Test Plan Document

Final Project - Pengujian Perangkat Lunak

]

[[RS01-1](https://github.com/mustafabaihaqi/RS01-1)]

Website Platform

Version 1.0.0

Created at May, 2025

**Table of Contents**

[I. DOCUMENT CONTROL 3](#_49x2ik5)

[i. Document Detail 3](#_2et92p0)

[ii. Version History 3](#_tyjcwt)

[1. Introduction 4](#_1t3h5sf)

[1.1 Objectives 4](#_2s8eyo1)

[1.2 Team Members 4](#_3rdcrjn)

[2. Scope 5](#_lnxbz9)

[2.1 Feature to be tested 5](#_35nkun2)

[2.2 Feature not to be tested (Items outside of scope) 10](#_1ksv4uv)

[2.3 Test Type 10](#_44sinio)

[3. Test Approach 10](#_2jxsxqh)

[4. Test Environment 14](#_2xcytpi)

[5. Milestones / Deliverables 14](#_qsh70q)

# Document Control

## Document Detail

| **Title** | *RS01-1 - Test Plan Document* |
| --- | --- |
| **Version** | *Kelompok 1* |
| **Date** | *08-May-2025* |
| **Contributors** | 1. *BAIHAQI MUSTAFA SURYA ATMAJA* 2. *RASYID BASKORO AGUNG* 3. *ADRIAN SYAH ABIDIN* |
| **Approver** | *PM* |
| **Document Status** | *Finished* |

## Version History

| **Version** | **Contributor** | **Issue Date** | **Description** | **[V]** |
| --- | --- | --- | --- | --- |
| v1.0d | *Adrian Syah Abidin* | *08-05-2025* | Create initial testplan |  |
| v2.0d |  |  |  |  |
| v3.0d |  |  |  |  |
| v4.0d |  |  |  |  |
| V5.0d |  |  |  |  |
|  |  |  |  |  |

# Introduction

Test Plan document explains how the software built can run according to the plan that has been set. Test plan is a document that is the basis used to test the suitability of software design results with predetermined objectives. With a well-detailed test plan, all aspects of the software can generally be covered and the results of the design are in accordance with the development plan that has been made before. Testing involves checking all software modules, business processes, interface interactions, software performance, and the relationship between pages in the source code.

## Objectives

The test plan will encompass both automated and manual testing methods.

1. Manual Test

The tester checks manually without using tools or scripts, the aim is to ensure that the application being tested is defect-free and the software application can work as expected. Manual testing also plays an important role when testing visuals where automation tools cannot do it.

1. Automation Test

Relying on pre-scripted tests that run automatically, its function is to compare expected results with actual results. So you can find out whether the application is running as expected, using automated testing which can be done repeatedly. So if the results are not the same as expected, you will get a bug.The test plan document is created to ensure the running of the android-based application for e-tickets can run according to the needs, including:

* Ensuring the tested application is in accordance with functional and non-functional requirements.
* Identifying components that will be tested
* Make recommendations and describe the testing strategy that will be carried out
* Identifying resource requirements such as databases

## Team Members

| **Resource Name** | **Role** |
| --- | --- |
| Baihaqi Mustafa Surya Atmaja | Backend Engineer |
| Rasyid Baskoro Agung | Frontend Engineer |
| Adrian Syah Abidin | QA Engineer |

# Scope

This document only discusses an application designed for both mobile and web platforms, facilitating ticket booking. The scope to be tested includes testing the source code, performance, security, and accuracy of the software to be created. In addition, testing will also be carried out on each form in the software. Testing is only done with a tester. The key features include

## Feature to be tested

Outline the items/features/functions that will be tested, which were defined in Product Requirements Document are need to bested

| **Testing Phase** | **Test item** | **Applicable roles** |
| --- | --- | --- |
| API Manual Testing |  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
| API Automation Testing |  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
| Web Automation Testing |  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
| Performance |  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |
|  | User |

A scope is needed in carrying out work so that the work produces optimal results.

## Feature not to be tested (Items outside of scope)

These feature are not be tested because they are not included in Product Requirements Document

1. Application Security

## Test Type

In the project FinSera, there are several types of testing that should be conducted.

| **Testing by** | **Test Type** |
| --- | --- |
| **Execution** | Manual Testing |
|  | Automation Testing |
| **Scope** | Regression Testing |
|  | Smoke Testing |
|  | Sanity Testing |
| **Functionality** | UI Testing |
|  | Integration Testing |
|  | API Testing |
| **Non Functionality** | Performance Testing |
| **Scenario** | Positive Test |
|  | Negative Test |

# Test Approach

The project is using an agile approach, with weekly iterations. At the end of each week the requirements identified for that iteration will be delivered to the team and will be tested. Exploratory testing will play a large part of the testing as the team has never used this type of tool and will be learning as they go. Tests for planned functionality will be created and added to TCT as we get iterations of the product.

| **Categori** | **Description** |
| --- | --- |
| Testing Purposes | This test has the main objective of validating the system as a whole and seeing whether the integration process and performance of each feature of the airline ticket booking platform are as required. |
| Scope Of Testing | All the main features of the ticket booking application are registration, login, plane ticket booking, payment and platform interface responsibility on various devices. |
| Testing Methodology | Testing is carried out using 2 methods, namely manual and automatic. Manual testing is carried out to test user interface feedback and automatic testing is carried out for regression and performance. |
| Testing Environment | Manually execute the test cases and document the results. If a test case fails, document the details of the failure so that it can be used for debugging. |
| Type of Testing | Using 2 types of testing, namely functional testing and non-functional testing. Functional testing is carried out by executing test data and checking whether the functional software is working properly. Test data is generated from software specifications. Non-functional testing is carried out by using test data to test all elements of a software program such as application performance and usability. |
| Test Plan | For each scenario, create detailed test cases. Each test case should include the test steps, expected result, and actual result. |
| Testing Passing Criteria | All software functional requirements are met and software performance is in accordance with requirements. |

# Test Environment

A new server is required for the web server, the application and the database.

| OS Version | Browser | Application |
| --- | --- | --- |
| Windows 11 | Google Chrome | Postman Newman JMeter Intelij Idea |

# Milestones / Deliverables

## Test Schedule

The initial test schedule follows :

| **Task Name** | **Start** | **Finish** | **Effort**  **(Duration)** | **Comments** |
| --- | --- | --- | --- | --- |
| Test Planning Sprint | 15-07-2024 | 18-07-2024 | 3 days | No problem occur |
| Test Scenario and Case Sprint 1 | 19-07-2024 | 21-07-2024 | 2 days | No problem occur |
| Manual API Testing Sprint 1 | 22-07-2024 | 24-07-2024 | 2 days | Very few features available for testing |
| Manual Mobile Testing Sprint 1 | 25-07-2024 | 27-07-2024 | 2 days | Very few features available for testing |
| Test Scenario and Case Sprint 2 | 28-07-2024 | 30-07-2024 | 2 days | No problem occur |
| Manual API Testing Sprint 2 | 31-07-2024 | 02-08-2024 | 2 days | No problem occur |
| Manual Mobile + Web Testing Sprint 2 | 03-08-2024 | 06-08-2024 | 3 days | No problem occur |
| Automated API Testing Sprint 2 | 07-08-2024 | 10-08-2024 | 3 days | No problem occur |
| Automated API Testing Sprint 3 | 11-08-2024 | 14-08-2024 | 3 days | No problem occur |
| Automated Mobile + Web Testing Sprint 3 | 15-08-2024 | 18-08-2024 | 3 days | No problem occur |
| Performance Testing | 19-08-2024 | 22-08-2024 | 3 days | No problem occur |
| Test Reporting Sprint | 23-08-2024 | 25-08-2024 | 2 days | No problem occur |

## Deliverables

| **Deliverable** | **For** | **Date / Milestone** |
| --- | --- | --- |
| Test Plan | Project Manager; QA Director; Test Team | 15-07-2024 |
| Test Results | Project Manager | 25-08-2024 |
| Test Status report | QA Manager, QA Director | 25-08-2024 |
| Metrics | All team members | 25-08-2024 |