**External Programs**

Although we will be using Unreal Engine 4 (v4.23.1, specifically) as the main development tool for the game, most assets: textures, models, animations and music, will be developed using external programs. A list of each is below:

**Photoshop:**

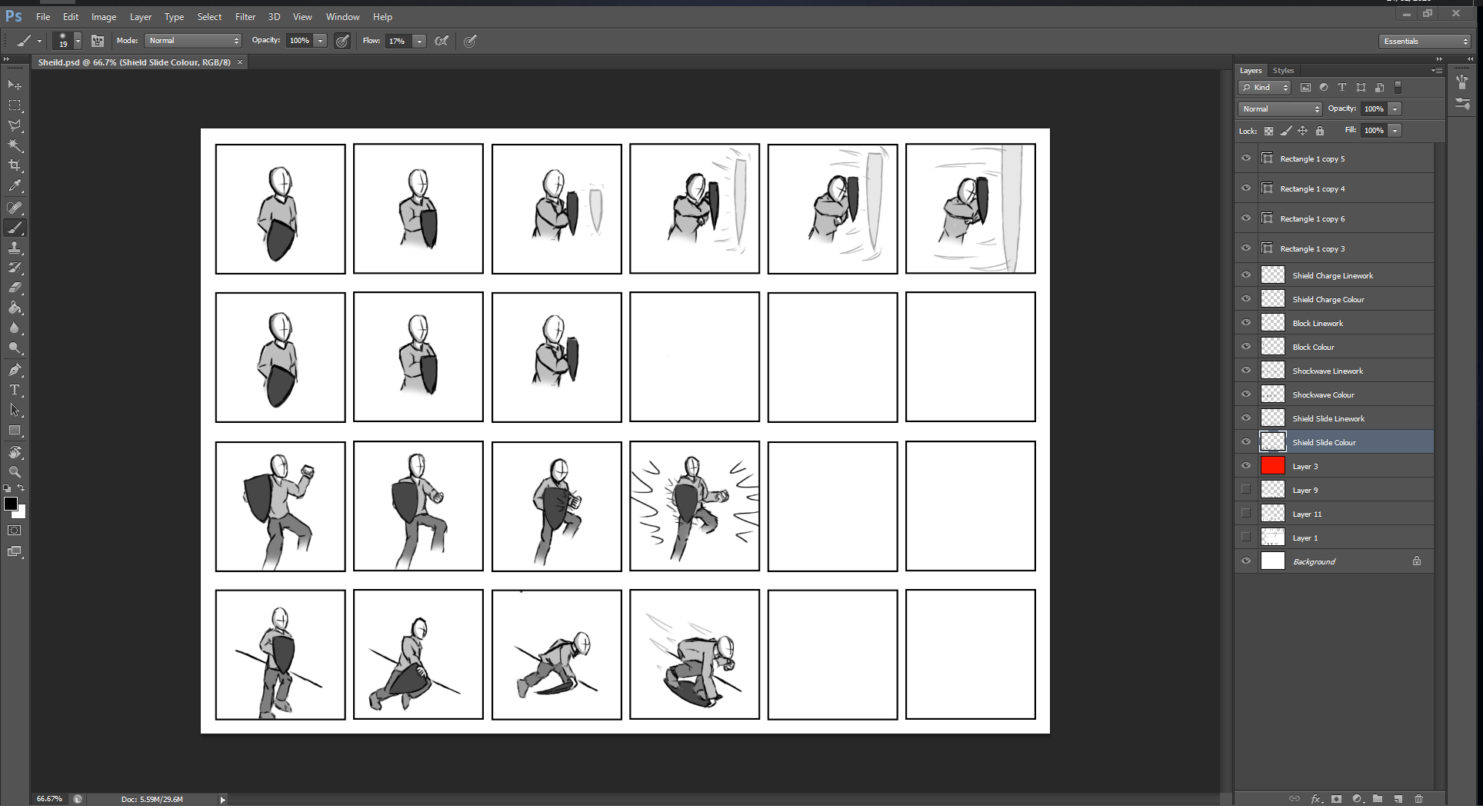
****

Figure 1 - Shield animation storyboard, drawn using Photoshop

Photoshop is an Adobe licensed image editing tool. We will be using this to create textures and other 2D assets, as well as using it in the design phase for its built-in animation timeline.

**Paint.net:**

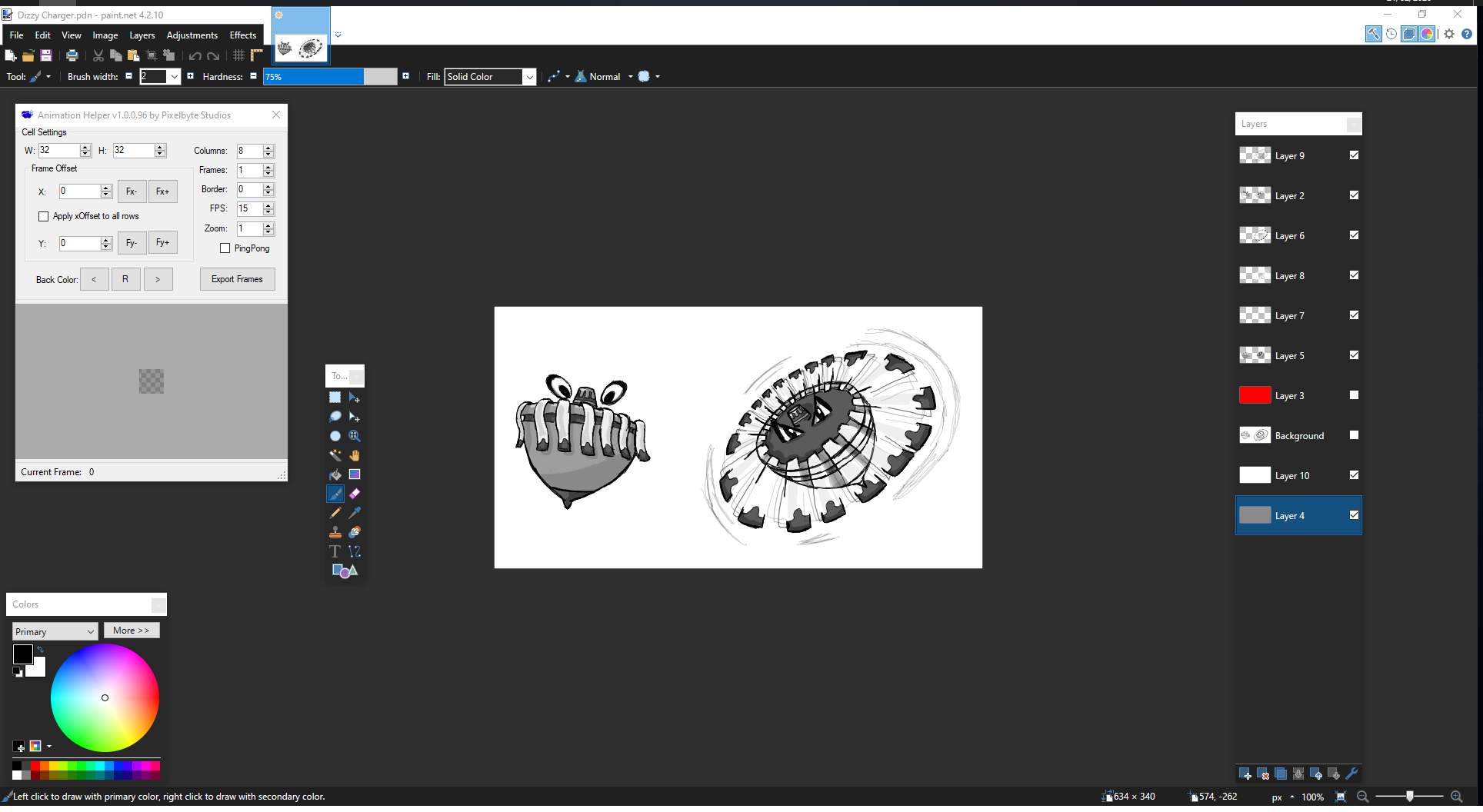
****

Figure 2 - Dizzy Charger enemy design, drawn using photoshop

Paint.net is an open source image editing tool, we will be using this, and a few community created plugins, for developing textures and animations.

**Audacity:**

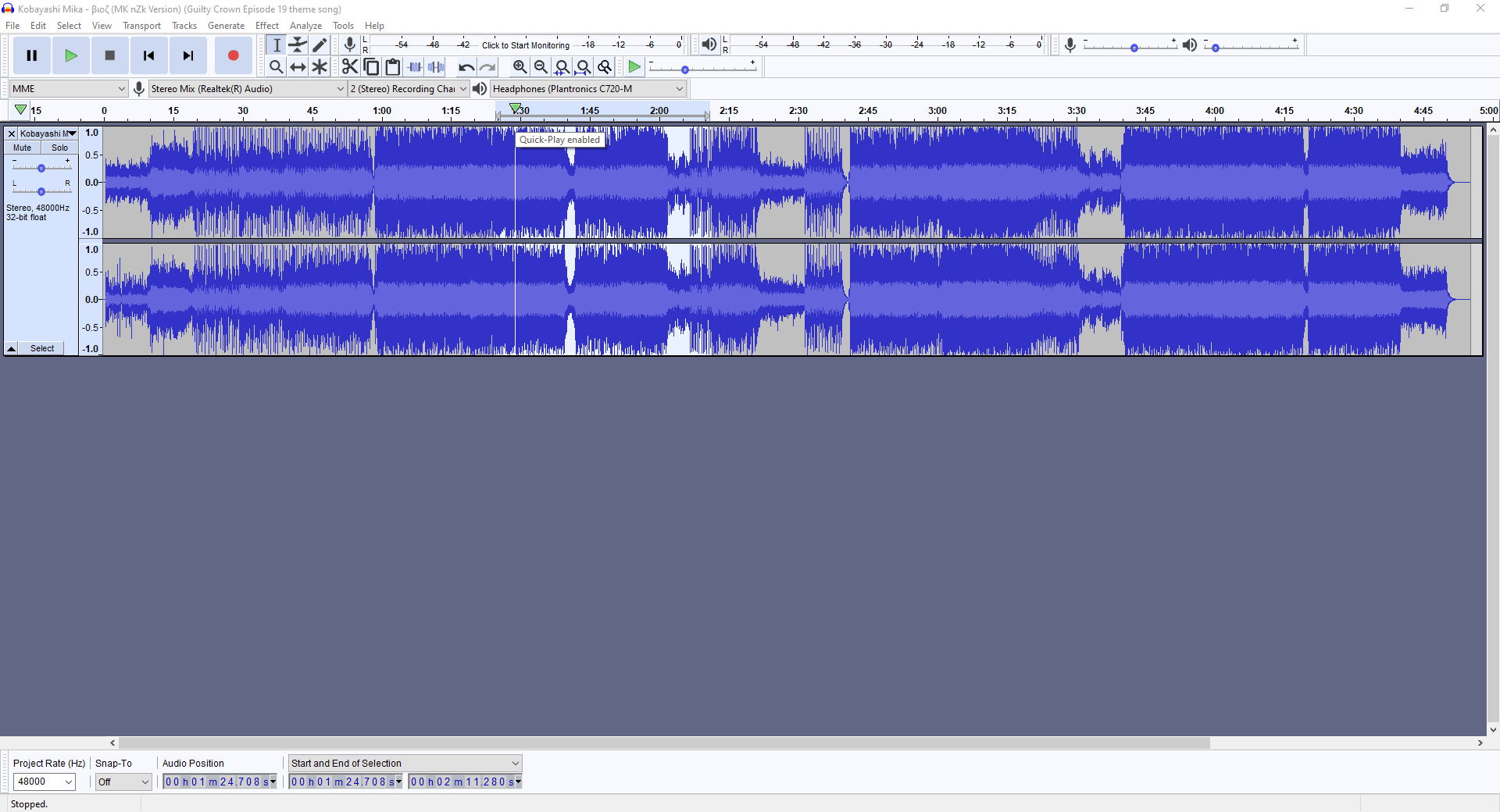
****

Figure 3 - Music file, edited using Audacity

Audacity is a simple audio-clip editing software, allowing the team to record and alter the audio effects we will be using in the game, while also allowing us to import music to convert and export it into a usable audio loop (Which may be done using the format of OGG Vorbis).

**Blender:**

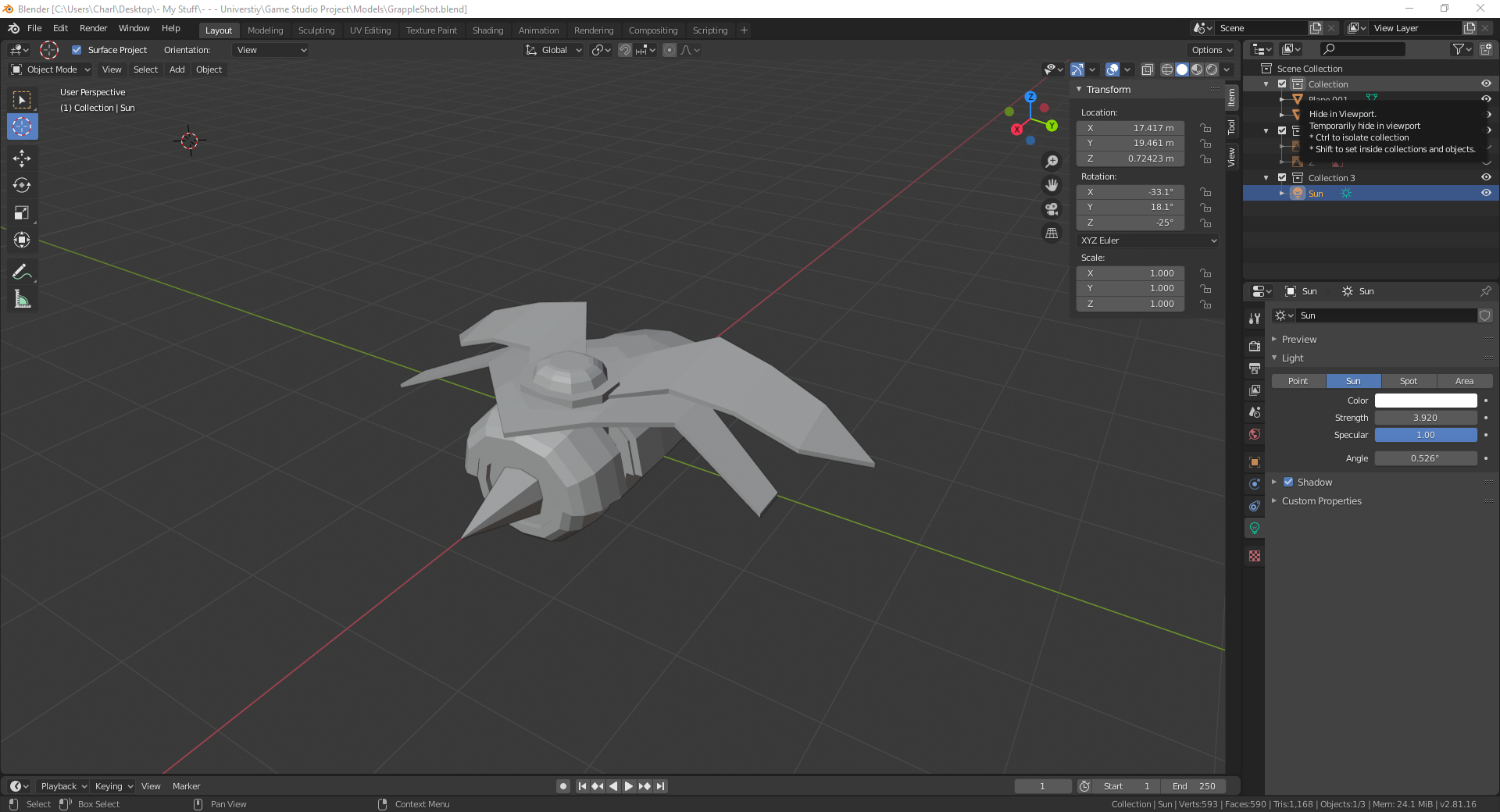


Figure 4 - Grapple-shot 3D model, created using Blender

Blender is an open-source modelling, sculpting, animating, texture painting and material creation software. Those on the team who have not used Maya before will be using this to create models and animations for the game.

**Maya:**

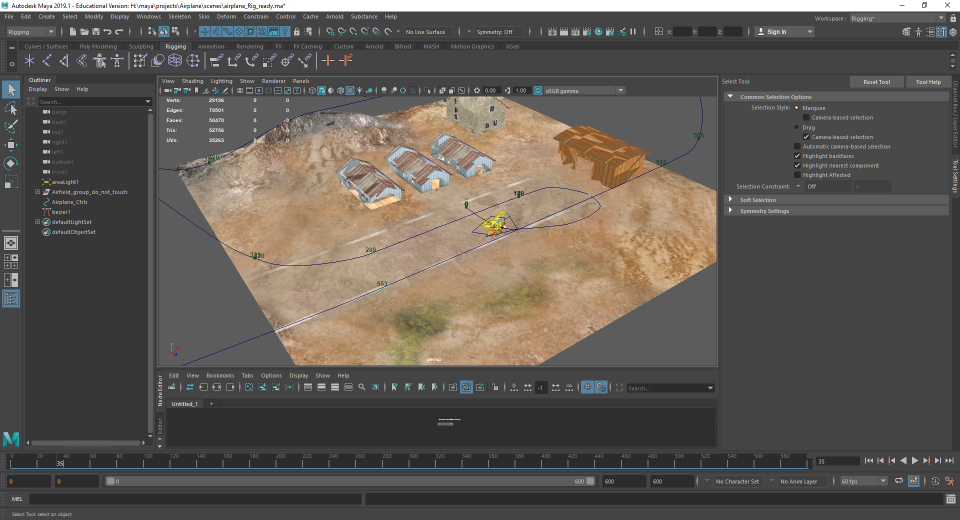


Figure 5 – Scene used to demonstrate animations in Maya

Maya is a closed-source modelling, animating, and texturing software. Those on the team who have used Maya before will be using this to create models and animations for the game. As Maya only has a student version for free, most Maya usage will be from university computers.

**Visual Studio:**

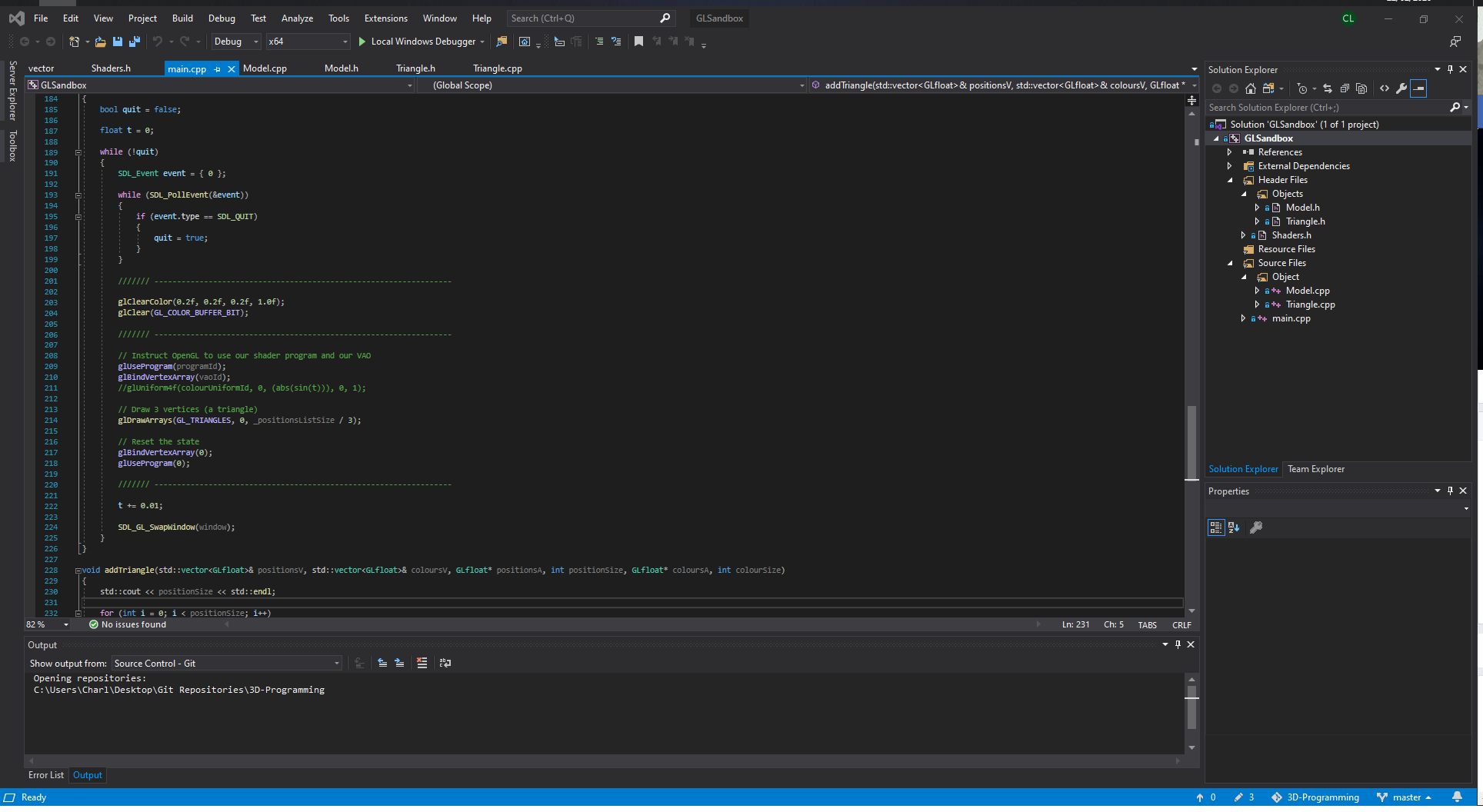


Figure 5 - C++ code, written using Visual Studio

Visual studio is a C++ editor that has integrated support with Unreal Engine 4. The programmers on the team will be using this to program advanced features into the game.

**Substance Designer:**

**A screenshot of a computer screen

Description automatically generated**

Figure 6 – Material Graph example made in Substance Designer

Substance Designer is a PBR material creator that will be used by the team’s designers. It allows for procedural workflows to generate materials that can be used in UE4 and can be imported into Substance Painter and Alchemist, to be used to a greater effect and in tandem with other materials.

**Substance Painter:**

A screen shot of a computer

Description automatically generated

Figure 7 – Matt with a wood material applied

Substance Painter is a PBR procedural material applier that can be used to texture meshes to suit the team’s needs. It utilises the maps that make up PBR materials. The designers on the team will use this program to generate custom materials that can be exported as a packed texture. This means that the Metallic, Roughness and Ambient Occlusion maps will be combined and can be filtered out using an RGB mask.

**zBrush:**

A screenshot of a computer screen

Description automatically generated

Figure 8 – Example zBrush project

zBrush is a digital sculpting application that can be used to make high poly models that can be baked into the low poly models in Substance Painter. This means that the designers can put incredible amounts of detail into the meshes and have little impact on the performance of the program.

**Adobe Illustrator:**

**A screenshot of a cell phone

Description automatically generated**

Figure 9 – Example Adobe Illustrator File

Adobe Illustrator is a 2D vector image creator. We will be using it to create the basic UI elements. It can export these as PNG images which can be easily imported into UE4.