

Project Proposal Draft for UTM CSCI 352 Battle Of The Professors

Mason Bearden Jordan Taylor

Abstract

We are creating a text adventure game which uses puzzles/trivia/logic problems to test the player. The game is based off UTM CSCI classes and professors. The game will include stat management, challenges to solve, and navigation through a map. The player is playing as a student who must traverse a map, solve puzzles, meet other students, and eventually fight a professor. The target audience is students and teachers specific to UTM, or gamers in general who find value in the game.

1. Introduction

The user plays as a character defined as a "student" who needs to pass through their CSCI classes. They will traverse through different maps acting as mazes with which they must solve events to raise/lower their grade. Each map will represent a different CSCI class which will be controlled by a different Professor. The student's health will act as their grade and as they complete challenges, it will decrease or increase. The student must beat the boss of the map "Professor" in order to pass the class. All events including the boss fight will unfold as puzzles/trivia/logic games. For example, while traversing the maze, a student may come across a challenge that asks for a series of questions to be solved. Somewhere in the maze will stand a fellow "Student" who will help the player by giving advice/crucial information on the class. Separate stats: Intellect and Sanity will determine the values lost/gained when completing challenges. Sanity will be a student's defense to health loss, while Intellect will influence how quickly a challenge can be completed. The expected users that would be interested in this game are the students (Past and Present of CSCI UTM), the professors (UTM), and those who enjoy RPGs (role-playing game). Although the game is specific to a University all gamers are welcome.

1.1. Background

We are interested in learning the concepts of a puzzle adventure game.. It will provide a variety of skill sets that we will be able to use in future programs. Having the experience of creating an RPG could potentially open up job opportunities.

1.2. Impacts

This game will hopefully provide enjoyment to those who are stressed or ill, making life a little more exciting.

1.3. Challenges

The toughest aspect of our game is managing our time. We have many ideas for implementation and little time for them all. I believe that the toughest portion of the game will be all the small things working together such as: traversing through the map, stat loss / gain and implementation of challenges. We would like to implement diversity in said challenges.

2. Scope

Scope: The bare minimum for the project that we want to accomplish is having at least one map with puzzles/trivia/logic problems working. A functional boss and student character and stat implementation. Stretch goals would include: 1. adding all maps/professor 2. Adding additional npc involvement

2.1. Requirements

The functional and non-functional requirements were difficult for us to determine. I went for things I think the player should be able to do in the game, when it came to the functional requirements. For the non-functional, I went for things the player should have access to.

Use Case ID	Use Case Name	Primary Actor	Complexity	Priority
1	Move through map	Player	low	1
2	Solve Challenge	Player	Med	1
3	Challenge Boss	Player	High	1

TABLE 1. SAMPLE USE CASE TABLE

2.1.1. Functional.

- Player should be able to save game state and load game state
- Player should be able to traverse map
- Player should be able initiate challenges and complete them
- Player should be able to create new game and restart progress

2.1.2. Non-Functional.

- Stats - Player should be able to track stats/stat changes

2.2. Use Cases

Use Case Number: 1

Use Case Name: Move through map

Description: The player will move through the map to explore/progress

- 1) Player is presented with map and options for movement.
- 2) Player picks direction and proceeds.
- 3) Player position and map are updated.

Termination Outcome: The player is now in a new location.

Alternative: Direction does not exist

- 1) Player tries to move in a wrong/unaccessible direction

Termination Outcome: Player is given warning that movement can not happen.

Use Case Number: 2

Use Case Name: Solve challenge

Description: The player is presented with a challenge that will determine health loss/gain

- 1) Player is presented with a challenge in the form of riddle/puzzle/logic
- 2) Player chooses the solution to challenge.
- 3) Player health is increased or decreased based on decision.

Termination Outcome: Player is now able to move on.

Use Case Number: 3

Use Case Name: Challenge Boss

Description: The player is presented with a boss that will determine if they complete the map and move to the next class

- 1) Player is presented with a challenge in the form of a boss
- 2) Player chooses battles boss in challenge
- 3) Player is able to move to the next class

Termination Outcome: Player is now able to move on.

2.3. Interface Mockups

3. Project Timeline

Go back to your notes and look up a typical project development life cycle for the Waterfall approach. How will you follow this life cycle over the remainder of this semester? This will usually involve a chart showing your proposed timeline, with specific milestones plotted out. Make sure you have deliverable dates from the course schedule listed, with a plan to meet them (NOTE: these are generally optimistic deadlines).

4. Project Structure

At first, this will be a little empty (it will need to be filled in by the time you turn in your final report). This is your chance to discuss all of your design decisions (consider this the README's big brother).

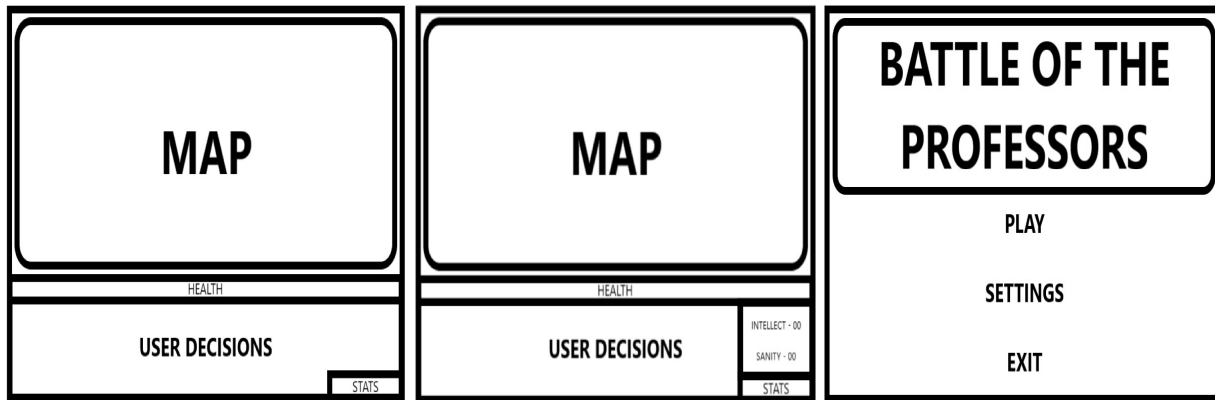


Figure 1. pic

4.1. UML Outline

4.2. Design Patterns Used

Make sure to actually use at least 2 design patterns from this class. This is not normally part of such documentation, but largely just specific to this class – I want to see you use the patterns!

5. Results

This section will start out a little vague, but it should grow as your project evolves. With each deliverable you hand in, give me a final summary of where your project stands. By the end, this should be a reflective section discussing how many of your original goals you managed to attain/how many desired use cases you implemented/how many extra features you added.

5.1. Future Work

Where are you going next with your project? For early deliverables, what are your next steps? (HINT: you will typically want to look back at your timeline and evaluate: did you meet your expected goals? Are you ahead of schedule? Did you decide to shift gears and implement a new feature?) By the end, what do you plan on doing with this project? Will you try to sell it? Set it on fire? Link to it on your resume and forget it exists?

References

- [1] H. Kopka and P. W. Daly, *A Guide to L^AT_EX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.