



Classify is a 2d hyper casual game aims to teach you the basic development concepts in unity, the gameplay of the game is so simple, the player will have different balls generated randomly in different colors and he need to switch the gate that is corresponding to the ball color.

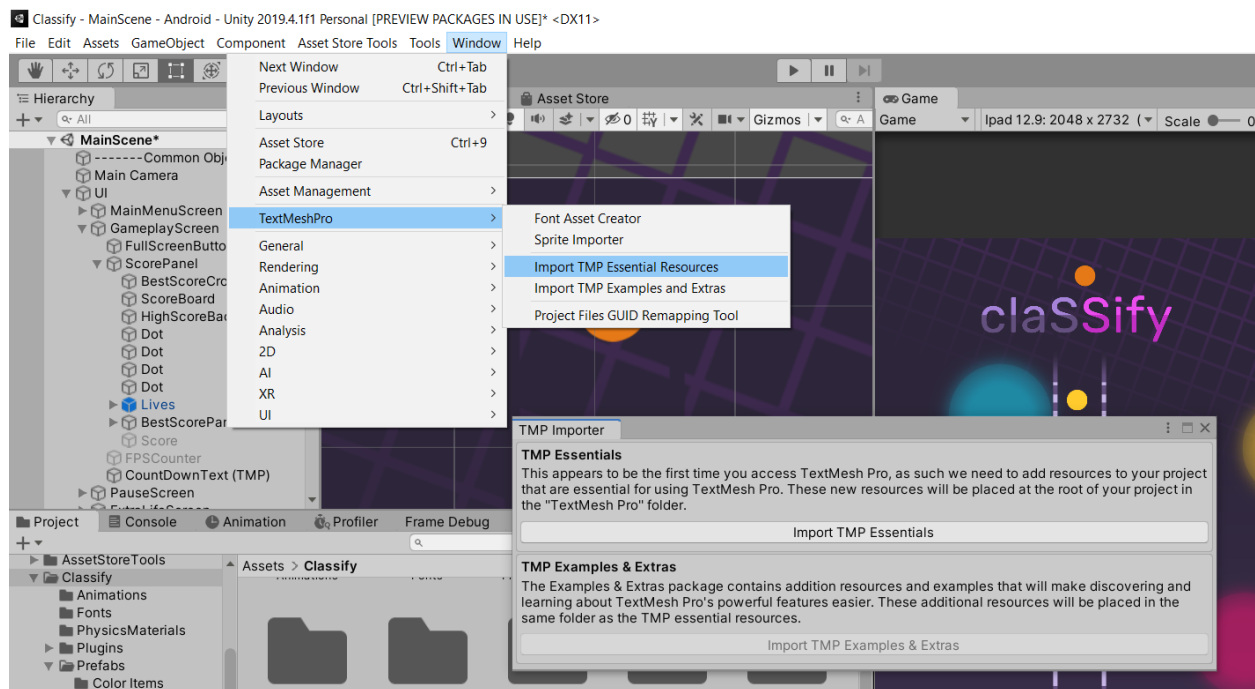
Try the game here:

Android: <https://play.google.com/store/apps/details?id=com.PerfectPixel.classify>

IOS: <https://apps.apple.com/app/id1555658108>

Getting Started:

The first step that you need to do after importing the asset is to import Text Mesh Pro essential assets otherwise game texts won't show correctly, you can do that by pressing Import TMP Essentials button from the TMP Importer Window that should popup automatically or got to Window > TextMeshPro > Import TMP Essential Resources



The whole game is in one scene called “Main Scene”, you can find it under the Scenes folder and you can start exploring everything from there.

Project Folder Structure:

Game assets are categorized in folders based on asset types and not grouped based on the modules:

Animations: contains all the animations and animators assets which are very few because the majority of the game animation has been created using DoTween

Fonts: Contains font files and TextMeshPro font assets

PhysicsMaterials: Contains physics materials

Prefabs: contains all game prefabs

Scenes: contain the mainScene of the project

Scriptables: Contains all the data and event scriptables, currently we don't have any events scriptables because we use Signals library instead

Scripts: Contains all scripts files of the project

Sounds: Contains sound files

Sprites: Contains all game sprites, UI sprites and also the atlases

Assets and Libraries:

[DoTween:](#)

This asset is used for animating all the UI elements of the game as it is more efficient to use tweening instead of animators for animating UI elements (see this talk for more details)

As a usage example of this see `MainMenuScreen.ShowTweening` function

[Signals:](#)

A messaging library used to fire events and send data safely between objects here how we use it in the Switch class when specific gate open:

```
Signals.Get<SwitchOpenGate>().Dispatch(gateIndex);
```

And here how we listen to this message from the `ColorsGameManager` `OnEnable` function:

```
Signals.Get<SwitchOpenGate>().AddListener(OnGateSwitched);
```

Unity Ads Integration:

The project uses unity ads service and to link your own unity ads account edit “[AdsManager](#)” script to replace the current gameId for IOS and Android to your gameId, you can get the gameId from unity Ads dashboard

```
public class AdsManager : IUnityAdsListener
{
    #if UNITY_IOS
        private string gameId = "3988742";
    #elif UNITY_ANDROID
        private string gameId = "3988743";
    #endif

    const string VIDEO_PLACEMENT_ID = "video";
    const string REWARDED_VIDEO_PLACEMENT_ID = "rewardedVideo";

    private AdsManager()
    {
        Advertisement.AddListener(this);
    }
}
```

For any questions please don't hesitate to send me an email: Abdulla.aldandarawy@gmail.com