Test plan for

QAMP project task

*General information*

|  |  |
| --- | --- |
| Customer | Unknown Business owner |
| Created by | Eldina Delić |
| Preparation date | 13.12.2021 |
| Version | 1.1 |
| Status |  |

*Revision history*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Author | Date | Approved by | |
| 1.1 | Eldina Delić | 12.12.2021. | Author | Date |
|  |  |  |  |  |

1.Introduction

Test Plan is prepared to verifiy the parking managment project element to commision the system for operation and evaluate the functional deployment. The Test Plan is also intended to demonstrate to project stakeholders that acequate consideration has been given to all aspects of the testing effort to demonstrate readiness of parking system for production use. This test plan purposes a review of recent advances sensing nd communiction technology concering parking systems. Furthermore, the system will be tested and evaluetes by stakeholders.The test plan identify the items to be tested,the resources and schedule, the types of testing to be performed... Finally, this plan discusses the benefits of engaging the stakeholders to develop the prototype. It is meant to be used as manual during testing works, but in the future it is planned to automated.

1.2.The Test Plan supports the following ***specific objectives***:

The test objectives are to verify the Functionality of parking system, the project should focus on testing the parking system such as parking lot machine and parking ramp to guarantee that all operations can work normally in real business environment.

* Functional and non-functional requirements (Functional requirements explain
* how the system must work, while non functional requirements explain how the system should perform)
* Demonstrate that all contract requirements have been provided
* Verify neccessary functions, communications, and operational interfaces
* Identify how each testavle requirement will be demonstrated
* Identify the results that will constitute success for each test
* Identify all the deliverables that will results from completing testing
* Ensure that issues are identified and fixed before go-live

**2.Scope**

2.1. Scope of the project

Testing of Manual QA- Mistral Test Project is in the scope of this test plan. The following components and functions would be tested:

1.Insert coins and banknotes by user

2.Scan ticket code

3.Test parking lot

4.parking lot machine

5.payment

***Test Logistic***

* Who will test?

-Test engineer (QA process/ logging found errors into the approvedbug tracking system)

-Project Manager: reviews the content of the Test Plan, Test Strategy and Test Estimates signs off on it

* When will test occur?

The tester will start the test execution when all the following inputs are ready:

* Software is available for testing
* Test Specification is created
* Test Environment is built
* Enough human resources for testing

***Test Team***

* Develop test conditions, test cases, expected results, and execution scripts.
* Perform execution and validation.
* Identify, document and prioritize defects according to the guidance provided by the Test lead.
* Re-test after software modifications have been made according to the schedule.
* Prepare testing metrics and provide regular status.

***Development team***

* Review testing deliverables (test plan, cases, scripts, expected results, etc.) and provide timely feedback.
* Assist in the validation of results (if requested).
* Support the development and testing processes being used to support the project.
* Certify correct components have been delivered to the test environment at the points specified in the testing schedule.
* Keep project team and leadership informed of potential software delivery date slips based on the current schedule.
* Define processes/tools to facilitate the initial and ongoing migration of components.
* Conduct first line investigation into execution discrepancies and assist test executors in creation of accurate defects.
* Implement fixes to defects according to schedule.

***Test Criteria***

1. *Suspension Criteria*

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

2*.Exit Criteria*

Specifies the criteria that denote a successful completion of test phase

-Run rate is mandatory to be 100% unless a clear reason is given

-Pass rate is 80%, achiving the pass rate is mandatory

3.Resources

The following tools will be used for this project:

|  |  |
| --- | --- |
| Name of process | Tool |
| Defect tracking | Jira |
| Test Cases | Microsoft Excel |

4.Human resources

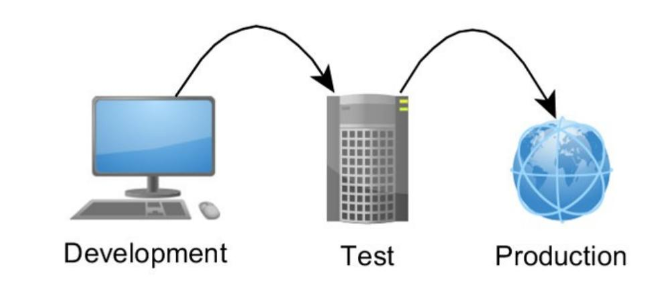
|  |  |
| --- | --- |
| 1.Test Manager | Manage the whole project  Define project directions  Acquire appropriate resources |
| 2.Test | Identifying and describing techniques/tools/automation architecture.  Verify and assess the Test Approach.  Execute the tests, Lopg results, Report the defects.  Outsourced members. |
| 3.Developer in test | Implement the test cases, test program, test suite etc. |
| 4. Test Administrator | Support tester to use the test environment for test execution |
| 5. SQA members | Check to confirm whether the testing proces is meeting specified requirments. |
| 6. Parking operator | A parking operator who has a contract to show parking availability on the parking systems |

Defect managment

|  |  |
| --- | --- |
| **Severity** | **Impact** |
| 1 (Critical) | * This bug is critical enough to crash the system, cause file corruption, or cause potential data loss * It causes an abnormal return to the operating system (crash or a system failure message appears). * It causes the application to hang and requires re-booting the system. |
| 2 (High) | * It causes a lack of vital program functionality with workaround. |
| 3 (Medium) | * This Bug will degrade the quality of the System. However there is an intelligent workaround for achieving the desired functionality - for example through another screen. * This bug prevents other areas of the product from being tested. However other areas can be independently tested. |
| 4 (Low) | * There is an insufficient or unclear error message, which has minimum impact on product use. |
| 5(Cosmetic) | * There is an insufficient or unclear error message that has no impact on product use. |

**Test environment**

Over here, we are going to see the various aspects related to test environment and its importance in software testing.

****

A testing environment is nothing but setup of hardware and software requirements for the testing teams to run the test cases. As such, it provides support for test excution with software, hardware and network arranged.

**A QA environment** is where we test upgrade procedure against data, hardware and software that closely stimulate the Production environment. In the present case we have parking lot machine (where it will be tested inserting coins and banknotes, scan tickets, ticket printing) and also testing how the ramp works when car is incoming or outcoming.

**A production environment** is where the latest version of software, products or updates are pushed live to the intended users. All testing is completed before this point and all bugs are removed.

5.Main testing type

***System testing*** : conducted on a complete, integrated system to evaluate the system's compliance with specified requirements

**System Testing** is a level of testing zhaz validates zhe complete and fully integrazed software product. The purpose of a systrm test is to evaluete the end-to-end-system specifications. Usually, the software is only one element of a larger computer-based system. Ultimatelly, the software is interfaced with other software / hardware systems. System Testing is actually a series of different tests whose sole purpose is to exercise the full computer-based system.

**Usability Testing**- mainly focuses on the user's ease to use the application, flexibility, in handing controls and ability of the system to meet its objecitives

**Equivalence Partioning** (Equivalence Class Partioning) is a software testing technique that divides the input data of a software unit into partitions of equivalent data from which test cases can be derived. In principle, test cases are designed to cover each partition at least once.

**Acceptance testing**- beta testing of the product done by the actual end users

**Test deliverables**

Test deliverables are provided as below

Before testing phase

-test plans document

-test cases documents

-test design specifications

**During the testing**

-Test Tool

-Simulators

-Test Data

-Test Trace-ability Matrix

-Error logs and execution logs

**After the testing cycles is over**

-Test results/reports

-defect report

-installation

-release notes