**Unity QA Task**

horizontal line

# 

I was given a task to finish within a week.

This is the documented report for the task to record the process and thoughts.

The task is basically about using Unity’s test runner to run some tests on one of their 2D features, and then upload the test result to somewhere to be observe later.

## Step 1

1) Download latest Unity Version available.

The simplest request in this task. Nothing to talk about.

## Step 2

2)Pick a 2D feature of your choice

a)Preferably something that can be tested by functional test (and not graphics tests)

I went through some 2D features and see which one to pick from.

But in the end, I picked texture2D because of being more familiar with it and don’t have to result in doing even more research into the functionality. There’s enough work ahead and limited time.

## Step 3

3) Using Unity's Test Runner write 10 tests for that feature

a) Pick tests that you think would be crucial for that feature

Firstly, I researched and looked into how to set up Test Runner and how to use it.

Then, I set up the edit mode & play mode test folders and write 10 tests for texture2D.

\*During test run, some tests failed when Run All, but will not have issue when running Individually.  
\*\* Seems like the data modified will bring forward to the next tests and affect the test result.

So, I rewrote some tests to use copied/dummy texture instead of from the source texture directly. This way, when the texture is changed, it only affects the dummy texture and not the source texture.

## Step 4

4) Create another small testing scene for that feature but with the focus for manual testing

a) Something that when you run it, you know that the basics are working well for that feature

This is another task that I can do within my capability without needing to do too much research. All I need to do is plan and design the test.

I created some game objects and wrote some scripts for the feature.

## Step 5

5) For automated test submit results somewhere automatically to see the results

a) Could be Google Sheet, Data Studio, any other

b) Something very basic: Test Name and pass/fail condition, all in one place

c) BONUS: Ability to compare results for each run

First, I planned ahead of what I should do for this step. I decided to go with Google Sheets because I have experience using it. As for Data Studio, I have no experience using it, and I have no idea what data to present aside from the total numbers of passing/failing tests.

I also decide against doing the Bonus task because I have to spend time to research and test out google spreadsheets API, and I can’t plan or predict ahead for this Bonus task without knowing how the API works first.

To start off the tasks, I researched about Google Spreadsheets API.

\*During the process, I found a "Google Sheets To Unity" tool in Unity Asset Store, but decided to not use it, because it's more interesting to learn and do it myself. That way I will learn new knowledge too. Not to mention the flexibility to customize the program to work how I wanted it to.

After that, I started setting up Google Cloud Platform API credentials, OAuth, scopes, etc, and hooking ITestRunCallback to Test Runner.

Then, I researched & tested out Google Sheets API to write data on spreadsheets.

Lastly, I write the codes to get the test results and automatically update the test results to the specific google sheet.

## Step 6

6) Submit your project to github repository

This is the first time I submitted a project to github despite using it for downloading others’ projects before. It’s truly an exciting experience.

Before I submit the project, I did a final test to make sure everything previously implemented above is working fine.

### Conclusion

This is an interesting task to challenge my limit & boundary due to the extra knowledge I needed to research before starting. It also allowed me to have some first time experiences on doing Automation testing.