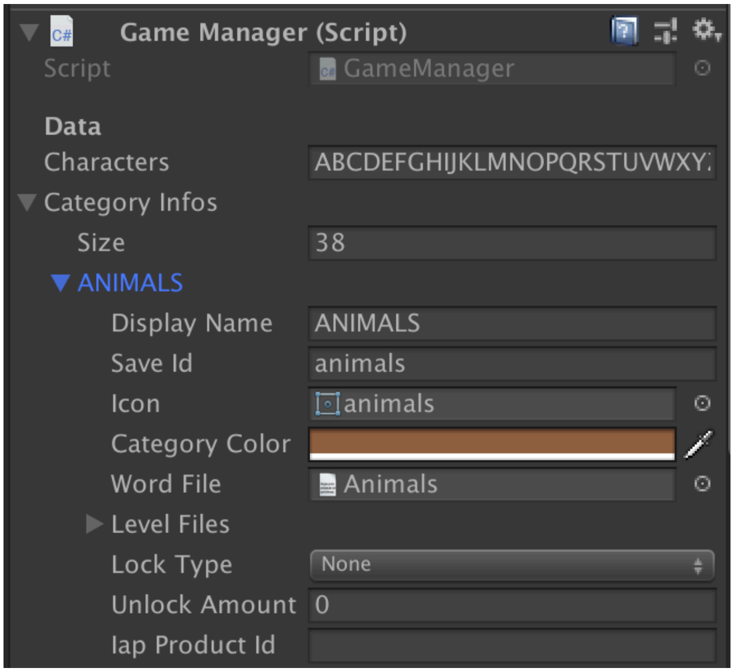
Word Search Documentation

Project

Adding / Editing Categories

The Category Infos list on the GameManagers inspector is used to add / remove / edit categories used in the game:



Game Manager (Script)

Script: GameManager

Data

Characters: ABCDEFGHIJKLMNOPQRSTUVWXY

Category Infos

Size: 38

ANIMALS

Display Name: ANIMALS

Save Id: animals

Icon: animals

Category Color

Word File: Animals

Level Files

Lock Type: None

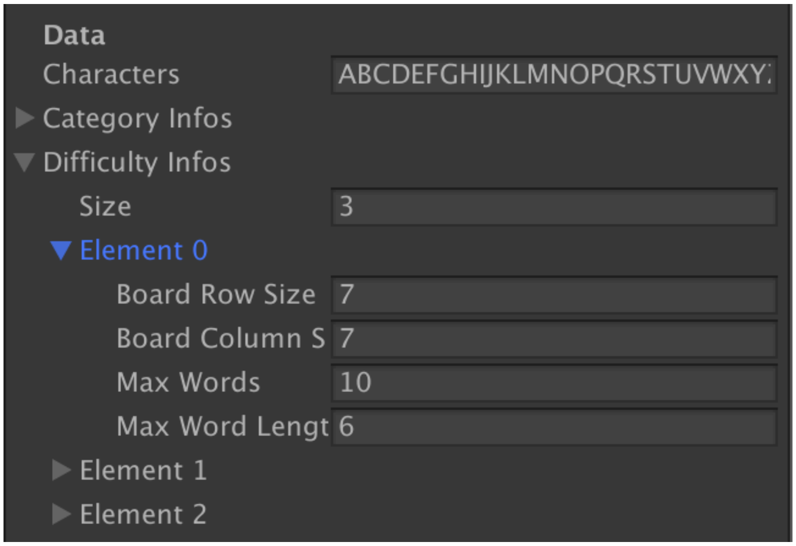
Unlock Amount: 0

Iap Product Id:

Display Name	The name that is displayed on the screen for this category
Save Id	Used in the save data to determine level progress and save boards. THIS MUST BE UNIQUE FOR EACH CATEGORY.
Icon	The icon that is displayed for this category
Category Color	The color that is used for the category list item
Word File	The text file that contains all the words to use when generating Casual game mode boards. Each word must be on a newline.
Level Files	The list of level files to use in the Progress game mode. The level files are generated using the Level Creator Window.
Lock Type	If set to None the level is not locked, if set to Coins the player must spend coins to unlock the level, if set to Keys the player must spend keys to unlock the level, if set to IAP the player must purchase the IAP product to unlock it.
Unlock Amount	The amount of coins or keys the player must spend to unlock the category if Lock Type is set to Coins or Keys.
Iap Product Id	The IAP Product Id for this category if LockType is set to IAP. This product id must appear in the IAP Settings window.

Casual Mode Difficulties

Casual modes Easy, Medium, and Hard difficulties are set using the Difficulty Infos list on the GameManagers inspector.



Data

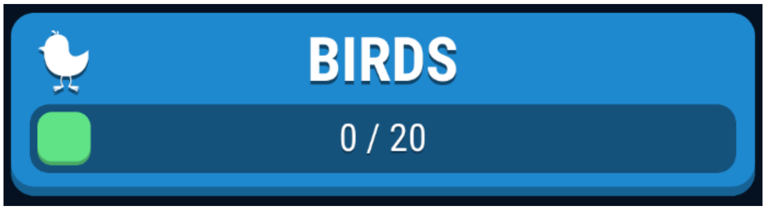
Characters	ABCDEFGHIJKLMNOPQRSTUVWXY
▶ Category Infos	
▼ Difficulty Infos	
Size	3
▼ Element 0	
Board Row Size	7
Board Column S	7
Max Words	10
Max Word Lengt	6
▶ Element 1	
▶ Element 2	

These settings are used when generating a random board to be used in the casual mode.

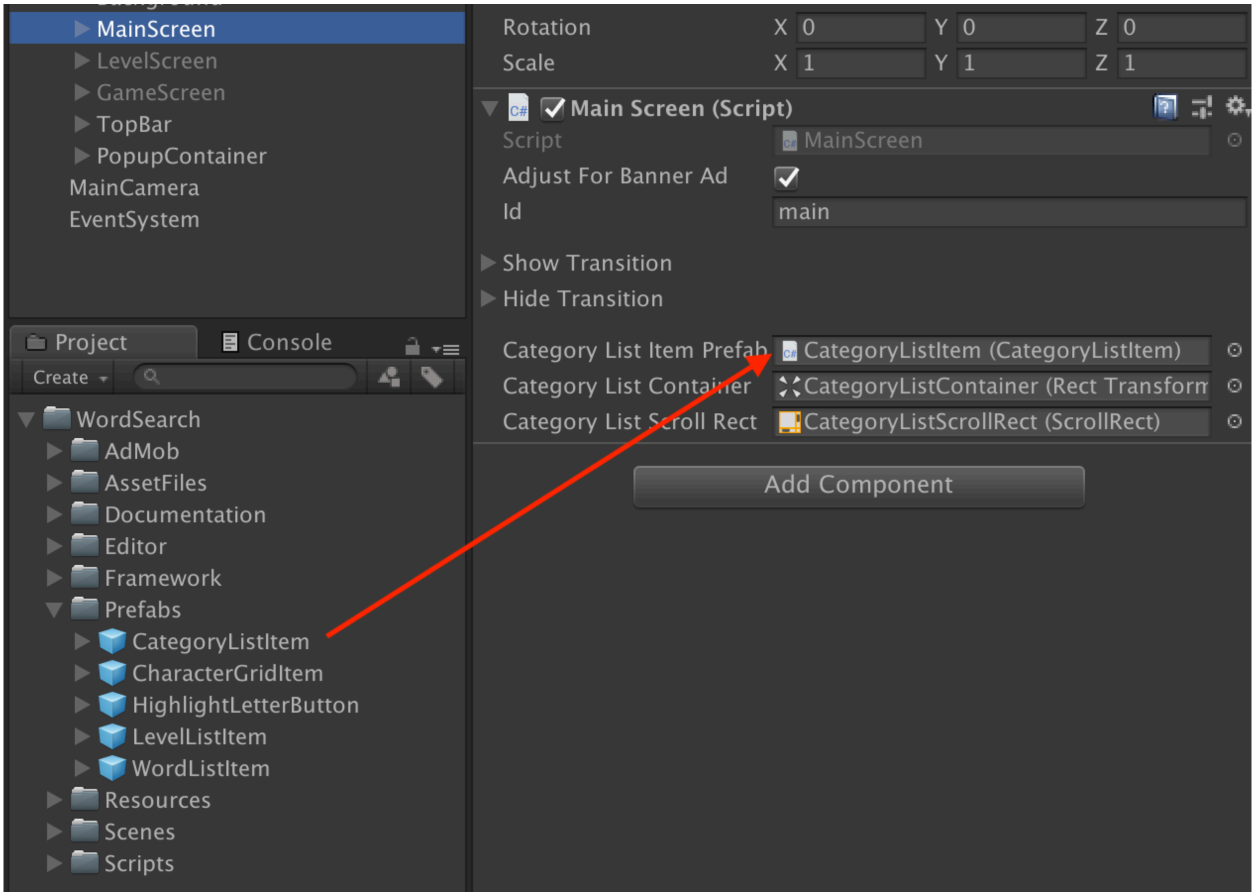
Other GameManager Settings

Characters	The letters to use when generating a random board in the casual game mode. After placing words on the board, empty spaces are filled randomly using the characters in this string.
Starting Coins	The amount of coins a new player starts with.
Starting Keys	The amount of keys a new player stars with.
Num Levels To Award Coins	The number of levels a player must completed in a category to be awarded some coins.
Coins To Award	The amount of coins to award.
Coin Cost Word Hint	The amount of coins it costs to use the HIGHLIGHT WORD hint.
Coin Cost Letter Hint	The amount of coins it costs to use the HIGHLIGHT LETTER hint.
Num Levels Between Ads	The number of levels that must be started by the player before a new interstitial ad will appear.
Character Grid	A reference to the CharcterGrid component used in the game.
Word List	A reference to the WordList component used in the game.
Loading Indicator	The GameObject that is set to active when a new casual board is generating.

Category List Items



The category list items that appear on the MainSceren are instantiated by the **MainScreen** script at run time. The prefabs that is used is called **CategoryListItem** and is located in the Prefabs folder.



Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

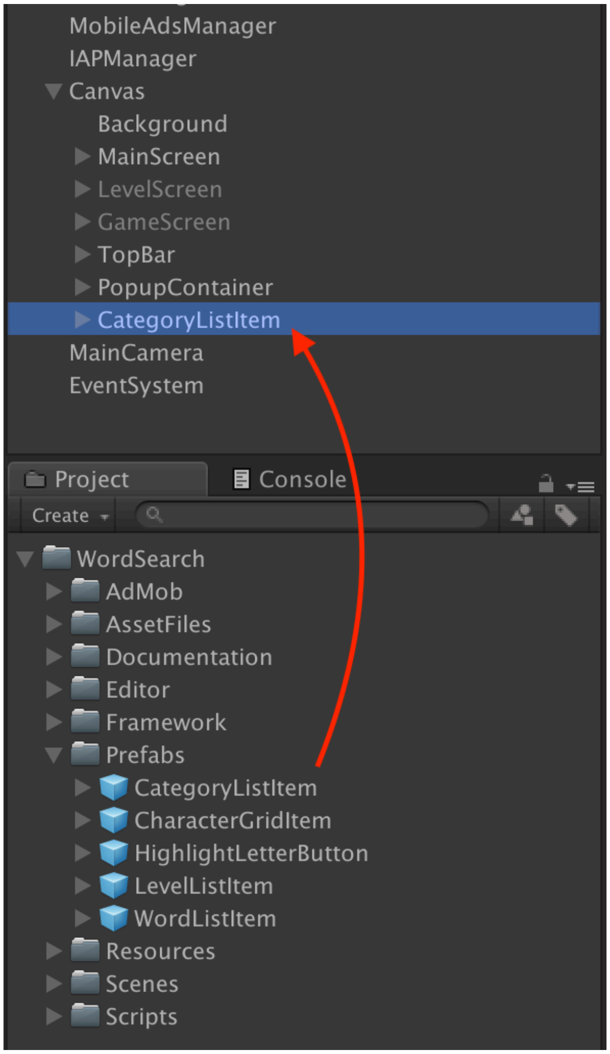
☑ Main Screen (Script)

Script: MainScreen
Adjust For Banner Ad: ☑
Id: main

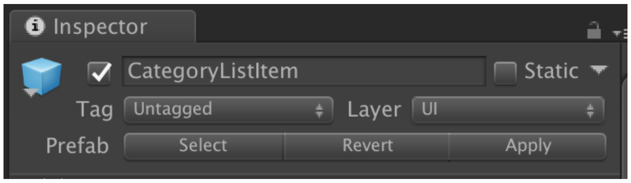
Show Transition
Hide Transition

Category List Item Prefab: CategoryListItem (CategoryListItem)
Category List Container: CategoryListContainer (Rect Transform
Category List Scroll Rect: CategoryListScrollRect (ScrollRect)

To edit the prefab, drag into the scenes hierarchy under the Canvas object.

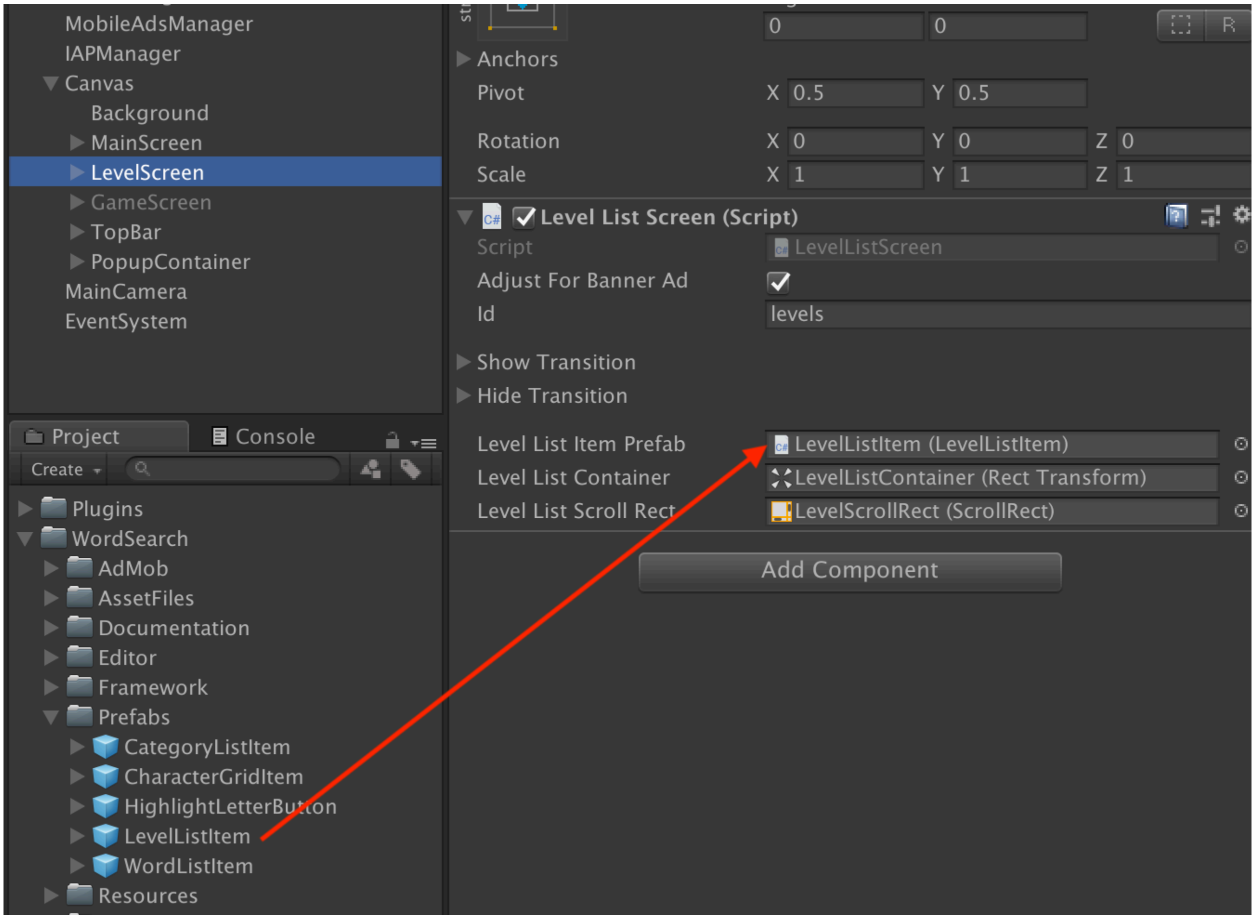


After you finished editing it click the Apply button at the top of the Inspector then delete it from the scene.



Level List Items

The level list items that appear on the LevelScreen are instantiated by the **LevelScreen** script at run time. The prefabs that is used is called **LevelListItem** and is located in the Prefabs folder.



0 0

Pivot X 0.5 Y 0.5

Rotation X 0 Y 0 Z 0

Scale X 1 Y 1 Z 1

☑ Level List Screen (Script)

Script: LevelListScreen

Adjust For Banner Ad: ☑

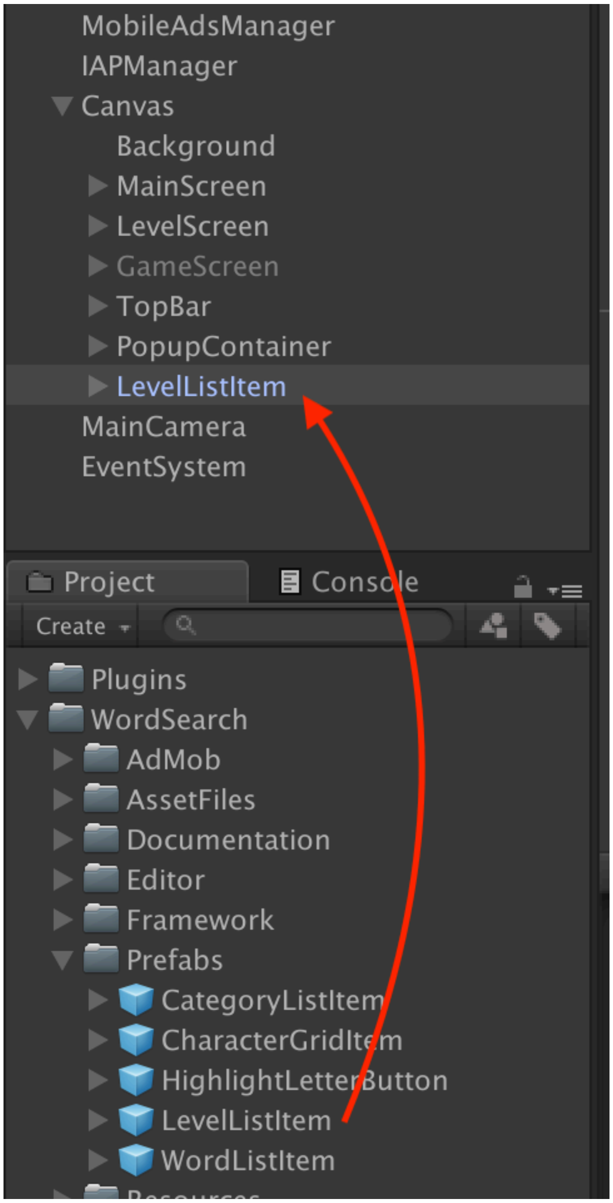
Id: levels

Level List Item Prefab: LevelListItem (LevelListItem)

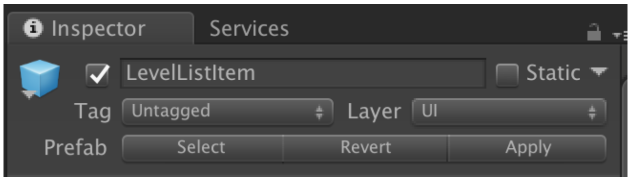
Level List Container: LevelListContainer (Rect Transform)

Level List Scroll Rect: LevelScrollRect (ScrollRect)

To edit the prefab, drag into the scenes hierarchy under the Canvas object.



After you finished editing it click the Apply button at the top of the Inspector then delete it from the scene.



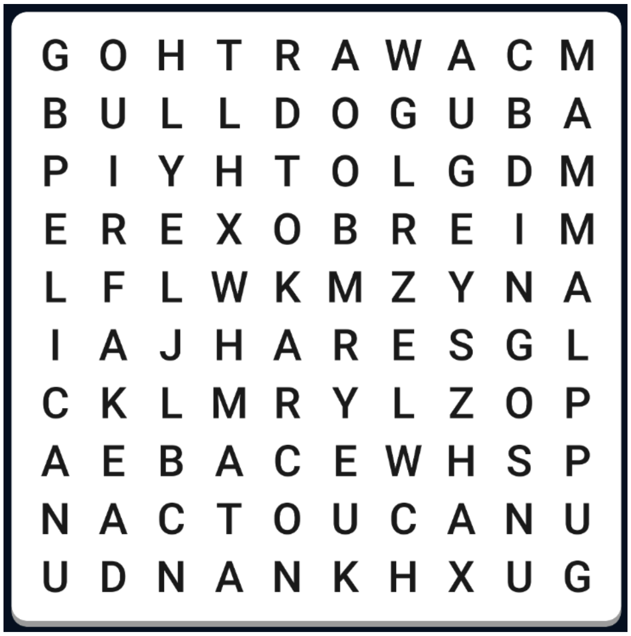
☑ LevelListItem

Tag Untagged

Layer UI

Prefab

Character Grid

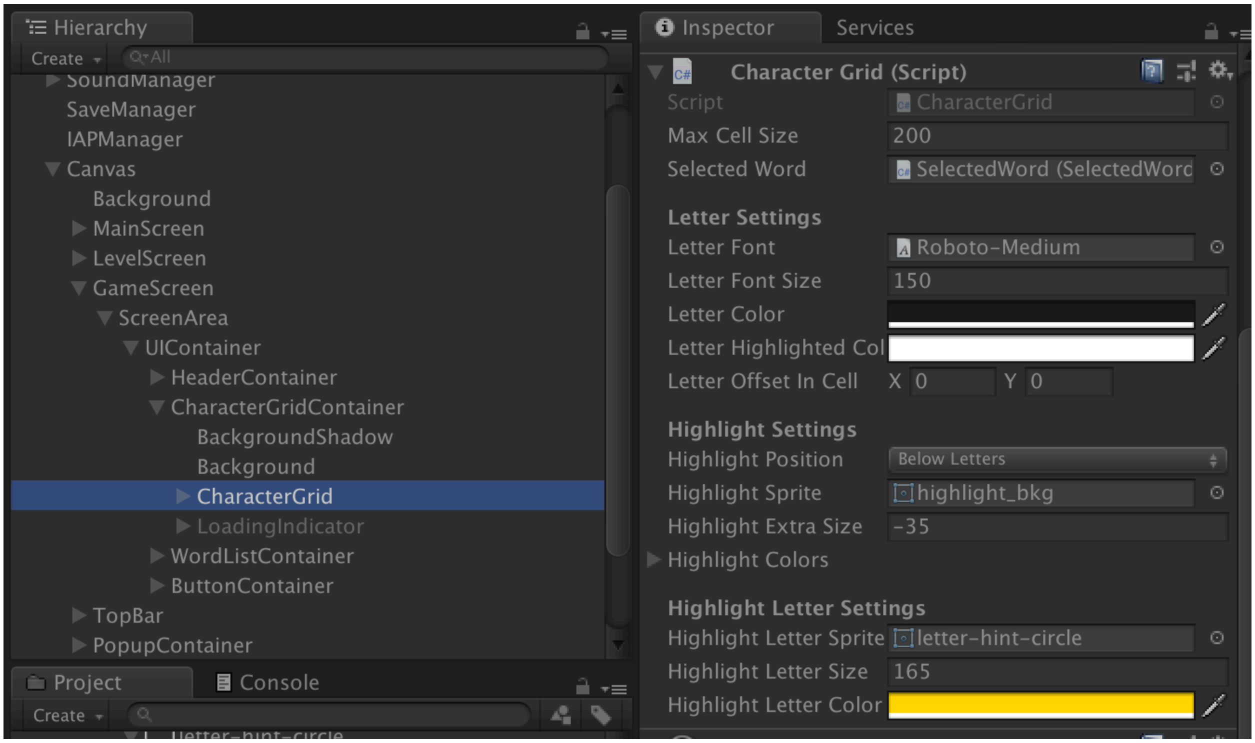


```
G O H T R A W A C M
B U L L D O G U B A
P I Y H T O L G D M
E R E X O B R E I M
L F L W K M Z Y N A
I A J H A R E S G L
C K L M R Y L Z O P
A E B A C E W H S P
N A C T O U C A N U
U D N A N K H X U G
```

The CharacterGrid script is responsible for creating the letters you see on the screen when a game starts. It is also responsible for highlighting words when the user swipes across the board.

The CharacterGrid GameObject has the CharacterGrid component attached to it and this is where you can change the look of the letters and highlights.

To change the font, size, color, ect of the letters select the CharacterGrid object in the scenes hierarchy:



```
▶ SoundManager
  SaveManager
  IAPManager
▼ Canvas
    Background
  ▶ MainScreen
  ▶ LevelScreen
  ▼ GameScreen
    ▼ ScreenArea
      ▼ UIContainer
        ▶ HeaderContainer
        ▼ CharacterGridContainer
            BackgroundShadow
            Background
          ▶ CharacterGrid
          ▶ LoadingIndicator
        ▶ WordListContainer
        ▶ ButtonContainer
  ▶ TopBar
  ▶ PopupContainer
```

letter-hint-circle

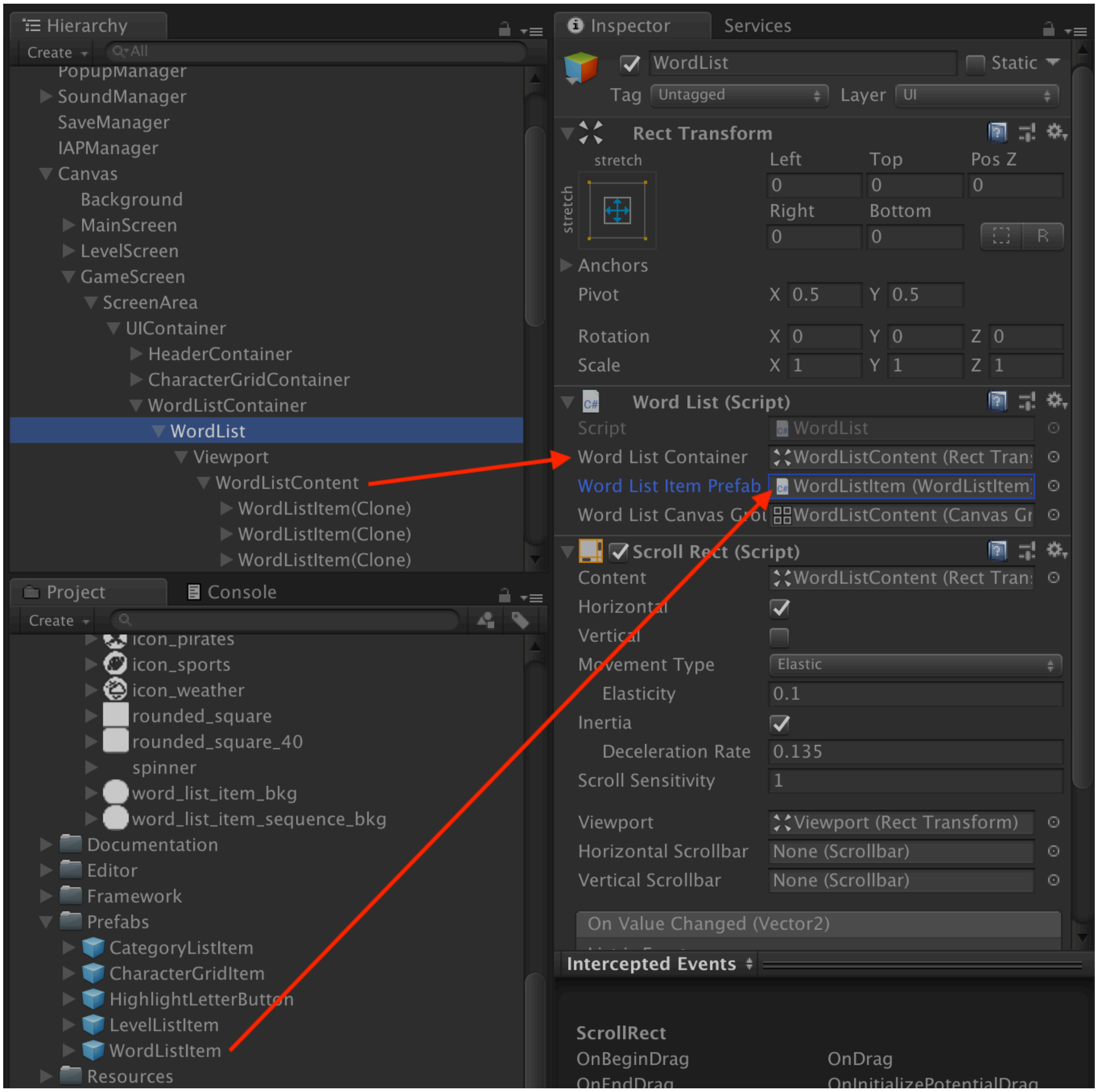
Character Grid (Script)

Script	CharacterGrid
Max Cell Size	200
Selected Word	SelectedWord (SelectedWord
Letter Settings	
Letter Font	Roboto-Medium
Letter Font Size	150
Letter Color	⬛
Letter Highlighted Col	⬜
Letter Offset In Cell	X 0 Y 0
Highlight Settings	
Highlight Position	Below Letters
Highlight Sprite	highlight_bkg
Highlight Extra Size	-35
▶ Highlight Colors	
Highlight Letter Settings	
Highlight Letter Sprite	letter-hint-circle
Highlight Letter Size	165
Highlight Letter Color	🟨

Max Cell Size	A cell on the character grid will not scale larger than this size. This is to keep boards with few rows/columns from having giant letters.
Selected Word	A reference to the SelectedWord component that will be updated every time the user selects different letters on the board.
Letter Font	The font to use for the letters.
Letter Font Size	The font size to use for the Text components that are created for each letter.
Letter Color	The color to use for the letters Text component.
Letter Highlighted Color	The color to use when a letter is highlighted
Letter Offset In Cell	An offset which is applied to the letter Text object in each cell.
Highlight Position	Sets the position of the highlights, either on top of the letters or below the letters.
Highlight Sprite	The sprite to use for the highlights Image component.
Highlight Extra Size	Extra size to apply to the highlight RectTransform.
Highlight Colors	The colors to use for highlights, a random color is pick from this list every time the user highlights letters.
Highlight Letter Sprite	The sprite to use for HIGHLIGHT LETTER hints.
Highlight Letter Size	The size of the letter hint Image object.
Highlight Letter Color	The color of the letter hint Image object.

Word List

The WordList script is responsible for taking the list or words on the board and generating a list of all those words. The **Word List Item Prefab** found in the Prefabs folder is used to instantiate copies which are then placed in the **Word List Container**



☑ WordList □ Static

Tag Untagged Layer UI

Rect Transform

stretch

Left	Top	Pos Z
0	0	0
Right	Bottom	
0	0	

Anchors

Pivot X 0.5 Y 0.5

Rotation X 0 Y 0 Z 0

Scale X 1 Y 1 Z 1

Word List (Script)

Script: WordList

Word List Container: WordListContent (Rect Tran:

Word List Item Prefab: WordListItem (WordListItem)

Word List Canvas Grou: WordListContent (Canvas Gr

☑ Scroll Rect (Script)

Content: WordListContent (Rect Tran:

Horizontal: ☑

Vertical: □

Movement Type: Elastic

Elasticity: 0.1

Inertia: ☑

Deceleration Rate: 0.135

Scroll Sensitivity: 1

Viewport: Viewport (Rect Transform)

Horizontal Scrollbar: None (Scrollbar)

Vertical Scrollbar: None (Scrollbar)

On Value Changed (Vector2)

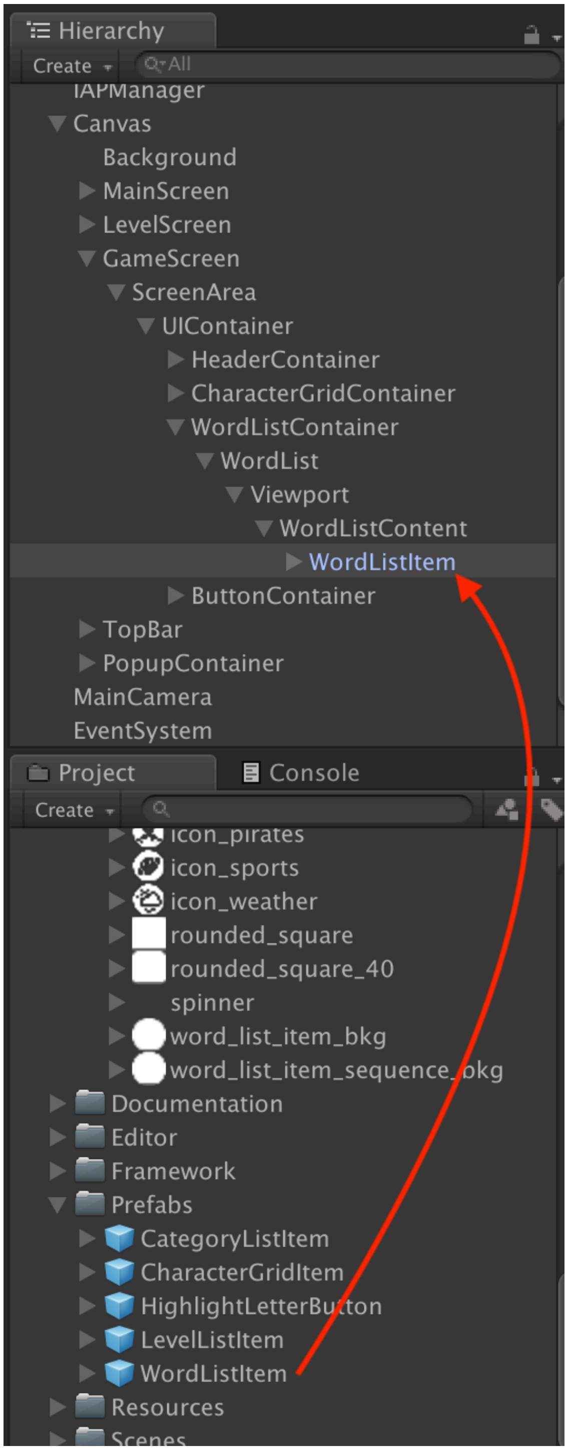
Intercepted Events

ScrollRect

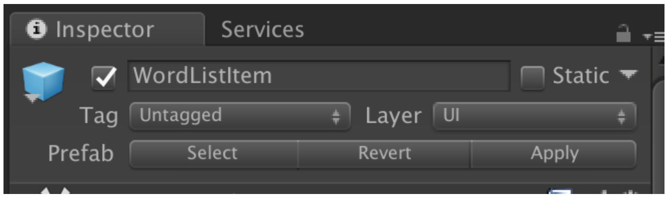
OnBeginDrag OnDrag

OnEndDrag OnInitializePotentialDrag

To edit the WordListPrefab prefab, drag into the scenes hierarchy under the **WordListContent** object.



After you finished editing it click the Apply button at the top of the Inspector then delete it from the scene.

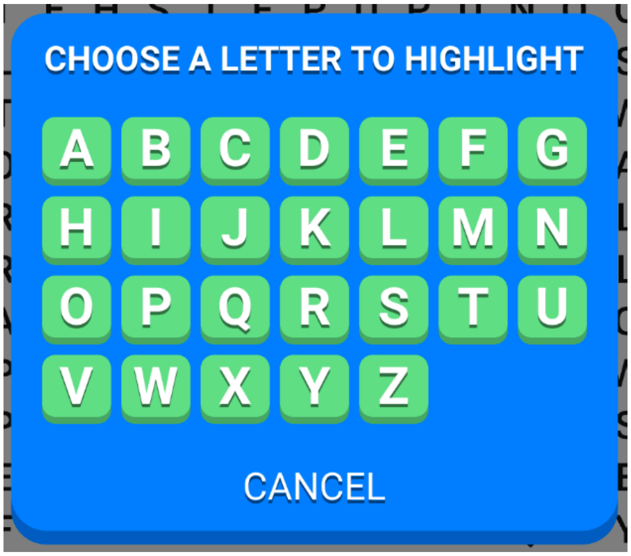


☑ WordListItem □ Static

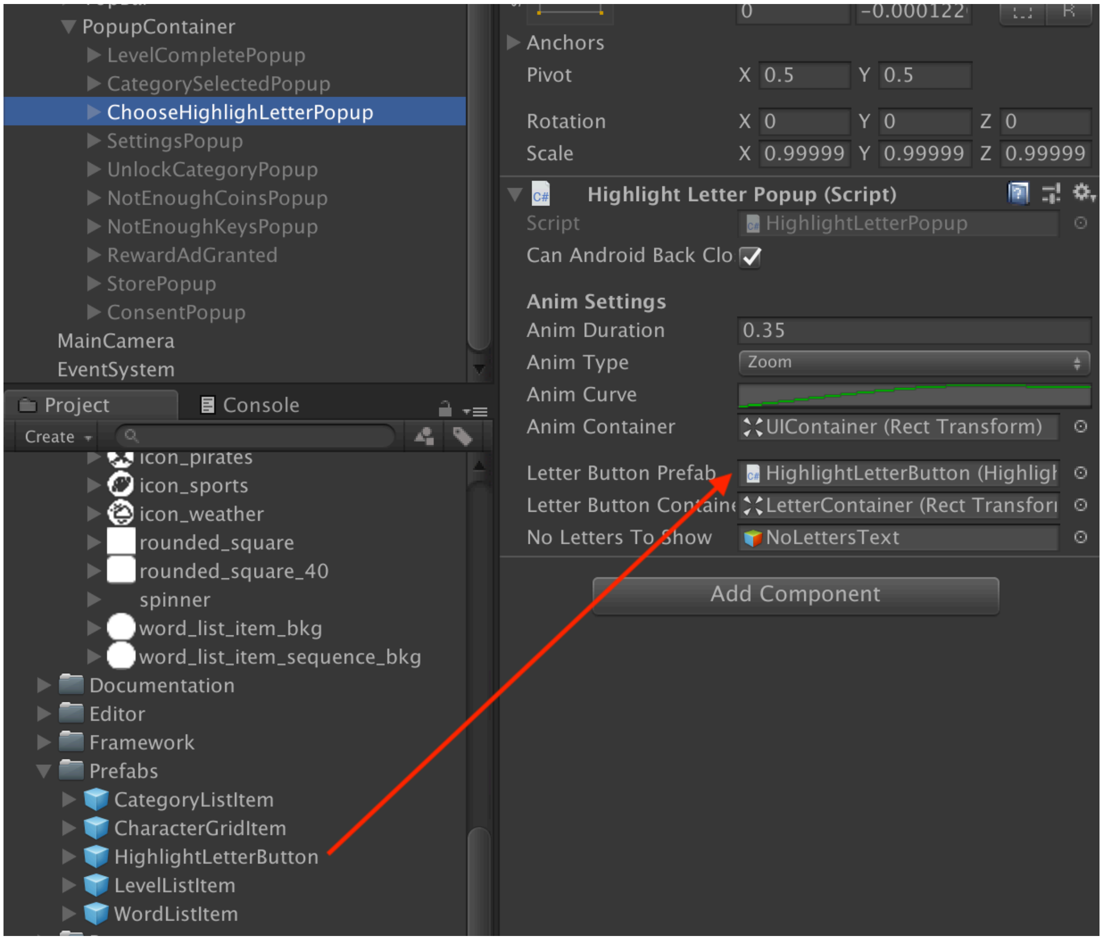
Tag Untagged Layer UI

Prefab

Highlight Letter Buttons



The letter buttons that appear on the “Choose a letter to highlight” popup are instantiated by the ChooseHighlightLetterPopup. The prefab that is used is called HighlightLetterButton and is located in the Prefabs folder.



Anchors

Pivot X 0.5 Y 0.5

Rotation X 0 Y 0 Z 0

Scale X 0.99999 Y 0.99999 Z 0.99999

Highlight Letter Popup (Script)

Script: HighlightLetterPopup

Can Android Back Clo ☑

Anim Settings

Anim Duration: 0.35

Anim Type: Zoom

Anim Curve

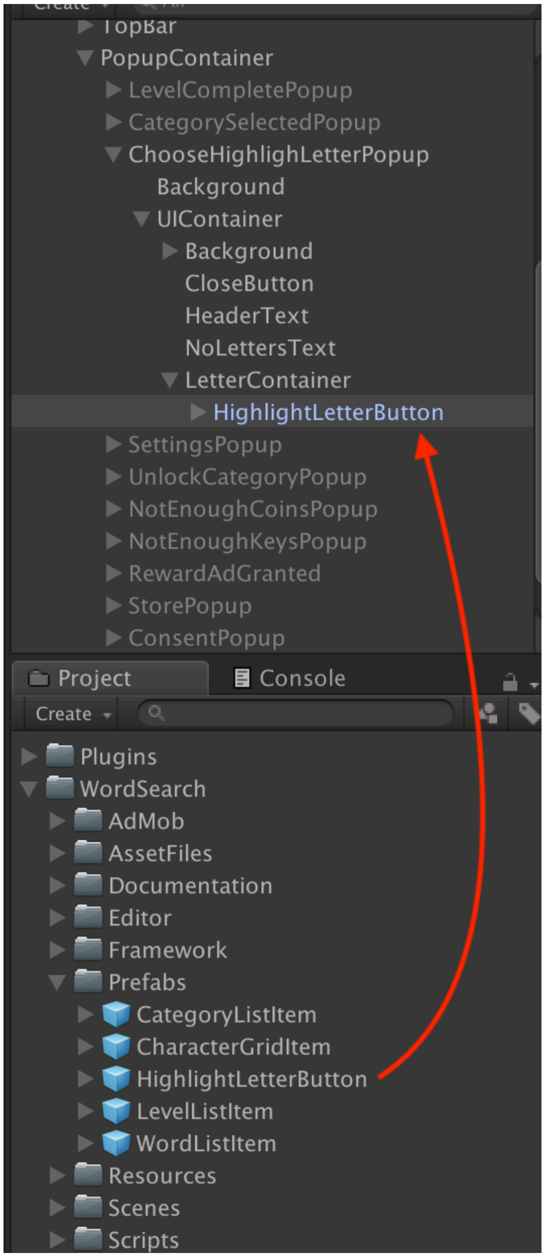
Anim Container: UIContainer (Rect Transform)

Letter Button Prefab: HighlightLetterButton (Highligh

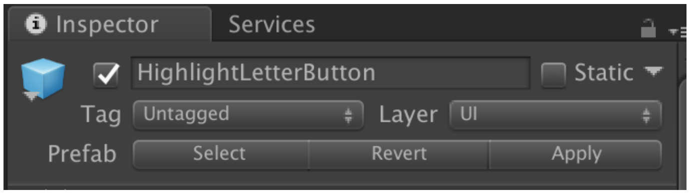
Letter Button Containe: LetterContainer (Rect Transfori

No Letters To Show: NoLettersText

To edit the prefab, drag into the scenes hierarchy under the **LetterContainer** object.



After you finished editing it click the Apply button at the top of the Inspector then delete it from the scene.



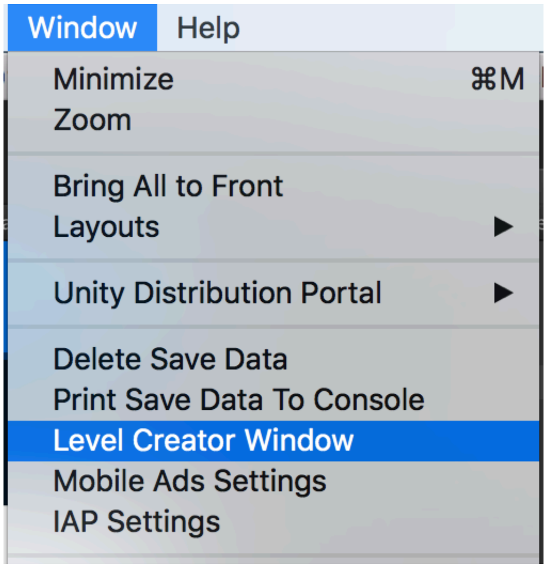
☑ HighlightLetterButton

Tag Untagged

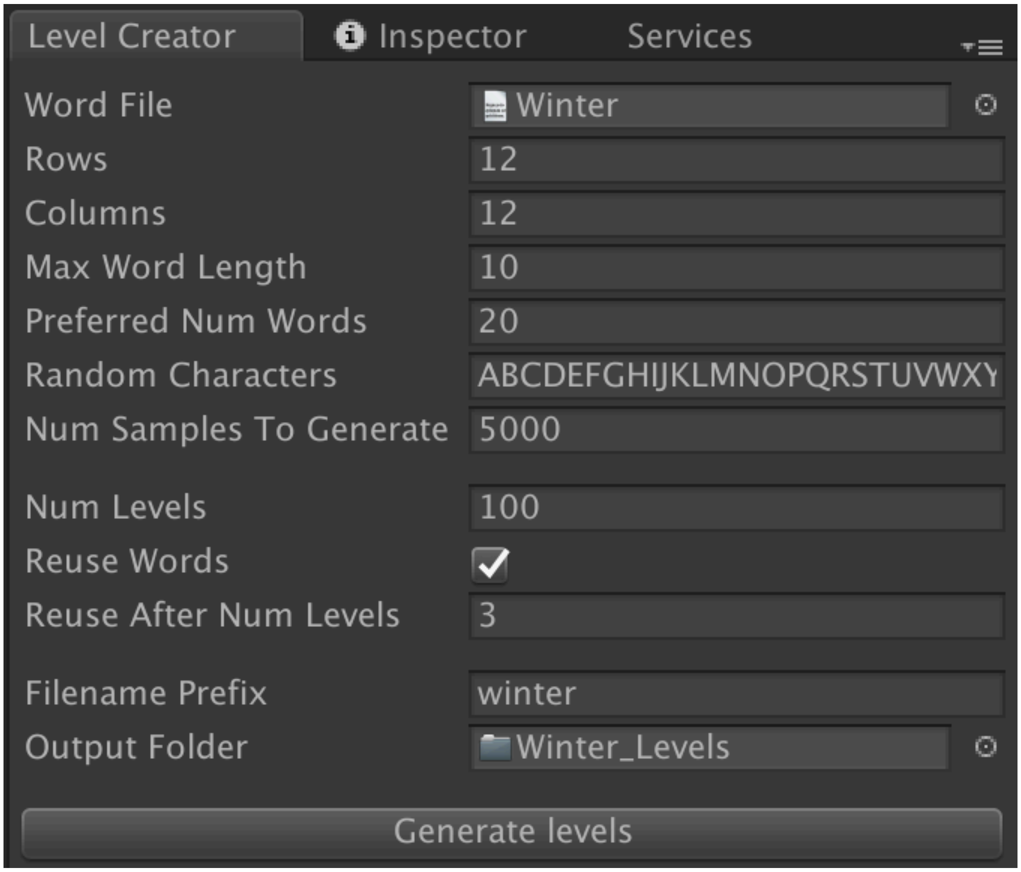
Layer UI

Creating Levels

To create new levels for Progress mode, open the Level Creator window by selecting the menu item **Window -> Level Creator Window**



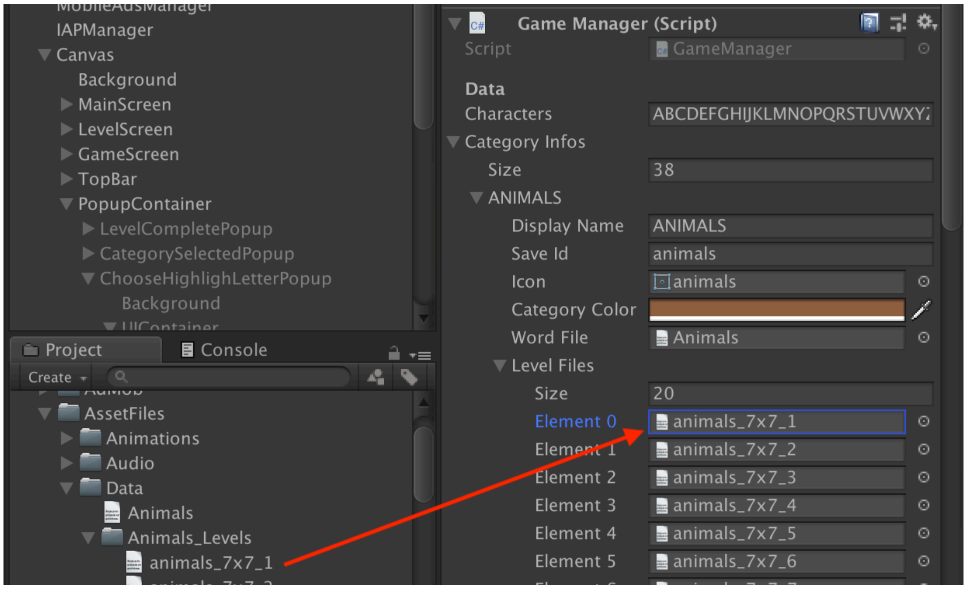
The Level Creator window can generate multiple levels at once using the settings.



Word File	Winter
Rows	12
Columns	12
Max Word Length	10
Preferred Num Words	20
Random Characters	ABCDEFGHIJKLMNOPQRSTUVWXY
Num Samples To Generate	5000
Num Levels	100
Reuse Words	☑
Reuse After Num Levels	3
Filename Prefix	winter
Output Folder	Winter_Levels

Word File	The text file that contains all the words which can be used in the levels that you are about to generate. Each word should be on a newline.
Rows	The number of letter rows on the board.
Columns	The number of letter columns on the board.
Max Word Length	The maximum length a word can be. All words in the Word File which are longer than this will not be used.
Preferred Num Words	The preferred number of words to place on the board. The algorithm will get as close to this number as possible but it wont always place this number of words on the board.
Random Characters	The random characters to use to fill in blank spaces on the board after all words have been placed.
Num Samples To Generate	The number of boards to generate for each level. It will pick the best board out of all the generated board (EI. the board with the most words placed on it)
Num Levels	The number of levels to generate.
Reuse Words	If selected, words will be reused in other levels. If not selected then words will not be reuse, once they are used in a level they will not be considered for other levels.
Reuse After Num Levels	The number of levels that must be generated before used words are placed back in the pool of words that can be selected for the next level.
Filename Prefix	A prefix to apply to all level files.
Output Folder	The folder from the Project window where generated level files will be placed.

After the level files have been generated you can add them to the Category Infos **Level Files** list:



Script: GameManager

Data

Characters: ABCDEFGHIJKLMNOPQRSTUVWXYZ

Category Infos

Size: 38

ANIMALS

Display Name: ANIMALS

Save Id: animals

Icon: animals

Category Color

Word File: Animals

Level Files

Size: 20

Element 0: animals_7x7_1

Element 1: animals_7x7_2

Element 2: animals_7x7_3

Element 3: animals_7x7_4

Element 4: animals_7x7_5

Element 5: animals_7x7_6

Sounds

Sounds in the game are controlled using the SoundManager. On the SoundManager's inspector you will find a number of Sounds Infos already created and used in the game.

Sound Info fields

Id - The Id used to play the sound in the game.
Audio Clip - The sound file from your project.
Type - The type of sound (Sound Effect or Music), this is used to turn on/off all sounds of a particular type.
Play And Loop On Start - If selected the Audio Clip will play when the game starts and will loop forever unless it is stopped.
Clip Volume - Sets the volume of the sound when it is played.

Playing Sounds

Sounds can be played by calling the **Play** method on the SoundManager like so:

```
SoundManager.Instance.Play(string id);
```

You can easily play a sound when a Button is clicked by adding the **ButtonSound** component to a GameObject with a **Button** component. The ButtonSound will play the sound with the specified Id every time the button is clicked.