authority, usually the client project manager (although some items are better confirmed by subject matter experts); record who agreed the validity of the assumption and when. The plan should not be signed off without all the assumptions first being explicitly agreed to.

| Description | Impact | Agreed By | Agreed Date |
| --- | --- | --- | --- |
| Testing on all components | High | Zak |  |
| Presenting the findings | Medium | Andy |  |
| Up to date versions of the app will be provided | High | Junaid |  |
|  |  |  |  |

1. Constraints

Constraints are things which do or definitely will restrict the way that work is carried out. Typically they are unequivocal and factual in nature. As well as stating a constraint it is essential to explain the impact this constraint will have on the testing in terms of efficiency, scope or risk.

The constraints need to be signed off by a suitable authority, typically the client project manager (though sometimes a subject matter expert is better placed to agree the constraints); the purpose of this is to ensure the client is aware of any constraints and so can potentially choose to do something about them.

| Description | Impact | Agreed By | Agreed Date |
| --- | --- | --- | --- |
| Time constraints set by Credersi | high | High | Zak |
| Developer for routes left | High | High | Andy |
| Inexperience of the team | medium | High | Jake |
|  |  |  |  |

1. Risks

A risk is something which might happen in the future and if it did would negatively impact the project. The risk is given numeric rating for impact (1-4) and likelihood (1-3). These numbers are multiplied together to generate the overall Risk Factor; the higher the risk factor the more effort should be invested to ameliorate the risk. Each risk needs an owner who is charged with monitoring the risk and taking proportionate steps to see that the risk does not occur. Alternatively, a low risk can simply be accepted by the project.

During the planning phase it is often possible to express risks as Assumptions or Constraints.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description | Impact | Likelihood | Risk Factor | Owner |
| Unseen absence | 2 | 2 | High | Zak |
| Software not working | 4 | 2 | High | Andy |
| Internet issues | 4 | 1 | High | Jake |
|  |  |  |  |  |

1. Templates

State which set of document templates will be used to support the project. This should be either the ROQ set of document templates, the client’s set of templates or some (defined) combination of the two.

* Test policy
* Test strategy
* Gant chart

1. Document Control
   1. Document Review

Record who has participated in both the internal and external reviews of the document. Where a person reviews a document multiple times it is only necessary to record the last date of review.

| Name | Role Title | Date |
| --- | --- | --- |
| Jake Falkner | Graduate test engineer | 06/09/2022 |
| Zak Hilton | Graduate test engineer | 06/09/2022 |
| Andy Heskett | Graduate Test engineer | 08/09/2022 |
|  |  |  |

* 1. Change History

Record the different versions of the document which get distributed. Each new version of the document should receive a minor increment (e.g. from 0.01 to 0.02) unless the document is a major revision (e.g. the document has been signed off)(e.g. from 0.03 to 1.00). The level of information recorded in the Description of Changes field depends on the amount of sign-off received. Where external sign-off has been achieved the changes should be recorded in a high level of detail.

Not every version needs approval. Generally approval indicates passing internal review and therefore being made available to the client, or passing external review and being signed off by the client.

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Description of Changes | Approval |
| 1.0 | 06/09/2022 | Creation of initial document | Zak Hilton |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

* 1. Distribution

List the people who the document will ultimately be circulated to. The roles are:

Review: The named person will review the document and feedback

Approval: The named person will approve the document on behalf of their organisation

Information: The names person may be interested in the document but is not reviewing or approving the document

|  |  |  |
| --- | --- | --- |
| Name | Organisation | Document Role |
| Amira | ROQ | Information |
| Floz | Credersi Vend | Review |
| Steven Knowles | ROQ | Approval |
|  |  |  |

* 1. Referenced Documents

List all the documents referenced in the production of this document. Each item needs a number so it can be uniquely identified. The document title and version should be specified. Finally, it is important to record who or where the document is available from so a reader of this document can get copies of all the references documents.

|  |  |  |
| --- | --- | --- |
| Ref | Document and Version | Available from |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

* 1. Document Sign-off

This document has been reviewed, and approved for issue at the indicated issue status by the following:

<<Client>> Project Manager or Authorised Representative

|  |  |
| --- | --- |
| Name: |  |
| Position: |  |
| Signature: |  |
| Date: |  |

ROQ Test Project Manager or Authorised Representative

|  |  |
| --- | --- |
| Name: |  |
| Position: |  |
| Signature: |  |
| Date: |  |