Project Report:

Game Title: Police Chase

The game I have implemented is a physics-based driving game, the player character consists of a police car model, the aim of this game is to ram opponent AI of into the boundaries of the playable game area within the 25 second time limit given. There are 8 AI characters rendered in game of 3 different types using 2 different models the aim of the game is too destroy as many enemies as possible while avoiding collisions with other AI characters.

The player can control their character by using the 1 and 2 keys, 1 allows the player to steer left and 2 allows the player to steer right. Ramming an opponent to the game boundaries will lead to an increase in score while ramming a brown car into the game boundaries will lead to a reduction of the players score by 5. Each collision with other cars in the game world will lead to a decrease in the players health unless they have picked up a powerup which makes the player invincible for a short period of time.

Assets Used:

Meshes:

* PEUGEOT 405 GLX - 517 Polygons - Game Ready, <https://www.turbosquid.com/FullPreview/Index.cfm/ID/1055619>, Nov 19 2019, Royalty Free License
* Snowmobile Free, <https://www.turbosquid.com/FullPreview/Index.cfm/ID/1036233>, Nov 10 2019, Royalty Free Licence
* NYPD Ford Mondeo 3D Model, <http://www.cadnav.com/3d-models/model-43219.html>, 20 Nov 2019, Free use licence
* Low-Poly Car, <https://free3d.com/3d-model/low-poly-car-40967.html>, 15 Nov 2019, Free use licence
* Low Poly Tree, <https://free3d.com/3d-model/low-poly-tree-73217.html>, 20th Oct2019, Free use licence
* health pack.blend, <https://www.turbosquid.com/FullPreview/Index.cfm/ID/514293>, 20 Oct 2019, Free use licence

Images:

* Roof image, <https://forum.rising-world.net/attachment/18657-texture-roof-lugher-texture-library-new-ceramic-roof-tiles-texture-of-ceramic-ro/>, Free use licence
* "Nature" texture pack, <https://opengameart.org/content/nature-texture-pack-256x256> Public domain licence
* Sky Box, <https://www.dropbox.com/sh/phslacd8v9i17wb/AABui_-C-yhKvZ1H2wb3NykIa?dl=0> Free use licence
* House image<https://www.pinterest.co.uk/pin/622974560931648071/> Free use licence
* Pickup image 1, <https://www.rgbstock.com/photo/p7lQBLm/Seamless+Glass+Tile+7> Free use licence
* Pickup image 2, <https://www.filterforge.com/filters/11319.jpg> Free use licence
* Office image <https://previews.123rf.com/images/pranodhm/pranodhm1612/pranodhm161200290/68820231-close-up-office-window-background-and-texture.jpg> Free use licence

Audio:

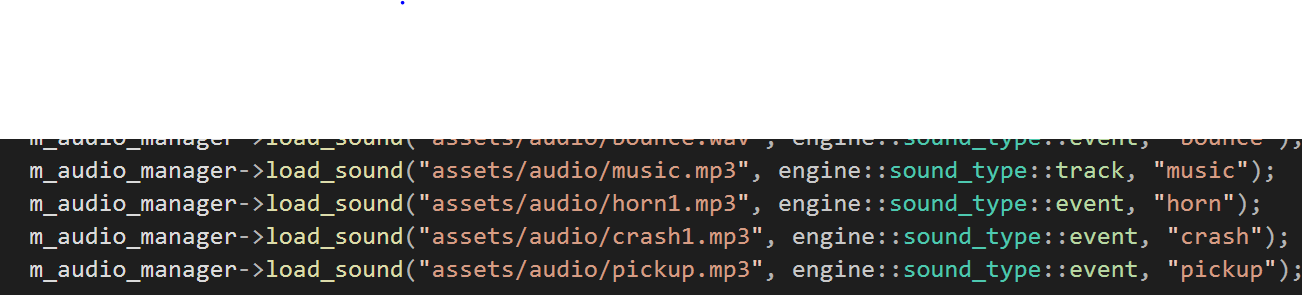
* BoxCat\_Games\_-\_05\_-\_Battle\_Boss, <https://freemusicarchive.org/music/BoxCat_Games>, Creative Commons Attribution License.
* Horn, <https://www.zapsplat.com/?s=car+horn&post_type=music&sound-effect-category-id=> Free use licence
* Crash Impact, <https://www.zapsplat.com/sound-effect-category/vehicle-impacts/> Free use licence
* Pickup, <https://www.zapsplat.com/?s=game+item&post_type=music&sound-effect-category-id=> Free use licence

Part 1:

I have created a final game level intro screen, this intro screen includes the name of the game, the simple keyboard controls for the player, the objectives and mechanics of the game and some description of the 2 pickup items that can be used in the game. This intro screen has been implemented using crossfade and the text renderer to draw text onto the screen before the start of the game. The player can dismiss the intro screen and start the game by pressing the spacebar. When the game ends a black screen is shown with the players final score, the text is rendered on top of a crossfade image rendered with negative z coordinates to allow text to be rendered above it.

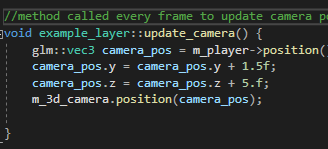
I have created some office building along the perimeter of the playable game area, these have been created with the cuboid included in the template. I have also added 2 houses to the perimeter of the game area using the cuboid shape and the tetrahedrons as roofs. All primitives have been textured correctly and have been placed in the game world using SORT. I have also create a transparent sphere with alpha effects, this is used with one of the power up items and is rendered when it is picked up by the player, the sphere is set to the position of the player every frame making it follow the player during gameplay. Skybox and terrain textures where changed in milestone 1. Houses are added to the background using cuboids and tetrahedrons, press w to remove fog to make these more visible.

I have included various events sounds in this game, this includes as horn synchronized to the player colliding with an AI this is done by playing the sound when the player is colliding with more than 1 object, a crashing sound synchronized with the AI colliding with the boundaries of the playable game area, this is implemented by playing the sound when a AI is detected to be crashed on the game area boundary. A sound has been added that are synchronized to play upon the player picking up one of the pickup items in the game. The background music has also been changed and is played when the player dismissing the intro screen and starts the game



Part 2:

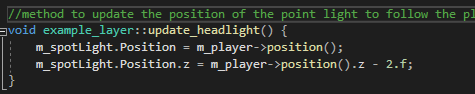
I have implemented the camera to follow behind the player model, the camera is set at a set position relative to the player and is update every frame following the players movement. This has been changed from an implementation of an on rails camera in the previous submission



This method is called every frame and updates a vector called camera\_pos, this is set to the position of the player and then moved upward and backwards to be behind the player, once the vector is set with the needed co ordinates the cameras position is set to the camer\_pos vector.

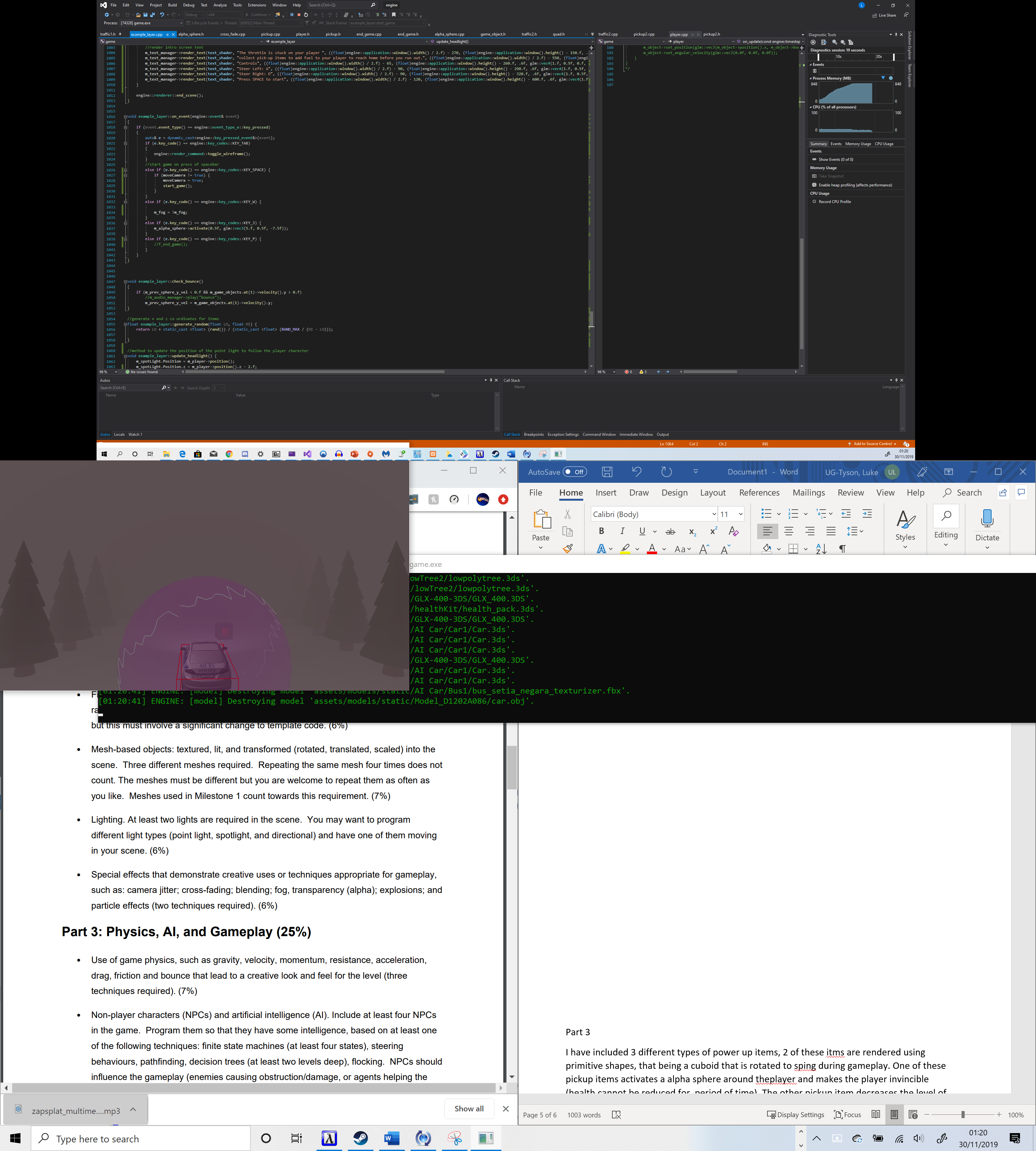
Various mesh-based objects are used in this game, I have included various mesh based objects in this game of various vehicles. I have rendered 3 different models of cars in this game one is used for the player and can be controlled using the described keyboard controls and the others 2 are used as for AI. There is also 1 model of a bus that is also used as AI. I have added a model of a tree to the game and this is rendered using a lop to set the boundaries of the playable game area. All meshes are rated and scaled appropriately using SORT.

I have dimmed the original directional light and have added a point light, the point light is set to the front of the player model and its position is updated every frame, this allows it to act as headlights for the player model as it moves with the player as the player moves around the game world.

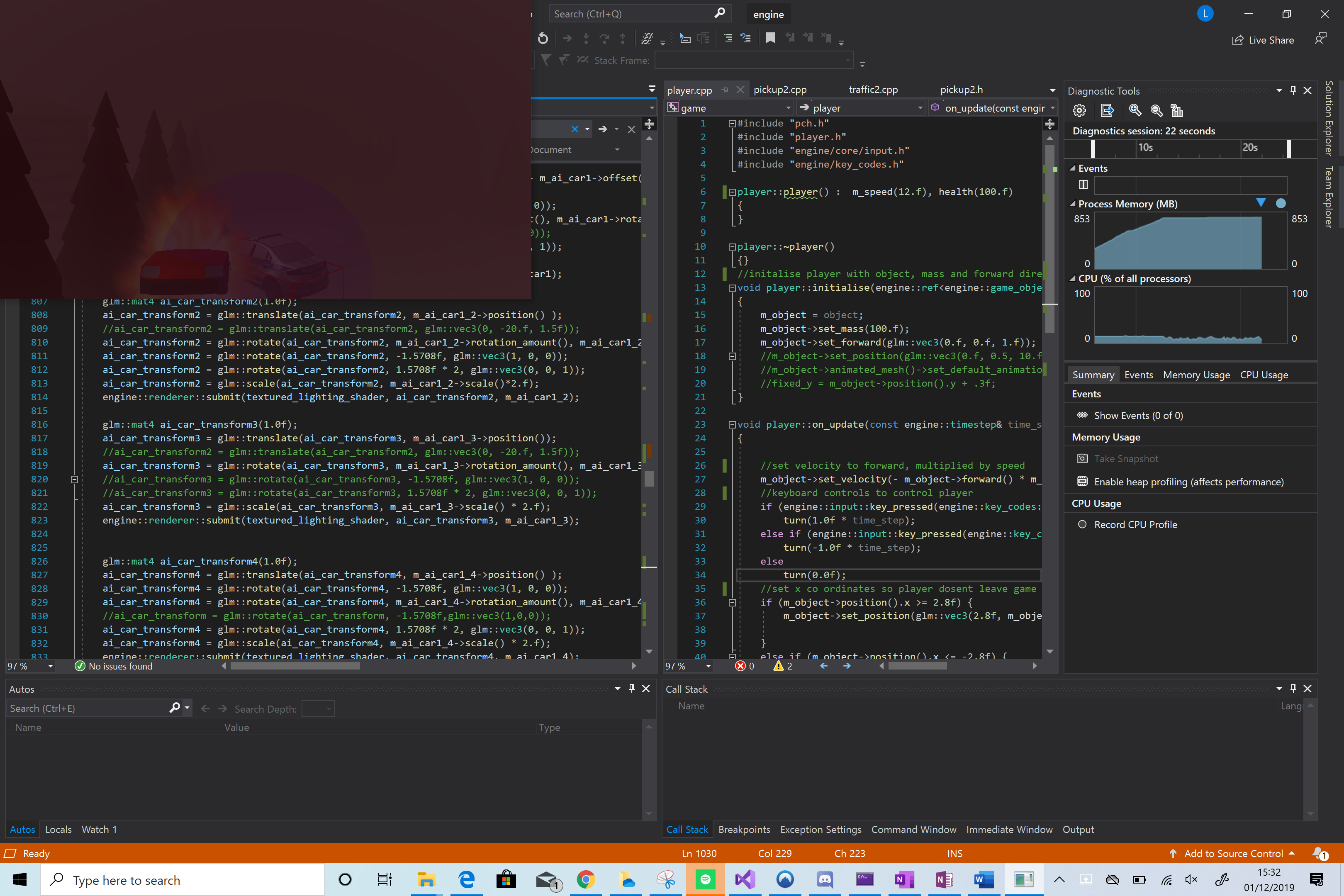


The light is set to the player position, after this the z co ordinate is reduced by 2, this allows the spotlight to be at the front of the player character and act as headlights, this method is called every frame to keep the light in the correct position relative to player movement.

I have made use of fog in this game to add difficulty to the game, I have also added alpha effects in the form of alpha spheres that surround the player when a certain pickup item is collided with by the player. The alpha sphere is set to the position of the player every frame and when the player picks up the item it is rendered by calling the render\_sphere method



I have also used and explosion effect which is played when an AI character reaches the perimeter of the playable game area, this effect is played at the location of the AI when the hit the perimeter of the game area. This uses the billboard effect from the FX examples.



The above image demonstrates the use of crossfade and billboard effects

Part 3

Gravity is applied to all characters in game by use of bullet physics.

The AI characters and player are moved forward by applying a constant velocity to each model forwards (forwards is set to move negatively along the z axis). The player character can be turned left and right by applying an angular velocity to the players character.

Friction is set to the terrain to allow the characters to move smoothly along the terrain surface, the level of friction is set to 0, during development it was seen that setting friction levels higher or lower led to unexpected movements of the AI characters.

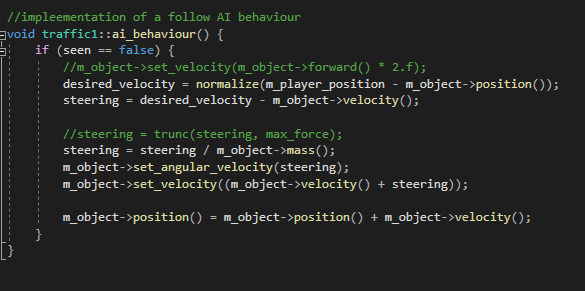
I have included 3 different types of power up items, 2 of these items are rendered using primitive shapes, that being a cuboid that is rotated to spring during gameplay. One of these pickup items activates a alpha sphere around the player and makes the player invincible (health cannot be reduced for a period of time).

The other pickup item increases the intensity of the headlights making it easier for the player to see through eh fog in the distance.

The third pick up is rendered using a model, this item restores some health to the player.

All power up items are rendered to rotate every frame making them spin in game.

2 different types of AI are implemented in this game, one is implemented using steering behaviours. A seeking algorithm has been implemented for this AI that allows the character to move towards the player. This algorithm was implemented with research from the following site: <https://gamedevelopment.tutsplus.com/tutorials/understanding-steering-behaviors-seek--gamedev-849>



The other AI implemented in this game makes use of a finite state machines consisting of various states shown below.

A picture containing electronics

Description automatically generated

Power ups have been implemented in this game.

2 power ups have been implemented rendered as small cuboid shapes in the game area. One makes the player not lose damage on collision with AI characters and renders an alpha sphere around the player. The other increase the viewable distance by increases the intensity of the spotlight in front of the player character.

1 power up has been rendered using a mesh downloaded from the internet, when this item is collected the players health is increased.

A timer of 25 seconds has been implemented, upon the timer ending the game is ended. The timer is reduced by 1 every 60 frames.

The player can press W to switch between hard and easy difficulty.

Conclusion:

The game level I have implemented makes good use of various game development concepts such as AI implemented with steering algorithms, a high number of AI have been added into the game world and various special effects have been use with a explosion effect added to AI characters when they crash into the perimeter of the game area this billboard effect is combined with a crossfade effect , fog effects are also implemented in the game. The project also implements power up items and timers. However I have had problems implementing the physics for this game, one of the many problems I have experienced during development has been car floating is AI cars colliding with the terrain and moving towards the edge of the game area, through various changes I have not been able to solve this problem in the time given. I have also found it hard to find working models, finding working models made up most of my time spent on this project as the vast majority of the models a tried to render in the engine either did not render correctly or where missing textures, this affected my ability to implement and test game code.

To expand this project into a full game this project will need its physics problems mentioned above fixed. The game level would need more varied and sophisticated AI for example AI who may attempt to ram the player vehicle or flee from the player vehicle, more game objects to be added into the scenery and a number of different levels rather than just 1. More detailed models will be needed and improvements to the objects in the background scenery will also help in turning this game level into a full game.

*Note, after further investigation the physics problems in the game seem to be related to skipped frames, poor game performance seems to adversely effect game physics*