Tabletop Character Sheet Manager

# [Team Members]

* Jarrett Ruppert
* Jared Matthews
* Adam Stoltz

# [Project Description]

*Write out the description of your project. Describe what the project is about, the basics of what the product will do, and the audience for whom the product is intended.*

*Some example application types include:*

Tabletop Character Sheet Manager (TCSM) would, through a graphical user interface, allow the user to either (a) hand-pick and create a new character, (b) randomly generate, given generic race/level templates, a new character, and (c) edit/save character sheets to/from a file, for the purpose of tracking characters throughout a game of Pathfinder. TCSM would ensure that the rules of Pathfinder are not violated throughout the random generation or the manual generation.

# Part 1: Functional Requirments

*Below is an example set of functional requirements. Your team must fully define what your system will do. Language and terminology must be appropriate for laypeople, as this document is for the client.*

## Manual Character Creation

* User dictates values of every field that makes up a character.
* App will validate that user input follows rules of the game.
* App will provide a set of default values with the option to customize.

## Random Character Creation by Template

* Prompts for name, level, race (preset list of races), and class (preset list).
* After generation, displays character and allows for any modifications (within Pathfinder rules).

## Data Persistence and Display

* Display a GUI for all operations, including character display.
* Define a file template to allow for loading/saving .txt files to view/edit characters.
* On startup, prompt user for filepath to directory with desired character sheets (allow for folder creation if it doesn’t yet exist).

# Part 2: Objective Coverage

* File I/O
* Inheritance and polymorphism
* Interfaces and abstract classes
* Enums

# Part 3: Basic Design

UML

# Part 4: Project Plan

1. By Feb 24th, figure out game rules and logic needed to enforce it inside FileEditor.
2. Become competent in basic GUI’s with JavaFX – by Feb. 24th
3. By March 3rd, ensure FileGenerator and FileManager are functioning
4. By March 9th, finish touches on GUI and work out any remaining bugs