

# Sprint 1 Testing Document

Testing Document

COMP90082

Planimation

Team: Bluering



SCHOOL OF  
COMPUTING &  
INFORMATION  
SYSTEMS

<students must use this table to track individuals' contributions to this document.

Every time you change this document, add the date you changed it, a description of your performed task and your name. For the version, please adopt the following format:

**01.00-D<number>** for draft versions related to Part 1 (any version before the final submission is considered draft). When your document is reviewed and finally ready to be submitted, change it to 01.00. For Part2, start with 02.00-D<number> and so on. This document should always be kept on GitHub>

## Revision History

Date	Version	Description	Author
5/9/2021	01.00-D01	Initial draft	Priyanka Shivampetha
10/9/2021	01.00-D02	Functional Test Cases For UC001	Priyanka Shivampetha
17/9/2021	01.00-D03	Functional Test Cases For UC005	Zenan
19/9/2021	01.00-D04	Functional Test Cases For UC002	Zenan

## Contents

### 1. Introduction 4

#### 1.1 Proposal 4

#### 1.2 Target Users 4

#### 1.3 Conventions, terms and abbreviations 4

### 1. Covered Requirements 4

#### 2.1 Functional or Product Requirements 4

### 1. Functional Test Cases 5

#### 3.1 UC001: Implementing of frontend page 5

##### 3.1.1 TC001: interface page implementing successfully 5

3.1.2 TC002: VGF upload function 6

3.2 UC003: Animation PDDL, domain and problems upload. 7

3.2.1 TC001: Add domain file, problem file and animation file 7

3.2.2 TC002: Animation Generation by domain and problem files - Unsuccessful 8

3.2.3 TC003: Animation generation by VFG file - Unsuccessful 8

1. Test Cases 8

4.1 TC0<number>: <name> 8

1. Entry Data 9

5.1 DATA: PDDL problem file 9

5.2 DATA: VFG file 9

• Introduction

The system integration test of the Planimation should validate from the requirements perspective that:

- All timekeeping functions work correctly.
- Frontend page work correctly.
- Animation functions work correctly.
- The system is easy to use by the end-users.
- All points of integration within the system work as defined in requirements.
- All points of integration with other systems work as defined in requirements.
- Recovery procedures are correct and can be performed by users.

The objective of this testing is to validate the system operation as a whole. At the conclusion of testing, the project team and the test team will have a high level of confidence that the system will work according to user requirements.

• Proposal

The purpose of this document is to define and present the test cases for project Planimation, covering the test cases for the system use cases.

• Target Users

This document in mainly designed for those responsible for executing the test cases in this project Shujin Zhang, Priyanka Shivampetha, Zenan Ji, Yinghao Wang, SWEN90082 Supervisor Doc Wallace.

• Conventions, terms and abbreviations

This section explains the concept of some important terms that will be used throughout this document. These terms are described in the following table, presented in alphabetical order.

Term	Description
PixiJS	A rendering library that will allow you to create rich, interactive graphics, cross platform applications. Used for the generation of animation for PDDL.
PDDL	The Planning Domain Definition Language (PDDL) is an attempt to standardize Artificial Intelligence (AI) planning languages.

- **Covered Requirements**

This section lists the system requirements covered in the test cases.

- **Functional or Product Requirements**

Requirement Identifier	Requirement Name
UC001	As a user, i should access the main interface when accessing the 4 sub-modules
UC002	As a user, i should upload the vfg file for the generation of the visualization
UC003	For the planning problem As a user, I can able to upload domain problems and animation files in order to generate the solution
UC004	As a user after upload the problem files I can able to see the animation visualization for the problem file
UC005	As a user for each plan i should check every step of plan,if a user selects any particular step then it should show detailed step information

- **Functional Test Cases**

This section describes the test cases that cover the product requirements of the system.

- **UC001:Checking whether User can Successfully Access the homePage**
- TC001: Goes To The Homepage Successfully

<b>Test Type:</b> Functional	<b>Execution Type:</b> Manual
<b>Objective:</b> Verify if the URL of the page is working Successfully	
<b>Setup:</b> <ul style="list-style-type: none"> <li>• Having The internet Access</li> </ul>	
<b>Pre-Conditions:</b> <ul style="list-style-type: none"> <li>• No</li> </ul>	
<b>Notes:</b> Open the webpage <ul style="list-style-type: none"> <li>• Check With Buttons of the page whether they are functioning correctly or not</li> <li>• check forward, backward and reload pages</li> </ul> The interface page can open successfully.	

**Time constraint:**

Minimum: less than 10 sec

Maximum: 1 min

- **UC002: For the planning problem As a user, I can able to upload domain problems and animation files in order to generate the solution**
- TC001: Checking Uploading Problem File

<b>Test Type:</b>	<b>Execution Type:</b>
Functional	Manual
<b>Objective:</b> Verify that the uploading function is performed successfully.	
<b>Setup:</b> <ul style="list-style-type: none"><li>• Need to have a successful Internet Connection</li></ul>	
<b>Pre-Conditions:</b> <ul style="list-style-type: none"><li>• No</li></ul>	
<b>Notes:</b> [1] Try to upload the problem file in the web page. [2] Try to upload the problem to the web page after refreshing the page. [3] Try to upload the different problem file in the web page. [ <i>* Server will list all files uploaded into the server.</i> <i>* All types of files are accepted by the server.</i>	
<b>Time constraint:</b> Minimum: Less than 2min Maximum: more than 3min	

- **UC002 For the planning problem As a user, I can able to upload domain problems and animation files in order to generate the solution**
- TC003: Checking With Domain File

<b>Test Type:</b>	<b>Execution Type:</b>
Functional	Manual
<b>Objective:</b> Verify if a domain file can be added successfully	
<b>Setup:</b> <List the pre-conditions to carry through this test case >	

**Pre-Conditions:**

<List the conditions after the test execution >

**Notes:**

- [1] Try to upload domain files on the web page.
- [2] Try to upload domain files to the web page after refresh the page.
- [3] Try to upload the different problem files on the web page

\* Server will list all files uploaded into the server.

\* All types of files are accepted by the server.

\* Server can receive data from the file.

**Time constraint:**

Minimum: less than 2 min

Maximum: more than 3 min

- UC002: For the planning problem As a user, I can able to upload domain problems and animation files in order to generate the solution
- TC004: Checking Uploading Animation File

<b>Test Type:</b>	<b>Execution Type:</b>
Functional	Manual
<b>Objective:</b>	
Verify that the uploading function is performed successfully.	
<b>Setup:</b>	
<ul style="list-style-type: none"><li>• Need to have a successful Internet Connection</li></ul>	
<b>Pre-Conditions:</b>	
<ul style="list-style-type: none"><li>• No</li></ul>	
<b>Notes:</b>	
[1] Try to upload the Animation file on the web page.	
[2] Try to upload the animation file to the web page after refreshing the page.	
[3] Try to upload the different Animation files on the web page.	
[	
* Server will list all files uploaded into the server.	
* All types of files are accepted by the server.	

**Time constraint:**

Minimum: Less than 2min

Maximum: more than 3min

- **UC002: For the planning problem As a user, I can able to upload domain problems and animation files in order to generate the solution**
- TC005: Checking Uploading Problem File Is Uploaded not successfully

<b>Test Type:</b>	<b>Execution Type:</b>
Functional	Manual
<b>Objective:</b> Verify that the uploading function is performed un successfully.	
<b>Setup:</b> <ul style="list-style-type: none"><li>• No</li></ul>	
<b>Pre-Conditions:</b> <ul style="list-style-type: none"><li>• No</li></ul>	
<b>Notes:</b> 1) user trying the problem file to upload when there is no internet 2) user trying to upload the empty files	
<b>Time constraint:</b> Minimum: Less than 2min Maximum: more than 3min	

- **UC002: For the planning problem As a user, I can able to upload domain problems and animation files in order to generate the solution**
- TC006: Checking Uploading Domain File Unsuccessful

<b>Test Type:</b>	<b>Execution Type:</b>
Functional	Manual
<b>Objective:</b> Verify that the uploading function is performed not successfully.	
<b>Setup:</b> <ul style="list-style-type: none"><li>• no</li></ul>	
<b>Pre-Conditions:</b> <ul style="list-style-type: none"><li>• No</li></ul>	

**Notes:**

- 1) user trying the Domain file to upload when there is no internet
- 2) user trying to upload the empty Domain files

**Time constraint:**

Minimum: Less than 2min

Maximum: more than 3min

- UC002: For the planning problem As a user, I can able to upload domain problems and animation files in order to generate the solution
- TC006: Checking Uploading Animation File Unsuccessful

<b>Test Type:</b> Functional	<b>Execution Type:</b> Manual
<b>Objective:</b> Verify that the uploading animation File function is performed not successfully.	
<b>Setup:</b> <ul style="list-style-type: none"><li>• no</li></ul>	
<b>Pre-Conditions:</b> <ul style="list-style-type: none"><li>• No</li></ul>	
<b>Notes:</b>  1) user trying the Animation file to upload when there is no internet 2) user trying to upload the empty Animation files	
<b>Time constraint:</b>  Minimum: Less than 2min Maximum: more than 3min	