Unity Test Framework: How-To Manual

By Garrett Wells, Kristie Olds, Nathaniel Palmer, Jeremy Wisecarver, Scott Martin

Contents

What is Unity Test Framework?

Background

The Unity Test Framework allows users to conduct tests on your game code through Unity. This is unique in its capability to create automated tests and be run in both edit mode and play mode. The Unity Test Framework is an integration of the NUnit library, an open-source unit testing library. NUnit supports testing of .Net languages and was ported from JUnit.

Edit and Playmode Tests

So what is the difference between play mode and edit mode tests? As the names imply, they are best suited to different cases where the code is used. **Edit mode** tests are best used for testing code that can be called independent of user input. For example, this test type would be used to test the output of a function against various automated inputs.

Play mode tests are most effectively used to test features that depend on inputs or use cases that would only apply while playing a game. An example of this is player movement or animation calls. These are features where the game engine itself is handling function calls in a way that cannot be seen or tested by our automated edit mode tests.

How To: Install Unity Test Framework

Installing the Unity Test Framework is simple and requires only the steps below to install.

Pre-requisites

1. Installed Unity Editor version 2019.2 or later.

Walkthrough

1. Open an existing project in the Unity Editor or create a new project.
2. Select **Window → Package Manager**. This will open the package manager where you can manage installed packages and install new ones.
3. From the second drop down menu at the top of the window select **Unity Registry**. This will show a list of all Unity packages that can be installed.
4. Search for, or scroll down to, **Test Framework**.

5. Click on **Test Framework**.
6. If the package has not been installed, there will be an option in the bottom right of the window to install. If the package has already been installed there will either be an option to upgrade the package or remove it.

How To: Create New Boundary Test Case

Definition

A boundary test case explores the variation of inputs to a function and it's behavior. This means testing the behavior with acceptable inputs, barely acceptable inputs, and unacceptable inputs. For example, when testing the behavior of an inventory system, we would create a boundary test case by testing the behavior when adding a new item to a full inventory as well as removing an item from an empty inventory.

In order to set up automated tests to do this we will need to take a closer look at how unit tests are created in Unity. Unit tests are run through scripts that inherit from the NUnit library and Unity's own TestRunner libraries. The unit test system differs from regular scripts in that test scripts are grouped into **assemblies.** Assemblies define scripts that reference the same libraries and should be compiled together. This prevents the system from needing to recompile all scripts in a project for every change saving time.

To create unit tests we need to:

1) Define an assembly for our test scripts.
2) Define the test functions.

Walkthrough

1. Open **Window → Test Runner**
    a. This will open the test runner window in the editor. This window will allow the user to manage unit tests, run tests, and view test output.
2. Navigate to the folder you want to store your tests in and **right click → Create → Testing →**
    a. The important part about naming these folders is to keep the Edit mode unit tests separate from the play mode tests because they have different dependencies. More specifically, edit mode tests need to be compiled with the UnityEditor.TestRunner dependency.
3. Since we are focussing on testing the boundaries of our scripts, open **EditMode_Tests** and then **right click → Create → Testing → Tests Assembly Folder**
4. This folder will contain our edit mode tests so rename it to **EditMode_Tests** or something similar.

5. Open **EditMode_Tests** and **right click → Create → Testing → C# Test Script**.

How To: Create New Stress Tests

A test that tries to do something for x quantity, then double and repeat test. Repeat this sequence until you see something break.

How To: Run Tests

Open the Test runner window in the unity editor by going to Window → General → Test Runner.

Go to either PlayMode or EditMode and press “Run All” and all of your unity tests will run.