Ben Robinson

R020140g@student.staffs.ac.uk

Group Game Development Project and Work Based Simulation

Assessment

Contents

[Game Engine Analysis 2](#_Toc42679094)

[1. Game Engine Overview 2](#_Toc42679095)

[1.1. Approach taken 2](#_Toc42679096)

[1.2. Pros and Cons 3](#_Toc42679097)

[1.3. Text based tile demo 3](#_Toc42679098)

[1.4. Git Hub and how it assisted in the project 3](#_Toc42679099)

[1.5. Conclusion 4](#_Toc42679100)

[Solo Game Analysis 5](#_Toc42679101)

[Appendices 6](#_Toc42679102)

[Appendix 1: My Repo 6](#_Toc42679103)

[Appendix 2: Game Engine Repo (My branch) 7](#_Toc42679104)

[Appendix 3: Examples 7](#_Toc42679105)

[Appendix 3.1: CSV’s in Excel 7](#_Toc42679106)

[Appendix 3.2: CSV’s in Text 8](#_Toc42679107)

[Appendix 4: Failure to read commas in excel 9](#_Toc42679108)

[Appendix 5: Code Snippets 9](#_Toc42679109)

[Appendix 5.1: Automatic integer reading 9](#_Toc42679110)

# Game Engine Analysis

## Game Engine Overview

The game engine was made using OpenGL C++ with Simple DirectMedia Layer (SDL) and ImGui to assist certain aspects of implementation. However many of these add-ons were unnecessary in my implementations for the engine

## Approach taken

* + 1. To create the engine, the primary goal was to create a set of Comma Separated Values (CSV) readers to allow developers to customise data, such as dialogue, enemies, players, and attacks that the players could learn. CSV is a data structure commonly used within the software industry to move information between programs. However, they are also used in the games industry to easily customise important data within the game.
    2. This enabled the data within the engine to be customised, either through using an Excel spreadsheet (See Appendix: 3.1) or notepad (See Appendix: 3.2).
    3. The CSV reader will accept two types of variable: Integer and String. Integers can be read into the file with no issues Only requiring the number to be read in between two commas as a String, then having the read in value turned into an Integer using the “stoi()” function from the std library. Strings however are far more difficult to read into the program. Some needing punctuation marks such as commas or speech marks making them unable to use normally. This is because both are necessary for reading CSV files. To allow these punctuation marks strings within the CSV require it to start with the less-than symbol and end with a more-than symbol, although due to this these symbols are not allowed to be used within the CSV outside of this purpose. Some CSV files have smaller data structures within them, having an underscore with a number relating them to the specific structure (example variable\_0). With this a for loop is used to allow the csv to read in the data and store it in a separate data structure to be stored in an array within the main data. In the case where there is less data than what is needed, symbol <n/a> can be used instead to cut the data off at this point and move on to the next set (in the case where this is used at the start of the new structure a default structure will be given instead).

## Pros and Cons

* + 1. The CSV format was chosen as opposed to over file formats such as the JSON format was due to the overall simplicity of the data being used within the program and the ease of access and implementation of the format. JSON files are very powerful when needing to read in larger and more complex data with differing data types, however due to all the area’s it could be used for only needing Strings and Integers this becomes unnecessary. On top of this, implementing a JSON file to a project requires the use of a JSON loader and editor for use in a project while CSV only needs Excel or Notepad.
    2. One issue with CSV is that some versions of Excel read them into the system by using semi-colons instead of comma’s (see Appendix 4) which can cause the CSV to be unable to load into the program. To fix this the prefix “sep=,” can be added to the top of the CSV in word to allow it to be read in Excel without any issues, however saving the document will return it to colons.

## Text based tile demo

* + 1. To assist Ben Dixon with work on the tile based movement for the game engine, I created a basic grid loader which creates a set of tiles and reads out their details including whether the player can move on the tile and if so whether the player is standing on it. This was done by creating a two by two vector array of Tiles, which contains the tiles location within the grid, and what type of tile is being stepped on. From here the player can move around the grid by inputting either the W, A, S or D key. If the player tries to move outside the range of the grid or onto a tile which cannot be moved to, the player will instead stay on the tile it currently occupies. This program was very basic and used more to show how grid movement can be done rather than to be implemented into the engine. However, aspects of this prototype did end up in the final build of the engine.

## Git Hub and how it assisted in the project

* + 1. For the creation of the engine a Git Hub organisation was created with five repositories. One of each member of the team (including myself) and one for the game engine which we all worked together to create. Work was primarily done from Ben-Rs-repo and the feature/file\_io branch within the Engine repository. The organisation helped with development of the engine due to allowing new work to be added to the engine without any fears of overriding the work of other members. If newer work would be needed to be added to the engine, a pull request can be made within the repository where it can be reviewed by all the members before being added to the engine.

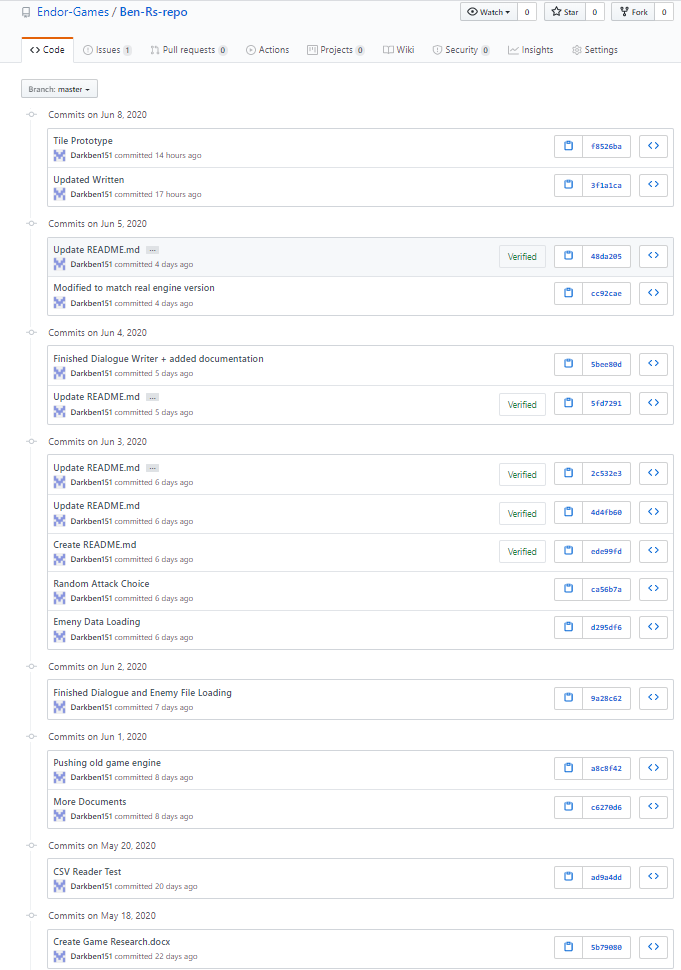
## Conclusion

* + 1. Overall

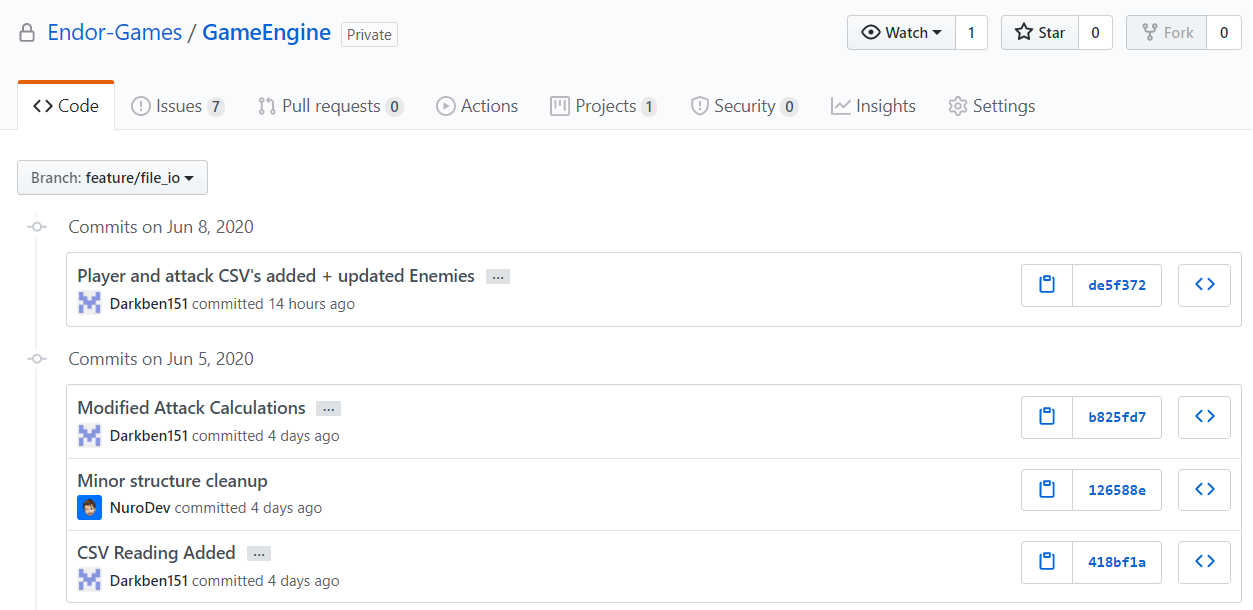
# Solo Game Analysis

# Appendices

## Appendix 1: My Repo

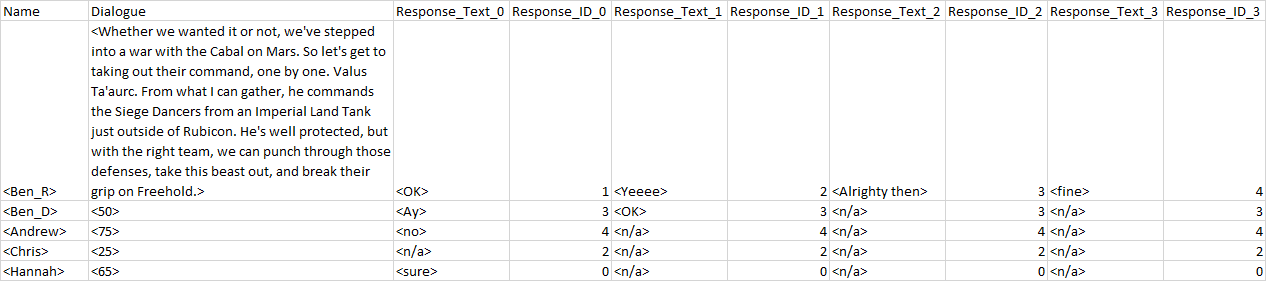


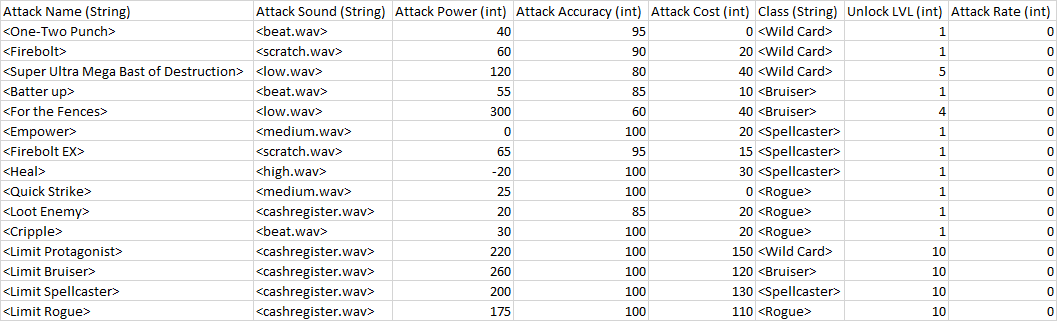
## Appendix 2: Game Engine Repo (My branch)



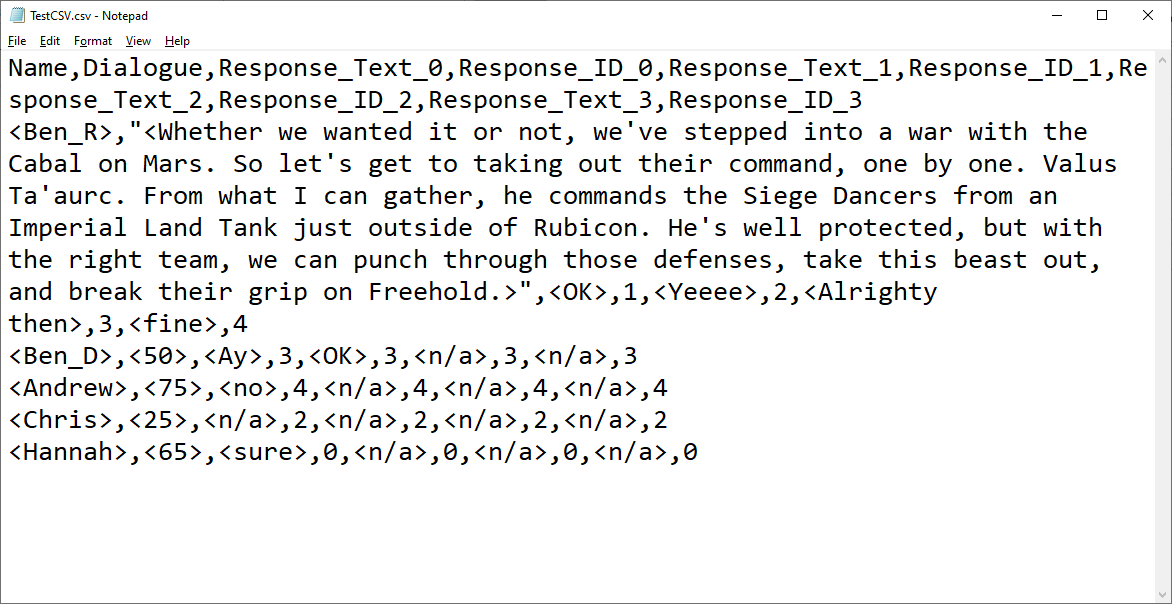
## Appendix 3: Examples

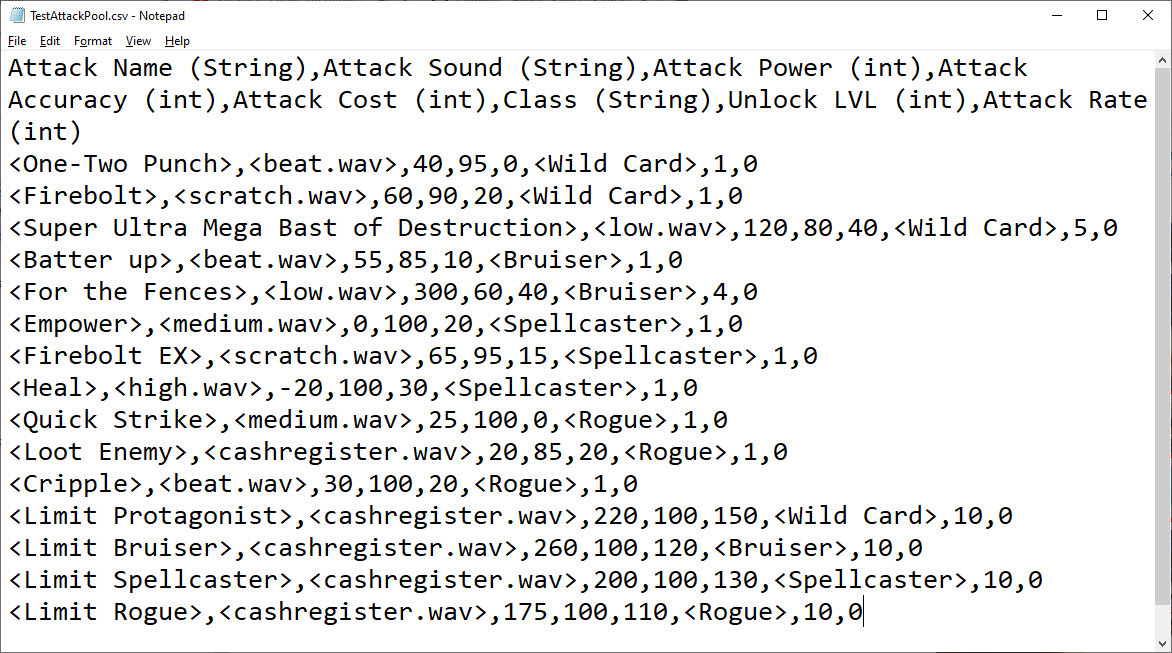
### Appendix 3.1: CSV’s in Excel



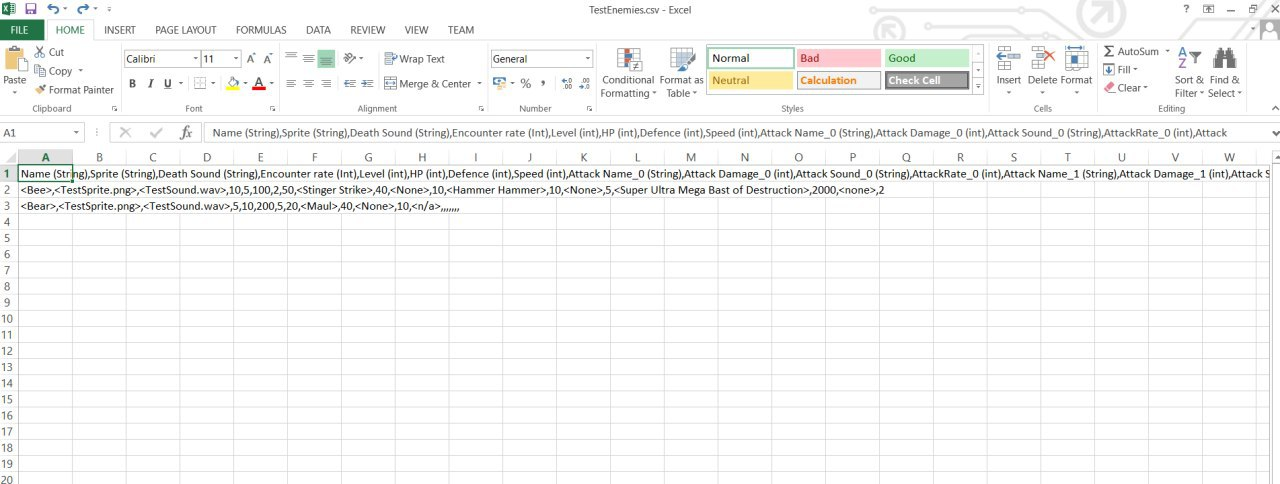


### Appendix 3.2: CSV’s in Text





## Appendix 4: Failure to read commas in excel



## Appendix 5: Code Snippets

### Appendix 5.1: Automatic integer reading

