

Codeer Studio

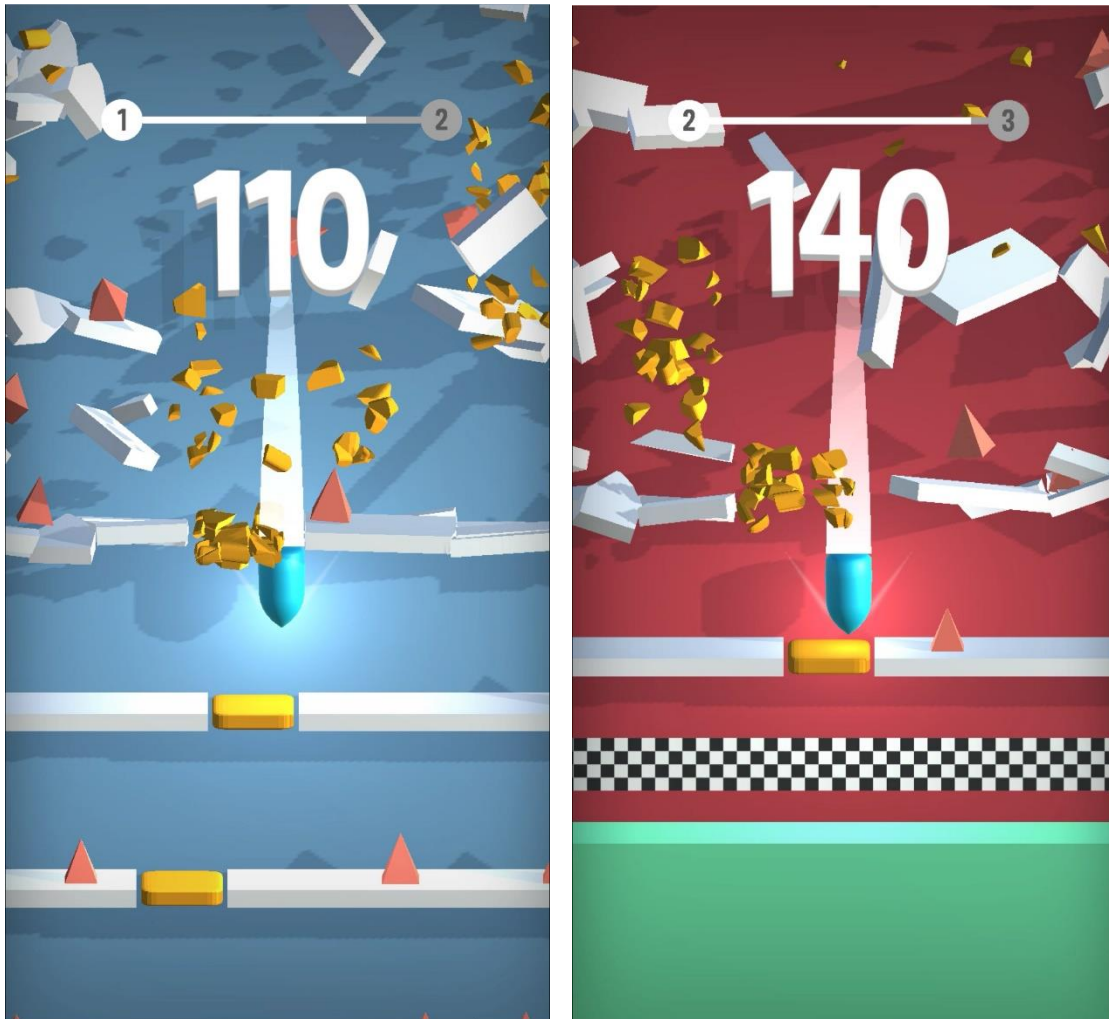
BREAK IT

Unity mobile game template

Thank you for purchasing Break It

Introduction

Break It is a fast paced, casual mobile game. The unique, one tap gameplay makes for a simple yet fun experience. The goal is to get your jumping cube character to the bottom of the level, by breaking through orange gates. Players can try their best to move quickly and earn the rocket powerup, which will boost through the platforms automatically. Players lose when the cube doesn't jump high enough before trying to break through a gate, or when the character hits a white platform.



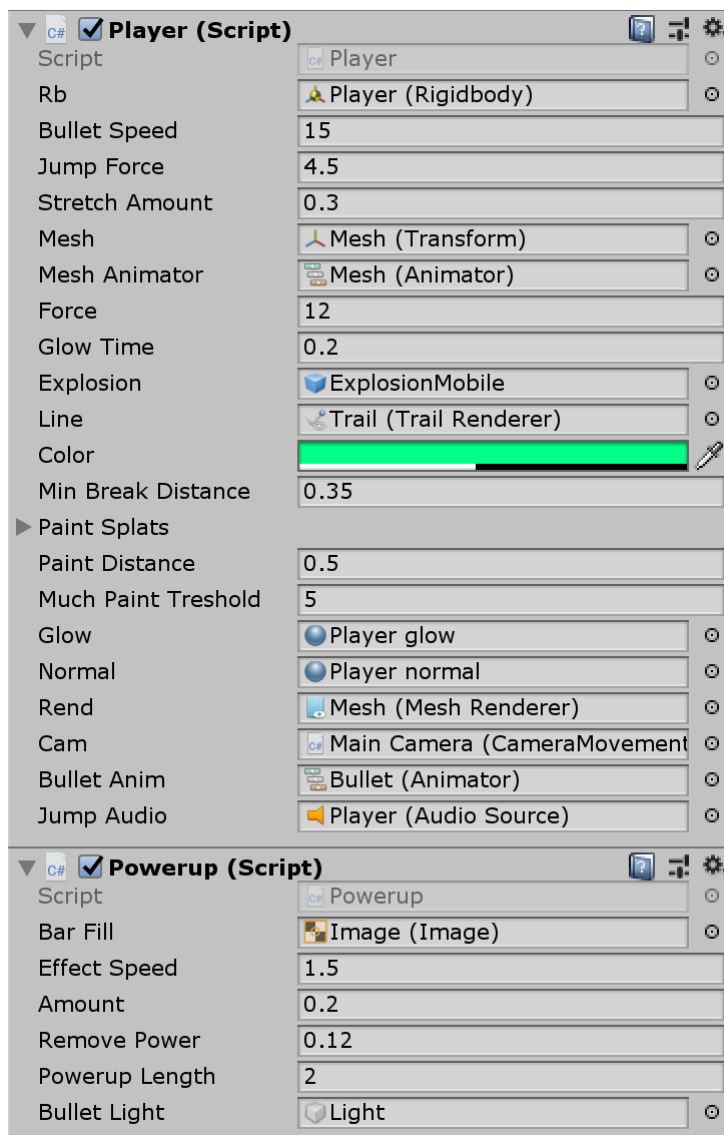
Rocket boost

Main Settings

There's three main objects when it comes to the main game settings; the Player, Game Manager and Generator. Here's a short explanation as to what each of these does.

Player

The player is the tiny coloured cube that needs to jump down to win the game. Please find the player object and see the inspector:



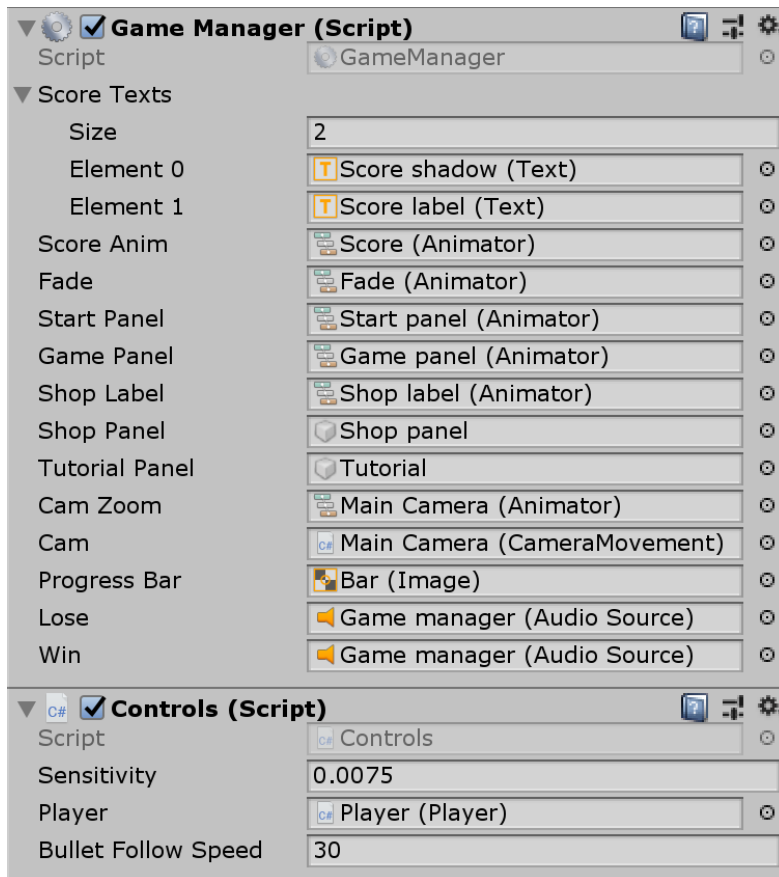
- Bullet speed: how fast does the bullet travel
- Jump force: how high does the player jump each time
- Stretch amount: how much does the player stretch when jumping (nice visual effect)
- Force: the force used to break through platforms
- Glow time: for how long does the player turn white when breaking platforms
- Colour: default colour, normally controlled by the player colour shop rather than the player script
- Min break distance: minimum height above platform to be able to break through it
- Paint splats: paint effect prefabs
- Paint distance: check range for paint splats
- Much paint threshold: jump force for the big paint splat

Powerup settings:

- Effect speed: smooth powerup indicator effect in the score text
- Amount: amount of power that is added each time the player breaks through a platform
- Remove power: how fast does the power drain
- Powerup length: bullet effect duration

Game manager

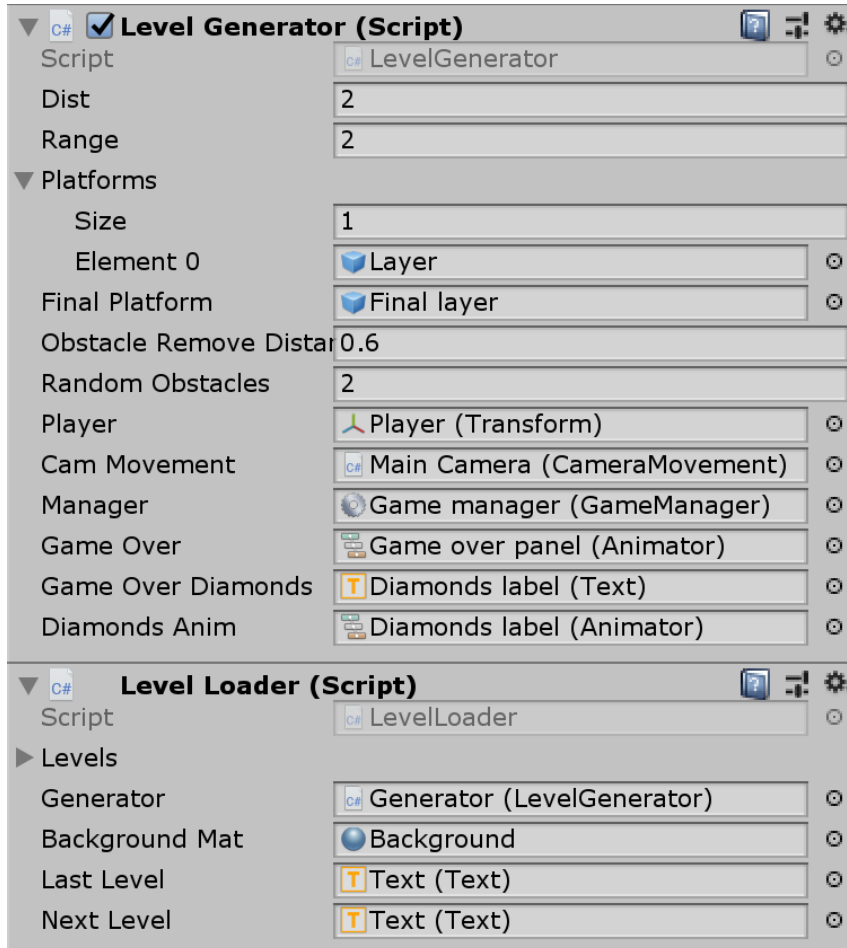
The game manager is mainly responsible for the game loop, user interfaces, etc. The main settings can be found in the 'controls' script attached below the game manager script itself:



- Sensitivity: drag sensitivity when aiming for the orange platforms
- Bullet follow speed: when the bullet powerup is activated, normally it would go straight down. However, since the platform centres move left and right (they don't stay in the centre the whole time) this causes an issue where after the powerup, the player cannot reach the next orange gate anymore. Therefore, while boosting down with the bullet, the player will follow the gates and make sure to stay centred according to the gate positions. The bullet follow speed determines how strongly the player character will try to line up with the gate positions.

Generator

The generator consists of two main components; the level generator, and the level loader:



The level generator is responsible for actually spawning the level and managing the layers & obstacles, where the level loaded simply sets the background colour, and passes the level length to the generator.

Level Generator:

- Dist: distance between platforms on the y axis
- Range: x range for platforms compared to the previous platform
- Obstacle remove distance: when the player breaks through a gate, we need to make sure there's no spikes directly underneath (that wouldn't be fair) so this is the minimum distance on the x axis between the player and the spikes
- Random obstacles: obstacle chance

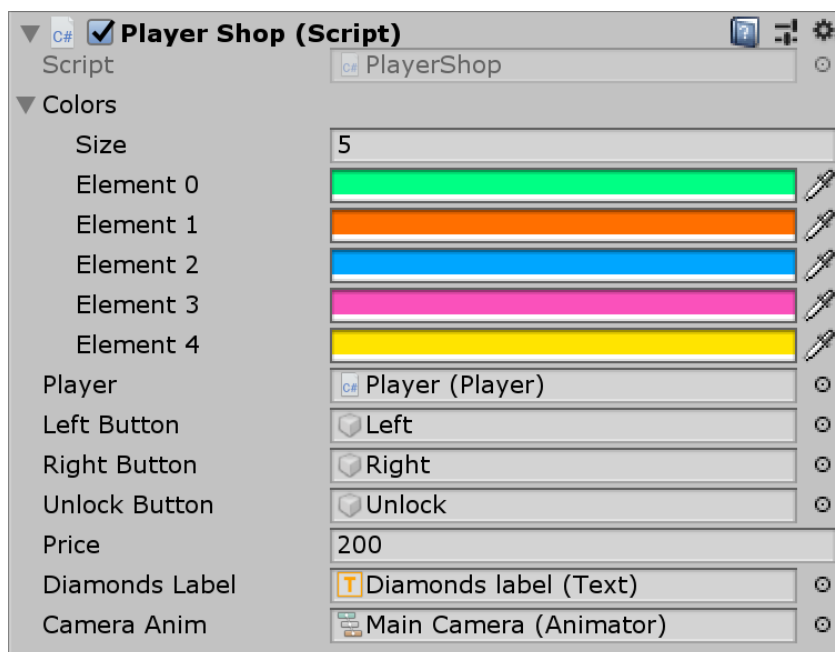
Level loader:

- Levels: list of levels, consisting of a colour and length for each level.

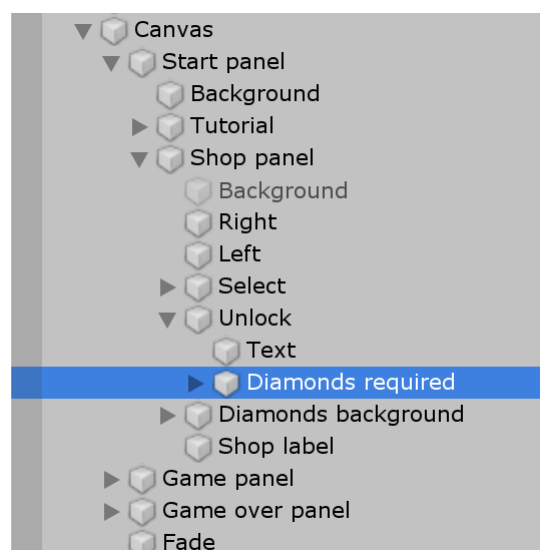
Player shop

The player shop provides different unlockable colours for the player character to unlock using diamonds. Diamonds can be collected by finishing levels and normally it will take about two levels to purchase one new player colour.

Main settings:



Please simply add colours to the colours list to make them unlockable for your players. You can also change the price per colour, just make sure to adjust the price label too:



Unity Ads

To use Unity Ads in your game, please use these steps:

- First, switch the build platform to Android/iOS
- Enable Unity Ads in the services window, and create a new game id to setup the ads
- Please open the dashboard, and in the new project, add a new placement type called '**banner**'. Now you should have these three placement types:



video ●
Video, Playable

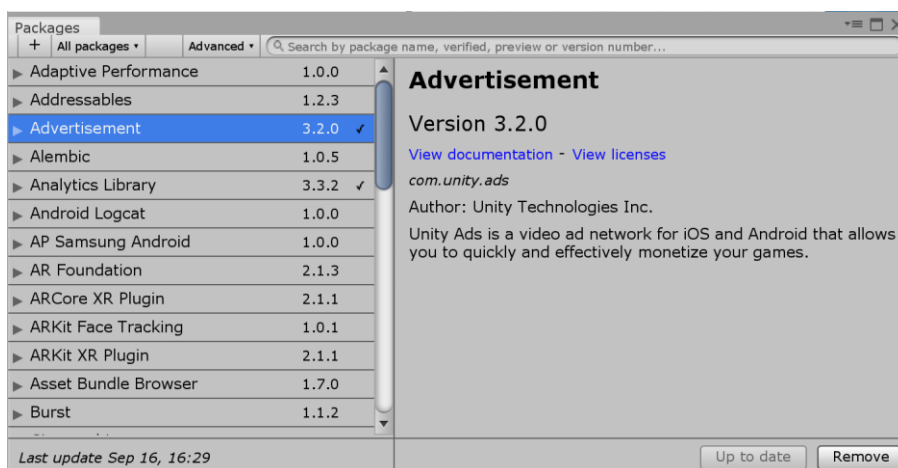


rewardedVideo ●
Video, Playable

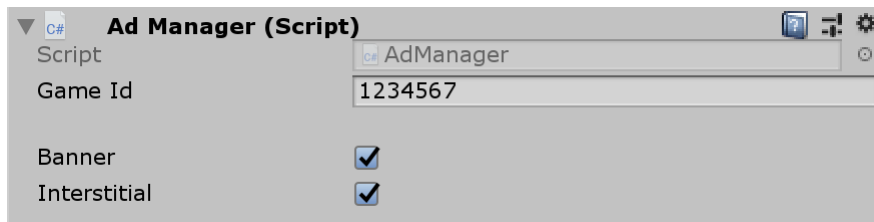


banner ●
Banner

- In the package manager window, please make sure Advertisements are enabled, and also make sure you have at least version 3.0 installed:



- Now, in Scripts -> Ads, there's two scripts; AdManager and AdManagerRewarded. If you just want to enable the interstitial ads, please open the game scene and find the Unity Ads object.



- Please find your game id in the ads dashboard and input it into the GameId field. Then you can choose which types to use.

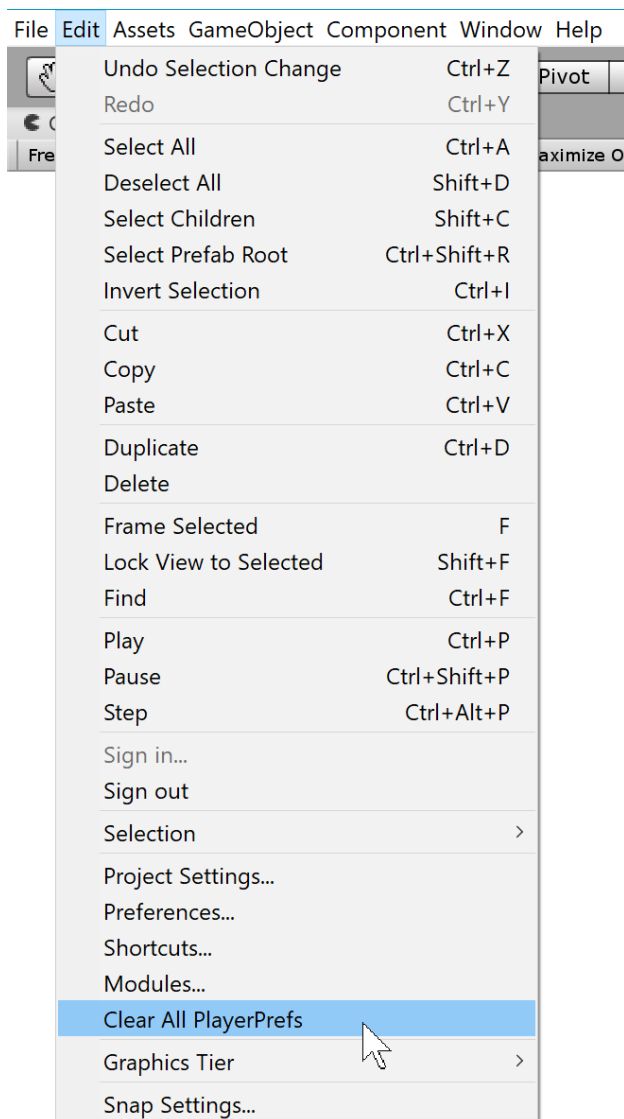
The AdManager object will be marked as DontDestroyOnLoad, and so you can use the following method anywhere in your project to show an interstitial ad:

```
void TryAd() {  
    #if UNITY_ADS  
    AdManager adManager = GameObject.FindObjectOfType<AdManager>();  
  
    if(adManager == null)  
        return;  
  
    adManager.Interstitial();  
    #endif  
}
```

The Ads folder also contains a script called 'AdManagerRewarded' that you can use to show a rewarded video advertisement. Simply put the script anywhere in your scene and call the **ShowRewardedVideo()** method (for example through the press of a UI button), to show a video. Then also make sure to add your rewards in the **RewardPlayer()** method below.

PlayerPrefs data

To save the player level and selected character color, the game uses Unity's build-in playerprefs. To reset the playerprefs, please go to Edit -> Clear all PlayerPrefs:



Conclusion

I hope you'll enjoy the '**Break it**' game template. Thanks again for purchasing. Please do not hesitate to contact me for any questions/suggestions regarding the asset via:

codeerstudio@gmail.com