IB Computer Science IA Report

Battle Map Generator

Criterion A - Planning

Criterion A: Planning

Defining the Problem

My client is an upperclassman of mine in Computer Science IB, from the graduating year 2023. We were both active members of the Dungeons and Dragons club at school and are players in a home-run campaign, so we had a shared understanding of the troubles one faces in this board game - “finding a unique and fitting battle map during a session”(Appendix A). There are plenty of battle map makers on the internet, however, some require a lengthy process which is not ideal for quick generations, while some sacrificed quality for speed, both do not satisfy their needs as an experienced Dunegone Master.

To fill this market gap, the solution my client proposed is to make the generation of quality battle maps more quick, accessible and customisable. As an art student, my client also had an affinity for hand-sketched-styled battle maps(Appendix B), which are rarely if ever offered in the battle map generation market. So they requested a map generator that could process user texture import and is applicable, easily changeable and adjustable to the generated map. It can also encourage user community engagement as they would have more agency.

Rationale for Solution

As the solution aims to be quick and easy, I have decided to make a desktop application which simplifies a battle map into mere grids or cells, where after the dungeon layout is generated procedurally, the textures can be filled in accordingly.

First, the location and sizes of the individual rooms or structures will be determined. Then paths will be generated to connect each of the rooms. After that, hazards or other map elements will be generated accordingly and added to the layout of the map. Finally, additional objects can be placed on the map before it is rendered using a pre-provided texture. In each of these steps, the user should survey each step of the process, and be able to add additional elements outside of the automated generation as they please, if not, a map should be created in under 30 seconds.

The map will be saved as a folder containing all the data necessary to render the map again on another device, which allows it to be revisited and further personalised. Additionally, the user can add a texture pack as a folder, containing all the necessary information for a map generation.

In terms of programming language, since I would have to organise mixed data types and a bitmap, I have decided to use Python for this program for its flexibility. This allows me to take advantage of PySimpleGUI for a lean front-end framework and PIL for rendering, whilst using Os for file manipulation.

Success Criteria

1. Ensure the application can be installed in any Windows 10 system with ease
2. A Home Screen which can access other windows
3. A Settings Screen where the user can adjust map settings
4. An Open File Screen where the user can import and open existing map files
5. The Open File Screen allows map previews
6. A Map Editor Screen to add and delete structures/cells
7. The user can save the project file, and export it to PNG, JPEG or PDF.
8. Allow maps to use imported textures.
9. A New Map Screen where the user can create a new Map with specific parameters
10. Automatic structure and path placement and generation
11. A Map can be generated in 5 minutes.