Version 1.0.1
showumywork@gmail.com

Overview

ProjectileManager is a plugin that enables you, the Unity developer, to shoot a projectile to a location of your input(mouse or touch). It's **fast, flexible, mobile compatible and easy** to use. It's great for assisting player to visualize where a projectile is going to hit.

Quick Start

Projectile Manager comes with a demo scene to help you understand the functionality of plugin.
Start by creating a new project and import all Projectile Manager package assets included sample scene file.

Step 1: Open the SampleScene scene located in Assets/ProjectileManager/Scenes/SampleScene

Step 2: Play in editor, click a position where you want to launch a projectile. while holding a click, a trajectory to show a path which a projectile will follow is drawn.

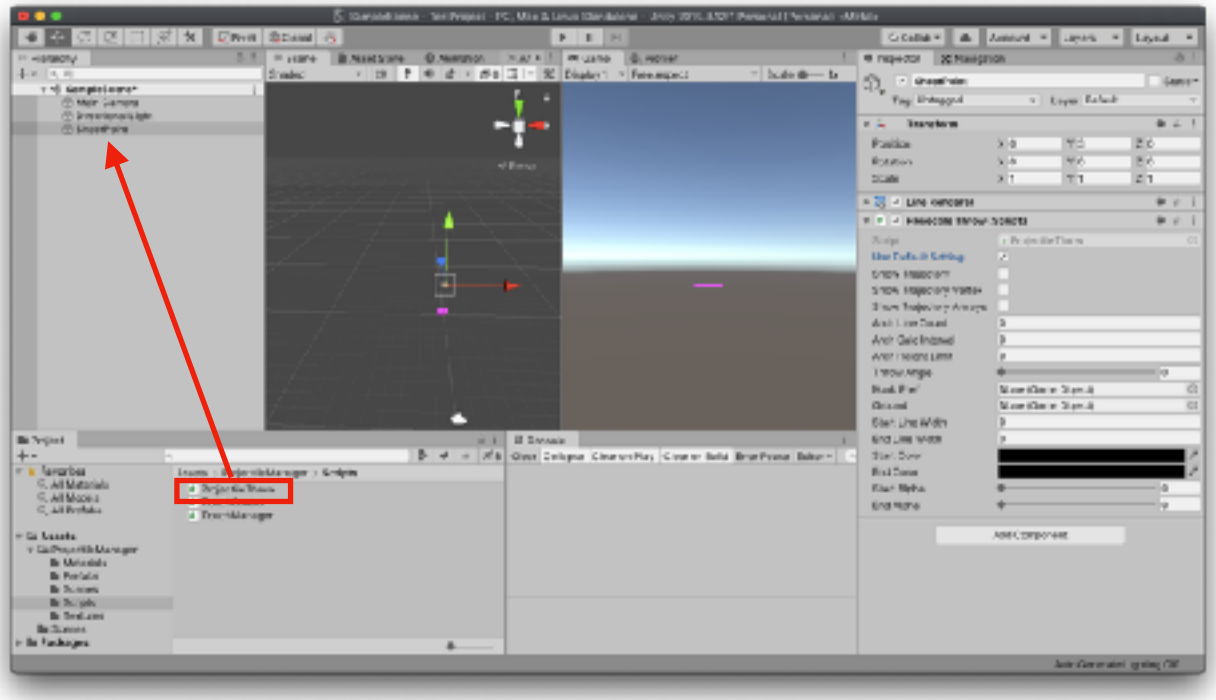
Core Module

ProjectileThrow.cs - This core module used to calculate a force to add to a projectile and visualize a parabolic trajectory which a projectile follows.

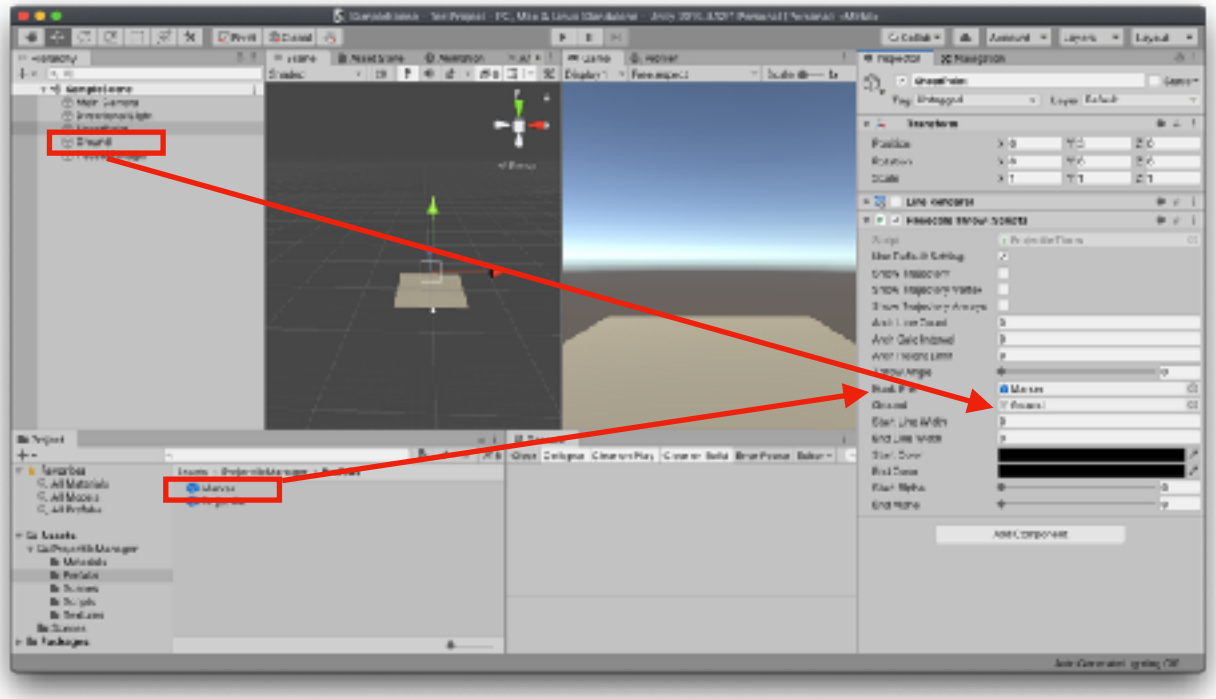
TouchManager.cs - This module used to obtain a position of a world space and state of input according to user's input.

Integration Guide

Step 1: Attach the **ProjectileThrow.cs** script to an empty game object in your scene. In this example, we named the empty game object **"ShootPoint."** The script contains functions calculating a force to add to a projectile, drawing a path of the projectile and shooting an object. Please be noted that the game object attached with the script becomes a start point of shooting.
Check the **UseDefaultSetting** variable if you would like to use the default setting.

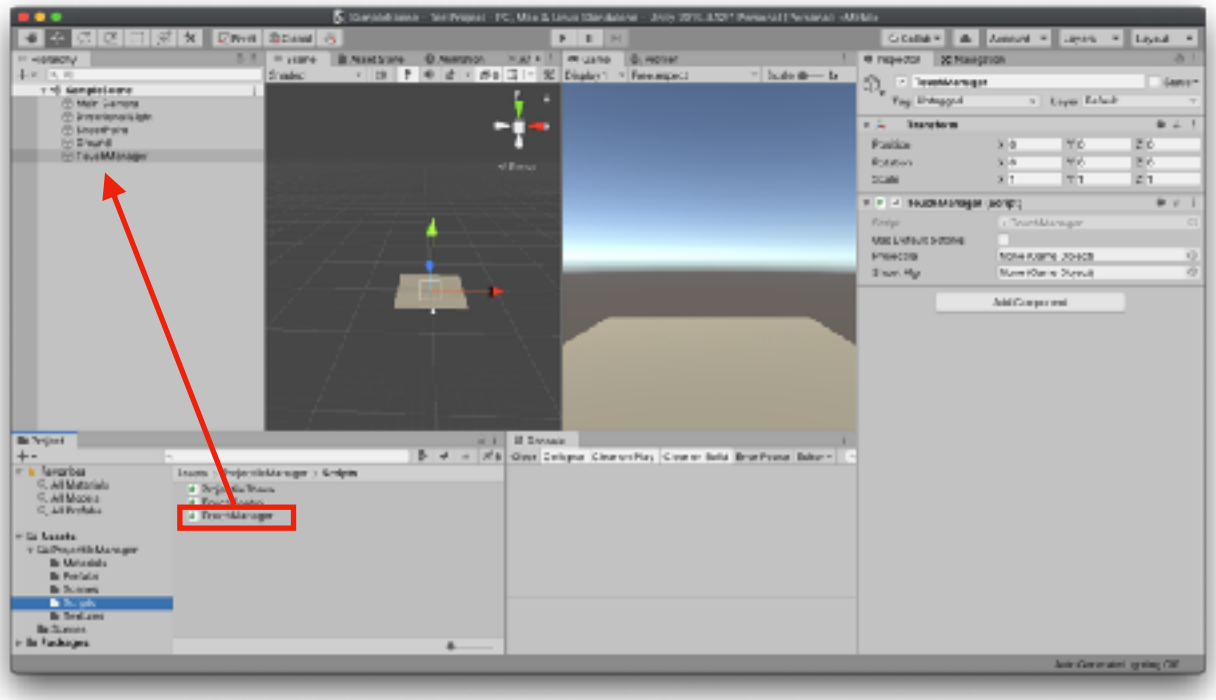


Step 2: Create a plane as a ground object and drag it into ProjectileThrow property and a Marker prefab(Assets/ProjectileManager/Prefabs/Marker) into the MarkPref property. In the example, we named the plane **"Ground." ProjectileThrow.cs** requires ground(any kind of collider) to shoot objects.

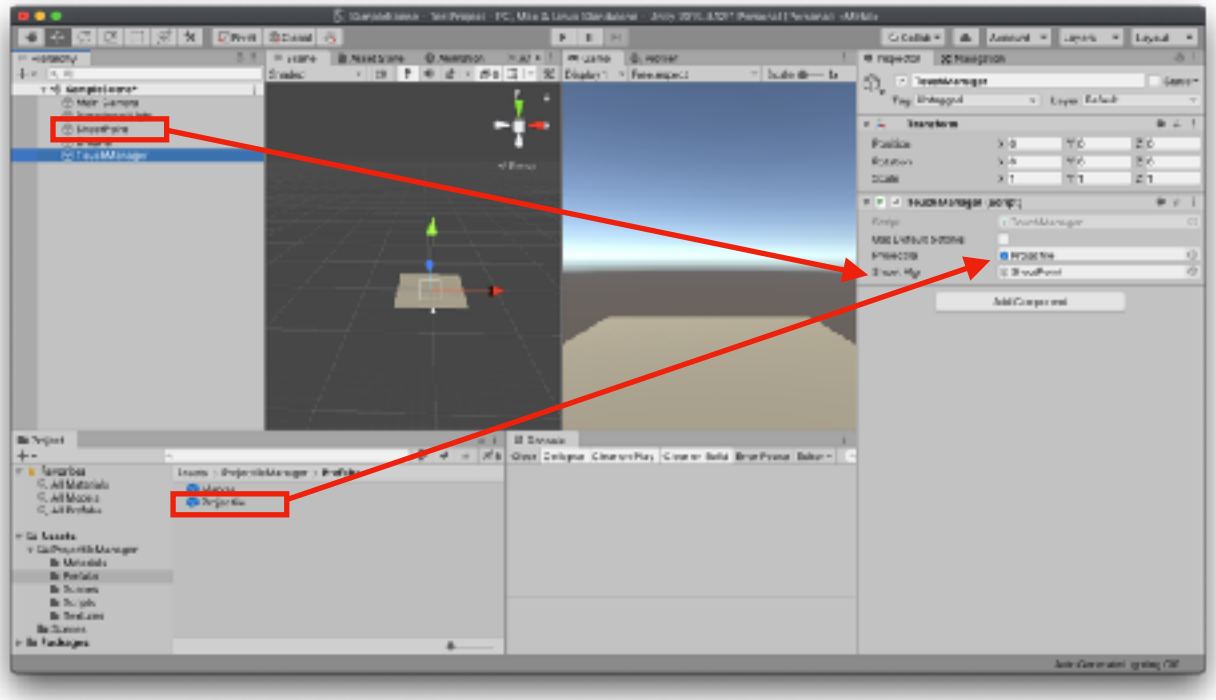


Step 3: Attach the **TouchManager.cs** to an empty game object in your scene, we named it **“TouchManager”**. This script will obtain information of user's input and a world position according to user's input and give the information to **ProjectileThrow.cs.**

MarkPref - used to visualize vertices of trajectory

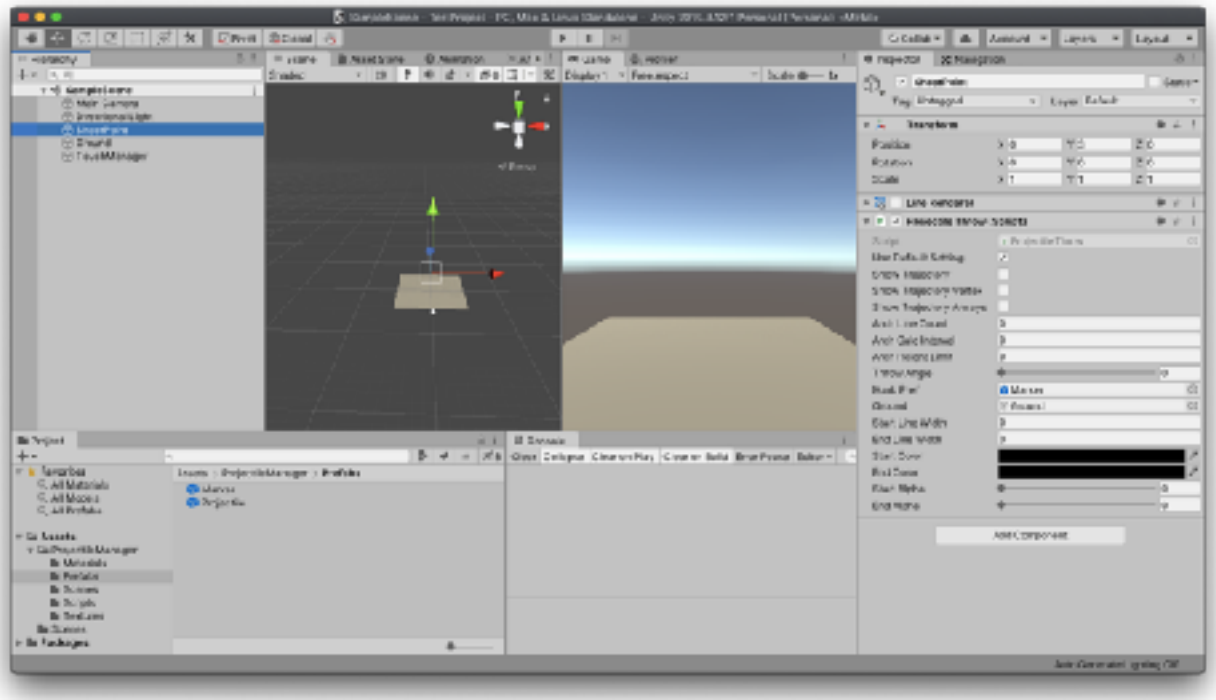


Step 4: Drag the game object containing the ProjectileThrow (in example **“Shoot Point”**) into the **TouchManager.cs** property **(ShootMgr)** and a **Projectile prefab**(Assets/ProjectileManager/Prefabs/Projectile) into the **TouchManager.cs** property. You can use your own prefab if you want.



Projectile - a projectile to be launched

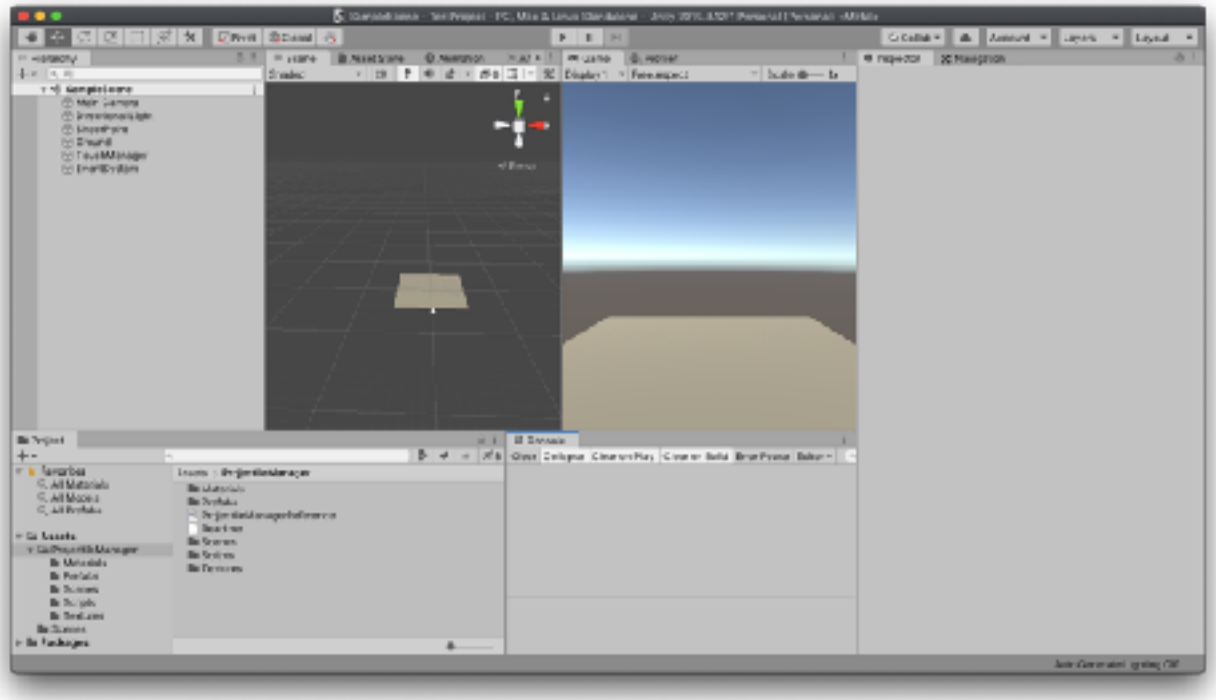
Step 5: Set a value of each variables. Or you can use default setting simply check the useDefaultSetting variable.



ShowTrajectory - when true, shows a trajectory of a launching object
ShowTrajectoryVertices - when true, shows a vertices of trajectory line
ShowTrajectoryAlways - when true, a drawn trajectory is kept after a projectile is launched
ArchLineCount - number of vertices to be rendered to show a trajectory of a projectile
ArchCalcInterval - interval of vertices in trajectory line
ArchHeightLimit - limit which a trajectory line stops to render, be noted if your y axis position of a shoot point is below this value, a trajectory line is never going to be rendered
Throw angle - Euler angle
StartLineWidth - start width of a trajectory line
EndLineWidth - end width of a trajectory line

Step 6: Play the scene! If you encountered an error in **TouchControl.cs**, add an EventSystem.

Final look of the scene



Feedback

If you have any feedback to Projectile Manager plugin, please email us directly, your suggestion will be very valuable to us. If you fail to integrate a plugin into your game, please contact us by showumywork@gmail.com and we will provide more help to let you share your awesome game more efficient.