

---

**Aniflix**

---

**Aniflix App  
Master Test Plan  
Version <1.0>**

Aniflix App	Version: <1.0>
Master Test Plan	Date: <dd/mmm/yy>
<document identifier>	

## Revision History

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

Aniflix App	Version: <1.0>
Master Test Plan	Date: <dd/mmm/yy>
<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	6
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	7
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	8
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	9
7.6	Additional Work Products	9
8.	Testing Workflow	9
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	10
10.	Responsibilities, Staffing, and Training Needs	10
10.1	People and Roles	10
10.2	Staffing and Training Needs	12

Aniflix App	Version: <1.0>
Master Test Plan	Date: <dd/mmm/yy>
<document identifier>	

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

Aniflix App	Version: <1.0>
Master Test Plan	Date: <dd/mmm/yy>
<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
- UnitTests
- 

### 1.2 Scope

- Feature Files → testing of the UI
- UnitTests → test a few of our functions
- 

### 1.3 Intended Audience

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

### 1.4 Document Terminology and Acronyms

n.a.

[This subsection provides the definitions of any terms, acronyms, and abbreviations required to properly interpret the **Test Plan**. Avoid listing items that are generally applicable to the project as a whole and that are already defined in the project's Glossary. Include a reference to the project's Glossary in the References section.]

### 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.

[Provide a brief description of the background surrounding why the test effort defined by this **Test Plan** will be undertaken. Include information such as the key problem being solved, the major benefits of the solution, the planned architecture of the solution, and a brief history of the project. Where this information is defined in other documents, you can include references to those other more detailed documents if appropriate. This section should only be about three to five paragraphs in length.]

### 2.2 Evaluation Mission

n.a.

[Provide a brief statement that defines the mission for the evaluation effort in the current iteration. This statement might incorporate one or more concerns including:

- find as many bugs as possible
- find important problems, assess perceived quality risks

Aniflix App	Version: <1.0>
Master Test Plan	Date: <dd/mmm/yy>
<document identifier>	

- advise about perceived project risks
- certify to a standard
- verify a specification (requirements, design or claims)
- advise about product quality, satisfy stakeholders
- advise about testing
- fulfill process mandates
- and so forth

Each mission provides a different context to the test effort and alters the way in which testing should be approached.]

## 2.3 Test Motivators

n.a.

[Provide an outline of the key elements that will motivate the testing effort in this iteration. Testing will be motivated by many things—quality risks, technical risks, project risks, use cases, functional requirements, non-functional requirements, design elements, suspected failures or faults, change requests, and so forth.]

## 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 Navigationbar
  - UI and Functions of the Homepage
  - UI and Functions of the Settings Page
- UnitTests:
  - Change Theme
- 

## 4. Outline of Planned Tests

n.a.

[This section provides a high-level outline of the testing that will be performed. The outline in this section represents a high level overview of both the tests that will be performed and those that will not.]

### 4.1 Outline of Test Inclusions

n.a.

[Provide a high level outline of the major testing planned for the current iteration. Note what will be included in the plan and record what will explicitly **not** be included in the section titled Outline of Test Exclusions.]

### 4.2 Outline of Other Candidates for Potential Inclusion

n.a.

[Separately outline test areas you suspect might be useful to investigate and evaluate, but that have not been sufficiently researched to know if they are important to pursue.]

Aniflix App	Version: <1.0>
Master Test Plan	Date: <dd/mmm/yy>
<document identifier>	

### 4.3 Outline of Test Exclusions

n.a.

[Provide a high level outline of the potential tests that might have been conducted but that have been *explicitly excluded* from this plan. If a type of test will not be implemented and executed, indicate this in a sentence stating the test will not be implemented or executed and stating the justification, such as:

- “These tests do not help achieve the evaluation mission.”
- “There are insufficient resources to conduct these tests.”
- “These tests are unnecessary due to the testing conducted by xxxx.”

As a heuristic, if you think it would be reasonable for one of your audience members to expect a certain aspect of testing to be included that you will not or cannot address, you should note it’s exclusion: If the team agrees the exclusion is obvious, you probably don’t need to list it.]

## 5. Test Approach

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

n.a.

[Provide a listing of existing resources that will be referenced to stimulate the identification and selection of specific tests to be conducted. An example Test-Ideas Catalog is provided in the examples section of RUP.]

### 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:	Execute each use-case scenario’s individual use-case flows or functions and features, using valid and invalid data, to verify that: <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:	-

Aniflix App	Version: <1.0>
Master Test Plan	Date: <dd/mmm/yy>
<document identifier>	

### 5.2.2 User Interface Testing

Technique Objective:	Exercise the following to observe and log standards conformance and target behavior: <ul style="list-style-type: none"> <li>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).</li> <li>Window objects and characteristics can be exercised—such as menus, size, position, state, and focus.</li> </ul>
Technique:	Feature Files
Oracles:	We assume all tests to pass.
Required Tools:	Flutter Gherkin
Success Criteria:	All tests pass.
Special Considerations:	-

## 6. Entry and Exit Criteria

n.a.

## 7. Deliverables

n.a.

[In this section, list the various artifacts that will be created by the test effort that are useful deliverables to the various stakeholders of the test effort. Don't list all work products; only list those that give direct, tangible benefit to a stakeholder and those by which you want the success of the test effort to be measured.]

### 7.1 Test Evaluation Summaries

n.a.

[Provide a brief outline of both the form and content of the test evaluation summaries, and indicate how frequently they will be produced.]

### 7.2 Reporting on Test Coverage

n.a.

[Provide a brief outline of both the form and content of the reports used to measure the extent of testing, and indicate how frequently they will be produced. Give an indication as to the method and tools used to record, measure, and report on the extent of testing.]

### 7.3 Perceived Quality Reports

n.a.

[Provide a brief outline of both the form and content of the reports used to measure the perceived quality of the product, and indicate how frequently they will be produced. Give an indication about to the method and tools used to record, measure, and report on the perceived product quality. You might include some analysis of Incidents and Change Request over Test Coverage.]

### 7.4 Incident Logs and Change Requests



Aniflix App	Version: <1.0>
Master Test Plan	Date: <dd/mm/yy>
<document identifier>	

n.a.

[Provide a brief outline of both the method and tools used to record, track, and manage test incidents, associated change requests, and their status.]

## 7.5 Smoke Test Suite and Supporting Test Scripts

n.a.

[Provide a brief outline of the test assets that will be delivered to allow ongoing regression testing of subsequent product builds to help detect regressions in the product quality.]

## 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 has to approve the Pull Request.

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

## 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.

[The specific elements of the test system may not be fully understood in early iterations, so expect this section to be completed over time. We recommend that the system simulates the production environment, scaling down the concurrent access and database size, and so forth, if and where appropriate.]

[Note: Add or delete items as appropriate.]

### 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			
Project Management	YouTrack	JetBrains	

Aniflix App	Version: <1.0>
Master Test Plan	Date: <dd/mmm/yy>
<document identifier>	

#### 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.

Human Resources		
Role	Minimum Resources Recommended (number of full-time roles allocated)	Specific Responsibilities or Comments
Test Manager	1	Provides management oversight. Responsibilities include: <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>

Aniflix App	Version: <1.0>
Master Test Plan	Date: <dd/mmm/yy>
<document identifier>	

Human Resources		
Role	Minimum Resources Recommended (number of full-time roles allocated)	Specific Responsibilities or Comments
Test Analyst	1	Identifies and defines the specific tests to be conducted.  Responsibilities include: <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	Defines the technical approach to the implementation of the test effort.  Responsibilities include: <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>
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>

Aniflix App	Version: <1.0>
Master Test Plan	Date: <dd/mmm/yy>
<document identifier>	

Human Resources		
Role	Minimum Resources Recommended (number of full-time roles allocated)	Specific Responsibilities or Comments
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.

*[The way to approach staffing and training will vary from project to project. If this section is part of a Master Test Plan, you should indicate at what points in the project lifecycle different skills and numbers of staff are needed. If this is an Iteration Test Plan, you should focus mainly on where and what training might occur during the Iteration.]*

*Give thought to your training needs, and plan to schedule this based on a Just-In-Time (JIT) approach—there is often a temptation to attend training too far in advance of its usage when the test team has apparent slack. Doing this introduces the risk of the training being forgotten by the time it's needed.*

*Look for opportunities to combine the purchase of productivity tools with training on those tools, and arrange with the vendor to delay delivery of the training until just before you need it. If you have enough headcount, consider having training delivered in a customized manner for you, possibly at your own site.*

*The test team often requires the support and skills of other team members not directly part of the test team. Make sure you arrange in your plan for appropriate availability of System Administrators, Database Administrators, and Developers who are required to enable the test effort.]*

## 11. Iteration Milestones

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

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

Aniflix App	Version: <1.0>
Master Test Plan	Date: <dd/mmm/yy>
<document identifier>	

<b>Risk</b>	<b>Mitigation Strategy</b>	<b>Contingency (Risk is realized)</b>
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>