# 5 - Evaluation

## 5.1. Main objectives

As predicted my biggest constraint was time since many optional objectives where missed however the main objectives and core functionality of the game itself is entirely complete with a fully working game.

* A user interface (menu system and inventory / options / game control system)
  + I have created a working main menu system alongside options and a control scheme for the game using the mouse and left mouse button which is simple to use.
  + However, since there are no items that could be used in the game there was no need for inventory management system. And as a result, no inventory was created. Perhaps if offensive items where to be added to the game then an inventory system would be necessary to sort weapons and use them
* Turn based movement that works as intended
  + This has been successfully implemented with no bugs. Turns alternate between the player and enemy and movement during these turns contains no bugs through extensive testing with relatives, friends and teachers
  + Turns work based on the number of steps you can take each turn for both enemies and players and pathfinding is used to create player and enemy paths
* Clear indication of game objectives / how to win and a win scenario in every level
  + A dungeon key is shown on each level and a golden location is highlighted on the map to indicate where the exit is.
  + Although this is clear there is no verbal indication as to how to win the game it is implied therefore a future consideration would be to add a prompter telling the player the objective.
* Working HUD
  + There is a HUD to the side of the map showing player health score and the current turn which is about as much information required since all of information as to player locations fields of view and item/ exit locations are visible on the grid
* Enemy AI that doesn’t make the game impossible to beat but still provides a challenge
  + The number of enemies is based on the size of the map
  + The number of steps enemies can take is two less than the number of steps the player can take allowing the player to have a chance of escaping
  + The enemies will follow a player they can see and will move to the last location the player was spotted if they cannot see the player (individual memory).
  + The enemies move randomly when they cannot see the player which dumbs them down. Also, they forget about the player once they reach the last known location of the player and then will move randomly from there still creating a sense of a hunt but making the game a bit easier
  + However, having different types of enemies that have different ways of sensing the player and approaching may have increased the strategy involved in the game a bit more
* Multiple difficulty levels or some form of challenge the longer the game goes on
  + The game can continue forever if the player can survive forever however if they lose health and continue onto a new map they (continuing onto a new map reduces score but gives players the chance to create a higher score) keep the same level of health with no regeneration increasing the intensity of the game as it progresses
  + However, an increase in map size every time a new game is started would make the game become a lot more progressively harder. This would have been implemented had there been enough time
* Procedural map generation and enemy/ player placement algorithms
  + This objective has been completed perfectly with procedural map generation working flawlessly and allowing for the functionality to change map sparsity and even create a perfect maze map type (as described in documented design)
  + Enemies are placed on different locations in the map apart from each other
  + Player is placed away from enemies and out of line of sight using ray casting algorithm so that the player isn’t immediately attacked
  + Also, as a deign choice enemies can only see you whilst it is their turn to make the game much more accessible to beat since without this the game would be extremely difficult
* Low hardware requirements high performance
  + The game has many efficiency improvements listed below which allows the game to be run on the majority of modern and some outdated hardware including an old Lenovo ThinkPad containing an Intel Pentium g4560
  + A heap is created for the walls around each enemy (priority queue) that stores the walls in order of closest to furthest from the enemy. As a result calculating the 60 rays coming off of the enemy is much more efficient since the ray casting only has to be done until a wall with an intersection is found when searching through the heap since the closest wall is the wall that the ray hits first increasing efficiency by a factor of 75x (from 16 seconds to 0.2 seconds (using time.time()) on a 200 \* 200 perfect maze)
  + When running Dijkstra path finding for players mouse the pathfinding is only run if the mouse changes which block it goes over which prevents the path finding from being run once every frame dramatically improving performance during players turn
  + However, a relatively large proportion of ram is taken up by the maze matrix graph representation (however it is in the below 100 MB as a result not affecting performance of modern and semi-recent computers)
  + The program is mainly CPU intensive as a result there is no requirement for any GPU
* No game breaking bugs / software issues
  + Through testing no game breaking bugs have been found that make the game unplayable which as a result means the game isn’t tedious to play
  + However, one graphical bug has been found where the enemies final state is shown (in ray colours) before they move but this is reset once they start moving
  + This doesn’t affect gameplay however since it is all within the enemies turn so players can’t use this to make predictions helping them win
* Playable on at the very least a Windows 10 PC
  + The game is playable on windows machines and has been tested on those machines

## 5.2. Optional Objectives

Optional objectives where neglected the most since these where ambitious objectives only set if there was any surplus of time to complete these objectives after main objectives where completed. Again, the biggest issue was time constraints.

* Clear user interface
  + The main menu is clear and self-explanatory including the options menu and the after-win game menu (continue or quit options)
  + Also, a HUD helps the player understand their in-game situation. Furthermore, the user interface for in-game player movement was very simple to understand and use (just point and click)
  + Although if there was time verbal indication may have been helpful to include the objective is still clear and the end goal is also clear after a single playthrough
* Remappable Controls
  + Unfortunately, there was no time to meet this objective for players (since it had relatively high demand) this wasn’t core to main functionality and the standard controls are very commonly used and easy to grasp
* Simple non-intrusive HUD (takes up small part of screen and doesn’t get in the way of game)
  + This objective was met reasonably well with health score and turn displayed on the side of the game screen away from the play area but simple and easy to understand
* Tile weightings and different terrain types
  + There was no time to add this unfortunately however it wasn’t essential to the game (it would have increased the amount of strategy involved)
* Enemy alertness levels
  + This was fulfilled to an extent since enemies had an alert and searching mode as well as a hunting mode where they move to the location that they last spotted the enemy
  + However no “hivemind” alertness level was added since in hindsight this would make the game very hard for smaller maps.
* Smooth animation/ pleasing graphics
  + Although the assets (sprites) used where not appealing to many, the colours chosen where chosen through feedback from users saying purple and orange colours for instance where ugly. As a result, background colours where chosen to please users and create a better aesthetic appeal.
  + Smooth animation for players and enemies was achieved very well with players and enemies using interpolation to animate movement as described in the technical solution
  + Also, the background for menus is animated to an extent and lose and win screens are animated somewhat with a fade in affect
* Resizable windows with scalability
  + This was thought to be a fairly simple task however dealing with different aspect ratios caused complications and it was deemed unnecessary to implement this in particular
  + However, in better news, the game works at any resolution that is 16:9 aspect ratio since all buttons sprites animations background and fonts are scaled with the width and height of the screen
  + The framework is there just not the option in game
* Cheating prevention
  + This wasn’t an objective that is accomplished separately. The game works with no way to exploit the game mechanics other than intended with no game breaking bugs so cheating isn’t possible through the testing conducted
* Fog of war
  + In hindsight implementing this would have made the game very difficult and less strategic since players wouldn’t be able to plan ahead of enemies. Furthermore, time constraints meant that this wasn’t prioritised
* Save files / pause functionality/ leader board system
  + There was no time to implement this, so this objective was not met alongside high score storage unfortunately
* Accessibility settings
  + These where not implemented
* Music and volume settings
  + Music settings was implemented to turn background music on and off however volume is pre-set with no option to change it

## 5.3. User Feedback

Users amongst my friends’ group and family members where asked to provide and criticisms they have of the game and any praise or features they enjoyed. Here are the most common responses.

* The background animation and colours were very nice and pleasing
* The responsive buttons where a nice touch (in terms of colour changes when pressing and hovering over)
* Maze generation worked well and made the game unrepetitive
* Enemy placement was good, and many praised the fact that the maze was able to generate different sizes with number of enemies and steps based on that size
* The option to turn music on and off was praised also however I was criticized for having a fixed volume and no choice of songs
* Animation was noted and the interpolation made the game slightly better for users
* The score system was enjoyed since it created competitiveness at a low level between play testers however a lack of a leader board had made it annoying to try remember the highest score
* Players didn’t notice the lack of a save system likely owing to the fact that each level was short and quick, and players lost before they considered saving to continue later
* The sprites where criticised however the colour scheme of backgrounds where approved of.
* The lack of powerups to fight against the enemies was also another criticism
* Some called it too easy and some too hard so a difficulty system may have helped remedy this for users
* The majority of play testers understood the objectives with trial and error after one playthrough but others did struggle to grasp it without assistance so some verbal information may have been useful
* The movement in game was satisfying as it created situations where players had to place themselves strategically to avoid being spotted but collect as much loot as possible.
* Music choice was requested as the chosen background music was loved by some and hated by others
* Adding different terrain when asked was agreed upon by the play testers to increase the strategic complexity of the game
* The fact that players couldn’t have a way to increase health made the game challenging over longer periods of time since players had to become more and more careful when going for higher and higher scores

## 5.4. Potential improvements

* **Sprites and colours**
  + Choice of colours for buttons and backgrounds may have allowed users to have a better aesthetic experience
  + Animated sprites that fit the game background would have improved the user experience
* **Difficulty**
  + Although difficulty naturally increases as the game goes on (due to lower health) difficulty options and the option to set the number of steps and health of enemies of players
  + Options to set maze size and maze sparsity etc
* **Gameplay**
  + Add powerups
  + Add animations such as sprites legs moving
  + More enemy types with different movement strategies
  + More terrain types to create a higher level of strategy for the game
  + Option to set speed of animation since some found it to take too long and some found it to fast
  + Add a leader board system of high scores all time
  + Add a save file systems for players who have to leave the application but wish to return to it in the future
  + Add effects for damage taken and animation for when enemy spots player (other than ray colour change)
  + Add character customisation with each character having different perks
* **Settings**
  + Add volume options for music and choices of music
  + Add resolution and scalability options
  + Add option to see highest scores
* **Multiplayer**
  + Add a co-op mode where players must work together to solve a problem to progress in the maze
  + Add local and online game modes
  + Add online leader boards and games

## 5.5. Analysis and Conclusion

Overall, my project met all its main requirements with no major bugs. It works as I had described it would and had all the core functionality it required to be a fully working game users can enjoy on their PC’s. Furthermore, I had made efficiency improvements to make sure the game was playable on older hardware with limited capabilities.

Users enjoyed the games strategy elements finding the game challenging but always beatable and never putting the player in a situation where they are not in control and capable of escaping or winning. However, the game was a bit slow based since the user had to wait for every enemy to complete their movement before moving. The objective of the game was clear implicitly however some users struggled to figure it out.

The main criticism was the way the game looked in terms of sprites and difficulty settings since users wanted the choice of different difficulties. The enemy ai worked well in making the game challenging and strategic especially hunt mode where some players used the hunt mode to their advantage to lure enemies away from loot then escape and collect loot. Users did want more customisation options and music options however the majority of their requests where not core to the functionality of the project

The biggest challenge I was faced with whilst creating this game was the fact that I had to learn how to program using classes and had to use many complex algorithms which I had to graph out onto whiteboards to be able to grasp and tackle such as Dijkstra, ray casting maze generation and all the efficiency improvements that came with it such as sparsity controller for maze generation which doubled as an efficiency improver since the program didn’t have to run as long. Also, there were a couple of issues with the python programming language that caused a lot of time to be spent diagnosing such as lists being passed by reference and iteration issues.

Time constraints where what held this project back the most preventing quality of life improvements from being created.

To conclude I was successful in creating a fun strategic complete arcade game with many mechanics and procedural refreshing maze generation however as a game there are some quality of life improvements that can be added to improve the user experience.

## Bibliography

Here is a list of resources used in my project.

Pygame: <https://www.pygame.org/>