COURSEWARE FOR DRIVING SIMULATION

CHEW SIN SENG

TECHNICAL SUBMITTED IN FULFILMENT OF THE DEGREE OF COMPUTER SCIENCE

FACULTY OF COMPUTER SYSTEM AND SOFTWARE ENGINERRING

2013

UNIVERSITI MALAYSIA PAHANG

BORANG PENGESAHAN STATUS TECHNICAL

JUDUL: …….

SESI PENGAJIAN: ……

SAYA ……(HURUF BESAR)

Mengaku membenarkan tesis/laporan PSM ini disimpan di Perpustakaan Universiti Malaysia Pahang dengan syarat-syarat kegunaan seperti berikut:

1. Tesis/Laporan adalah hakmilik Universiti Malaysia Pahang.
2. Perpustakaan Universiti Malaysia Pahang dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institut pengajian tinggi.
4. **Sila tandakan (√)

☐ SULIT (Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972) *

☐ TERHAD (Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan) *

☐ TIDAK TERHAD

Disahkan Oleh

…… ……

Alamat tetap: Penyelia:

Tarikh: …… Tarikh:……..

*Sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali sebab dan tempoh tesis/laporan ini perlu dikelaskan sebagai SULIT atau TERHAD.

DECLARATION

I declare that the work in this technical report is my own except for quotations and summaries which have been duly acknowledged.

14 MAY 2013 Name: CHEW SIN SENG
ID No: CD10027
Signature:

_____

SUPERVISOR DECLARATION

I hereby declare that I have read this thesis and in my opinion this technical/report is sufficient in terms of scope and quality for the award of the degree of Bachelor of Computer Science ( )

Signature :……

Supervisor Name:

Date :……

ACKNOWLEDGEMENT

I wish to express my eternal gratitude and appreciation to my beloved research supervisor, Miss Noraniza Binti Samat, for her invaluable guidance, empowering support and profound advice throughout the preparation and realisation of this undergraduate project.

I also heart felt gratitude goes to the respondents who were UMP students for this research. Really thanks for their fully support and co-operation to make this research possible. Besides that, I would like to thank Madam Rahmah Binti Mokhtar for assist me on the technical report through this whole undergraduate project. Moreover, a very thank you to my parents too who always loved me and support me during this entire research.

Finally, above all, to Almighty God for giving me a good health, strength and perseverance to complete this technical report for undergraduate project.

ABSTRACT

Courseware for Driving Simulation

Chew Sin Seng

Faculty of Computer Systems and Software Engineering, Universiti Malaysia Pahang,

Lebuhraya Tun Razak, 26300 Gambang, Kuantan, Pahang Darul Makmur

This project is to develop a Courseware of Driving Simulation. Driving simulation is a simulation of driving environment. Driving simulation is useful but not commonly used by driver for driving learning purpose. Thus, the car driving accidents increase sharply. Main purpose for develop the courseware of driving simulation is to decreasing car driving accident case and provide a platform to driver for training their own driving skill. The new driving simulation software will be providing a serious game training on driving skill. User will use either keyboard or other hardware to control the car and multiple driving learning courses will be providing inside the simulation. At the end of the simulation process, result of evaluating the performance during the course will be shown. In a nutshell, user can review their result and realise which driving skill is their weakness thus can improve their skill.

Keywords: Driving simulation, Driver, Courseware

ABSTRAK

Kursus untuk Simulasi Memandu

Chew Sin Seng

Faculty of Computer Systems and Software Engineering, Universiti Malaysia Pahang, Lebuhraya Tun Razak, 26300 Gambang, Kuantan, Pahang Darul Makmur

Projek ini adalah untuk membangunkan satu Kursus Memandu Simulasi. Memandu simulasi adalah simulasi persekitaran memandu. Memandu simulasi adalah berguna tetapi tidak biasa digunakan oleh pemandu untuk memandu tujuan pembelajaran. Oleh itu, kemalangan memandu kereta meningkat dengan mendadak. Tujuan utama untuk membangunkan perisian kursus memandu simulasi adalah untuk mengurangkan memandu kereta kes kemalangan dan menyediakan platform untuk pemandu untuk melatih kemahiran memandu mereka sendiri. Perisian simulasi memandu baru akan menyediakan latihan permainan yang serius untuk memandu kemahiran. Pengguna akan menggunakan sama ada keyboard atau perkakasan lain untuk mengawal kereta dan pelbagai kursus pembelajaran memandu akan menyediakan dalam simulasi. Pada akhir proses simulasi, keputusan penilaian prestasi semasa akan ditunjukkan. Secara ringkasnya, pengguna boleh menyemak keputusan mereka dan menyedari mana kemahiran memandu adalah kelemahan mereka itu boleh meningkatkan kemahiran mereka.

Keywords: Simulasi Memandu, Pemandu, Kursu

TABLE OF CONTENT

PART 3

LIST OF FIGURE

LIST OF TABLES

PART 1

INTRODUCTION

1.0 Introduction

Driving has become a common way being transportation into our daily life style. Besides that, some countries regard driving as a representation of freedom and independent. Thus, increasing of driver will cause the increasing of accident case not only because of the number of driver increases but due to the insufficient experience's beginner driver and high aged adult driver. To solve this problem, practice is a must for those learners. So the driving simulation are the best way to solve, decrease, improve, protect and train their problem on driving skill, accident case, driving ability and most importantly their precious life.

What is driving simulation? Driving simulation is the simulation of driving environment which can be used for entertainment as well as in training for driver's education. The driving simulation basically has three major purposes which are training, analysis and evaluating the driver driving skill, technique, response and behavior. First for training part is a practical to train the driving skill and technique in order to impart the safe driving technique for them. Secondly for the analysis part is to show and analysis the user who using this simulation of their performance, behavior and responses so they can realize what is their weakness on driving skill. Thirdly the evaluating part is for the driving educational school, they can refer the evaluating result from this driving simulation to decide and given the suitable and proper lesson for the driving learner. So what is driving simulation courseware? Driving simulation courseware can be considered as a portable driving simulator because only installations inside computer are essentially needed. Unlike other driving simulator which are unmovable for example like National Advanced Driving Simulator, SIMUVEG,

Portable In-vehicle Driving Simulator, TUTOR and etc. User just need to install this courseware inside their computer then can start using this driving simulation.

1.1 Problem Statement

Old age adult people and beginner driver have low or lack of driving performance on road which are high risk danger not only to themselves but all road users. Beginner driver don't have enough experience, low performance and lack of confidence when driver on road. They are easily nervous and unable to make a correct decision or response when facing certain incident on road driving.

For example the changing of weather condition, the traffic jam situation, the road sign, signboard on road, traffic light, and many sudden incident happen which can make them nervous and lost their steadiness to make a good decision or action. Old age adult have enough driving experience but some of them have long period away from vehicle. This may cause them dropping of their driving standard, cognitive, mental and physical ability. These effects could bring along the increasing of accident risk on road driving.

Besides that, some driving's rule on road which had been changed so they are unable to realize due to the long period away from society. In addition, current driving simulation system did not input the local traffic rules which will becoming an issue for the foreigner user due to unfamiliar of local traffic's rule and driving pattern.

1.1.1 Purpose of project:

To develop a driving simulation courseware that will enhance, train and solve beginner driver or old age adult people's driving skill, performance, technique, driving responses and driver's behaviors.

1.1.2 Objective of project

i. To create an interactive learning environment for beginner driver to learn the basic traffic rules and local driving pattern information.

ii. To develop a courseware that helps to increase the learning skills of beginner driver by practicing the simulation compare to study from book that less effective.

iii. To show the driver performance and result through this simulation can help user to understand where the mistake occur.

1.1.3 Scope of project

This driving simulation software more or less is target for the beginner driver and the high aged adult driver. This software can be used in driving school, home and any location which located with computer since this software is portable installing inside computer then straight can use it. This software currently main platform is computer. This software will collect the data from the user driving analytic for example the mistake count through this driving simulation, the time for certain mission, the smoothness of controlling the car, the driver's behavior when driving the car and many else. The technique will be used or apply are mainly using 3D modeling and Virtual Reality.

1.1.4 Conclusion

In this chapter, the driving simulation is to improve the beginner driver and high aged adult driver to improve, regain, maintain and understanding the ability, performance, response, skill, rule, negative and positive action. Since this is portable by installing inside computer then can straight using it. This software's advantages obviously are save cost, time and safe. Comparing to test-track, on-road, instrumented vehicle, and many other driving simulators. The effect from this driving simulation can be an aid to a lot of sector in society which help them to reduce the danger and accident case on road toward not just driver but to all road users.

1.2 Literature review

1.2.1 Existing system: National Advanced Driving Simulator (NADS) and CarSim

NADS is the National Advanced Driving Simulator which located in the University of Iowa's Oakdale Research Park Campus. National Advanced Driving Simulator can consider a center or a home which provider a huge range of simulator by giving varies level of driving realism. NADS's employees are collection and the combination of all different area of expert in driving simulator to develop best and high technology system.

The Development and Research team in NADS are sponsor by government, military, and industry partners which aim to saves lives, improves quality of life for vehicles, advances the state of the skill in driving simulation, and improves the efficiency and productivity of the vehicle manufacturing sector. This simulation aim to improve and solve the problem of infeasible, too costly, or unsafe in the real world, also including the assessing of cognitive or physical ability, receive and analyses the performance and driving behaviour of the system. Figure 1.1 shows the structure of National Advanced Driving Simulator.

Figure 1.1: structure of National Advanced Driving Simulator

CarSim is one of the driving simulator which been used widely in the world now. CarSim consist many type which different in size and price from $20,000 Desktop system until $100,000,000 full-vehicle simulator. Most of the CarSim simulator only locates at some research factory, University and some big company due to the high build cost. So CarSim developer team has develop some low cost system like CarSimDS from mechanical simulation technique. Figure 1.2 show the structure of CarSimDS model. There are several advantages from using CarSimDS like low-cost, real-time operation in Windows environment by using the standard PC. Besides of the driving skill training purpose, it suitable for car engineer to test their concept car by using the virtual car inside the simulator on proving ground roads and analysis how the vehicles behave. Figure 1.3 shows the CarSimDS using laptop as platform for operate the driving simulation. Data analysis and data collection from this simulator can be used for discriminate the difference in tires, stability control algorithms, race car chassis setups, and many other items.

Figure 1.2: structure of CarSimDS

Figure 1.3: structure of CarSimDS on laptop

1.2.2 Improve beginner driver's performance

To improving the beginner driver's performance, confidence and the fast response on road driving through driving simulation. Practices make perfect which is true for the thing that no familiar with so beginner driver can using this system for training and practice purpose. According to Lisa Dorn (2005), it would appear that professional driver training affects simulated driving performance with trained drivers demonstrating a potentially safer driving style than untrained drivers. Besides that, according to Anderson (1980), early research has demonstrated improvements in accident risk. McKenna (2006) reported that hazard anticipation in driving can be significantly improved by training in the laboratory using video simulation techniques. In addition, Lonero (2008) states that nowadays, there is a much better appreciation of the strengths and shortcomings of driver training programmers and new technologies are available to deliver training programmers more effectively. Which can conclude that the driving simulation can improve the driving's performance.

1.2.3: Assisting high age adult driver to regain their driving technique and ability

To help old age adult driver to maintain, improve and regain their driver performance and ability by using this portable driving simulation software compare on-road test or practice. This is because some of the eldest over confidence their driving skill. According to Holland(1993), Marattoli and Richardson(1998), Groeger and

Brown(1989) state that Older drivers generally perceive their driving ability to be better than or equal to that of their peers and better than that of younger drivers. According to J Gerontol (1994), Our driving population is aging and faces increased risk for injury and death from motor vehicle crashes. Over confidence and unwilling to train are dangers for high age driver. According to Groeger and Brown (1989), Holland (1993), and Marottoli and Richardson (1998), older drivers assign high ratings to their perceived driving ability. So this system not only to repair, recover and regain their driving skill but also decreasing the risk of their danger from accident.

1.2.4 Data analysis

Data analysis through this simulation can help user to understand the mistake or the bad habit which they always done eventually they able to change it. According to Bryan Reimer(2006), Lisa A. D'ambrosio, Joseph F. Coughlin, and Michael E. Kafrissen(2006) and Joseph Biederman(2006) state that, to draw inferences confidently about real driving behaviors from driving simulation data. According to Bella (2008), Godley (2002), Lee (2004), Törnros (1998) and Yan (2008) state that simulation can provide a valid index of driving performance. Therefore, the result of data index from the simulation can be analysis to visualize the mistake which cause by the user.

1.2.5 Learning from mistake

Beginner driver able learn from mistake through this driver simulation. This is an advantage because the beginner driver didn't drive on real road but can gain the knowledge from error through this driving simulation nevertheless this is one of the safety ways for learning compare real road test. According to Kulhavy (1977) states that, Errors are usually salient, unexpected events that can motivate further learning about a task. The negative feedback provided by errors creates an element of surprise which temporarily halts task performance while learners try to work out why the error occurred.

1.3 Current system and its limitation

Most of the existing driving simulations are launch base on the architecture of driving simulator instead of choosing a system like driving simulation software. This feature consist advantage but follow with the disadvantage. The advantage of driving

simulator is state by Nilsson (1993) which is the use of an advanced driving simulator has many advantages over similar real-world or on-road driving research, including experimental control, efficiency, expense, safety, and ease of data collection. The driving simulator for example like NADS is using the three dimension virtual reality technique to display the virtual environment as much similar as reality. Klee (1999) states that driving simulator could provide a realistic driving experience. But the problems are the build cost, importable and require huge space for developing a driving simulator. Due to the problem, the advantages of courseware of driving simulation comparing NADS are portable, low cost and only require a functional computer. The feature from NADS like high resolution of virtual environment is unavailable for courseware of driving simulation but simple structure will be built to ensure the requirement of most of the computer can be minimize to max. So people no need to purchase expensive high feature computer to using this courseware of driving simulation. Besides that, the courseware are portable since is target build in CD form so can easy to transfer to other computer to use it. In addition, comparing the huge size of the simulator, this courseware of driving simulation only require the computer storage for installation of this software instead of using actual home space for placing purpose. Besides that, US traffic's rule are most common foundation for implement inside the driving simulation system, thus most of the user are learning driving through simulation but base on others country traffic's rule. When users drive in real car in actual environment, they will drive follow the learning driving pattern inside the simulation since they get used to the driving pattern already. This will become a potential dangerous issue for them.

1.4 Terminology

NADS	National Advanced Driving Simulator
CarSimDS	CarSIm Driving Simulation

Table 1.1: Terminology in Literature Review

1.5 Scope and limitation of the study

1.5.1 Scope of the study

The scope of the study is focussing on both which are the people who having driving experience and the people who didn't having driving experience. For the people who have driving experience they can use this system as practicing. For example the people who didn't drive for a long period already, they can use this driving simulation system as a practice to recall their driving skill. Most of these kinds of people are high age adult or the worker who had go overseas country a long period. Besides that, not just assisting them to recall back their driving skill but can teaching them the traffic's rule of current time because long period of absent of driving, they maybe didn't realise any changes of the current traffic's rule compare to their memory. Secondly is for the people didn't having any driving experience they can using this driving simulation as learning and training. Thus they can learn without driving a real car on road which is safe, can learn in house which is easy and can learn wherever they want by installing the simulation into their computer which is convenient.

1.5.2 Limitations of the study

The limitation from the study is lacking of current traffic's rule knowledge, lacking of game programming knowledge and the limited of development time given.

- Lack of traffic's rule knowledge: didn't familiar with all traffic rule in Malaysia for example certain place where specific sign board need to be understand its meaning consist.
- Lacking of game programming knowledge: crucial issue for driving simulation development phase due to lacking of programming knowledge.
- Limited time give: 1 semester time (4-5 months) is given for developing this driving simulation.

PART 2

REPORT BODY

2.1 User Requirement

The collection and gathering of user requirement is using the survey technique which is questionnaire. The reason using questionnaire is because this driving simulation system need to base on what user needed as a first priority to develop and will be adding or improve certain feature compare to existing driving simulation software. The user requirement for this driving simulation development project is using questionnaire and distribute to the 30 respondents which are randomly selected. After collecting back the questionnaire the answer from the respondents is being analysis. The result consists of two parts. One is Section B for analysis the satisfaction of current existing system another one is Section C for collecting the requirement on developing new driving simulation. The result of Section B is shown below.

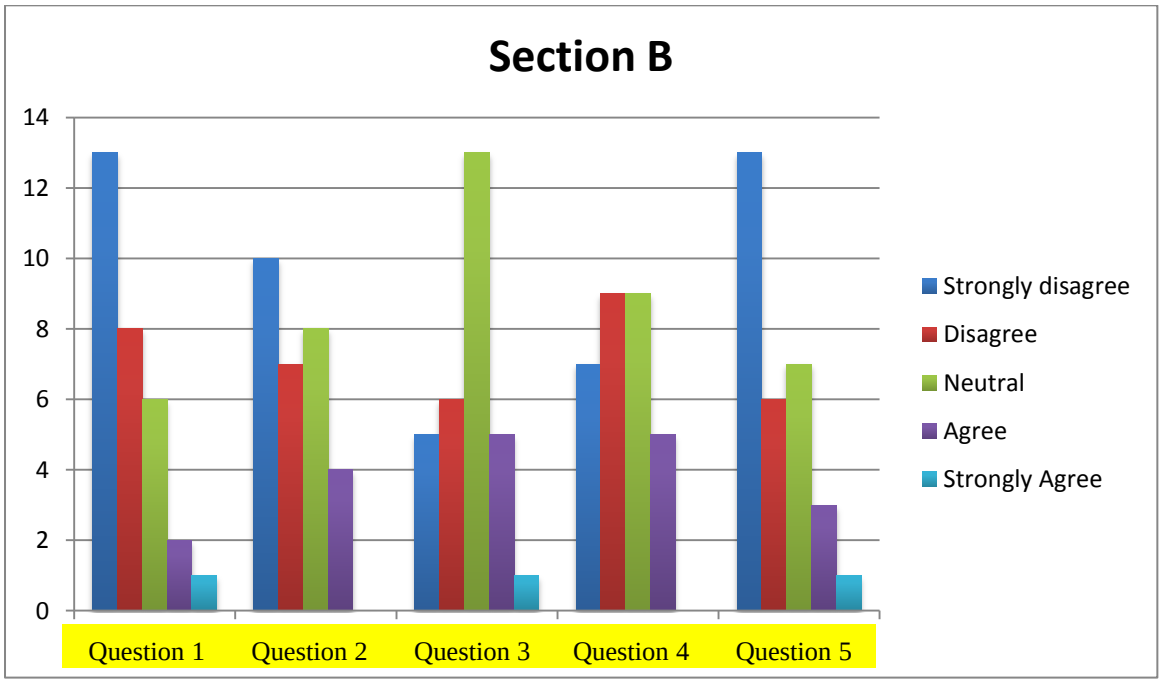


Figure 2.1: Section B in Questionnaire

The result shows that the two most highest data is the respondents did not using the driving simulation as a practice or training for their driving skill and the users notice the existing driving simulation did not follow local traffic's rule for example country Malaysia. The data show the average of the result is negative perspective for the existing driving simulation because of certain issue for example not easy to use, too expensive which users are unaffordable to bought it and certain driving simulation is require driving steering to function in order to more near realistic but will also become a expenses for user. The next histogram is showing the Section C result which is showing what the needed criteria for the driving simulation are.

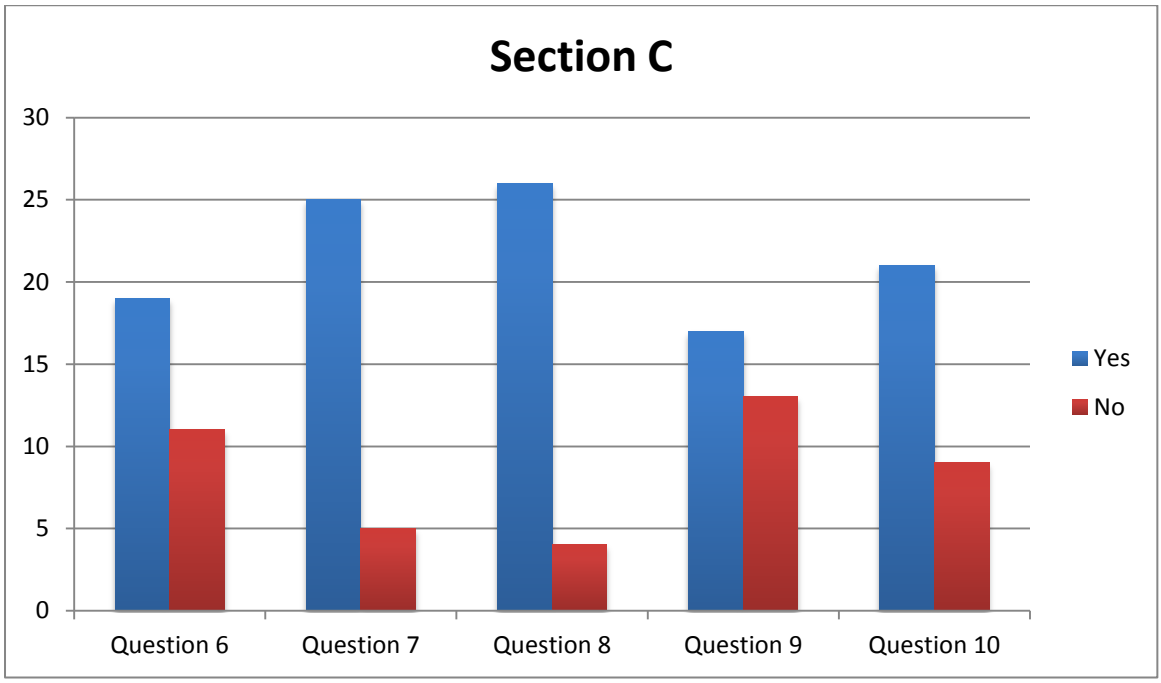


Figure 2.2: Section C in Questionnaire

This is the result from Section C from the questionnaire. The result shows that most of the respondents prefer portable, easy to locate, simple but perfect and using local traffic's rule driving simulation. The question 9's result is not so obvious maybe is because certain respondents are prefer using driving steering which more close to reality and part of the respondents are prefer using keyboard for functioning the driving simulation. Thus the new driving simulation can be develop to both controllable function by letting users to select which type of control function they want.

2.2 Design Description

The design for the driving simulation is consists few important criteria which are the driving simulation will base on local (Malaysia) traffic's rule as a foundation to design, user can change the driving learning course inside the simulation for example user can learn parking, turning, and many else. Besides that, before enter the simulation, it will enter the simulation configuration part which allow user to select the screen resolution, graphics quality, changing of input and click either want window form or not. Below is the design of the system interface.

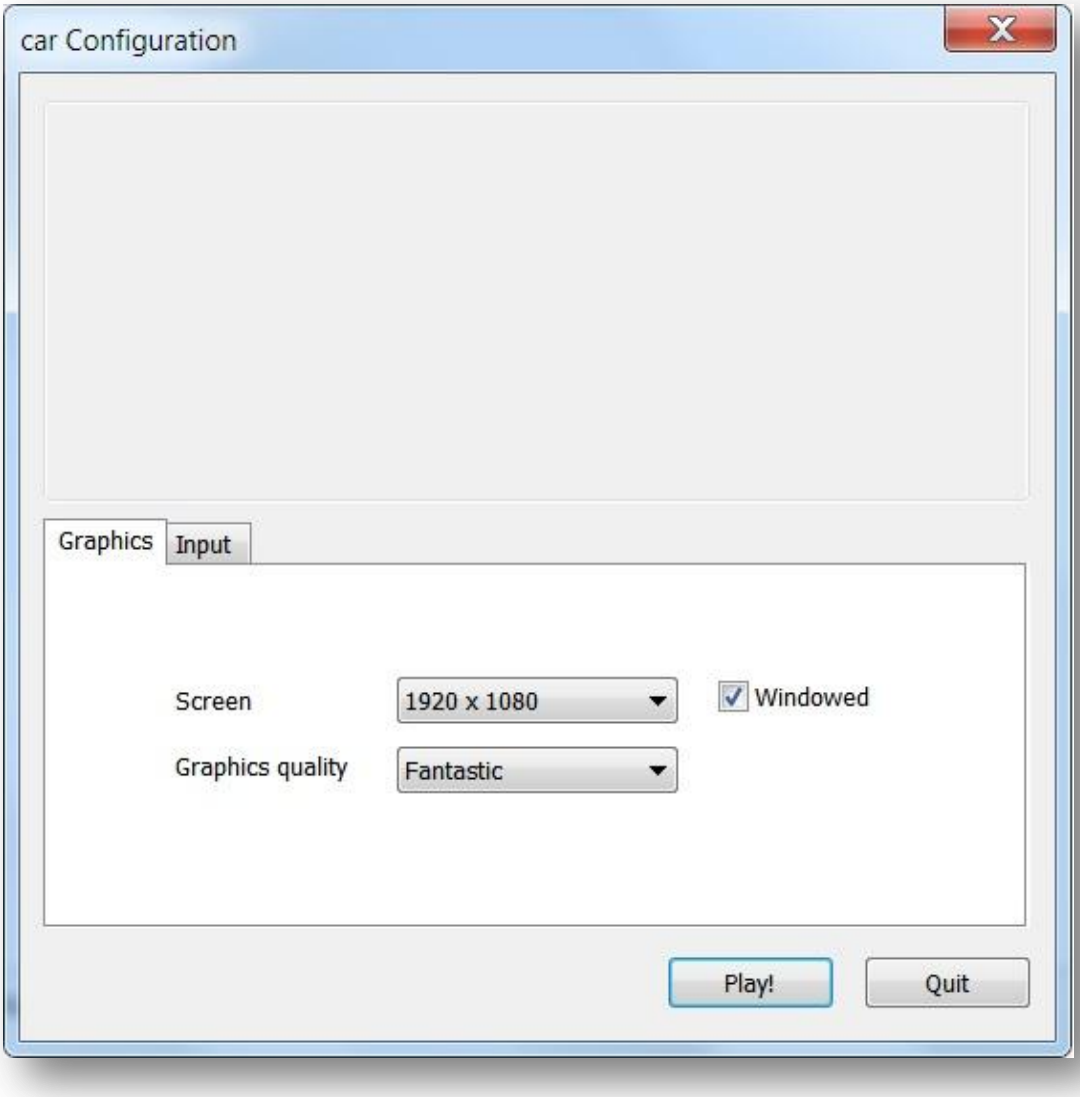


Figure 2.3: Car Configuration

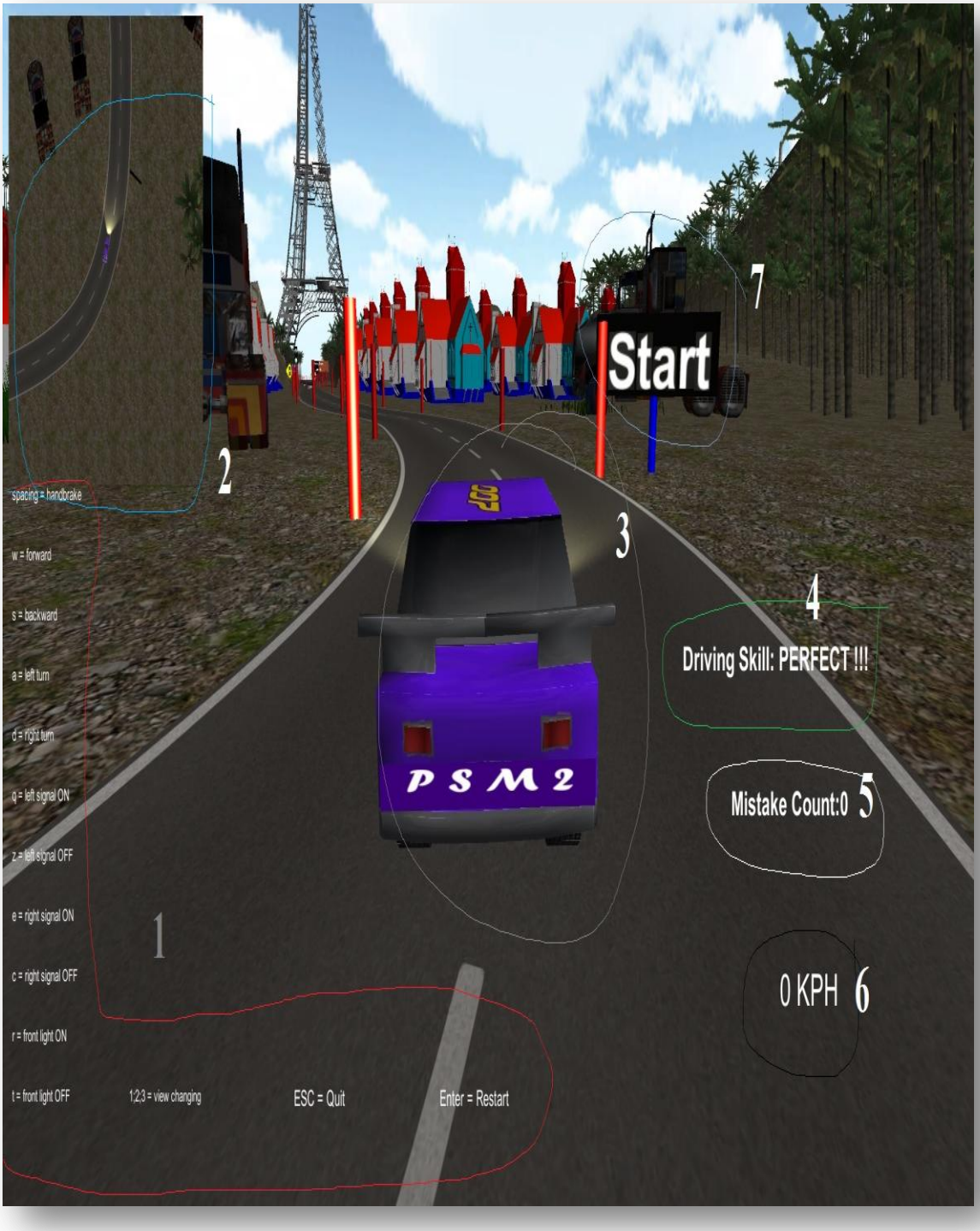


Figure 2.4: Display Interface

Number	function
1	Show the user on the functioning key on the keyboard. Can be disable the view by pressing F1 or showing the control manual by pressing F2
2	A mini map from the top view on the car to allow user observe the surrounding of the car
3	Car : main object which manipulate by user through this simulation
4	Driving Skill: showing the driving skill of the user in real time which affect by the mistake count.
5	Mistake Counts: counting the number of mistake occur by the user through this simulation
6	Speedometer: showing current speed of the car.
7	Start: Starting point of the simulation.

Table 2.1: Description of Figure 2.4

Figure 2.5: System Log-In Interface

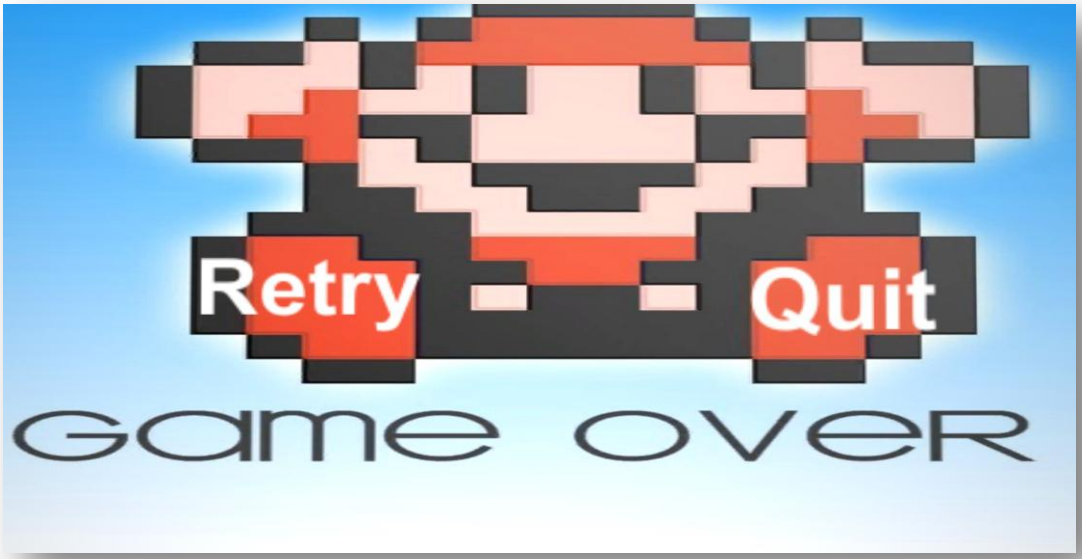


Figure 2.6: System Pop-out Interface (when user fails)

Figure 2.7: System Ending Interface (user successful Finish)

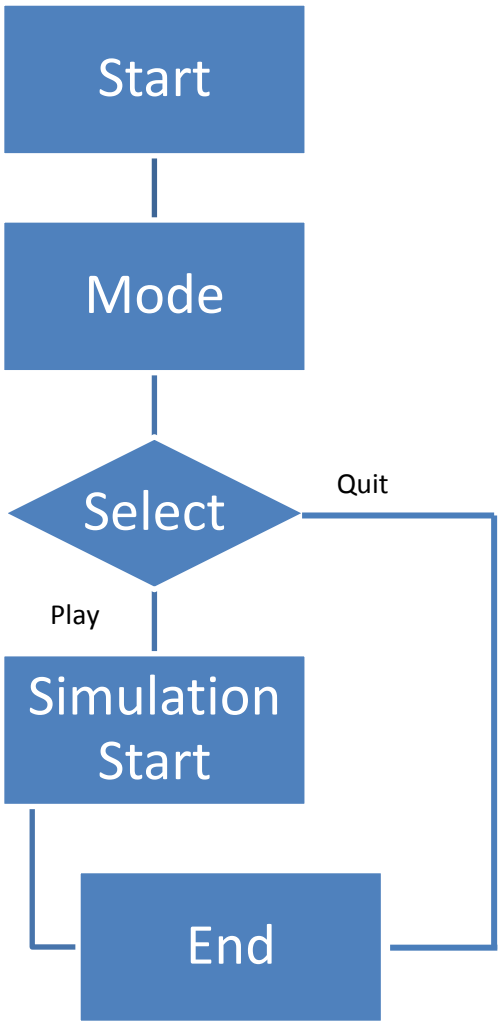


Figure 2.8: System Algorithm

In Figure 2.5, the system's initial interface is given user to choosing two sections which are Play mode and Rules. For Play mode users can start the driving simulation or enter Rules to understand the pathway and rules inside the simulation first before start. Figure 2.8 show the algorithm of this driving simulation system.

2.3 Development Plan

For the development part, there are several development methodology can be used. The methodology use for this project is Rapid Prototyping Model. Below diagram show the flow chart of the Rapid Prototyping method.

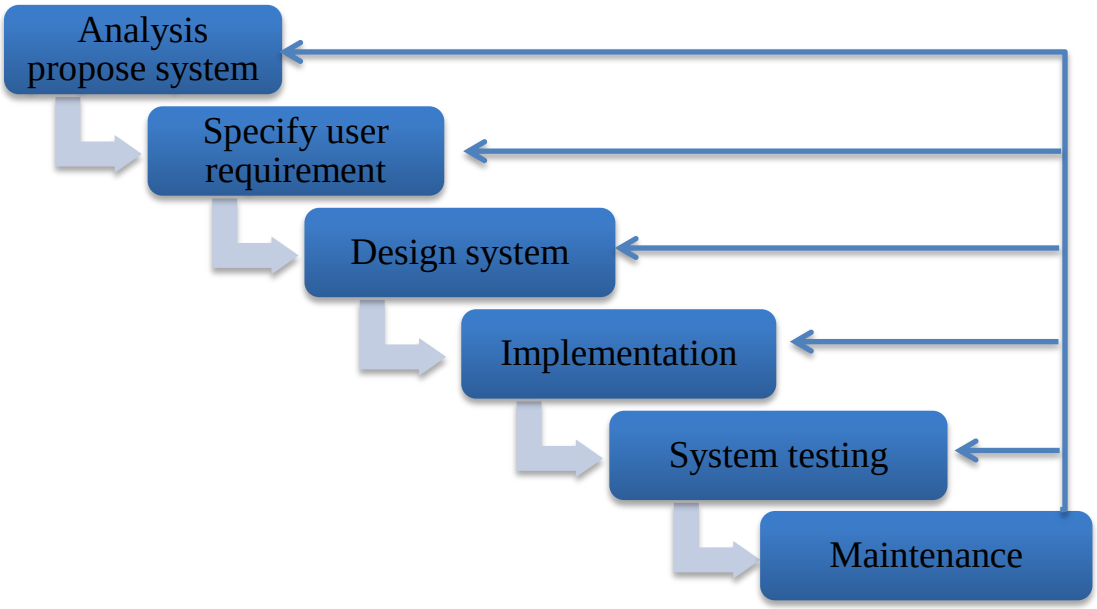


Figure 2.9: Development Plan

The Rapid Prototyping method consists of major six steps that are Analysis Propose System stage, Specify User Requirement stage, Design System stage, Implementation stage, System Testing stage and Maintenance stage. The first stage is Requirements which mean developer has to collect the user requirement and project requirement either in hardware and software part.

2.3.1 Analysis Propose System

For the analysis propose system stage, developer had to analysis the detail of the title which had proposed and confirmation of the value inside the system. Understand the problem facing and the objective to solve the problem through this system.

2.3.2 Specify User Requirements Stage

For the driving simulation development, at the requirement stage, the requirement finding and gathering are important to initial the development progress. A clear requirement can avoid waste of time and the risk of re-do the driving simulation development due to requirement undetected. The requirement gathering technique is using survey technique which is questionnaire. Distribute to 30 respondents randomly to collect their feedback base on existing driving simulation system and the need to add-on

inside new driving simulation system. After collecting the feedback, requirement analysis process will start to find the critical attribute which affect the efficiency of the system. Thus can improve or adding specific function into certain part according the user requirement data. In addition, hardware and software are also one of the requirements for the driving simulation system development. Suitable software and hardware selection are important because it can help and support in the following stage of the waterfall model. For the driving simulation development, Unity will be select as the main tools or software to develop the system. Hardware will be using computer as development tool and driving steering as control support inside the driving simulation.

2.3.3 Design system stage

The design stage for the driving simulation development is the phase to design the algorithm and the system's interface. At the algorithm part, how the driving simulation start until the end of the pathway need to be design well. To ensure the error or the confusion occur through the driving simulation. Besides that, specific driving course will be design for the user in order examine and training their driving skill inside virtual world. For example user can alter the situation inside the driving simulation by changing the learning course that user prefer. At system's interface design, the interface will be using 2D and 3D mode to generate the driving simulation course. Users can change their view either 2D view mode or 3D view mode. This can be helping them to observe and learn about the pattern of their driving thus achieve driving learning purpose. Overall design stages are acquiring the requirement at previous stage then design the architecture by following design specification.

2.3.3.1 Step 1

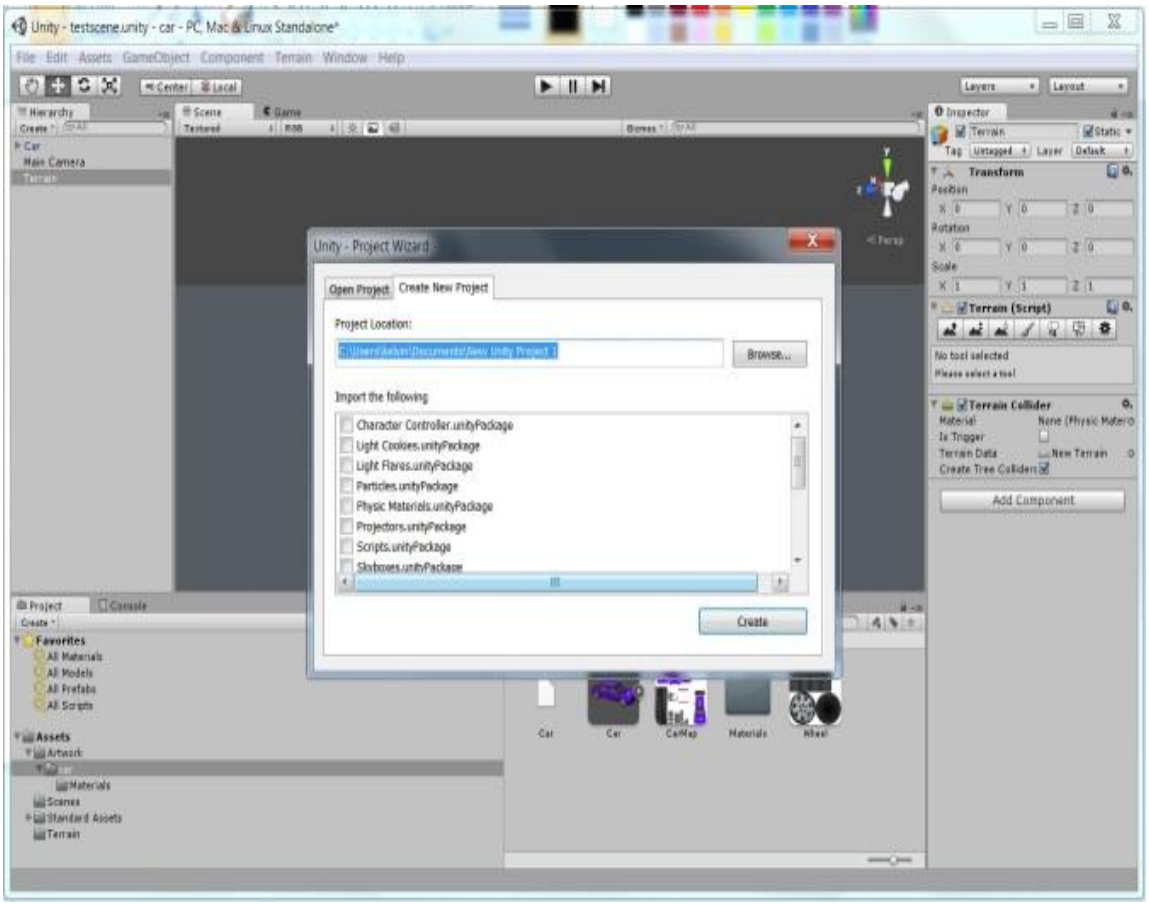


Figure 2.10: First Step

Firstly, before start to create the car simulation game had to create a new project and click the following necessary asset to import inside the project as show in figure 2.10.

2.3.3.2 Step 2

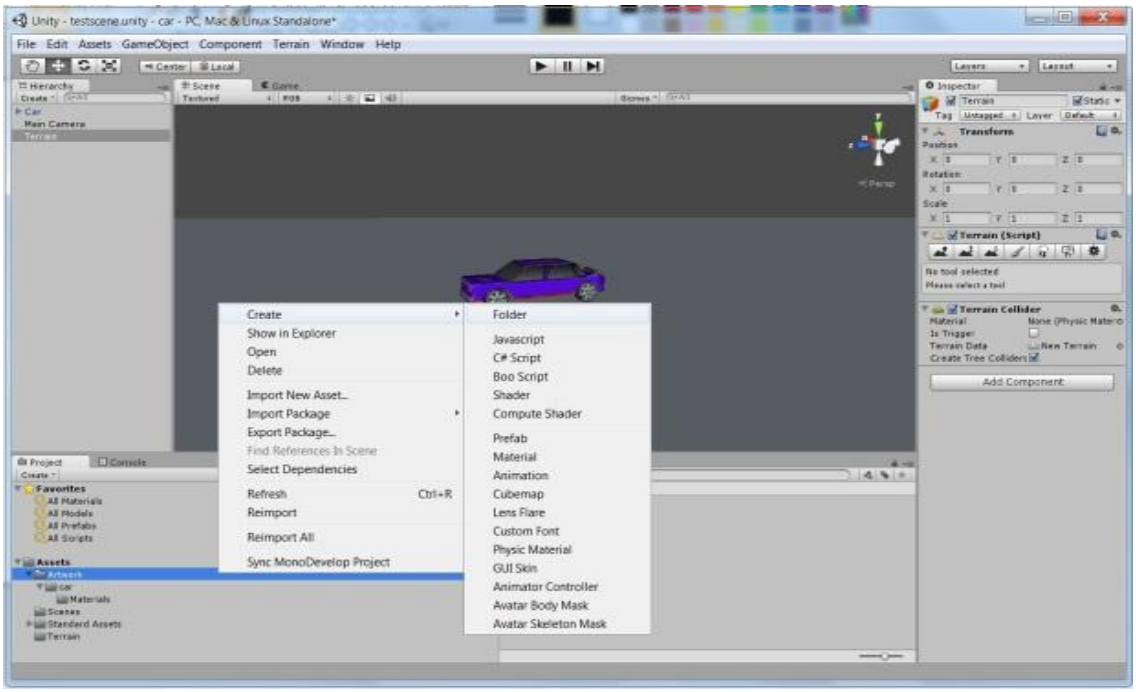


Figure 2.11: Create Folder

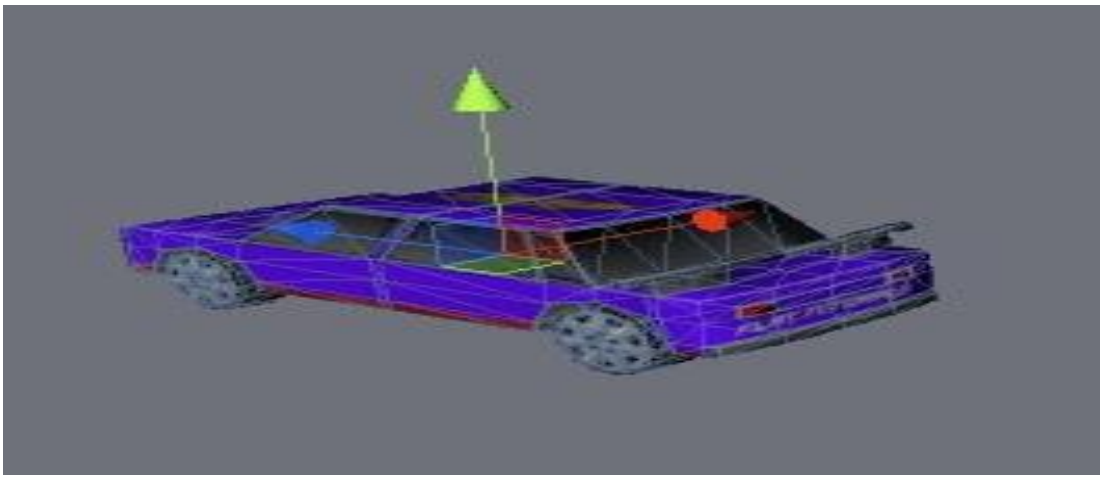


Figure 2.12: Import Car Model

Then create the folder inside the project panel there to put the certain asset as a child of a model, terrain, audio or script to easier in categories as show in figure 2.11. Figure 2.12 shows the 3D model of the car inside the project.

2.3.3.3 Step 3

Figure 2.13: Set Main Camera

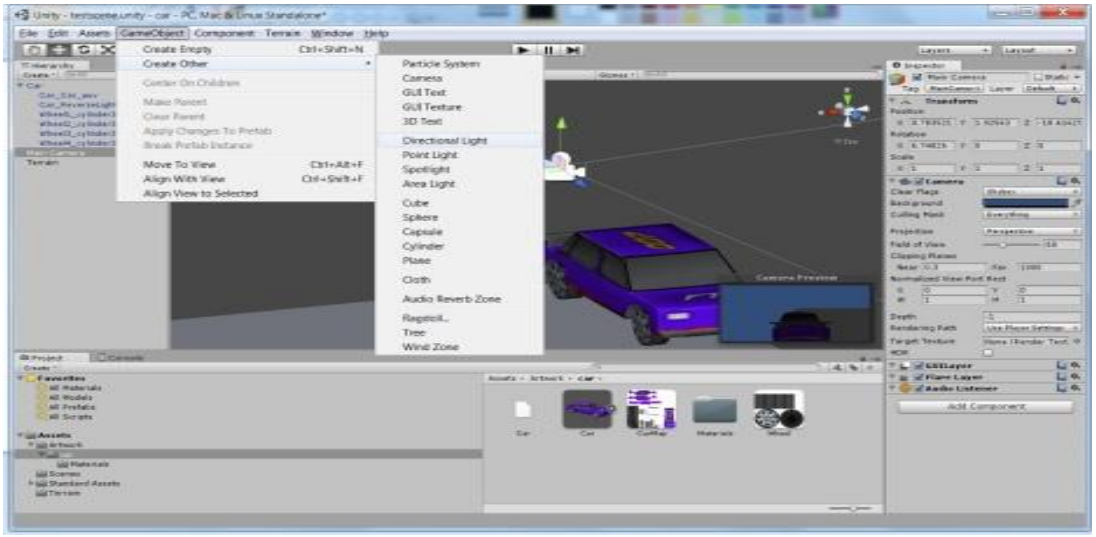


Figure 2.14: Create Light

Figure 2.13 shows the setting of the main camera. Main camera functions as the user or player's view during the simulation. The main camera will set the position at the upper top at the behind of the car thus user can view the movement of the entire car. Directional light will be set at the same position with main camera because of to ensure a clear view with enough light intensity for user during function the simulation. In addition, both of the components will be code to stick with the car model thus the view can follow the car track.

2.3.3.4 Step 4

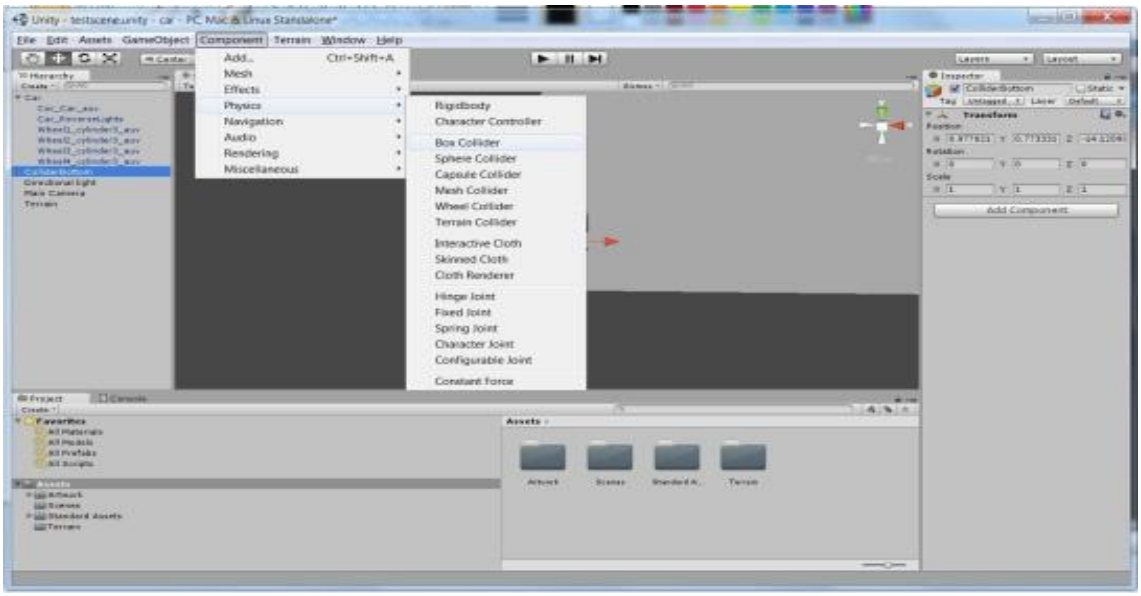


Figure 2.15: Set Box Collider

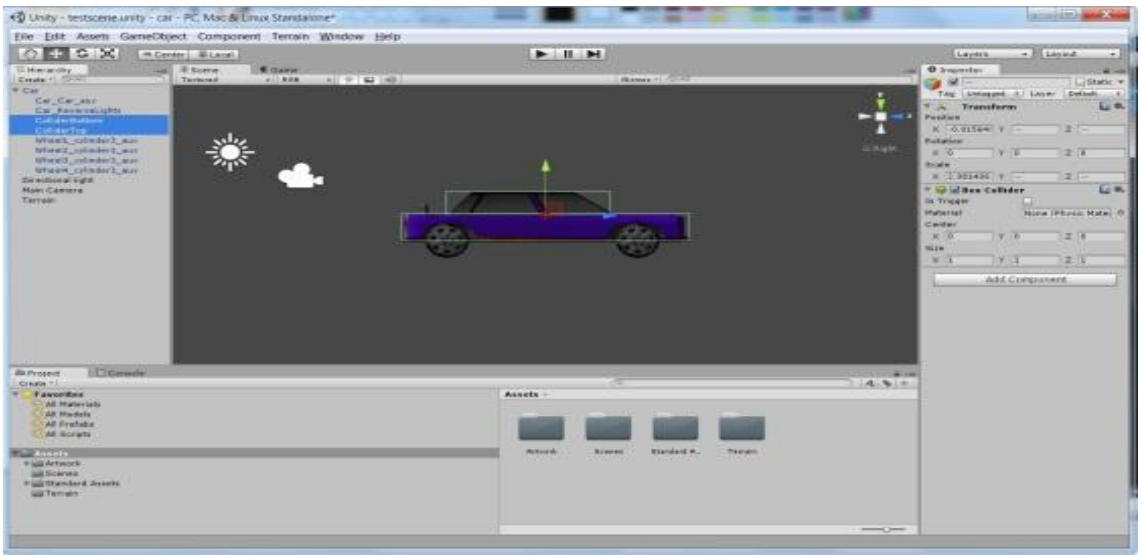


Figure 2.16: Set Position

Now the collider will be set for the body of the car as show in figure 2.15. The reason to set a collider for car is because by setting a collider in car so the car can land on the terrain; without collider the car will be fall under the terrain just like a ghost with view but without physical body. Figure 2.16 show the where the position of box collider will be put inside the car model.

2.3.3.5 Step 5

Figure 2.17: Import SkyBox

Figure 2.18: Set Car Surface Texture

Next is the setting of the texture for terrain. As show in figure 2.17, the terrain floor had set the texture of sand track and the sky is using the skybox to and select Sunnyday2 material. Then implement the reflection effect for the surface of the car to make it look more realistic by select diffuse reflective as show in figure 2.18.

2.3.3.6 Step 6

```
function Controle(){
currentSpeed = 2*22/7*WheelRL.radius*WheelRL.rpm*60/1000;
currentSpeed = Mathf.Round(currentSpeed);
if (currentSpeed < topSpeed && currentSpeed > -maxReverseSpeed && !braked ){
WheelRR.motorTorque = maxTorque * Input.GetAxis("Vertical");
WheelRL.motorTorque = maxTorque * Input.GetAxis("Vertical");
}
else {
WheelRR.motorTorque =0;
WheelRL.motorTorque =0;
}
if (Input.GetButton("Vertical")==false){
WheelRR.brakeTorque = decellarationSpeed;
WheelRL.brakeTorque = decellarationSpeed;
}
else{
WheelRR.brakeTorque = 0;
WheelRL.brakeTorque = 0;
}
var speedFactor = rigidbody.velocity.magnitude/lowestSteerAtSpeed;
var currentSteerAngel =
Mathf.Lerp(lowSpeedSteerAngel,highSpeedSteerAngel,speedFactor);
currentSteerAngel *=Input.GetAxis("Horizontal");
WheelFL.steerAngle = currentSteerAngel;
WheelFR.steerAngle = currentSteerAngel;
}
```

Figure 2.19: Code for Car Control

```
using UnityEngine;
public class camerafollow : MonoBehaviour
{
        public Transform Target;

        void LateUpdate()
        {
                transform.position = new
Vector3(Target.position.x,transform.position.y,Target.position.z);
        }
}
```

Figure 2.20: Code for Main Camera

After setting the car model then come to coding the movement of the car. The code for the basic movement of car is code inside the CarControlScript as show in figure 2.19. And the main camera which follows the movement of the car will code inside CarCameraScript as show in figure 2.20. Besides create the basic movement of the car but also code to create the transform angle for the car's front tires so the front tires can turn left or right's angle slightly when user turn left or right thus it will make the car movement more close to realities. The code is highlighted in figure 2.19.

2.3.3.7 Step 7

Figure 2.21: Design of Backlight Texture

```
if (braked){
WheelFR.brakeTorque = maxBrakeTorque ;
WheelFL.brakeTorque = maxBrakeTorque ;
WheelRR.motorTorque =0;
WheelRL.motorTorque =0;
if (rigidbody.velocity.magnitude>1){
SetSlip(slipForwardFriction ,slipSidewayFriction);
}
else {
SetSlip(1 ,1);
}
if (currentSpeed < 1 && currentSpeed>-1){
backLightObject.renderer.material = idleLightMaterial;
}
else {
backLightObject.renderer.material = brakeLightMaterial;
}
```

Figure 2.22: Code for Car's Backlight

Next is creating the back light of the car model as show in figure 2.21. The back light of car need to be code when pressing the brake and backward then only will active. But actually is not the light but changing of the texture from dim red to light red colour. The code is show in figure 2.22.

3.2.8 Step 8

```
function EngineSound(){
for (var i =0; i < gearRatio.length; i++){
if(gearRatio[i]> currentSpeed){
break;
}
}
var gearMinValue : float =0.00;
var gearMaxValue : float=0.00;
if (i==0){
gearMinValue=0;
}
else{
gearMinValue=gearRatio[i-1];
}
gearMaxValue=gearRatio[i];
var enginePitch : float = ((currentSpeed-gearMinValue)/(gearMaxValue -
gearMinValue))+1;
audio.pitch = enginePitch;
}
```

Figure 2.23: Code for Engine Sound

```
#pragma strict
var destroyAfter : float = 2;
private var timer : float;
function Start () {

}

function Update () {
timer += Time.deltaTime;
if (destroyAfter <= timer){
Destroy(gameObject);
}
}
```

Figure 2.24: Code for Destroy Temporary Data

Next is creating the engine sound by import the sound audio and implement for car model. The flow is when the simulation start the engine sound will be start at low volume but repeat then when user speed up the car the engine sound will increase its volume by looping and creating multiple times. The code is shown in figure 2.23. Because of looping thus another script name as Destroy had to be creating to get rid or delete of the sound product create by looping function to ensure the smoothness process in the simulation. Code is shown in figure 2.24.

2.3.3.9 Step 9

```
#pragma strict
private var currentFrictionValue : float;
var skidAt : float = 1.5;
var soundEmition : float =15;
private var soundWait : float;
var skidSound : GameObject;
var skidSmoke : GameObject;
var smokeDepth: float = 0.4;
var markWidth : float = 0.2;
private var skidding : int;
private var lastPos = new Vector3[2];
var skidMaterial : Material;
function Start () {
skidSmoke.transform.position = transform.position;
skidSmoke.transform.position.y -= smokeDepth ;
}
function Update () {
var hit : WheelHit;
transform.GetComponent(WheelCollider).GetGroundHit(hit);
currentFrictionValue = Mathf.Abs(hit.sidewaysSlip);
if( skidAt <= currentFrictionValue && soundWait <=0){
Instantiate(skidSound,hit.point,Quaternion.identity);
soundWait = 1;
}
soundWait -= Time.deltaTime*soundEmition;
if( skidAt <= currentFrictionValue ){
skidSmoke.particleEmitter.emit = true;
SkidMesh();
}
else {
skidSmoke.particleEmitter.emit = false;
skidding = 0;
}
}
function SkidMesh () {
var hit : WheelHit;
transform.GetComponent(WheelCollider).GetGroundHit(hit);
var mark : GameObject = new GameObject("Mark");
var filter : MeshFilter =mark.AddComponent(MeshFilter);
mark.AddComponent(MeshRenderer);
var markMesh : Mesh = new Mesh ();
var vertices = new Vector3 [4];
var triangles = new int [6];
if (skidding ==0){
vertices[0] = hit.point +
```

```
Quaternion.Euler(transform.eulerAngles.x,transform.eulerAngles.y,transform.eulerA
ngles.z)* Vector3(markWidth,0.01,0);
vertices[1] = hit.point +
Quaternion.Euler(transform.eulerAngles.x,transform.eulerAngles.y,transform.eulerA
ngles.z)* Vector3(-markWidth,0.01,0);
vertices[2] = hit.point +
Quaternion.Euler(transform.eulerAngles.x,transform.eulerAngles.y,transform.eulerA
ngles.z)* Vector3(-markWidth,0.01,0);
vertices[3] = hit.point +
Quaternion.Euler(transform.eulerAngles.x,transform.eulerAngles.y,transform.eulerA
ngles.z)* Vector3(markWidth,0.01,0);
lastPos[0] = vertices[2];
lastPos[1] = vertices[3];
skidding = 1;
}
else {
vertices[1] = lastPos[0];
vertices[0] = lastPos[1];
vertices[2] = hit.point +
Quaternion.Euler(transform.eulerAngles.x,transform.eulerAngles.y,transform.eulerA
ngles.z)* Vector3(-markWidth,0.01,0);
vertices[3] = hit.point +
Quaternion.Euler(transform.eulerAngles.x,transform.eulerAngles.y,transform.eulerA
ngles.z)* Vector3(markWidth,0.01,0);
lastPos[0] = vertices[2];
lastPos[1] = vertices[3];
}
triangles = [0,1,2,2,3,0];
markMesh.vertices =vertices;
markMesh.triangles=triangles;
markMesh.RecalculateNormals();
var uvm: Vector2[] = new Vector2[4];
uvm[0]=Vector2(1,0);
uvm[1]=Vector2(0,0);
uvm[2]=Vector2(0,1);
uvm[3]=Vector2(1,1);
/*var uvm = new Vector2[markMesh.vertices.length];
for (var i = 0; i < uvm.length; i++){
uvm[i] = Vector2 (markMesh.vertices[i].x,markMesh.vertices[i].z);
}*/
markMesh.uv = uvm;
filter.mesh = markMesh;
mark.renderer.material=skidMaterial;
mark.AddComponent(DestroyTimerScript);
}
```

Figure 2.25: Code for Create Skidding

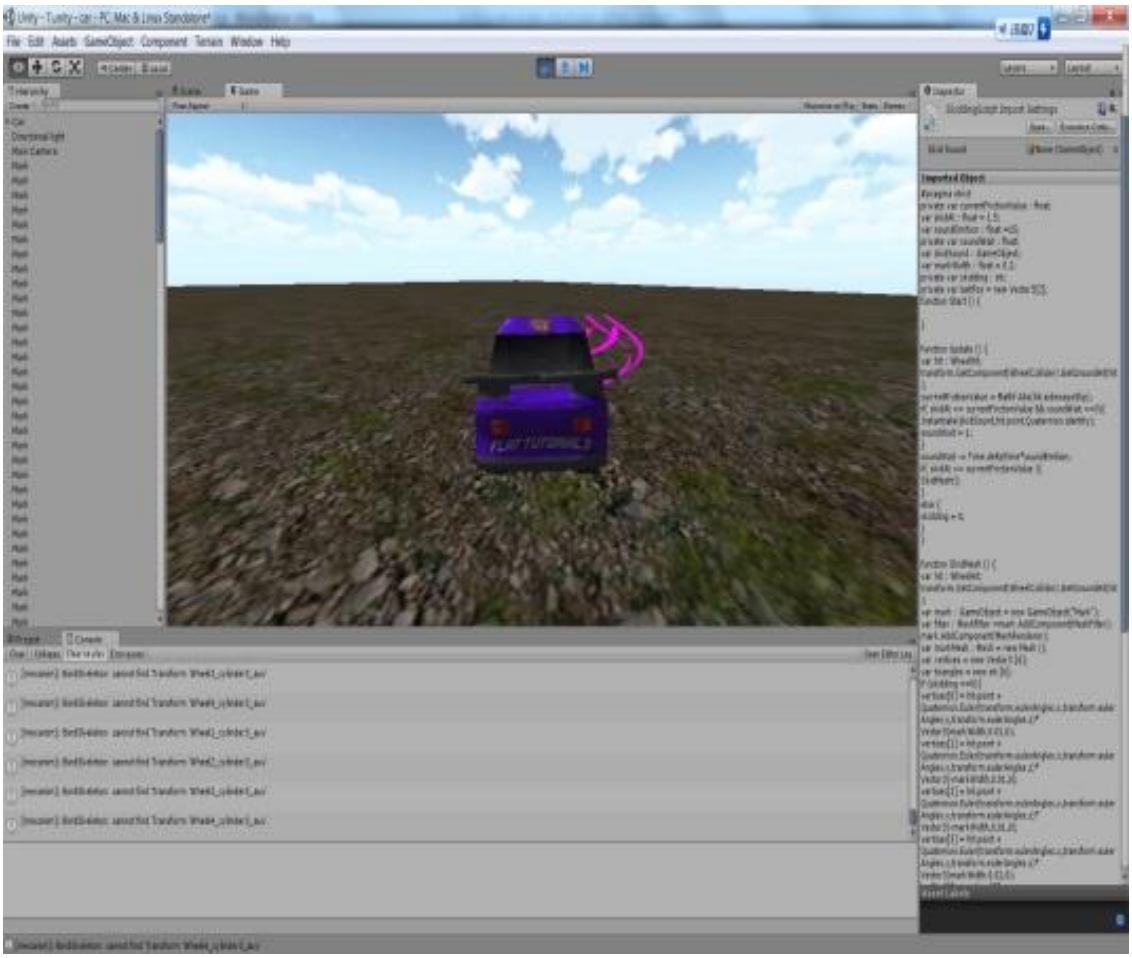


Figure 2.26: Skidding

In figure 2.25 it show the code of skidding. Skidding is to create the car tires crack when car driving too fast and suddenly had a fast turn or brake. The result is show in figure 2.26 which the pink colour shows the car tires track when car is driving too fast and suddenly press the handbrake. Besides that, the skidding sound is implementing inside the car model which similar as the engine sound function. So when the car tires track produce, it also produces the sound of skidding.

2.3.3.10 Step 10

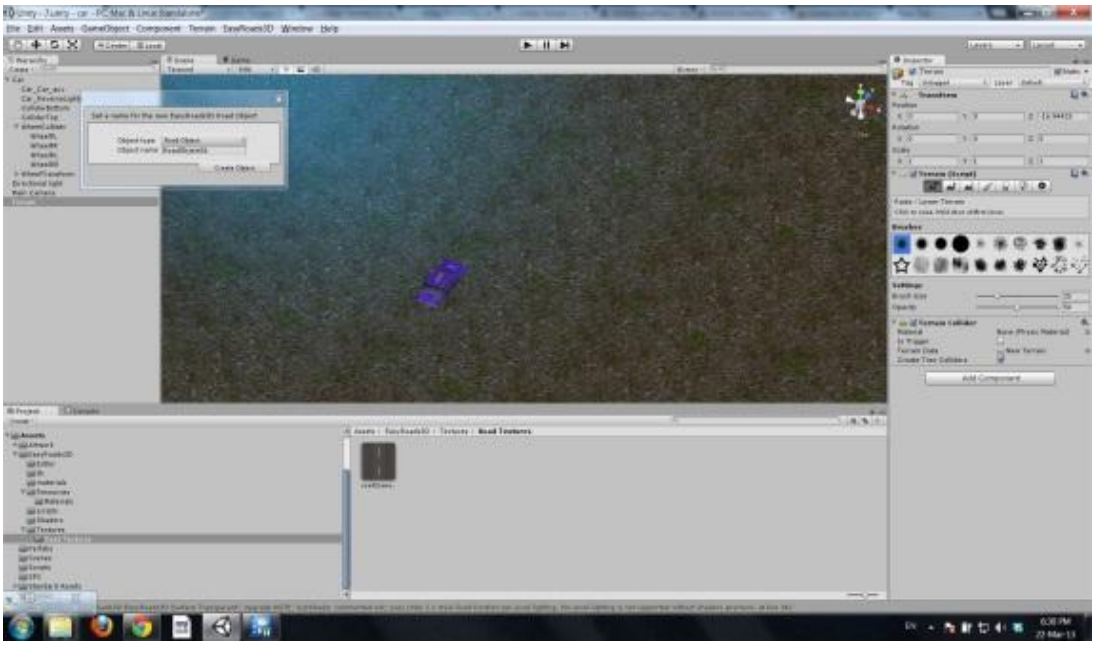


Figure 2.27: Create RoadTrack

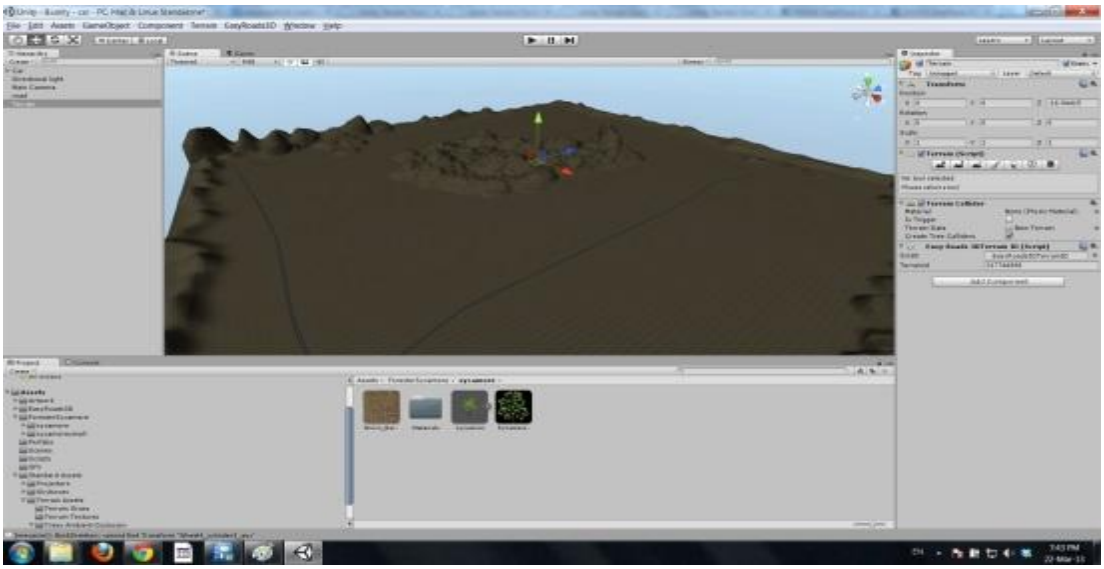


Figure 2.28: Result of RoadTrack

Figure 2.27 shows how to create the road track by using the plugin of the Unity3D which is EasyRoad3D to create the road. But due to free version so it only provide a basic road pattern. Figure 2.28 shows the modification and result of the terrain and the road track.

2.3.3.11 Step 11

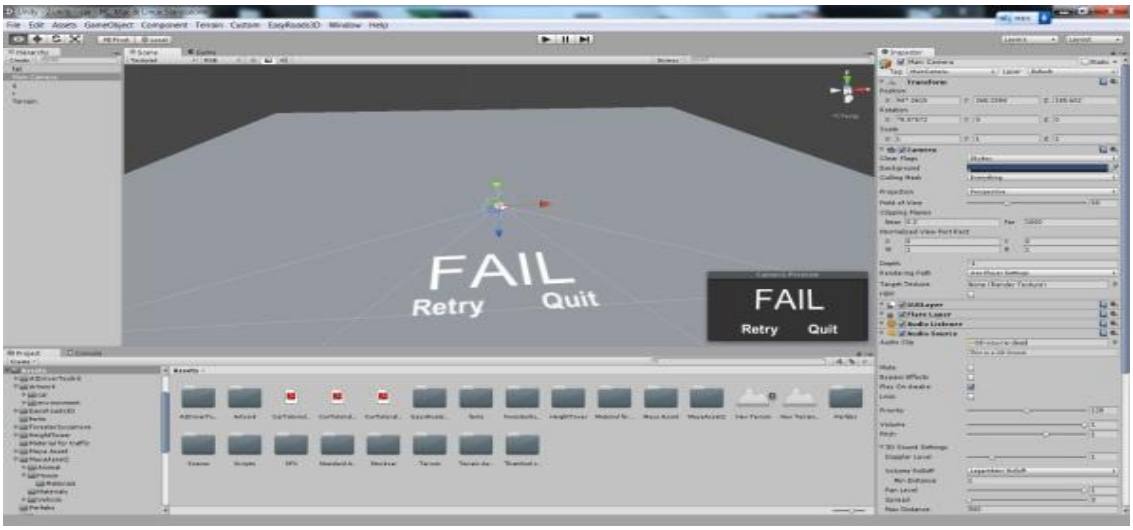


Figure 2.29: Fail Interface

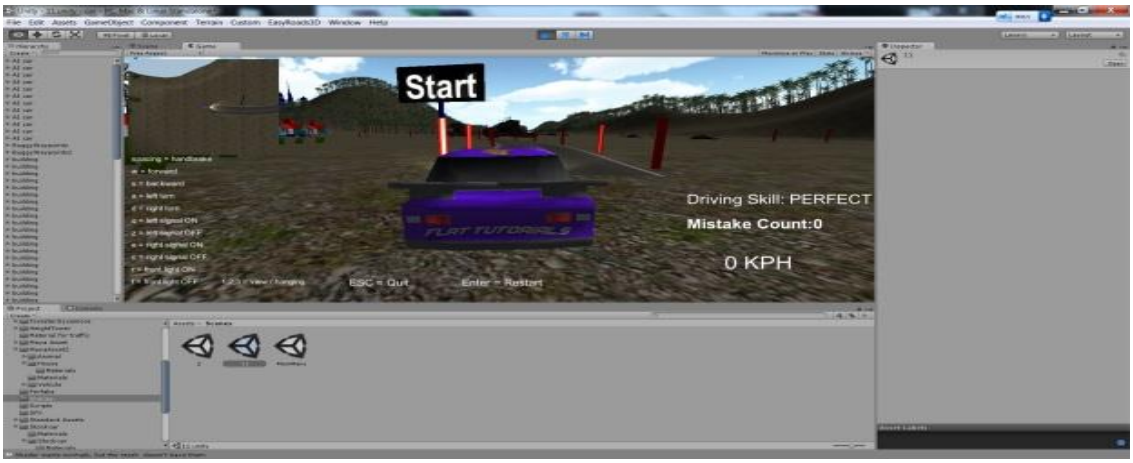


Figure 2.30: Starting Point

Figure 2.29 show the scene which will show if the user had fail his driving simulation game, which is make mistake more than 20. It notice a FAIL word to user and below consist two button which are Retry and Quit allow user to select whether want to restart the simulation or exit the program. Figure 2.30 show the overall gameplay during car simulation, upper left show the mini map which views the car surrounding from top view. Below the white text show the functional key on keyboard on how to control the car. Right side show the mistake count, speedometer and the driving skill of the user which depend on the mistake count.

2.3.4 Implementation Stage

Implementation stage is using the architecture which builds at the previous stage by input the coding, build the design of the system and media editing. The implementation of the system will be using software Unity by implement the game programming language inside the driving simulation system. The game programming language will be uses inside Unity are C++ and JavaScript. The product develop from this stage maybe one or more than one since it is build according to a pre-define coding standard and debugged, tested and integrated to satisfy the system architecture requirements. Certain changing maybe occurs due to the inability to satisfy the system architecture requirements. Figure below show the implemented material inside the system and step of how to function it.

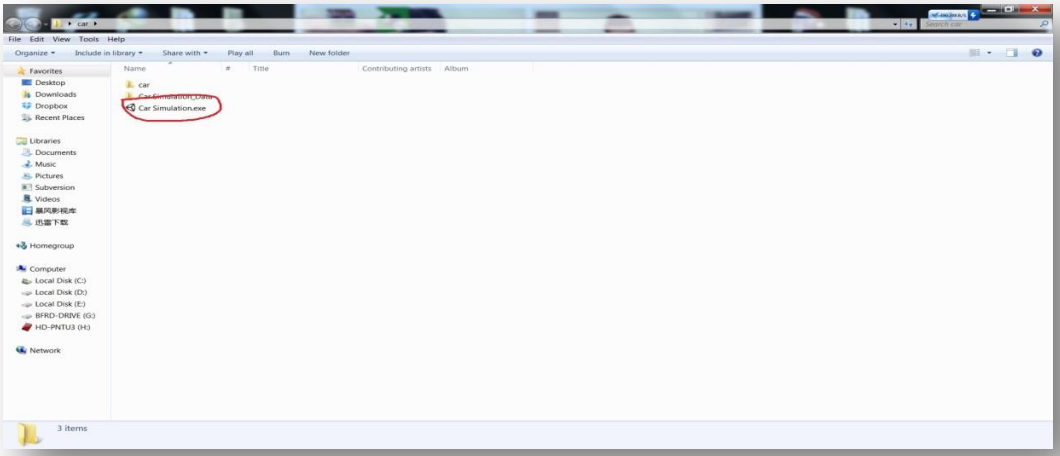


Figure 2.31: Manual 1

1. Select the car simulation exe file (red circle) as show on the figure 2.31.

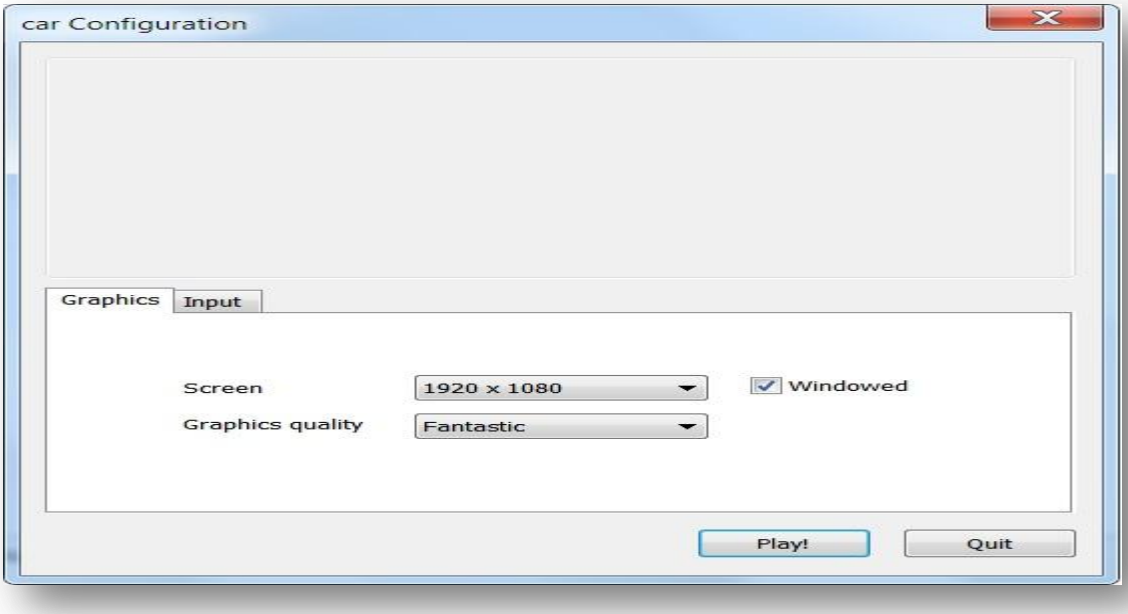


Figure 2.32: Manual 2

2. After click the exe it will pop out the car configuration box as show in figure 2.32, select any screen resolution and graphics quality as user prefer then click play button.

Figure 2.33: Manual 3

3. The main menu of the car driving simulation will show. It provides two selective choices which are Play and Quit. User can select either both to enter the simulation or exit this program.

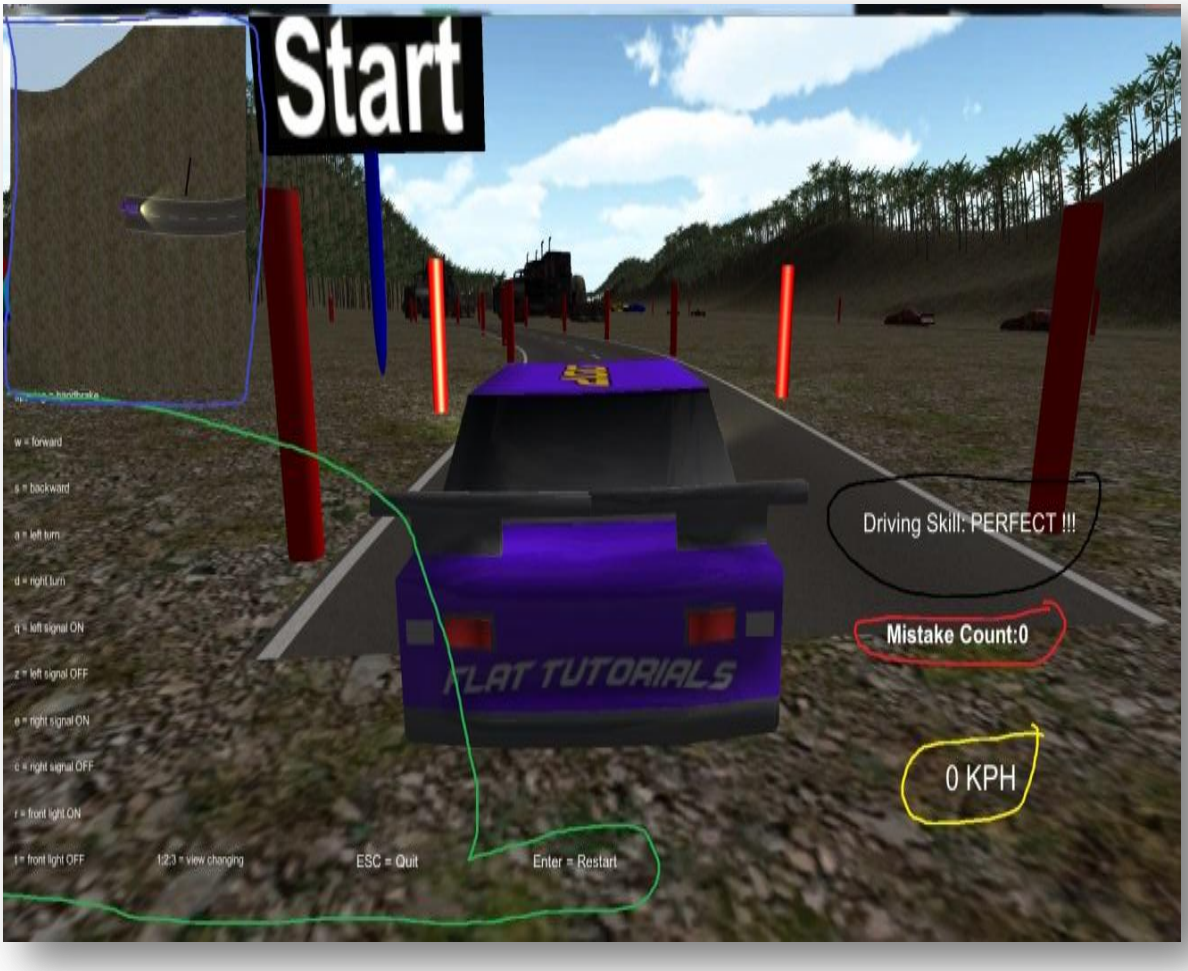


Figure 2.34: Manual 4

4. When click play at the main menu, it will jump into the simulation which as show in figure 2.34. The car will initially place at the Start place then user had to drive follow the road. Blue circle show the mini map which is the top view of the car. Green circle show the functional keyboard. Yellow circle show the speedometer of the car. Red circle show the mistake count during the simulation. And the last one black circle show the driving skill of the user during this simulation on real time which is depending on the number of mistake count. Initially the driving skill is PERFECT but it will drop when the mistake count increase.

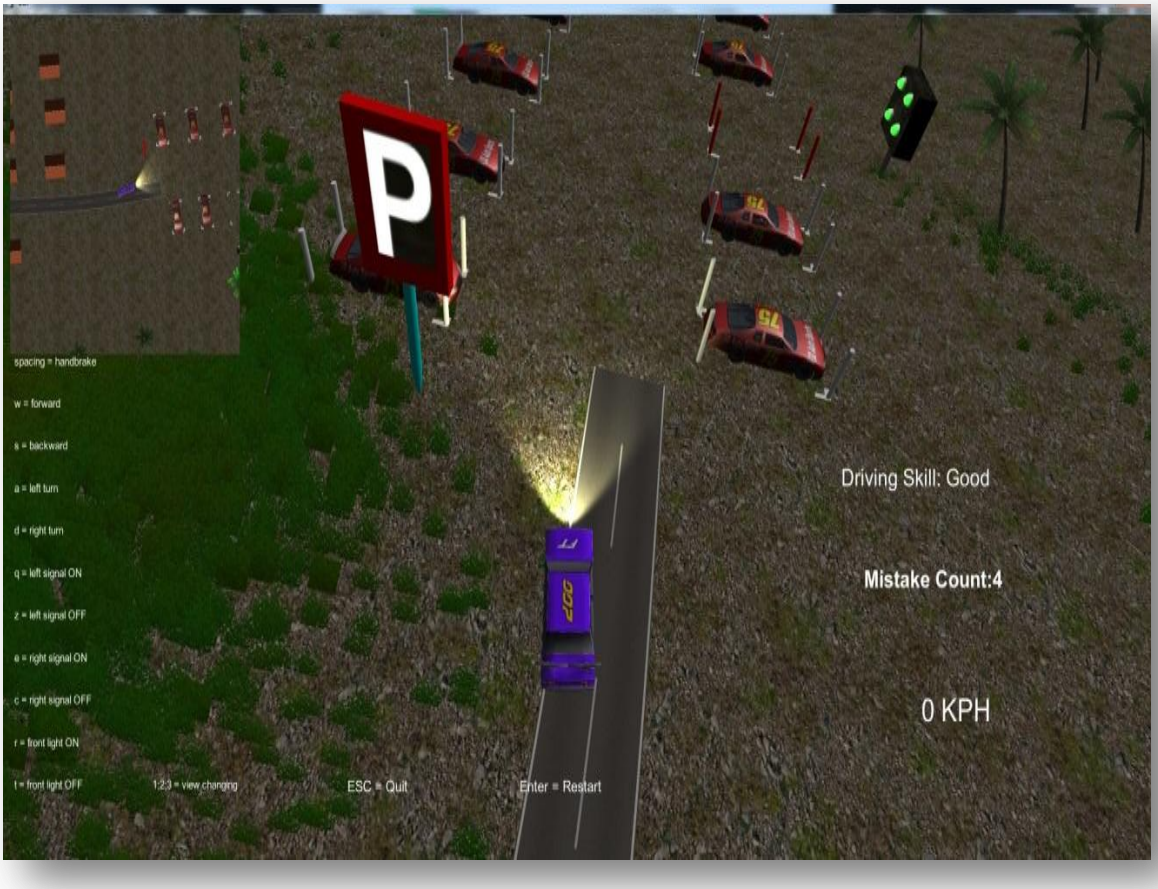


Figure 2.35: Manual 5

5. After depart from starting place at the first T-junction user can see there is a right turn which is for practice the parking skill.

Figure 2.36: Manual 6

6. This parking test is built for user to park the car inside the middle of the parking slot without knock fall the red cylinder model. If knock fall the notice board (black board with four green lights) will change to red colour depending on which cylinder model had fall. At the figure 2.36, the white circle on the board is red due to the position of which circle at red cylinder is knocking off. The green circle follows the same rule also.

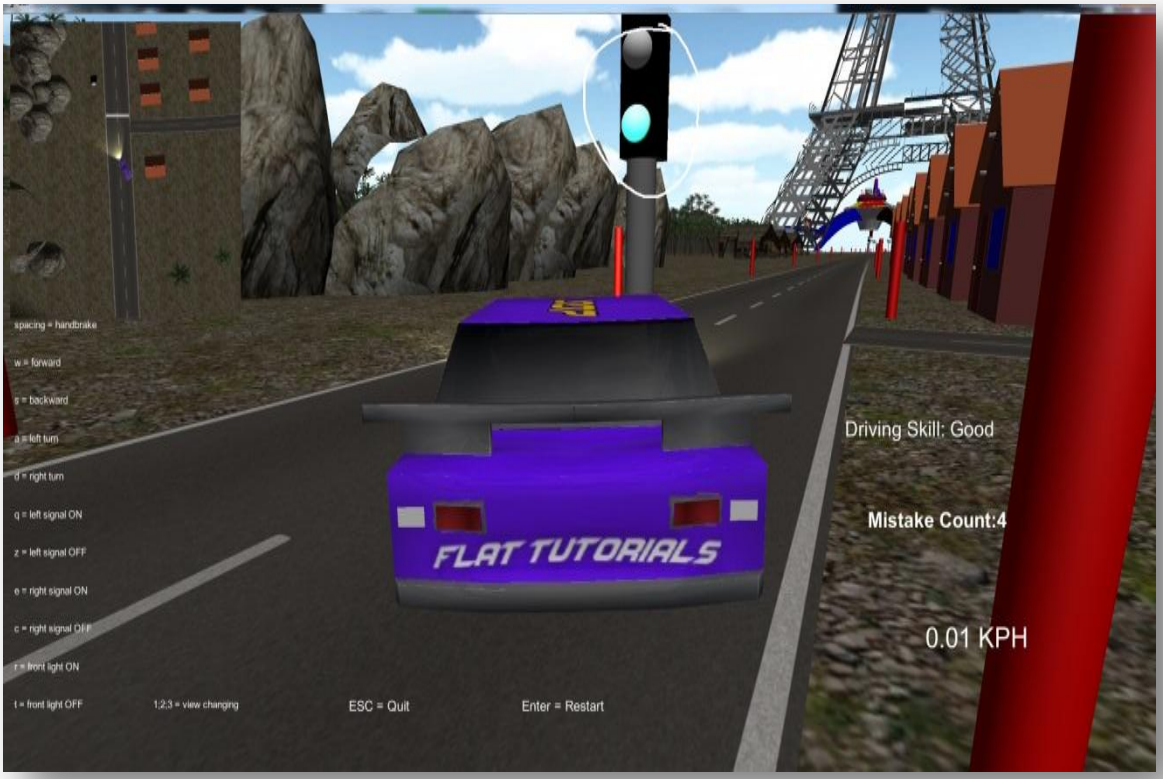


Figure 2.37: Manual 7

7. At the first T-junction there will a first traffic light, if user didn't follow traffic rule which drive through a red light, the mistake count will be add two since this is a serious mistake.

Figure 2.38: Manual 8

8. If user's mistakes count is lower than 20, and reach the End point which show in figure 2.38, then the simulation is finish.

Figure 2.39: Manual 9

9. When the simulation is finish without exceeding 20 mistake counts and reach the finish point, the result will pop out as shown in figure 2.39.

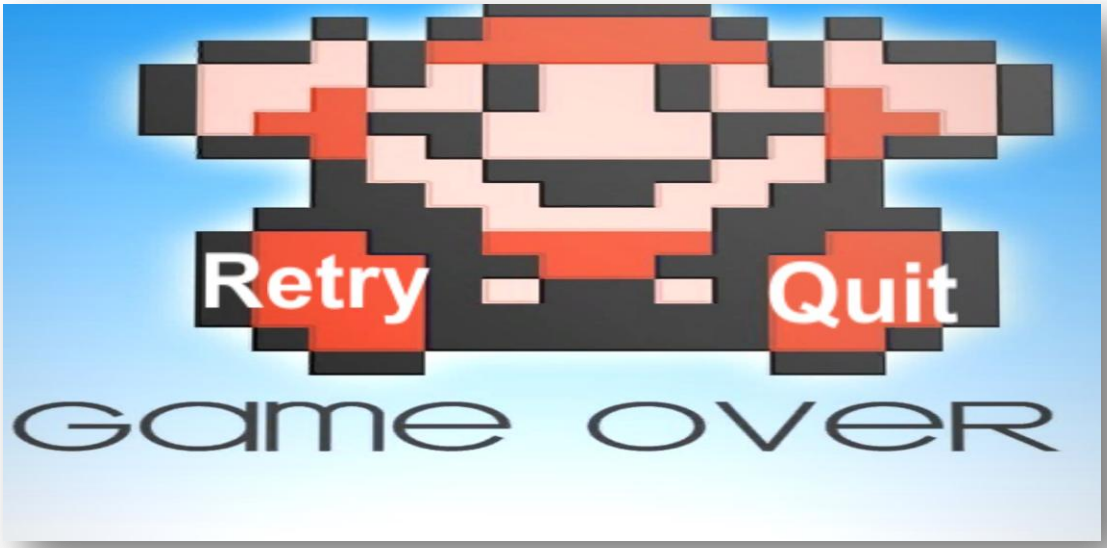


Figure 2.40: Manual 10

10. If user had reach 20 mistake count the simulation will force to stop while pop out this notification. User either can select Retry to use the simulation again or select Quit to exit the simulation program.

2.3.5 System Testing stage

Inside the system testing stage for the driving simulation system, majorly consist 2 part which are testing and installation. At testing part, initial product or prototype will be test by few testers in order to find the error, bug and function's ability of the system. Correction will be apply if any error or bug occur thus will go back to previous stage implementation. Next one is installation phase; the installation will be testing by install in several computers to test the functional of the driving simulation system since the final product will be burn into a CD as a portable causeway of driving simulation. For the testing part consisting 2 parts which are testing during the development and another is test the prototype by the tester.

(A) Development Phase Testing

Testing 1: Car movement and feature functionality

Problem 1: the car unable stands on the terrain but fall under the terrain.

Solution: By discover through online the solution is had to set a collider for the car thus the object can consisting a physical body then can stand on the terrain.

Problem 2: searching the camera views of the car where the suitable placement is.

Solution: through testing of the location of camera view at every possible place, the final decision is place at the upper behind of the car thus user can view the entire car while driving.

Problem 3: how to generate the backlight when press backward button (S) on keyboard?

Solution: by writing script putting two games object which for the left backlight. Initially grey colour of the game object plane and another game object plane putting as red colour. So when key S buttons the grey colour will disable and the red colour will enable. It will generate the backlight of the car.

Problem 4: how to generating the smoke and the skidding of the car?

Solution: The car tires skidding and smoke both require texture thus Photoshop is required to use for design those textures. After design the textures it will implement into a game object thus adding the script for the smoke and the skidding which connect together. Result will be when skidding is happening the smoke will be generating.

Problem 5: how to creating the collider detection for the car to other object?

Solution: putting the if else function for the collider thus can putting every object inside of the mistake count function thus each collide will increasing the number count of the mistake. By putting the car as the object thus every object collide by car will be add into the count.

Problem 6: how to generate the traffic light and the detection of passing through red light penalty?

Solution: create the cylinder with 2 sphere on it make it look like a traffic light, thus adding two light which 1 is red another is green colour which point on the sphere make it look like a real traffic light. Then create a script writing the lightswitch function and setting the period of changing about 10 second so while red colour light is ON the green colour light will be OFF but after 10 seconds the red colour light will be OFF and the green colour light will be ON. The traffic light penalty when passing through the red light is same as previous by create a cube and putting the collider detection inside it thus place on the terrain right after the traffic light to act as a sensor then adding 1 script link the red colour light and the cube together. So the result will be when red colour light is ON and the cube is colliding by the car so the mistake count will be +1.

Problem 7: how to set the signal can blink when click button?

Solution: using Boolean function to put WaitForSecond code to set the period of blinking the signal. Initially set the signal to invisible thus set the material on for orange colour (click signal) so it will starting to blink according the second input; click button Z to stop the signal by changing the texture to transparent material so the blinking still function but is invisible for user.

Testing 2: simulation environment observation

Problem 1: how to setting the terrain and the sky with texture?

Solution: terrain's texture initially unity consist of few texture, so by trying on those texture the rough mud road texture had been select for the terrain. The sky had to using the skybox to add the sky texture.

Problem 2: how to create the road path for simulation?

Solution: first trying is using the plane to design the road but at the end fail due to the adjustment is difficult. Then the plug-in which name as EasyRoad3D in the unity is used to generate the road for driving.

Problem 3: How to create tree?

Solution: by using the default function of terrain, it provides an easy way to create tree by using terrain tree plan function. Main selection is the palm tree for this environment.

Testing 3: Parking Test

Problem 1: how to link and create the parking place, the result board and the rules?

Solution: the parking place will be putting 4 red cylinders along with collide detection function inside it and link with the 4 light on the result board. The positions of the 4 green light represent the same position of the 4 red cylinders. So when user try park the car inside the parking slot, if the user's car knock fall either 1 of the red cylinder, it will trigger the green colour of the same position where the red cylinder is fall off become red colour.

Testing 4: AI car routine confirmation

Problem 1: lacking of information for setting the AI car waypoint

Solution: install the plug-in of the unity which name as buzzyAIcar to create the AI way point so the AI car will follow the way point as their path.

Testing 5: Human Character building and movement

Problem 1: didn't know how to generate a human character which moving with its own skeleton

Solution: in order to create people crossing the road, alternative way had been used which is creating a board with human picture on it then script it to move across the road repeatedly. So this action can form a penalty if the car collide to a human board which act as people hit by a car. In result this forming a penalty and will increasing the mistake count during the simulation.

(B) Target User Testing (30 Respondents)

Please fill in with relevant information

Gender: ☐

Age: ☐

First Try

Fail ☐ Success ☐

If Success at first try

Time uses for finish this simulation: <5minutes ☐ <10minutes ☐ <15minutes ☐

Mistake count: ☐

Skill level: ☐

Control of the car**: Easy** ☐ **Medium** ☐ **Hard** ☐

Rating for this simulation (1-5): ☐

Learning the skill of driving: **No** ☐ **Yes** ☐ **Some** ☐

Success in second try

Mistake count decreasing? **Yes** ☐ **No** ☐

If fail at first try

Reason:

Hard ☐

Lag ☐

Quit Simulation ☐

Other reason: ☐

The graph analysis for the 30 Tester's response answer:

First Try

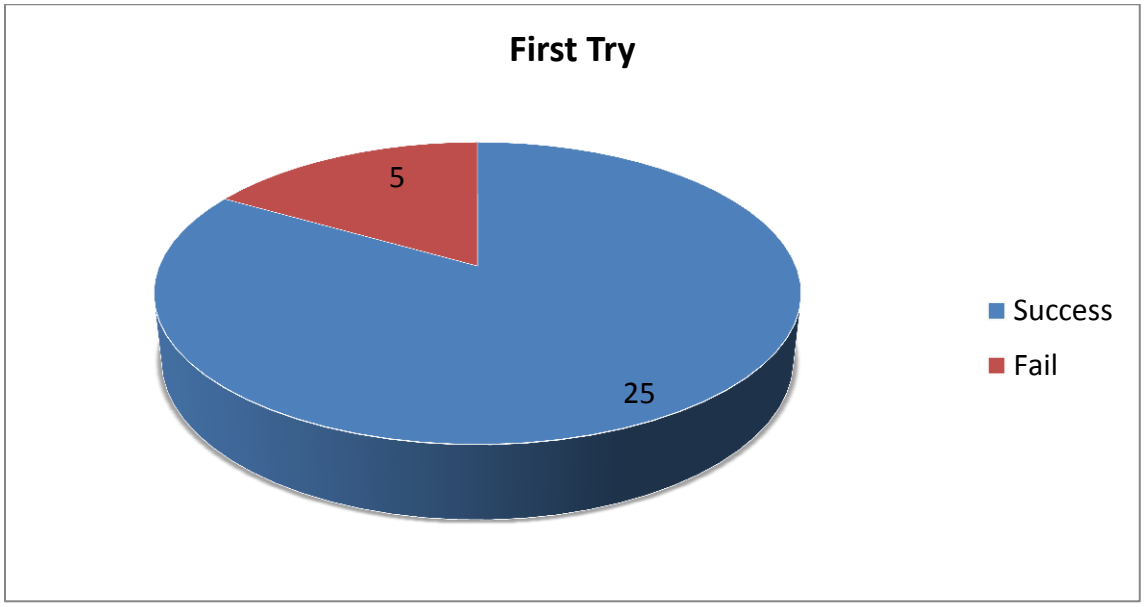


Figure 2.41: First try

Figure 2.41 shows the result of the 30 Tester after testing the system. Result show 5 of the 30 Tester had failed the simulation but the remains 25 were success complete the simulation.

If successes at first try:

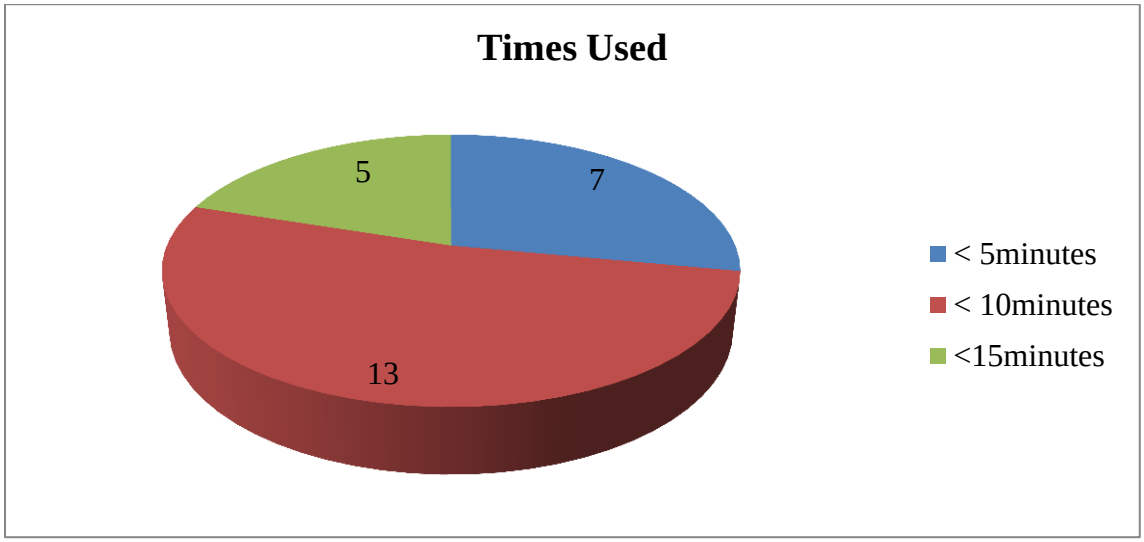


Figure 2.42: Time Used

Figure 2.42shows the pie chart of the time used for completing the simulation from the 25 Tester who successes complete the simulation.

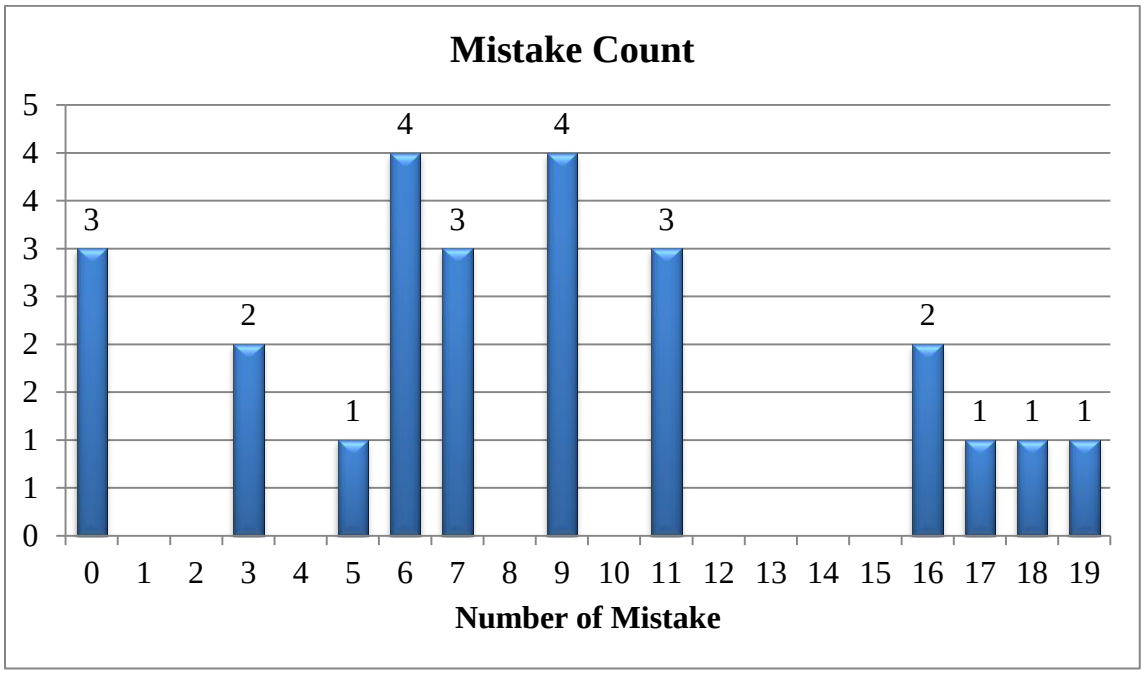


Figure 2.43: Mistake Count

Figure 2.43 shows the number of mistake count from the 25 Tester; majority is between 6 to 9 mistake counts during the simulation.

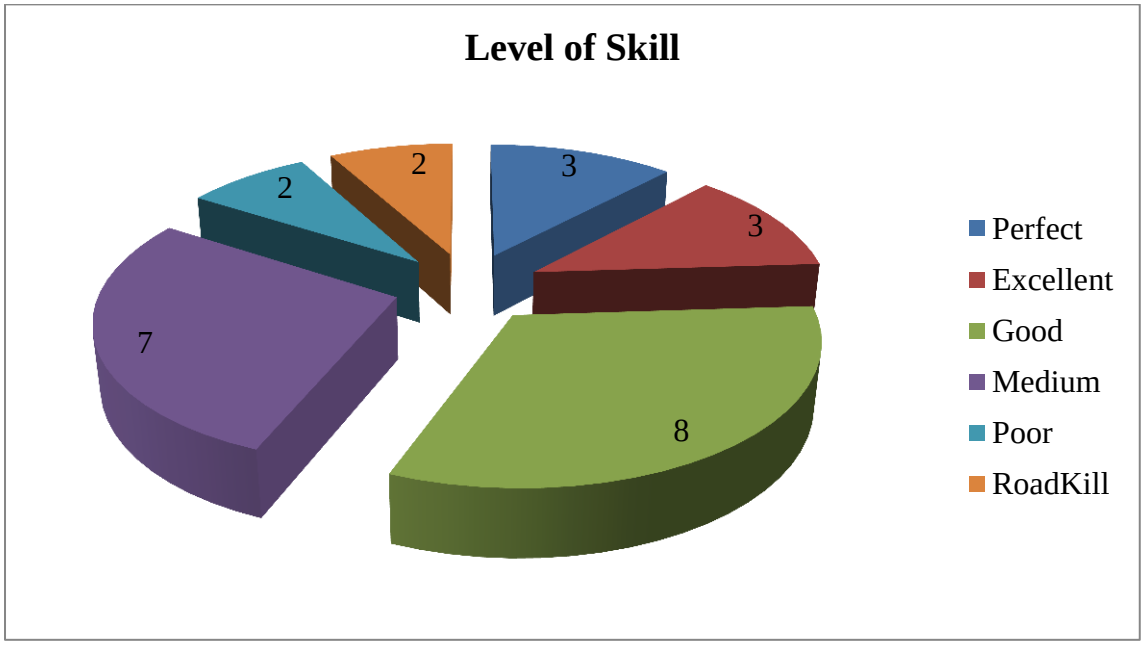


Figure 2.44: Level of Skill

Figure 2.44 shows the level of skill of the 25 Tester which categories are according the number of mistake counts. Majority is between the level's range of Good and Medium.

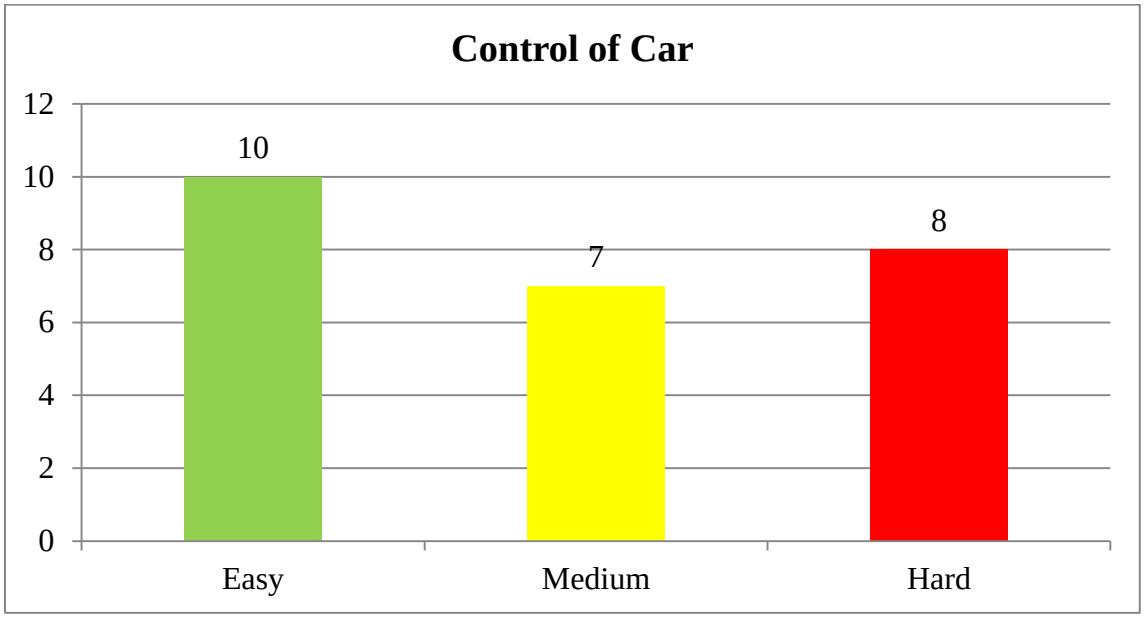


Figure 2.45: Control of Car

Figure 2.45 shows the response of the Tester for the rating of the control of the car inside the simulation. Majority comment that controlling the car inside the simulation is Easy but condition is requiring some time to handle it well.

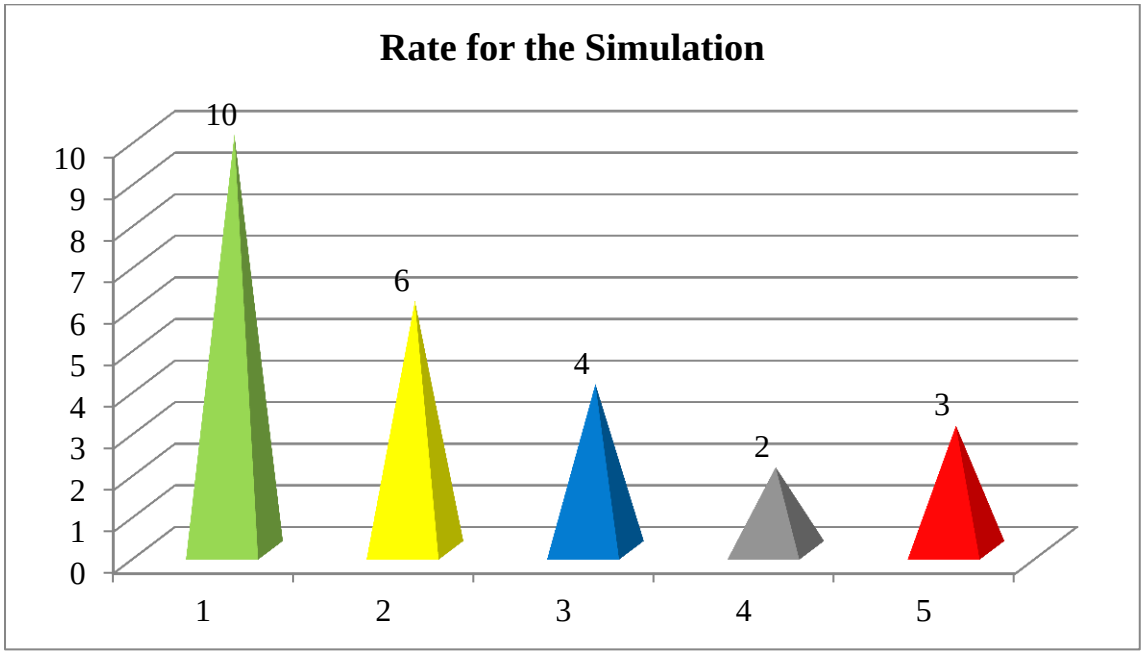


Figure 2.46: Rate for this Simulation

The Rating for the simulation is shown in figure 2.46 which majority is rating the simulation at 1.

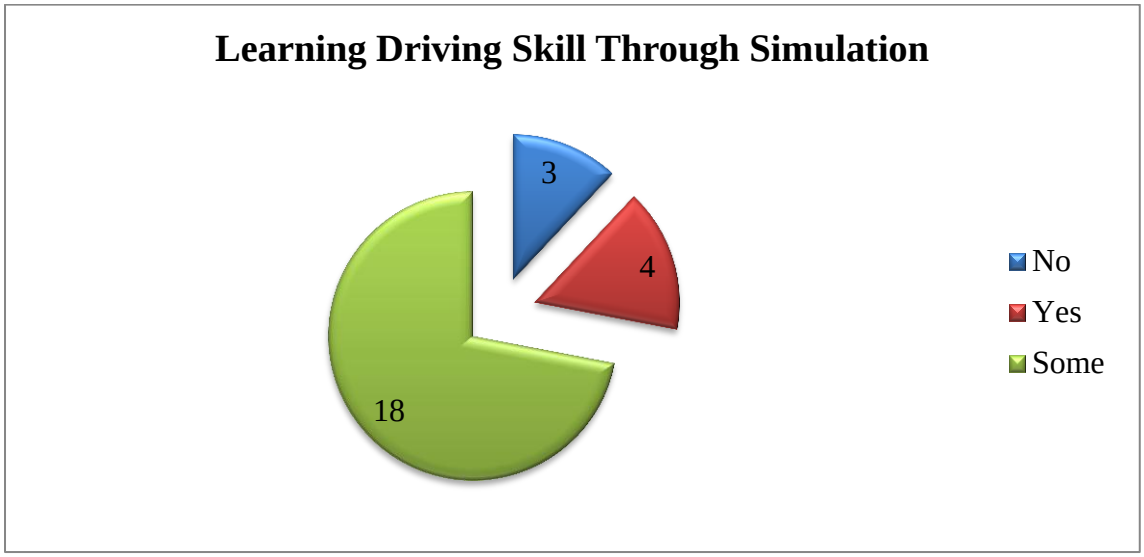


Figure 2.47: Learning through Simulation

Pic chart inside the figure 2.47 show the comment of learning driving skill through this simulation. Majority of Tester selecting learning "Some" for learning driving skill through this simulation.

At Second Try

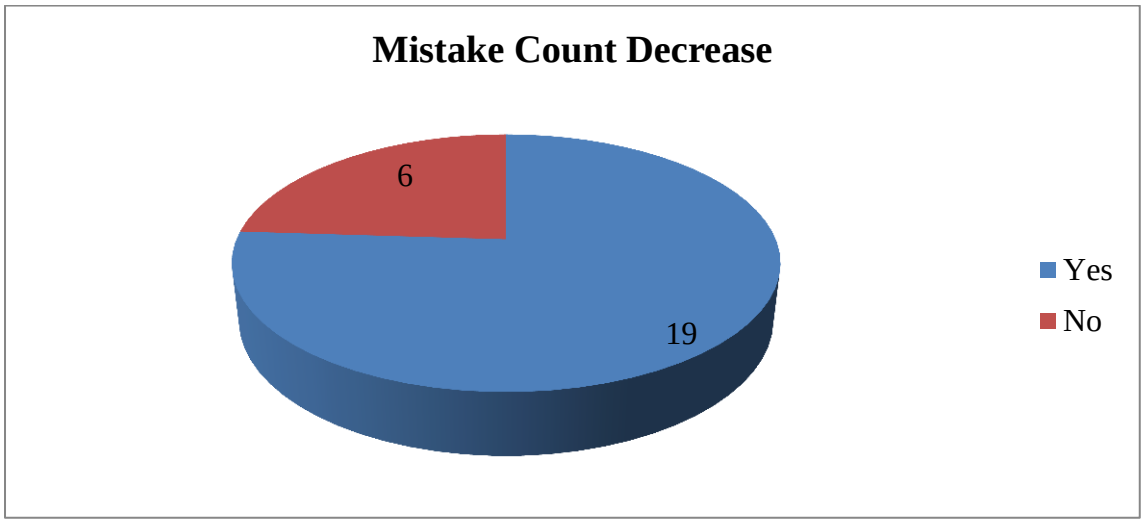


Figure 2.48: Mistake Count Decrease

Figure 2.48 shows the result of 25 Tester who success in 1st try then proceed to 2nd try to this simulation to check whether had improve their skill during 2nd try. Result of the mistake count during 2nd try was decrease of the 19 Tester which mean their skill had improve better than 1st try but still had 6 Tester didn't decrease their mistake count during 2nd try but increase or maintain the same as 1st try.

Fail in 1st Try

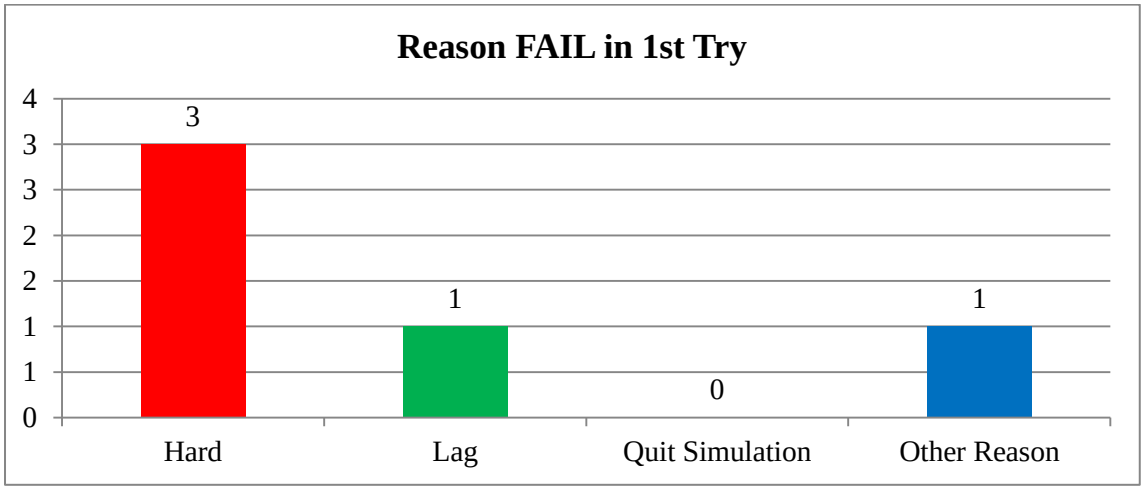


Figure 2.49: Reason Fail in 1st Try

Figure 2.49 show the reason of the 5 Tester who fail during the 1st try of go through the simulation. Majority response that the simulation is "Hard" for them and the rest of complaining the simulation was lag and other reason.

2.3.6 Maintenance

Maintenance generally is for the updating version of adding certain attribute, criteria, implement new feature, using new technology and other else. Maintenance may not be happen for this driving simulation system since it is generally focus on Malaysians resident. Update version if possible will be adding multiple selection of World region so user can learn other country driving pattern and their country traffic's rule. Further improvement maybe occur that are the graphic level, more specific's rule, using actual environment, car selection, and many else.

PART 3

CONCLUSION

Courseware Driving Simulation is learning software which provides a driving environment inside a computer platform for user to learn, train, improve and understand all things about drive.

Courseware Driving Simulation provides a convenient platform for user due to its portable part. Because user minimum only require a computer to installing the courseware then can operation the driving simulation inside their computer. Additional hardware can be added for example to getting more close to reality; user can acquire a steering handle as controller during the driving simulation.

Upon completion of the system, it will provide a lot of advantage for example multiple driving courses available inside the system: user can learn different driving basic knowledge through this part; performance analysis : analysis user performance inside the driving simulation then provide the performance result thus user can understand their driving weakness base on the result; implement local traffic rule: local traffic rule will implement inside the driving simulation thus user can understand about local traffic rule, this function more targeting on foreigner driver.

In a nutshell, this system won't be a perfect system if no user are using. Updating version will be launch if receiving user's feedback, additional feature add-on inside system, bug occur, traffic's rule update and many else. So enhancing for this system is possible in future.

REFERENCES

Bryan Reimer, Lisa A. D'ambrosio, Joseph F. Coughlin, and Michael E. Kafrissen and Joseph Biederman. 2006. Using self-reported data to assess the validity of driving simulation data. *Behavior Research Methods.* **38**(2):314-324

C.A. Holland.1993.Self-bias in older drivers' judgments of accident likelihood.*Accid. Anal. Prev*.25 (4):431–441

F. Bella.2008. Driving simulator for speed research on two-lane rural roads. *Accident Analysis and Prevention*.**40**(4):1078–1087

F.P. McKenna, M.S. Horswill, J.L. Alexander.2006. Does anticipation training affect drivers' risk taking?*Journal of Experimental Psychology*.Applied.**12** (1):1–10

H.C. Lee, D. Cameron, A.H. Lee.2004. Assessing the driving performance of older adult drivers: On-road versus simulated driving. *Accident Analysis and Prevention*.**35** (4):797–803

H. Klee, C. Bauer, E. Radwan, H. Al-Deek.1999.Preliminary validation of a driving simulator based on forward speed.*Transport. Res. Rec*.1689 (1):33–39

J.A. Groeger, I.D. Brown.1989.Assessing one's own and others' driving ability: the influences of sex, age and experience. *Accid. Anal. Prev*. **21** (2):155–168

J Gerontol .1994. Performance-Based Driving Evaluation of the Elderly Driver: Safety, Reliability, and Validity. *Journal of Gerontology*. **49** (4): M153-M159.

J.R. Anderson.1987.Methodologies for studying human knowledge. *Behav. Brain Sci*. **10**(1):467–505

J. Törnros.1998. Driving behaviour in a real and a simulated road tunnel: A validation study. *Accident Analysis and Prevention*.**30**(4):497–503

Kulhavy, R. W. 1997. Feedback in written instruction. *Review of Educational Research*.**47**(12)211-232.

Lisa Dorn and David Barker.2005. The effects of driver training on simulated driving performance. *Accident Analysis & Prevention*.**37**(1):63-69.

L. Lonero.2008.Trends in driver education and training.*American Journal of Preventive Medicine*, **35** (3):316–323

L. Nilsson.1993.Behavioural research in an advanced driving simulator-experiences of the VTI system. *Proceedings of the Human Factors and Ergonomics Society 37th Annual Meeting*:612–616

R.A. Marottoli, E.D. Richardson.1998.Confidence in, and self-rating of, driving ability among older drivers. *Accid. Anal. Prev*. **30** (3):331–336

S.T. Godley, T.J. Triggs, B. Fildes.2002.Driving simulator validation for speed research. *Accident Analysis and Prevention*.**34**(4):589–600

X. Yan, M. Abdel-Aty, E. Radwan, X. Wang, P. Chilakapati.2008. Validating a driving simulator using surrogate safety measures. *Accident Analysis and Prevention*.**40**(4):274–288

Appendices

Appendix A

Table below shows the sample of the questionnaire.

Questionnaire PSM1

Section A

Please fill in with relevant information

Gender:

Faculties:

Status:

Section B

Website for existing system: http://www.stisimdrive.com/

This system is develop for driving simulation which for driving training purpose. Evaluation this system is important for my new system. Please mark the appropriate column with a tick (√).

Description	Strongly disagree	Disagree	Neutral	Agree	Strongly agree
1. User has using this system before driving real car					
2. The system is affordable for user					
3. The system is easy to use for beginner user					
4. Users didn't feel troublesome for the system which requires supporting hardware to function for example driving steering.					
5. The system implements local traffic's rules inside the driving simulation.					

Section C

Please circle the appropriate answer

6. User are more prefer simple driving simulation rather than the complex driving simulation which require high feature computer to support it.

   Yes ☐

   No ☐

7. Users prefer cheap and low cost diving simulation compare to expensive driving simulation system.

   Yes ☐

   No ☐

8. Users would like to practice their driving skill in home rather than locate specific location for driving practice.

   Yes ☐

   No ☐

9. Normal computer equipment (monitor and keyboard) for control driving simulation are more convenient to use.

   Yes ☐

   No ☐

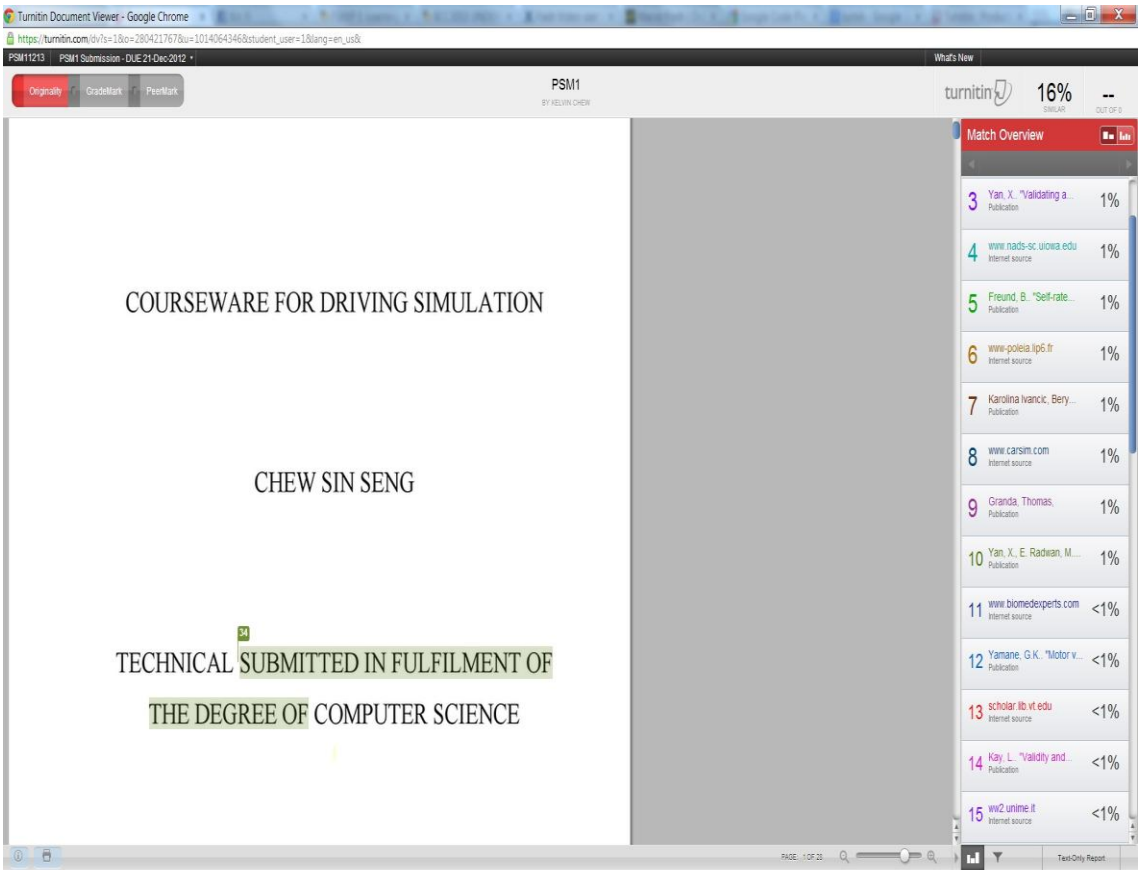
10. Implement local traffic's rules inside driving simulation are a benefit for users to understand more about local driving's style.

    Yes ☐

    No ☐

Appendix B

Plagiarism Checking Result: 16%



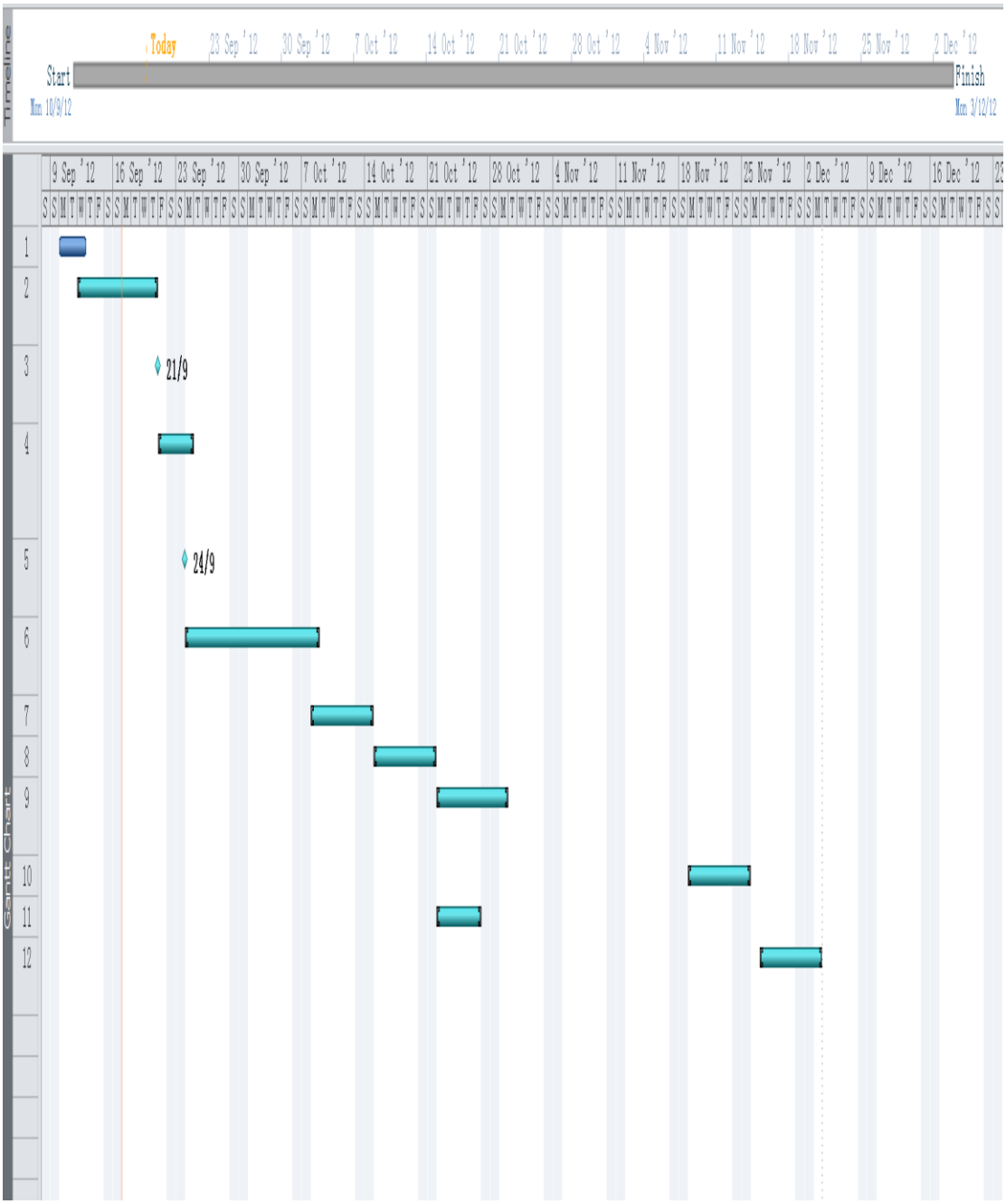
Report link:

Turnitin_Originality_Report_280421767.html

Appendix C

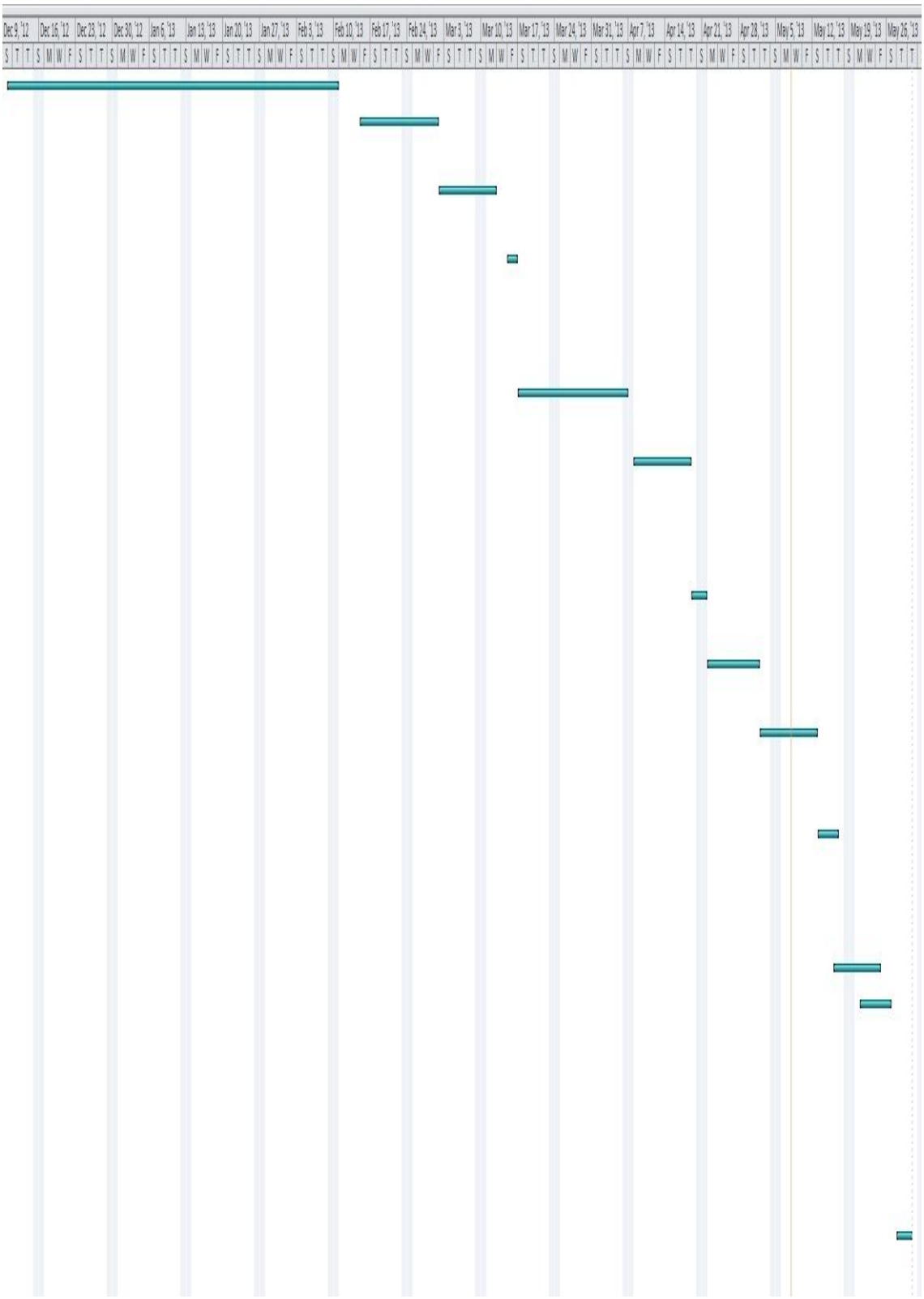
Gantt chart: PSM 1

CD10027 Chew Sin Seng



Duration	Task
10/09/2012-12/09/2012:	First meeting with SV Know the new SV because of changing from the previous selected
12/09/2012-20/09/2012:	PSM 1 mini proposal discussion with SV - Discuss about the detail, how and what to write inside the mini proposal. - Changing of the title maybe occur due to the changing of the SV
21/09/2012-21/09/2012:	Submission of the mini proposal
21/09/2012-24/09/2012:	Discussing with the SV about selected title - Discuss and getting know about the new title
24/09/2012-24/09/2012:	Submission for PSM purchasing requirement
08/10/2012-14/10/2012:	Chapter 1 for PSM 1
15/10/2012-21/10/2012:	Chapter 2 for PSM 1
22/10/2012-29/10/2012:	Finalise about the project title, area and type for us
22/10/2012-26/10/2012:	Chapter 3 for PSM 1
19/11/2012-25/11/20012:	Meeting with SV - Discussing about the progress of PSM 1
27/11/2012-03/12/2012:	Final Complete Report for PSM 1

Gantt chart: PSM 2



Duration	Task
10/12/2012-10/02/2013:	Study about Unity
15/02/2013-01/03/2013:	Design and input the algorithm into project
02/03/2013-12/03/2013:	Gather and prepare the material for the project
15/03/2013-16/03/2013:	Meeting with SV for discussing about the implementation phase and idea for the project
17/03/2013-06/04/2013:	Design the first prototype of the project
08/04/2013-18/04/2013:	Showing the prototype to SV and improve the problem of feedback from SV
19/04/2013-21/04/2013:	Meeting with SV again to get the clear idea to solve the problem at previous discussion
	Start correct the problem in development phase
22/04/2013-01/05/2013:	Adding the result interface for the project and showing to SV
02/05/2013-12/05/2013:	Doing documentation
	Showing the progress of documentation for SV to check
13/05/2013-16/05/2013:	Whole report submission for hardcover
	Fill in the form of SV signature
16/05/2013-24/05/2013:	Poster Printing
21/05/2013-25/05/2013:	Executive summary submit
	Demo Presentation to SV
	Hardcover submission sign by SV
	Presentation schedule arrangement
28/05/2013-30/05/2013:	PSM2 presentation