# Senior Project Proposal

Presented by: Ethan Powell

Bachelor of Science in Cybersecurity & Computer Science

Presented to: Dr. Paul West

Expected Graduation Date: May 2022

Problem Statement

A dungeon-master for the modern day fifth edition of Dungeons and Dragons must keep track of an incredible number of things at one time. From enemy health, player abilities, player inventory, spell ranges, damage types, and more, a dungeon-master’s job is to ensure that the session runs smoothly, and the game keeps a sense of realism in response time for actions that occur. This is quite simple at lower player levels, as abilities and inventories have only just begun, but as the campaign progresses, increasingly difficult enemies and ramping player power can cause this process to become hectic and time consuming. However, despite the increasing load that continues to be placed on the dungeon-master, the help of a software assisted game tool can significantly decrease the time needed to process a turn and increase the enjoyment of not only the players, but the dungeon-master themselves. When using a software assisted game tool, a dungeon-master can quickly calculate the damage for a spell, keep track of benefits and detriments to player characters, and run test combat encounters to ensure an enjoyable game when a session begins. In contrast to other tools of similar aim, this tool will collate the use of inventory tracking, combat running, and spell referencing in one easy to use and free package, removing the need to consult multiple books for information and use several expensive tools to achieve the same effect. The creation of an all-in-one software assisted game tool for fifth edition Dungeons and Dragons is necessary for relieving the stress that is placed on a dungeon-master and ensuring a positive experience for players involved.

Project Description

This project will focus on providing several tools that a dungeon-master can utilize in tandem with one another to accomplish daunting tasks that would otherwise take days if done by hand such as: track character information, run combat encounters as multiple players, calculate spell ranges, damage types, and targets, and reference database information for custom abilities and items. With the ability to upload player’s form filled PDF character sheets and monster stat information files, manual input of every piece of information will become obsolete, and information input will be streamlined and simple. This will allow the use of such information to be acted upon in a combat encounter test system, providing the option for one dungeon-master to run a theoretically infinite number of players at one time without the need to reference abilities and actions. Supported by a back-end database structure, information will be able to be searched easily using a user-friendly search system, providing a manual alternative if automatic encounters are not preferred.

Implementation Languages

This project will utilize C++ as its main implementation, providing tried and true language based on the foundational language of C. This will be supported by several graphical libraires, as well as SQL implementation for back-end database support. The database will be accessed through the use of SQL queries using the SQLite C/C++ interface, providing a lightweight and easy to learn database engine for use in personal and professional projects.

Libraries, Packages, and Development Kits

The following libraries and packages that will be used contains, but is not limited to the following:

1. **Elements**  
   “[A] lightweight, fine-grained, resolution independent, modular GUI library” This library will be the basis for most, if not all, the UI involved in the program.
2. **SQLite**  
   The database engine that will support data access and storage in a lightweight and easy to use package.
3. **C++ STL libraries**  
   These libraries will be utilized to complete simple actions such as creating random numbers, allowing for quick sorting, easy implementation of common data structures, and more.

Additional Software & Equipment

The following software and equipment will be needed in the construction of the project:

1. **Fifth Edition Dungeon’s and Dragon’s Rulebook**  
   This is the game that the project will be based around.
2. **Text Editor**Will be used to code and upload test files for development.
3. **Oracle VirtualBox**  
   Required for development inside a Linux environment.
4. **Ubuntu Linux**Required for development operating system and testing environment.
5. **G++**Required for compilation.

Motivation

My motivation for this project stems from my personal love for tabletop roleplaying games and my experience in simple software development from my collegiate career. As an avid fifth edition Dungeons and Dragons player myself, attempting to use several previously developed tools never felt quite right, and none of them ever met the requirements that I felt should have been obvious. It is my goal with this project to ensure that I gain experience in developing a useful software tool that would be of an appropriate level to be a commercial product while also filling the need for such a tool in the Dungeon’s and Dragons community that will be free to use and available to people just like me. For the foreseeable future, I find myself continuing to enjoy this game with friends and having a tool to make my life easier in such an endeavor that meets all of my needs (or is able to be changed to do so) is an attractive proposition.

Outline of Research Efforts

The first goal of this project will be to implement the base functions of the game. This will include rolling dice, uploading a character sheet, and creating monster statistics. This portion will just be groundwork to be improved upon once the GUI is implemented. Secondly, database interaction will be introduced, ensuring that game statistics such as stats for characters and spell effects can be uploaded and referenced from inside the program. Because of the standardized spell framework already provided in the tabletop game, little modification will be required to ensure the program can interpret such information. Finally, the GUI will be implemented such that a user will be able to perform the actions of adding new characters, monsters, and spells easily and without much training. A short tutorial will be implemented to teach a new user how to use the program at the base level, along with help sections on each “screen” of the tool. Finally, a combat encounter tester will be implemented, tying each section together such that the stats from monsters, spells, and characters can be referenced to replicate an actual combat encounter that would be completed in the game. One roadblock (or setback) that I foresee will be the uploading of information for the base game. Because of the extent of spells and monsters that already exist, much of the time will be spent ensuring that all are readily accessible for the program to interpret. Research will also be needed to ensure an understanding of the SQLite framework and its limitations that may interfere with the software’s ability to perform the required actions.

Tentative Schedule

|  |  |  |
| --- | --- | --- |
| Task | Start Date | Completion Date |
| Create Project | January 3, 2022 | January 3, 2022 |
| Install Software and Libraries | January 3, 2022 | January 7, 2022 |
| Complete Base Software Functions | January 10, 2022 | January 21, 2022 |
| Complete Database Functionality | January 24, 2022 | February 4, 2022 |
| Create GUI Functionality | February 7, 2022 | February 25, 2022 |
| Create Combat Encounter Functionality | February 28, 2022 | April 1, 2022 |
| Create Test Plan | April 4, 2022 | April 8, 2022 |
| Testing Phase | April 11, 2022 | April 15, 2022 |
| Analyze and Fix Bugs | April 18, 2022 | April 29, 2022 |
| Complete and Defend Project | April 29, 2022 | May 1, 2022 |