**Assets Lab**

Assets are any external media object being used by your program such as sound effects, images, or fonts.

This lab will deal with loading assets into your program so that you can use them.

**Enemy + Player Sprites**

Go to the folder called AssetLibrary inside of Documents -> DigiGirlzGame on your computer. This folder filled with art-assets to use in your game. Browse through the characters folder and pick one that you like. Each character has sub folders “Walk,” “Stand”, “Hit”, “Die”, etc… Within these are the animation images for those characters. Write down the charcter you pick below

* What character did you pick? 5 pts \_\_\_\_\_\_\_

For the character animations you’ll find a sequence of 9 images animating that character in the pose specified by the folder name (Walk will be walking).

Open up the Assets.xcassets folder of your project. There are two folders in your Assets.xcassts involving the player. One is labeled PlayerAttack another named PlayerMove. You should see something similar to the following.

A screenshot of a cell phone

Description automatically generated

You will see items inside of this folder. These are called imagesets. Image-sets hold the different resolution sizes of an image so that images appear crisp on different screensizes. For the purpose of our application we will only provide a 2x image.

You will see that the imagesets are labeled PlayerAttack0 – PlayerAttack9. Now take the corresponding Attack image from your AssetLibrary. 0 -> PlayerAttack0, 1 -> PlayerAttack1, etc… and drag it onto the 2x spot on your imageset. Repeat this for all the playerAttack and playerMove images. (If you miss any frames your animation will look weird)

A screenshot of a cell phone

Description automatically generated

When you are done, run your program. You should see that your player will run when you hit the run buttons and will do the attack image when you tap.

(Teachers Note: Are you wondering how this works? The reason is, some code was already hooked up before hand in order for the program to know which imagesets to use. The code to do this is relatively simple and lives in the Player.swift file. Towards the top you will see playerShootAnimation and playerMoveAnimation variables defined with values set to a series of “SKTextures” that use the images we just filled in on our ImageSets. (If you haven’t done the variables or functions labs yet, this may be a bit confusing, but its actually not too bad!) Then in the movePlayerLeft, movePlayerRight, and playerShoot functions you can find a statement to run the animation sequence. If you wanted you could define more animations for when the player dies, etc, and then set them up to run when those events happen.)

* Setup the PlayerMove Sprites with a character of your choice 15 pts \_\_\_\_\_\_\_\_
* Setup the PlayerAttack Sprites with a character of your choice 15 pts \_\_\_\_\_\_\_\_

Now lets do the enemy. There is a folder for called Enemies in your AssetLibrary. Now you can pick one of a series of different Enemies. Find one you like. Inside you will find animations for Fly, Die, Idle. We will be using the Fly animation. Repeat the same process of loading your chosen images into the corresponding imagesets for the enemy, you’ll notice that the enemy fly animations only have 4 frames (this was how the artist made them). There is also a Die animation, this animation has two parts and is 8 frames. Repeat the same process to pick a death animation for your enemy.

* Setup the EnemyMove Sprites with an enemy of your choice. 15 pts \_\_\_\_\_\_\_\_
* Setup the Enemy Die Sprites with an enemy of your choice. 15 pts \_\_\_\_\_\_\_\_

**Background**

Find the Backgrounds folder in your AssetLibrary. In it you will find a series of 40 fun backgrounds you can choose from. Pick one that you like for your level. Find the background imageset in your Assets.xcassets directory. Drag the background you pick into the 2x of this imageset. (Note: The resolution of the background is very low, I personally would have preferred higher res images, but couldn’t find enough good options of the same size. Sorry!)

* Setup the Background for your level. 15 pts \_\_\_\_\_\_\_\_

***BONUS EXERCISE – Logo: (Do this last)***

Logos are fun! They are what you see in your notifications and home screen. Typically a logo gets designed in some sort of design software like Photoshop or Illustrator, but in our case I’ve made several sample logos for you to use using a tool called LogoMaker. You can find LogoMaker on your iPad if you want to make your own custom logo. Otherwise use one of the pre-made ones in your Logo folder in the AssetLibrary

**How to load a logo into your Game**

1. Find the AppIcon imageset in your Assets.xcassets directory
2. Drag each corresponding image asset to the size specified. (There are a ton of these! And be careful, you have to use the right size file or it WILL NOT WORK. For example, one of the items is called iPhone Notification 2x. The size is listed is 20pt. This means 2x is 40pt, and 3x is 60pt. For iPhone Settings 29 pt 2x, you need to use the 58 pixel image, for 3x you need the 87 pixel image.

**How to use LogoMaker**

1. Open Logo Maker
2. Hit “Create” in the bottom left
3. Choose “BG” From the bottom menu. (Pick a background)
4. Choose “Text” From the bottom menu (Write DigiGirlz 2020 or some other name you like)
5. Choose anything else you want from the bottom menu.
6. Hit the Export Button in the top right
7. Hit the “Save to gallery” button.
8. Open your photos.
9. Select your Logo.
10. Share with “Airdrop”
11. Airdrop to your computer. (Find your computers name by clicking the Apple icon in the top left of your mac. Should read “Log Out COMPUTER\_NAME”
12. Visit <https://appiconmaker.co/>. Drag your logo to the “Choose File” button
13. Hit Generate
14. Hit Download iOS icons zip
15. Load the icons into your game

* Add a new logo to your game 20 pts \_\_\_\_\_\_\_\_

(Teachers Note: There are lots of awesome open source and free content you can use for your games if you aren’t an artist. The images for this project were from <https://www.gamedevmarket.net/>)