Game Design and Development Report

Names and UB Numbers: Nashme Al-Hadid (18021211) - Amir Sabbagh (17011921)

Game Title: Duck Escape

Module Code: GAV5002-B

Module Title: Game Design, Programming and Development

Academic Year: 2019-20 - Semester - 1

**Table of Contents**

[1. General Description of the Game project, Aim and Objectives 3](#_Toc26982705)

[2. Research and Analysis, Subject Review 3](#_Toc26982706)

[3. Game Overview: Game Idea and Game Play 3](#_Toc26982707)

[4. Themes and Game Flow 3](#_Toc26982708)

[5. Game Features 4](#_Toc26982709)

[6. Game World and Game Layout 4](#_Toc26982710)

[7. Game Interfaces 5](#_Toc26982711)

[8. Game Characters, and Game Weapons 6](#_Toc26982712)

[9. Music and Sound Effects 7](#_Toc26982713)

[10. Game Management and Development Plan 7](#_Toc26982714)

[11. Enemy AI and Behaviours 7](#_Toc26982715)

[12. Game Performance 8](#_Toc26982716)

[13. Project Evaluation, Game Testing, Individual Appraisal 8](#_Toc26982717)

[14. User Instructions 8](#_Toc26982718)

[15. References (replace the followings with your relevant references) 9](#_Toc26982719)

[Assets 9](#_Toc26982720)

[Sprites 9](#_Toc26982721)

[Sounds 10](#_Toc26982722)

[Guides & Tools Used 11](#_Toc26982723)

# General Description of the Game project, Aim and Objectives

The game project for our semester will be a basic 2D action platformer in which the player has to reach the end of each level all while fighting their way through enemies, completing difficult platforming jumps, and collecting coins on their way. Our aim for this project is to try to learn the best way to tackle any situation we encounter whilst developing a 2D game and be able to problem-solve errors we encounter to ensure we finish our project before the deadline and deliver a bug-free game with smooth, fun, and exciting experience.

# Research and Analysis, Subject Review

For making an action platformer game, we looked at different games in a design perspective. For example, games such as Dead Cell, Duck Game, and Super Mario. For the theme of the game and the general art direction, Dead Cell’s level design and sprite work was an inspiration which we took into account when designing and planning the levels. Creating good platforming, which also needs good level design was researched through games like Super Meat Boy, and Super Mario, which they succeeded in creating an enjoyable platforming experience. Lastly, for the feel of the 2D weapons we created, we looked at Duck Game’s gameplay footage and took some notes to make the shooting mechanics better. After we did enough and sufficient research for our game idea, we planned to put them in place.

# Game Overview: Game Idea and Game Play

The general gameplay idea we had in mind was a 2D platformer, with both emphasises on platforming and shooting. At first, we were at a loss for which to focus on, but when we reached the level design, we were easily able to focus on making the best of both worlds in each level. The gameplay features the player going through levels crawling with enemies, traps to avoid, and coins to collect. Each level, the player is required to pick up a weapon in order to help him/her fight their way against the enemies or break through blockades. Throughout playing the level, the player will encounter a lever that will open the gate, allowing the player to finish the level, and going to the next one.

Clearly from the in-game assets, the player’s character is a duck (hence the name Duck Escape). This gave the game a silly feeling, despite the serious atmosphere. This also allowed us to find interesting cartoon pictures of ducks for the background of our game menus.

# Themes and Game Flow

When looking for our game’s theme, the main inspirations were the game assets. They had a “underground dungeon” feel to them. This gave us the idea of the character escaping some form of underground lab complex, hidden away in a dungeon. The flow of the game works well with how the movement is designed. Firstly, the most important part of any platformer video game is the character handling within the levels. The platforms have been put in such a way that they do not require too much precision and skill, nor are made easy. However, as the player progresses through the game, the platforms get harder and harder.

Secondly, the weapon mechanics have also been tuned and balanced to provide a challenge when encountering melee or ranged enemies. This gives the player more options when encountering enemies and can decide to destroy them or not, which allows for the levels to be played in many different ways. The game generally feels fast-paced, allowing for a more adrenaline-filled, action-packed gameplay, though the player can take the levels at their own pace as they wish.

# Game Features

We added a plethora of game features in order to spice up our game and give the player different experiences in each level through different gameplay loops, things to play for, and different ways to play. The main thing we worked on were the enemies. We added different types of enemies, from stationary ones, to patrolling ones, even ones that shoot you down. This gives the player different challenges when facing them, and different ways of engaging them. We also implemented coins into the game. Each level is scattered with coins for the player to collect, which gives the player more gameplay options when traversing the levels. Some can also be placed in areas which rewards the players with good platforming skills.

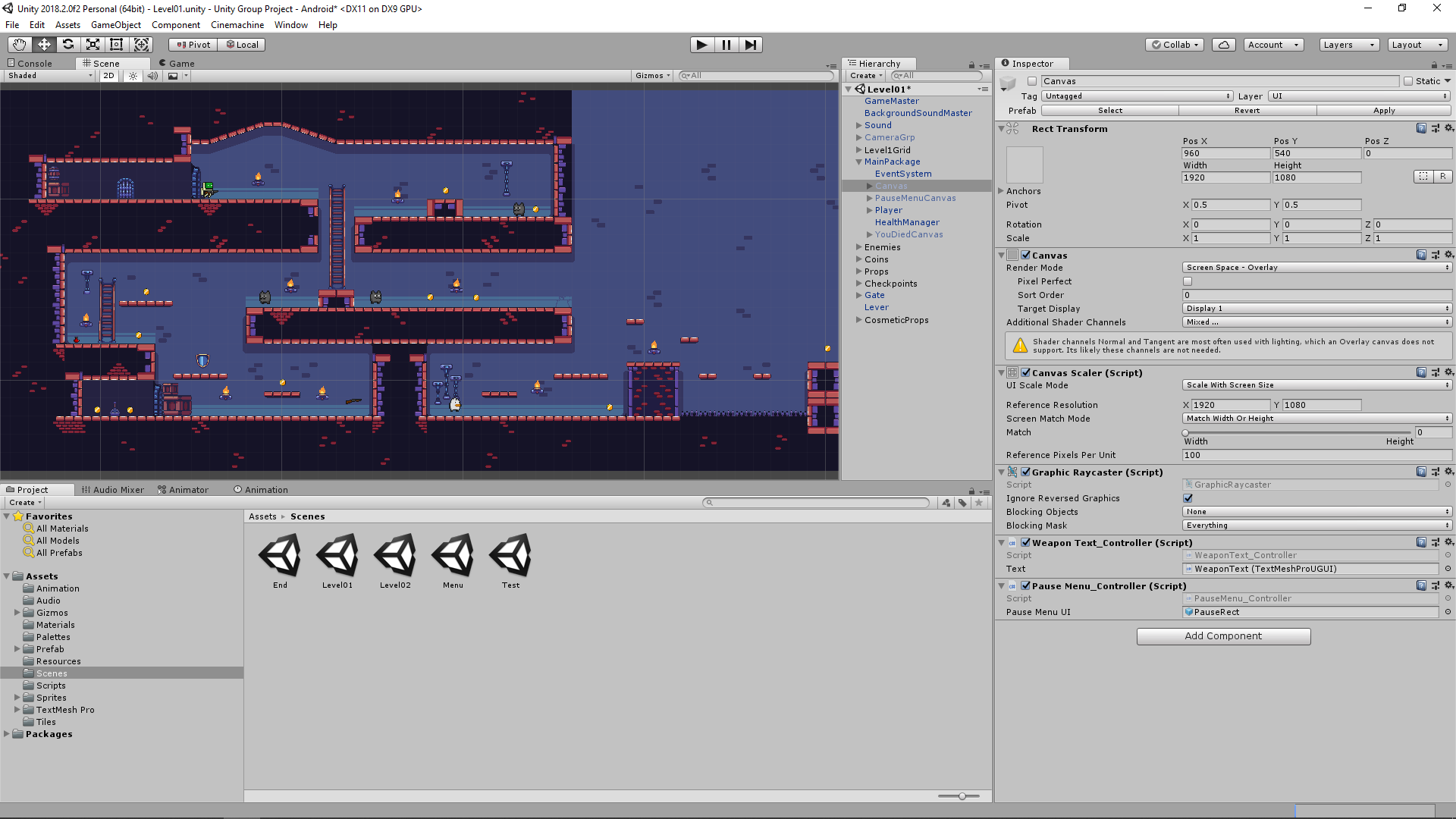
We also added checkpoints, which allows the player to restart at the position it is located at in case they lose. That way, the player wouldn’t have to jump back to the beginning to complete the level, which makes it easier for the player to enjoy the game without going through the hassle of repeating parts they’ve already played.

Despite the healthy array of enemies in the game, we decided to add scripted events; things that happen in the game when the player enters a certain area. From enemies falling through pipes, to a mini-boss fight that blocks your path, these give the game a new breath of life to spice up the gameplay and surprise the player.

# Game World and Game Layout

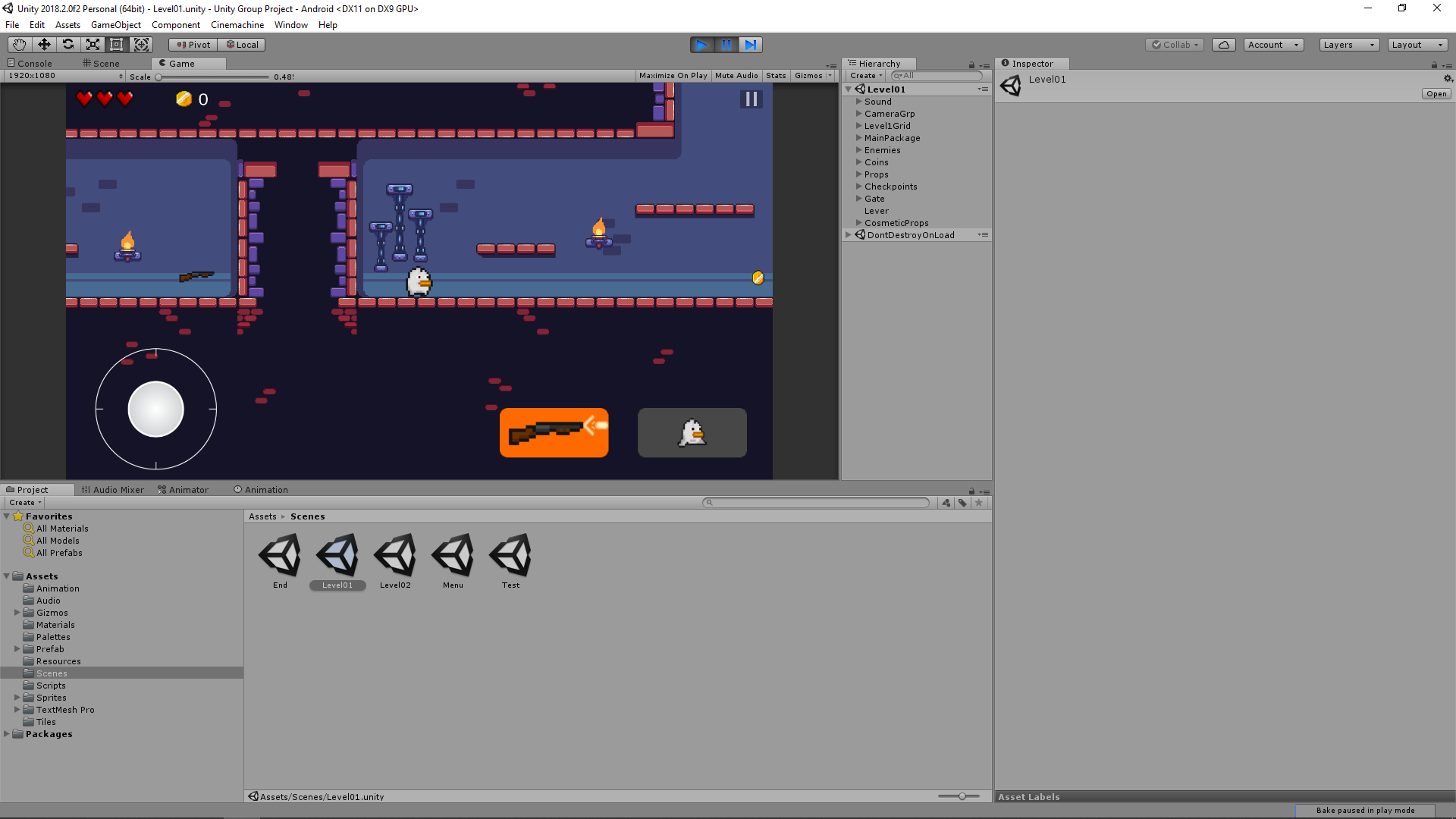
The world that the game is set in is a dungeon-themed area which the player needs to traverse in order to escape each level. The story revolves around the duck trying to escape an underground laboratory filled with clones of experimental cat monsters that the evil scientists were working on. The duck has to fight its way out of the lab and stop the masterminds behind such evil ingenuity.

When creating the game world, rather than using normal sprites for creating an environment the levels are set in, we used tile mapping which improves on the level creation flexibility. Tile mapping gave us the chance to easily prototype the levels without committing and change the layout on the go. The overall layout of the levels are experimented and tested on in order to have unique level layouts. We touched on backtracking, which allows the player to turn back at parts of the level to find hidden areas and coins. The other thing that we focused on while creating the general layout was to make sure that the platforms and the ground were placed accordingly to the player jump height so that the player would be able to reach it properly. With tile mapping, it was easy to experiment with making more difficult platforms to jump on, which gave a range of different jump difficulties.



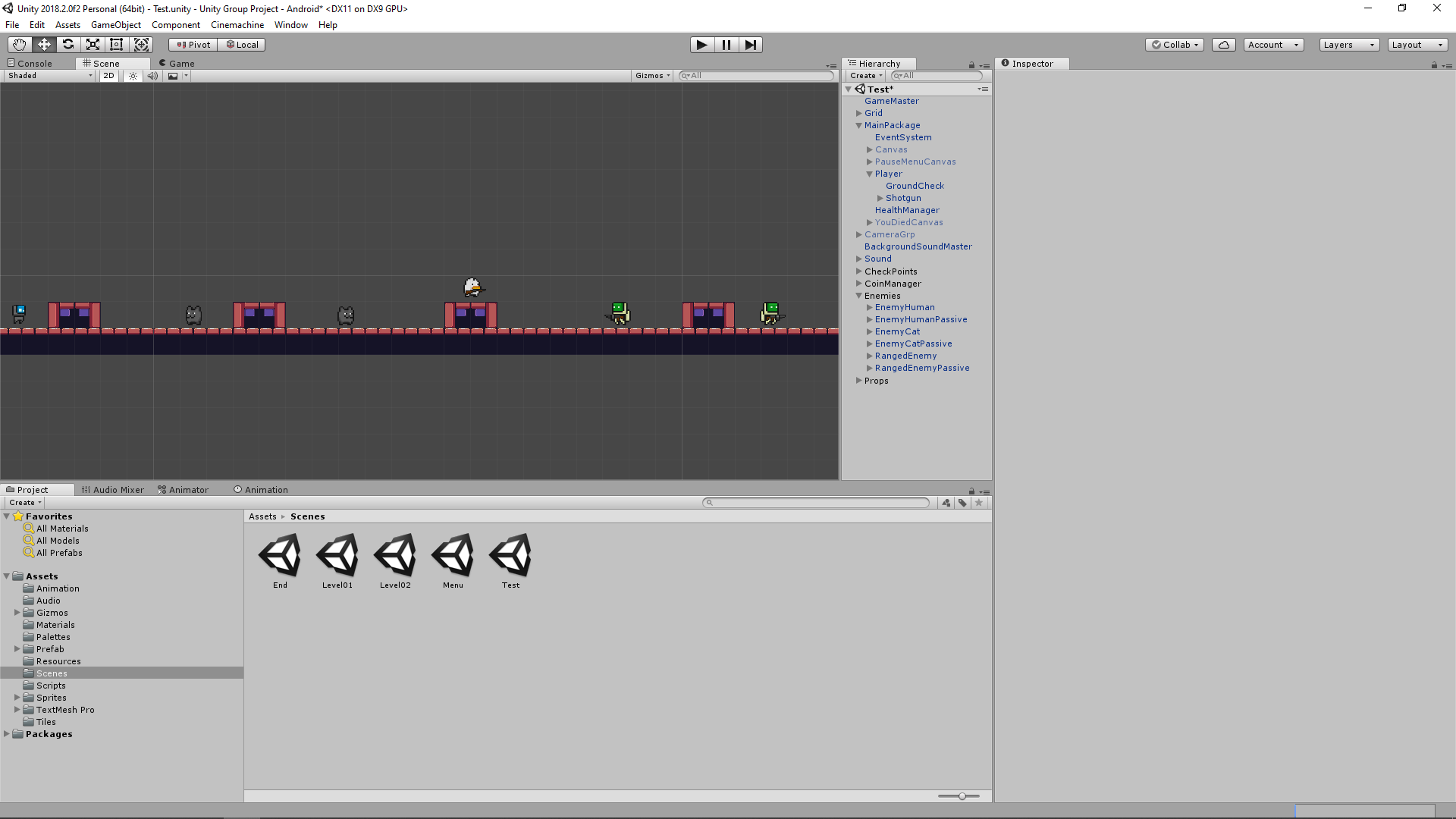
# Game Interfaces

The game that we made as a group was targeted for a mobile phone platform, so we approached the interface design differently from how it would be if it were on PC or any other device. Fortunately, Unity’s user interface (UI) is also compatible with touch controls so creating the pause menu and the main menu was a straightforward procedure. We created two buttons for jumping and shooting and were placed on the bottom right corner of the screen and are displaced evenly from each other to make it easier for the player to use the controls. We decided to go for an easier layout for the buttons, adding a small picture indicating what each button does. We also added a button for climbing the ladder which only shows up when the player is near one and disappears once the player has moved away from it, this is made in such a way as to not confuse the player. The hardest thing we encountered once we started making the UI was creating the joystick controls. The reason being is that the joysticks found in the asset store do not match the same version we are using for the game. Nevertheless, we had to follow guides on the internet to implement it from scratch by a script and test it via Unity Remote to make sure it was snappy and felt right with how the player is moved in-game.



# Game Characters, and Game Weapons

When thinking about what our game was meant to be, all we could figure out was the it has to be an action platformer. When searched some free assets, we managed to find suitable characters that we were able to use for our enemies and for the player. After some decision making, we settled to have 8-bit duck to be the main character of the game. We used the rest of the other characters as enemies, which included a cat, and two human scientists. Some of the enemies roamed around platformers, searching for the player and chased the player down. Some were stationary, and some were even equipped with a gun, anticipating the duck’s arrival. The main weapon for the player is a semi-automatic rifle that the player has to pick up in every level. The weapon has no ammo capacity or reserve, it is infinite. Adding this could give the game unnecessary detail and ruin the flow of the game.



# Music and Sound Effects

We planned and worked together to find sound effects off of the internet that fit the theme of the game, and that was pretty successful. The sound effects worked wonders within the game and did not sound awkward or extra while playing the game. We also tried to use some 8-bit sound effects to fit the game’s assets, which added a more retro-feel to the game.

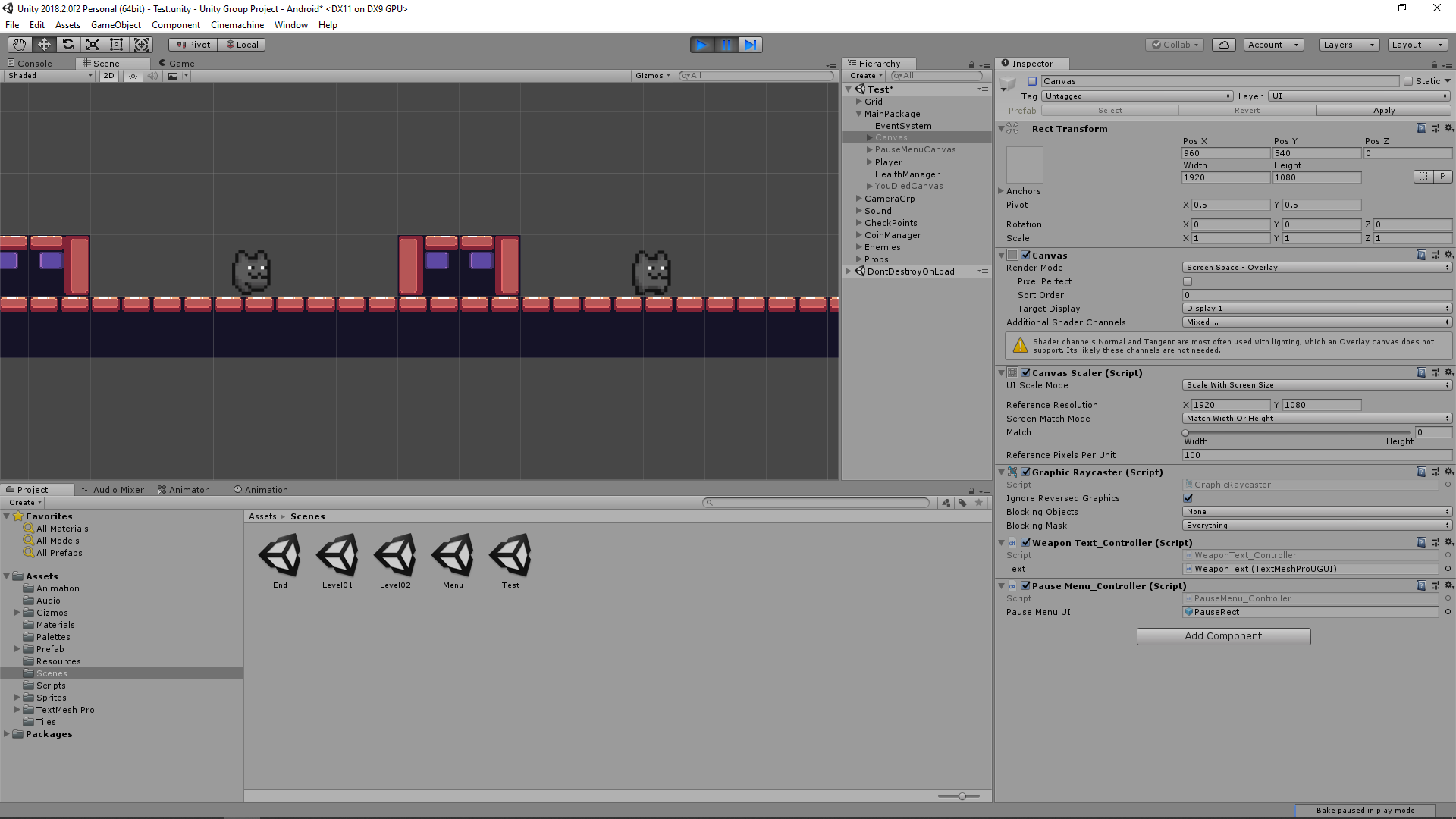
In terms of music, we decided to go Halo theme music for the main menu. For background music of the game, we decided to use the Black Ops Dead Ops Arcade music as it fits the style and theme of the game.

# Game Management and Development Plan

Before we started making the game, we planned the game in different stages to manage what to focus on first in order to finish the core of the game early. The other important factor we were planning was time management. In our diary, we dedicated time for every stage of development and created a section for optional features. As a group, we shared tasks on a weekly basis and reviewed our work each week. The development plan for our game went through many changes as we progressed with the game development and some initial main features became secondary and vice versa. Overall, we are very happy with our plan because we managed to finish most of the game on time because of our planning.

# Enemy AI and Behaviours

Objects such as the player, crates, and the weapons were fundamentally easy to make because they followed a straightforward procedure. However, the enemy AI was by far the most complex thing we implemented in the game. There are four main behaviour scripts for the enemy AI which are called idle, patrol, follow, and shoot. For detecting the player, we had to use ray casting which was proven to us difficult to make it work in most scenarios. There are two raycasts attached to the enemies, one in the front, and one in the back, which are used to detect the player on either side. We set a specific range for the ray casts which if the player is within it, the enemy will change behaviours. For the enemy types that are patrolling, there’s an extra ray cast which checks the ground so that the enemy doesn’t go out of its boundaries. The other main issue we had was to flip the enemy sprite according to the direction which is either following or patrolling. We had to redo the scripts multiple times in order to make this work in all scenarios, which in the end, we managed to simplify the scripts of unnecessary code that was causing errors.



# Game Performance

As we were finishing setting up each level, we would go multiple playthroughs in order to see what worked and what didn’t. We also tested them on the mobile phone afterwards to check if the performance was stable. It ran perfectly on the mobile phone with no lag whatsoever. We managed to have a full game experience without cluttering up the game screen and adding unnecessary objects for the player to view and loading the levels were pretty instant. We were happy with how the game turned out on the mobile phone as we expected it to be slower than we hoped for.

# Project Evaluation, Game Testing, Individual Appraisal

The overall final product of our game is really close to our initial version and plans we intended to have. However, we had to test the game thoroughly on the mobile phone platform. Most of our initial testing was on the computer and we had to make sure that the controls and the layout felt just as responsive on the mobile phone as well. One of the main things we have done for making improvements to the game is that we used the Unity Remote to test the game on multiple occasions. Also, we created an early build of our game on the phone to give to other colleagues to test it and give us additional feedback which we can improve upon.

# User Instructions

The user instructions are pretty straight forward. The controls and objective are outlined in the game and is easily readable. The joystick is used to move the player around the environment. Two buttons are laid out for both the jumping and shooting. A third button pops out whenever the player is standing near a ladder and will allow the player climb ladders. The player has to finish each level by finding a weapon, shooting the lever, and going through the gate. The player is left free to finish the levels the way he/she wants. Killing all the enemies and collecting all the coins is optional and is only added to give the player incentives other than finishing the level.



# References (replace the followings with your relevant references)

# Assets

## Sprites

itch.io. (2019). 2D Pixel Item Asset Pack. [online] Available at: https://startledpixels.itch.io/2d-pixel-item-asset-pack [Accessed 8 Dec. 2019].

itch.io. (2019). NT Like Sprites Pack. [online] Available at: https://jfaexe.itch.io/nt-like-sprites-pack?download [Accessed 8 Dec. 2019].

itch.io. (2019). Gems / Coins Free. [online] Available at: https://laredgames.itch.io/gems-coins-free [Accessed 8 Dec. 2019].

Menus images from Sabage-bu! (Sabagebu! -Survival Game Club! -) anime

Sinestesia (2017). 2D Explosion Animations | Frame by frame. [online] OpenGameArt.org. Available at: https://opengameart.org/content/2d-explosion-animations-frame-by-frame [Accessed 8 Dec. 2019].

UnityAssetStore. (2018). Medieval pixel art asset FREE - Asset Store. [online] Available at: https://assetstore.unity.com/packages/2d/environments/medieval-pixel-art-asset-free-130131 [Accessed 8 Dec. 2019].

Waldo. (2019). YouTube. Available at: https://www.youtube.com/channel/UCPfM\_A90oIk6o7N91OyP2AA/videos [Accessed 8 Dec. 2019].

## Sounds

artisticdude (2011). RPG Sound Pack. [online] OpenGameArt.org. Available at: https://opengameart.org/content/rpg-sound-pack [Accessed 8 Dec. 2019].

Brad Wesson. (2011). CollectCoin. [online] Available at: https://freesound.org/people/bradwesson/sounds/135936/ [Accessed 8 Dec. 2019].

Bungie. Halo 2 Theme Song. [online] Internet Archive. Available at: https://archive.org/details/tvtunes\_7628 [Accessed 8 Dec. 2019].

Call of Duty: Black Ops - Dead Ops Arcade song “Clockwork Squares” James McCawley. (2010). YouTube. Available at: https://www.youtube.com/watch?v=nsIIrMlVSxY [Accessed 8 Dec. 2019].

FromSoftware. You Died. [online]. Available at: https://www.myinstants.com/instant/you-died/ [Accessed 8 Dec. 2019].

Koenig, M. (2019). M4A1 Single Sounds | Effects | Sound Bites | Sound Clips from SoundBible.com. [online] Soundbible.com. Available at: http://soundbible.com/1804-M4A1-Single.html [Accessed 8 Dec. 2019].

Koenig, M. (2019). Rifle Sounds | Effects | Sound Bites | Sound Clips from SoundBible.com. [online] Soundbible.com. Available at: http://soundbible.com/1338-Rifle.html [Accessed 8 Dec. 2019].

LittleRobotSoundFactory. (2015). Hit\_01. [online] Available at: https://freesound.org/people/LittleRobotSoundFactory/sounds/270326/ [Accessed 8 Dec. 2019].

LittleRobotSoundFactory. (2015). Jump\_03. [online] Available at: https://freesound.org/people/LittleRobotSoundFactory/sounds/270323/ [Accessed 8 Dec. 2019].

Myinstants (2010). quack - Sound Button. [online] myinstants. Available at: https://www.myinstants.com/instant/quack/ [Accessed 8 Dec. 2019].

qubodup (2012). Handling Guns. [online] OpenGameArt.org. Available at: https://opengameart.org/content/handling-guns [Accessed 8 Dec. 2019].

timgormly. (2012). 8-bit Explosion2. [online] Available at: https://freesound.org/people/timgormly/sounds/170144/ [Accessed 8 Dec. 2019].

wobbleboxx (2014). Level up, power up, Coin get (13 Sounds). [online] OpenGameArt.org. Available at: https://opengameart.org/content/level-up-power-up-coin-get-13-sounds [Accessed 8 Dec. 2019].

# Guides & Tools Used

Alexander Zotov. (2017). How to add sound or audio effects SFX to Unity 2D arcade game | Very simple Unity 2D Tutorial. (2017). YouTube. Available at: https://www.youtube.com/watch?v=8pFlnyfRfRc [Accessed 12 Nov. 2019].

Alexander Zotov. (2017). How to make enemy be killed by a bullet in Unity 2D game | OnCollisionEnter2D Unity 2D tutorial. YouTube. Available at: https://www.youtube.com/watch?v=LMOGPN5p4NU [Accessed 7 Dec. 2019].

Alexander Zotov. (2018). Unity 2D Tutorial How To Pick Up An Item Pressing Specific Button. YouTube. Available at: https://www.youtube.com/watch?v=yFKg8qVclBk [Accessed 7 Dec. 2019].

Blackthornprod. (2018). 2D DOUBLE / TRIPLE JUMP PLATFORMER CONTROLLER - EASY UNITY TUTORIAL. YouTube. Available at: https://www.youtube.com/watch?v=QGDeafTx5ug [Accessed 7 Feb. 2019].

Blackthornprod. (2018). 2D PLATFORMER PATROL AI WITH UNITY AND C# - EASY TUTORIAL. YouTube. Available at: <https://www.youtube.com/watch?v=aRxuKoJH9Y0> [Accessed 7 Dec. 2019].

Blackthornprod. (2018). ADVANCED AI IN UNITY (Made EASY) - STATE MACHINE BEHAVIORS. YouTube. Available at: https://www.youtube.com/watch?v=dYi-i83sq5g [Accessed 7 Dec. 2019].

Blackthornprod. (2018). HOW TO MAKE ANIMATION TRANSITIONS - UNITY TUTORIAL. YouTube. Available at: https://www.youtube.com/watch?v=HVCsg\_62xYw [Accessed 7 Dec. 2019].

Blackthornprod. (2018). HOW TO MAKE CHECKPOINTS IN UNITY - EASY TUTORIAL. YouTube. Available at: https://www.youtube.com/watch?v=ofCLJsSUom0 [Accessed 7 Dec. 2019].

Blackthornprod. (2018). HOW TO MAKE HEART/HEALTH SYSTEM - UNITY TUTORIAL. YouTube. Available at: https://www.youtube.com/watch?v=3uyolYVsiWc [Accessed 7 Dec. 2019].

Blackthornprod. (2018). HOW TO MAKE 2D LADDERS IN UNITY - EASY TUTORIAL. YouTube. Available at: https://www.youtube.com/watch?v=Ln7nv-Y2tf4 [Accessed 7 Dec. 2019].

Bowl and Cereal. (2017). Unity Tutorial: Playing Audio. (2017). Unity Tutorial: Playing Audio (Music and Sound Effects). YouTube. Available at: https://www.youtube.com/watch?v=egxNXuwf0\_g [Accessed 12 Nov. 2019].

Brackeys. (2017). How to make CRISP TEXT in Unity. YouTube. Available at: https://www.youtube.com/watch?v=D33d4w89vTs [Accessed 7 Dec. 2019].

Brackeys. (2017). PAUSE MENU in Unity. YouTube. Available at: https://www.youtube.com/watch?v=JivuXdrIHK0 [Accessed 7 Dec. 2019].

Brackeys. (2017). START MENU in Unity. YouTube. Available at: https://www.youtube.com/watch?v=zc8ac\_qUXQY [Accessed 7 Dec. 2019].

Brackeys. (2018). 2D Animation in Unity (Tutorial). YouTube. Available at: https://www.youtube.com/watch?v=hkaysu1Z-N8&list=PLPV2KyIb3jR6TFcFuzI2bB7TMNIIBpKMQ&index=3 [Accessed 7 Dec. 2019].

Brackeys. (2018). 2D Camera in Unity (Cinemachine Tutorial). YouTube. Available at: https://www.youtube.com/watch?v=2jTY11Am0Ig [Accessed 7 Dec. 2019].

Brackeys. (2018). 2D Shooting in Unity (Tutorial). YouTube. Available at: https://www.youtube.com/watch?v=wkKsl1Mfp5M&list=PLPV2KyIb3jR6TFcFuzI2bB7TMNIIBpKMQ&index=5 [Accessed 7 Dec. 2019].

Brackeys. (2018). TILEMAPS in Unity. YouTube. Available at: https://www.youtube.com/watch?v=ryISV\_nH8qw [Accessed 7 Dec. 2019].

Brackeys. (2018). TOUCH CONTROLS in Unity!. YouTube. Available at: https://www.youtube.com/watch?v=bp2PiFC9sSs [Accessed 7 Dec. 2019].

BurgZerg Arcade. (2017). Perspective vs Orthographic Cameras in Unity. YouTube. Available at: https://www.youtube.com/watch?v=c\_WVffzc6GU [Accessed 7 Dec. 2019].

Charger Games. (2015). Unity Scale Game UI To Any Screen Size - Unity Android Game Development. YouTube. Available at: https://www.youtube.com/watch?v=rld\_3SgrZw4 [Accessed 7 Dec. 2019].

Chris’ Tutorials. (2018). Easy 2D Tilemap Collisions | Unity 2018 Tutorial. (2018). YouTube. Available at: https://www.youtube.com/watch?v=VDdKv0DgY5I [Accessed 7 Dec. 2019].

Digitalnativestudios.com. (2016). Cannot Access TextMeshPro object. [online] Available at: http://digitalnativestudios.com/forum/index.php?topic=1082.0 [Accessed 7 Dec. 2019].

GameDev Mode. (2017). Changing Text Of TextMeshPro Via Script. YouTube. Available at: https://www.youtube.com/watch?v=-U29sx9x3Zk [Accessed 7 Dec. 2019].

Gameplusjames. (2015). Unity Endless Runner Tutorial #17 - Making An After Death Menu. (2015). YouTube. Available at: https://www.youtube.com/watch?v=FPXxTyJqIWU [Accessed 7 Dec. 2019].

Gameplusjames. (2016). Unity 2 Player Game Tutorial #7 - Adding Sound. (2016). YouTube. Available at: https://www.youtube.com/watch?v=QSTKy9cHmcI [Accessed 12 Nov. 2019].

khmer2040. (2014). Unity3D - object moving up and down continuously within a y-axis range. [online] Stack Overflow. Available at: https://stackoverflow.com/questions/26010855/unity3d-object-moving-up-and-down-continuously-within-a-y-axis-range [Accessed 7 Dec. 2019].

Learn Everything Fast. (2017). How to Continue Firing when Button is Touched? (2017). YouTube. Available at: https://www.youtube.com/watch?v=onGJELQ7SIU [Accessed 7 Dec. 2019].

N3K EN. (2015). Unity 5 Virtual Joystick [Tutorial][C#] - Unity 3d. (2015). YouTube. Available at: <https://www.youtube.com/watch?v=uSnZuBhOA2U> [Accessed 7 Dec. 2019].

N3K EN. (2018). How to set up Android SDK + JDK with Unity 2018. YouTube. Available at: https://www.youtube.com/watch?v=ojt7ITyklGk [Accessed 7 Dec. 2019].

reddit. (2013). r/Unity2D - How can I make a sprite blink in white when receive a hit? [online] Available at:https://www.reddit.com/r/Unity2D/comments/8xcw8g/how\_can\_i\_make\_a\_sprite\_blink\_in\_white\_when/ [Accessed 7 Dec. 2019].

Sutcliffe, A. (2016). How do I call this IEnumerator in unity c#. [online] Stack Overflow. Available at: https://stackoverflow.com/questions/35730376/how-do-i-call-this-ienumerator-in-unity-c-sharp [Accessed 7 Dec. 2019].

Unity Forum. (2014). Reload level but not all object... [online] Available at: https://forum.unity.com/threads/reload-level-but-not-all-object.261719/ [Accessed 7 Dec. 2019].

Unity.com. (2010). Activate/Deactivate Script from another Object. - Unity Answers. [online] Available at: <https://answers.unity.com/questions/29313/activatedeactivate-script-from-another-object.html> [Accessed 7 Dec. 2019].

Unity.com. (2012). Instantiate as a Child of the Parent - Unity Answers. [online] Available at: https://answers.unity.com/questions/260100/instantiate-as-a-child-of-the-parent.html [Accessed 7 Dec. 2019].

Unity.com. (2013). Instantiating object after some fixed time... - Unity Answers. [online] Available at: https://answers.unity.com/questions/491732/instantiating-object-after-some-fixed-time.html [Accessed 7 Dec. 2019].

Unity.com. (2014). Delete Object After Animation (2D) - Unity Answers. [online] Available at: https://answers.unity.com/questions/670860/delete-object-after-animation-2d.html [Accessed 7 Dec. 2019].

Unity.com. (2015). Set Objects Child to Active/Inactive (Solved) - Unity Answers. [online] Available at: <https://answers.unity.com/questions/894211/set-objects-child-to-activeinactive.html> [Accessed 7 Dec. 2019].

Unity.com. (2015). slider bar in inspector - Unity Answers. [online] Available at: https://answers.unity.com/questions/186832/slider-bar-in-inspector.html [Accessed 7 Dec. 2019].

Unity.com. (2017). [HELP PLEASE] How can I check if an animator transition from animation to animation has finished? - Unity Answers. [online] Available at: https://answers.unity.com/questions/1385483/help-please-how-can-i-check-if-an-animator-transit.html [Accessed 7 Dec. 2019].

Unity.com. (2018). Make enemy move up and down by itself in 2D space? - Unity Answers. [online] Available at: https://answers.unity.com/questions/1499153/make-enemy-move-up-and-down-by-itself-in-2d-space.html [Accessed 7 Dec. 2019].

Unity Technologies (2018). Unity - Manual: Unity User Manual (2018.2). [online] Unity3d.com. Available at: https://docs.unity3d.com/2018.2/Documentation/Manual/index.html [Accessed 7 Dec. 2019].