

PUZZLE STARTER KIT - Created by BestSoft © 2014.

This Standard package was developed to help create full 2D puzzle games. For any problems or comments email bestsoftapp@gmail.com

SHORT PRESENTATION

A short video can be found here: <http://www.youtube.com/watch?v=RQKQVSeOUq4>

The game (or starter pack) presented is a classic puzzle game. It is based on placing parts (at the left of the screen) in the places right (under the board). The pieces are clearly outlined while their housings are drawn like shadows.

After the player has managed to put all the pieces in their places, the score will increment with one unit and will appear a menu of three buttons. The player can choose to stay on current board (resume button), to go to the next board (next level button) or to leave the game (quit button). Pieces are moved by drag and drop effect. The pieces are activated by clicking with the mouse (or tap on them). After activation, the piece is autoscale itself to slot size that should be placed.

We have, therefore, only two important elements in the game: the piece (we'll say next: piece) and where it should be that piece placed (slot we will call simple: shadow). So we have pieces and shadows.

THE PACKAGE

The package contain six folders:

- 1) **Documentation:** with this explanatory document.
- 2) **Extra GUI Skins:** it is an asset from Unity (check this link:
<https://www.assetstore.unity3d.com/#/content/2>). I've used their OrangeGUISkin but a little modified. The GUI, the menu and all the buttons are for showing you an full game example but if you like other GUI systems or menus please be free to use.
- 3) **PreFab:** my intention was to keep this as simple as i can so i've managed to create this game with only five prefabs:
 - **Background:** the "Background" prefab represent the game board. It has a SpriteRenderer component where is no Sprite selected (i wanted to use one material for all backgrounds, no mater how many levels we have). The Sprite for background objects will be set in the inspector.

- BackgroundMusic: the “BackgroundMusic” prefab manage the game music. It can act like a playlist if you have more than one music for your game. It has two C# scripts attached on it, first is Singleton.cs script who is responsible of carry over the music through entire game. All you have to know is that the Singleton.cs script don't allow the attached object to be destroyed when another scene is loaded and is checking for object existence in loaded scenes. The second script is ContinousMusic.cs and this is our music player in game. The music can be set in the inspector.

- Delimiter: the “Delimiter” prefab is very simple with one single purpose, that of making a clear visible left zone for puzzle pieces. It has a SpriteRenderer component where is no Sprite selected. The Sprite for delimiter objects will be set in the inspector.. The effect is an transparent gray area on left of the screen.

- Piece: the “Piece” prefab it has a Rigidbody2D (GravityScale has to be 0 !!!), a BoxCollider2D (for 2D collisions, isTrigger has to be checked!), a SpriteRenderer, an AudioSource (for playing clips) and a C# script (PiecesBehaviour.cs) attached to it. The PiecesBehaviour.cs script is responsible of “DragAndDrop” behaviour and also play a sound when piece is moved in the right slot (on the correct shadow) and set a little nice glow effect on it. In the inspector you'll have access to two params: Shadow (is the spot GameObject where piece should fit) and glowPieceSprite.

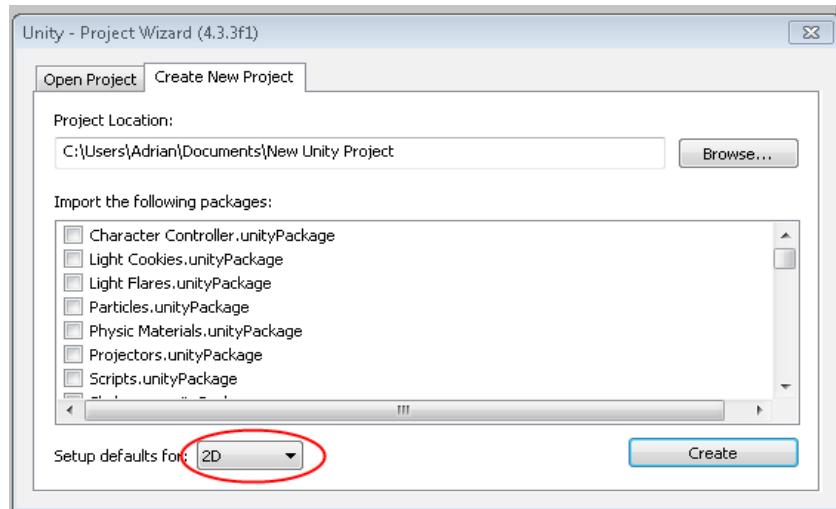
- Shadow: the “Shadow” prefab is yet another simple object with a BoxCollider2D (for collision with the piece) and a SpriteRenderer component where is no Sprite selected. Shadow sprite is set in the Inspector.

- 4) **Resources**: it is straightforward folder with other subfolders like: Audio (game music, level sounds) and Sprites (common game textures and level textures).
- 5) **Scenes**: it will be two scenes: Level1 and Level2. In this document i'll include an step-by-step tutorial on how i'll make Level2 scene based on level1 scene. When first you import this package there maybe will be none scenes in build options so don't forget to include current levels (as they are or modified by you) and further ones into build options.
- 6) **Scripts**: there are four scripts all in C# language. The only one i've not told you is Menu.cs script. The Menu.cs is attached to MainCamera object and is responsible with GUI events, GUI buttons, labels, etc. I've included also the EndLevel() method responsible of validating the level or with clearing the level by the player.

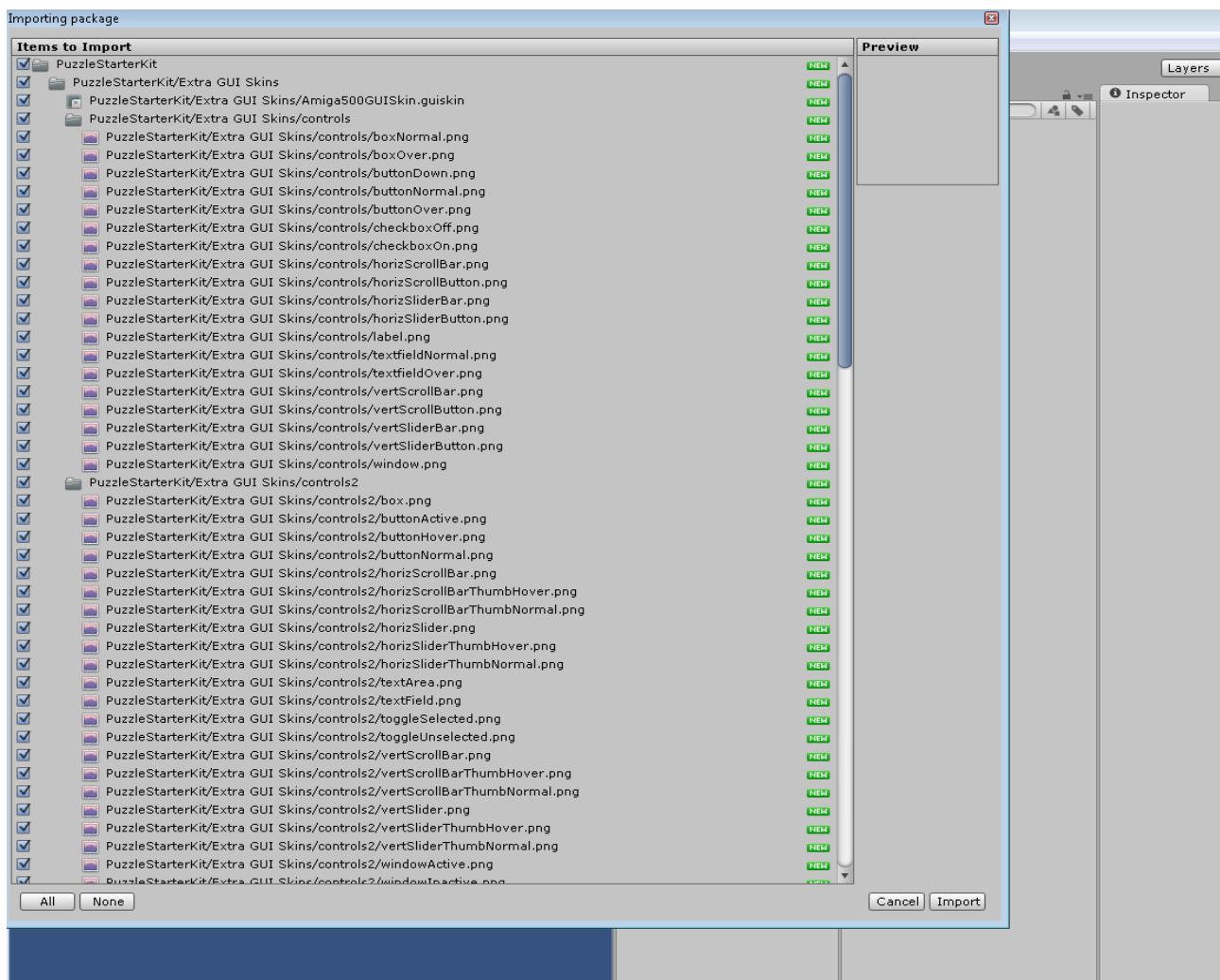
IMPLEMENTATION AND EXPAND – a step by step tutorial on this package.

IMPLEMENTATION

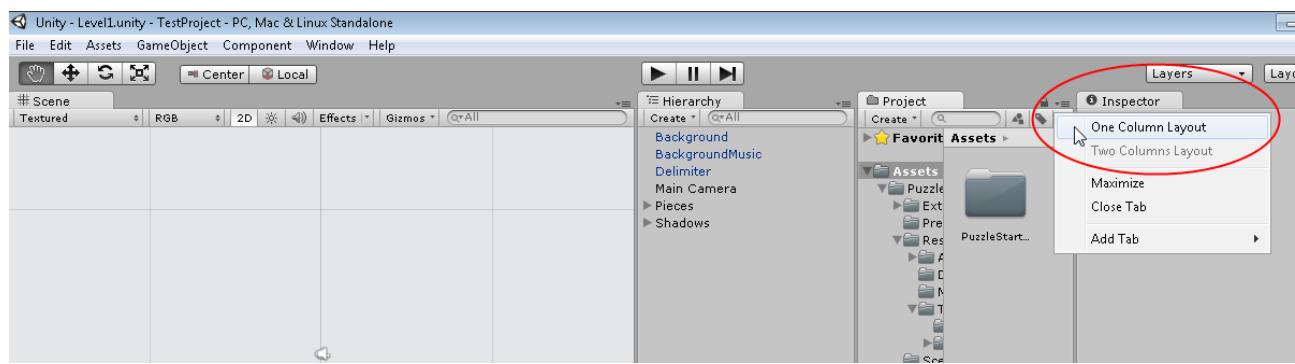
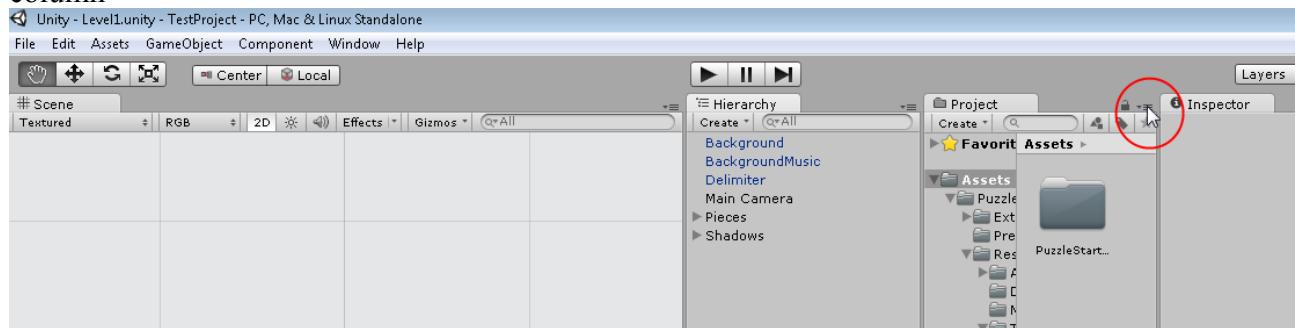
Step1: In a Unity create a new empty 2D project!



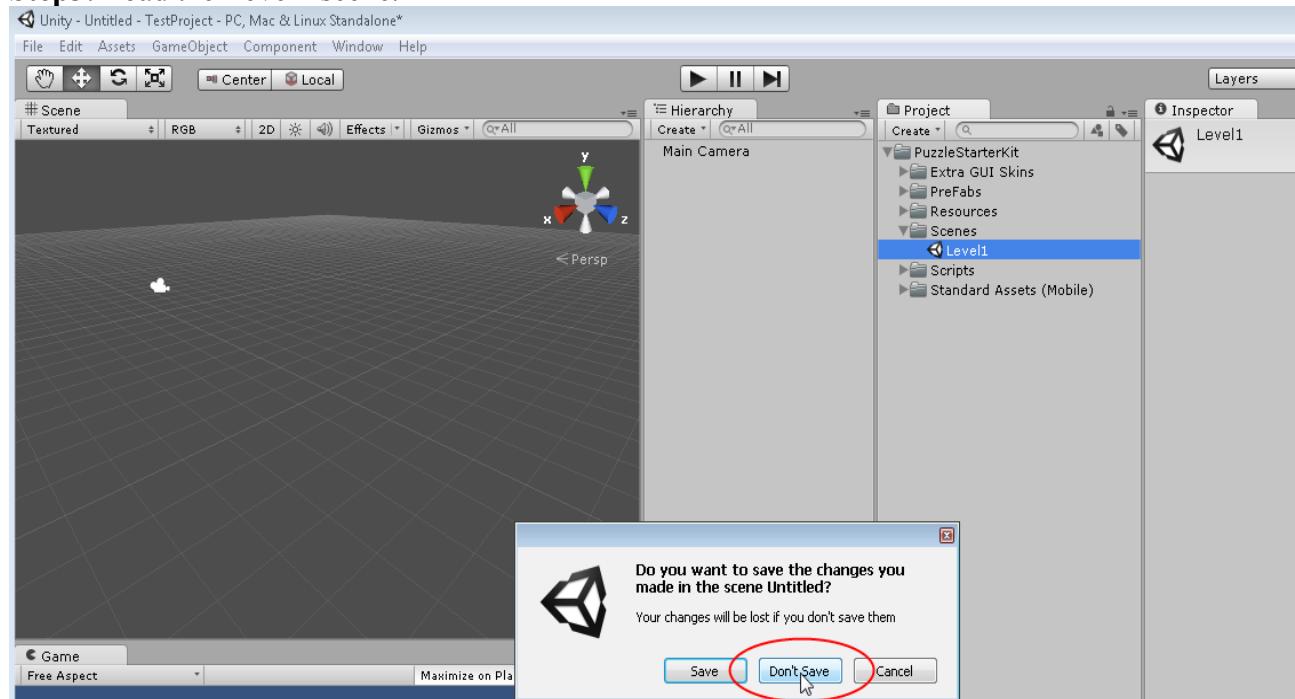
And in this new empty project please download and import the package.



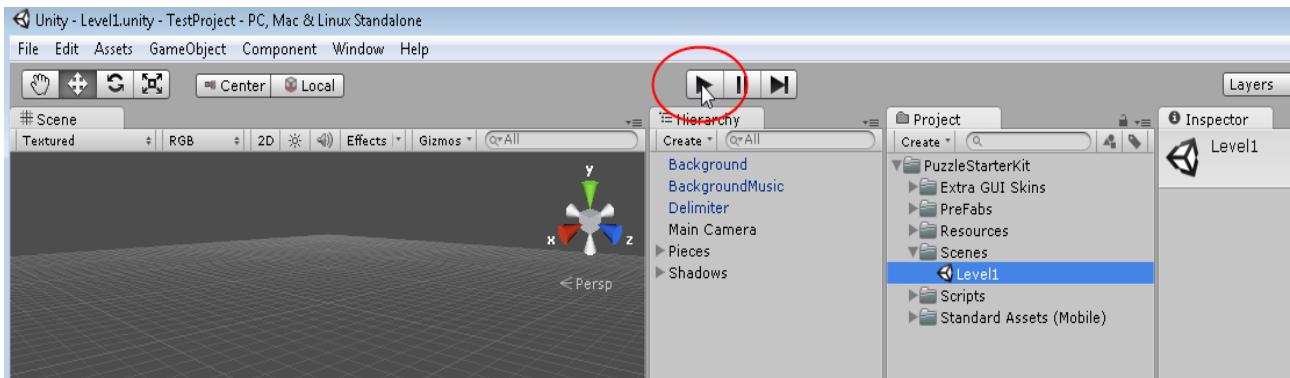
Step2: Set the window layout to 2 by 3 from Window/Layouts/2by3 and on Project window set one column



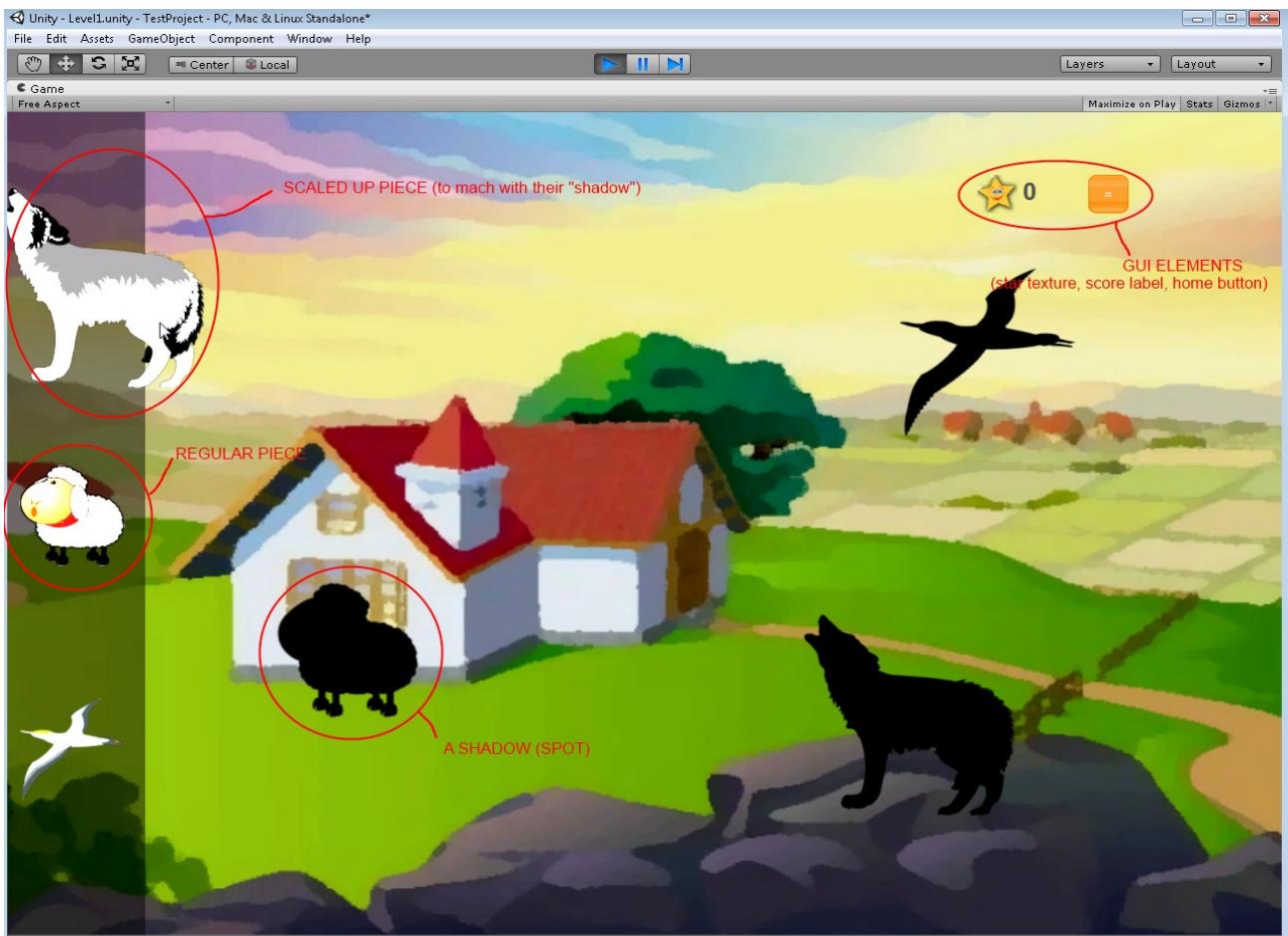
Step3: Load the Level1 scene.



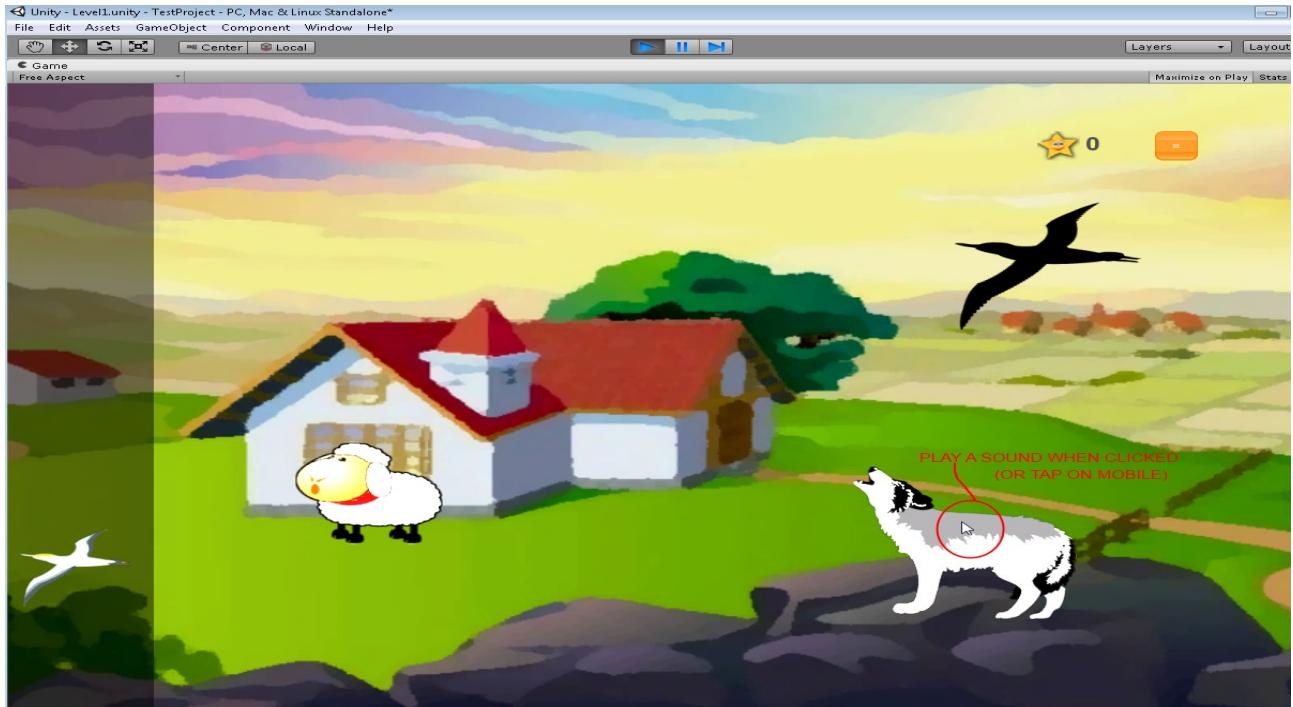
Step4: Play the level.



Step5: See if level is playing



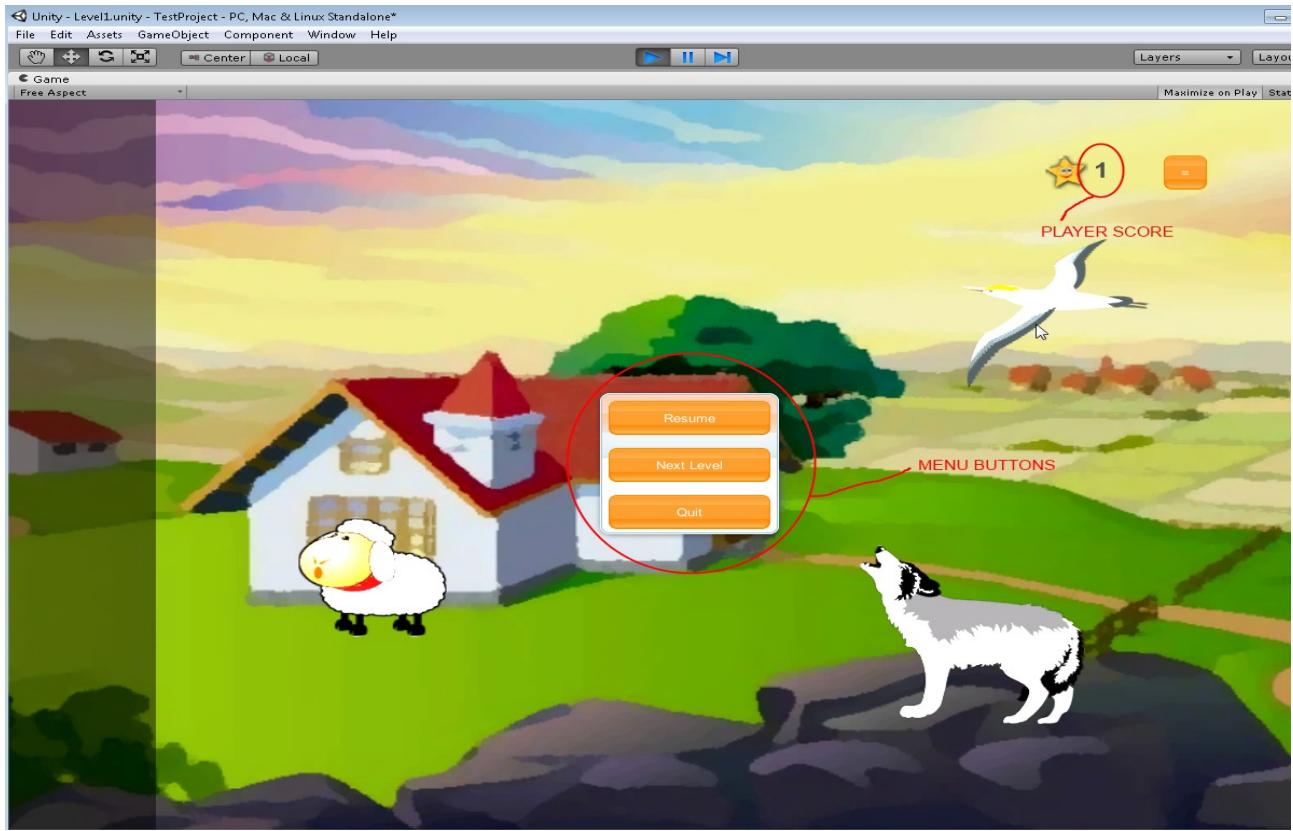
Step6: Put maximum two pieces (if you'll put all three level is finished, leave this later please) on board and click on one of them. You'll notice that the sound played when you put the piece first it will be played every time you click on piece.



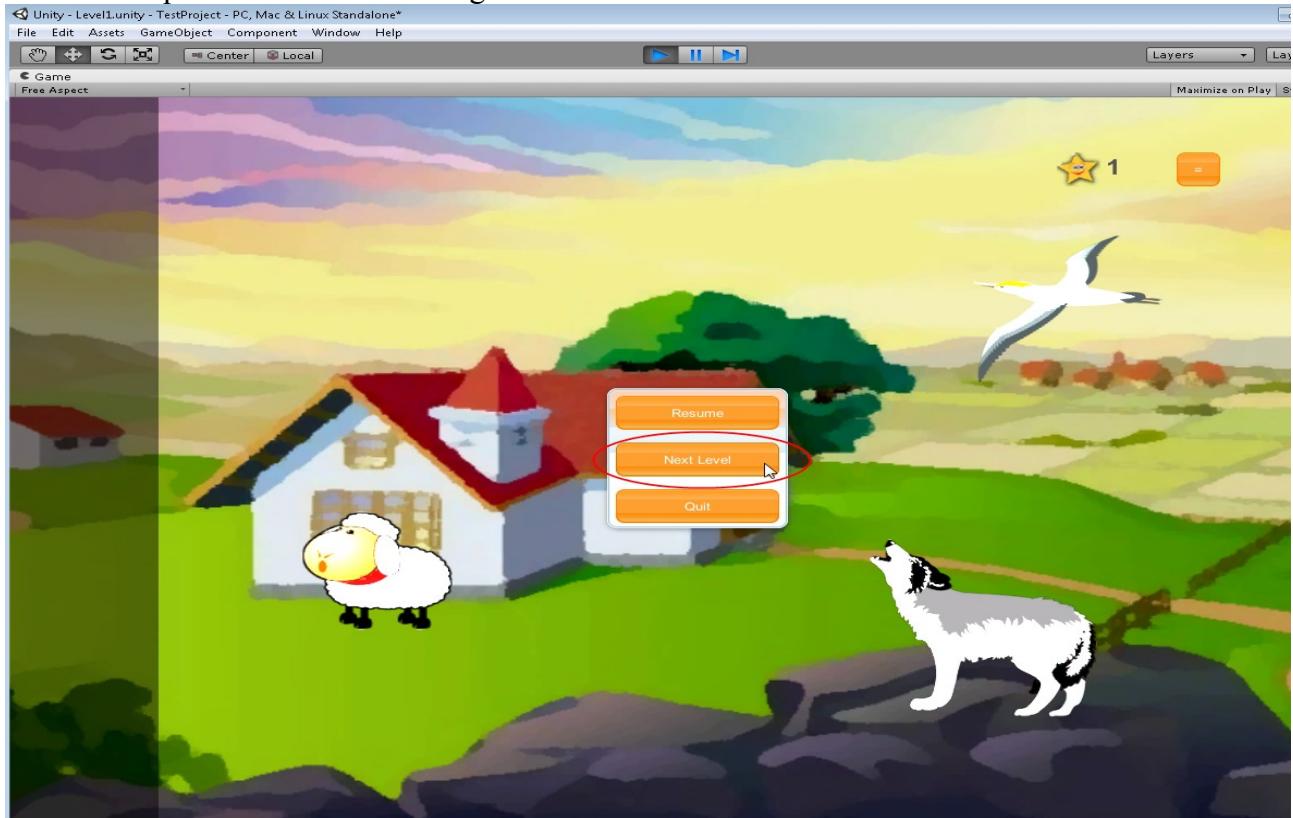
Step7: Also when tap (or click) on a piece you'll notice a nice glow effect on it.



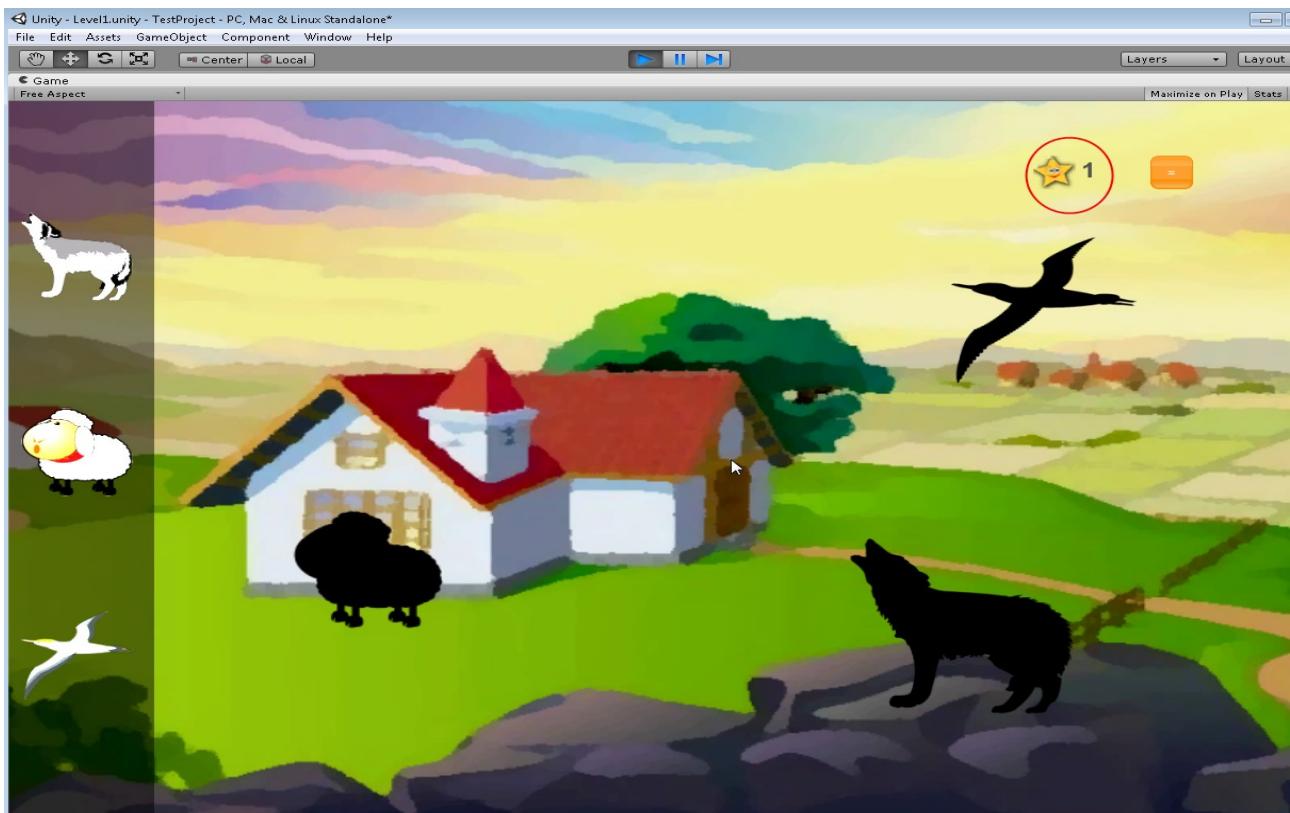
Step8: When all the pieces are in their spots (on their shadows) then menu will appear and player score its incremented by one unit.



Step9: When click on Next Level button, if no other level is available then current level will be loaded with pieces initialized once again.



Step10: Notice that after reloading the level the score is saved and remain incremented.



Step11: Click on Maximize on Play button



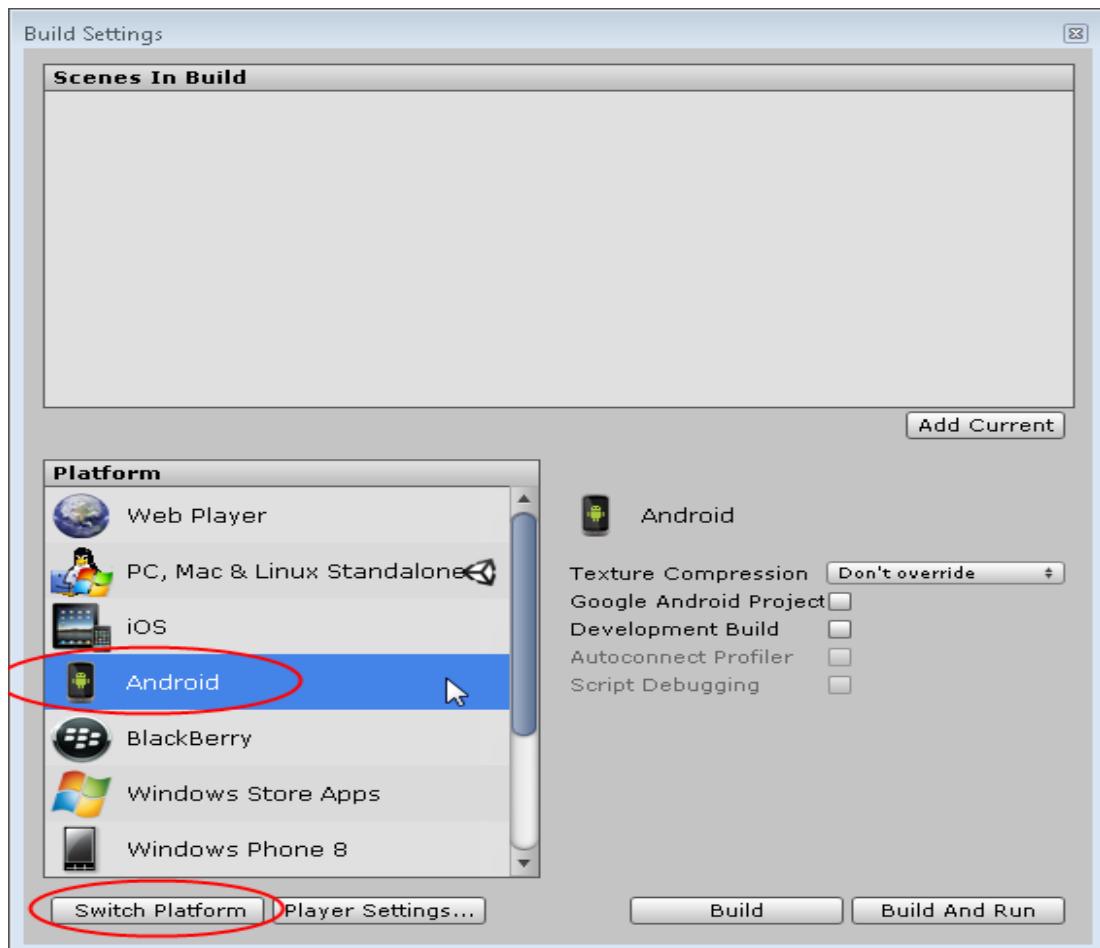
Step12: Click on middle arrow and stop the game from playing anymore.



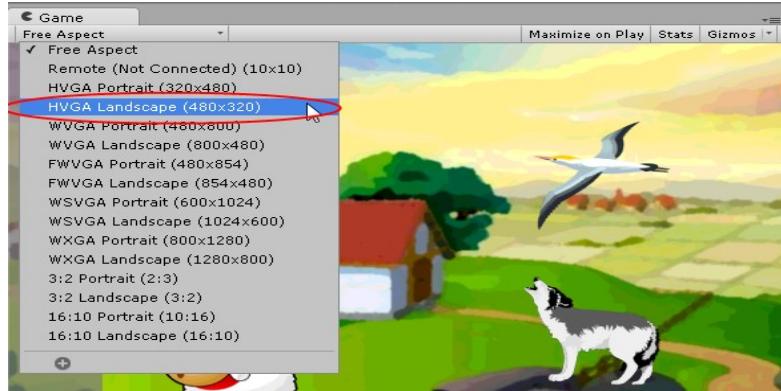
Step13: At this point you have a working level of your puzzle game. This level you can change as you wish but first let see its structure and do few more settings on environment.



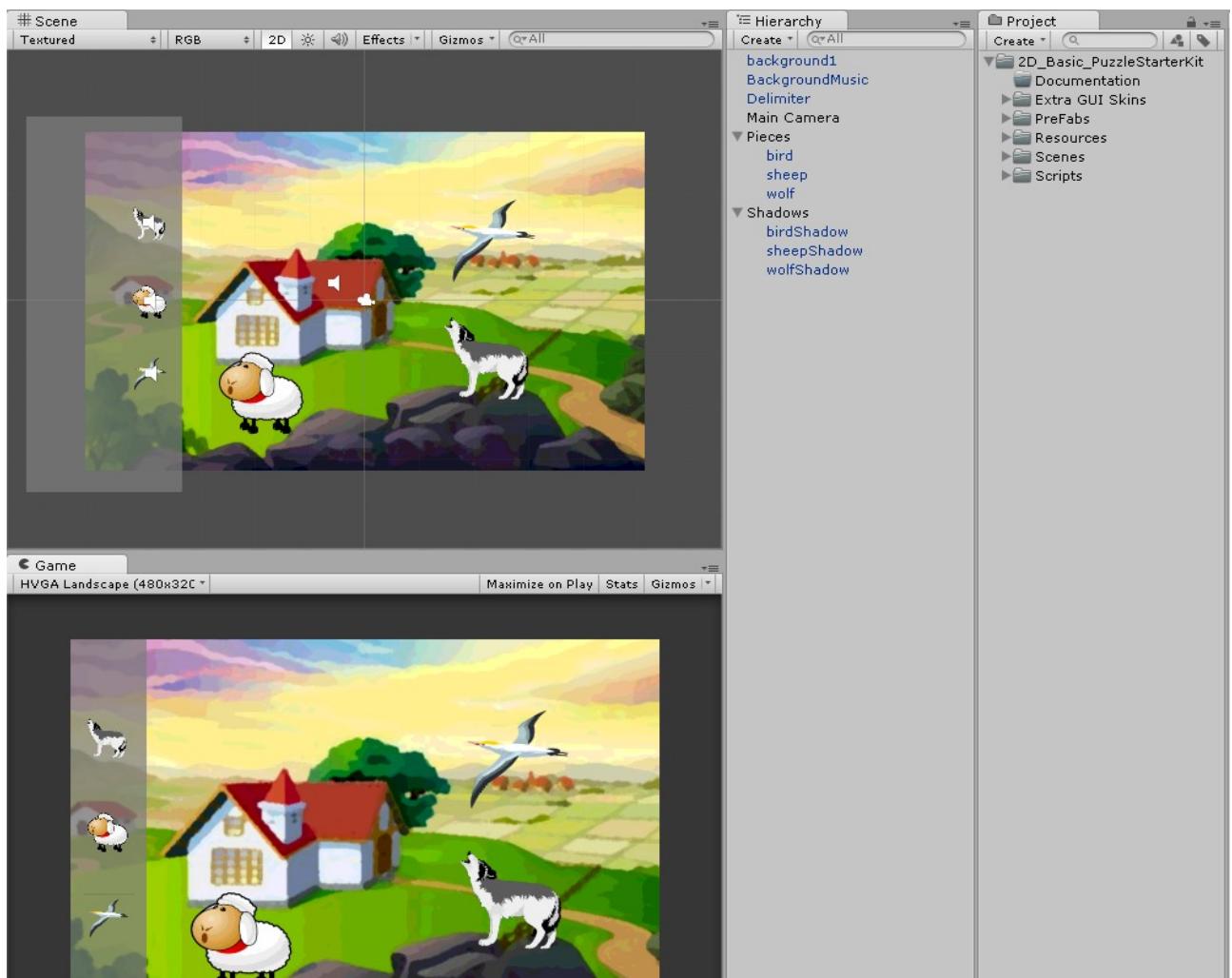
Step14: Let us change the platform for our game to Android.



Step15: And change the resolution.

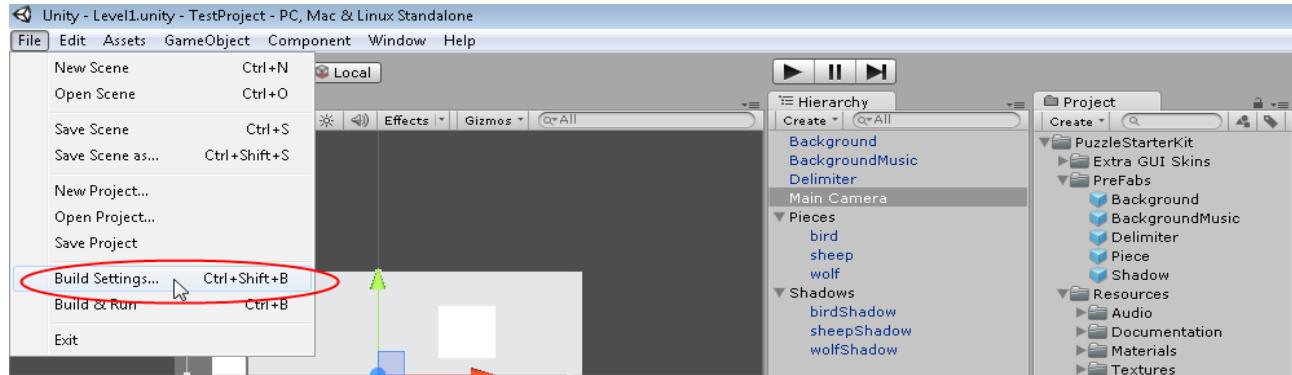


Step16: Now you've set up your environment and can go further.

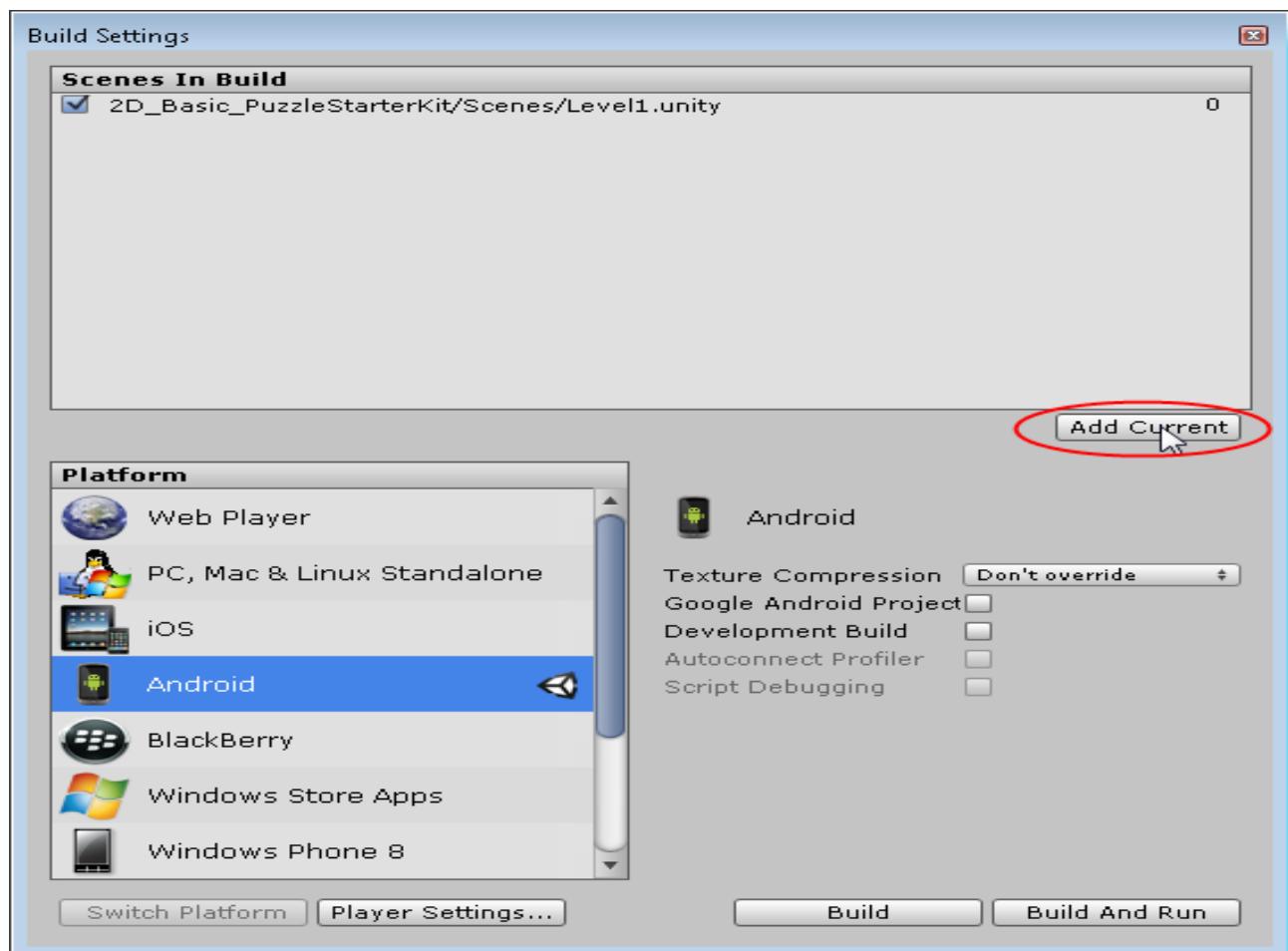


After going on please take a little time to inspect this scene (Level1). Basicly you have a Background, BackgroundMusic, Delimiter, MainCamera and two groups: Pieces (three pieces: bird, sheep, wolf) and Shadows(birdShadow, sheepShadow, wolfShadow). If we exclude MainCamera, all our objects are instantiated from their prefabs (explained at the begining of this document).

Step17: Go to File/BuildSettings.



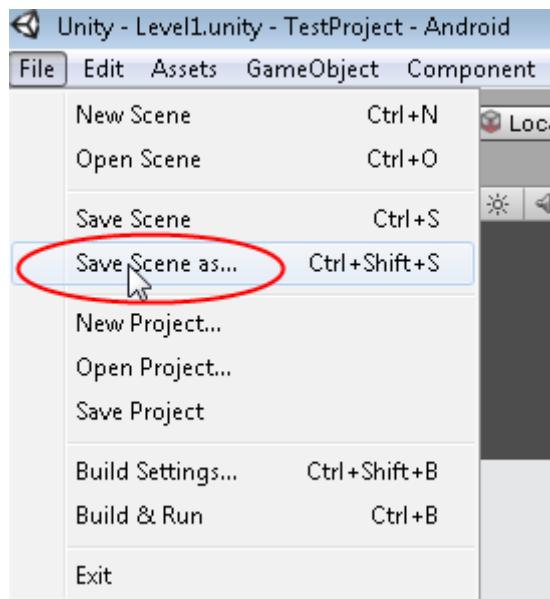
Step18: Click on Add Current button to add scene to the game build



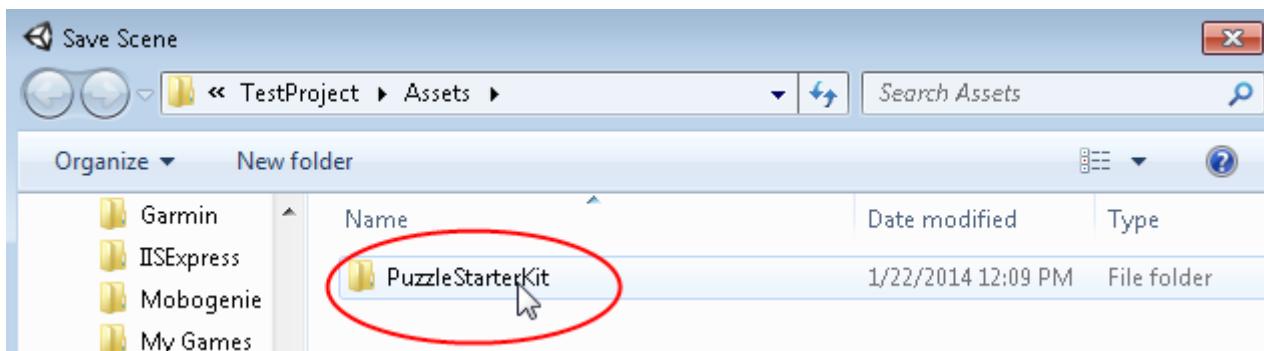
Now you are ready to expand your puzzle starter pack. In next section we'll see how to make another level.

EXPANDING THE GAME.

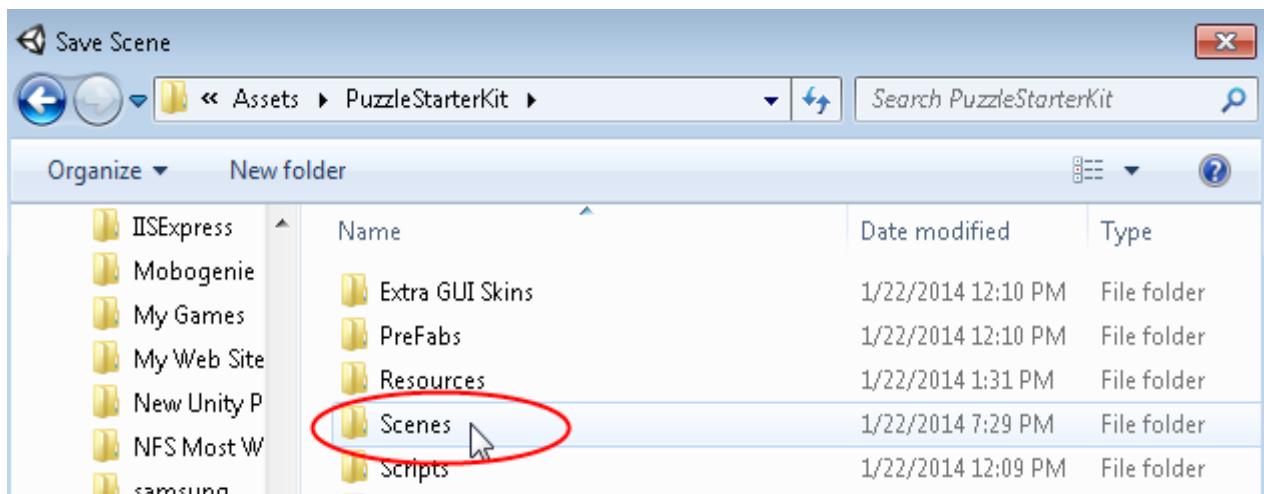
Step1: Go to File and chose Save Scene As...



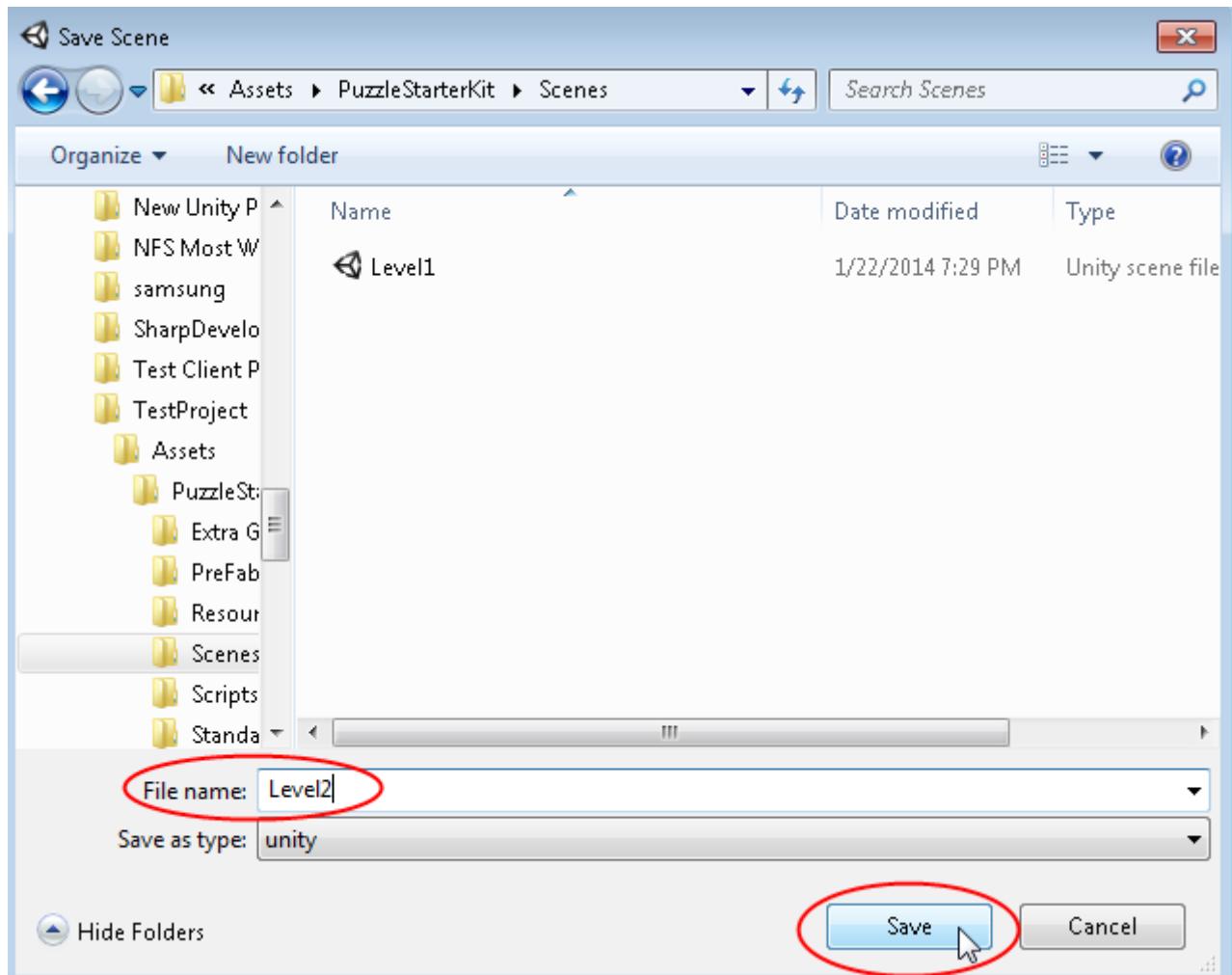
Step2: Double click on PuzzleStarterKit...



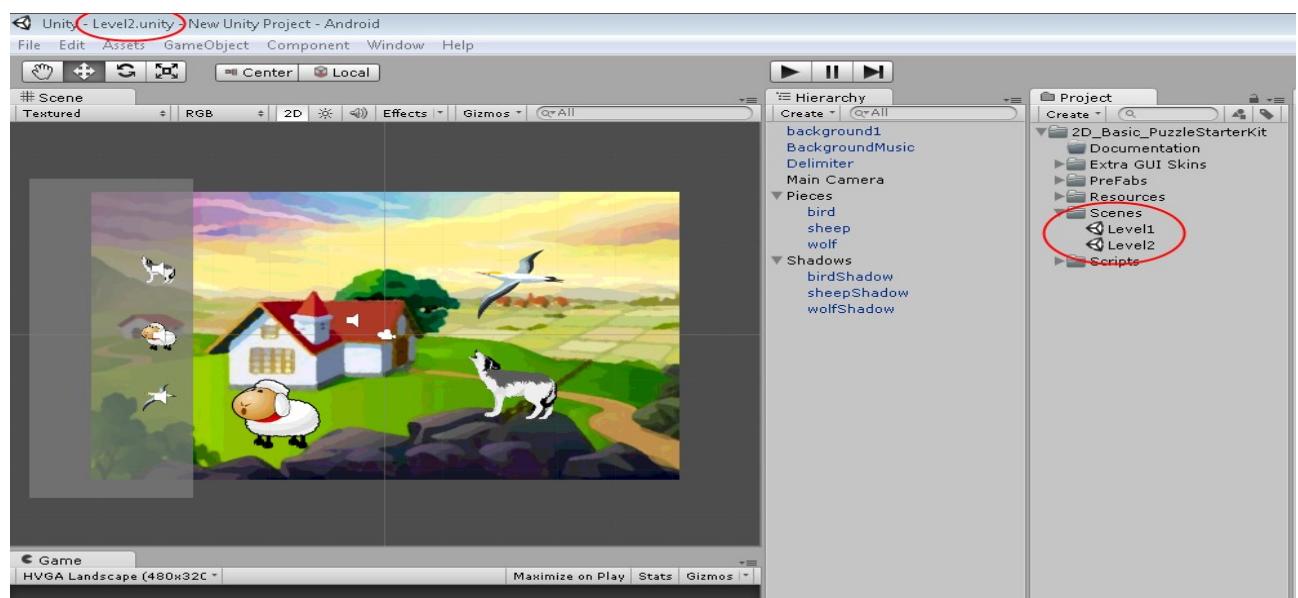
Step3: double click on Scenes...



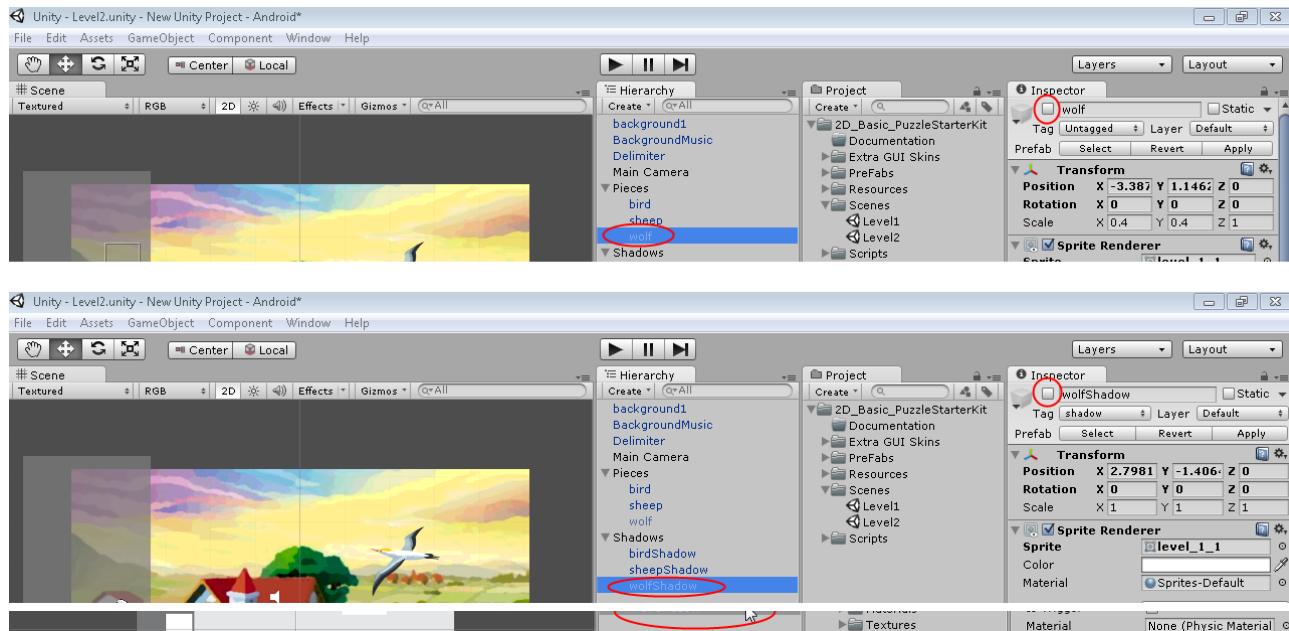
Step4: name your new scene as Level2 and click save.



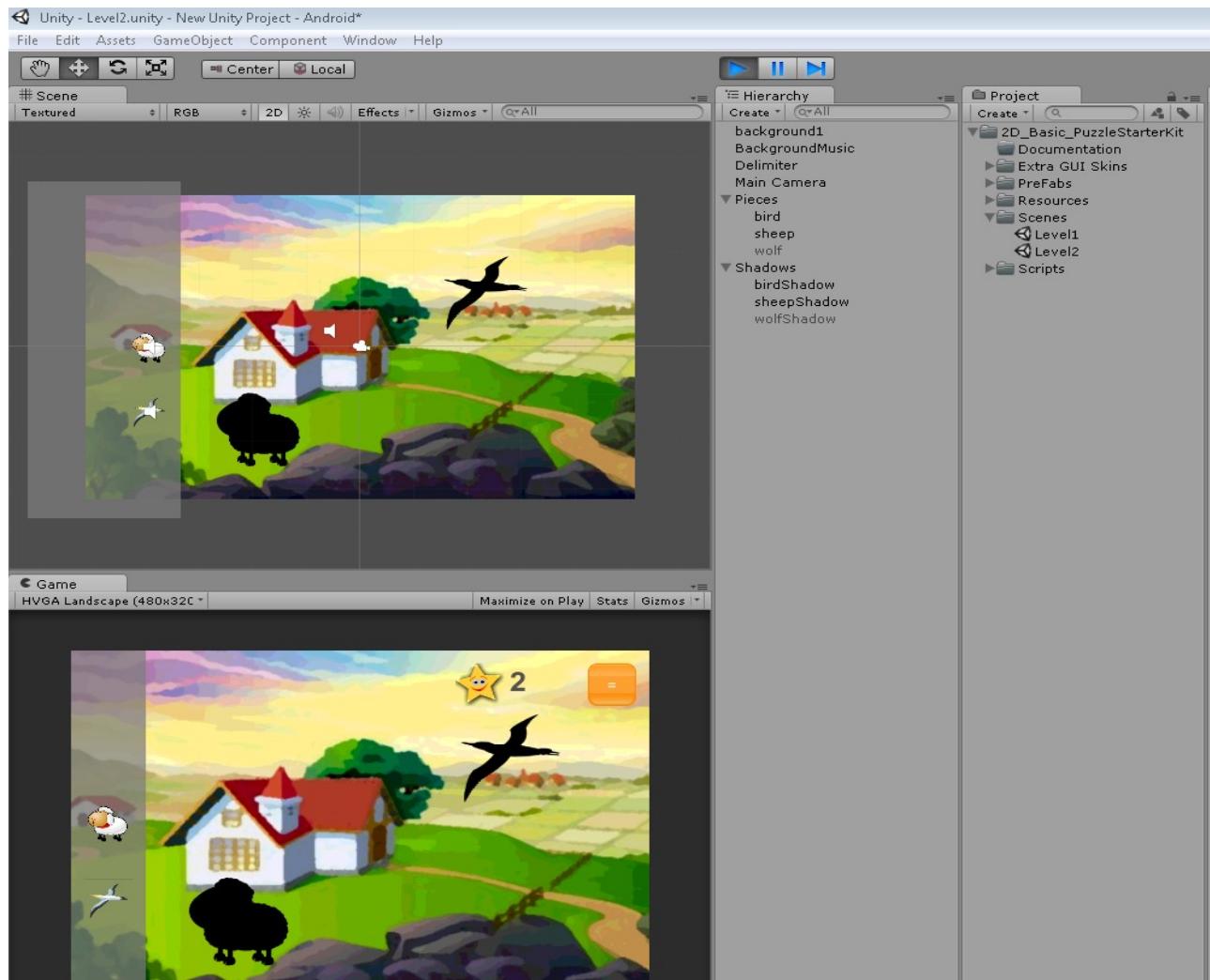
Step5: Now you'll have Level2 opened and it can be viewed in Scenes folder.



Step6: For testing purposes we change Level2 a little by deselecting “wolf” and “wolfShadow”.

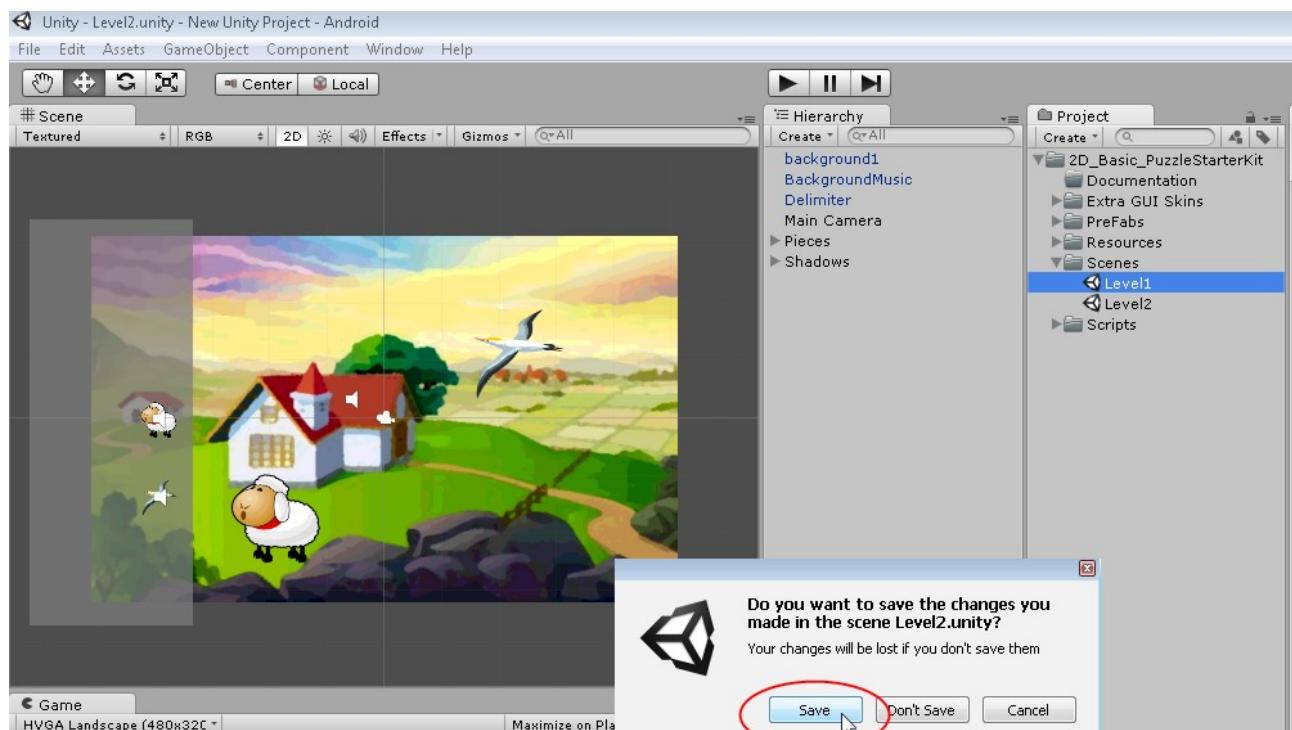


Step7: Now will have a level with two pieces: the bird and the sheep and their shadows.

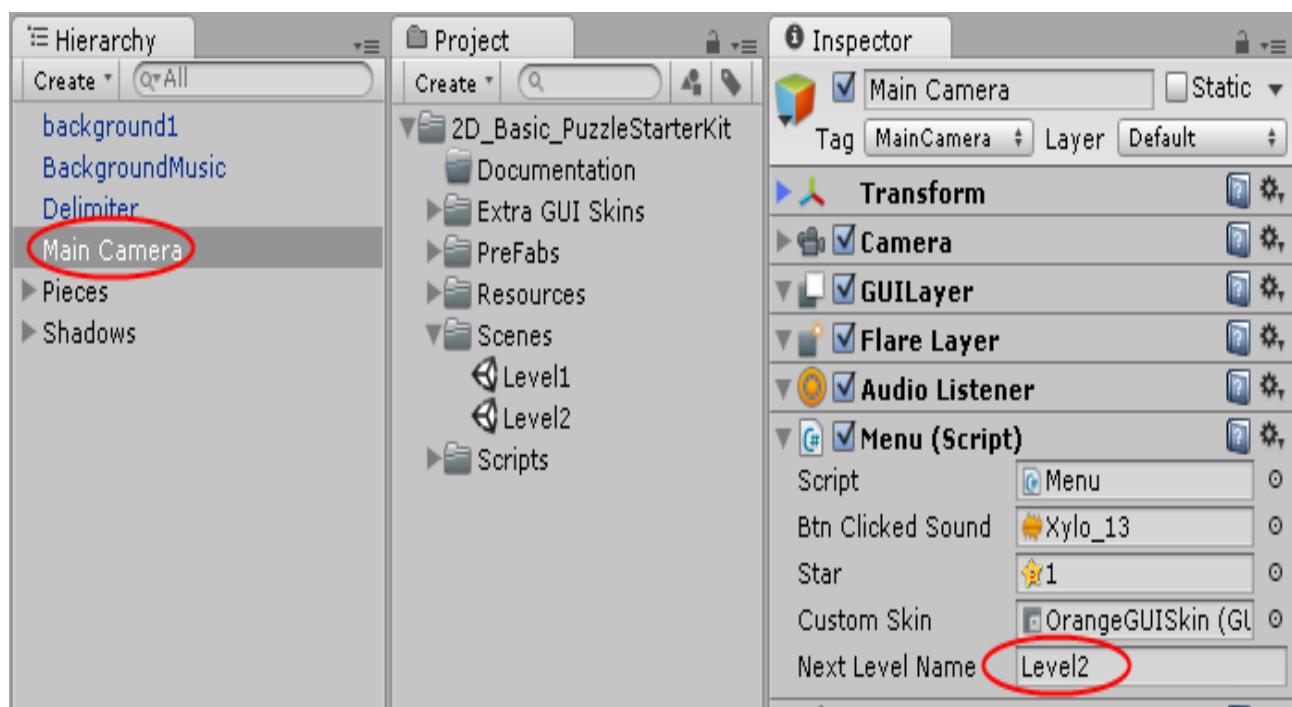


Step8: Before anything else, please add Level2 to the game build (remember?

File/BuildSettings...and click on Add Current). Now we load Level1 by double clicking on it from Scenes folder and also click Save for those little changes made on Level2.



Step9: Select MainCamera from Level1 and in the Inspector look for Next Level Name and write your second level name (Level2 in our case).

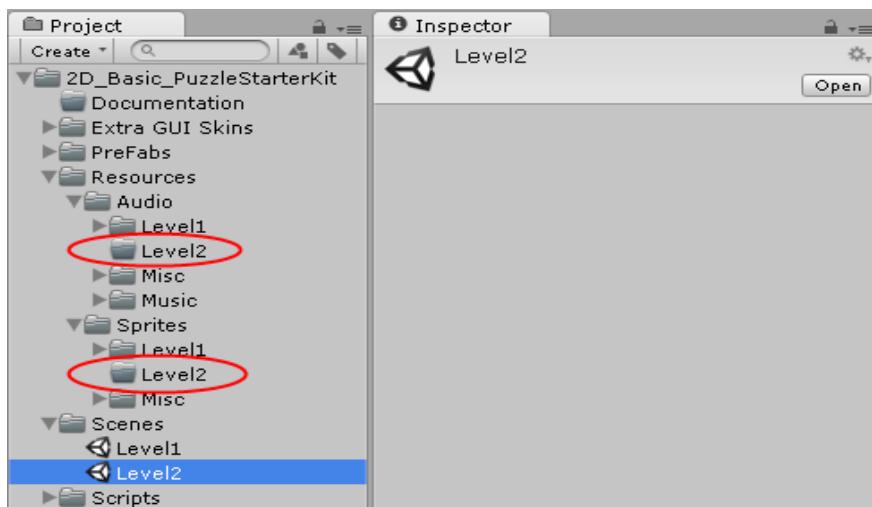


Step10: Hit play button and put all three pieces on their shadows. After that you'll see Level2 being loaded because there is no wolf and wolfShadow objects anymore. So you managed to successfully add another level to your game. Now we going to change at all this new level as we want to be different than first one.

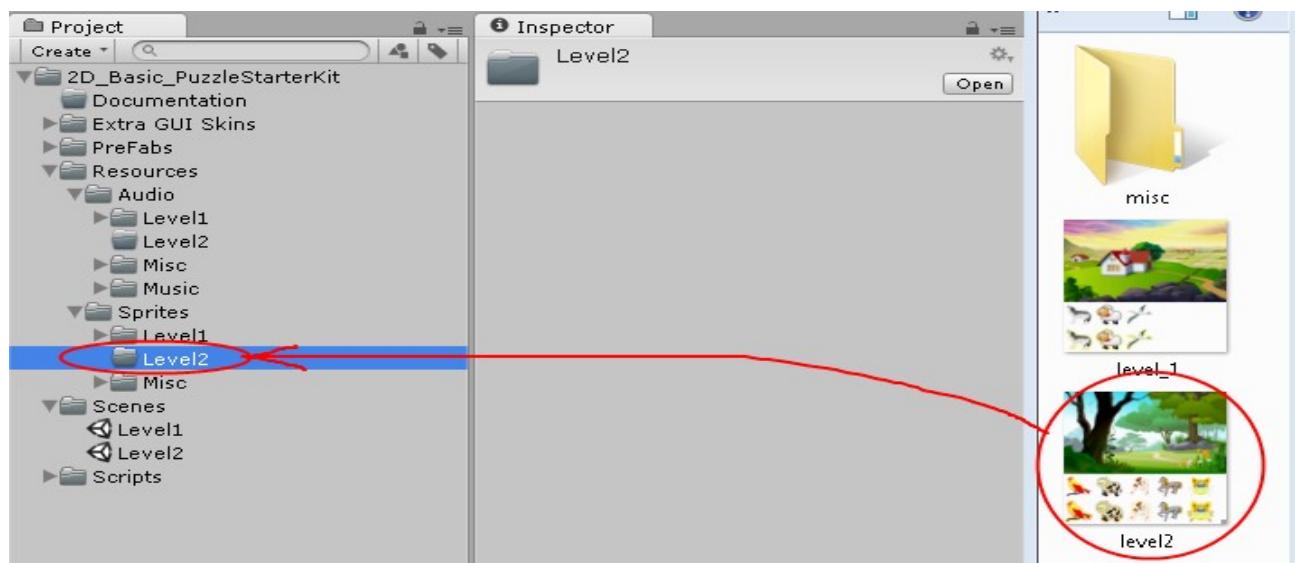
All we need in order to change all level2 is at least same number of textures from the copied level (in our case Level1). So if we change audio and textures from those three (wolf, bird, sheep) pieces and the level background all will look new and fresh. So we took our time and find some cool textures and sounds for this task.

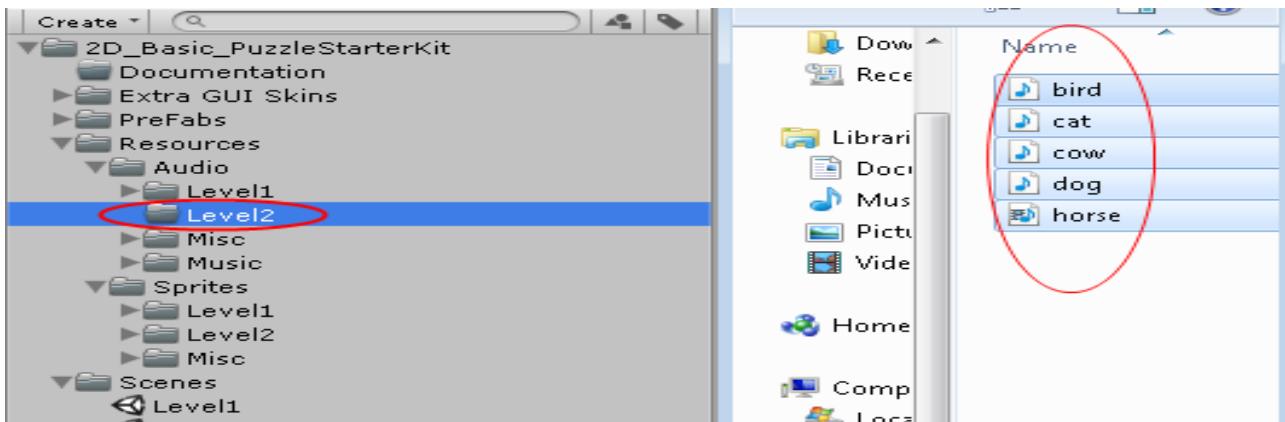
Please, at first, load again Level2 scene and select in the Inspector the wolf piece and the wolfShadow spot. Now we have all three objects (pieces) ready for changes.

Step11: We need to create two folders: in Resources folder, make one folder (called Level2) in Audio and another in Sprites.

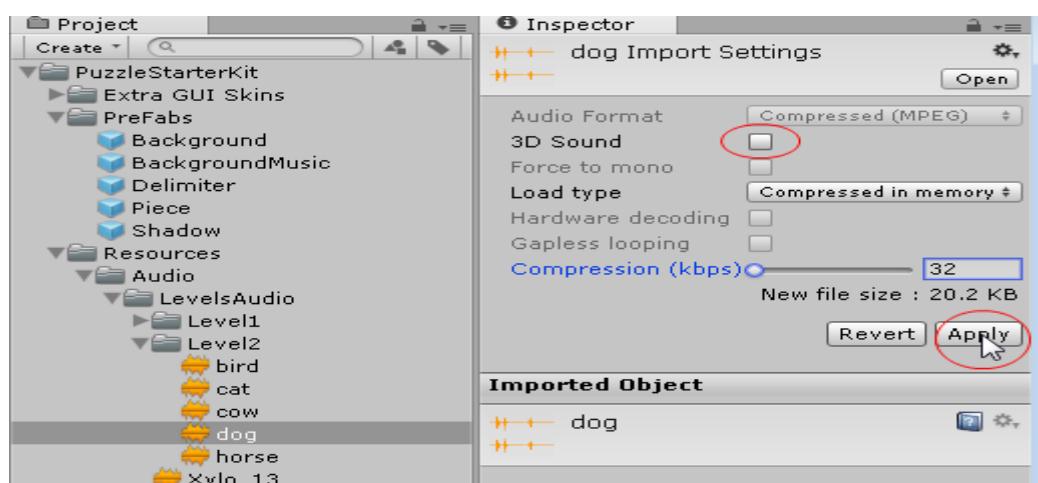


Step12: Then drag your level2 Sprite Sheet and audio in their newly created folders.

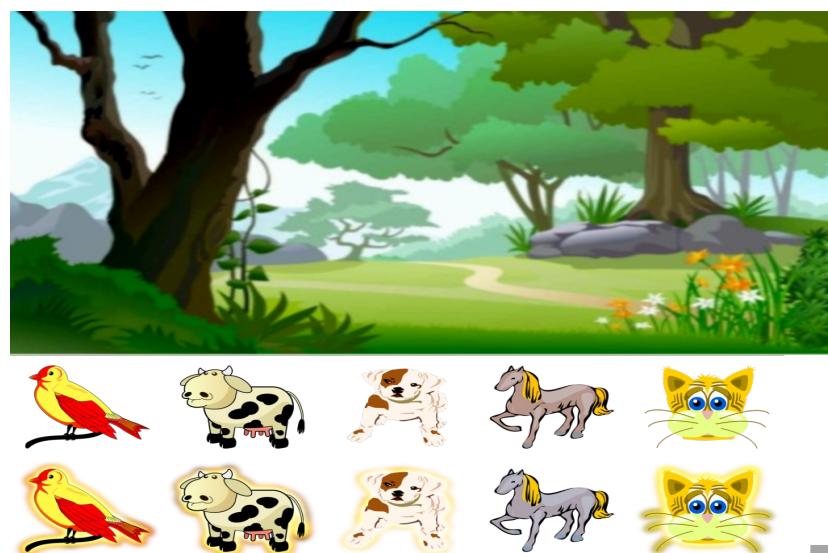




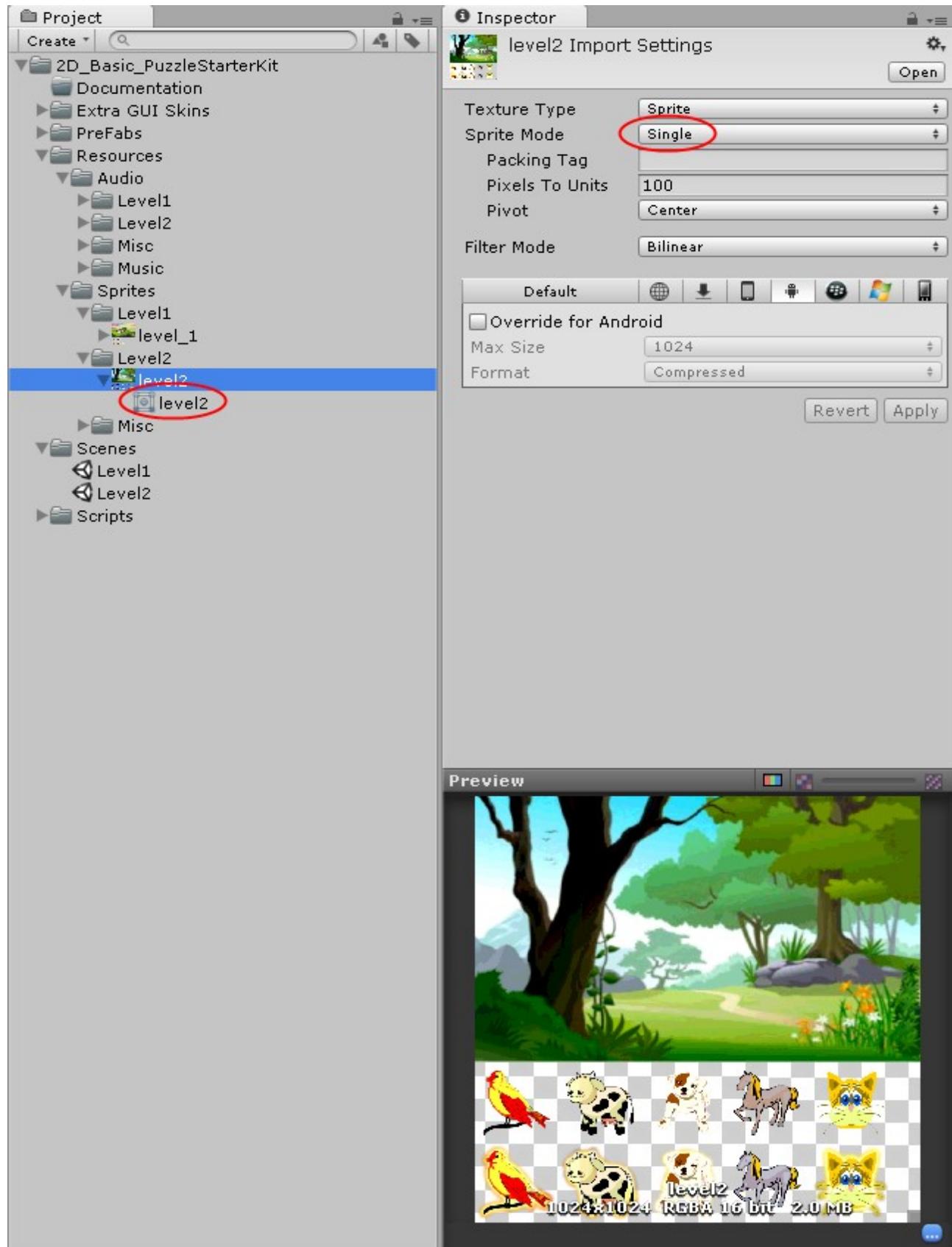
Step13: Must pay attention, please, to uncheck 3D Sound checkbox because we don't needed as all the action have place in a vertical plane.



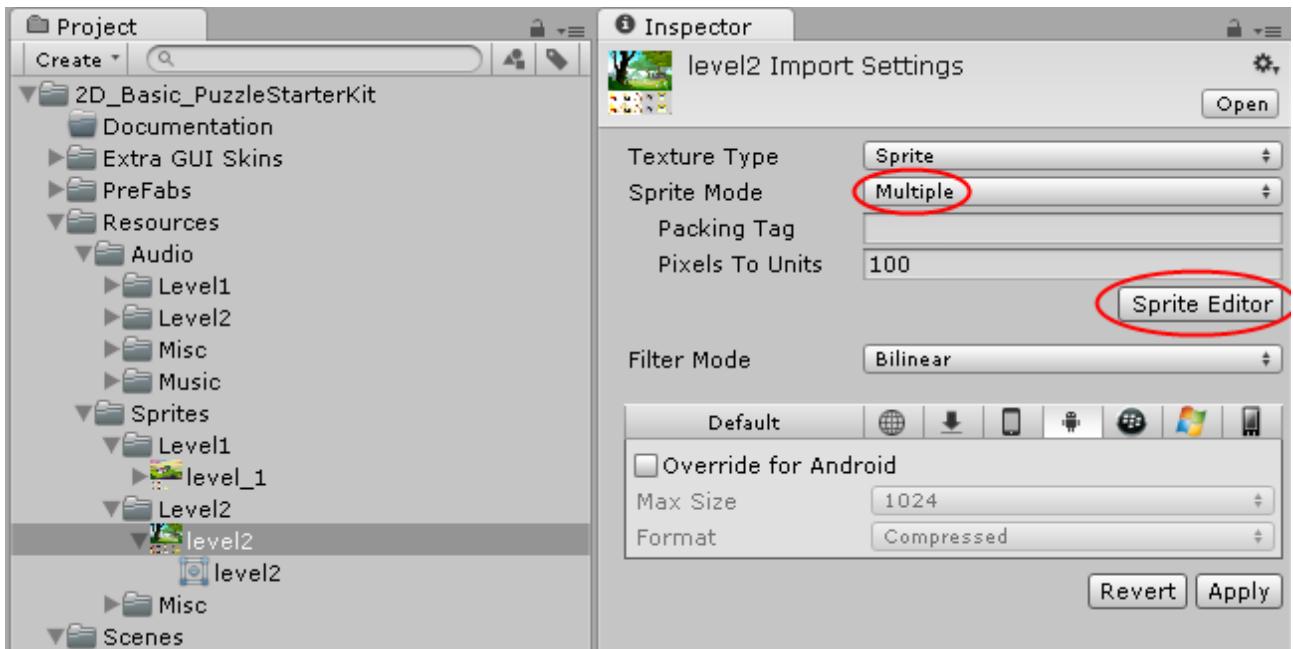
Step14: Before going further must explain a cool feature Unity 2D have: Sprite Sheets. In our case level2 Sprite Sheet will include all sprites needed for it.



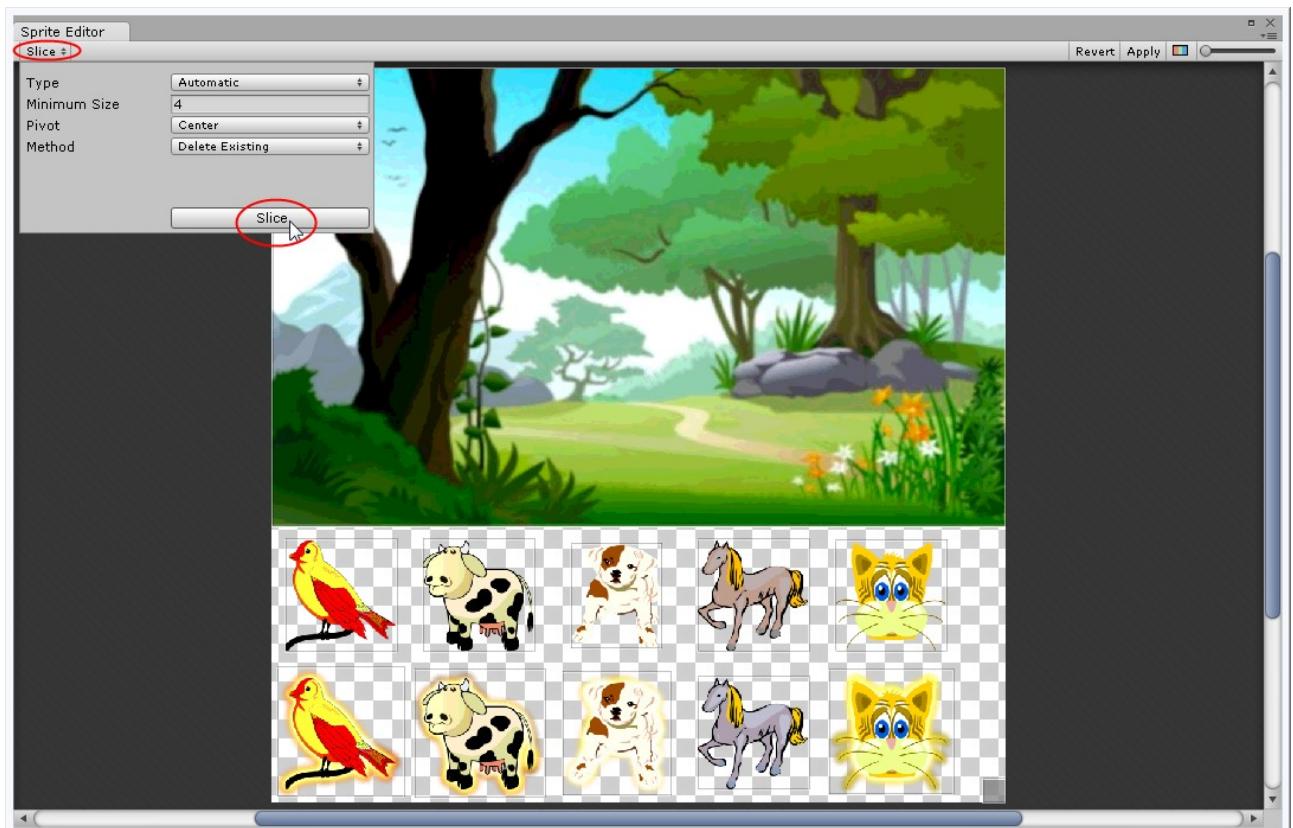
When you'll inspect your level2 SpriteSheet you'll observe that unity will make it by default as a single sprite (useless in this state):



Step15: All we have to do is to change that single sprite setting to multiple:

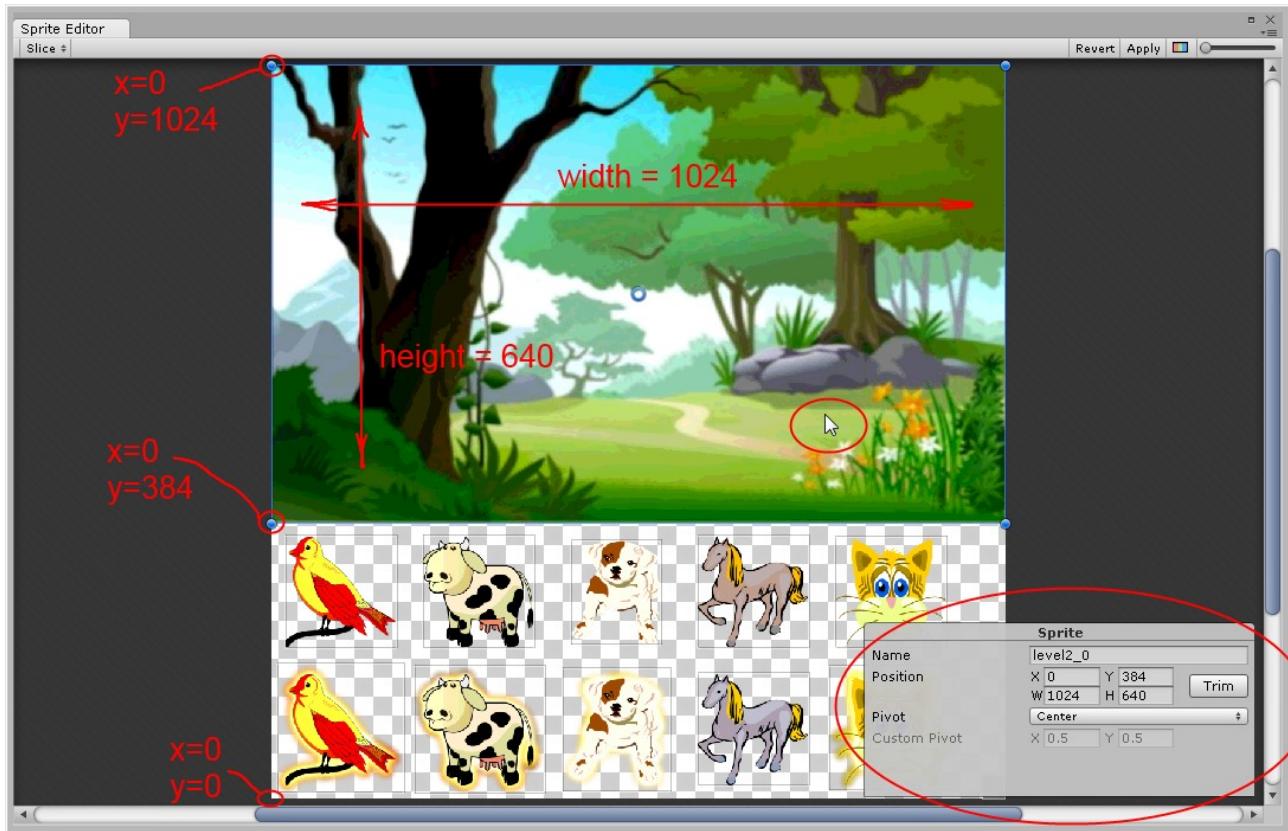


Step16: After this a SpriteEditor button will appear. Please click on that button and you'll have a sprite editor window. Click on upper left Slice button and (leave all those settings as they are) click on Slice (where my cursor is in following picture):



Step17: Unity will automaticaly try to slice every sprite in level2 spriteSheet (and generaly is doing a great job on that) but i prefer to adjust this sliced sprites. We'll click on every sprite and adjust their settings manually:

Click on background sprite and a sprite window will appear:



Step18: You notice that, in background case, Unity done a perfect slice (background sprite is 1024x640), so we can move to next sprite: lower left bird (glowed bird sprite):



Step19: All animals sprites need to be 192x192 so we'll adjust Unity's sliced sprites as follow: in Sprite Window change X=0, Y=0, W=192 and H=192 and the result will be:



Step20: select the second bird from the bottom:



And change its values to: X=0, Y=192, W=192 and H=192, the result will be:



Step21: Select the glowed cow sprite from the bottom:



And change its values to: X=192, Y=0, W=192 and H=192, the result will be:



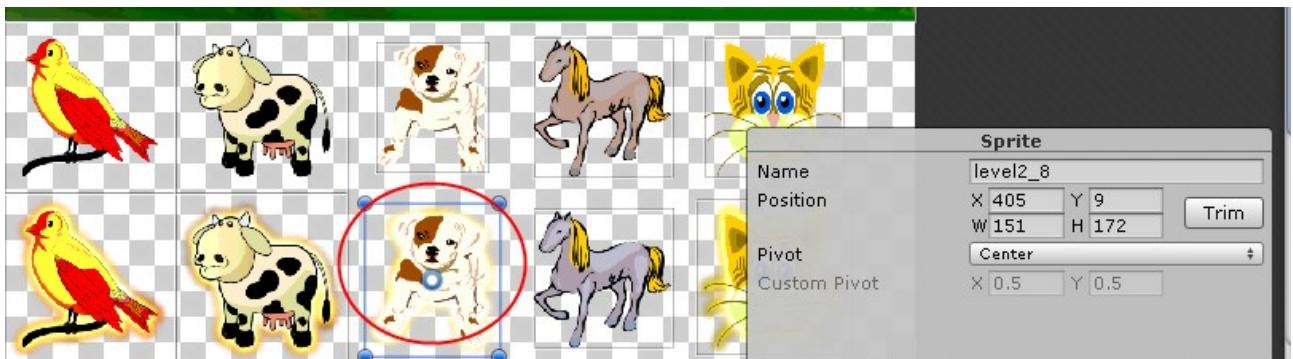
Step22: Select the second cow from the bottom:



And change its values to: X=192, Y=192, W=192 and H=192, the result will be:



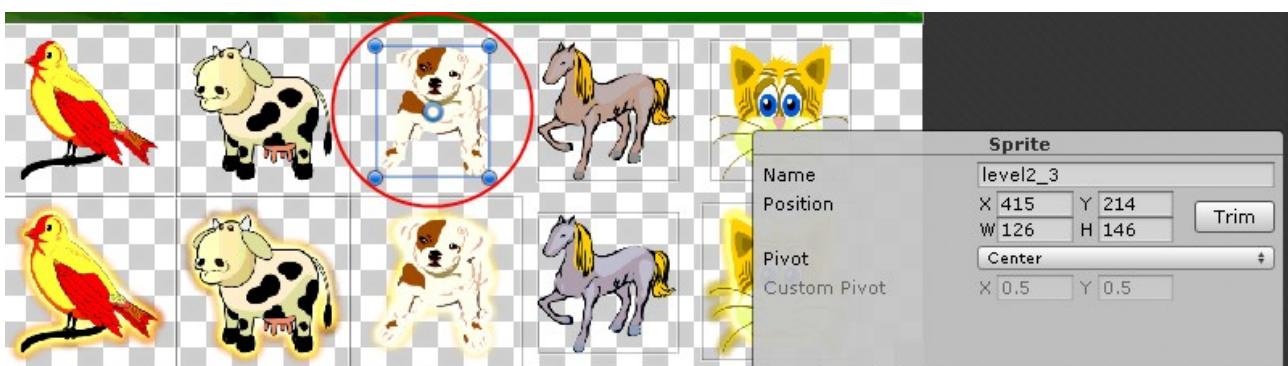
Step23: Select the glowed dog from the bottom:



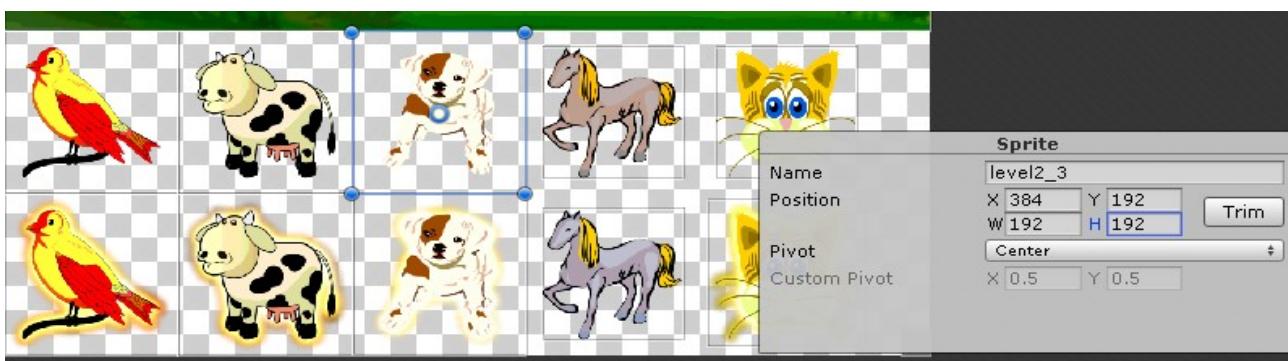
And change its values to: X=384, Y=0, W=192 and H=192, the result will be:



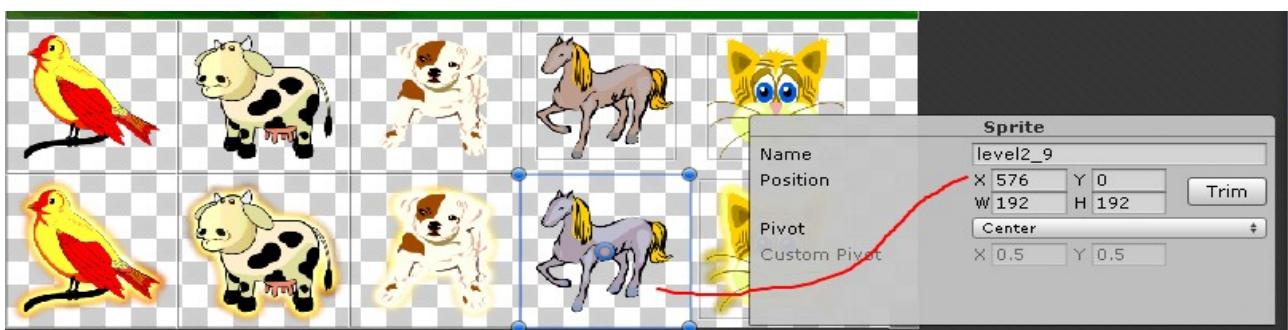
Step24: Select the second dog from the bottom:

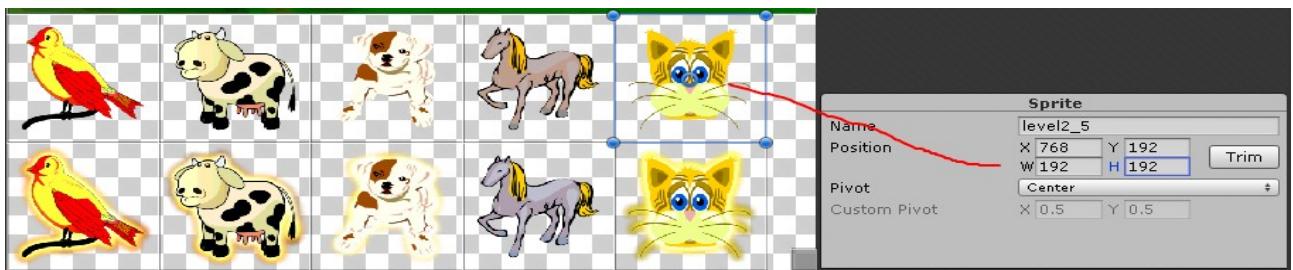
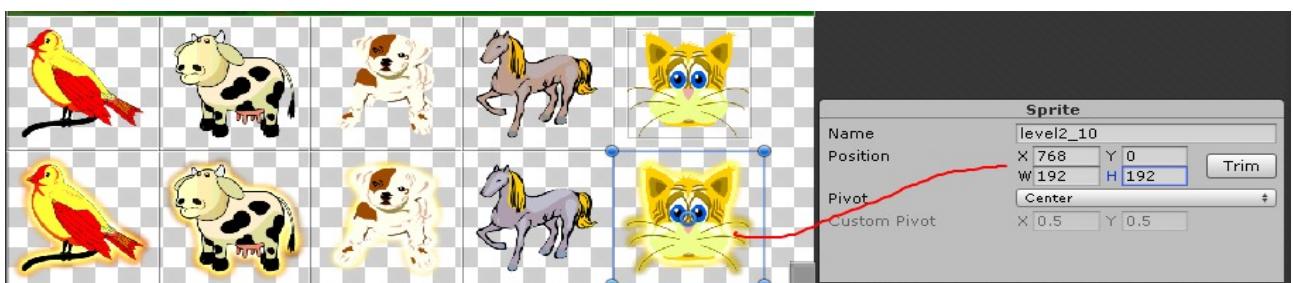
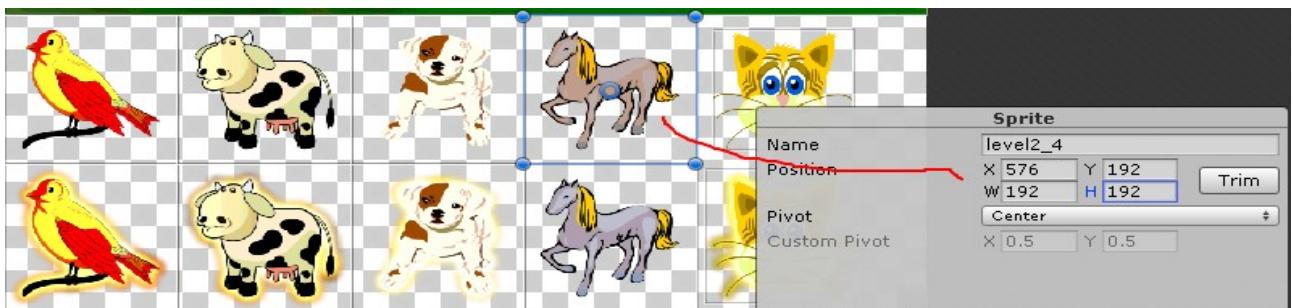


And change its values to: X=384, Y=192, W=192 and H=192, the result will be:



Step25: For the next four animals i'll just show (i guess from previous steps you know how to change these values) the results:



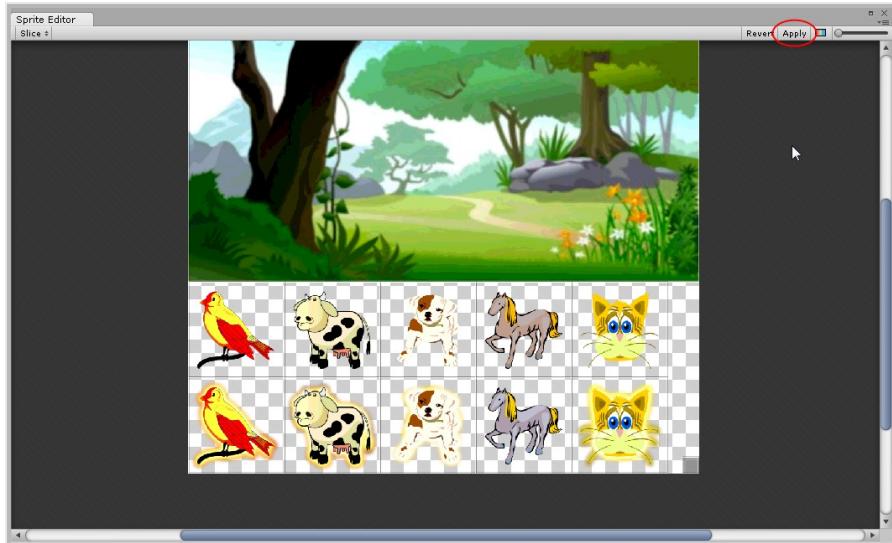


Step26: It has left only the sprite for delimiter object. Let's select it:



All we need is to have a 32x32 slice for that sprite, so its position doesn't matter much. But need to ensure that only grey are selected, so maybe a little adjustment must be done with X=992.

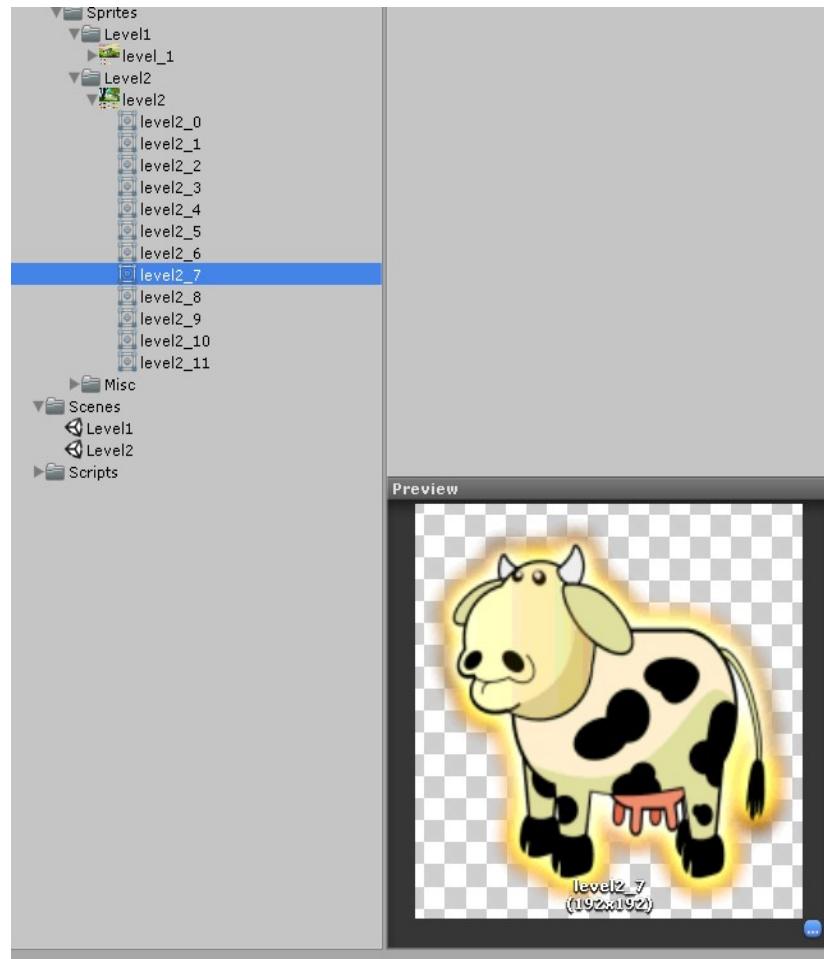
Step27: After all the changes are done click on Apply and close the Sprite Editor:



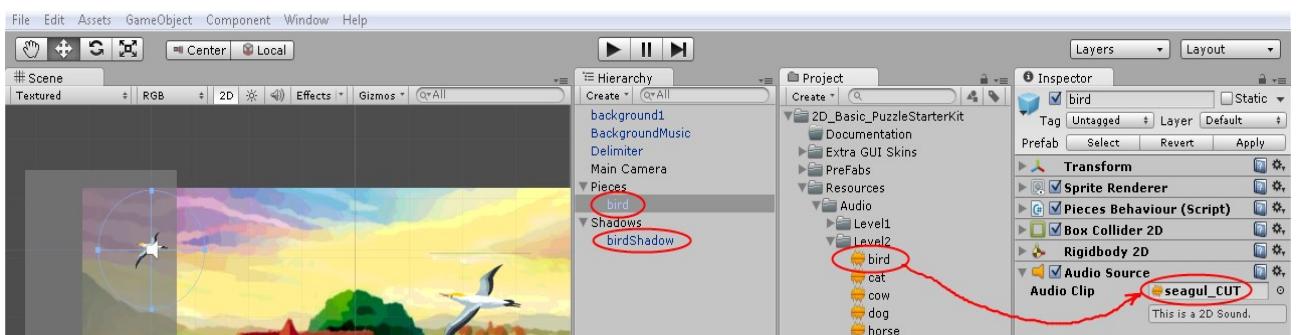
Step28: Now the level2 sprites can be utilized from their spriteSheet:

Previous level2 sprite now is sliced into 12 separated sprites.

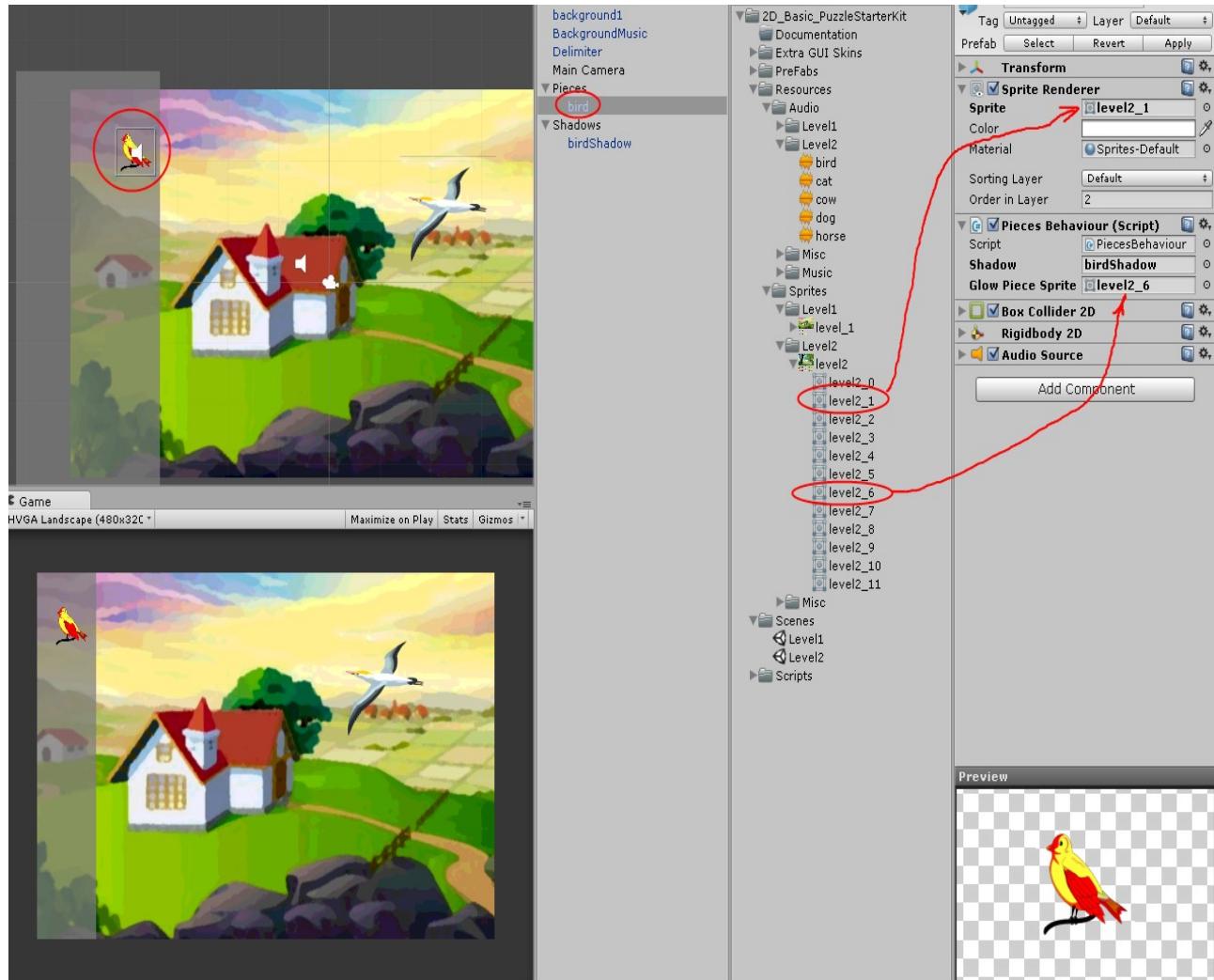
Step29: If you click on any sprite you can see it in the Preview area:



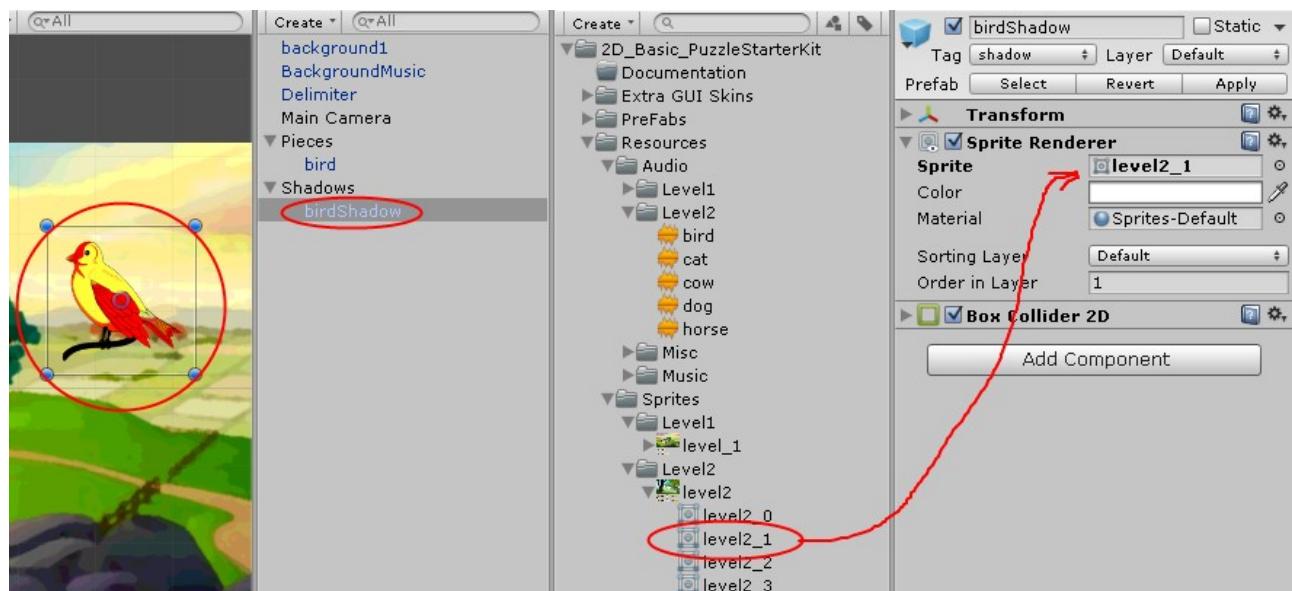
Step30: For easy to mantain controls on following changes at first we remove all the pieces and their shadows from Level2 scene and will let just one piece and one shadow (in our case we let the bird). Then we put bird piece on top left corner of screen to let other objects more vertically room. Also will change in the Inspector the audioclip from audiosource to "bird" clip from Audio/Level2.



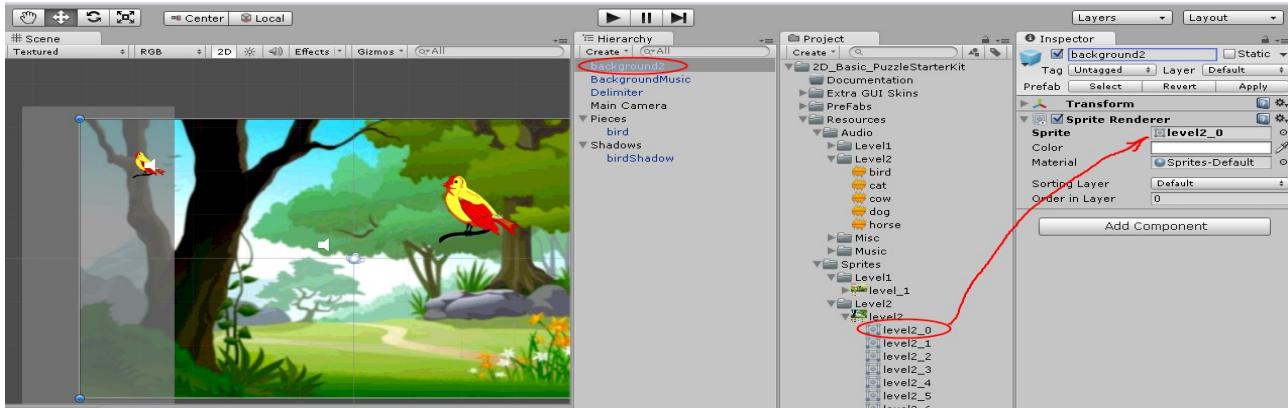
Step31: Will change the bird piece sprite and glow piece sprite with the ones from Sprites/Level2:



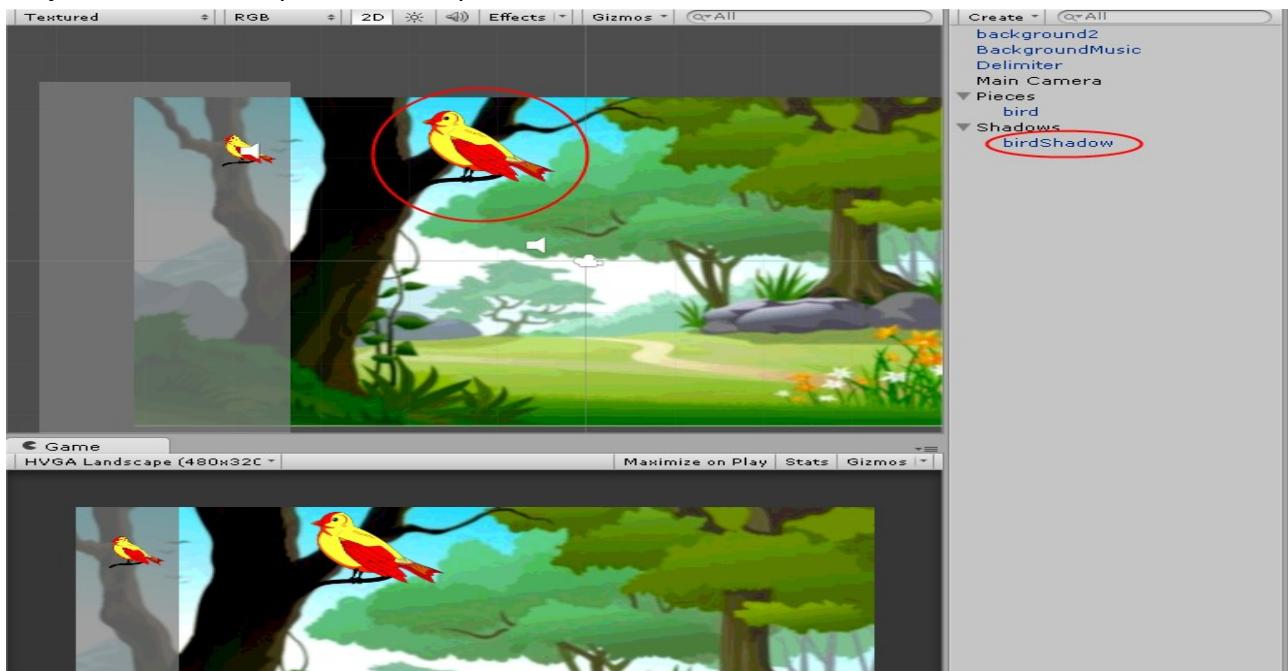
Step32: And change sprite for birdShadow too:



Step33: I forgot about background so let's change this sprite right now and rename it background2:



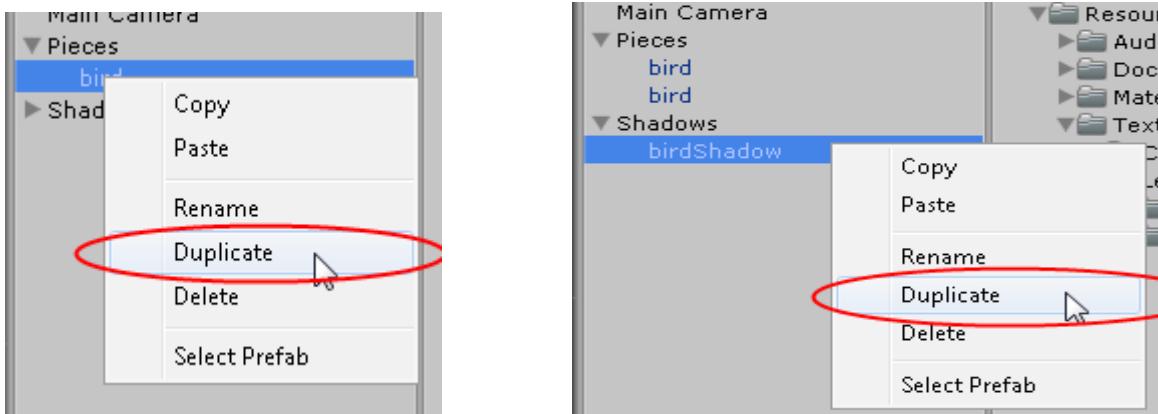
Step34: Now we can proceed with position of the birdShadow:



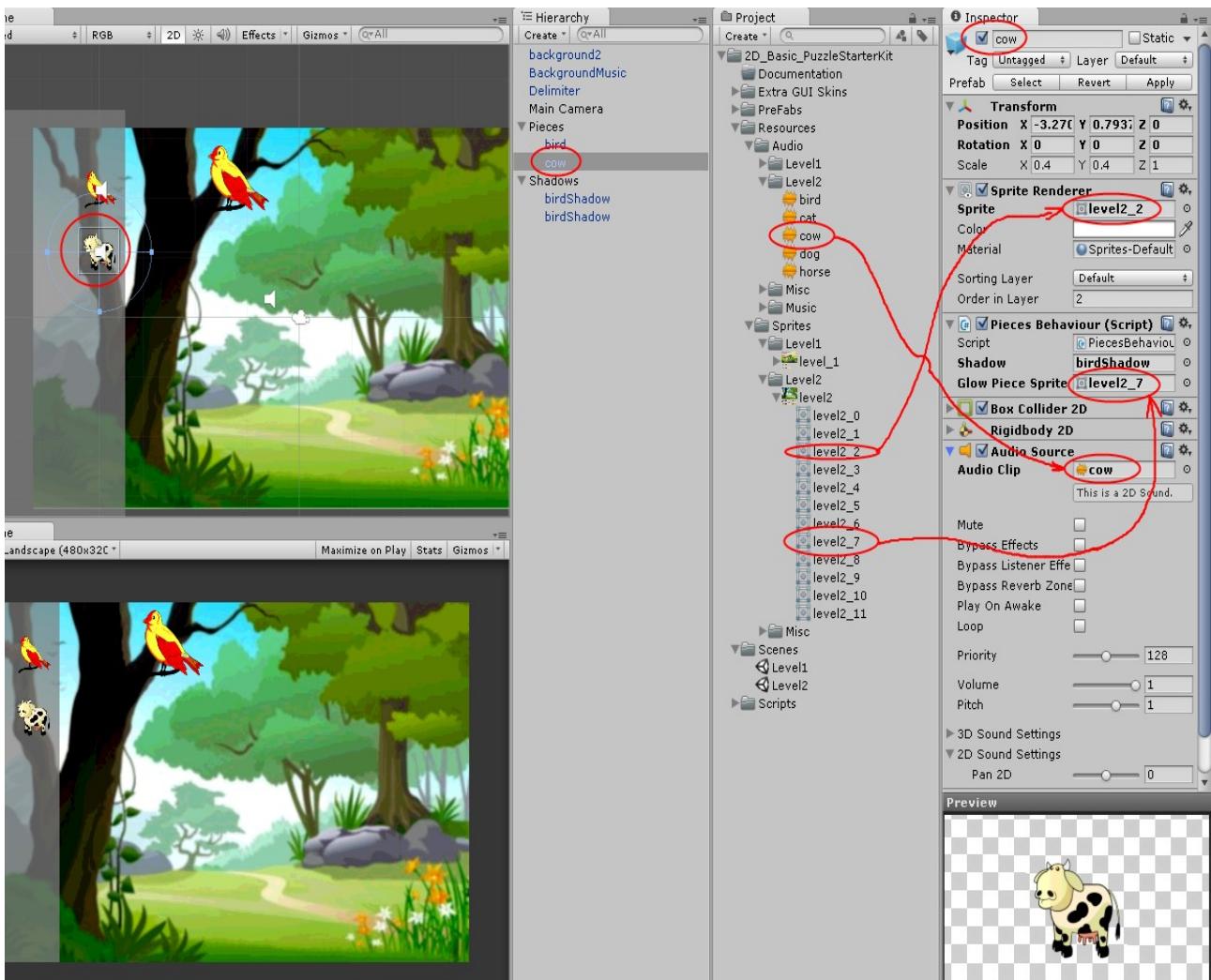
Step35: If the shadow is too large for our background we can make it a little bit smaller in the Inspector. And after that we can hit play game to see whole result.



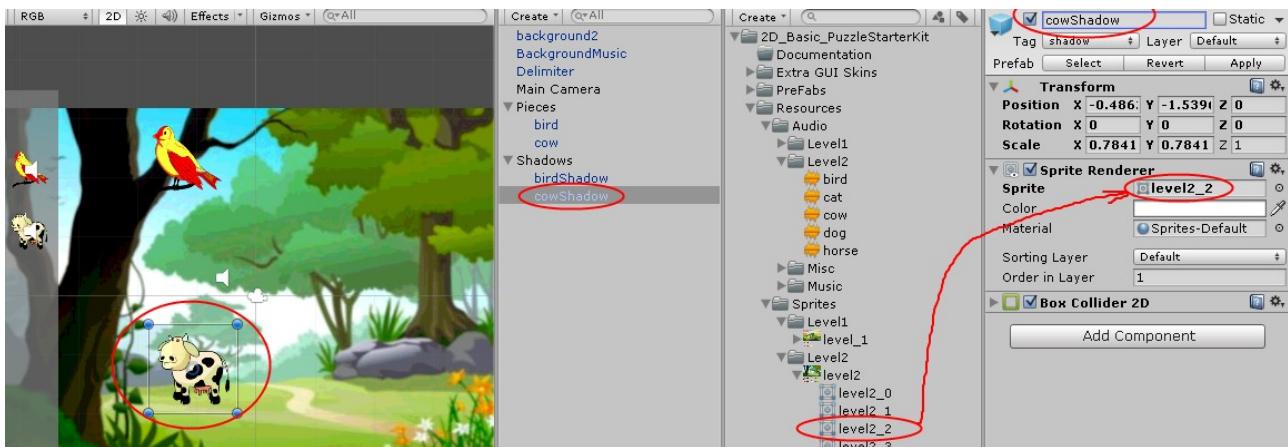
Step36: Now will start to duplicate our bird and change the duplicated object to became one of desired animals in our case. We'll do it four times so at the end we'll have five pieces and five shadows.



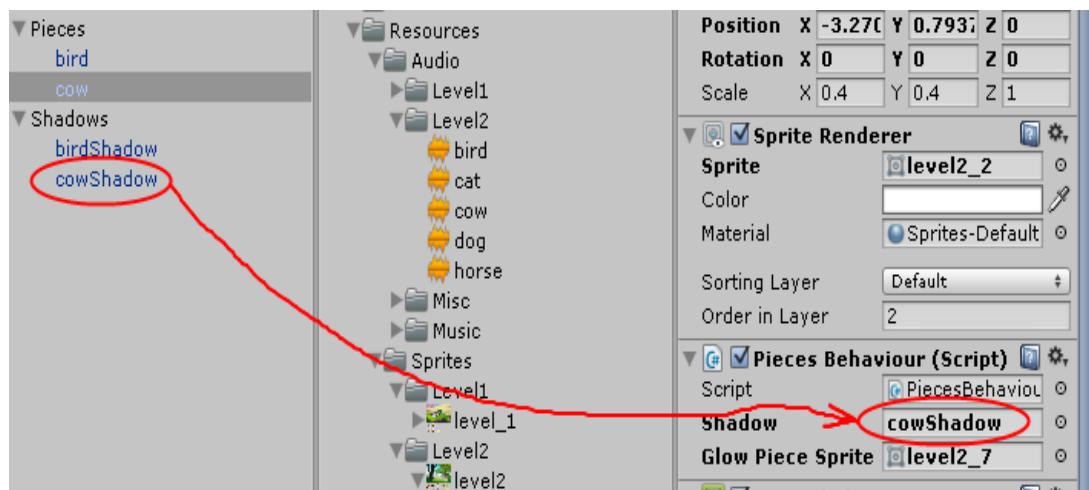
Step37: Duplicated “bird” piece we'll change it in the Inspector: rename as “cow” and drag and drop cow AudioClip and cow sprite, cowGlow sprite as it is shown in the following screenshot:



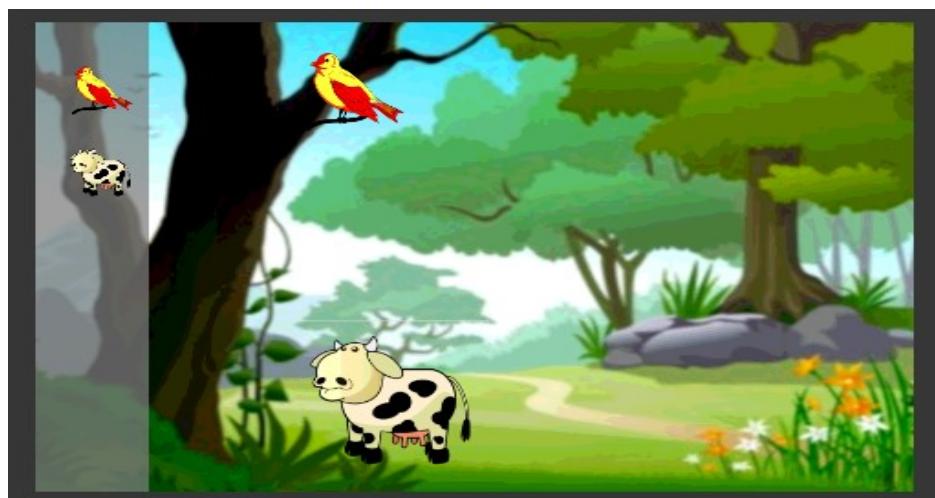
Step38: We'll rename duplicated “birdShadow” into “cowShadow” and change their sprite too.



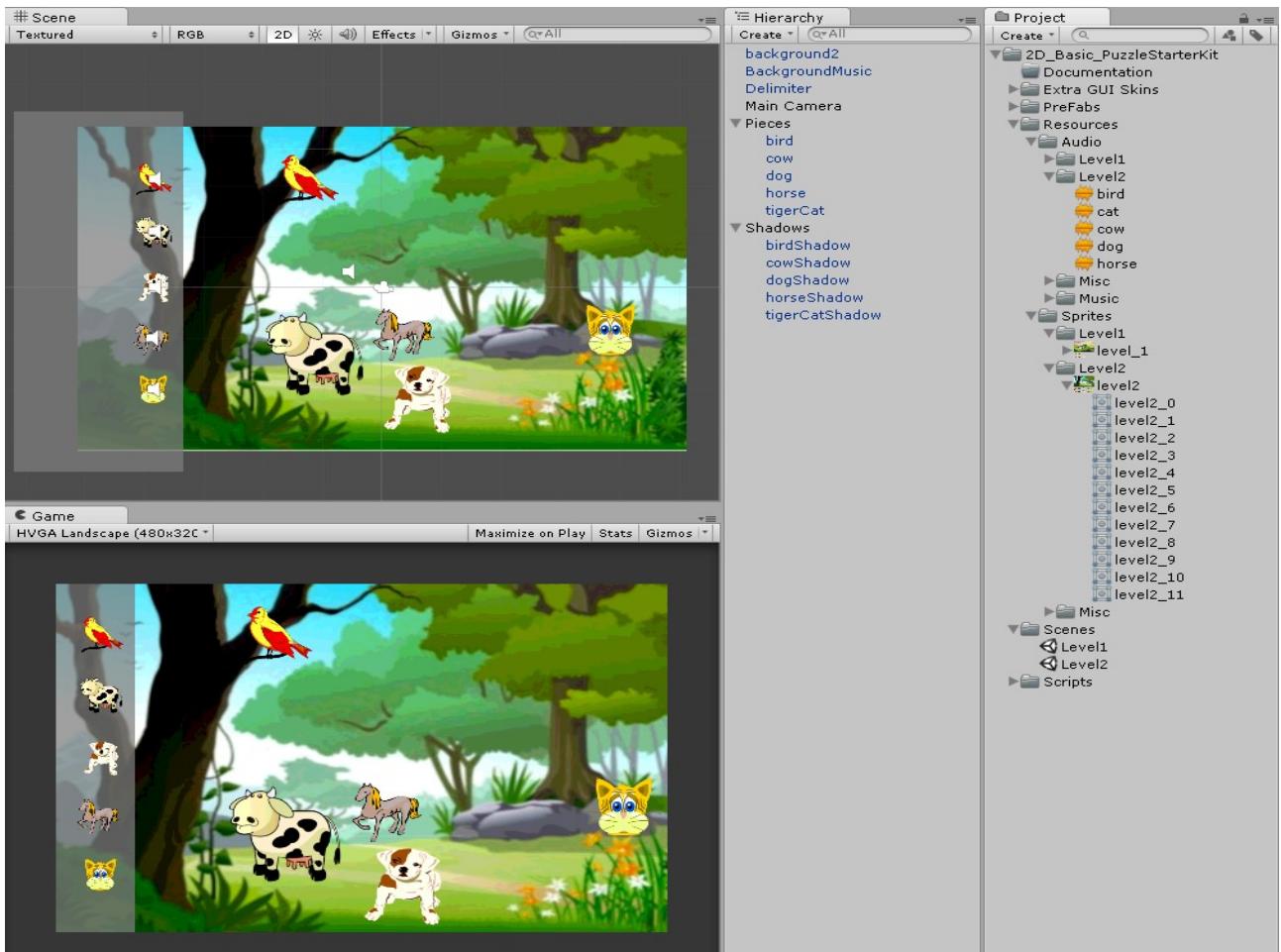
And after selecting again “cow” piece we simply drag and drop “cowShadow” object onto Shadow parameter from the Inspector.



Step39: Now we have other two different objects on Level2 scene: cow and cowShadow. You can position these objects (sprites) anywhere. And once again you can change their size too if you like the cow to be bigger than bird , for example. I suggest to position them like this:



Step40: Now we going to repeat steps 36, 37, 38 and 39 for the remaining objects. Take your time and pay attention and the result it will be something like this:



Now you'll have another level (level2) for your kids puzzle. Remember when you'll finish level3 to write Level3 scene name in the level2 MainCamera's Menu script:NextLevelName property (in the inspector).

VERSION UPDATE 1.1.3 - Created by BestSoft © 2014.

This updated package was developed to illustrate the GUI optimization. For any problems or comments email bestsoftapp@gmail.com

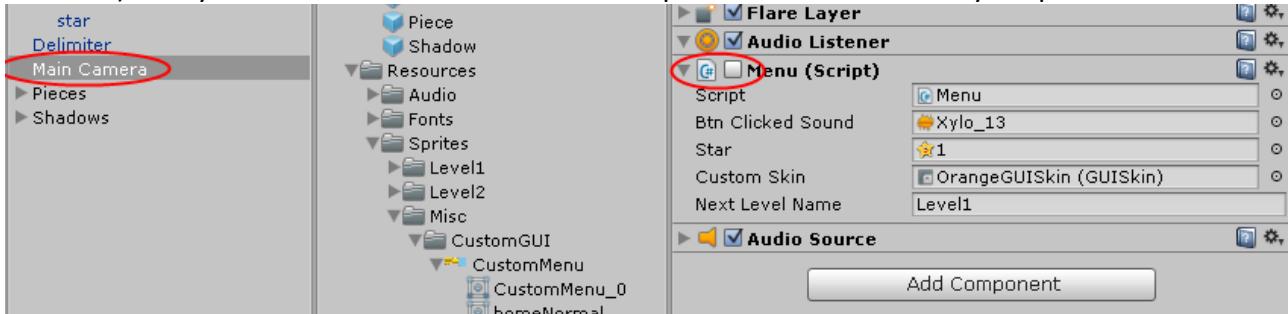
THE UPDATE

For further optimizing your mobile application it will worth to mention that in the end you'll reach to the GUI optimization. If you don't need an verry intensive and heavy GUI oriented application then this package could be of great help:

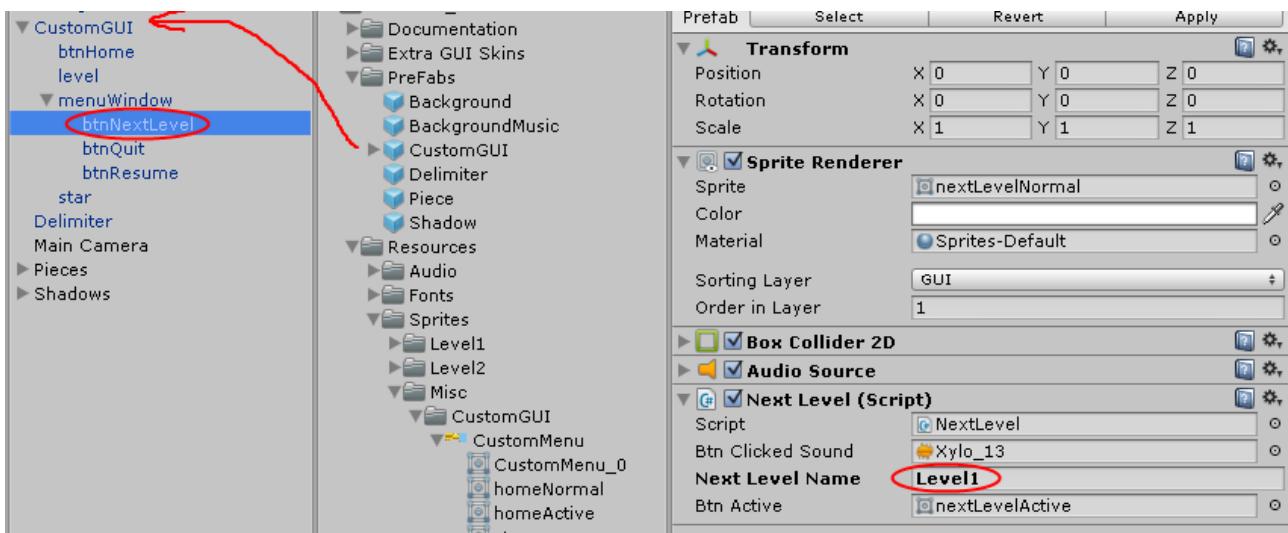
The updated package contain:

- 1) One more scene: "CustomMenu". In this scene we've crafted our custom menu.
- 2) One more script folder (under the Scripts folder): "CustomGUI" with six C# scripts:
 - Home.cs – attached to home button this script reveal or hide the main menu window.
 - MenuWindow.cs – attached to menuWindow object initialize it as invisible at runtime.
 - Nextlevel.cs – attached to next level button. Will load the desired level when button is clicked.
 - PlayerScore.cs – attached to level (score) label. This script will decide when the current level is clear or not.
 - Quit.cs – attached to quit button, handle the exit for user (quit the game).
 - Resume.cs – attached to resume button in case that home button is clicked accidentally.
- 3) One more spriteSheet (named CustomMenu) , under the Sprites/Misc/CustomGUI folder. The sheet is treated like the other two sprite sheets presented in this documentation.
- 4) One more prefab "CustomGUI". This can replace the old Unity standard GUI with just few adjustments:

a) First you'll have to deselect Menu.cs script from MainCamera in your puzzle Level:



b) Then you just drag and drop the CustomGUI prefab to your scene and set the name of next level:



Mention: you'll notice that pieces are on Piece layer, shadows are on Shadow layer and CustomGUI is set to be on GUI layer. That way we can manage more comfortable the depth of all items in the scene.

We reach the end of our presentation, hope you'll enjoy this package. If something goes wrong, if you don't understand something from this presentation or you have any ideas to share please feel free to write to us: bestsoftapp@gmail.com.

Thank you for purchasing our product.

Kind regards,

BestSoft Team.

PS: if you like this package please support us with a positive review in the assets store. Thank you.