Hello. Thank you for buying my asset.

***How to setup?***

1. Before building this game to your phone you need to download the latest version of Admob plugin for Unity from there: <https://github.com/googleads/googleads-mobile-unity/releases/tag/v5.4.0>
2. After downloading complete import **GoogleMobileAds.unitypackage** to project. And finish steps from guide here:

<https://developers.google.com/admob/unity/start>

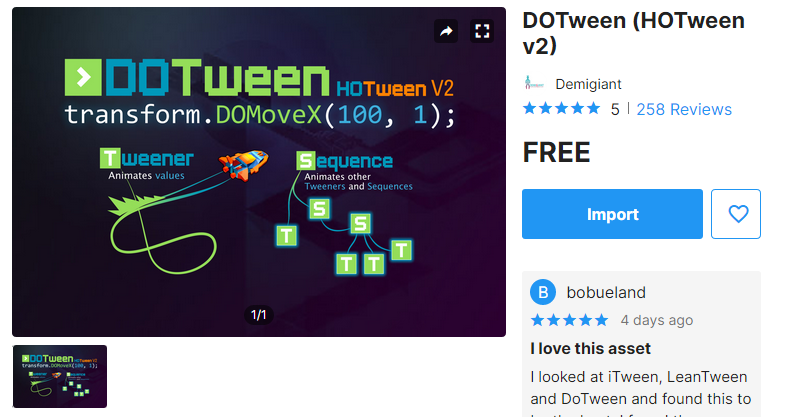
1. In the Unity editor click **for Android**:

Menu→Assets→External Dependency Manager→Android Resolver→Force Resolve.

**For iOS:**

Menu→Assets→External Dependency Manager→ iOS Resolver → Install Cocoapods

1. Install free **DOTween (HOTween v2)** plugin from Unity Asset Store for animations.

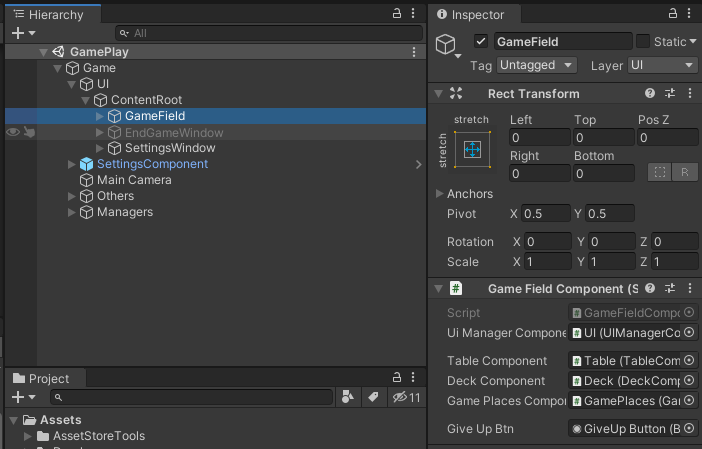


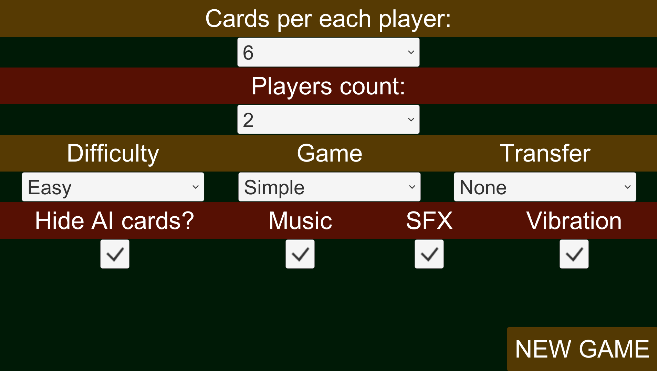
1. Install free **JSON .NET For Unity** plugin from Unity Asset Store for serialize settings.



1. Success. Now you can use asset.

1) The entry point of this game is:



*Settings* window contains 9 type of parameters:  


***Cards per each player*** (2..6) – How much cards user can hold in session

***Players count*** (2..6) – How much users will be play in session (include YOU)

***Difficulty*** – The power of AI. Easy and Hard.

***Game*** – The game type/mode. Simple and Throw-in.

***Transfer*** – Will be available in the next releases.

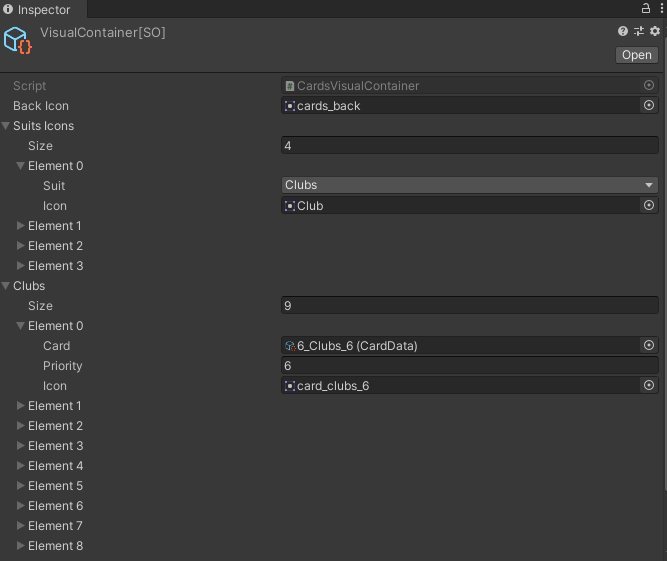
***Hide AI cards*** – Show/Hide cards in game window for AI players.

***Music*** – On/Off background music. (State is saved in persistent data)

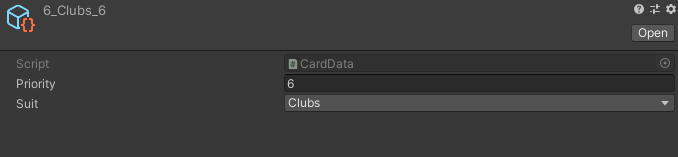
***SFX*** – On/Off SFX music. (State is saved in persistent data)

***Vibration*** – On/Off vibration. (State is saved in persistent data)

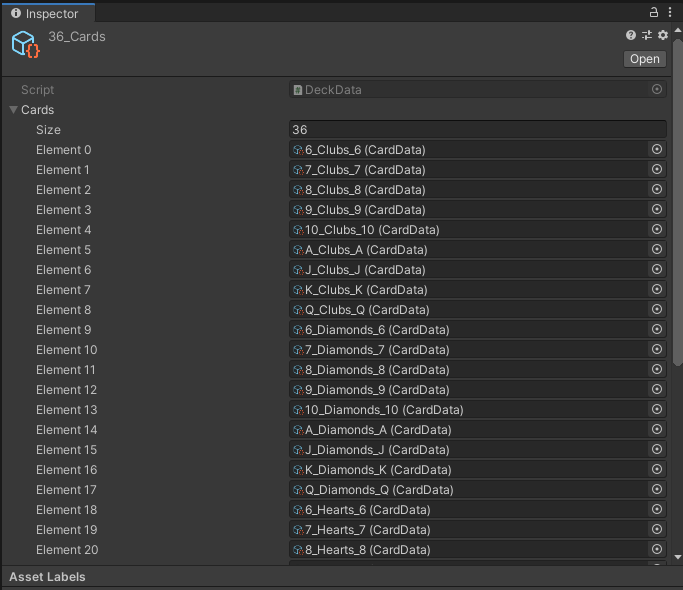
***VisualContainer[SO]*** contains data and links to art and CardData scriptable object. In future releases I will add way to change back and face art of card.



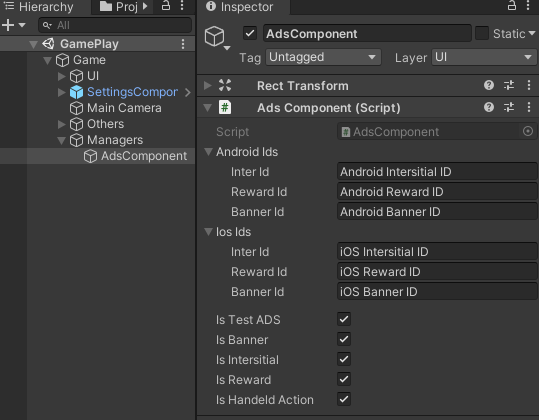
Each card is scriptable object that contains Suit and Priority.



***DeckData*** has links to all cards in Deck.



To setup ***Admob*** settings you should fill AdsComponent:



And **GoogleMobileAdsSettings.asset**

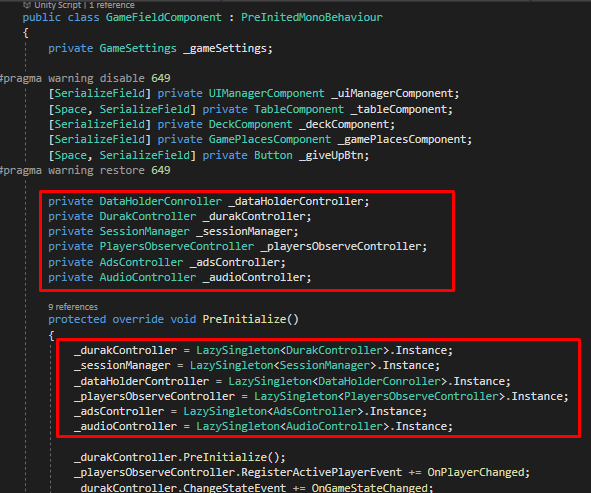
For activating test ADS you should set **IsTestADS** variable to **TRUE** before play (like on screen). For deactivating test ADS you should set **IsTestADS** variable to **FALSE** before play.

This asset represent the correct way of development game on Unity and OOP. So, the controllers responsible for logic of game, components and views responsible for visual and provides data for controllers. If logic is simple and very local you can write it in component. But for large logics you should create controller and write logic there.

To simplify access to controllers and there is no need to create new instances of controllers.

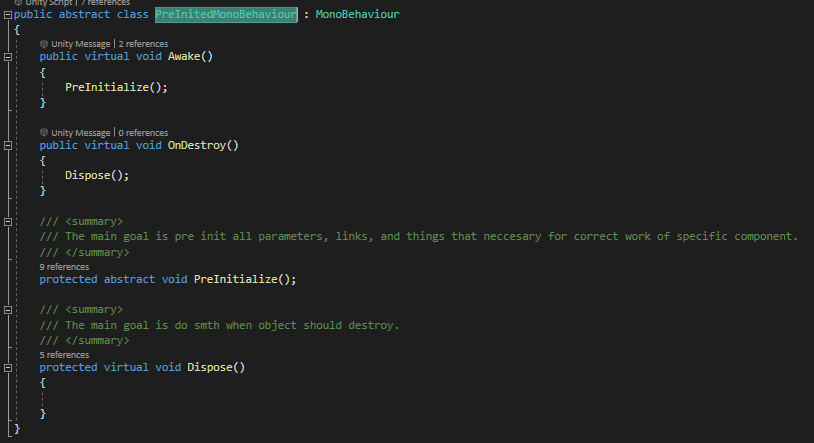
I commented each method so you can easy understand how it works.

I provide in this asset LazySingleton example:



This way will get rid of multiple links in inspector and streamline process of accessing to controllers in game components. So if you has MonoBehaviour component on scene and you need to get access to this component in other MonoBehaviour you should link them via field in inspector. If you need link to controller you should use the example above.

I provide abstract component **PreInitedMonoBehaviour** that will be initialized in Awake function (This need to correctly provides all links to controllers for MonoComponents, like on screen above).



For other questions write me an email: *support@selvassets.ltd.ua*

Thank you for advance!