

# PICTURE QUIZ +



<b>ABOUT THE GAME .....</b>	<b>2</b>
<b>PREPARE YOUR PROJECT .....</b>	<b>3</b>
<b>SETTING UP YOUR OWN CONTENT .....</b>	<b>10</b>
<b>KNOWN ISSUES .....</b>	<b>17</b>

# ABOUT THE GAME

Picture Quiz is a game where user should guess what is going on on a picture and input a right answer in a given field. On the start he only has an information about word`s length. Also there is a group of mostly random letters that includes right ones. A user can tap (click) this letters to let them fill empty fields. If an entered word matches a right answer this level considered as complete. In case if a user get stacked there are three standard hints available. This hints costs some coins. A user can earn them in a couple of ways. The first is to complete a level and get coins for it. The second is to watch an AD and to get coins as reward. And the third is to buy coins for real money. Now about the hints. The first is "Reveal a letter". It opens the first (second etc.) right letter in a word and block it to avoid clearing. The second hint is "Remove letters". It removes almost all wrong random letters that are given at the start. This hint lets a user to assemble a word from given right letters as a way to complete level. And the last hint is "Get an answer". It automatically enters a right word and complete a level. Other hints depend on the type of level. Check it in the game after you complete configuring your project.

# PREPARE YOUR PROJECT

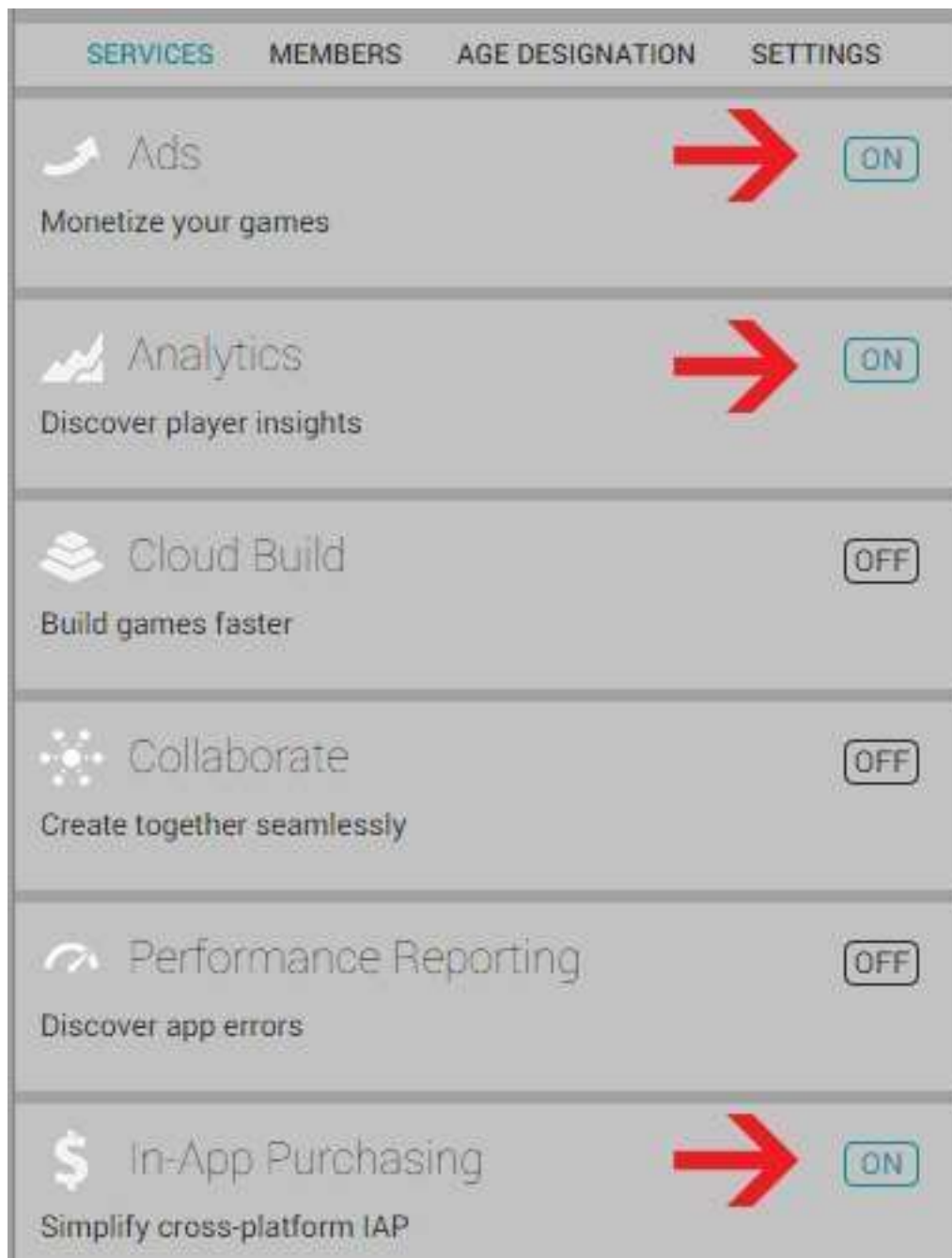
1. Prepare your new 2D clean project. Switch platform to Android or IOS. You should install Unity IOS or Android support for the editor if you don't have ones. Than install Android SDK and Java if needed.

The official guide:

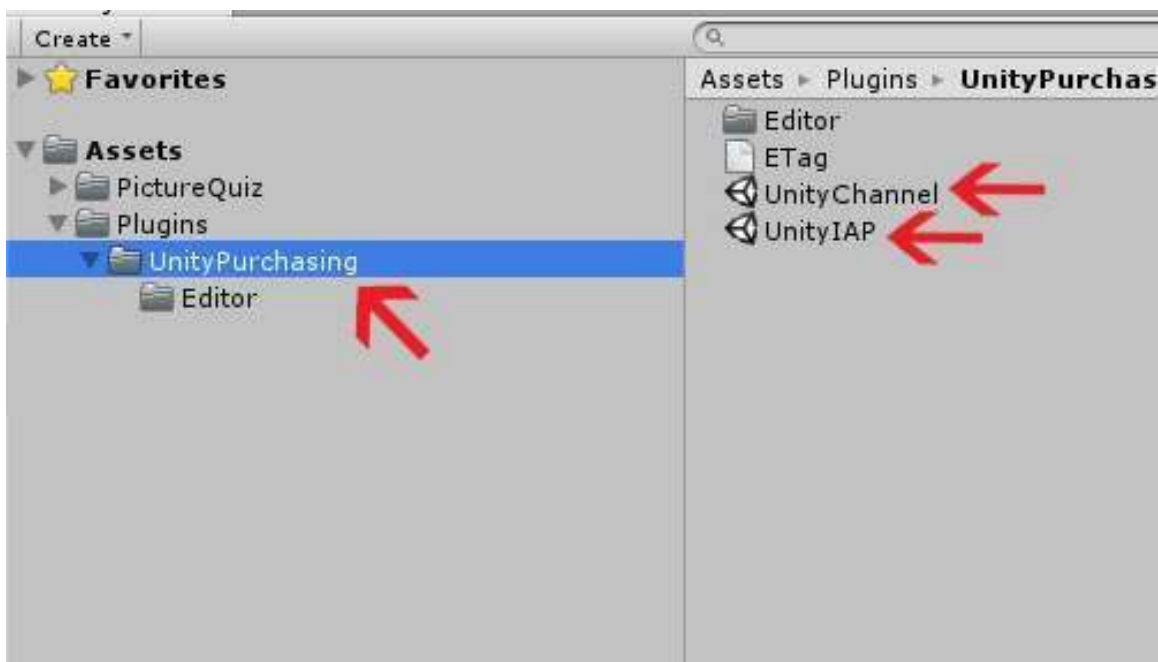
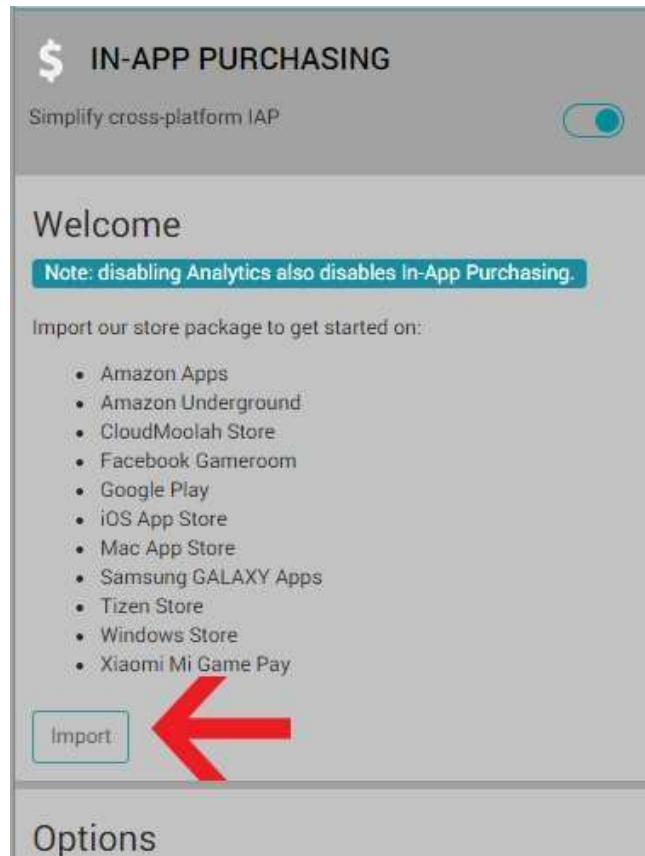
<https://docs.unity3d.com/560/Documentation/Manual/android-sdksetup.html>



2. Go to Services and create new Project ID. Then turn on Ads, Analytics and In-App Purchasing if you need them.



Go to In-App Purchasing and click the Import button. “Plugins” folder will be installed in your root Assets folder. Go there and install this packages

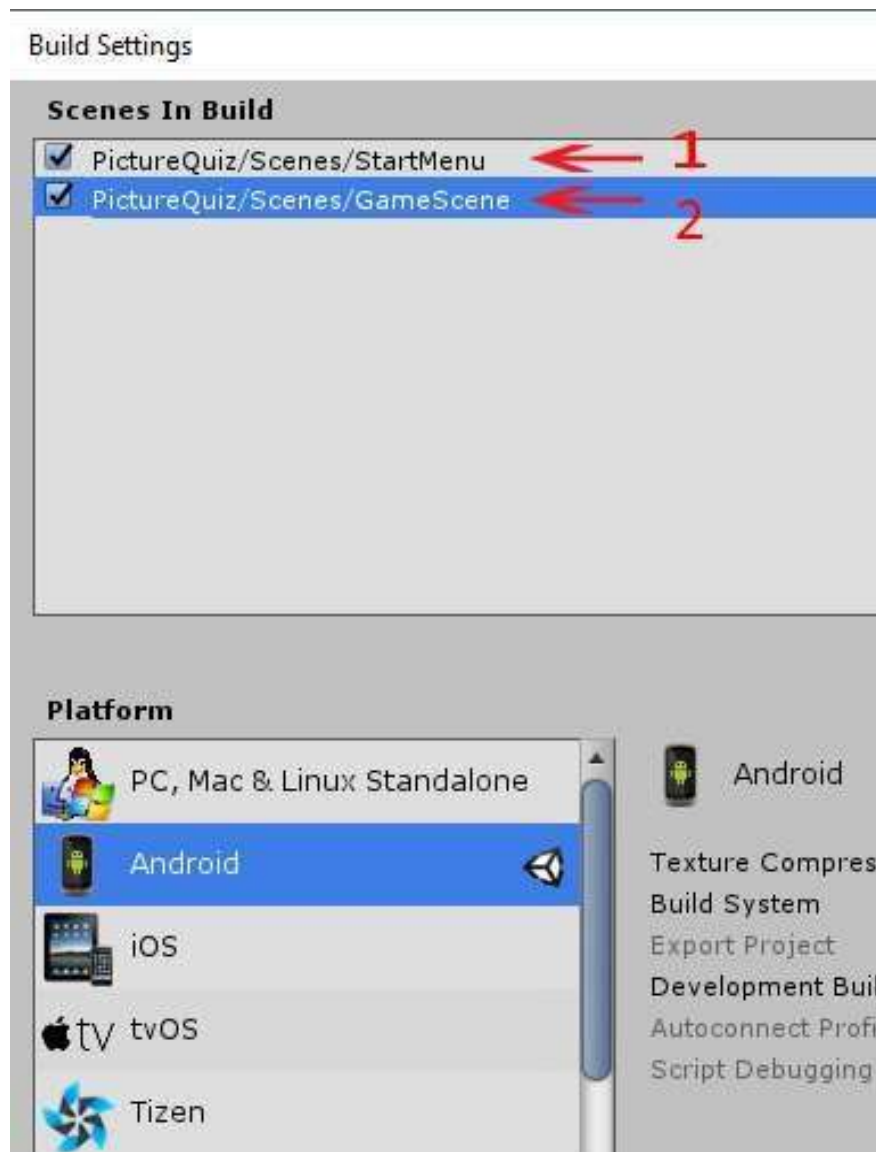


3. To set up In-App Purchasing please visit an official guide. Note that you can do it only after you have published the game in the Google Play or the App Store but you can test this while in the editor.

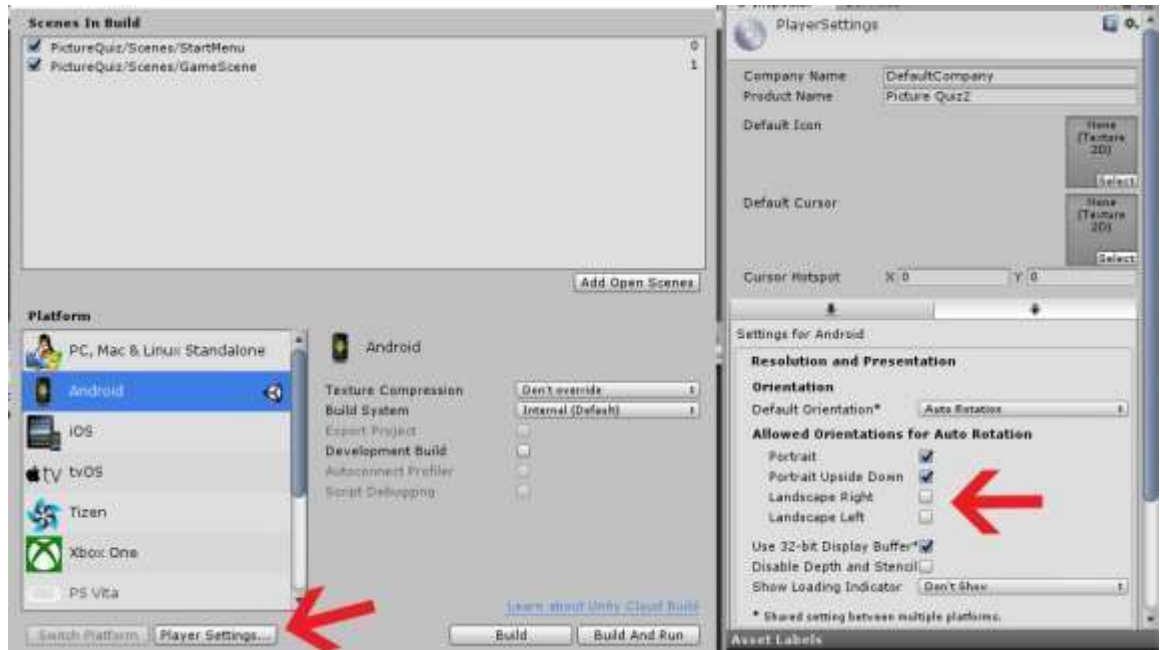
<https://docs.unity3d.com/Manual/UnityIAPGoogleConfiguration.html>

<https://docs.unity3d.com/Manual/UnityIAPAppleConfiguration.html>

4. Go to File>Build Settings and drag and drop scenes from "PictureQuizPlus/Scenes" in a following order



5. You also need to uncheck Landscape aspect ratios in “Build settings/Player Settings/Resolution and Presentation”



6. Select any portrait resolution in Game Tab.





7. In order to configure all Advertisements and IAP settings please follow to **Tools > Open ADsIAP Settings**. Also AdsSettings.asset could be found and clicked in Scripts folder.



Check or uncheck needed platforms, wait a bit for dropdown menus to appear, configure them or follow the guides by clicking buttons from above if you don't know how to implement this.

If both Ads platforms are enabled than the target platform would be chosen randomly 50/50%;

When using AdMob you should set the timer not less than for 10 seconds to have time to prepare new rewarded video.

Now the game is ready to start.

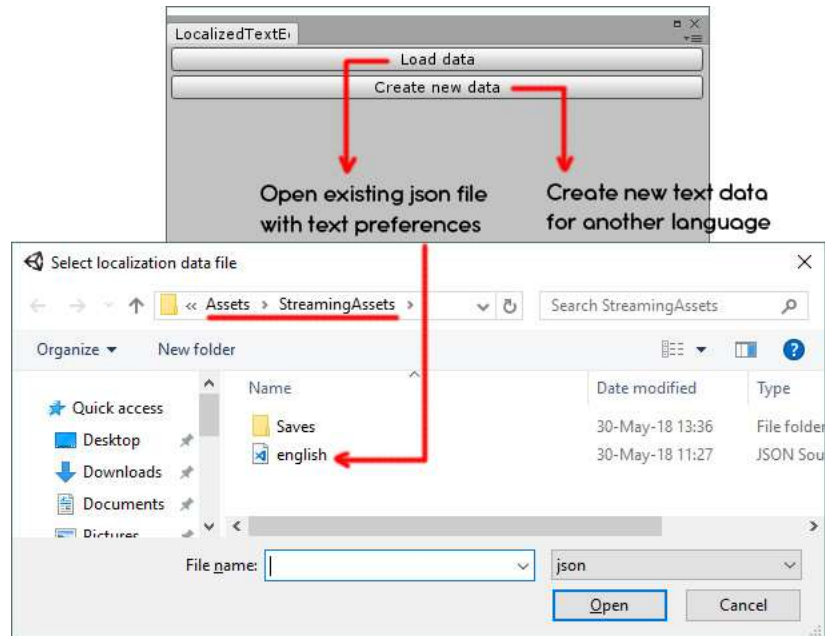
Double click on **"StartMenu"** scene in "Picture QuizPlus/Scenes" and click Play. Note that you should always run the game from that scene.

# SETTING UP YOUR OWN CONTENT

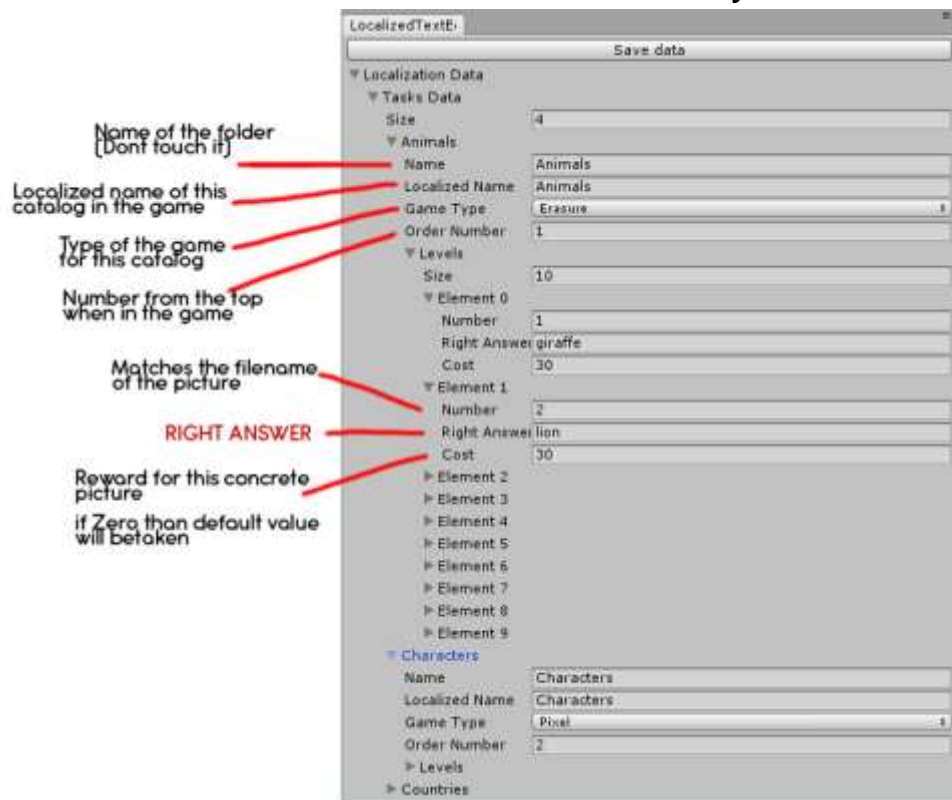
1. All your pictures must be placed in “Resources” folder and supposed to have a square shape. Every subfolder represents a new directory. The name of the subfolder matches the directory in the game. The pictures filenames in subfolders supposed to be numbers. This numbers represent an order that pictures should appear in a game.
2. In APK size and RAM optimization purposes you should set ETC2 compression for all your pictures in Resources. Chose all pictures in a folder with Shift. Go to Inspector, click Override for Android, chose ETC2 compression and hit Apply.



3. Go to **Tools> Localized Text Editor**. To watch and edit existing file click Open file and chose filename with related language. Otherwise to set up new content (in case if you add more tasks or directories or want to add another localization) click Create new data.



4. When you chose Create new data tasks data is automatically lined up based on the names and contents of your folders. Fill the other fields manually.

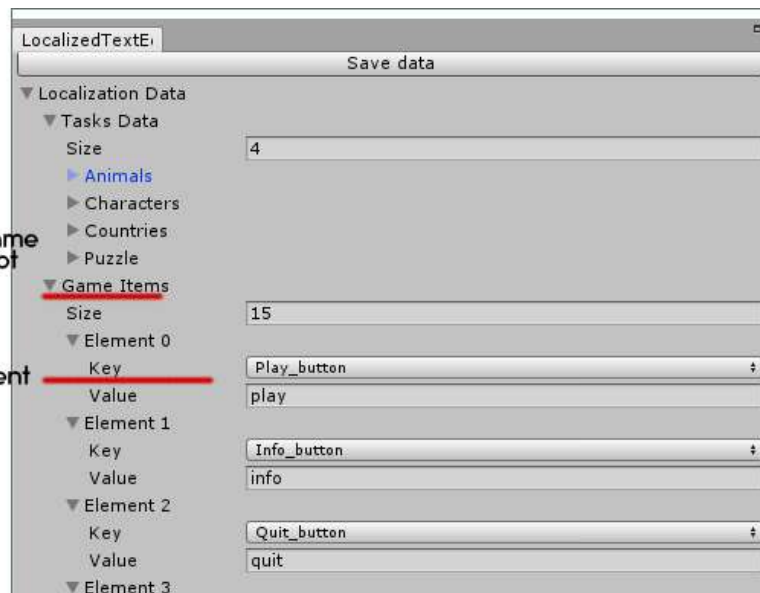


Game Types: Erasure – erasing with a spatula  
 Pixel – pixelated picture  
 Planks – shooting planks with an aim  
 Default – default opened for look picture

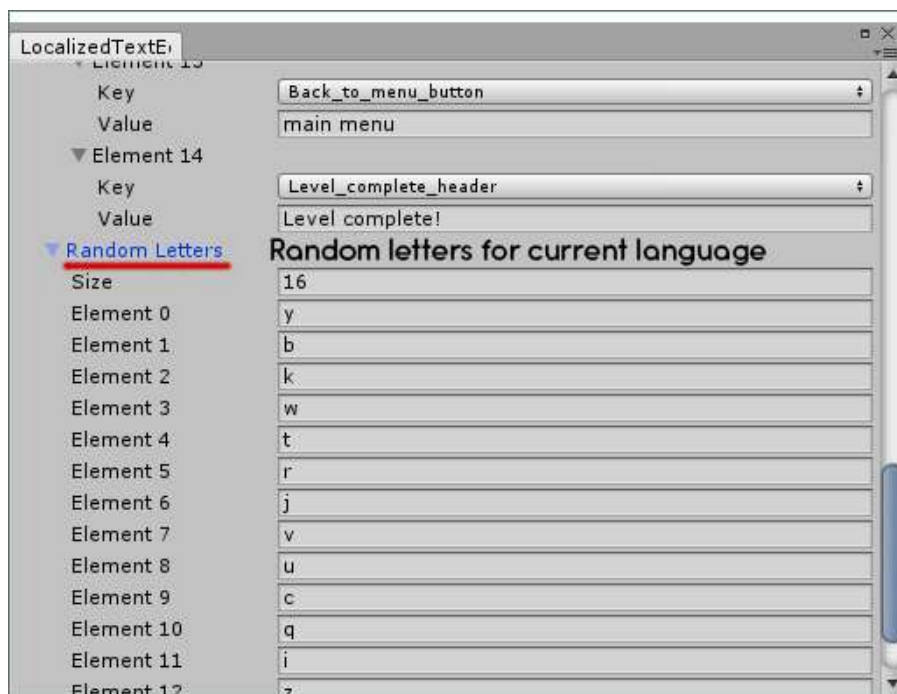
Fill up Game Items with needed localized text values for different game elements.

Text elements in the game  
 with LocalizedText script  
 attached  
 And their values

Type of the element  
 and its text



Then add random letters that are going to be implicated.



5. When you finished DON'T FORGET TO CLICK SAVE DATA. Save the data to the json file in StreamingAssets folder.

### IMPORTANT RULES !!!

- ▷ For each language you want you should assemble new file.
  - ▷ Each file with localized data should be placed in StreamingAssets folder which is located in Assets root folder. It allows the game to store this data separately after build.
  - ▷ The name of the file should match the language name as you want to see it on the language button in the game (english, deutsch etc..)
  - ▷ If you only have one localization the language button not going to appear, but the file with data should be stored in StreamingAssets still.
  - ▷ The flag image file should be stored in the PictureQuizPlus>Resources folder with the filename that matches the name of the language data file (for example english.jpg for english.json)
  - ▷ Be sure that all your languages are represented right in the DataManager.Languages array, that can be founded in the Inspector when PersistentManagers are chosen. This array is automatically filling up when you Save or Load json files.
- If you did everything right you should not face any problems with languages. The system automatically prompts the user to select a language and displays the buttons.

- Next set up global game preferences in the PersistentManagers > DataManager that can be found in the Hierarchy tab of the Editor

**Colors for levels environment** — Colors

**Get an answer hint cost (on its base all other prices are calculated)** — How Much Letters On Board: 14

**What is pixel grid should be for the pictures transition stages** — Pixelate Game Type

- Pixelate First: 17
- Pixelate Second: 22
- Pixelate Third: 27
- Final Image: 50

**Animation speed of the transition** — Animation Speed: 4

**How much coins costs one transition** — Pixelate Cost: 40

**Erasing following speed** — Erasure Game Type

- Pen Speed: 0.2
- Pen: eraser\_pen
- Erasure Cost: 5

**Planks Game Type**

- Grid Size: 5
- Aim Speed: 6
- Is Moving Aim Type: ☒
- Plank Cost: 20

**Level Locking**

- Enable Level Locking: ☒
- Unlocked At Start: 1
- Unlock Step: 1

**Ads Settings** — AdsSettings (AdsIapSettings)

**Language Popup** — LanguagePopup

**Annotations:**

- Uncheck this if you dont want the aim to be in the game. — Is Moving Aim Type
- User can click the planks to remove them — Is Moving Aim Type
- Should level locking be enabled — Enable Level Locking
- How much directories are unlocked at start — Unlocked At Start
- How much directories should be unlocked after locked ones complete — Unlock Step

**Rate App Popup**

Is Rate Popup Needed ☒

After Eeach Level

App Url

Remove Clear Buttons ☐

▼ Languages

Size

Element 0

In v.1.2 rate popup and ability to remove “clear” buttons are implemented.

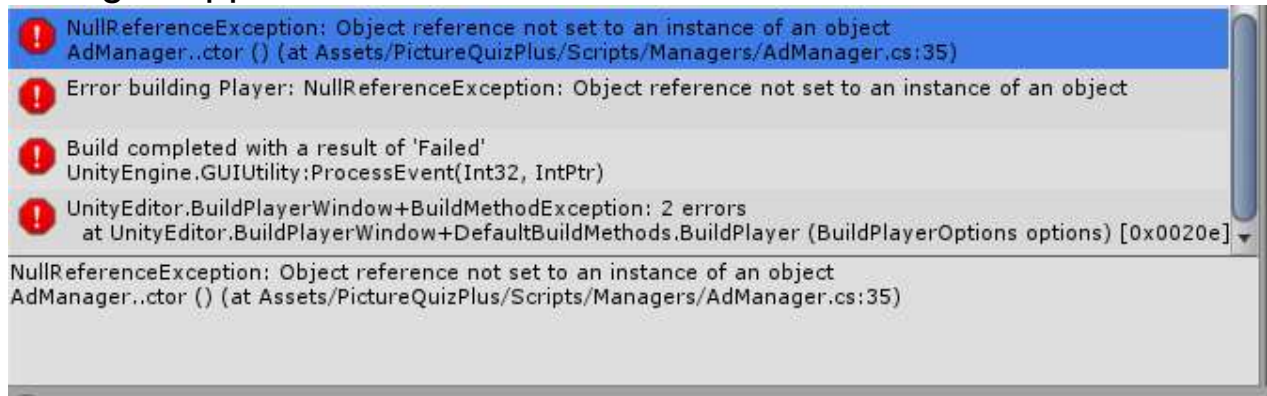
Also you can place the cursor on each element to get hint in the Inspector.

7. Now you can test all your settings in the game. Use **Tools > Clear Saved Data** in case If you need to return the game state to the first launch. Also use this function when you have made changes to the any text data.



# KNOW ISSUES

When Build you can face this error. It is an Editor issue with Scriptable Objects. It is happening when the scene is manually or automatically saved and no more changes applied before Build.



When it happens, to prevent this, you should run the game in the Unity Editor for few seconds before building apk.

We are still searching for solution.