|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task** | **Allocated completion time** | **Actual completion time** | **Allocated to (the name of the group member)** | **Comments** |
| Documentation. | 3 hours | 6 hours | Jack Matters  Michael Bell  Cordell Smith | All documentation required (cover sheet, group declaration, design and software documentation, etc). |
| Implement Bullet Physics API into project. | 30 minutes | 1 hour | Jack Matters | Had trouble with CMake, but got it working eventually. |
| Create physics world to handle all collision detection. | 1 hour | 1 hour | Jack Matters | Fairly simple, just followed a guide. |
| Create static and dynamic rigid bodies. | 1 hour | 30 minutes | Jack Matters | Fairly simple, just followed a guide. |
| Test that collision detection works. | 30 minutes | 1 hour | Jack Matters | Took longer than expected as I had trouble drawing shapes to the screen so I could visually see collision was working. |
| Add collision box to the camera. | 1 hour | 3 hours | Jack Matters | This took a lot longer than expected. Adding a force to the camera rigid body, then moving the camera to the resulting position was difficult. Got it ‘mostly’ working. |
| Refactor code to read all game object data from data structure, to create all rigid bodies. | 30 minutes | 1 hour | Jack Matters | Not hard, just long time to write code. Got it all set up, but without having the data structure implemented, the code can’t be implemented. |
| Create heightfield terrain shape from heightfield file. | 1 hour | 2 hours | Jack Matters | Rather difficult. Eventually got the data being read in and creating some kind of terrain rigid body, but without having a visible terrain, it is hard to see if it is working properly. |
| Terrain generation & temporary texture loader | 2 hours | 3 hours | Michael Bell | Difficult, Attempted to implement a terrain loader to the project and required textures on it to verify it was working correctly. Could not get any textures to display within our project. |
| Lua script reader | 1 hour | 1 hour | Michael Bell | Implemented a basic lua reader |
| Gameworld class setup | 30 minutes | 30 minutes | Michael Bell | Initial setup for game world objects |
| Creating skeleton program to work from | 1 hour | 2 hours | Cordell Smith | Creating a skeleton program to work from. This includes implementation of GLFW windows and creating a working OpenGL context to begin implementing our ideas. |
| Splitting up components to delegate | 30 min | 30 min | Cordell Smith | Planning of which components needed to be completed and who was going to do which part. |
| Create class UML | 1 hour | 2 hours | Cordell Smith | Made a few changes multiple times before the UML was correct. |
| Research shaders | n/a | n/a | Cordell Smith | Research about shaders to be able to implement them in our 3D engine. |
| Implement shaders | n/a |  | Cordell Smith | Began implementing basic shader useage into our current working project so that our program didn’t need to make hard coded glDraw() calls and rather used the more efficient method of shaders vertex array objects and vertex buffer objects. |
| Change file structure | 1 hour | 1 hour | Cordell Smith | Changing the file structure of the program to maintain a consistent system. This included changing class names to help understand the purpose of the class. |
| Code refactoring | 3 hours | 3 hours | Cordell Smith | Changing parts of the code to maintain a consistent coding syntax. |
| Create a simple floor to display that the shaders work | 3 hours | 3 hours | Cordell Smith | Using the skeleton class setup I was able to rotate the simple coloured panel to make a floor, then just looped through to change their position and rendered again. |
| Research into possible file importers |  |  | Cordell Smith | I researched possible libraries that could be used to import many file types. I decided on Assimp Import Library and continued to research about how it handles loading scenes and .obj files which we intended to use in our game engine. |
| Using assimp to load model data |  |  | Cordell Smith | After implementing assimp import library, I followed tutorials to import the .obj file type by taking a file path. The relevant information was then stored in a vector data structure for further use. |
| Created the GameAssetFactory class | 1 hour | 2 hours | Cordell Smith | Created this class to fulfill the software design pattern requirements to separate the creation of different game assets without knowing the type. |
| Created the IGameObject interface class | 1 hour | 2 hours | Cordell Smith |  |
| Created object classes to inherit from the IGameObject interface class | 2 hours | 3 hours | Cordell Smith |  |
| Plan 3D engine |  |  | Cordell Smith |  |
| Plan resource loading |  |  | Cordell Smith |  |
| Doxygen Commenting | 4 hours | 5 hours | Cordell Smith | Going through each header file and adding correctly formatted doxygen comments to each class function. |
| Code cleanup | 3 hours | 2 hours | Cordell Smith | Removing any unneeded lines of code as well as smaller things such as code syntax and line indenting. |
| Debugging | n/a |  | Cordell Smith | I have spent many unknown hours debugging the program during the tasks listed above. |
| Source control code reviewing | n/a |  | Cordell Smith | Reviewing other group members code before merging with the master branch. |
| Organise meeting times | 5 minutes | 5 minutes | Cordell Smith | Remind team about meeting times and organising where to meet. |
| Meeting agenda and notes | 10 minutes | 10 minutes | Cordell Smith | This was simply having points to work on at each meeting. What we needed to work on and direct how the meeting will be structured. During each meeting I would also take notes and take a photo and post to the slack channel for future reference. |