

ALTERRA ACADEMY

Test Plan Website Forum Group Discussion, API, Mobile

DOCUMENT TEST PLAN

Made By Group 15:

|  |  |
| --- | --- |
| Muhammad Yogi | UI/UX |
| Muhammad Ramanda Julyansyah | UI/UX |
| Elfrida R.D. Tampubolon | QE |
| Reyhan Septri Asta | Flutter |
| M Hendro Junawarko | JVS |
| Jonathan Iskandar | JVS |
| Lucky Nursarastri | JVS |
| Gde Agung Mandala Bendesa | React |
| Muhamad Jaya | React |
| M.Zaky Syukur | React |
| Muhammad Hafiz Hisbullah | React |

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Description | Author |
| 26-05-2022 | 1.0 | Automation Testing, API Testing, Web Testing, Mobile Testing, and UI Testing | Elfrida R.D. Tampubolon |

# TABLE OF CONTENTS

[1. INTRODUCTION 4](#_Toc103711349)

[1.1. Purpose 4](#_Toc103711350)

[1.2. Background 4](#_Toc103711351)

[1.3. Scope 5](#_Toc103711352)

[1.3.1 Web Testing 5](#_Toc103711353)

[1.3.2 API Testing 5](#_Toc103711354)

[1.3.3 Mobile Testing 6](#_Toc103711355)

[1.4 Definitions of Acronyms and Abbreviations 6](#_Toc103711356)

[2. REQUIREMENTS FOR TEST 6](#_Toc103711357)

[3. TEST STRATEGY 6](#_Toc103711358)

[3.1. Testing Types 6](#_Toc103711359)

[3.1.1 Web Testing 6](#_Toc103711360)

[3.1.2 API Testing 9](#_Toc103711361)

[3.1.3 Mobile Testing 11](#_Toc103711362)

[3.2 Tools 13](#_Toc103711363)

[4. RESOURCES 13](#_Toc103711364)

[4.2 System 14](#_Toc103711365)

[5. PROJECT MILESTONES 14](#_Toc103711366)

[6. DELIVERABLES 14](#_Toc103711367)

[6.1 Test Models 15](#_Toc103711368)

[7. APPENDIX A: PROJECT TASKS 15](#_Toc103711369)

[LAMPIRAN 16](#_Toc103711370)

# INTRODUCTION

In this introduction, we will discuss the purpose, background, scope and definitions of acronyms and abbreviations the testing that will be carried out.

## Purpose

The purpose of making a test plan in general is to make it easier for developers to carry out testing so that the testing carried out becomes clear so that the results are more useful and efficient.

1. Testing checks for functionality, and usability of the website.
2. Tests check functionality, and usability of mobile applications.
3. Tests the methods of API.
4. To verify it fulfills its expected functionality.
5. Provide limits on testing so that in further development it can complete testing

## Background

Software is a series of instructions for the computer that perform a particular task, called a program; the two major categories of software are system software and application software. System software is made up of control programs [1]. Software testing is the process to make the software system defect free. In this process, a comparison is done between the expected results and actual results. It also gives a hand to identify errors, gaps or missing requirements in contrary to the actual requirements. A software product should only be released after it has gone through a proper process of development, testing and bug fixing. Software testing can be done manually and automatically [2].

Manual testing is the most basic and rigorous type of software testing. It is the most crucial phase of all testing types. In this software testers write the test cases using human resources without making use of any special software. The main purpose of manual testing is to prepare a software product that should be defect-free. It should work according to the functional requirements specified by the customer. The automatic technique to write and execute test cases, where no human intervention is involved. This process is known as Automated Testing. In this process, the tester writes scripts with the use of suitable software to test the application. It is basically an automation process of a manual process where no human intervention is required. Automated Testing is a simplified process of manual testing.

Researchers often find bugs or errors after the application or software is submitted to the user. To reduce the occurrence of bugs or errors, it is very necessary to do both manual testing and automation testing. But on this occasion, the researcher will do automation testing.

In this test plan document, we will do website testing, mobile testing, web UI, and software code will be automated using CI/CD. Website testing is a software testing technique exclusively adopted to test the applications that are hosted on the web in which the application interfaces and other functionalities are tested. API testing is a software testing type that validates Application Programming Interfaces (APIs). The purpose of API Testing is to check the functionality, reliability, performance, and security of the programming interfaces. Mobile testing is the process by which applications for modern mobile devices are tested for functionality, usability, performance and much more. CI/CD is a method to frequently deliver [apps](https://www.redhat.com/en/topics/cloud-native-apps) to customers by introducing [automation](https://www.redhat.com/en/topics/automation) into the stages of [app development](https://www.redhat.com/en/topics/cloud-native-apps/why-choose-red-hat-cloud-native). The main concepts attributed to CI/CD are continuous integration, [continuous delivery](https://www.redhat.com/en/topics/devops/what-is-continuous-delivery), and continuous deployment. Continuous Integration (CI) and Continuous Delivery (CD) use automation to ensure that new application code is always tested, secure, and ready to use so teams can deliver to production on time.

## Scope

In this test plan, the tester will do Web Testing, Mobile Testing, and Web UI.

### 1.3.1 Web Testing

On the Forum Group Discussion Web page, functional testing is carried out on 2 main features, namely Home and Transactions. On the User Web page, the tester will test the login, register, and discussion with friends. On the transaction page, the tester will test the function of the product, the function of adding or reducing the number of products. The tester will perform automated tests using IntelliJ.

### 1.3.2 API Testing

In the Forum Group Discussion API, we can test the Get method to get data, and also the Post method to add data. We can do this method on Categories, Products, Authentication, Transactions, Ratings, and Comments.

### 1.3.3 Mobile Testing

In the Forum Group Discussion mobile application, just like the Forum Group Discussion website, we can perform authentication, view products, make product purchases, and can also add products to the cart.

## 1.4 Definitions of Acronyms and Abbreviations

Table 1. Definitions of Acronyms and Abbreviations

|  |  |  |
| --- | --- | --- |
| No | Definition | Explanation |
|  | API | Application Programming Interface |
|  | Developer | People who create or develop software. |
|  | CI | Continuous Integration |
|  | CD | Continuous Delivery |
|  | UI | User Interface |
|  | Hardware | Devices that support computer systems |
|  | Tools | Applications used during project work |

# REQUIREMENTS FOR TEST

Requirements for this test includes the system being tested, and tools that are used for testing such as IntelliJ, Vysor, Git and Microsoft Excel. Then, tools are used for the test documentation such as Microsoft Word.

# TEST STRATEGY

The strategy consists of all plans carried out to carry out testing on software built.

## Testing Types

### 3.1.1 Web Testing

1. Function Testing

Table 2. Function Testing

|  |  |
| --- | --- |
| **Test Objective:** | * Form input (all the form needs the input data) so that the form can be processes. * The report form can generate transaction results in accordance with existing inputs and processes. |
| **Technique:** | * Testing each button on the form * Test the input form with various input conditions. |
| **Completion Criteria:** | The output issued is in accordance with the inputs and transactions already made. |
| **Special Considerations:** |  |

1. User Interface Testing

Table 3. User Interface Testing

|  |  |
| --- | --- |
| **Test Objective:** | Ensure all the components in each form can work well |
| **Technique:** | * *Login form*:   **Input**: Enter characters to perform sql injection,  Perform brute force passwords.  **Output**: The login form can only accept users who have access rights.   * *Register form:*   **Input:** Enter valid characters for full name, email, and perform brute force passwords.  **Output:** The registration form can only create a new account if the characters are valid. |
| **Completion Criteria:** | The appearance of the software is easy to use by the user |
| **Special**  **Considerations:** |  |

1. Integration Testing

Table 4. Integration Testing

|  |  |
| --- | --- |
| **Test Objective:** | Combines all the existing cases in the feature so that it can run continuously |
| **Technique:** | * Test each button on each page (register, login, etc) * Test various input conditions on the input page |
| **Completion Criteria:** | * When the transaction is carried out, it is expected that the output matches the input * Ensure that all report result are as expected |
| **Special Considerations:** | - |

1. Manual Testing

Table 5. Manual Testing

|  |  |
| --- | --- |
| **Test Objective:** | To avoid bugs it is necessary to check all application features to function properly |
| **Technique:** | * Analyze requirements * Create test case * Create test scenario * Looking for bugs (errors) * Fixed bugs (errors) |
| **Completion Criteria:** | After testing all feature can function properly |
| **Special**  **Considerations:** |  |

1. Automation Testing

Table 6. Automation Testing

|  |  |
| --- | --- |
| **Test Objective:** | To avoid bugs it is necessary to check all application features to function properly |
| **Technique:** | * Analyze requirements * Create test case * Implementing automation testing with IntelliJ * Fixed bugs (errors) |
| **Completion Criteria:** | After testing all feature can function properly |
| **Special**  **Considerations:** |  |

### API Testing

1. Function Testing

Table 7. Function Testing

|  |  |
| --- | --- |
| **Test Objective:** | * Testing the Get and Post methods on each endpoint * Testing the response code generated from the request |
| **Technique:** | * Test each endpoint with various condition |
| **Completion Criteria:** | * Provide the appropriate response code * Data was successfully added or retrieved |
| **Special Considerations:** |  |

1. Integration Testing

Table 8. Integration Testing

|  |  |
| --- | --- |
| **Test Objective:** | Combines all the existing cases in the feature so that it can run continuously |
| **Technique:** | * Test each endpoint with various condition * Test combine all cases in features |
| **Completion Criteria:** | * Provide the appropriate response code * Data was successfully added or retrieved |
| **Special Considerations:** | - |

1. Automation Testing

Table 9. Automation Testing

|  |  |
| --- | --- |
| **Test Objective:** | Provides higher efficiency, effectiveness and accuracy |
| **Technique:** | * Analyze requirements * Create test case * Implementing automation testing with IntelliJ * Fixed bugs (errors) |
| **Completion Criteria:** | * Provide the appropriate response code * Data was successfully added or retrieved |
| **Special**  **Considerations:** |  |

### Mobile Testing

1. Function Testing

Table 10. Function Testing

|  |  |
| --- | --- |
| **Test Objective:** | * Form input (all the form needs the input data) so that the form can be processes. * The report form can generate transaction results in accordance with existing inputs and processes. |
| **Technique:** | * Testing each button on the form * Test the input form with various input conditions. |
| **Completion Criteria:** | The output issued is in accordance with the inputs and transactions already made. |
| **Special Considerations:** |  |

1. User Interface Testing

Table 11. User Interface Testing

|  |  |
| --- | --- |
| **Test Objective:** | Ensure all the components in each form can work well |
| **Technique:** | * *Login form*:   **Input**: Enter characters to perform sql injection,  Perform brute force passwords.  **Output**: The login form can only accept users who have access rights.   * *Register form:*   **Input:** Enter valid characters for full name, email, and perform brute force passwords.  **Output:** The registration form can only create a new account if the characters are valid. |
| **Completion Criteria:** | The appearance of the software is easy to use by the user |
| **Special**  **Considerations:** |  |

1. Integration Testing

Table 12. Integration Testing

|  |  |
| --- | --- |
| **Test Objective:** | Combines all the existing cases in the feature so that it can run continuously |
| **Technique:** | * Test each button on each page (register, login, etc) * Test various input conditions on the input page |
| **Completion Criteria:** | * When the transaction is carried out, it is expected that the output matches the input * Ensure that all report result are as expected |
| **Special Considerations:** | - |

1. Manual Testing

Table 13. Manual Testing

|  |  |
| --- | --- |
| **Test Objective:** | To avoid bugs it is necessary to check all application features to function properly |
| **Technique:** | * Analyze requirements * Create test case * Create test scenario * Looking for bugs (errors) * Fixed bugs (errors) |
| **Completion Criteria:** | After testing all feature can function properly |
| **Special**  **Considerations:** |  |

1. Automation Testing

Table 14. Automation Testing

|  |  |
| --- | --- |
| **Test Objective:** | To avoid bugs it is necessary to check all application features to function properly |
| **Technique:** | * Analyze requirements * Create test case * Implementing automation testing with IntelliJ * Fixed bugs (errors) |
| **Completion Criteria:** | After testing all feature can function properly |
| **Special**  **Considerations:** |  |

## 3.2 Tools

Table 15. Tools

|  |  |  |
| --- | --- | --- |
| Tool | Description | Version |
| IntelliJ IDEA | Automation Testing Tool | 2022 |
| Microsoft Excel | Test Cases, Test Scenario, Role Matrix | 2010 |
| Microsoft Word | Test Plan | 2010 |
| Vysor | Android Control Tool | 4.1.77 |
| Github | CI/CD Tool |  |

# RESOURCES

This sub-chapter describes the recommended resources for testing Forum Group Discussion Web, API, and Mobile to perform transactions on Forum Group Discussion web.

## 4.2 System

The following will describe the hardware and software requirements detailed in the hardware implementation environment table and the software implementation environment table.

Table 16. Hardware

|  |  |  |
| --- | --- | --- |
| **No** | **Hardware** | **Spesification** |
| 1 | *Processor* | Intel(R) Core(TM) i7-8750H CPU @ 2.20GHz 2.21 GHz |
| 2 | *Memory* | 8,00 GB |
| 3 | *Hardisk* | 1 TB |

Table 17. Software

|  |  |  |
| --- | --- | --- |
| **No** | **Software** | **Spesification** |
| 1 | *IntelliJ* |  |
| 2 | *Microsoft Excel* |  |
| 3 | *Microsoft Word* |  |
| 4 | *Operating System* | Windows 11 |
| 5 | *Browser* | *Google Chrome* |
| 6 | *Web Forum Group Discussion* | https://qa.alta.id |

# PROJECT MILESTONES

Table 18. Project Milestones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Milestone Task | Effort | Start Date | End Date |
|  | Test Plan Document | 100% | 26/04/2022 | 28/04/2022 |
|  | Role Matrix | 100% | 29/04/2022 | 30/04/2022 |
|  | Test Cases | 100% | 1/05/2022 | 6/05/2022 |
|  | Execution Test Cases | 100% | 7/05/2022 | 17/05/2022 |

# DELIVERABLES

The results of this test are reports from Forum Group Discussion web testing, API testing, mobile testing, and automation using Github Actions. The reports are in the form of test plans, role-matrixes, test cases, and bug reports.

## 6.1 Test Models

Testing is done by using black box testing.

# APPENDIX A: PROJECT TASKS

Table 19. Appendix

|  |  |
| --- | --- |
|  | Test Plan |
|  | Identify Module |
|  | Identify Test Scenario |
|  | Identify Test Case |
|  | Execute Test Cases |
|  | Report Bug |

# LAMPIRAN

[1] Padmini C., “Beginners Guide To Software Testing,” *Softw. Test. Cl.*, pp. 1–41, 2013.

[2] H. H. Syed, “a Comparison Between Manual Testing,” vol. 5, no. May, pp. 323–331, 2021.