

---

**Aniflix**

---

**Aniflix App  
Master Test Plan  
Version <1.1>**

Aniflix App	Version: <1.1>
Master Test Plan	Date: <14/07/20>
<document identifier>	

## Revision History

Date	Version	Description	Author
07.05.2020	<1.0>	Created Test Plan	Kevin Bobbe Pascal Stephan
14.07.2020	<1.1>	Update Test Plan	Kevin Bobbe

Aniflix App	Version: <1.1>
Master Test Plan	Date: <14/07/20>
<document identifier>	

# Table of Contents

1.	Introduction	5
1.1	Purpose	5
1.2	Scope	5
1.3	Intended Audience	5
1.4	Document Terminology and Acronyms	5
1.5	References	5
2.	Evaluation Mission and Test Motivation	5
2.1	Background	5
2.2	Evaluation Mission	5
2.3	Test Motivators	5
3.	Target Test Items	6
4.	Outline of Planned Tests	6
4.1	Outline of Test Inclusions	6
4.2	Outline of Other Candidates for Potential Inclusion	6
4.3	Outline of Test Exclusions	6
5.	Test Approach	7
5.1	Initial Test-Idea Catalogs and Other Reference Sources	7
5.2	Testing Techniques and Types	7
5.2.1	Function Testing	7
5.2.2	User Interface Testing	7
6.	Entry and Exit Criteria	8
7.	Deliverables	8
7.1	Test Evaluation Summaries	8
7.2	Reporting on Test Coverage	8
7.3	Perceived Quality Reports	8
7.4	Incident Logs and Change Requests	8
7.5	Smoke Test Suite and Supporting Test Scripts	8
7.6	Additional Work Products	8
8.	Testing Workflow	8
9.	Environmental Needs	9
9.1	Base System Hardware	9
9.2	Base Software Elements in the Test Environment	9
9.3	Productivity and Support Tools	9
9.4	Test Environment Configurations	9
10.	Responsibilities, Staffing, and Training Needs	9
10.1	People and Roles	9
10.2	Staffing and Training Needs	11

Aniflix App	Version: <1.1>
Master Test Plan	Date: <14/07/20>
<document identifier>	

11.	Iteration Milestones	11
12.	Risks, Dependencies, Assumptions, and Constraints	12

Aniflix App	Version: <1.1>
Master Test Plan	Date: <14/07/20>
<document identifier>	

# Master Test Plan

## 1. Introduction

### 1.1 Purpose

The purpose of the Iteration Test Plan is to gather all of the information necessary to plan and control the test effort for a given iteration. It describes the approach to testing the software, and is the top-level plan generated and used by managers to direct the test effort.

This *Test Plan* for the Aniflix App supports the following objectives:

- Feature Files
- Unit Tests
- Widget Tests

### 1.2 Scope

- Feature Files → testing of the UI
- Unit Tests → test a few of our functions
- Widget Tests → test Custom UI Components

### 1.3 Intended Audience

This document is intended for the use of the Aniflix Team.

### 1.4 Document Terminology and Acronyms

n.a.

### 1.5 References

- SAD
- SRS

## 2. Evaluation Mission and Test Motivation

Testing is done to provide a stable software, that does, what it is intended to do!

### 2.1 Background

n.a.

### 2.2 Evaluation Mission

- find as many bugs as possible
- find important problems, assess perceived quality risks
- advise about product quality, satisfy stakeholders
- advise about testing

### 2.3 Test Motivators

quality risks

technical risks

use cases

design elements

Aniflix App	Version: <1.1>
Master Test Plan	Date: <14/07/20>
<document identifier>	

### 3. Target Test Items

The listing below identifies those test items—software, hardware, and supporting product elements—that have been identified as targets for testing. This list represents what items will be tested.

- Feature Files:
  - Functionality of the Navigation bar
  - UI and Functions of the Homepage
  - UI and Functions of the Settings Page
- Unit Tests:
  - Change Theme
  - API Calls
- Widget Tests:
  - Custom Components

### 4. Outline of Planned Tests

All Tests will be done this Iteration

#### 4.1 Outline of Test Inclusions

n.a.

#### 4.2 Outline of Other Candidates for Potential Inclusion

n.a.

#### 4.3 Outline of Test Exclusions

n.a.

Aniflix App	Version: <1.1>
Master Test Plan	Date: <14/07/20>
<document identifier>	

## 5. Test Approach

### 5.1 Initial Test-Idea Catalogs and Other Reference Sources

n.a.

### 5.2 Testing Techniques and Types

#### 5.2.1 Function Testing

Technique Objective:	Exercise target-of-test functionality, including navigation, data entry, processing, and retrieval to observe and log target behavior.
Technique:	<p>Execute each use-case scenario's individual use-case flows or functions and features, using valid and invalid data, to verify that:</p> <ul style="list-style-type: none"> <li>the expected results occur when valid data is used</li> <li>the appropriate error or warning messages are displayed when invalid data is used</li> <li>each business rule is properly applied</li> </ul>
Oracles:	We assume that all test pass.
Required Tools:	n.a.
Success Criteria:	All tests pass
Special Considerations:	-

#### 5.2.2 User Interface Testing

Technique Objective:	<p>Exercise the following to observe and log standards conformance and target behavior:</p> <p>Navigation through the target-of-test reflecting business functions and requirements, including window-to-window, field-to-field, and use of access methods (tab keys, mouse movements, accelerator keys).</p> <p>Window objects and characteristics can be exercised—such as menus, size, position, state, and focus.</p>
Technique:	Feature Files
Oracles:	We assume all tests to pass.
Required Tools:	Flutter Gherkin
Success Criteria:	All tests pass.
Special Considerations:	-

Aniflix App	Version: <1.1>
Master Test Plan	Date: <14/07/20>
<document identifier>	

### 5.2.3 Widget Testing

Technique Objective:	Exercise the following to observe and log standards conformance and target behavior:  Window objects and characteristics can be exercised—such as menus, size, position, state, and focus.
Technique:	Build each Custom Component and check if they contain the expected Subcomponents and the correct number of each Subcomponent
Oracles:	We assume all tests to pass.
Required Tools:	n.a.
Success Criteria:	All tests pass.
Special Considerations:	-

## 6. Entry and Exit Criteria

n.a.

## 7. Deliverables

n.a.

### 7.1 Test Evaluation Summaries

Summary after each pull Request.

Summary contains the Code Coverage and the outcome of the tests.

### 7.2 Reporting on Test Coverage

Codecov.io

### 7.3 Perceived Quality Reports

n.a.

### 7.4 Incident Logs and Change Requests

n.a.

### 7.5 Smoke Test Suite and Supporting Test Scripts

n.a.

### 7.6 Additional Work Products

Feature Files in the GitHub

## 8. Testing Workflow

On each Pull Request to the master branch all Tests are running.

At least one other Team member must approve the Pull Request.

Merging is only possible if all Tests pass and one other Team member approves the Code.



Aniflix App	Version: <1.1>
Master Test Plan	Date: <14/07/20>
<document identifier>	

## 9. Environmental Needs

### 9.1 Base System Hardware

The following table sets forth the system resources for the test effort presented in this *Test Plan*.

n.a.

### 9.2 Base Software Elements in the Test Environment

The following base software elements are required in the test environment for this *Test Plan*.

Software Element Name	Version	Type and Other Notes
Android Studio		Emulator
GitHub		CI Environment

### 9.3 Productivity and Support Tools

The following tools will be employed to support the test process for this *Test Plan*.

Tool Category or Type	Tool Brand Name	Vendor or In-house	Version
Test Management	Codecov.io		
Project Management	YouTrack	JetBrains	

### 9.4 Test Environment Configurations

The following Test Environment Configurations needs to be provided and supported for this project.

n.a.

Configuration Name	Description	Implemented in Physical Configuration
Average user configuration		
Minimal configuration supported		
Visually and mobility challenged		
International Double Byte OS		
Network installation (not client)		

## 10. Responsibilities, Staffing, and Training Needs

### 10.1 People and Roles

This table shows the staffing assumptions for the test effort.

Aniflix App	Version: <1.1>
Master Test Plan	Date: <14/07/20>
<document identifier>	

Human Resources		
Role	Minimum Resources Recommended (number of full-time roles allocated)	Specific Responsibilities or Comments
Test Manager	1	<p>Provides management oversight.</p> <p>Responsibilities include:</p> <ul style="list-style-type: none"> <li>• planning and logistics</li> <li>• agree mission</li> <li>• identify motivators</li> <li>• acquire appropriate resources</li> <li>• present management reporting</li> <li>• advocate the interests of test</li> <li>• evaluate effectiveness of test effort</li> </ul>
Test Analyst	1	<p>Identifies and defines the specific tests to be conducted.</p> <p>Responsibilities include:</p> <ul style="list-style-type: none"> <li>• identify test ideas</li> <li>• define test details</li> <li>• determine test results</li> <li>• document change requests</li> <li>• evaluate product quality</li> </ul>
Test Designer	2	<p>Defines the technical approach to the implementation of the test effort.</p> <p>Responsibilities include:</p> <ul style="list-style-type: none"> <li>• define test approach</li> <li>• define test automation architecture</li> <li>• verify test techniques</li> <li>• define testability elements</li> <li>• structure test implementation</li> </ul>

Aniflix App	Version: <1.1>
Master Test Plan	Date: <14/07/20>
<document identifier>	

Human Resources		
Role	Minimum Resources Recommended (number of full-time roles allocated)	Specific Responsibilities or Comments
Tester	3	Implements and executes the tests. Responsibilities include: <ul style="list-style-type: none"> <li>• implement tests and test suites</li> <li>• execute test suites</li> <li>• log results</li> <li>• analyze and recover from test failures</li> <li>• document incidents</li> </ul>
Designer	1	Identifies and defines the operations, attributes, and associations of the test classes. Responsibilities include: <ul style="list-style-type: none"> <li>• defines the test classes required to support testability requirements as defined by the test team</li> </ul>
Implementer	3	Implements and unit tests the test classes and test packages. Responsibilities include: <ul style="list-style-type: none"> <li>• creates the test components required to support testability requirements as defined by the designer</li> </ul>

## 10.2 Staffing and Training Needs

This section outlines how to approach staffing and training the test roles for the project.

n.a.

## 11. Iteration Milestones

Milestone	Planned Start Date	Actual Start Date	Planned End Date	Actual End Date
> 20% Test Coverage	07.05.2020	07.05.2020	07.06.2020	12.06.2020
Have Functional Tests	07.05.2020	07.05.2020	07.06.2020	07.06.2020
Have Unit Tests	07.05.2020	07.05.2020	07.06.2020	07.06.2020
Have Widget Tests	07.05.2020	07.05.2020	07.06.2020	12.06.2020
Tests Integrated in CI	07.05.2020	07.05.2020	07.06.2020	10.05.2020
Iteration ends			07.06.2020	12.06.2020

Aniflix App	Version: <1.1>
Master Test Plan	Date: <14/07/20>
<document identifier>	

## 12. Risks, Dependencies, Assumptions, and Constraints

Risk	Mitigation Strategy	Contingency (Risk is realized)
Unit Tests don't work with the App.	<Test Designer> will design new Tests that work with the App	<ul style="list-style-type: none"> <li>Publish the test</li> </ul>