

# Cal Poly Computer Engineering Senior Project

## Dungeon Master's Sheet Aid

Adviser: Dr. Andrew Danowitz

By: Alex Grazela



Table of Contents:

<b>Table of Contents:</b>	1
<b>Abstract:</b>	2
<b>Background:</b>	3
What is a tabletop rpg?	3
Previous Solutions and Their Problems:	4
Problem Statement:	5
<b>Engineering Specifications:</b>	5
<b>Engineering Design:</b>	6
<b>Conclusion/Future Work:</b>	10
<b>References:</b>	<b>Error! Bookmark not defined.</b>

## Abstract:

The Dungeon Master's Sheet Aid is a program made to assist players of tabletop rpg games in keeping track of and using their "character sheets", a vital part of the game. In particular. It is made to be helpful to the players taking the role of "Dungeon" or "Game" Master, as these players need to manage many more character sheets than anyone else.

## Background:

### What is a tabletop rpg?

Tabletop rpgs are games wherein a single player takes the role of “master” and the rest of the players take the roles of “player characters” within a game universe created by the master. To describe their “player character” in the game world, the players make use of information compiled into a “character sheet”. Depending on the game being played, character sheets usually include things such as the character’s name and player’s name, character’s backstory, what their stats, such as strength or intelligence, are, what they are skilled in, and often how they can fight. In order to do things in the game, players roll dice, and add modifiers based on information about their character to determine success and failures of their actions, so keeping a character sheet that tabulates that information is essential. The players need to keep track and carefully manage their own sheets, while the master needs to know all of the player’s sheets, as well as sheets for important non-player characters in the game world that he creates. As such, the number of sheets a master needs to look at during the game can become overwhelming, and if any player loses their information it makes it difficult to play the game.

## Previous Solutions and Their Problems:

Two previous ways of keeping track of character information that were taken into account for this project were: Holding onto physical paper sheets, and keeping electronic sheets in the online application “roll20.net”<sup>1</sup>. Paper sheets are the most easily lost, and so the most unreliable method, on top of the fact that they cannot be used for online play. Roll20.net is a browser application that is meant to facilitate online tabletop rpg playing, and it includes it’s own electronic character sheets. The character sheets in roll20 are helpful and detailed, providing features such as automated rolling, but for the master, who needs to be looking at multiple sheets at a time, they are far too bulky, as shown below in figure 1.

Figure 1: A character sheet in roll20

<sup>1</sup> <https://roll20.net/>

This sheet takes up about half the screen, meaning opening any more than 2 will cause overlapping, and prevent the master from seeing the representation of the setting that they made (seen in the background to the left of the character sheet), thus making the features difficult to use at the same time.

## Problem Statement:

Based on the shortcomings of the previous methods, the problem statement for this project is: "Create a small sized, quick to navigate character sheet displaying gui that can display multiple character sheets without overlapping, nor taking up so much room as to make it impossible to look at anything else at the same time, provides automated rolling services for different player actions, and can save entered data to be used later on the same or a different device."

## Engineering Specifications:

In order to meet the problem statement, two particular engineering specifications have to be met; namely being able to display a gui, and having a persistence layer. For choosing how to display the information, a decision had to be made on what type of device the application would run on. If the project was to run on mobile environments, it would have to use the android or IOS toolkits, whereas running it on a desktop environment opened options for things like javascript, javax.swing, or Qt. For the data persistence layer the choice was between using a regular file, using an sql database, or

something else. Since python provides a “pickle” module<sup>2</sup> that works as a persistence layer by writing to a special bytestream file, which can then be “unpickled” to load the data, and also has an implementation of Qt<sup>3</sup>, which would make it easy to generate the gui I needed. These factors, along with the fact that any time a user would I therefore decided to make the code solely in python, meant to run on a desktop environment,

## Engineering Design:

In order to meet the requirement of limited space usage, some way to share space with other character sheets, as well as limit the amount of information displayed at once was needed. The solution was a double tabbed gui, pictured in figure 2 below.

---

<sup>2</sup> <https://docs.python.org/3/library/pickle.html>

<sup>3</sup> <http://pyqt.sourceforge.net/Docs/PyQt5/index.html>

Ability	Score	+	-	Save	Save Button
Strength	17	+	-	<input checked="" type="checkbox"/> Strength save	Save
Constitution	17	+	-	<input checked="" type="checkbox"/> Constitution save	Save
Dexterity	14	+	-	<input type="checkbox"/> Dexterity save	Save
Wisdom	10	+	-	<input type="checkbox"/> Wisdom save	Save
Intelligence	14	+	-	<input type="checkbox"/> Intelligence save	Save
Charisma	6	+	-	<input type="checkbox"/> Charisma save	Save

Figure 2: The double tabbed gui, the upper level of tabs are characters, and the lower level is the information needed, broken up into relevant subgroups

Using this gui, the automated rolling feature is implemented by way of the visual pushbuttons. For example, clicking the “save” button in the strength row will roll a “strength save”, which is a type of action the player may need to take during the course of the game. It takes into account the strength score, given in the text box in the same row, and the check box for “strength save” which indicates proficiency in the action. The result of clicking the button is shown in figure 3 below.

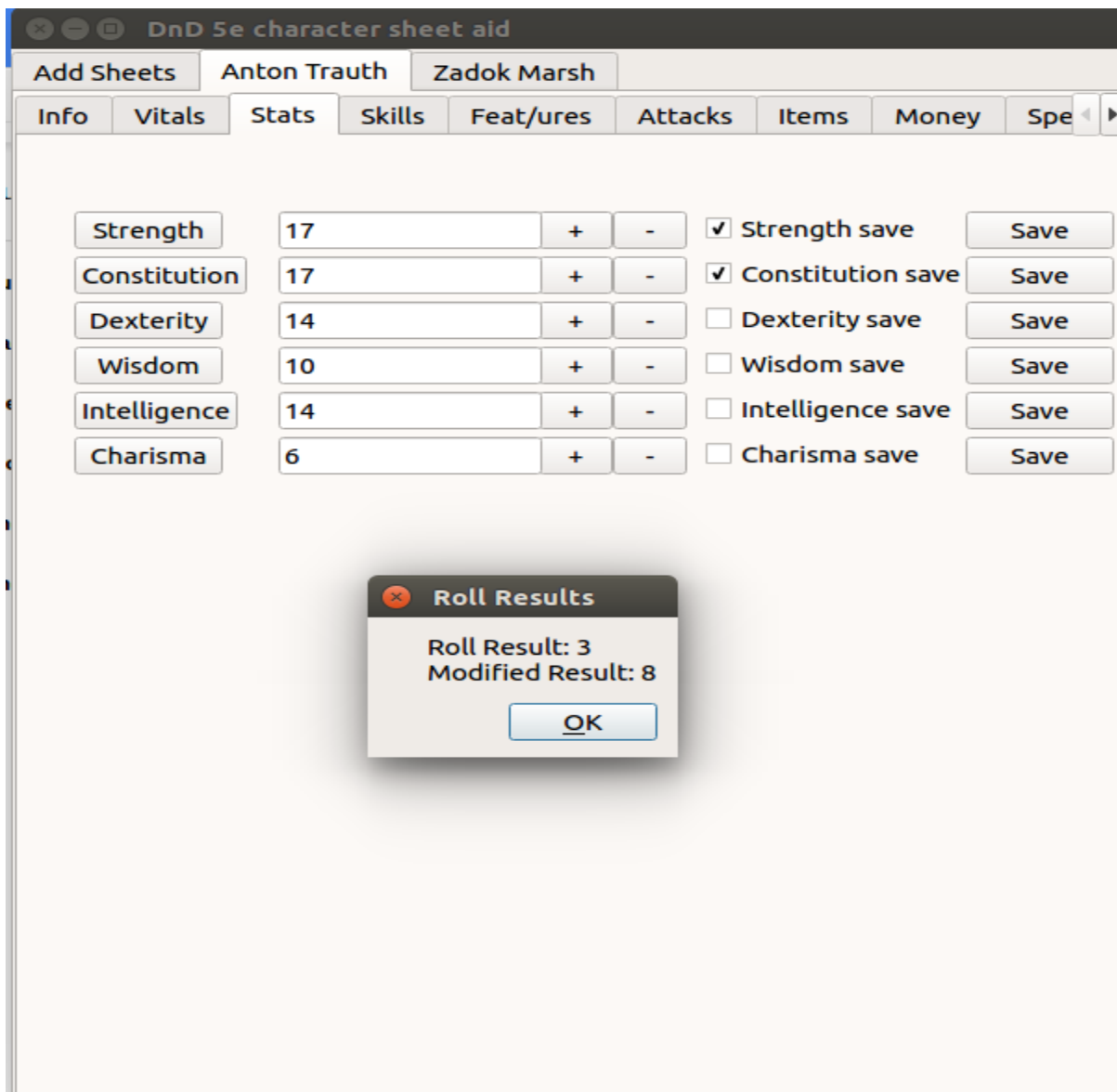




Figure 3: An automated roll. The popup dialog displays what the raw dice results were, as well as the modified result, which is based on a number of other pieces of saved data

To load characters from the data persistence layer, there is a starting tab on the upper tab level named “Add Sheets”, it contains 2 buttons, one of which, “new” generates a new character with a given name and defaults it’s information, the other, “open”, opens a file dialog, defaulting to ../characters, to search for pickle bytestream files to open, both tab and dialog are shown in figure 4 below.

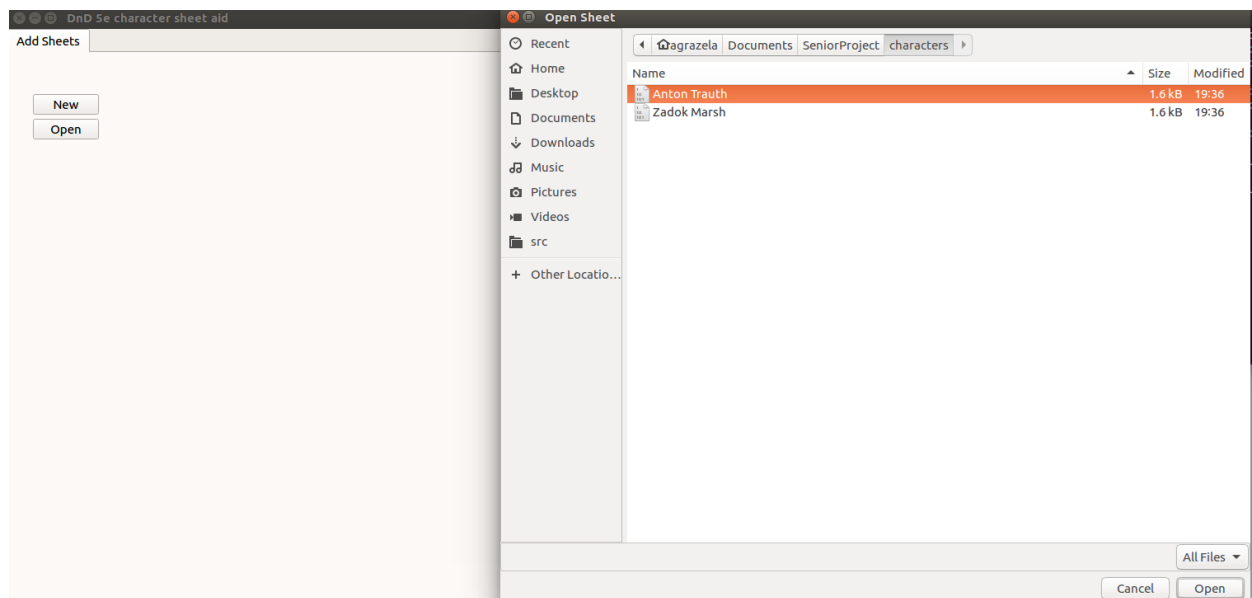
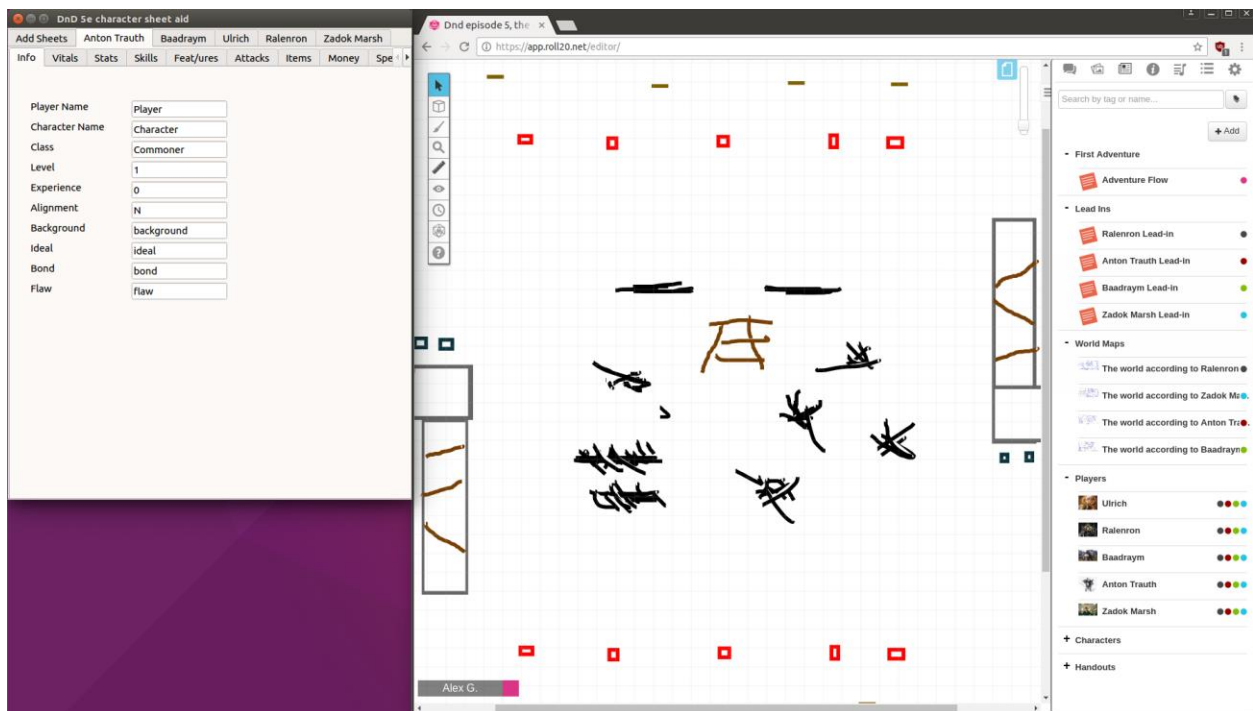


Figure 4: The Add Sheets is the default tab for the application, and it has 2 buttons for making new and opening old character sheets.

In order to save to the persistence layer, one need simply close the application.

## Conclusion/Future Work:

In conclusion, the objective of making an automated rolling, low screen space usage (see figure 5 below), multiple character sheet gui with the ability to save input information was a success. In order to improve upon it in the future, changes could be made to allow for multiple different tabletop rpg games to be used, as the only supported one currently is Dungeons and Dragons 5th edition<sup>4</sup>, which could be done with more modular code structures, or having multiple set types of gui for different game systems. It also could be tested and if needed, changed to work with the pyQt library available to raspbian, so that running it on a raspberry pi would be easy. This would make it much more portable in a physical sense, and make it easy to play anywhere anytime.



<sup>4</sup> <http://dnd.wizards.com/>

Figure 5: Screen utilization displaying all 5 player character sheets. Compare with figure 1, in which a single sheet was displayed

## References:

1. Roll20.net, an online tabletop rpg facilitator <https://roll20.net/>
2. Pickle module, a python data serializer <https://docs.python.org/3/library/pickle.html>
3. PyQt 5 documentation <http://pyqt.sourceforge.net/Docs/PyQt5/index.html>
4. Dungeons and Dragons website <http://dnd.wizards.com/>
5. Title Page Image <https://s-media-cache-ak0.pinimg.com/736x/ce/94/32/ce94322efc4aaf09ef83115b6c8b1626.jpg>