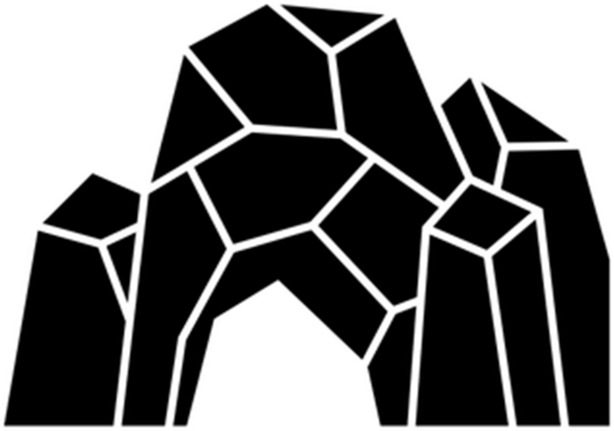
Collaborative Coding Project

Cavern: A Project Overview

Assignment 1



By the Collaborative Coding Project
8-9-2024

Introduction

This report will outline an overview of our project, what it is and how we plan on completing it. Our team's name is CCP (Collaborative Coding Project). We are coding a game called Cavern and will be coding it using C++ (CPP).

From left to right in the picture, our names are Ben Darlington, Ben Nicholson, Sam Bugden and Alex Poore.

Ben Darlington: darbe172@student.otago.ac.nz

Ben Nicholson: nicbe305@student.otago.ac.nz

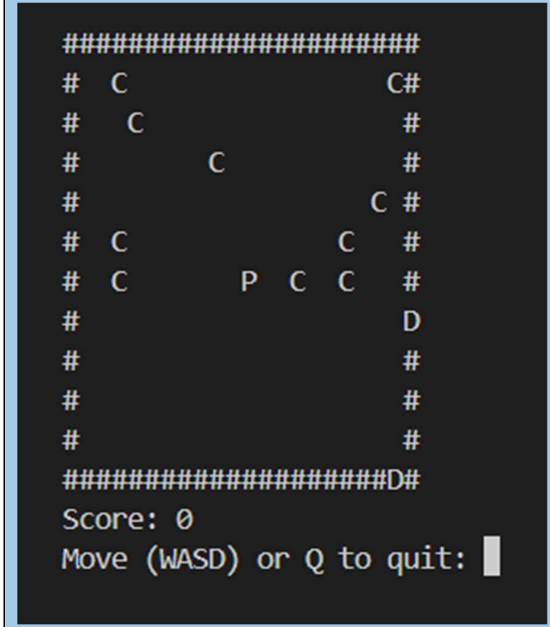
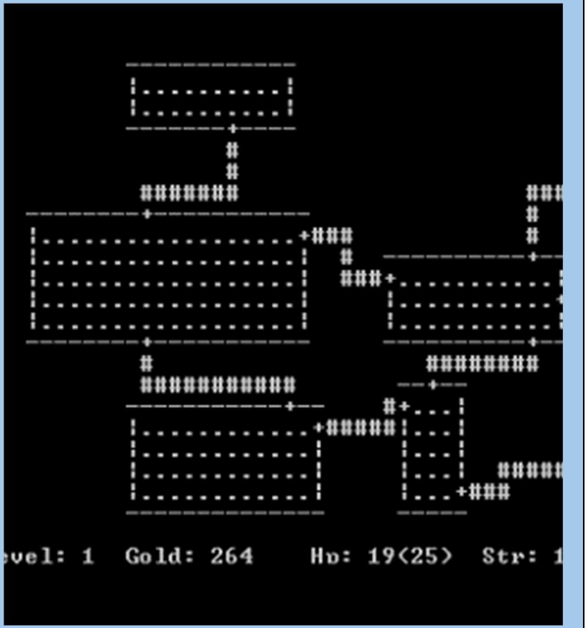
Sam Bugden: bugsa135@student.otago.ac.nz

Alex Poore: poale763@student.otago.ac.nz

Project Overview

We are making a game called Cavern. It will be a procedurally generated dungeon crawler that uses ASCII characters in the command line to create visualizations of rooms, items, enemies, and doors to new areas. Players start on the first level, fighting enemies, discovering keys to unlock doors, and progressing deeper into the game. Along the way, they can upgrade their health and attack points by collecting coins. Fighting enemies involves solving math equations on the screen as quickly as possible. Initial levels begin with simple plus and minus equations but as the player descends into deeper levels, divide and multiply equations begin to occur to increase complexity and difficulty. The game climaxes at a boss level in level 5, featuring a difficult boss to defeat.

This game is inspired by the 1980 classic "Rogue," one of the first games in procedural gameplay. We plan on taking Cavern back to basics by returning to the origins of the original dungeon crawler experience. This game aims to be a homage to one of the most basic yet influential games of the dungeon crawler genre.

Our Game "Cavern"	***The 1980's Game "Rogue"***
 'P' represents Player '#' represents Walls 'C' represents Coins 'D' represents Doors	

```
######################
#  C                C#
#   C                #
#        C           #
#                  C #
#  C             C   #
#  C       P  C  C   #
#                    D
#                    #
#                    #
#                    #
####################D#
Score: 0
Move (WASD) or Q to quit: 
```

```
         ------------
         |..........|
         |..........|
         -------+----
                #
                #
         ########                     ###
  --------+------------               #
  |..................+###             #
  |..................|  #   ----------+-
  |..................|   ###+...........
  |..................|      |...........
  |..................|      |...........
  --------+------------     ----------+-
          #                    ########
          ###########       ---+--
         -----------+---    #+...|
         |............+######|...|
         |............|      |...|
         |............|      |...|  #####
         |............|      |...+###
         --------------       -----

evel: 1   Gold: 264     Hp: 19(25)   Str: 1
```

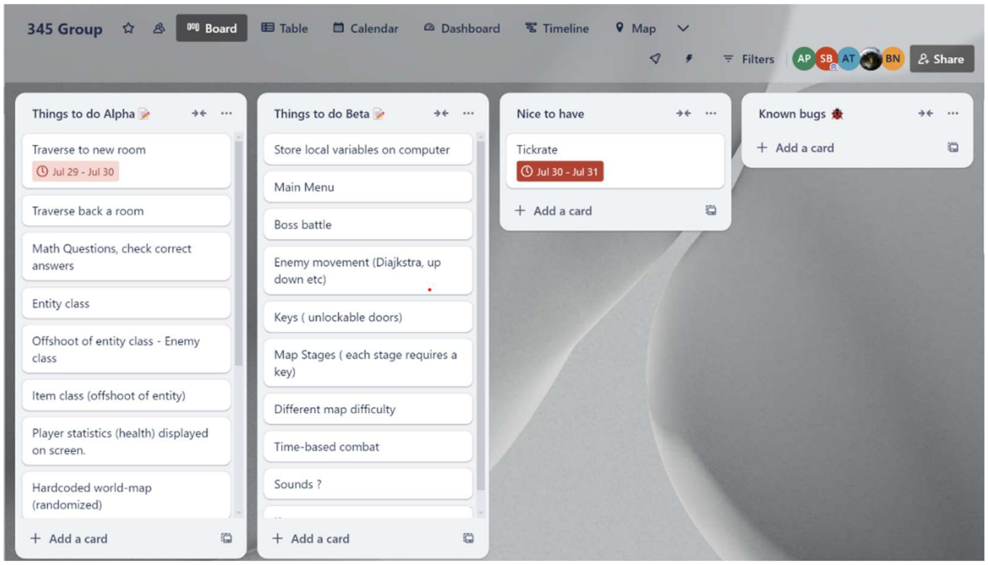
The design process

Communication

To maintain strong communication throughout the project, we meet in person four times a week: twice in tutorials, once for a lab session, and once as a self-directed meet up. Additionally, we consistently communicate via Facebook Messenger and keep our tasks updated on Trello. This ensures we are aware of each team member's tasks, the project backlog, and completed tasks.

Trello:

Trello is used to manage and allocate tasks for each team member. This is a great way to visually identify tasks that need to be done, who is working on each task and which ones have been complete. You can access our Trello board through this link: 345 Group | Trello.



Things to do Alpha

- Traverse to new room (Jul 29 - Jul 30)
- Traverse back a room
- Math Questions, check correct answers
- Entity class
- Offshoot of entity class - Enemy class
- Item class (offshoot of entity)
- Player statistics (health) displayed on screen.
- Hardcoded world-map (randomized)
- + Add a card

Things to do Beta

- Store local variables on computer
- Main Menu
- Boss battle
- Enemy movement (Diajkstra, up down etc)
- Keys (unlockable doors)
- Map Stages (each stage requires a key)
- Different map difficulty
- Time-based combat
- Sounds ?
- + Add a card

Nice to have

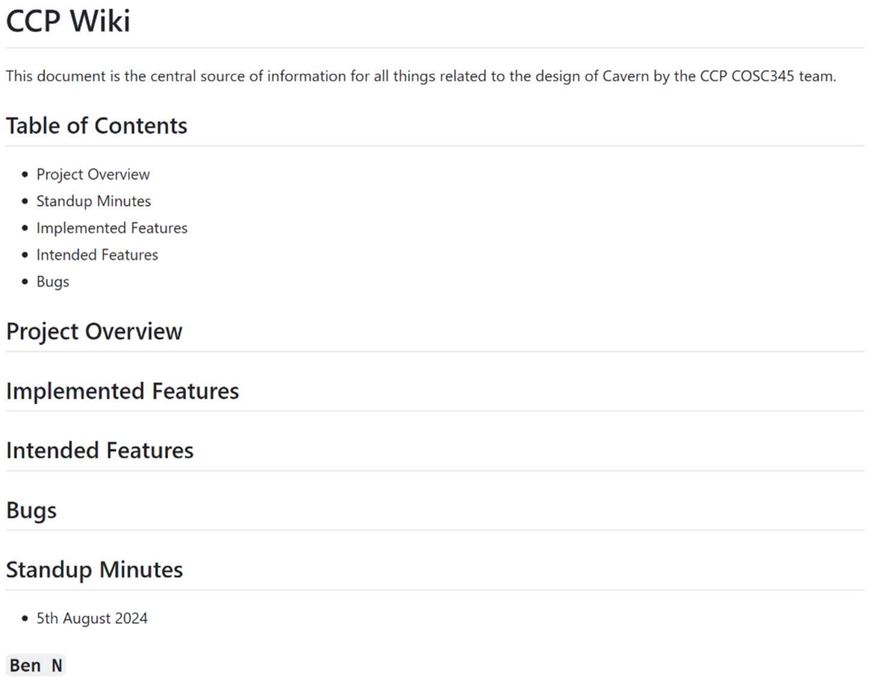
- Tickrate (Jul 30 - Jul 31)
- + Add a card

Known bugs

- + Add a card

Project Wiki:

To keep a well-documented codebase, we are using the GitHub Wiki feature to keep track of important information such as our meeting minutes, project backlog and resources like UML diagrams that should be known by the whole team.



CCP Wiki

This document is the central source of information for all things related to the design of Cavern by the CCP COSC345 team.

Table of Contents

Project Overview

Implemented Features

Intended Features

Bugs

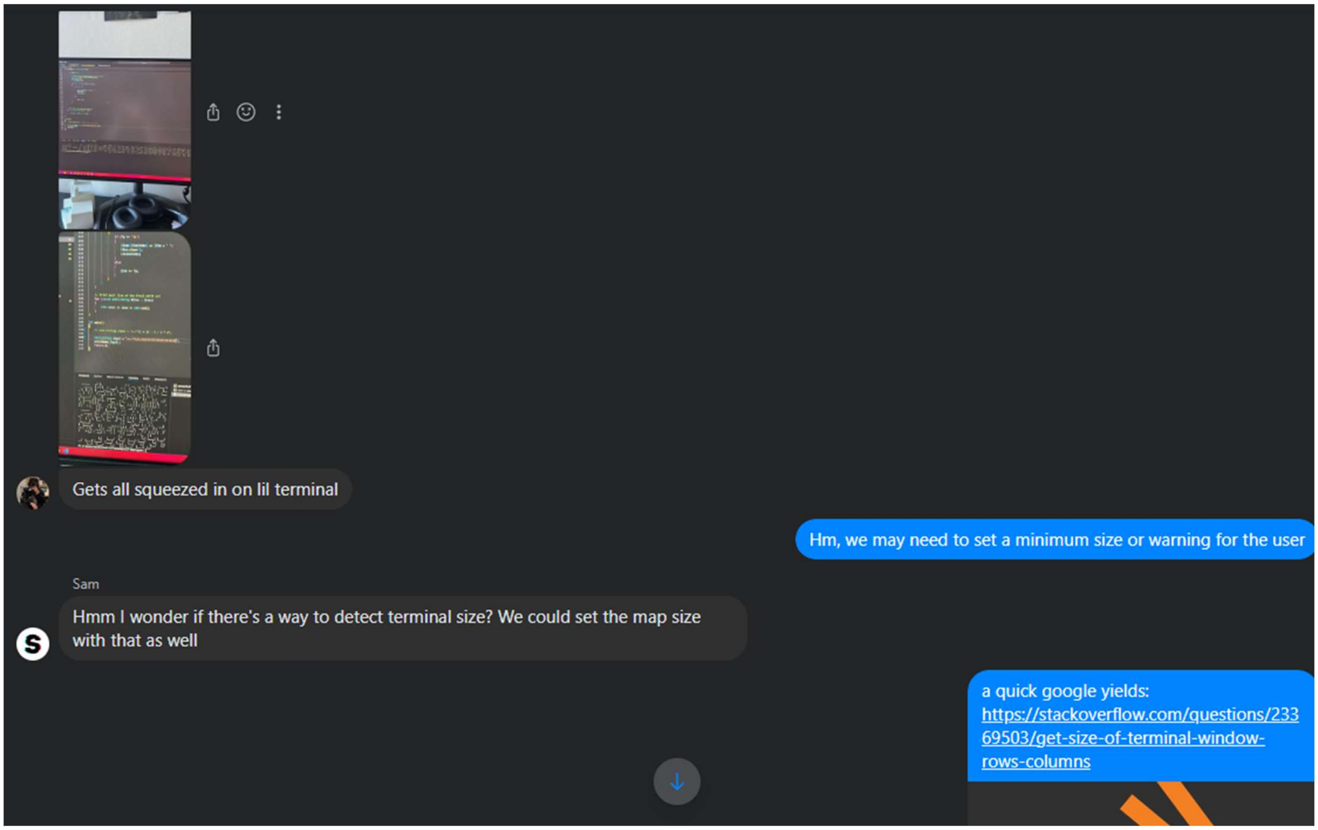
Standup Minutes

- 5th August 2024

Ben N

Messenger Group chat:

We use Facebook Messenger for an open communication pathway between team members. This is an easy way to communicate freely and keep everyone up to date on what is happening in the project.



Gets all squeezed in on lil terminal

Hm, we may need to set a minimum size or warning for the user

Sam

Hmm I wonder if there's a way to detect terminal size? We could set the map size with that as well

a quick google yields: https://stackoverflow.com/questions/23369503/get-size-of-terminal-window-rows-columns

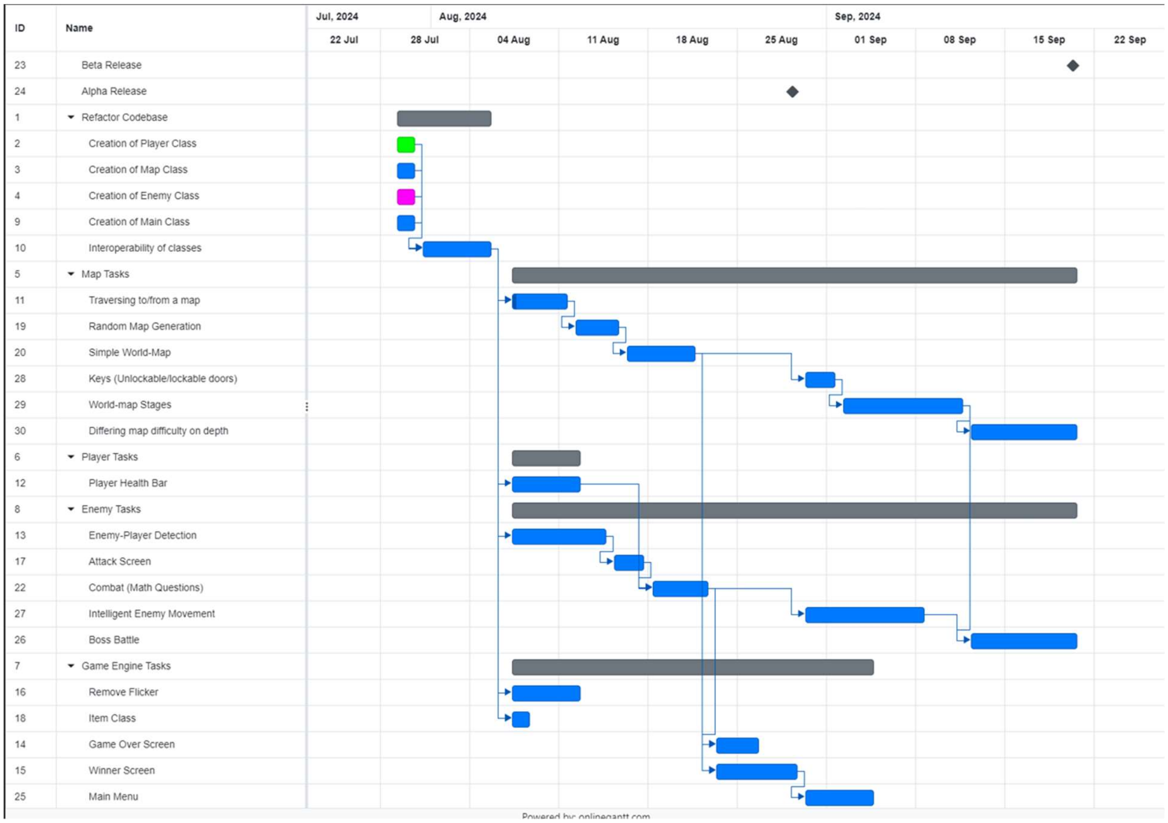
Planning

To ensure we develop a fully completed game, we have devised an in-depth plan on what tasks need to be completed for stage Alpha and Beta releases.

Tasks to Complete for Alpha	Tasks to Complete for Beta
Traverse to new room	Store local variables on computer
Traverse back to a previous room	Main Menu
Math Questions screen, check correct answers given player input	Boss battle
Entity class for enemy, chests, keys, etc.	Enemy movement (Dijkstra pathfinding and simple up down movement)
Player statistics (health) displayed on screen.	Keys to unlock doors
Hardcoded world-map (for alpha release only)	Procedural map generation
Combat detection triggered by moving onto an enemy	Different map difficulty
Separate Attack Screen that triggers combat based ono your enemy	Time-based combat
Game Over screen, triggered when player health decreases to 0.	Sound effects
Win Screen when player reaches their 5th level.	Keys
Add stairs/door to next level	
Smooth Movement (remove flicker with cursor position)	

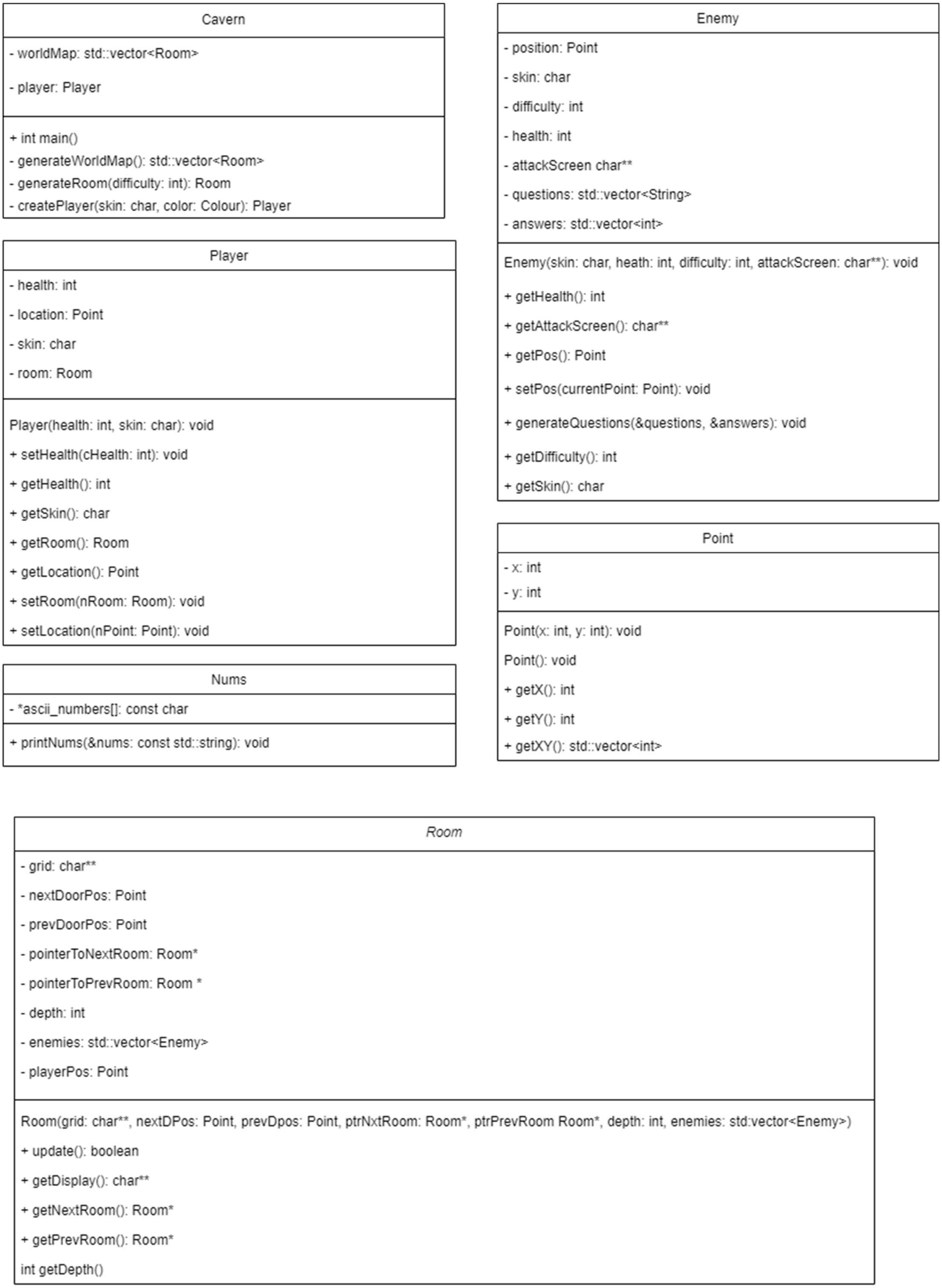
Project Gannt Chart:

This Gannt chart visualizes each task in our project backlog. It shows how long it is estimated to take and what dependencies are attached to it. The project deadlines are represented with a diamond (Alpha and Beta release).



ID	Name
23	Beta Release
24	Alpha Release
1	Refactor Codebase
2	Creation of Player Class
3	Creation of Map Class
4	Creation of Enemy Class
9	Creation of Main Class
10	Interoperability of classes
5	Map Tasks
11	Traversing to/from a map
19	Random Map Generation
20	Simple World-Map
28	Keys (Unlockable/lockable doors)
29	World-map Stages
30	Differing map difficulty on depth
6	Player Tasks
12	Player Health Bar
8	Enemy Tasks
13	Enemy-Player Detection
17	Attack Screen
22	Combat (Math Questions)
27	Intelligent Enemy Movement
26	Boss Battle
7	Game Engine Tasks
16	Remove Flicker
18	Item Class
14	Game Over Screen
15	Winner Screen
25	Main Menu

Early Alpha UML Diagram:



Cavern

- worldMap: std::vector<Room>
- player: Player

+ int main()
- generateWorldMap(): std::vector<Room>
- generateRoom(difficulty: int): Room
- createPlayer(skin: char, color: Colour): Player

Enemy

- position: Point
- skin: char
- difficulty: int
- health: int
- attackScreen char**
- questions: std::vector<String>
- answers: std::vector<int>

Enemy(skin: char, heath: int, difficulty: int, attackScreen: char**): void
+ getHealth(): int
+ getAttackScreen(): char**
+ getPos(): Point
+ setPos(currentPoint: Point): void
+ generateQuestions(&questions, &answers): void
+ getDifficulty(): int
+ getSkin(): char

Player

- health: int
- location: Point
- skin: char
- room: Room

Player(health: int, skin: char): void
+ setHealth(cHealth: int): void
+ getHealth(): int
+ getSkin(): char
+ getRoom(): Room
+ getLocation(): Point
+ setRoom(nRoom: Room): void
+ setLocation(nPoint: Point): void

Point

- x: int
- y: int

Point(x: int, y: int): void
Point(): void
+ getX(): int
+ getY(): int
+ getXY(): std::vector<int>

Nums

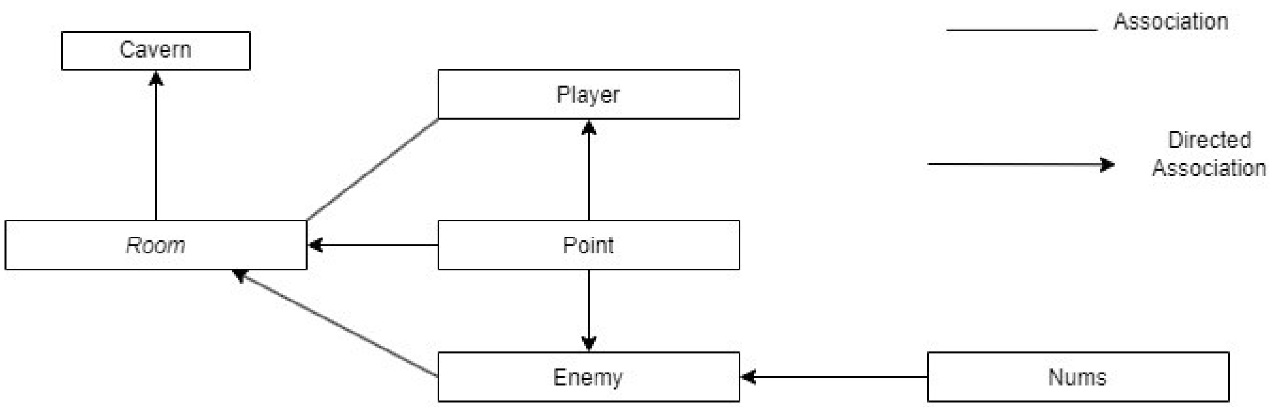
- *ascii_numbers[]: const char

+ printNums(&nums: const std::string): void

Room

- grid: char**
- nextDoorPos: Point
- prevDoorPos: Point
- pointerToNextRoom: Room*
- pointerToPrevRoom: Room *
- depth: int
- enemies: std::vector<Enemy>
- playerPos: Point

Room(grid: char**, nextDPos: Point, prevDpos: Point, ptrNxtRoom: Room*, ptrPrevRoom Room*, depth: int, enemies: std:vector<Enemy>)
+ update(): boolean
+ getDisplay(): char**
+ getNextRoom(): Room*
+ getPrevRoom(): Room*
int getDepth()



Risk Assessment

Risks are an important factor to consider in development projects such as this, so we have developed a detailed risk assessment that not only identifies potential risks but to also outlines a plan to counteract their consequences.

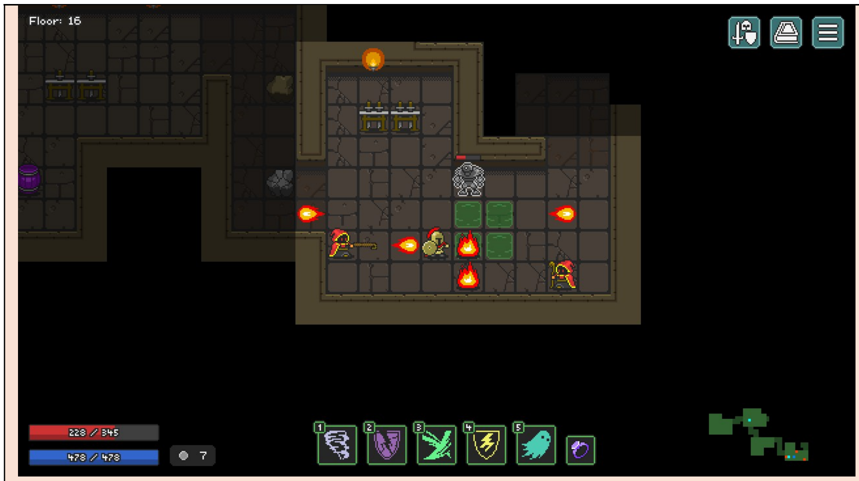
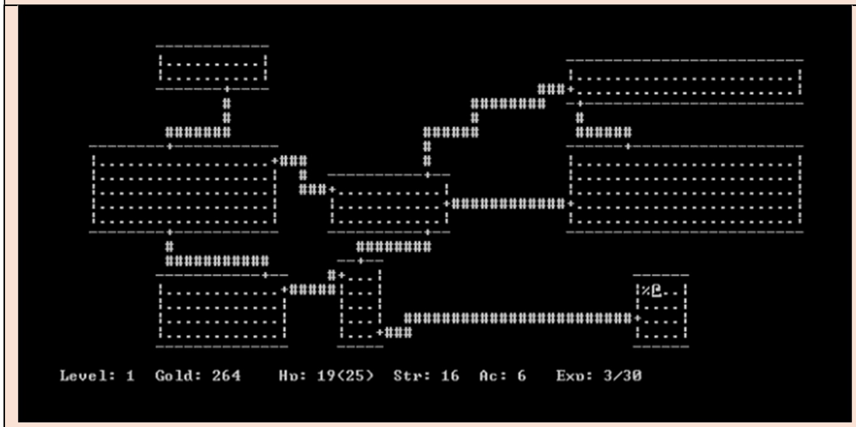
Risk	Strategy to Combat It
Issues related to Git	Establish best practices for Git usage; maintain backups of repositories and practice Git functions thoroughly to get confident using them.
Team member incapable to complete a task due to an unforeseen circumstance.	Provide additional support and resources; redistribute workload temporarily to other team members.
Project guidelines change	Maintain flexibility in planning; communicate changes clearly and promptly.
Communication breakdown	Set up regular check-ins (weekly); use multiple communication channels.
Team burnout from sprints	Ensure adequate breaks; monitor workload; encourage time management.
Hardware issues on our personal computers	Maintain backups through Git. In the case of lost code, re-evaluate the priorities of current tasks to recover sections that were lost.
Worldwide disaster	Follow guidelines from authorities, re-evaluate tasks according to the specific scenario.
Team member illness	Have a contingency plan to redistribute workload to other team members.
Lack of progress due to other deadlines in the University.	Prioritize tasks; adjust timelines; allocate resources effectively.
Issues with C++ due to inexperience in the code.	Consult documentation; seek external expertise if needed.
Need to refactor code	Schedule regular code reviews; allocate time for refactoring in sprints.

Tech debt due to sprints	Address technical debt regularly; balance new features with debt repayment.
Overlap of work	Clearly define roles and responsibilities; improve task allocation.

Similar Apps on the Market

There are several existing apps that are similar to our project, however we believe we are bringing a fresh combination of ideas that has not been seen before. Some programs similar to our game include Rogue, Labyrinth of Legendary Loot, Binding of Isaac and Times attack. Rogue is an old and simple ascii based dungeon crawler that heavily influences our games visual style and movement design, labyrinth of legendary loot is a recent game that inspired the combat aspect of our games and item collection systems. Times attack is a three-dimensional dungeon exploring game that inspired our games math-based combat and educational approach.

	The Binding of Isacc A randomly generated action RPG shooter with heavy Rogue-like elements.
	Times Attack A third-person 3D adventure game that guides students through a dangerous fantasy world with a focus on improving times tables.

 	**Labyrinth of Legendary Loot** A simple turn-based roguelike dungeon crawler focused on tactical combat.
 Level: 1 Gold: 264 Hp: 19(25) Str: 16 Ac: 6 Exp: 3/30	**Rogue** Players control a character as they explore several levels of a dungeon seeking a prize in the dungeon's lowest level

Customer interest

Cavern’s unique combat mechanics require players to solve math problems before defeating enemies. This educational aspect of the game could be an invaluable tool to educators who can teach children about arithmetic through a fun and engaging game.

This technique of game-assisted learning is a proven method, as Manar S Alotaibi states in her paper “Game-based learning in early childhood education: a systematic review and meta-analysis"[1]. The paper states: “Key findings across the studies show that game-based learning was effective in improving various early-learning outcomes including numeric skills, literacy”. The paper went on to state: “Digital game formats like mini-games, educational apps and programs promoted cognitive development, problem-solving and creativity”[2]. This paper clearly backs up the notion that games such as our own are useful for childhood educators. Improving a student's numeracy skills whilst also increasing their general confidence using computers.

[1] Manar S Alotaibi - “Game based learning in early childhood education: a systematic review and meta-analysis” 3/08/2024 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC11018941/#:~:text=The%20study%20found%20that%20the,development%20in%20early%20childhood%20education

[2] Manar S Alotaibi - “Game based learning in early childhood education: a systematic review and meta-analysis” 3/08/2024 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC11018941/#:~:text=The%20study%20found%20that%20the,development%20in%20early%20childhood%20education

Furthermore, a report by researchandmarkets.com states: “The global market for Education Apps was valued at an estimated US$5.4 billion in 2023 and is projected to reach US$10.8 billion by 2030”[3](Figure a). Another source states that “the gaming market is experiencing a resurgence of 90s-2000s classics, with an 851% increase in the popularity of old-school titles from that era[4]. These sources clearly show that the market for educational games big and also growing. Cavern is not only a game inspired by the 1900’s roguelike genre but also falls within the education category. Therefore, we can be sure of customer interest in Cavern.

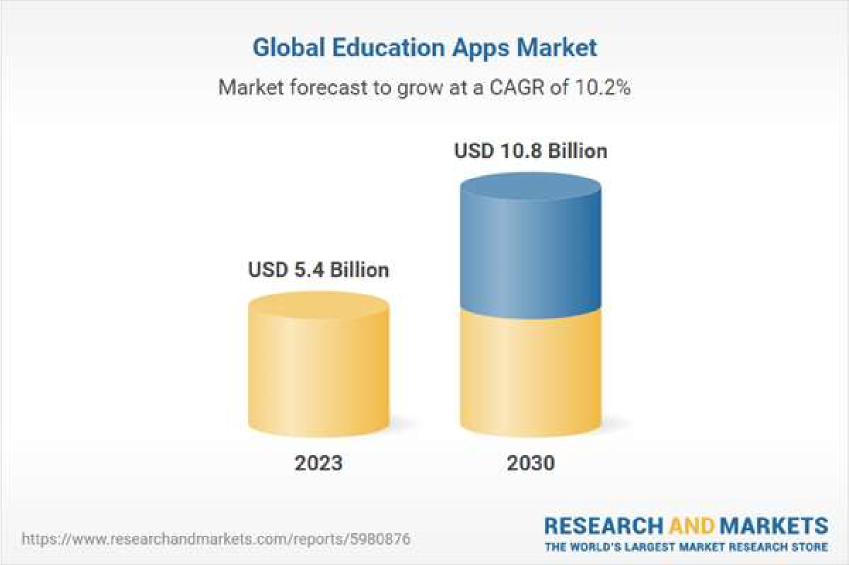


Figure a. Source: https://www.researchandmarkets.com/reports/5980876/education-apps-global-strategic-business-report?utm_code=rfzj9d&utm_exec=chdomspi 3/08/2024

[3] Education Apps –Global Strategic Business Report 3/08/2024 https://www.researchandmarkets.com/reports/5980876/education-apps-global-strategic-business-report?utm_code=rfzj9d&utm_exec=chdomspi

[4] Source: Retro classics are making a comeback as people get back into gaming during lockdown, G2A data reveals – G2A.COM - Official Corporate Website