**Test Plan**

**Quality Engineer Kelompok 9**

**Web UI, Rest API and Mobile**

**Joko Mart**

# Table of Contents

[**Table of Contents**](#_heading=h.sirmes2ilxr9) **2**

[**I. Introduction**](#_heading=h.30j0zll) **3**

[1.1 Purpose](#_heading=h.3znysh7) 3

[1.2 Background](#_heading=h.2et92p0) 3

[1.3 Scope](#_heading=h.tyjcwt) 3

[1.4 Project Identification](#_heading=h.3dy6vkm) 4

[**II. Requirement**](#_heading=h.ac6b7tnvwl0c) **5**

[**III. Test Strategy**](#_heading=h.25hwypye8394) **5**

[3.1. Test Types](#_heading=h.jr2m2ty40cbo) 5

[3.1.1 Web UI](#_heading=h.17dp8vu) 5

[3.1.2 Mobile](#_heading=h.9qnf6ev792gq) 7

[3.1.3 API](#_heading=h.gdchblh7irz) 10

[3.2. Tools](#_heading=h.o0sftnuwunoz) 10

[**IV. Resources**](#_heading=h.bfweosuxaok1) **11**

[4.1 System](#_heading=h.35nkun2) 11

[**Project Milestones**](#_heading=h.s8ye2e3m8q8w) **12**

[**Deliverables**](#_heading=h.kpw5tzma70oc) **12**

[**Appendix A: Project Tasks**](#_heading=h.b7wbq3fjicgs) **13**

# **Introduction**

Test Plan is a document used to describe in detail how the test will be carried out. It gives the information, the approach, the complete plan of testing. It is the basis for software testing. The system that will be tested is the web UI, API, and mobile application of Loyalty Point Agent. The test plan describes important information about the test project, any resources that will be used for testing, any features that will be tested, etc.

## 1.1 Purpose

The purpose of this test plan document is to :

1. Identify features or components of the software that are to be tested.
2. Define the type of testing that will be applied to the test.
3. Determine the tools to be used for testing.

## 1.2 Background

Testing needs to be done to ensure that the software or system being tested is in accordance with the requirements made. In addition, testing is also carried out to identify possible bugs that appear on the newly created system. To perform the test, a test plan document is required. This test plan is used as a reference in conducting tests so that the expected goals and results can be achieved.

## 1.3 Scope

This section contains what is being tested, includes testing the functions of a specific product, test manually, user interface, and automation of each feature on the web such as the authorized user (register, login, logout), homepage (add product, view product, select category), Shopping cart (view transaction), and transaction. For mobile app, the scope to be tested is using testing that is the same as web UI, just the feature on mobile that will be tested is auth, and homepage.

This document contains testing of the Joko Mart web system. The scope to be tested includes testing the functionality, manually, user interface, and automation of each feature on the web such as the authorized user (register, login, logout), homepage (add product, view product, select category), Shopping cart (view transaction), and transaction. For mobile app, the scope to be tested is using testing that is the same as web UI, just the feature on mobile that will be tested is auth, and homepage.

## 1.4 Project Identification

| Document | Created or Available | Received or Reviewed | Author or Resources |
| --- | --- | --- | --- |
| Requirements Specification | ⓒ Yes ⓒ No | ⓒ Yes ⓒ No | Eragradianysah and Aryanti |
| Functional Specification | ⓒ Yes ⓒ No | ⓒ Yes ⓒ No | Eragradianysah and Aryanti |
| Project Plan | ⓒ Yes ⓒ No | ⓒ Yes ⓒ No | Eragradianysah and Aryanti |

# **Requirement**

Requirements for this test include :

For Web UI the system being tested was : JokoMart website. For API the API URL being tested, which is:

https://documenter.getpostman.com/view/23491997/2s8YmSqznk

For Mobile app is JokoMart.apk.

Tools that are used for testing such as :

1. Katalon Studio for automation testing for all software.
2. Google Docs for document test and report.
3. Google Sheet for test documentation.
4. Android Studio used for emulator mobile testing.

Both tools are used for all systems that will be tested.

# **Test Strategy**

## 3.1. Test Types

### 3.1.1 Web UI

1. Function Testing

| Test Objective | Ensure the functionality of all features run as expected. |
| --- | --- |
| Technique | Test every button on the system is running well.  Test every form input with various conditions |
| Completion Criteria | Verify whether the input displayed matches the input made. Verify the result are same as expected |

1. User Interface Testing

| Test Objective | Ensure all the components in each form can work |
| --- | --- |
| Technique | Test every input form can display data the same as expected. |
| Completion Criteria | The appearance of the application is easy to use by the user |

1. Integration Testing

| Test Objective | Combines all the existing cases in the feature so that the feature can run well over. |
| --- | --- |
| Technique | Test every input form can make result same as expected |
| Completion Criteria | The appearance of the application is easy to use by the user |

1. Manual Testing

| Test Objective | Ensures whether the application is working manually, as expected or not and to identify bugs, ensure the product is error-free. |
| --- | --- |
| Technique | Analyze system.  Create a document test plan.  Create scenarios  Create test case  Finding bugs and errors and fixed it |
| Completion Criteria | Verify requirement.  Verify the result are same as expected |

1. Automation Testing

| Test Objective | to automate repetitive tests which are difficult to perform manually. and can run test cases repeatedly. |
| --- | --- |
| Technique | Get the object  Create test cases  Run test suite  Check the result and make a report. |
| Completion Criteria | Verify every object that is to be used. Verify the results are the same as expected. |

### 3.1.2 Mobile

1. Function Testing

| Test Objective | Ensure the functionality of all features run as expected. |
| --- | --- |
| Technique | Test every button on the system is running well.  Test every form input with various conditions |
| Completion Criteria | Verify whether the input displayed matches the input made.  Verify the result are same as expected |

1. User Interface Testing

| Test Objective | Ensure all the components in each form can work |
| --- | --- |
| Technique | Test every input form can display data the same as expected. |
| Completion Criteria | The appearance of the application is easy to use by the user |

1. Integration Testing

| Test Objective | Combines all the existing cases in the feature so that the feature can run well over. |
| --- | --- |
| Technique | Test every input form can make result same as expected |
| Completion Criteria | The appearance of the application is easy to use by the user |

1. Manual Testing

| Test Objective | Ensures whether the application is working manually, as expected or not and to identify bugs, ensure the product is error-free. |
| --- | --- |
| Technique | Analyze system.  Create a document test plan.  Create scenarios  Create test case  Finding bugs and errors and fixed it |
| Completion Criteria | Verify requirement.  Verify the result are same as expected |

1. Automation Testing

| Test Objective | to automate repetitive tests which are difficult to perform manually. and can run test cases repeatedly. |
| --- | --- |
| Technique | Get the object  Create test cases  Run test suite  Check the result and make a report. |
| Completion Criteria | Verify every object that is to be used. Verify the results are the same as expected. |

### 3.1.3 API

1. Function Testing

| Test Objective | Ensure the functionality of all endpoint, request, and response data. |
| --- | --- |
| Technique | Test endpoint and request url with various conditions. |
| Completion Criteria | Verify response data.  Verify the result are same as expected |

1. Automation Testing

| Test Objective | To automate repetitive tests which are difficult to perform manually. and can run test cases repeatedly. |
| --- | --- |
| Technique | Get endpoint  Set method  Create request body and test cases  Run test suite  Check the result and make a report. |
| Completion Criteria | Verify every endpoint and method request body that is to be used. Verify the results are the same as expected. |

## 3.2. Tools

|  | Tool | Vendor | Version |
| --- | --- | --- | --- |
| Web UI, Mobile and API Testing Tools | Katalon Studio | Katalon | 8.2.5 |
| Test Case Document | Google Sheet | Google | 2022 |
| Mobile Test Emulator | Android Studio | Android Studio | 2021.3.1 |
| Test Plan and Report | Google Docs | Google Docs | 2022 |

# **Resources**

## 4.1 System

The following is a list of equipment requirements from the implementation of the test.

| System Resources | |
| --- | --- |
| Resource | Name/Type |
| Web UI  API  Mobile App | <https://alta-shop.vercel.app/>  <https://documenter.getpostman.com/view/5781191/2s7YYpekQS#a76392f3-40f2-4d22-8032-b3c742ec083b>  app-release.apk |
| Laptop  Processor  RAM  Operation System | AMD Ryzrn 7 4800  16 GB  Windows 10 Pro 64-bit |
| Software Test | Katalon Studio  Google Sheet  Android Studio |

## 

# **Project Milestones**

| No | Milestone Task | Start date | End date |
| --- | --- | --- | --- |
| 1. | Plan Test | 24/10/2022 | 26/10/2022 |
| 2. | Design Test | 26/10/2022 | 28/10/2022 |
| 3. | Implement Test | 28/10/2022 | 30/10/2022 |
| 4. | Execute Test | 30/10/2022 | 2/11/2022 |
| 5. | Report Test | 2/11/2022 | 4/11/2022 |

# **Deliverables**

The deliverable of the tests that have been carried out is in the form of a list of test results documentation. As :

1. Document test plan.

That is a document created using Microsoft Word that contains a plan on how the testing will be carried out and the requirements and types of tests that will be implemented.

1. Document matrix

Contains information about the modules that will be tested on each system complete with what features are in each module. This document will make it easier to create test scenarios that will be made.

1. Document Test Scenario.

Is a document that contains information about what test scenarios will be run complete with test cases, test data, and expected results.

1. Report

Contains an overall report related to the results of tests that have been carried out on each system.

# **Appendix A: Project Tasks**

| 1. | Plan Test |
| --- | --- |
| 2. | Identify Requirements for Test |
| 3. | Design Test |
| 5. | Identify and Describe Test Cases |
| 6. | Implement Test |
| 7. | Record or Program Test Scripts |
| 8. | Execute Test |
| 9. | Evaluate Execution of Test |
| 10. | Report Test |