Desktop Unity Net.Rumble Demo Game with PlayFab Party and Multiplayer

(Steam Store with Steam login variant)

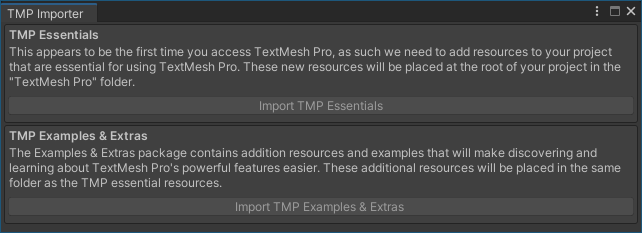
*\* This sample has been developed with Unity 2020.3.24f1, the Steamworks SDK and the PlayFab Unity SDKs on Windows.*

# Description

This is a simple multiplayer game. It is expected that developers should be familiar with the basics of using PlayFab Unity SDKs and PlayFab API as well as using Steamworks SDK, creating, configuring and developing games for Steam and uploading them to Steam Storefront with a Steam developer account, before building this sample game. They also need to create a PlayFab game title for this project in PlayFab developer portal, and know its Title ID. Additionally, they also need to create a Steam game for this project with their Steam developer account and know its AppId. They will need to configure Steam Add-on in PlayFab developer portal with their Steam AppId and Steam Web API Key. This sample app is intended to demonstrate how developers can use PlayFab Unity API and Steamworks API together to perform the following functions:

* Login to the Steam server.
* Login to the PlayFab server using Steam authentication token (Steam Ticket).
* Get the avatar, username, and user ID of the Steam login user.
* Find game lobbies, join game lobbies, create game lobbies, customize lobby attributes, and use game matchmaking using PlayFab Multiplayer Unity API.
* Create a Party network, have players join the Party network for message/data exchange and voice chat using PlayFab Party Unity API.
* Look up the Steam friends list and invite friends to join the game lobby.
* Accept a Steam friend invitation when not running the game, run the game and join the lobby.
* After joining the lobby, display other Steam player avatars, player names, other player information, update player information, synchronize information of players in the lobby, and change ship color, ship type.
* After joining the lobby, start a multiplayer game and Party voice chat.
* Get Steam player inventory as well as update player inventory.
* Get Steam player achievements, scores, and display leaderboards.
* Create new leaderboards, upload game scores, and upload statistical information.

# Building the Sample

* This example depends on the PlayFab Unity SDKs, Steamworks.NET Unity package and Steamworks SDK (upload tools only). Developers will need to download PlayFab Unity SDKs, Steamworks.NET Unity package and import them into Unity to satisfy the SDK dependencies required by the example.
* PlayFab Party Unity SDK download link: [PlayFab/PlayFabPartyUnity (github.com)](https://github.com/PlayFab/PlayFabPartyUnity) (release 1.7.6.0-main.0 was used at the moment of writing this document)
* PlayFab Multiplayer Unity SDK download link: [PlayFab/PlayFabMultiplayerUnity: PlayFab Multiplayer Unity SDK (github.com)](https://github.com/PlayFab/PlayFabMultiplayerUnity) (release 1.3.0.0-main.0 was used at the moment of writing this document)
* PlayFabEditorExtensions Unity plugin download link: <https://aka.ms/PlayFabUnityEdEx>
* Steamworks.NET Unity package download link：[Releases · rlabrecque/Steamworks.NET (github.com)](https://github.com/rlabrecque/Steamworks.NET/releases)
* In opened Unity project, navigate to Assets/Import Package/Custom Package in the main menu to import the Steamworks.NET Unity package.
* To use the Steamworks.NET, make sure that a macro symbol "STEAMWORKS\_NET" is defined in Player à Other Settings à Script Compilation à Scripting Define Symbols fields of Project Settings screen.
* Make sure that “Allow ‘unsafe’ Code”is enabled (checked) in Player à Other Settings à Script Compilation section of Project Settings screen.
* The example has only one "SampleScene.unity" scene and the path is: "Assets/Sample/Scenes".
* Upon opening the scene, you may be prompted to import TextMesh Pro resources in a popup window, please import both essentials and extras:  
  
* Create a new empty object in SampleScene, name it SteamManager and add the SteamManager.cs script to it. The path to SteamManager.cs is "Assets\Scripts\Steamworks.NET\SteamManager.cs"
* The SteamManager.cs originates from: [rlabrecque/Steamworks.NET-SteamManager: This is a basic Unity script which controls Steamworks.NET (github.com)](https://github.com/rlabrecque/Steamworks.NET-SteamManager)
* Please check whether the PlayFabEnvInitializer object exists in SampleScene. This is a prefab, add it to the scene if necessary. Please check its Inspector panel if the script is mounted, to see if the PlayFabMultiPlayerManagerPrefab and PlayfabMultiPlayerEventProcessorPrefab of the PlayFabEnvInitializer script are empty. If they are empty, please add them.

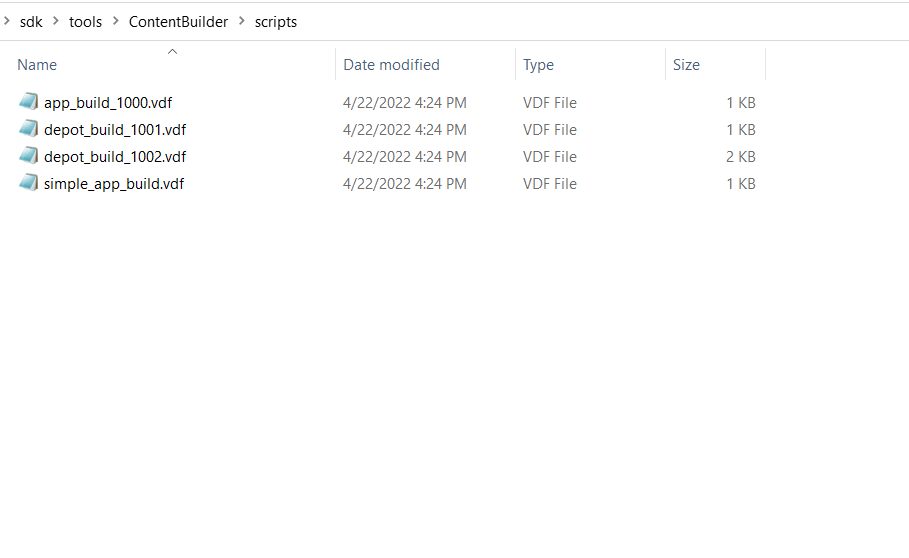
PlayFabMultiPlayerManagerPrefab local path： Assets\PlayFabPartySDK\Prefabs\PlayFabMultiPlayerManagerPrefab.prefab

PlayfabMultiPlayerEventProcessorPrefab local path：Assets\PlayFabMultiplayerSDK\Prefabs\PlayfabMultiplayerEventProcessor.prefab

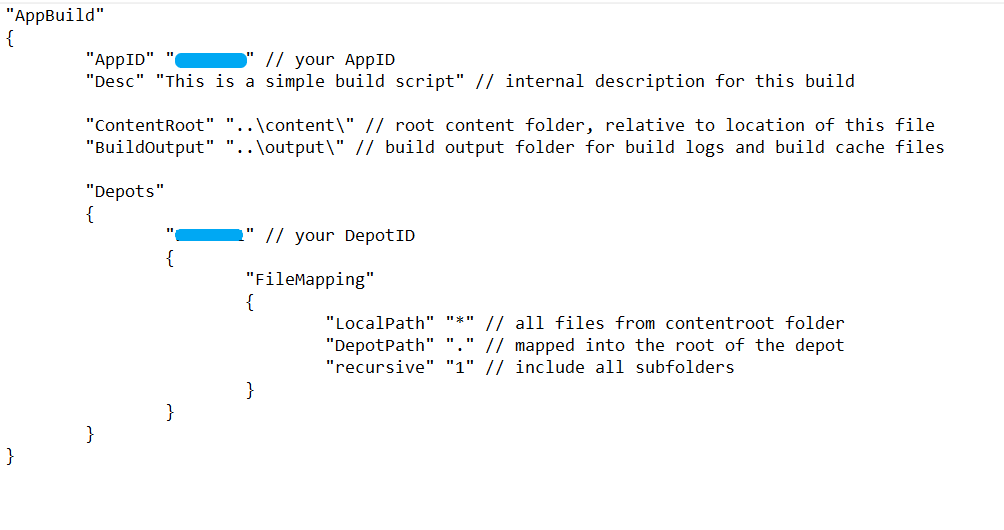
* Open the PlayFabEnvInitializer script, modify TITLE\_ID to set your PlayFab Title ID. PlayFabEnvInitializer local path: Assets\Sample\Scripts\PlayFabLogic\PlayFabEnvInitializer.cs
* Download and unzip the Steamworks SDK with upload tools before building, download link：<https://partner.steamgames.com/downloads/steamworks_sdk.zip>

# Packaging Tutorial：

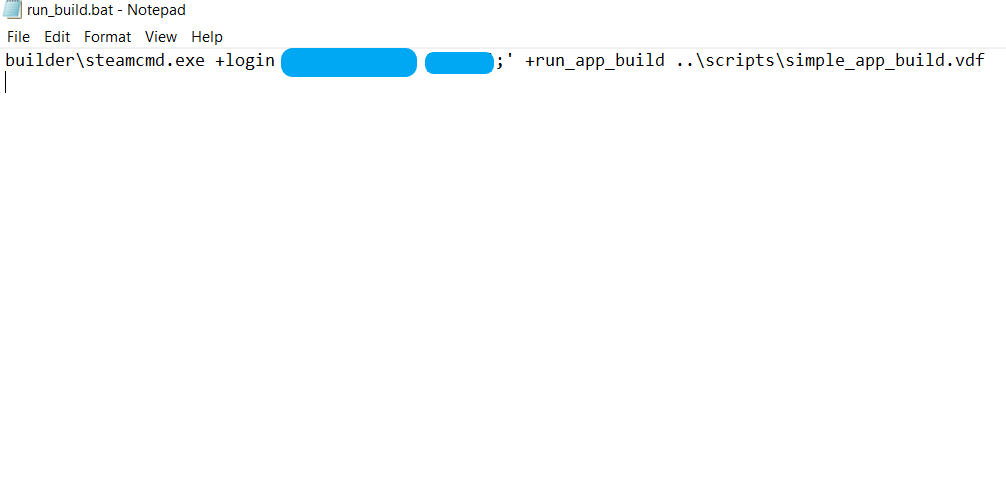
1. Find "simple\_app\_build.vdf" in the Steamworks SDK folder: "\SDK \tools\ContentBuilder\scripts" and open it in a text editor



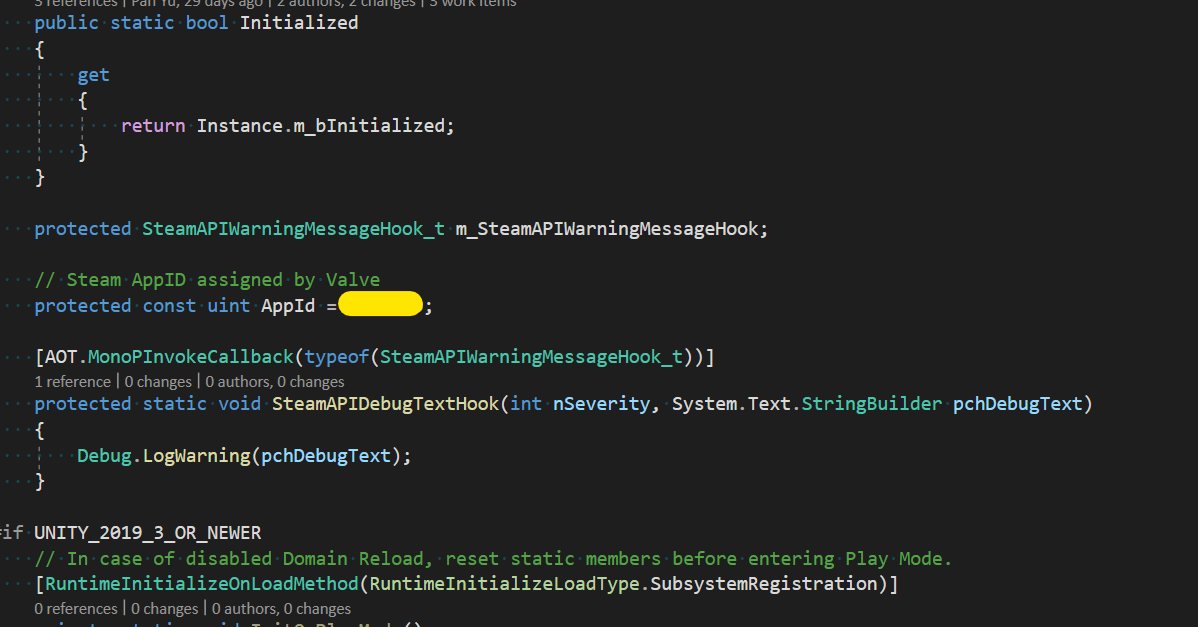
1. Enter your Steam game AppID and your DepotID, save and close.



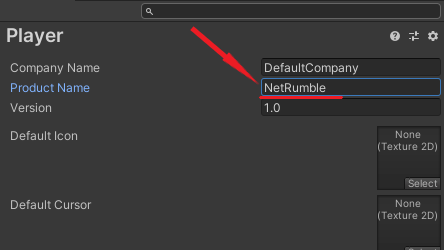
1. Modify "\SDK\tools\ContentBuilder\run\_build.bat" by adding the required Steam account login information as shown in the following figure, save and close:



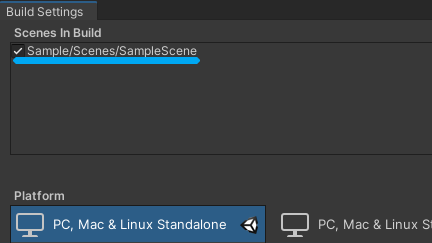
1. Use an editor to open the SteamManager.cs file, file path is “\Assets\Scripts\Steamworks.NET\SteamManager.cs” , find the line that defines AppId constant (AppId =  ) and set its value to your Steam game's AppId , as shown in the following figure:



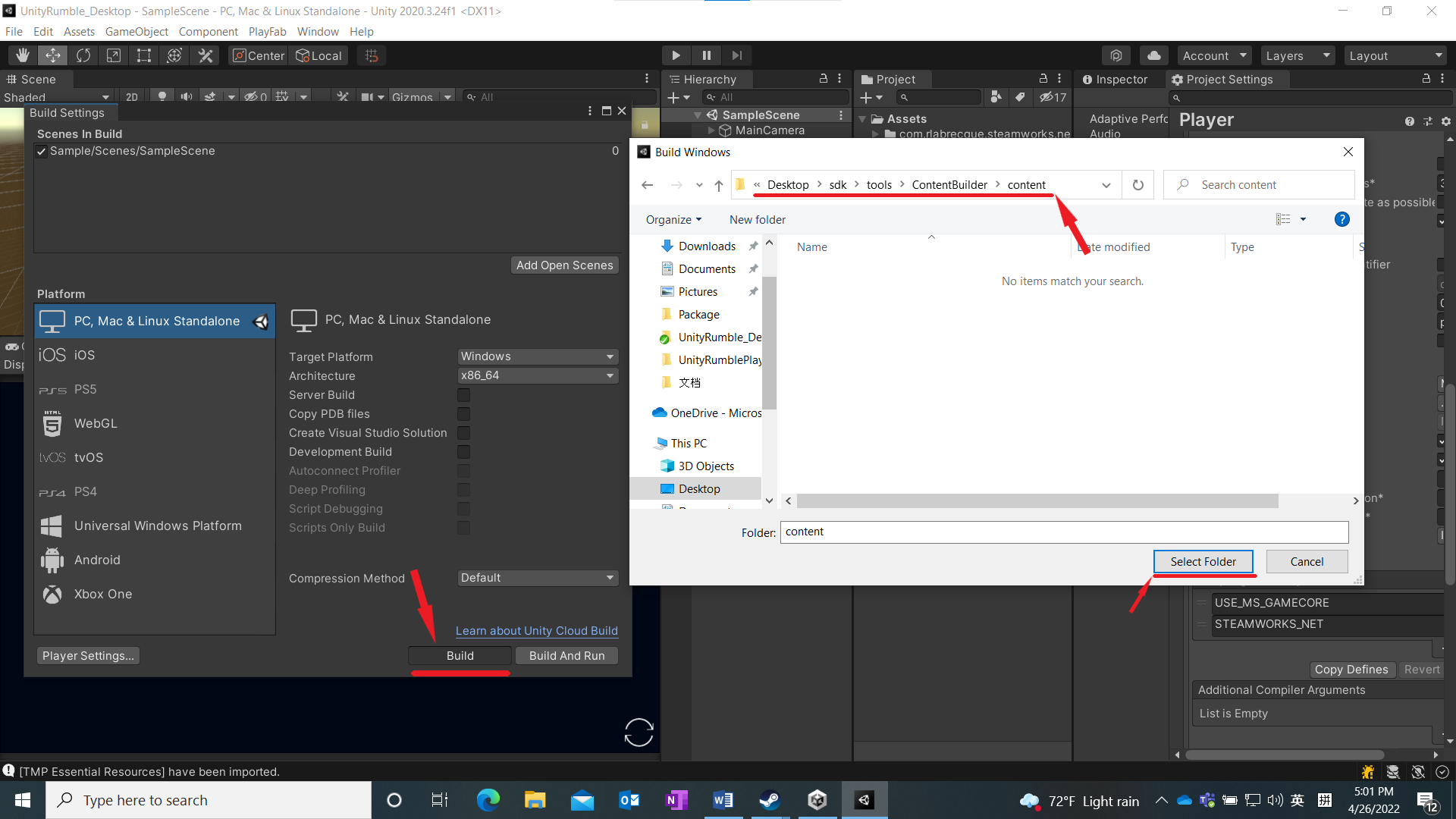
1. Set Product Name to be the same as the game name set on STEAMWORKS dashboard (otherwise Steam will not be able to launch the game)：



1. Add SampleScene to Scenes in Unity Build Settings.



1. Click the "Build" button to select the build path and set the build path to "\SDK\Tools\ContentBuilder\Content\", click select Folder button to start building, and the build Package Folder will be opened automatically after the build is completed. The folder structure is shown in Figure 2 below:

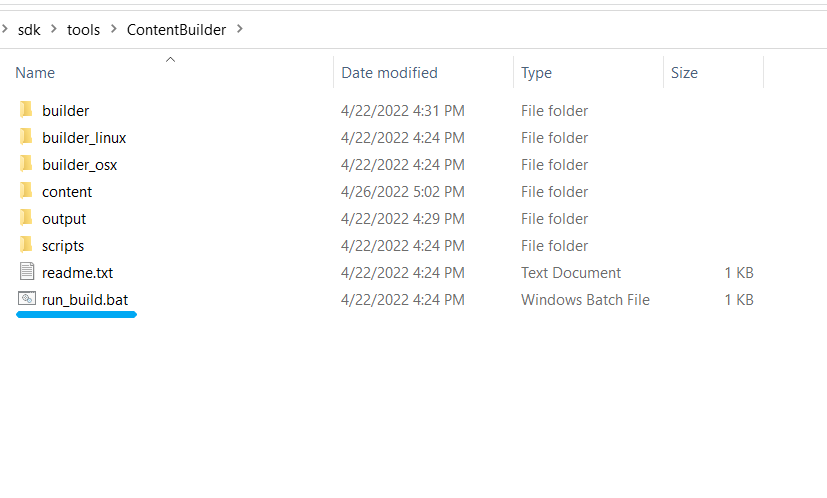


（Figure 1）

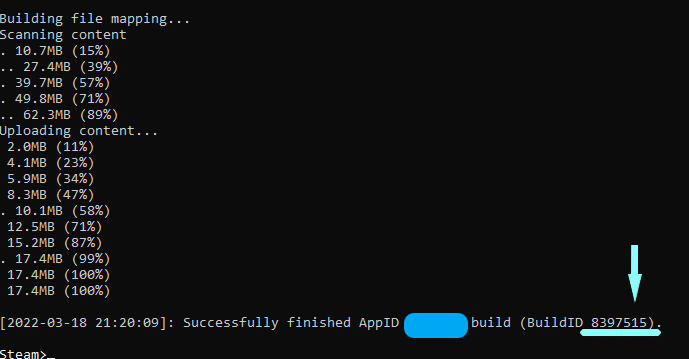


（Figure 2）

1. Run the run\_build.bat



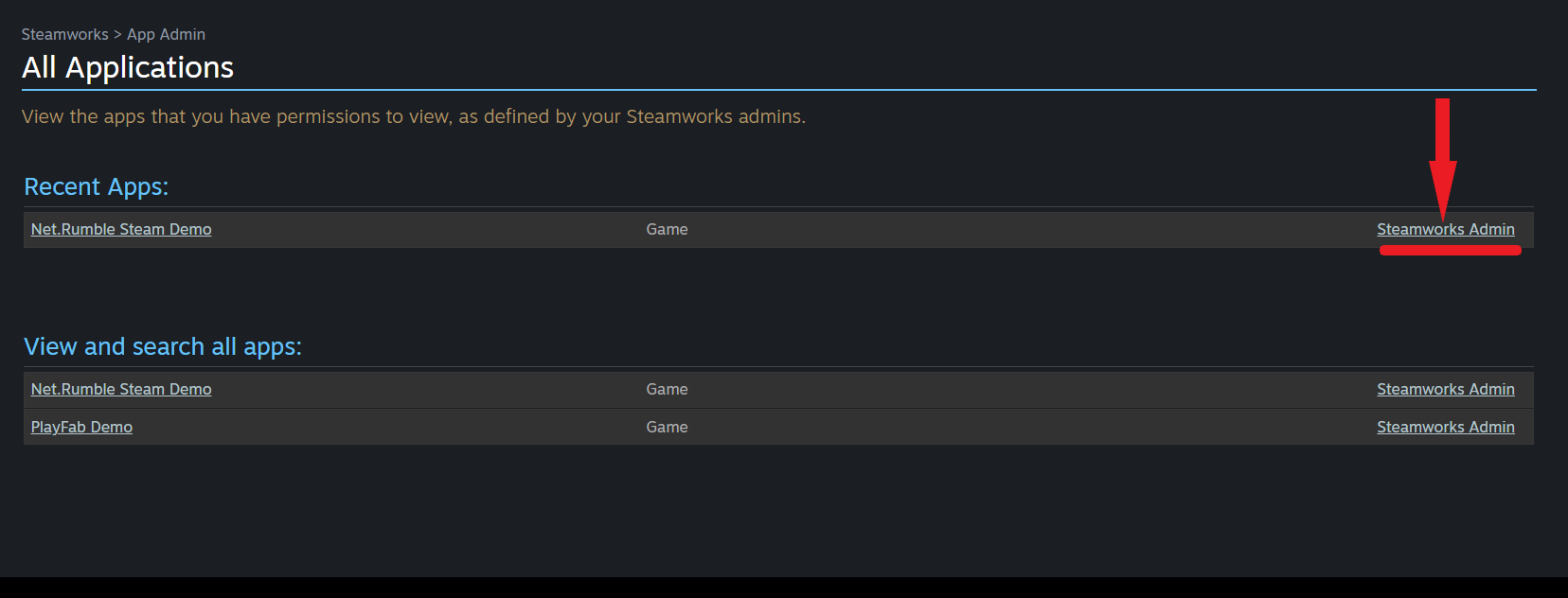
1. After the package is uploaded, view the Build ID



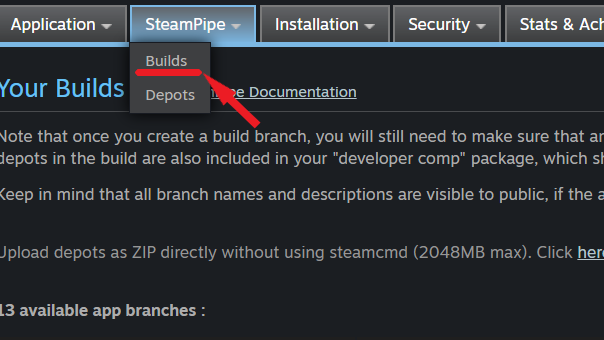
1. Open the Steam Dashboard, log in with Steam Account, and click All Applications.



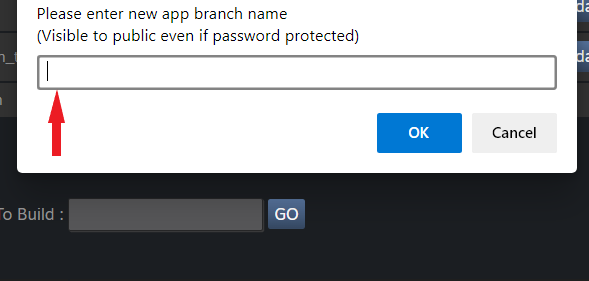
1. Find your project, click the Steamworks Admin.



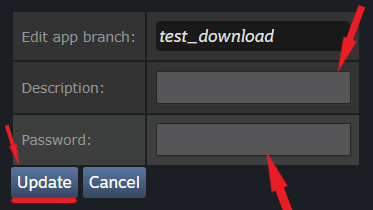
1. Click SteamPipe and click Builds.



1. Click the "Create new app branch", enter the branch name to input box, and click OK



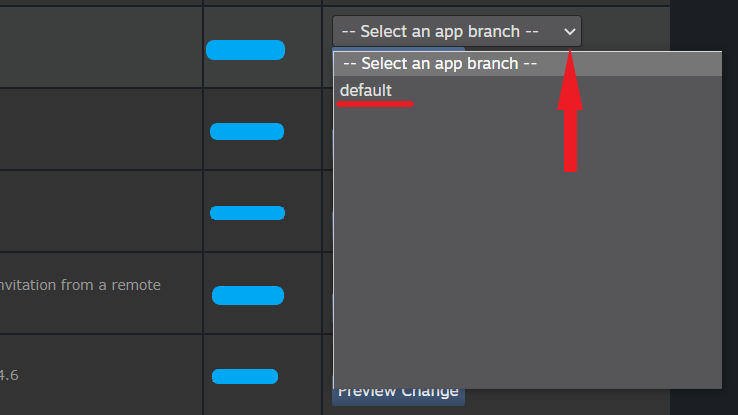
1. Enter the password in the Password input box, enter the description in the Description input box, and click the Update button.



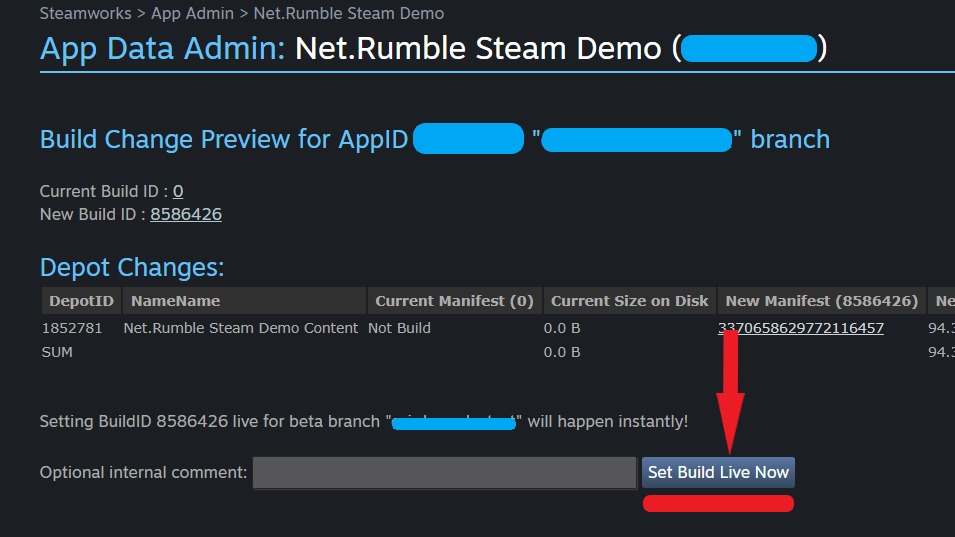
1. The red box is a build, red line is build id. Find the right build in the list.



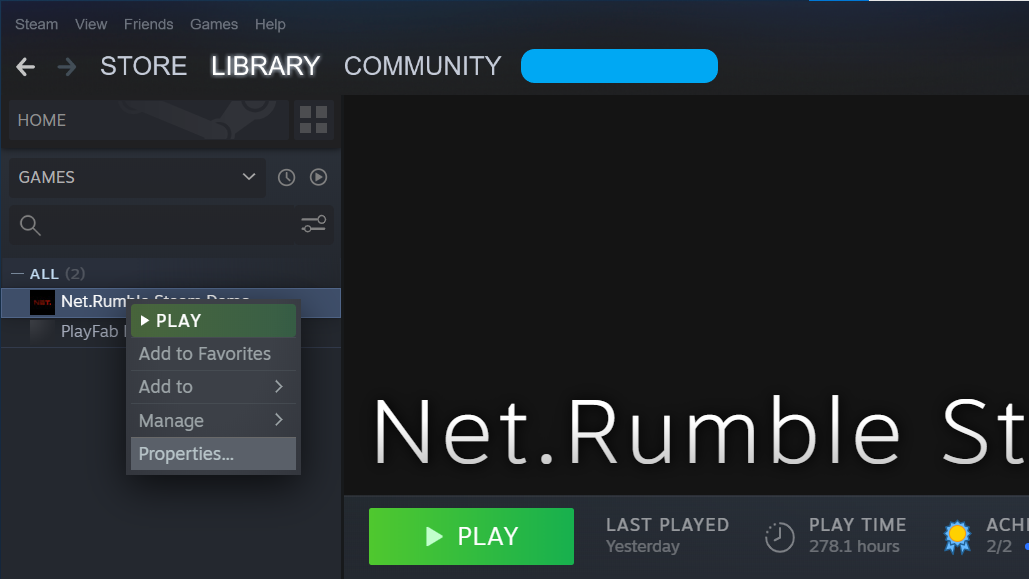
1. Update the newly generated build version to the created branch. Select the branch and click it.



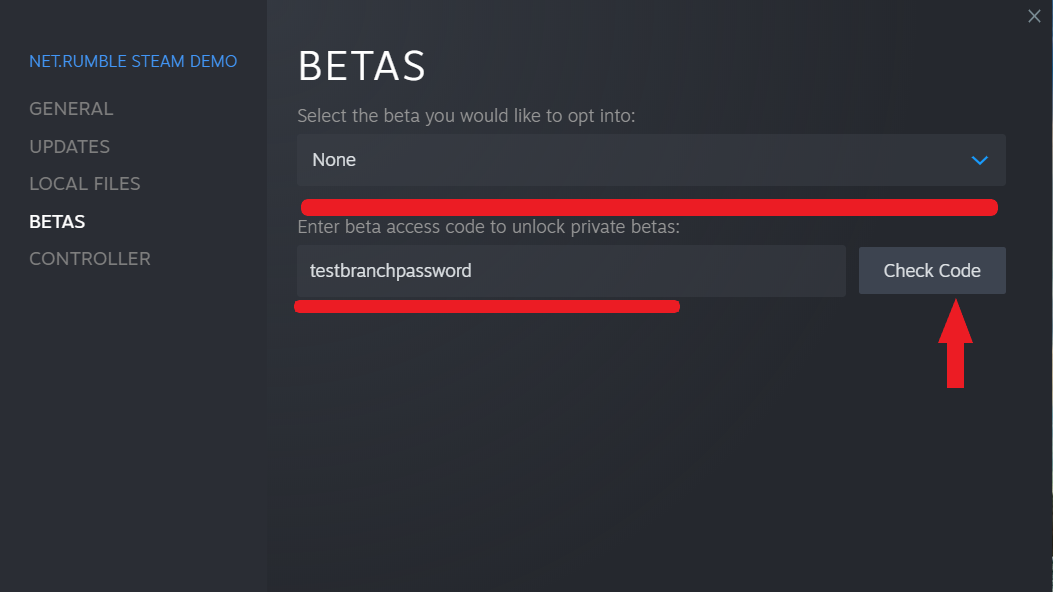
1. Click the Set Build Live Now button to update the build version.



1. Open Steam, click Library, right-click the test game. Click the Properties button to jump to the properties interface



1. Click BETAS to open BETAS screen. Enter the beta access code to unlock private beta and click the Check Code button.





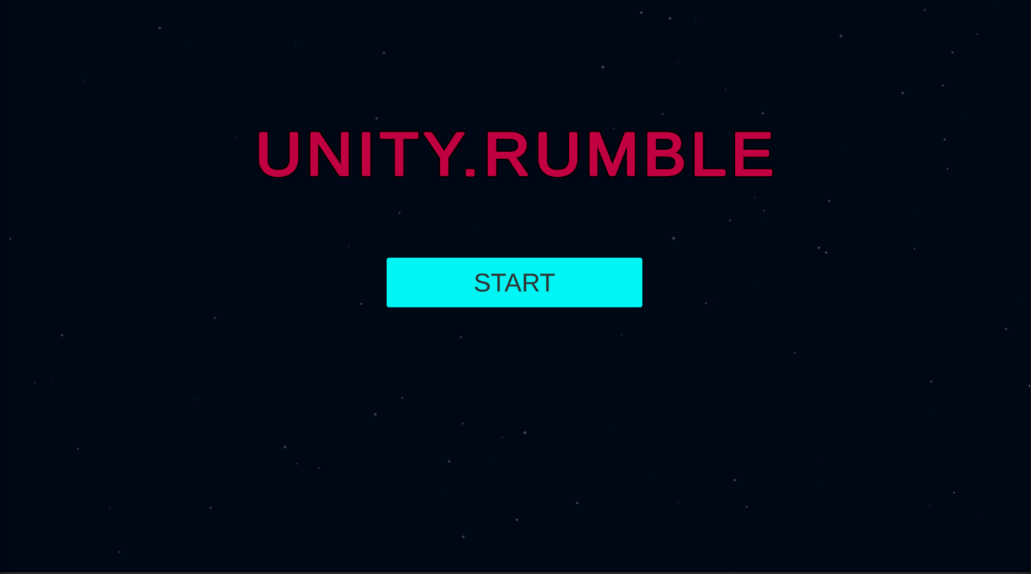
**Steam will automatically update apps downloaded from the beta branch.**

# Running the sample

After the app has been successfully built and installed on a Windows 10 PC, there will be an app icon called "Net. Rumble Steam Demo" on the desktop. Double click the application icon to launch the game.



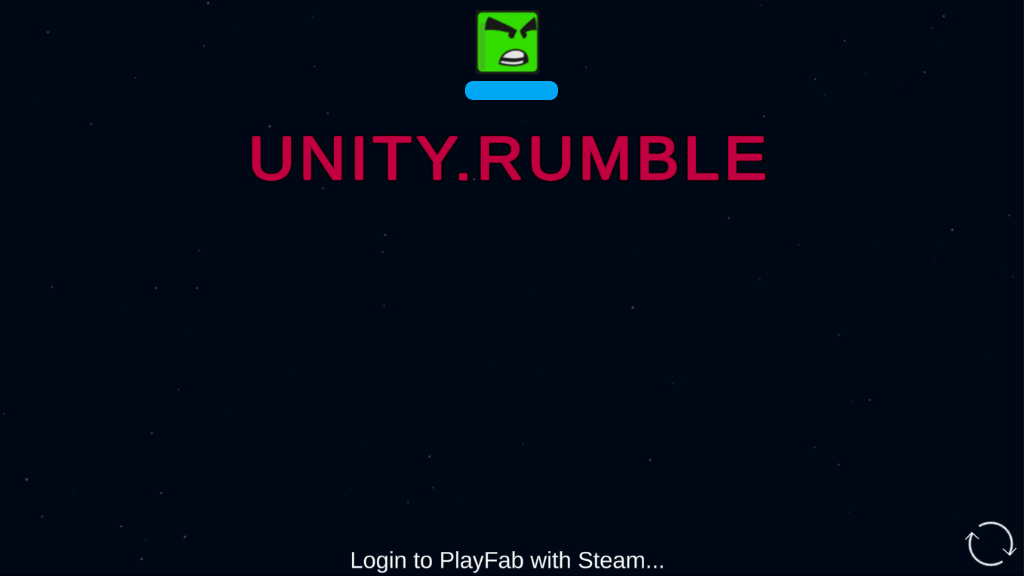
# Sample Start Screen



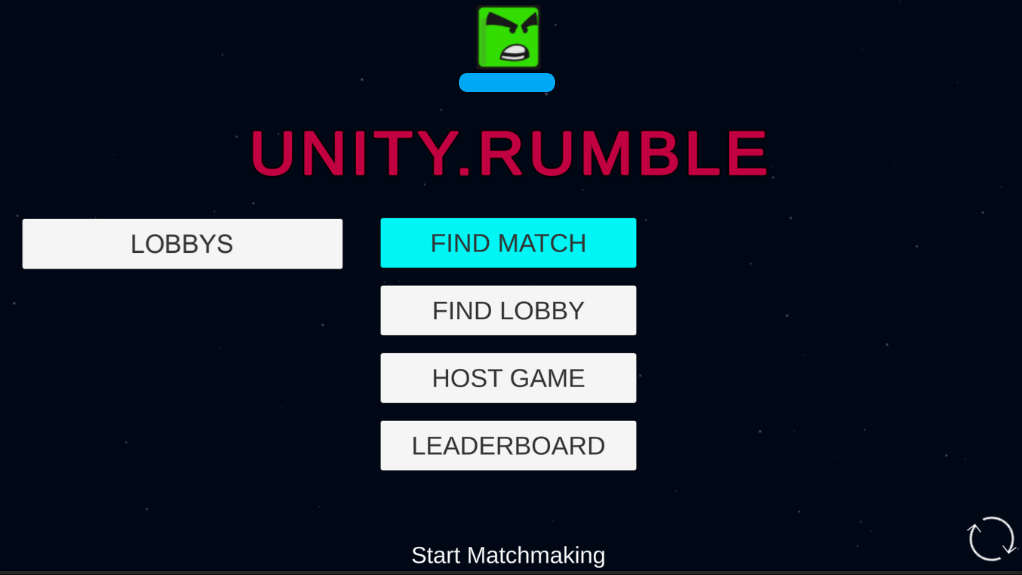
The screenshot shows the first interface when the sample is launched. Click the "Start" button to log into the Steam and PlayFab servers. If any of the login steps fail, a failure message is displayed at the bottom of the screen and the screen remains on the start screen.

Common failure scenarios may be that users are unable to access their current Steam account, their Internet connection is disconnected, or PlayFab is experiencing some form of service disruption.

# Login to PlayFab with Steam

The screenshot above shows the next screen that appears when the sample launches. PlayFab user login process that is ongoing after clicking the start button. The bottom of the screen shows that you are logging in.

# Sample Main Menu Screen



When the user successfully logs into PlayFab, the Main Menu screen for the sample is displayed. The four main features provided are:

1. Click the "FIND MATCH" button to start matching the game.

2. Click the "HOST GAME" button to create a game lobby which will make the lobby available for other players to find and join it.

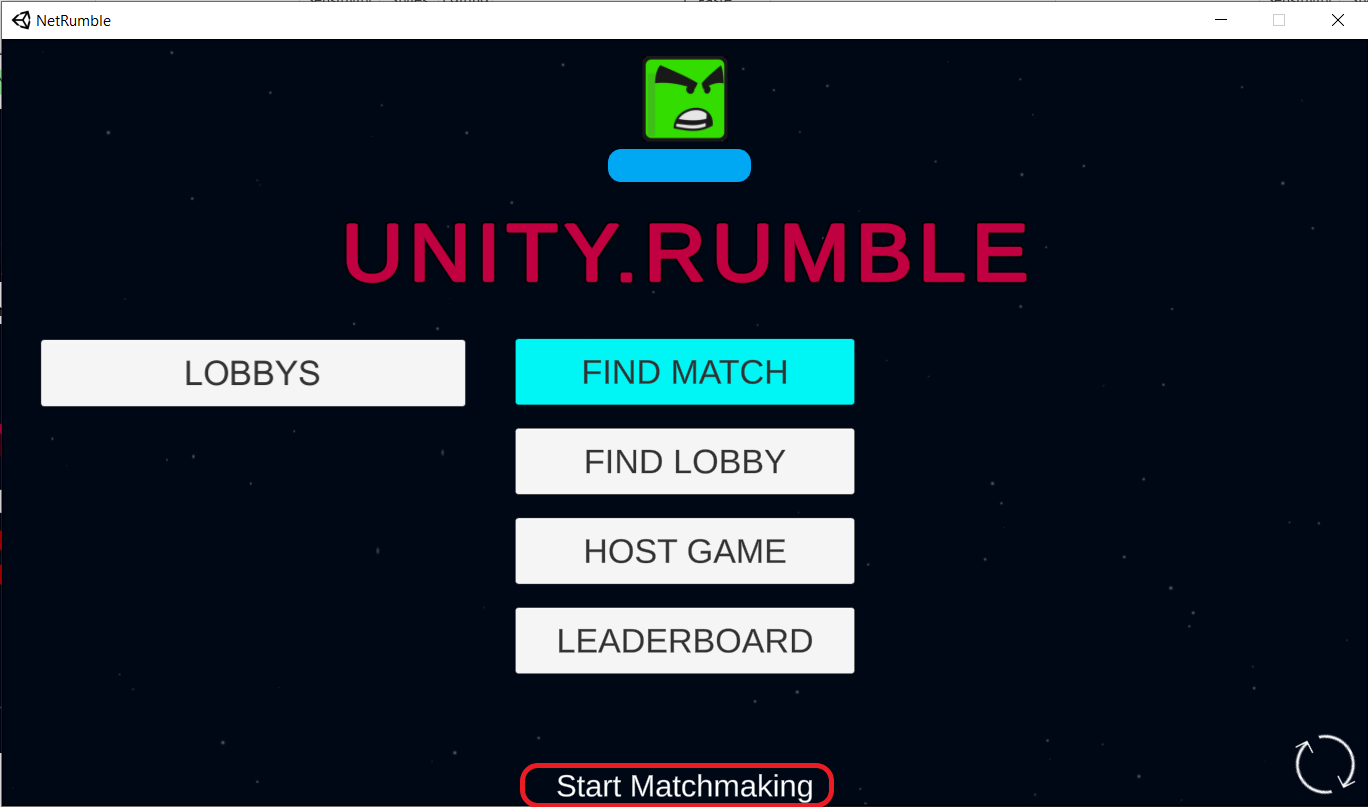
3. Click the "FIND LOBBY" button to display all the available game lobbies. After a lobby has been found all available lobbies will be displayed under "LOBBIES"

4. Click the "LEADERBOARD" button to open the Leaderboard Screen.

If any of the above functions fail, the error message is displayed at the bottom of the screen and the user is left on the Main Menu screen.

# Find Match Screen

When the "FIND MATCH" button is clicked, the player starts matching the game, as shown below, with the message that the player is matching at the bottom of the screen



When the player successfully matches the game, the lobby screen opens, as shown below:



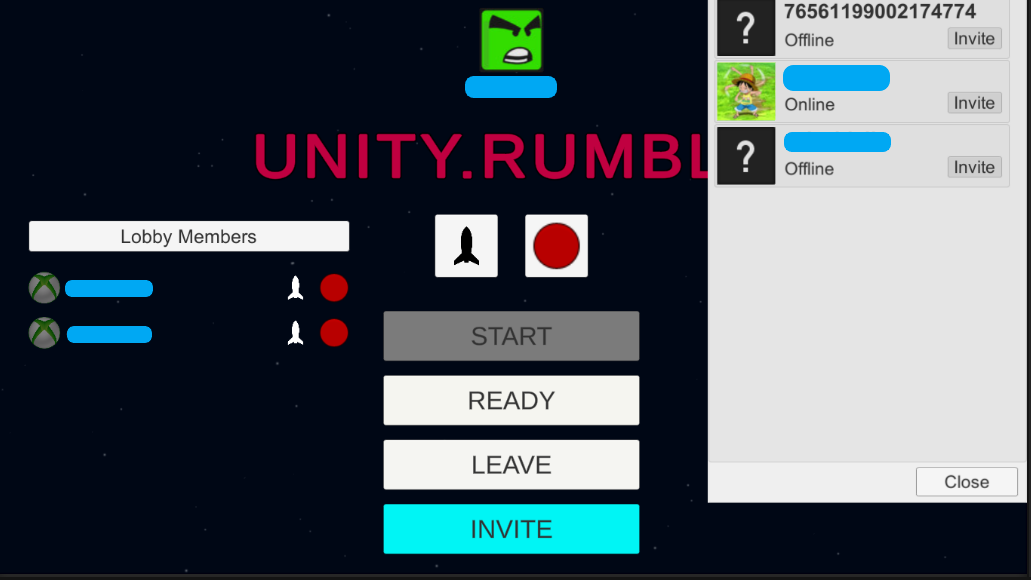
# The Game Lobby Screen

Click the "HOST GAME" button and PlayFab will create a Lobby for the player, which can be found by other players. Once the lobby is created, the GAME Lobby Screen will open, as shown below:

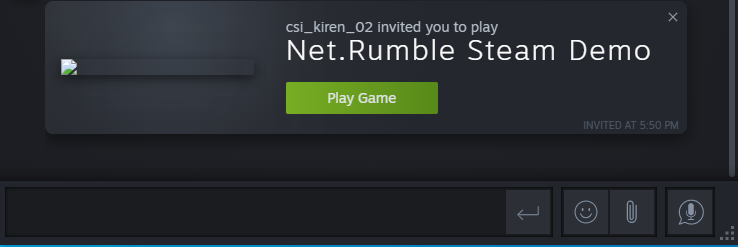


# Host and Invite Friend into a Lobby

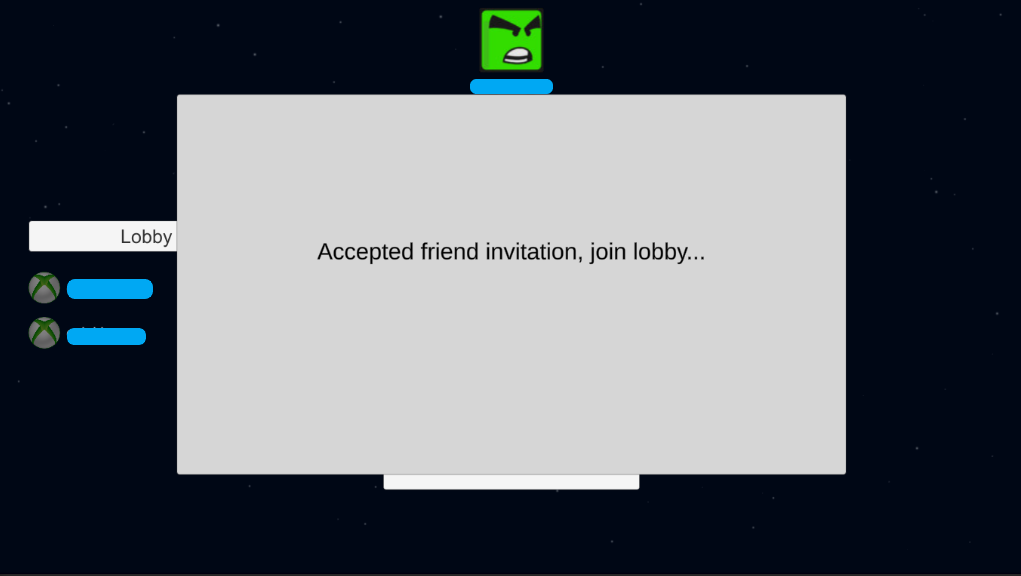
The player creates the game Lobby on the "Host Game" button on the "Main Menu" screen, and the "Invite" button appears on the Game Lobby screen. Clicking the "Invite" button will open the Steam friends list, where users can select friends to invite, as shown below:



In Steam, if the invitation is not cancelled, the recipient will receive a notification that they have been invited to play Unity Rumble at which time they can either accept or decline the invitation. If the invitation is accepted and the Steam application is not running, the game will automatically start and join the lobby they were invited to. If the Steam application is already running, the player will automatically be invited to the lobby and the game screen will move to the game lobby. To start the game, click the “Play Game” button to accept the invitation, as shown below (There is no game icon in the invite screen because the project is not published):



If the game is already running and the game invitation is accepted, the accept invitation interface will open, and the player will automatically join the invited room, as shown in the following figure:



# The Ship Style and Color

Lobby members can change the style and color of their ships by clicking the "Ship" button and the “Color” button.

Ship Button：

Color Button：

This example has four different ship styles and six different ship colors. The following picture shows the interface after clicking the ship button and the color button:



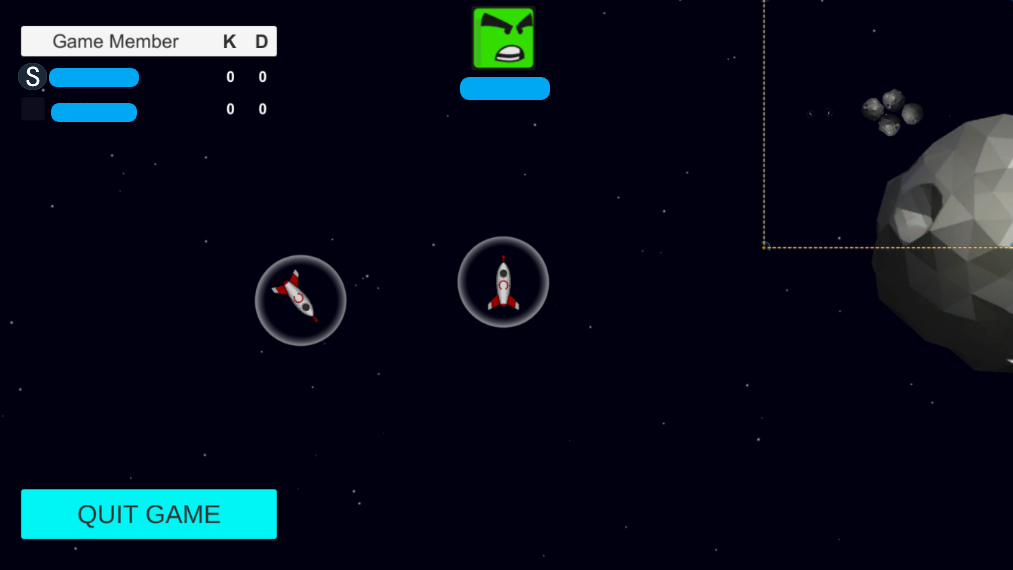
# The Lobby All Ready

When the members of the Lobby are ready to begin playing, they can change their state of ready by clicking the "Ready" button will synchronize the state with other members of the Lobby. When all players are ready, the player who created the Lobby can click the "Start" button to start the game and notify all members of the lobby that the game is beginning. All members of the lobby will see a countdown at the bottom of the screen indicating that the game is about to begin. The ready status is displayed before each username. Gray indicates that the user is not ready, and green indicates that the user is ready.

The following picture shows all users ready to play:



# The Game Play Screen



The upper left corner of the Game Play screen shows the current game's player, with username and "kill" and "death" statistics.

The “S”(Steam) icon to the left of the host's name indicates whether it is the owner. If a user clicks the "Quit Game" button to quit the game, he will disappear from other player's player lists. Players can make real-time voice calls after joining the game.

The game ends when a player has a kill score of 5. At this point, the game end screen will pop up, and click the "OK" button to return to the Main Menu screen, as shown below:



The Leaderboard Screen

Click the Leaderboard button on the Main Menu screen to open the Leaderboard screen as shown below:



In the Leaderboard Screen, the overall Leaderboard is displayed in the middle position, and its Leaderboard data is displayed in the lower right corner. Click the REFRESH button to REFRESH Leaderboard information, and click the CANCEL button to return to the Main Menu Screen.

# The Profile Screen

Almost all screens display the player information, including their profile picture and username, as shown below:



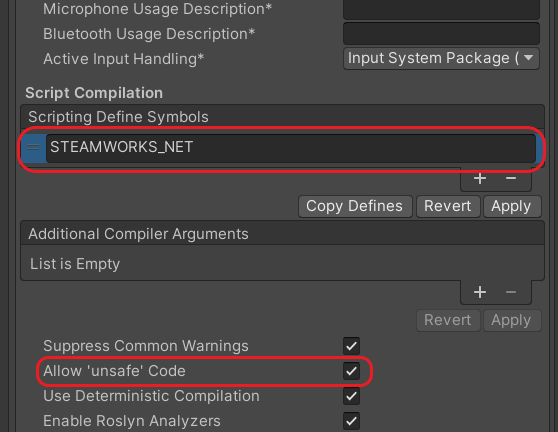
## Implementation notes

You can find the sample scripts in the “Assets/Sample/Scripts" folder, which contains "UI/View"-related code and core logic.

• PlayFab Unity SDKs-based code for Party, Lobby, Matchmaking and voice chat are in Assets\Sample\Scripts\PlayFabLogic

• Steamworks.NET-based code for Profiles, Achievements, Leaderboard and Inventory are in Assets\Sample\Scripts\SteamLogic

•To use the Steamworks.NET plugin, make sure that a macro symbol "STEAMWORKS\_NET" is defined in Scripting Define Symbols, as shown in the image below:



## Known issues

This example is based on the latest version of PlayFab Unity SDKs and Steamworks.NET at the time of development. Using newer versions of PlayFab Unity SDKs and the Stemworks.NET Unity plugin or integrating their different versions may cause incompatibilities with the current code.

## Privacy Statement

For more information about Microsoft’s privacy policies in general, see the [Microsoft Privacy Statement – Microsoft privacy](https://privacy.microsoft.com/en-us/privacystatement/).