A·P·U

ASIA PACIFIC UNIVERSITY
OF TECHNOLOGY & INNOVATION

3D COMPUTER GRAPHICS

GROUP ASSIGNMENT

CT001-3-3

Members: ANG CHEE SIAH (TP038259)
LIM ZHI DA (TP041644)
SEE RONG JIE (TP038306)

Intake code: UC3F1805CGD

Module: 3D COMPUTER GRAPHICS (CT001-3-3)

Lecturer's Name: MR. BAHARUL ISLAM

Submission Date: 5th SEPTEMBER 2018

Table of Contents

Acknowledgement

In this assessment, we would like to send our gratitude to Mr. Baharul Islam, our module lecturer of this 3D Computer Graphics module. Although we were faced difficulties in terms of module delivery, his advices in providing a framework of this assessment proved a great help for us in completing this assessment.

In the meantime, we would like to thank our classmates in the UC3F1805CGD intake, who were willing to contribute opinions and feedbacks on the output's concept design. The feedback enables us to further improvise the application compared to the initial ideas of the assessment's output. Their contributions were greatly appreciated.

Lastly, we would like to thank the *Ludum Dare* community for providing us a platform to use this assessment's first output as the submission for their latest game jam session. *Ludum Dare* is a game community where submissions of game prototypes would be evaluated accordingly, and the community's feedback on the output's first prototype assisted in improvising the output's quality as much as possible.

Introduction

This assessment focuses on delivering a prototype of 3D graphics, featuring a 'dynamic 3D scene, animation or application with interactive controls'. The prototype can be presented in any theme. Therefore, a mobile application featuring a story-based selection game using 3D models as graphical assets is developed for this assessment.

This documentation, at beginning, would introduce an overview on how the application should like, and the background and requirements of the application users.

The following of this documentation would describe the process of developing the prototype, including the design planning of the application, 3D asset modelling, 3D asset animation, and its implementation into the game editor program.

Output Overview

The output is a mobile application that could currently be used in Android platform. In this case, the mobile application is of a story-based game genre, where the player would have to select choices to advance through the story.

The game's title was initially *3DayCab*, where the game features a driver who must earn enough money within 3 days of driving 4 customers around the place. Once the player selects the customer, the customer would enter the car, initiate conversations during the drive, and exit the car short after with the amount of money received displayed.

However, in this game, once you have driven a type of customer too much (in this instance, more than or equal to 5 times), the game would forcefully trigger a 'bad end' featuring the driver being murdered by the customers in various methods and blocks out random information of the game until all customers' 'bad end' are reached.

Target Users' Background & Requirements

- ✓ **Region(s)**: Undefined
- ✓ **Age**: 15 years old or above
- ✓ **Race**: Undefined
- ✓ **Literature Fluency**: Decent understanding of English
- ✓ **[Optional] Recommended Understandings/Reference**: News of fatal shooting by Uber driver in Denver, published in CNN website at 2nd June 2018 (Ellis & Jones, 2018)

Output Design

The game application is designated to be developed via Unity3D, a game engine and editor that is capable with 3D and 2D game environment settings. The game's **main mechanics** would be created from **Unity3D**.

As for the game's **3D assets**, it would be created in both **Unity3D and 3ds Max**. Unity3D mainly deals with the environmental 3D assets such as roads and pavements, while 3ds Max was mainly used in creating more complex 3D assets such as buildings and characters.

When creating the prototype, several processes have been conducted where the steps could be classified as: **Concept Design**; **Asset Creation**; **Game Programming**; and **Asset Implementation**.

In **Concept Design** stage, the game's idea was first reviewed and referenced with other games with similar mechanics. The required assets in-game was discussed beforehand to form a workflow for the asset creation process.

In **Asset Creation** stage, the 3D assets along with necessary 2D sprites (for designing user interfaces) were created once the concept design was determined.

In **Game Programming** stage, the game's mechanics and story selection routes were made to ensure the game was able to run properly and able to integrate with the 3D assets by placing them with the created assets, or if not finished, placeholder assets as a means of testing.

In **Asset Implementation** stage, all the finished 3D assets would be integrated into the game and was further adjusted if there were any errors on the imported 3D assets.

3D Asset Modelling

In this assessment, there are 2 types of modelling techniques involved.

The first type involved either the usage of 3ds Max or Unity3D. Simple geometry objects would be created, and each component would overlay together as multiple objects with a parent.

Using the example in Unity3D, the component ‘Environment’ uses a mix of ‘cube’ components (Mesh components included) with different dimensions to create a road environment on a plane (unity3d.com, 2018), as shown in ***FIGURE 1.1***. The pros of this method include less requirement of using any complex modifiers, although this method is only meant for creating simplistic assets that could be made from basic geometry shapes.

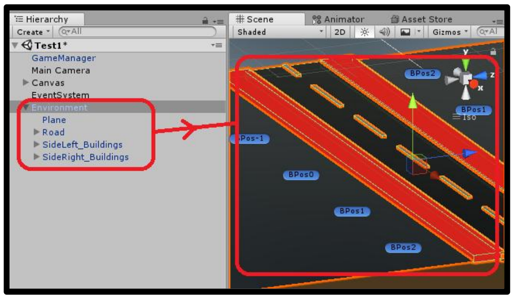


FIGURE 1.1: Multiple Objects under 1 Parent Object in ‘Environment’ Prefab

The second type involves the use of the 3ds Max only, where the geometry objects were transformed into editable polygons to manually forming the objects in form of characters.

As seen in one of the character file in 3ds Max file format, the object ‘Driver’ was originally a cube. But with a mix of in-program modifiers such as ‘Editable Poly’, ‘Symmetry’, ‘Extrude’, ‘Cut’, and many others, a humanoid shape was created as seen in ***FIGURE 1.2***. While this method could create more complex 3D objects, this method is relatively time-consuming and sometimes might result to suffering in low quality if the 3D object was not properly modelled.

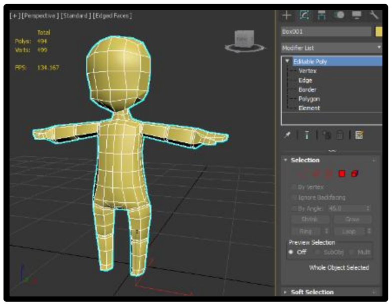


FIGURE 1.2: Humanoid 3D Character Made from Editable Polygon Modifier

After completed the ‘Driver’ character model, it is used as the template for other 4 characters which are ‘Businessman’, ‘School Girl’, ‘Old Man’ and ‘Part. Each character included with their own characteristics by using the in-program modifiers too. ***Figure 1.3*** shows the output of the other 4 characters.

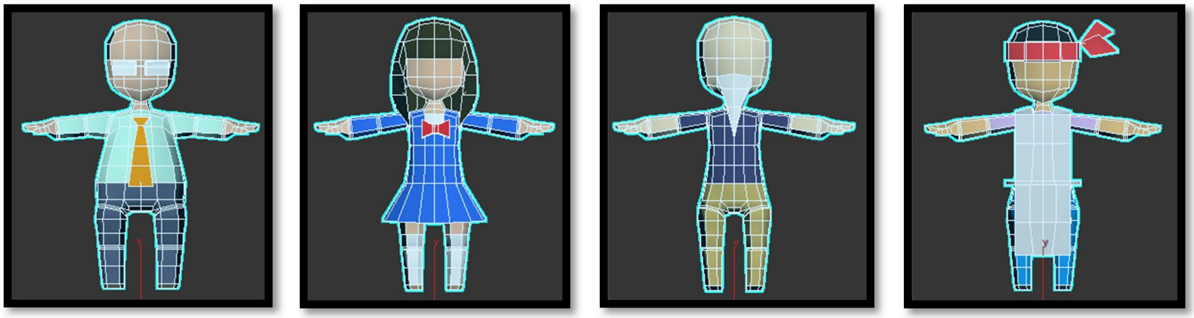


FIGURE 1.3: Other Humanoid 3D Characters Made from Driver Template

The Cab car or the Taxi is also the main asset as seen in ***FIGURE 1.4***. The car was originally a box, by changed it into editable-poly, the box was modified into a car shape including all the important features such as car window, doors, lights, signals, car plate and many more. Only the car tyres are made with cylinders, it started with 8 sides and eventually smoothed it with NURMS Subdivision modifier. Result will be seen when the tyres are attached to the car body.

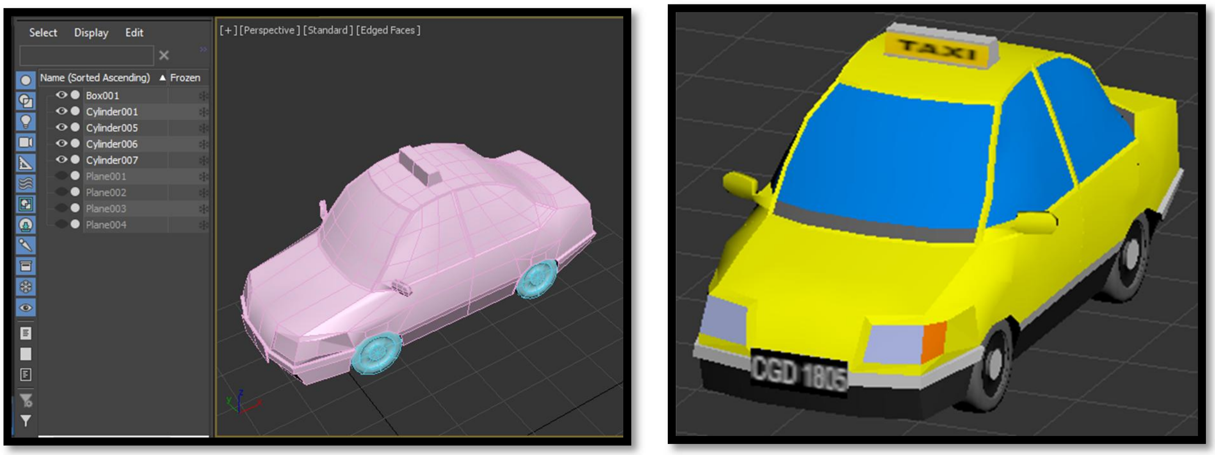


FIGURE 1.4: Progression of Cab Car Model

Besides, there are the buildings made by 3ds Max to be spawn when the game starts. Depending on the buildings look, it can be build using box or cylinder as shown in ***FIGURE 1.5***. The buildings are made as simple as possible because the purpose of the buildings is just to present the city view.

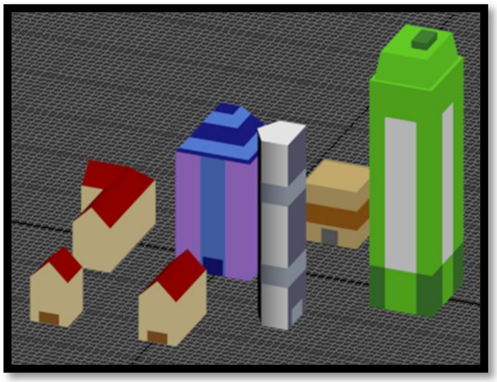


FIGURE 1.5: Simple Building Models

Lastly is about the texturing process, all the characters, car model and building are given plain color. The main purpose is to reduce the rendering time especially for a mobile game as the capability of mobile phone are weaker than computer. The only texture applied is the wording part of the cab car.

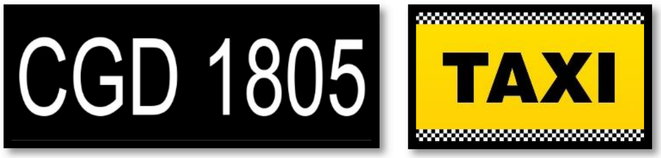


FIGURE 1.6: Texture Applied for the Car

3D Asset Animation

The 3D assets' animation would be focused on the all the characters and the Cab car.

<u>3ds Max</u>

For each of the characters, their animation is done in 3ds Max first thus exported into Unity for the next stage. Before doing the animation, all the models must be checked thoroughly so that it will not have any problem while rigging and animating the models. Biped is used in this animation, translate, rotate and scale the biped parts accordingly so that it fits properly into the character model (Autodesk.Help, 2017). Next is the skinning process where it is the part to connect the biped with the model as shown in ***FIGURE 2.1***.

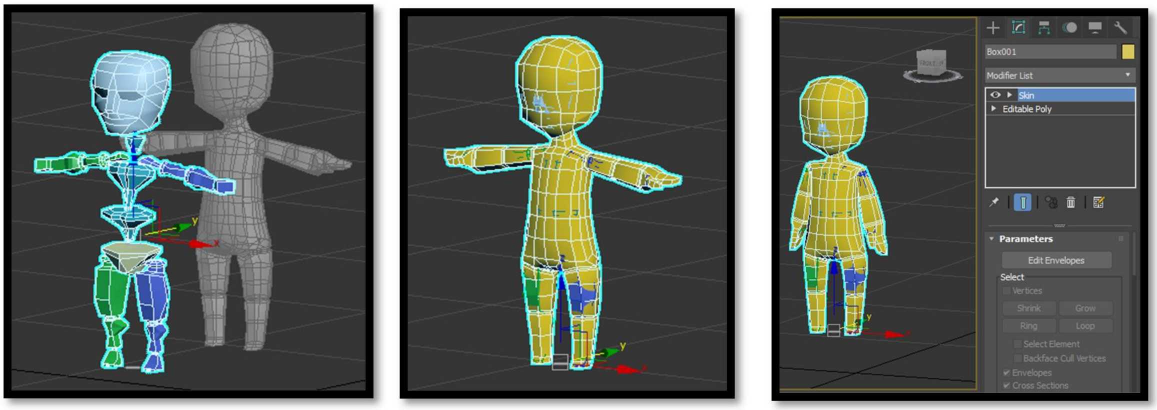
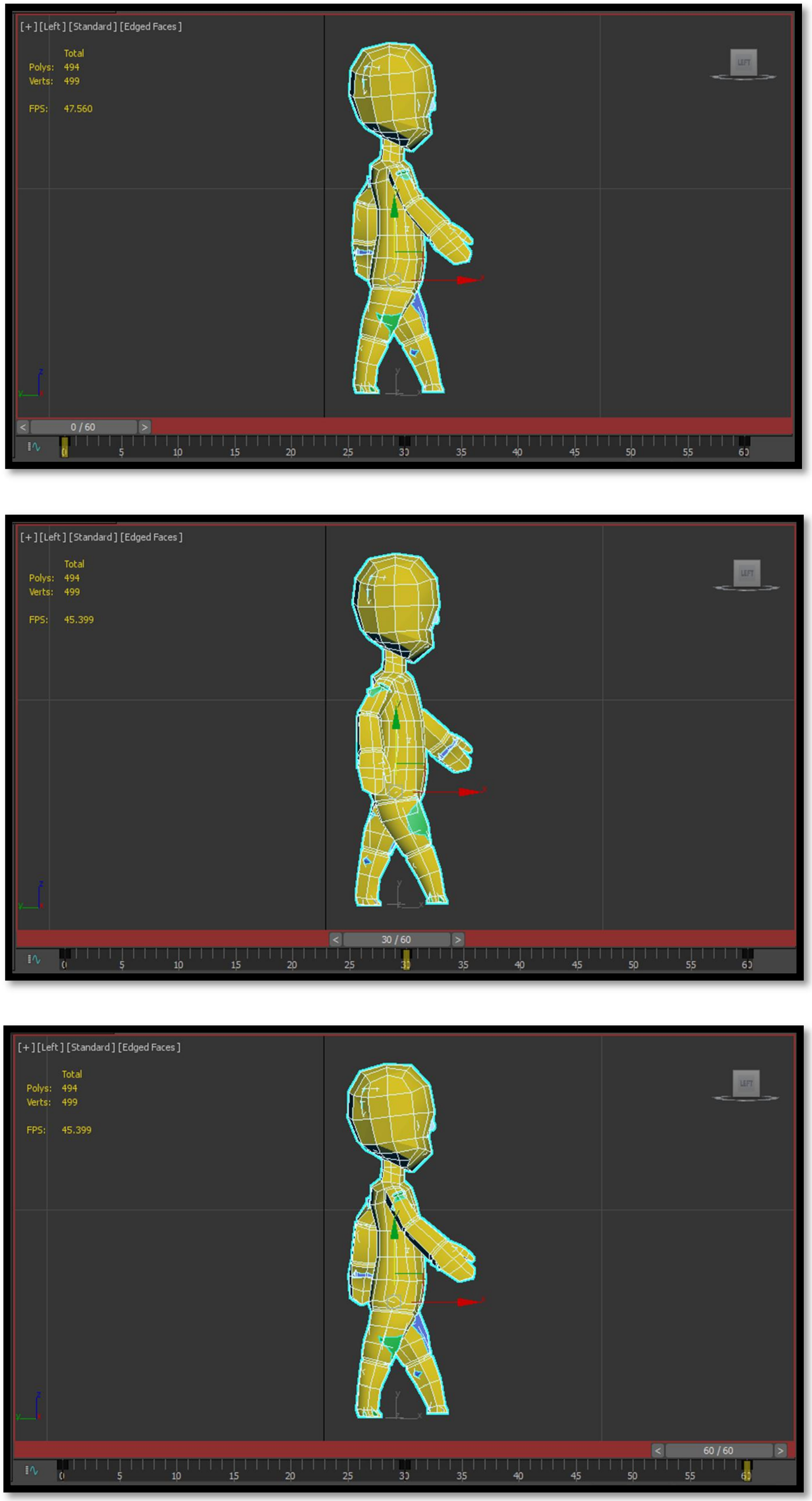


FIGURE 2.1: Process from Rigging to Skinning.

After the step above is completed, it is time to do the walking animation. The time duration is set from 0 frame to 60 frames as the walking animation is not very long. In addition, the looping concept can be used for it. Auto Key would be enough for this walking animation, the walking pose will be set each 30 frames while at 60th frames the pose should be similar to 0th frames so that the looping concept can be done naturally. All the adjustments will be at the arms, legs and head in order to produce a natural walking animation as shown in ***FIGURE 2.2***.



```
[+] [Left] [Standard] [Edged Faces]
        Total
Polys:  494
Verts:  499

FPS:    47.560

0 / 60
```

```
[+] [Left] [Standard] [Edged Faces]
        Total
Polys:  494
Verts:  499

FPS:    45.399

30 / 60
```

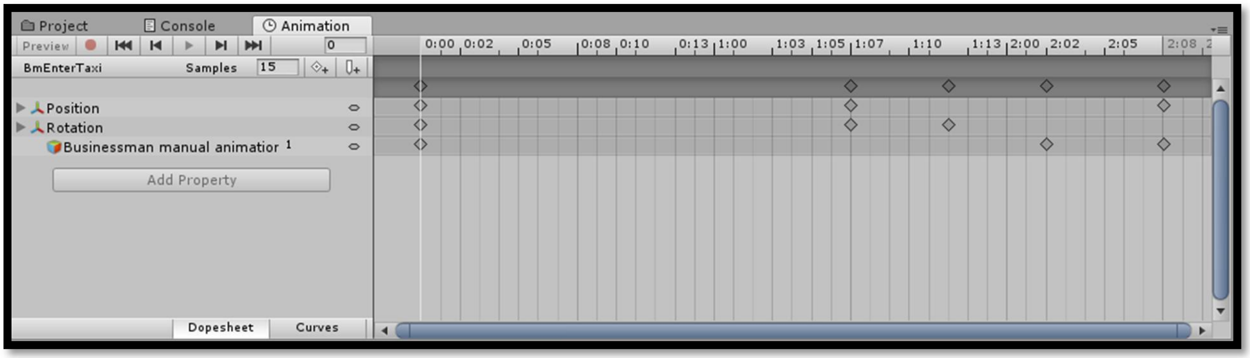
```
[+] [Left] [Standard] [Edged Faces]
        Total
Polys:  494
Verts:  499

FPS:    45.399

60 / 60
```

FIGURE 2.2: Walking Animation

Unity

All the models and their animations will be brought into Unity as assets to the game. All the animation will be done in 'Animation' window thus bring it to the 'Animator' to control their behavior. By linking the 'Animator' with the characters, the in-game animation will be completed.

The 'Enter of Car' animation is around 3 seconds where the character's transition is transition and rotation, when character entered the taxi the model disappears as shown in ***FIGURE 2.3*** and ***2.4.***



BmEnterTaxi

Samples 15

FIGURE 2.3: Animation of Businessman Enters the Taxi

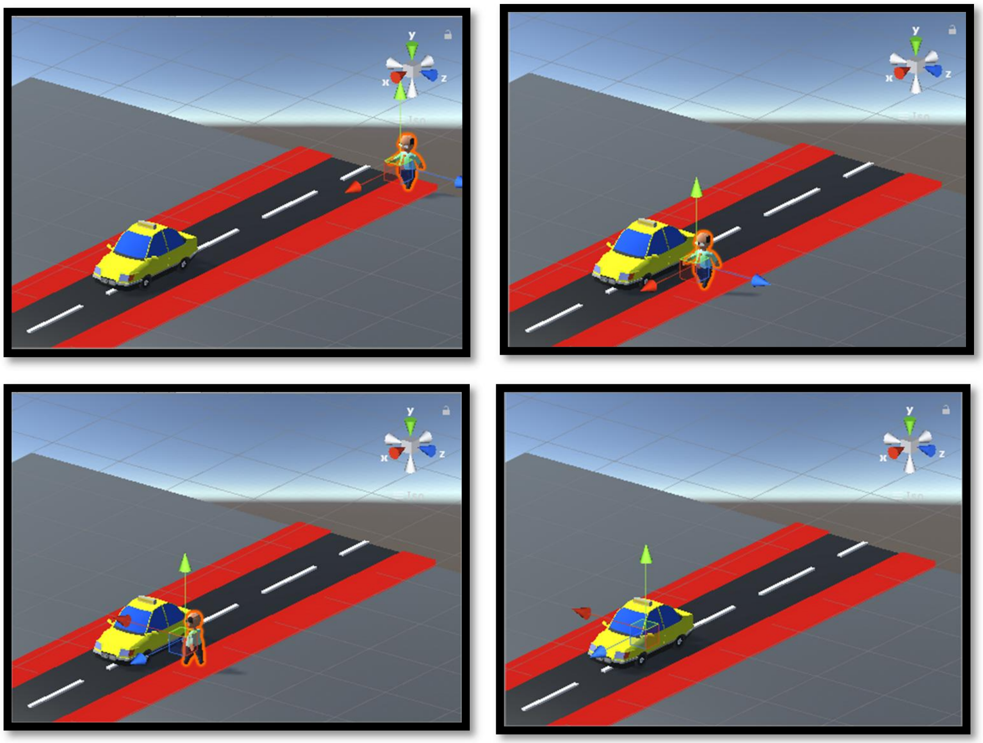


FIGURE 2.4: Output of the Animation

Same goes for the ‘Exit of Car’ animation is also around 3 seconds where the character’s transition is transition and rotation, when character exited the taxi the model appears as shown in ***FIGURE 2.5*** and ***2.6.***

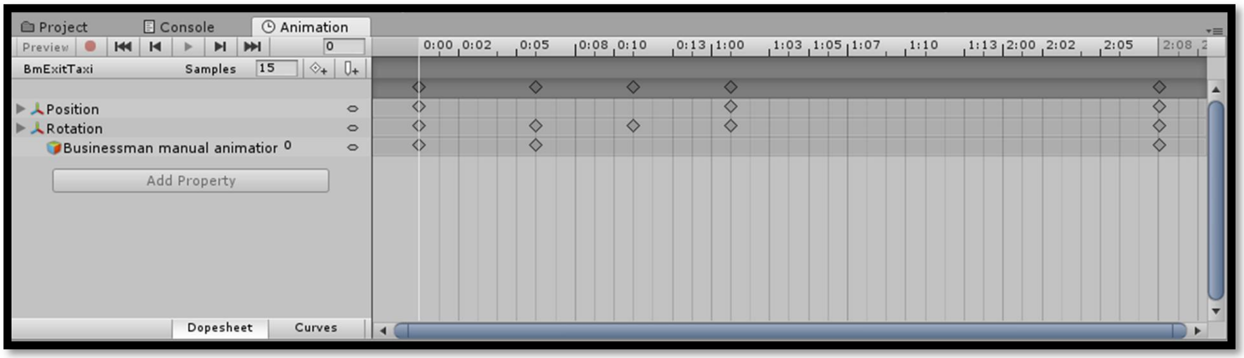


FIGURE 2.5: Animation of Businessman Exit the Taxi

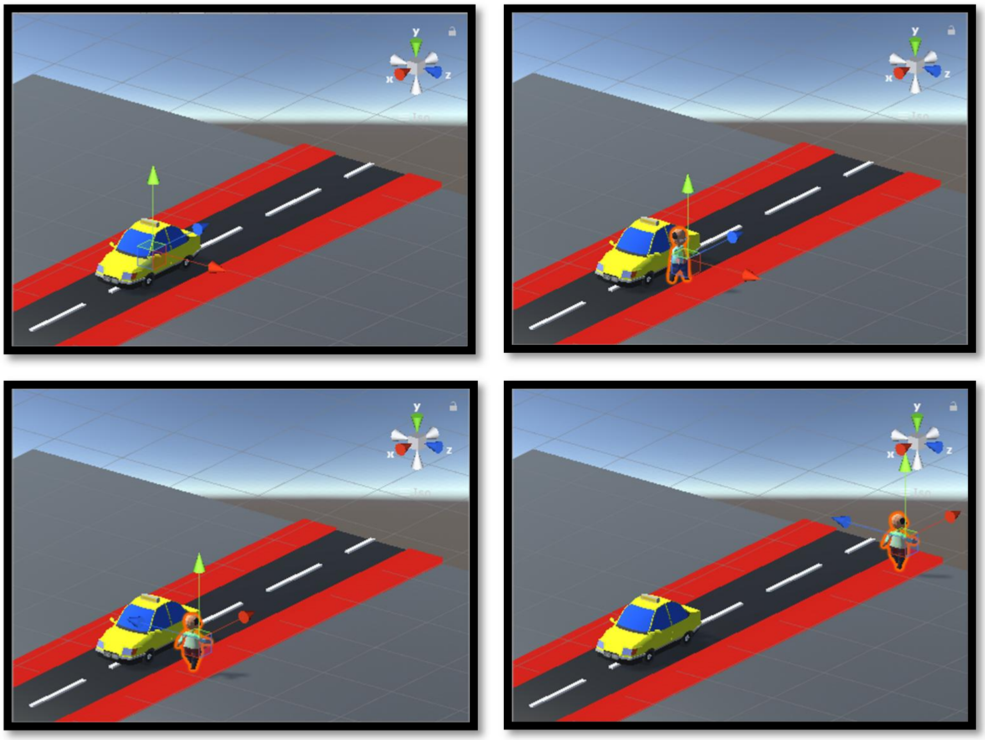


FIGURE 2.6: Output of the Animation

The same animation is applied to other characters as well except the driver as the exit animation is not required.

If the players triggered the bad end, there is animation prepared for it. The animation is for the taxi model where it moves zigzag that indicated something happened inside the car as shown in ***FIGURE 2.7*** and ***2.8***.

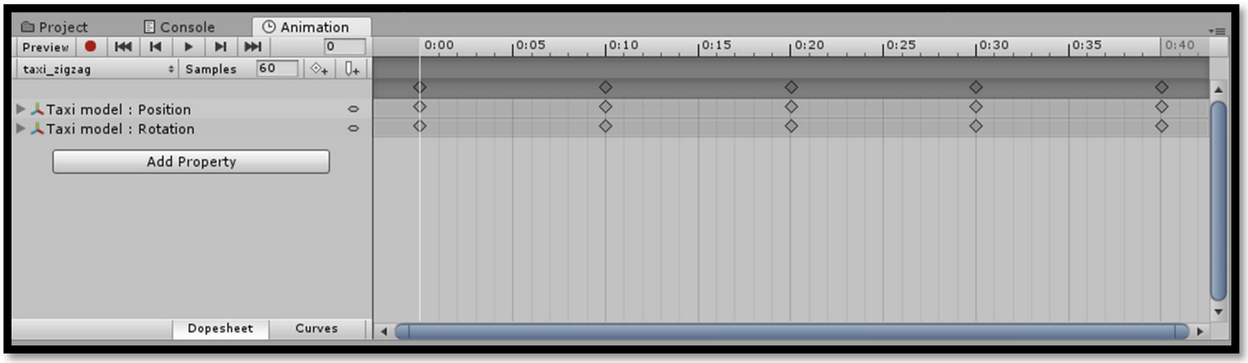


FIGURE 2.7: Animation of Car Going Zigzag

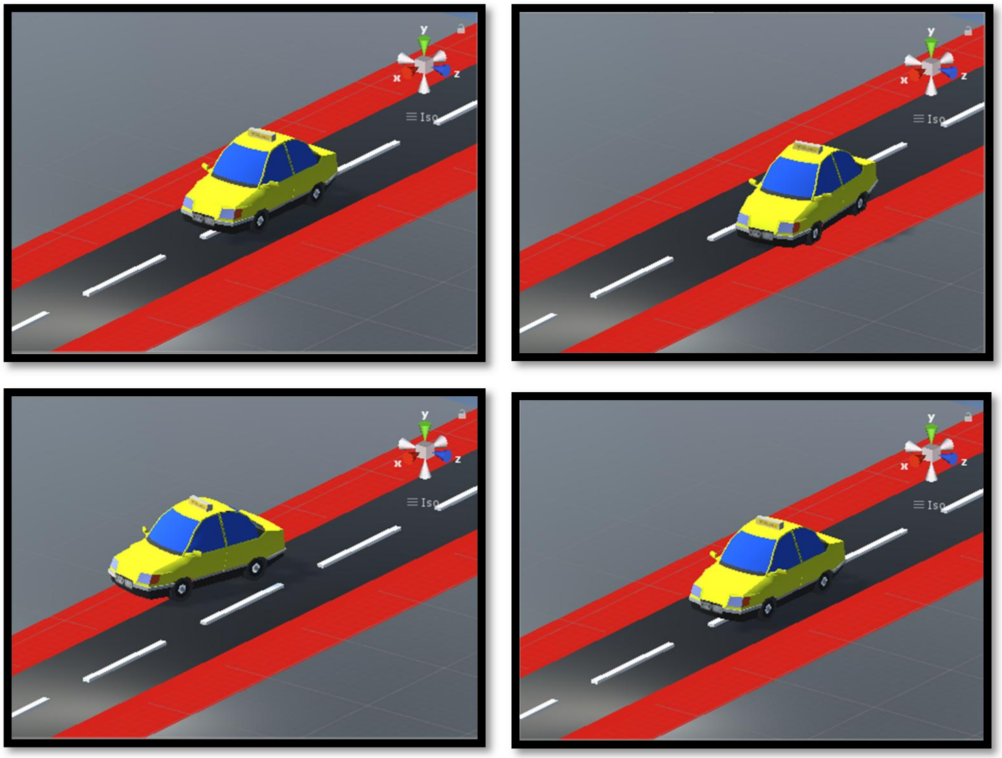
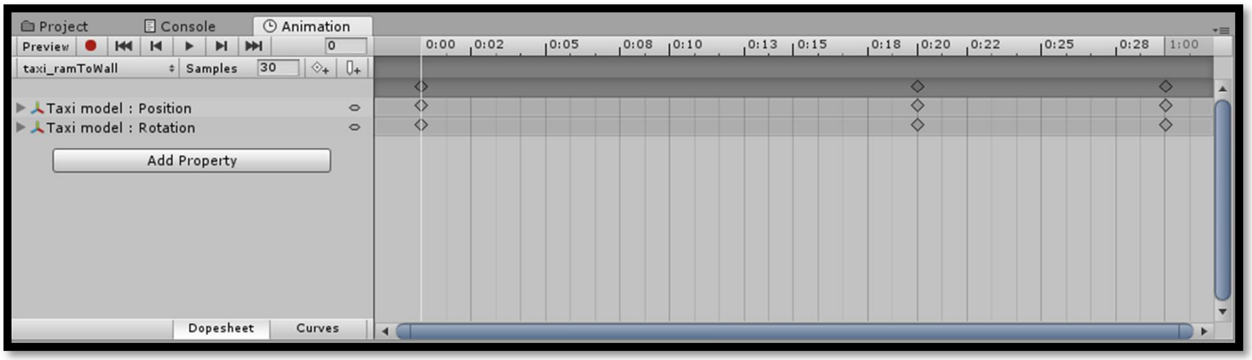


FIGURE 2.8: Output of the Animation

There is another animation where the car lose control and rammed into something, also for the taxi model only as shown in ***FIGURE 2.9*** and ***2.10***.



taxi_ramToWall
Samples 30
Taxi model : Position
Taxi model : Rotation

FIGURE 2.9: Animation of Car Lose Control

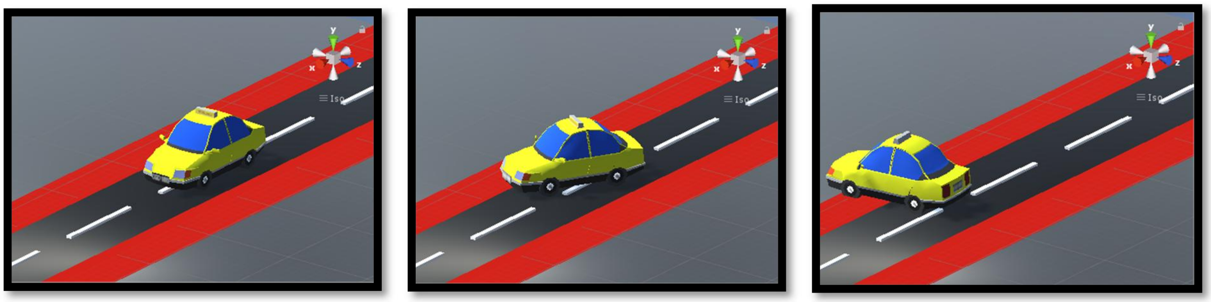
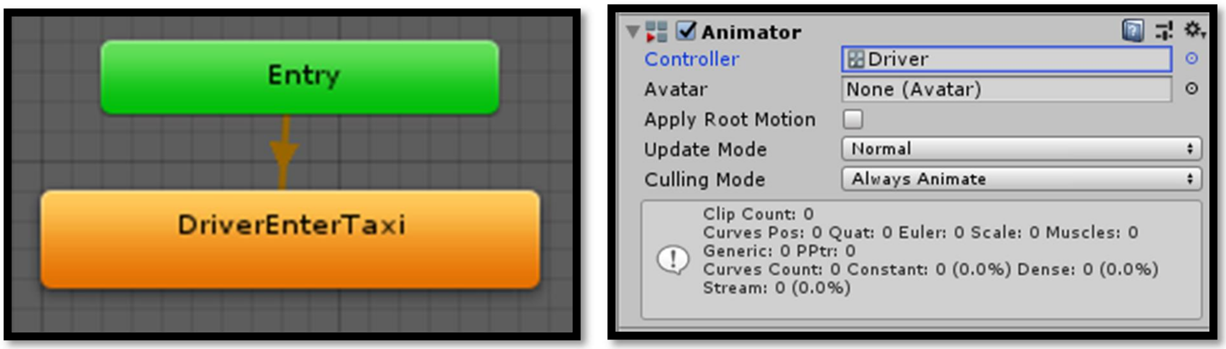


FIGURE 2.10: Output of the Animation

When the 'Animation' is completed, the 'Animator' will be created to link with the 'Animation' to perform the actions. Next, the character model will be linked with the 'Animator' so that the character can move as per designed as shown in ***FIGURE 2.11***.



☑ Animator
Controller: Driver
Avatar: None (Avatar)
Apply Root Motion: □
Update Mode: Normal
Culling Mode: Always Animate

Clip Count: 0
Curves Pos: 0 Quat: 0 Euler: 0 Scale: 0 Muscles: 0
Generic: 0 PPtr: 0
Curves Count: 0 Constant: 0 (0.0%) Dense: 0 (0.0%)
Stream: 0 (0.0%)

FIGURE 2.11: The linking between Animation, Animator and Driver

3D Asset Implementation

The 3D assets, after imported into Unity3D engine, would be registered as Prefabs to enable more convenient spawning feature of the assets into the game.

How to import assets from 3ds Max to Unity3D

There are a few of steps for exporting assets from 3ds Max to Unity3D, in our case, we use the step which directly import the models in the max file format(".max") generated by 3ds Max into Unity3D engine, by pressing right click of the mouse in Unity3D's project panel, then select the **"Import new asset" selection**, and the assets file could be selected and imported from the file explorer.

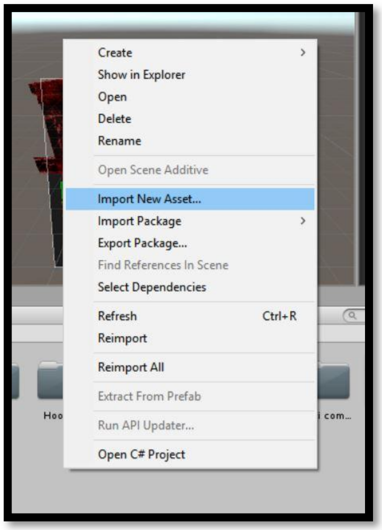
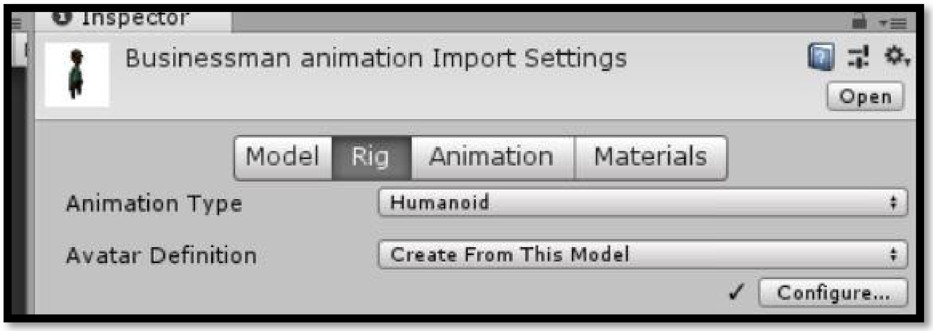


FIGURE 3.1 "Import New Asset" selection in Unity3D

At this stage, we're allowed to test the assets that we've imported in Unity3D, we may drag and drop the asset icon in project panel to the scene, and the **basic transform value** such as rotation and scale **would be inherited** from the settings in 3ds Max. After dragging the object to the scene, we may edit the asset by Unity3D's functions such as adding a C# script to make it controllable by the player or add a "Rigid body" component to make it responses to Unity3D's physics.

The animation and the texture that have been attached to the asset would be imported along with the models, in Unity3D, the model itself would be the main object which contains of the animation file and the texture. In our case, the textures that we used for the models are solid color which can be brought into Unity3D without any additional operation, but the animation is needed to link with the model manually in Unity3D. For using the animation, we need to modify the "Rig" section of the model in the animation import settings. After selected the "Rig" section, the first option would be the "Animation type" which uses the "Generic" as default selection, we need to change the animation type to "Humanoid" to function our animation made in 3ds Max. The "Humanoid" animation type would be the best model if we're using the animation which is human based, and the animation would be imported smoothly if we did the rigging process perfectly in other software like 3ds Max.



Businessman animation Import Settings

Animation Type: Humanoid

Avatar Definition: Create From This Model

FIGURE 3.2 Animation import settings in Unity3D

How to affect the 3D assets in Unity3D

In Unity3D, we need two components which are **"Animation" and "Animator"** for running character's animation, the "Animator" is the holder and the controller of "Animation", while animation is the walking animation that we've implemented in game. In "Animator", we may modify the flow of multiple animations, for instance if we want the Idle animation to play as default animation, we may right click the Idle animation and select the "Set as layer default state", and the Idle animation would be played first and keep looping if we want (unity3d.com, 2018).

Besides adding the default animation, we may create the flow of the animations by right clicking the animation, then select "make transition", an arrow will appear in the window and we may link it to another animation to create a flow, for example the figure below is showing the walking animation which is going to be played after the Idle animation is ended.

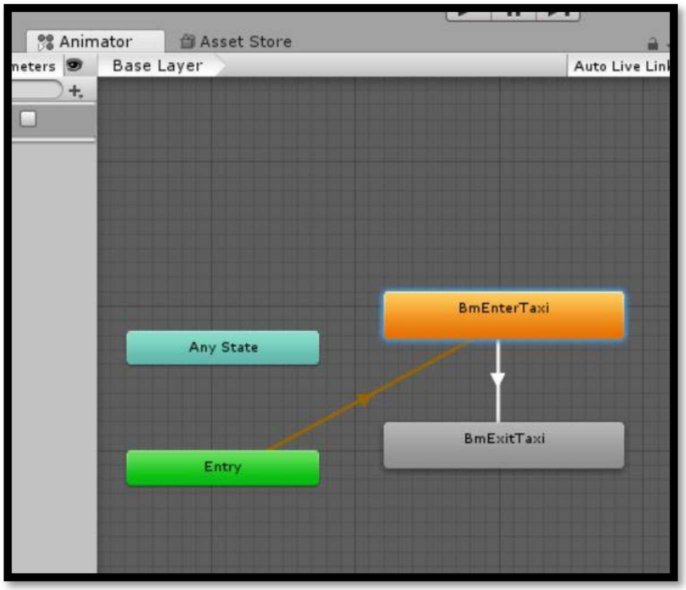


FIGURE 3.3 Animator window and basic animation flow

We may add condition to trigger certain animation also, for instance a “Death” variable can be created to check whether the object is death animation is going to be played or not, in programming side, if the condition is met and the “death” variable was changed to True, the death animation would be played, else it wouldn’t be played.

Task Layout / Storyboard

While creating the game, several templates were created during the planning process to create the references of the output.

Character design templates were created as a guidance to create the 3D models of the game, as shown in the ***FIGURE 4.1***:

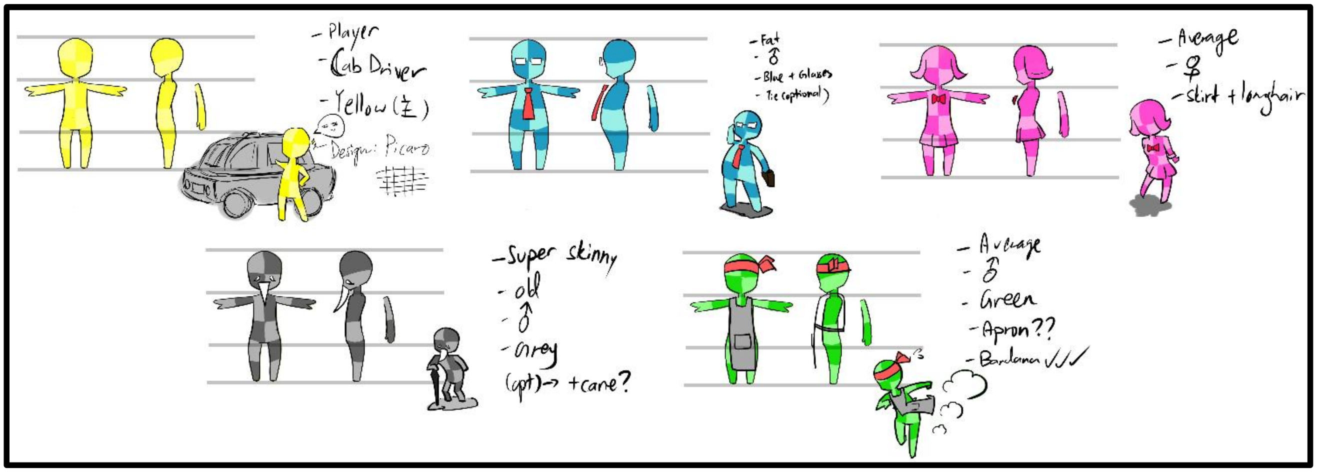


FIGURE 4.1: 3D Character Design Templates of 3DayCab

Cab car design template shown in Figure 4.2 was created as a guidance to create the model:

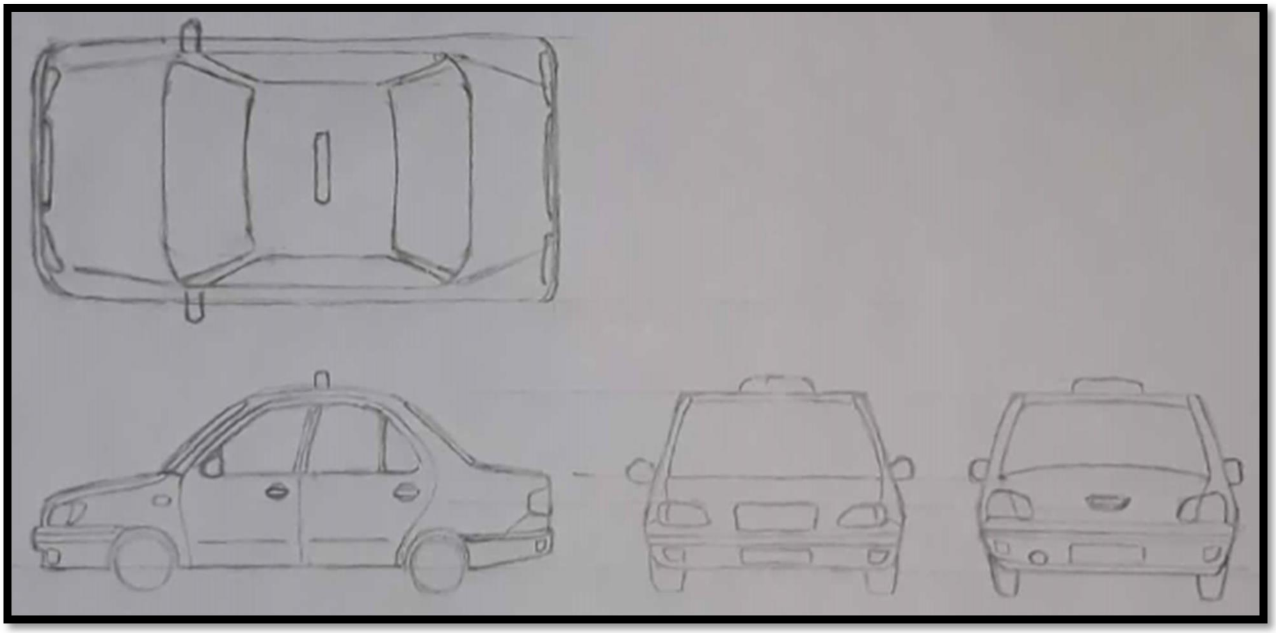
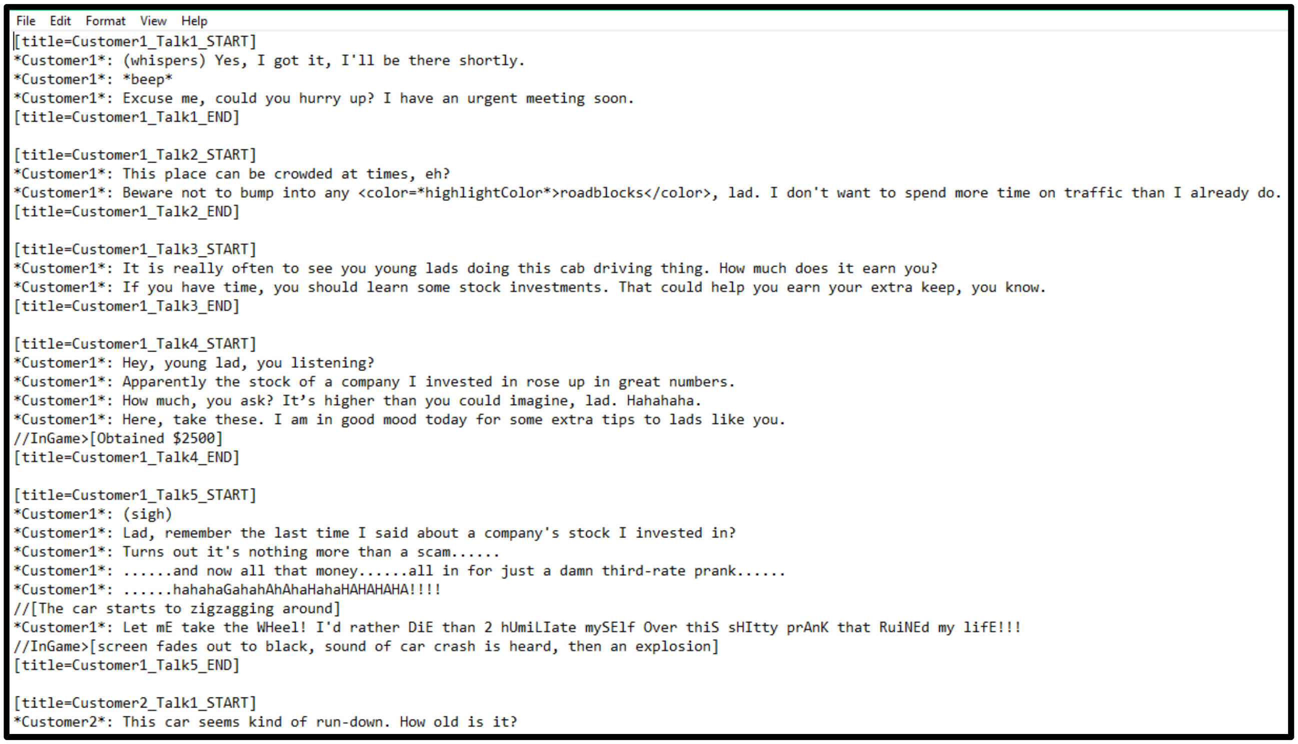


FIGURE 4.2: Cab Car Design Template

The game's story for each character were written and edited beforehand in a notepad file, as shown in ***FIGURE 4.2***:



```
[title=Customer1_Talk1_START]
*Customer1*: (whispers) Yes, I got it, I'll be there shortly.
*Customer1*: *beep*
*Customer1*: Excuse me, could you hurry up? I have an urgent meeting soon.
[title=Customer1_Talk1_END]

[title=Customer1_Talk2_START]
*Customer1*: This place can be crowded at times, eh?
*Customer1*: Beware not to bump into any <color=*highlightColor*>roadblocks</color>, lad. I don't want to spend more time on traffic than I already do.
[title=Customer1_Talk2_END]

[title=Customer1_Talk3_START]
*Customer1*: It is really often to see you young lads doing this cab driving thing. How much does it earn you?
*Customer1*: If you have time, you should learn some stock investments. That could help you earn your extra keep, you know.
[title=Customer1_Talk3_END]

[title=Customer1_Talk4_START]
*Customer1*: Hey, young lad, you listening?
*Customer1*: Apparently the stock of a company I invested in rose up in great numbers.
*Customer1*: How much, you ask? It's higher than you could imagine, lad. Hahahaha.
*Customer1*: Here, take these. I am in good mood today for some extra tips to lads like you.
//InGame>[Obtained $2500]
[title=Customer1_Talk4_END]

[title=Customer1_Talk5_START]
*Customer1*: (sigh)
*Customer1*: Lad, remember the last time I said about a company's stock I invested in?
*Customer1*: Turns out it's nothing more than a scam......
*Customer1*: ......and now all that money......all in for just a damn third-rate prank......
*Customer1*: ......hahahaGahahAhAhaHahaHAHAHAHA!!!!
//[The car starts to zigzagging around]
*Customer1*: Let mE take the WHeel! I'd rather DiE than 2 hUmiLIate mySElf Over thiS sHItty prAnK that RuiNEd my lifE!!!
//InGame>[screen fades out to black, sound of car crash is heard, then an explosion]
[title=Customer1_Talk5_END]

[title=Customer2_Talk1_START]
*Customer2*: This car seems kind of run-down. How old is it?
```

FIGURE 4.3: Segment of each Customers' Dialogue Script

The game's mechanics were made in form of flowcharts, as shown in the ***FIGURE 4.3***:

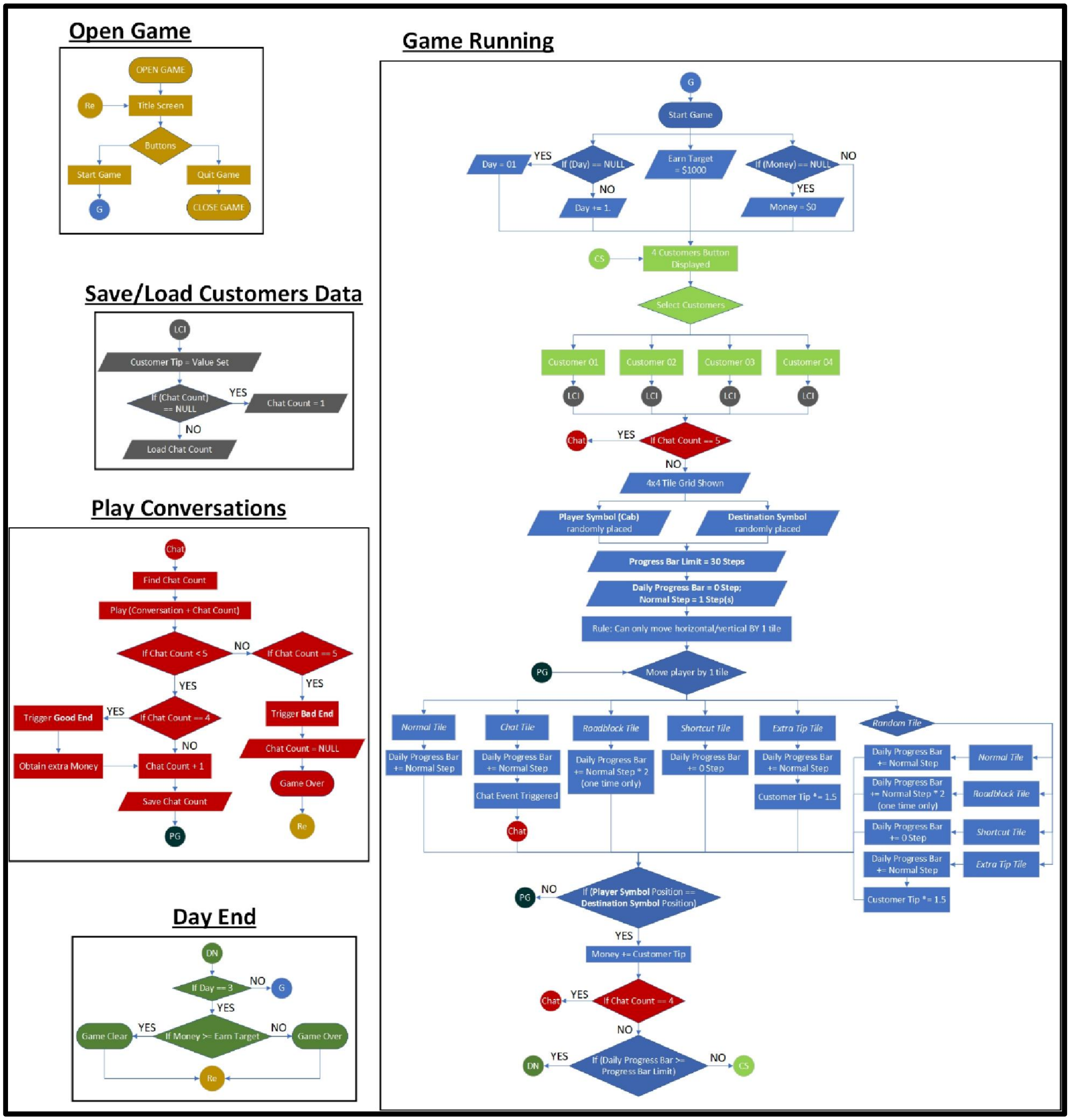


FIGURE 4.4: 3DayCab Game-flow Layout

Output Screenshots

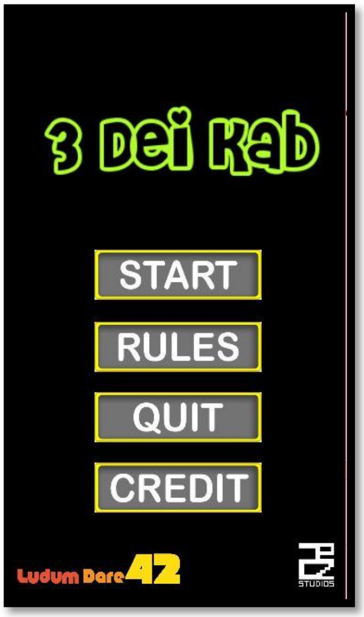


FIGURE 5.1: 3DayCab Title Screen

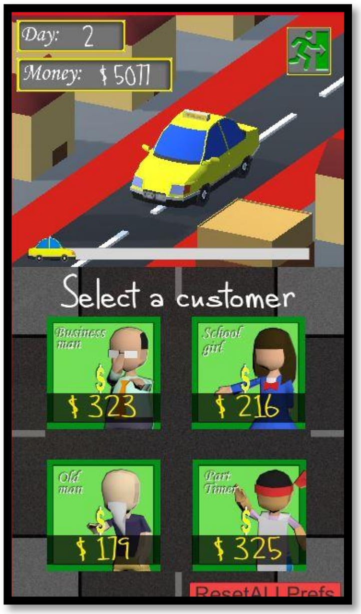
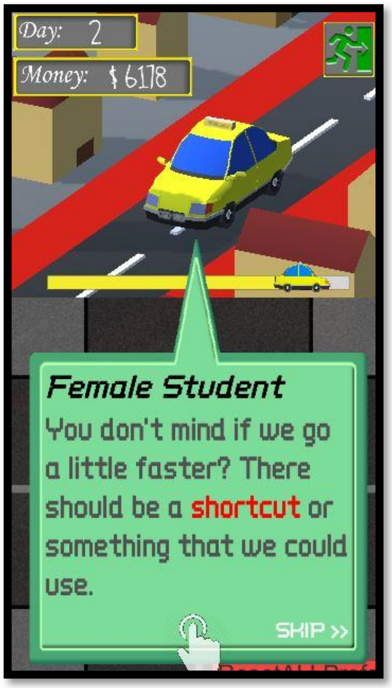


FIGURE 5.2: 3DayCab Customer Selection Window

FIGURE 5.3: Cab and Female Student in scene

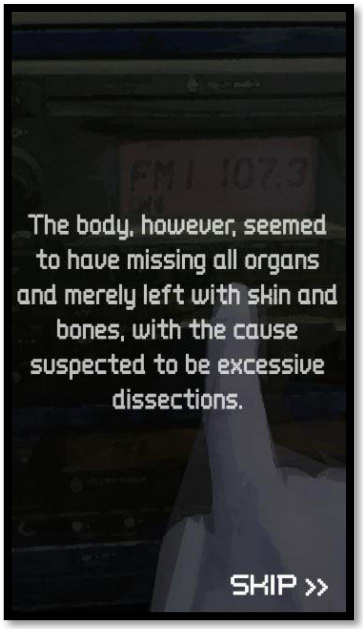
FIGURE 5.4: Entering the cab & FIGURE 5.5: Coming out from the cab



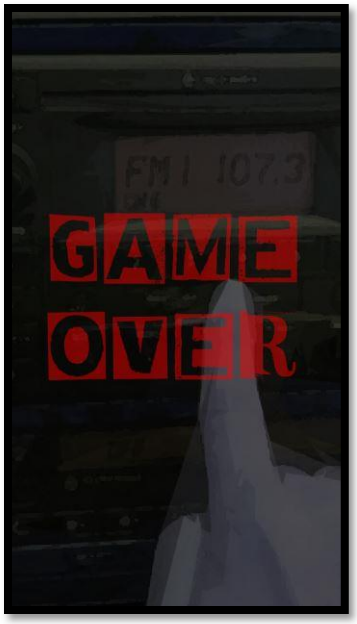
Female Student

You don't mind if we go a little faster? There should be a shortcut or something that we could use.

FIGURE 5.6: Conversation window



The body, however, seemed to have missing all organs and merely left with skin and bones, with the cause suspected to be excessive dissections.



GAME OVER

FIGURE 5.7: Bad End cutscene & FIGURE 5.8: Game Over Display

Conclusion

Throughout this assessment, a mobile application featuring a story-based selection game of a taxi driver is created. The game is made with the game environment incorporated with the 3D objects, mixed with 2D sprites that act as user interfaces that affect the animations of the 3D objects in the game.

In the process of developing the output, the methods of creating 3D objects using various possible modifiers were learnt. The differences in between creating simplistic and more complex 3D objects were identified and thus different approaches were adapted in creating both sides of the 3D assets.

While in assets animation, the concept of rigging and skinning were learnt. Making sure the models are completed so that the rigging and skinning will be easier. When animating, small details cannot be overlooked to give the natural feels of animation. The animation is more complicated than expected especially the interaction part thus many attempts are implemented to create a wanted result.

Importing asset from 3ds Max to Unity3D is simple since Unity 3D supports the .max file format generated by 3ds Max, while the texture and the animation can be brought along with the model, the point to take note is only that we need to change the animation type to humanoid to enable the human based animation such as walking animation.

References

- Autodesk.Help, 2017. *3ds Max Help | Biped.* [Online]
  Available at: https://knowledge.autodesk.com/support/3ds-max/learn-explore/caas/CloudHelp/cloudhelp/2017/ENU/3DSMax/files/GUID-2F6BC5D1-DD45-4C2E-AC3A-D8C6E0F5DEB1-htm.html
  [Accessed 25 August 2018].

- Ellis, R. & Jones, S., 2018. *Uber driver held after fatal shooting of passenger in Denver.* [Online]
  Available at: https://edition.cnn.com/2018/06/01/us/uber-driver-shooting-denver/index.html
  [Accessed 19 July 2018].

- unity3d.com, 2018. *Unity Manual 2018.2 | Animator Window.* [Online]
  Available at: https://docs.unity3d.com/Manual/AnimatorWindow.html
  [Accessed 26 August 2018].

- unity3d.com, 2018. *Unity Manual 2018.2 | GameObject.* [Online]
  Available at: https://docs.unity3d.com/Manual/class-GameObject.html
  [Accessed 18 August 2018].