

Scrambled Word Game

Game documentation and HowTo guide.



This document contains:

Package Description and features	2
Try the Webplayer	2
Try the WebGL	2
Update history	2
Credits	3
Overview of the game's library contents	4
Customization Guide	4
Getting started	4
The Game Controller	5
The Words List	7
How letters are organized	8
Editing the default letter object	9
UnityAds Integration	10
Frequently Asked Questions	12
Does this package work on mobile?	12
My sprites are not showing on iOS	12
How to change font in the game?	12
More games by Puppeteer	14

Package Description and features

Scrambled Word Game is a full Unity template that gives you a quick and easy way to create a scrambled word game, along with a timer, levels, varying bonuses, and some nice audio-visual feedback.

How to Play?

Click on one of the letters and then click on another letter to swap them. Clear the all pairs to advance to the next level. Reorganize the word to its correct form to win!

[Try the Webplayer](#)

[Try the WebGL](#)

Features:

- Game ready for release straight out of the box, just build and play!
- Works on all platforms, PC, Mac, iOS, Android, etc
- Supports multiple resolutions and aspect ratios, automatically.
- Supports Mouse, Keyboard, Gamepad, and Touch controls.
- Easily customizable with lots of options to control game difficulty.
- Great learning resource with commented scripts and documentation.
- All assets included: graphics, sounds, and code.

Current version 1.2

Update history

1.2 (28.06.2017)

- You can choose to mix whole sentences or individual letters, regardless of whether you have a sentence in the list or not.
- No need to set the letter increase number. The next word/sentence length is found automatically when leveling up.
- Updated to latest Unity versions.

1.13 (24.10.2016)

- Fixed a case in which words appear correct from the start, then you have to mix them and organize them again to win. Now this doesn't happen.
- Added an outro animation to the hint objects so now they transition more smoothly.
- Improved gamepad/keyboard support.
- Minor changes.

1.11 (18.09.2016)

- A new game mode allows you to have full sentences in the word list. The words are mixed instead of letters.
- Automatically look for the shortest word/sentence in the list, so there is no need to manually set the number of letters words we are looking for in level 1.
- Better switch transition between tiles in the grid, in all directions.
- Solved button Select/Deselect issue, and the problem of letters locking Z position when selected while moving towards the center of the screen.
- Solved input bug on mobile devices that made buttons un-clickable.
- Corrected words count for each level. Previously it would count 1 extra.

1.04 (22.08.2016)

- The words list are now separated from the main gamecontroller, which allows you to set word list more easily, and assign lists to several game levels.
- Added a new words list with hints. This list allows you to add image and text hints to each word, and either display it when the word appears or when you press a hint button, which could also cost you some bonus points.
- You can now assign a start screen that will appear before the game starts. Use this to show instructions or a description for the game before playing.
- Fixed a bug where buttons didn't work on handheld devices.
- Fixed a bug where the game would end if there were not enough words with a certain letter count in the list, instead of finding a word with more letters.
- Minor changes, removed extra assets and added some more animations.

1.0 (22.08.2016)

- Initial version

Credits

The main font used is [Fava Black by Themnific](#)

The sounds are courtesy of [the free sound project](#).

Music is Waterford by Kevin MacLeod (Public Domain)

Credits go to these authors for their great sound samples: **xyzz-kx**, **isaac200000**, **harris85**, **speedygonzo**, **wagna**, **jimhancock**, **sforzman**

Please rate my file, I'd appreciate it 😊

Overview of the game's library contents

Let's take a look inside the game files. Open the main SWGAssets folder using Unity3D 5.3.0 or newer. Take a look at the project library, usually placed on the right or bottom side of the screen. Here are the various folders inside:

- **Animations:** Holds the animation clips made with Unity's built-in animation system.
- **FLA:** Holds the object graphics made with Flash CS3. These are vector graphics that can be easily scaled without loss of quality and then exported as PNG to be used in Unity.
- **Fonts:** Holds the font used in the game.
- **Prefabs:** Holds all the prefabs used in the game. These are distributed to various folders for easier access, Buttons, Enemies, Objects, etc. It also holds all the canvases in the game which are used to hold buttons and other UI elements.
- **Scenes:** The first scene that runs in the game is MainMenu. From this scene you can get to the Game scene.
- **Scripts:** Holds all the scripts used in the game. Each prefab contains one or more of these scripts.
- **Sounds:** Holds all the sounds used in the game. Correct, Wrong, etc
- **Textures:** Holds all the textures used in the game which are used as sprites in Unity.

Customization Guide

Getting started

Scrambled Word Game (SWG) is considered a complete project, and as such is supposed to work as the starting point of your planned game, rather than an addition to an existing project. That said, you may of course pick and choose some of the scripts/models to import into your existing project, but SWG works best as a starter kit which you can customize any part of to your liking.

The Game Controller

The Game Controller is the main prefab that controls all the progress of the game from start to finish. It controls the UI of the game, creates scrambled words and checks the level up condition.



Words – (This is the old words list. In the next update it will be removed completely, so remember to move your words to one of the attached lists SWGWords or SWGWordsWithHints) A list of all the words in the game. The words are displayed from the ones with the least letters to the ones with the most letters.

You can enter any number of words in this list, and you don't have to organize it based on the letter count. Just remember to set the value of

Randomize Words – Randomize the list of words so that we don't get the same words each time we start the game.

Letters Count – The number of letters in the first word group. This means we will look for a word with X letters and display it.

Words Per Level – The number of words to display from a certain level before moving on to the next level (Each level displays words with certain number of letters such as 3 letters, 4 letters, etc).

Letters Increase – The number of letters added to the game in each level.

Bonus Per Level – How many points we get for each word in the level. This value is multiplied by the number of the level we are on. Ex: Level 1 gives 100 points, Level 2 gives 200 points.

Time Bonus Per Level – How many extra seconds we add to the timer in the level. This value is multiplied by the number of the level we are on. Ex: Level 1 gives 5 seconds at the start, Level 2 gives 10 seconds at the start.

Time – How many seconds are left before game is over.

Canvases – These UI screens are assigned from the scene for each level.

Main Menu Level Name – The level of the main menu that can be loaded after the game ends.

Confirm Button – The keyboard/gamepad button that will restart the game after game over.

Pause Button – The keyboard/gamepad button that pauses the game.

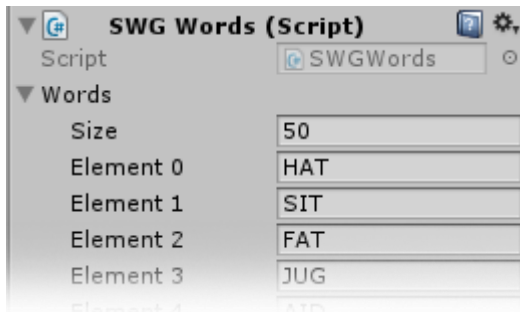
User Interface – Various canvases for the UI, assign them from the scene.

Sounds – Various sounds that play during the game.

Sound Source Tag – The audio source from which the Game Over sound plays.

The Words List

The Words list script contains all the words in the game, and must be attached to a gamecontroller in order to work. There are two types of list, the first one simply contains a list of words you can fill out.



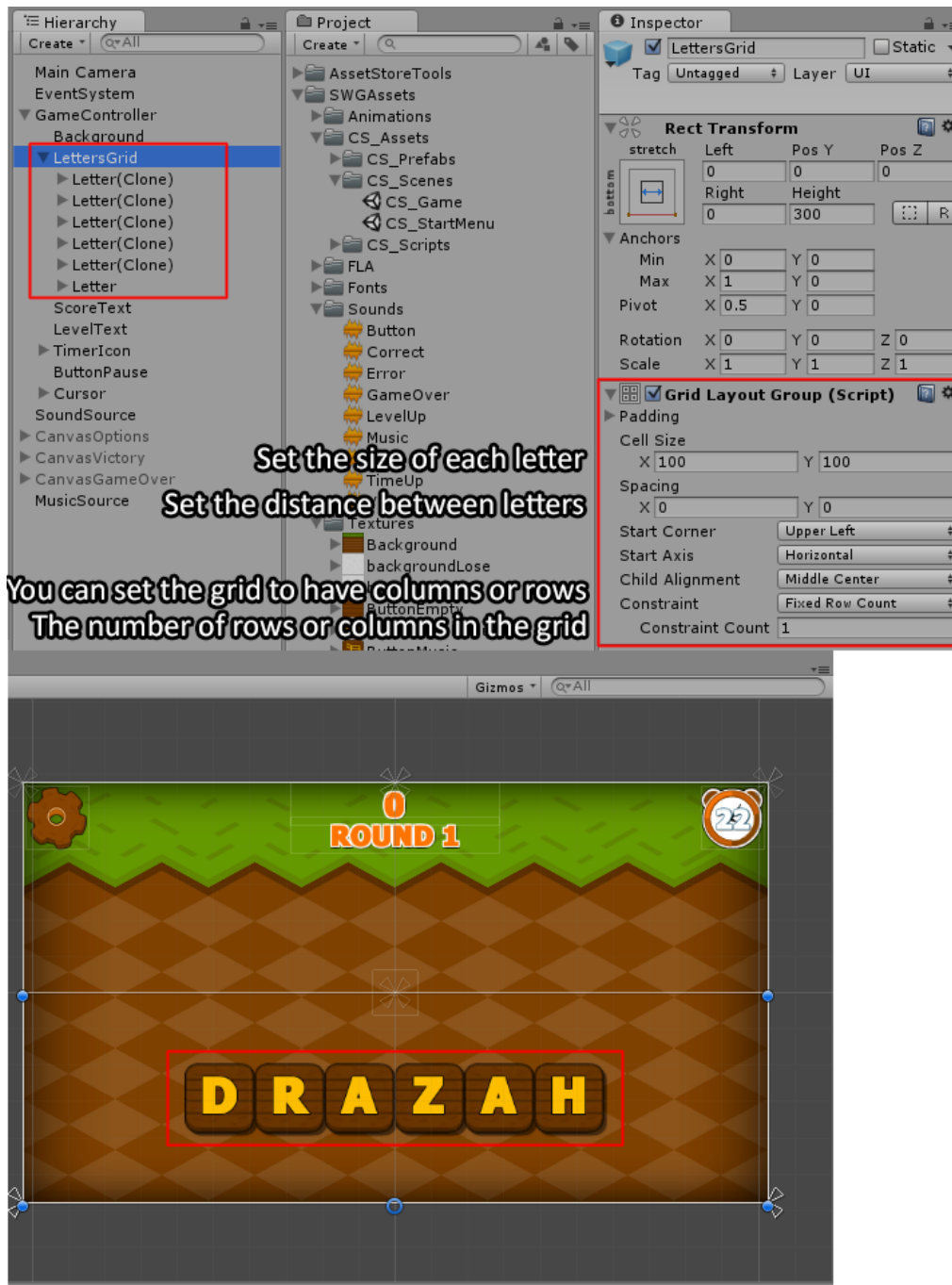
The other list is more advanced, it contains additional fields for a text hint and an image hint. It also has options to show the hint when the word is displayed, or to show a button which when pressed it shows the hint, at the cost of some bonus points. You can set all these values in the **Words With Hints** component:



Note: As of version 1.11 you can also have full sentences in the word list. When a sentence is detected in the list, the words in it are mixed instead of letters. The size of the grid tiles is the 50 multiplied by the average number of letters in a word, based on the sentence we have. **You can either have a list with Words or a list with Sentences, but not both mixed together.**

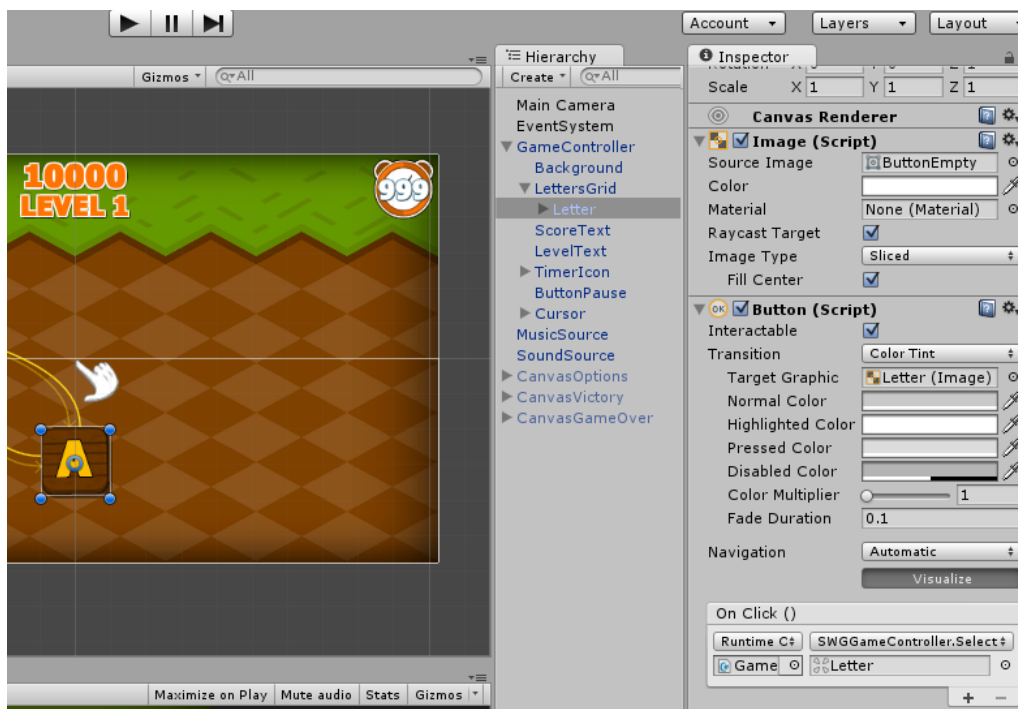
How letters are organized

Letters are organized into a letters grid, which displays them one after the other based on the values we set in the component **Grid Layout Group**. If you take a look at the **LettersGrid** you can see they have a component which organizes the objects into a nice looking grid.



Editing the default letter object

Each letter object is made up of the same base object, which contains an image (the tab behind the text), a text, and button to listen for clicks on it. To take a look at the button and change its appearance, choose it from inside the **LettersGrid** object.



You can change the background wooden tab, or the font of the text, the color and shape, and the letter will be cloned to form words during play.

UnityAds Integration

Since Unity 5.2 UnityAds integration has been simplified, here's how you can have full screen video ads in your game.

This video shows a quick process of integrating UnityAds into your project. In the example we used one of my templates, but it works on all my other templates too.

<https://www.youtube.com/watch?v=EQNTgfV35DU>

Here is what we did in the process:

1. Sign in to your Unity account in order to allow Unity Services such as UnityAds to be activated.
2. Open Build Settings and switch the platform to one of the supported ones (iOS, Android).
3. Download Puppeteer's UnityAds package from:
<http://puppeteerinteractive.com/freebies/PUPUnityAds.unitypackage>
4. Drag the downloaded package into your Unity project, and import it. This UnityAds prefab can be used to display ads every several minutes.
5. Drag the prefab into any scene where you want ads to be shown. Make sure to save changes.
6. The time check is shared between all prefabs in all scenes, so you will never show too many ads.
7. The final step is to activate UnityAds services and get your unique project ID.
8. Open the services window and choose your organization, then click create.
9. Choose UnityAds from the list and turn it On.
10. Choose age group for your project (Will affect the nature of ads shown), and save changes.

11. While working on your project keep Test Mode activated. But when you are ready to release the final project, switch Test Mode off.
12. That's it! Now when you start the game, an ad will be shown after 3 minutes. The ad will never appear during gameplay or post-game screen. Instead, it will wait until the next level load (restart, main menu, etc) and then show the ad.

Before releasing a game, make sure you uncheck **Enable Test Mode**.

For more info about integrating UnityAds read this:

<http://unityads.unity3d.com/help/monetization/integration-guide-unity>

Frequently Asked Questions

Does this package work on mobile?

Yes, this package has been successfully tested on both Android and iOS devices. The scripts for each lock type include controls for mobile that are detected automatically based on the platform it's built on.

My sprites are not showing on iOS

Sprite-based textures made with the new Unity 4.3 can sometimes disappear when working on the iOS platform.

You can notice this by opening a scene playing it. When you switch from your current platform to the iOS platform the sprite textures become invisible.

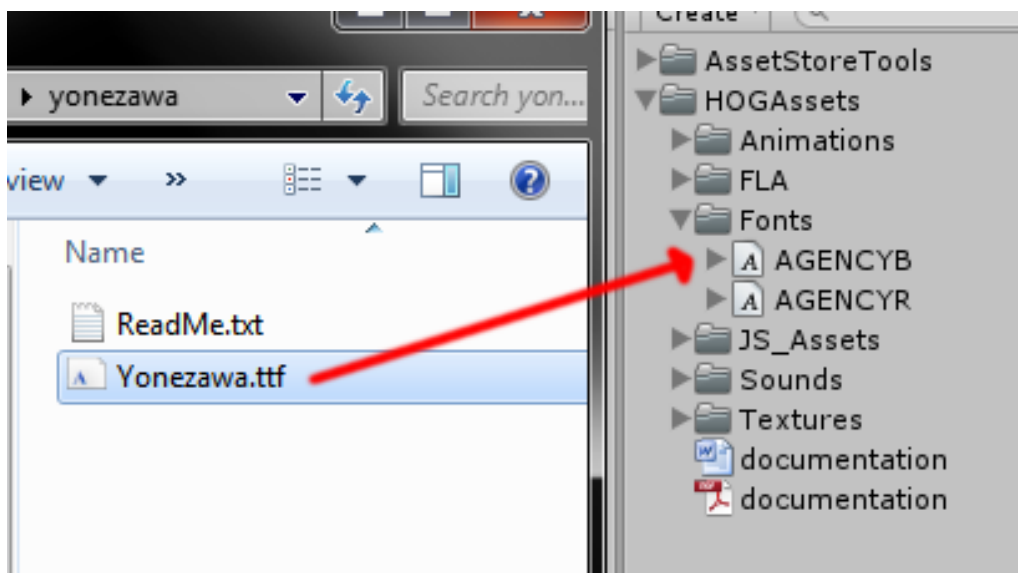
To solve this we must change the texture compression format for iOS. Follow these steps:

1. Click on a texture in the project view.
2. Click on the override for iPhone button on the right side.
3. Change the format to 16bit.
4. Click Apply.

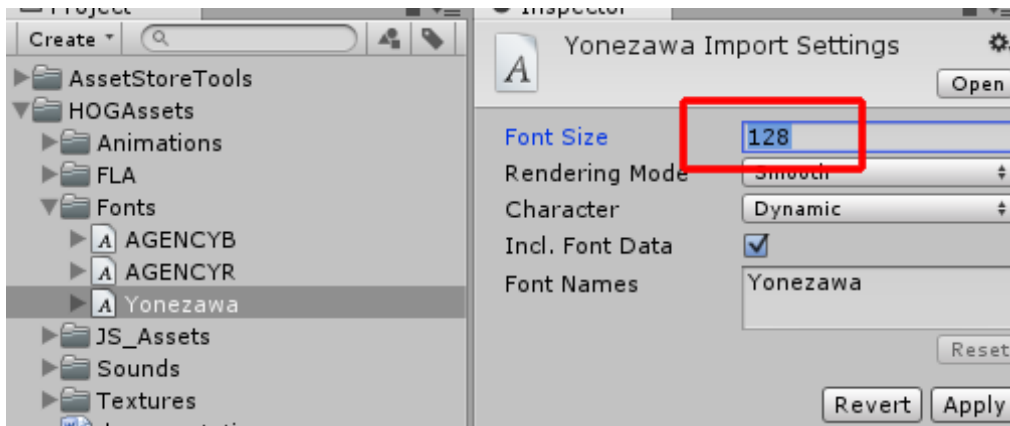
How to change font in the game?

To change a font in the game do the following:

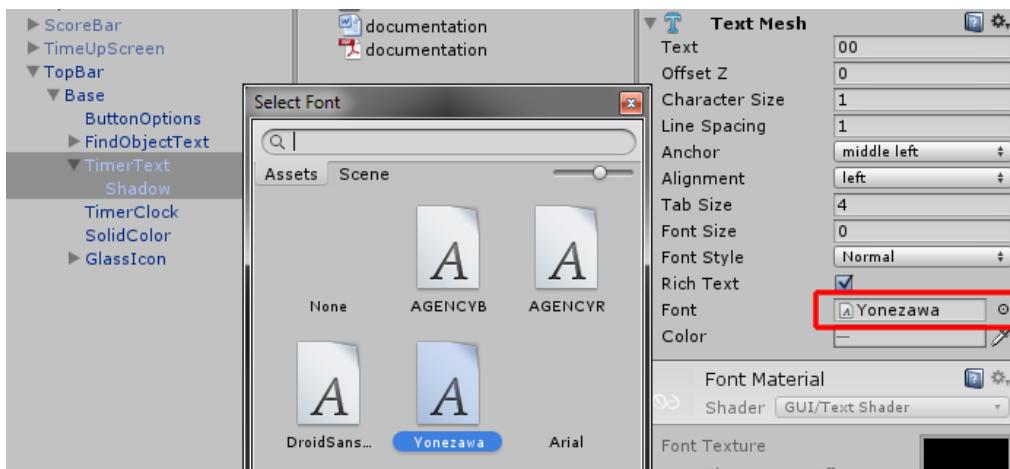
Find a font you like and drag the .ttf file over to the Fonts folder in your game.



Click on the font you added and edit its attributes. I personally set all my fonts to a high number (and then scale the text object down) so that they look crisper in-game.



Select any text object in the game and change its font to the new font you have. Sometimes the text might disappear, but it's normal. Just write something in the text box above and it will refresh. Also, make sure you change the text for the shadow; you can select both the main text and its shadow and edit them together.



SPACE ACE!



- FULL GAME READY FOR RELEASE
- EASILY CUSTOMIZABLE
- MOBILE FRIENDLY



HIDDEN OBJECT GAME

- FULL GAME READY FOR RELEASE
- EASILY CUSTOMIZABLE
- MOBILE FRIENDLY

A promotional image for a game titled "MARTIANS VS ROBOTS". The title is displayed in large, stylized letters: "MARTIANS" in white with a black outline, "VS" in orange with a black outline, and "ROBOTS" in purple with a black outline. Below the title, the subtitle "HORIZONTAL TOWER DEFENSE STARTER KIT" is written in white capital letters. The background is a dark blue sky with a yellow ringed planet on the left. The ground is a brown, textured surface. Several robotic units are visible: a grey tank-like robot on the left, a blue robot with a red eye in the center, and a blue robot with a red eye on the right. A small white robot is also visible on the far right.

Lockpicking & Safecracking Toolkit

Follow me on twitter for updates and freebies!

Page 14