The Dungeoneers Champions’

Campaign Companion

A Dungeon Masters Toolset

Design Specification

Author: Cory Valentine

Reviewer: Josh Whitfield

Table of Contents

Table of Contents ii

Introduction 1

Purpose 1

Intended Audience and Reading Suggestions 1

Project Scope 1

Overall Description 1

Product Perspective 1

Product Features 2

User Classes and Characteristics 2

User Documentation 2

Assumptions and Dependencies 2

System Features 3

System Feature 1 3

System Feature 2 4

System Feature 3 4

External Interface Requirements 4

User Interfaces 4

Other Nonfunctional Requirements 4

Appendix A: Issues List 5

References 5

# Introduction

## Purpose

The Campaign Companion

The purpose of this product is to improve the workflow and productivity of a dungeon/game master, primarily for the Dungeons and Dragons Fifth Edition game however future scope would be to introduce other games and frameworks to the product and databases.

The scope of this product is relatively wide as I wish to produce an integrated suite of tools all in one application to assist the GM (Game Master)/DM (Dungeon Master), in all aspects of running the game, from creating an immersive atmosphere for their players to keeping informative notes on each campaign session and much more.

In sections below I will expand on the intended features of the product and the use cases that I wish to explore and improve.

## Intended Audience and Reading Suggestions

The intended audience of this document is primarily users and potential users (i.e. DM/GMs). However, anyone who is interested in this product is welcome to read as this will explain is copious detail the types of issues that this product will solve for the users and give a wide picture of the types of features that are intended to be included.

Developers who are interested in contributing to the product may find certain sections of this document useful and others less so.

## Project Scope

As mentioned above, the scope of this project is particularly wide as I have identified that the existing tools that contribute to solving the issues encountered by GMs are all separated and often come from different companies and suppliers meaning that the user usually has to have multiple applications open at once to run their campaigns efficiently.

I want to solve this problem by including all essential tools in one application, all the way from simple random tables for loot, encounters and names etc.to an interactive notebook that is integrated with an extensive library of information useful to be highlighted when written.

The main goal of the project is to make the DMs life easier and more efficient and allow them to provide a more fluent experience to their players.

# Overall Description

## Product Perspective

The idea for this product comes from the personal experience with existing solutions and having to tropes around the internet looking for a number of different tools to help run the game, it would end up meaning a web browser with 3 or 4 tabs open, a pdf of the campaign or a word document if it was a homebrew and then another word document still to keep track of each sessions events and to track stats and initiative etc. This to me seems so convoluted and unnecessary.

There are a few tools out there that offer payed solution that integrate a large number of the tools needed by DMs and GMs like D&D Beyond (wizards of the coast, 2019) which is officially licensed by the makers of D&D and Roll20.net (The Orr Group, LLC) which is a community owned solution. Both these toolsets attempt to negate the issues discussed above and both take different approaches but both also miss out on key features.

D&D Beyond focuses on the information and character creation side of things and the best thing about it is that it is owned by the Wizards of the Coast, and therefore they have the rights to all the player’s handbooks and the campaign guides, this means that you wouldn’t need go to another tool to integrate your purchased pdfs you can just do it straight through their service. However, the biggest downside of it is that is mainly an information bank and character creator.

Roll20 on the other hand focuses mainly on the playability side of things with an interactive map system with the ability to facilitate group calls and the ability to move your characters around the map and as the DM bring in and remove new assets on a hidden layer from the players. Roll20 also has a ‘compendium’ that contains the essential information which you can expand by purchasing wizards of the coast licensed information which matches up with the various player’s handbooks. This is probably the best integrated solution out there but there are still things missing.

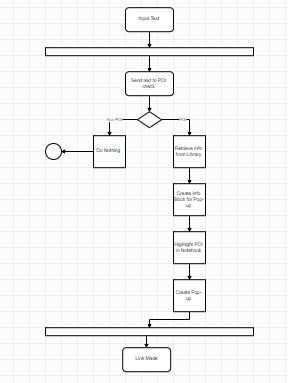
This is why with the solution I plan to develop I will include everything I as a DM feel would be beneficial to me running my campaigns and help to keep my delivery fluent and as natural as possible to my players.

With regards to the development process, I will be running an Agile scrum process, for a few reasons. The primary reason I will be running a scrum process is that it allows me to break down the process into small manageable chunks that are clearly defined from stories and tasks all the way up to epics and phase goals that then I can plan based on priority order what I will attempt to work on by a sprint by sprint basis and create the priorities based on a 6-month end of phase goal.

The secondary reason I will approach this project from a scrum technique is that it is the methodology that I am most familiar with and therefore I will work at my most efficient using this framework.

As an overall reason why Scrum is a better option than the alternatives like Kanban and other agile frameworks or even more so traditional Waterfall, Kanban can produce problems where things keep getting pushed to the bottom of the priority order and as new things come in they push their way to the top meaning that you are constantly re-prioritizing. Waterfall on the other hand has the opposite problem in that you are locked into a solid plan for as long as the contract exists.

ACTIVITY DIAGRAM



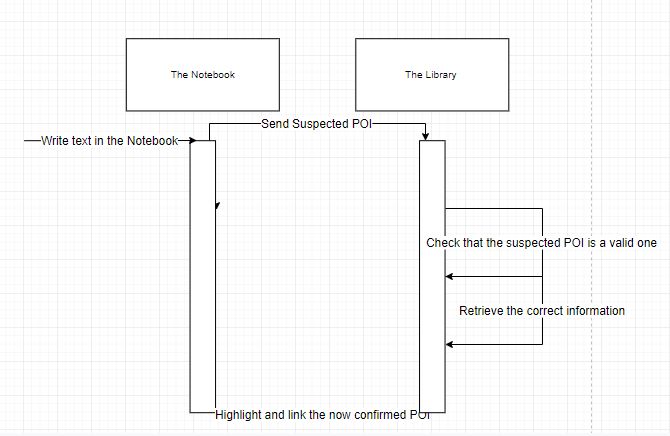
## Product Features

The overall toolset will contain 3 main suites for the DM to utilize, the first and largest in terms of how much work it will take to develop is the ‘Playing Floor’ which will consist of everything the DM needs to run the campaign on the day, random tables for loot, encounters and npc names etc. along with a stat and initiative tracker, dice roller, interactive notebook map generators and more, this will be the primary suite that will be loaded on startup.

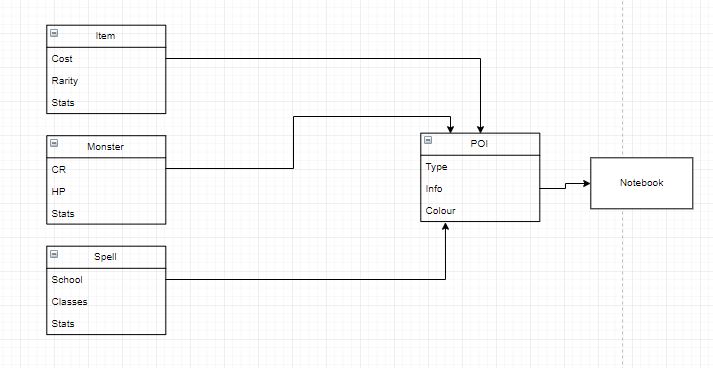
The second, and also second most taxing to develop will be the ‘Library’ this will be a comprehensive information bank of items, monsters, spells, effects, diseases and lots more. All these will be accessible either through its own section on the program or through the interactive notebook that when a word is written in it that matches with an object in the library it will be highlighted and when hovered over will produce a popup containing the key information like stats, rarity, challenge rating etc. that way the user doesn’t need to leave the playing floor at all.

The third and final suite being the one with the least work attached to it is the ‘Atmosphere’, this will be a large library of sound effects and background music to help the DM create immersion and atmosphere when they run their sessions.

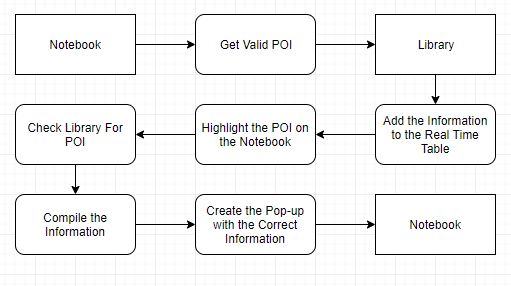
SEQUENCE DIAGRAM



CLASS DIAGRAM



DATA FLOW DIAGRAM



## User Classes and Characteristics

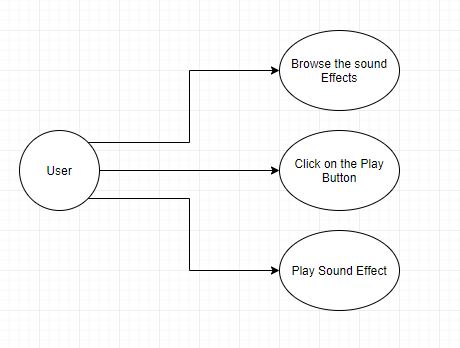
This product, as explained in previous sections is designed to be used by the DM to enhance their experience of playing D&D and by extension improve that of their players as well. In terms of a primary user class it would have to be the intermediate to advanced DMs who know how like to run the game but need a way to run it better and easier.

I envisage that they will open up the program and navigate to the playing floor where they will use the notebook to start introducing the session, they will then go to the atmosphere and play an appropriate background song and maybe a few sound effects to get the players immersed and their blood pumping. After that they will go back to the playing floor where they with have their notes set up for the session either through an imported pdf or note they wrote themselves in preparation for the session. Whilst running the session they will write note on what the players have done, who they interact with, where they go, they will use the random encounter generator to catch their players off guard, they’ll use the initiative tracker to run the encounters making use of the library for information on the monsters involved and the loot they can receive. Once they’ve finished up for the day they will save the progress from the session in the notebook, ready for the next one.

Although they are the assumed primary user class, this tool can be used by anyone who is involved in D&D from DMs running their first campaign to DMs that have been running the game since the 70s and even players could make use of this program by using it as a way of getting familiar with the mechanics of the game, they could even use it as their own way of recording the happening of each settings, or if the DM allowed they could use the library as a real library in game where the player would perhaps be looking for information on a mystical creature and they’ve been given access to great library by the king, although allowing the players access to the atmosphere might be a mistake on the DMs part.

However, there could be future scope to develop a player oriented version of the software that had tools in it tailored more to the needs of a player rather than a DM.

USE CASE DIAGRAM



## User Documentation

I plan on producing a wide range of resources for the users to access in order to gain full understanding of how the system works and the ways in which they can take full advantage. I will produce an extensive user manual which be provided in pdf format when acquiring the program, there will also be a wiki produced with a fully comprehensive list of features and examples on how to use them correctly. As a bonus action a video tutorial series will be produced for those users who are more visual and auditory learners.

## Assumptions and Dependencies

With regards to dependencies there are very few for the project as a whole, in fact the core functionality can be achieved without any outside interaction at all. However, when it comes to the optional features there are potential hang-ups. For example, if we wanted to make it so that you can import commercially available information from the Wizards of the Coast then we would need to have an arrangement made with them as to how the purchase of that information would be handled. Additionally, if we wanted to include commercially licensed music and sound effects in addition to the custom and royalty free ones we plan to include, an arrangement will need to be made with the artist of those songs or the producers of those effects.

# System Features

## Interactive Notebook

3.1.1 Description and Priority

This is one of the main features of the playing floor suite and the project as a whole, it is a notebook that is interactively linked to many of the other tools in the toolset like the library and the character sheets. The most important aspect of the notebook is that when a keyword is written in it, this could be the name of an item, a spell, a creature, or even a character and that word or phrase will be highlighted by the notebook and become a point of interest.

A point of interest will be colour coded and will when hovered over produce a pop-up that displays all the relevant/pertinent information related to it like the stat block if it’s a creature or character, or a brief description if it’s a location or deity, or the stats/price and rarity if it’s an item or spell. This will speed up the process of information delivery to the players and will also help the DM to stay immersed in combat or the story they are trying to tell by preventing them from having to navigate and find the information they are after.

Priority – Critical (9)

3.1.2 Stimulus/Response Sequences

As soon as the user inputs text into the notebook it will send that text through to an algorithm that will check each word or phrase against a bank of points of interest as to see if any action is to made.

If the algorithm decides that it isn’t a point of interest then it will do nothing, but if it is a point of interest, the algorithm will fetch the relevant information from the library or the character information centers and store that information in an active table so that it can be accessed quickly.

Once the information has been received the notebook will then highlight the point of interest in a colour that has been coded to relate to the type of point of interest it is i.e. Items will be highlighted one colour, Spells another, Monsters another and so on.

Once the point of interest is highlighted, whenever the user places the mouse over it, the information that was received earlier will be displayed in a pop-up box next to the mouse and will disappear when the mouse is moved away.

Test Cases –

* Points of interest should be highlighted when written
* Highlighted POIs should be colour coded correctly
* Text that doesn’t match a POI should not be highlighted
* When highlighted and moused over POIs should produce a pop-up
* Pop-ups should contain the correct information
* Both library stored and user created POIs should be treated equally

3.1.3 Functional Requirements

NB-001: The highlight, the system needs to be able to within a reasonable amount of time, ideally less that a second, highlight POIs within the text on the notebook preferably in real time without the additional action of a button pus or similar action to initiate it.

NB-002: The Pop-up, the system needs to be able to identify the information relating to the POI and display it on demand via a mouse over.

NB-003: The live table, in order for NB-002 to be implemented in a way that is efficient and provides the best user experience a live table will need to be created for the notebook where all the information relating to each POI that has been included in the open notebook should be stored allowing for quick access rather than having to search for it each time the mouse is hovered over the POI.

## The Random Generator

3.2.1 Description and Priority

The random generator is a feature that will be used by the DM on a regular basis to make their world seem more alive and unpredictable, and also to help them to think of the right types of loot to give to a party or the right number of enemies to place them against and so on. It’ll also help to keep the DM from having to think so hard on their toes with things like random town names/structures and random npc names.

This is a really important tool that will have a widespread number of use cases.

Priority – Highest (8)

3.2.2 Stimulus/Response Sequences

When the user wants something to be randomized they will go to the random generator tab and start to choose their options.

Once inside the RG tab they will be presented with a wide range of options arranged in tabs and drop down menus. If they choose the random loot tab they will be asked to provide a number of key points of information about the party, like the number of players, their levels, their classes etc. and also some information on the encounter itself like the number of enemies, what they were and what kind of environment the encounter occurred in. (optionally we may experiment with a feature that allows the user to import that information from the live notebook).

After filling in the information required (this goes for any of the tabs: loot, encounter, town, npc, plot hook etc.) the user will press the ‘generate’ button which will then send all the input information to an algorithm that will randomly select the required output and display it on the screen in a well-organized and easy to follow way.

It will then output a block into the interactive notebook to indicate that something has happened in the session.

It will be in the format of a message e.g.

‘Random – your players have encountered a group of goblins on the road

4x – Goblins

1x – Hobgoblin

1x – War Pig

Good luck!’

A different message will be displayed for each type of random event generated.

Test Cases –

* Button on the main page should take you to the RG page
* When on the RG page each of the different generator pages should be accessible
* Each of the generators should run correctly and generate the right things in a relevant way
* When the generators are run they should send the output to the notebook
* The notebook messages should be in the correct format and contain the correct information

3.2.3 Functional Requirements

RG-001: The tables, there will need to be a large number of tables in the back end that will hold the values that the generator will chose from based on the information provided by the user.

RG-002: The output, when the generator output it need to be random and not produce the same output each time its run with the same inputs.

RG-003: The message, when the generator outputs the message that is displayed in the notebook needs to reflect that of the output displayed on the RG page and needs to be in the correct format based on the type of generator used.

## The Dice Roller

3.3.1 Description and Priority

This feature as a whole is of quite high priority however having an animated dice tray is relatively low. So the feature itself is as simple as it sounds in that it is a number of basic random number generators that corresponds to a particular polyhedral dice and can be told how many times to ‘roll’ that particular dice and will then produce a number.

Priority – Higher (7) and Minor (3)

3.3.2 Stimulus/Response Sequences

The user will navigate to the DR tab where they will provide the number of times they wish each dice to be rolled and also any modifiers that need to be applied to the total.

Once the user has set up the roller as they wish they will click the ‘Roll Dice’ button which will then cause the random number generators to run and then add any modifiers they require and produce a number in total.

A breakdown of each number generated will be provided in a message box adjacent to the roller.

Test Cases –

* Check that modifiers are being applied to the dice rolls
* Check that the ranges of each dice are correct
* Check that the totals are being add up correctly
* Check that dice that have a 0 next to them are not run

3.3.3 Functional Requirements

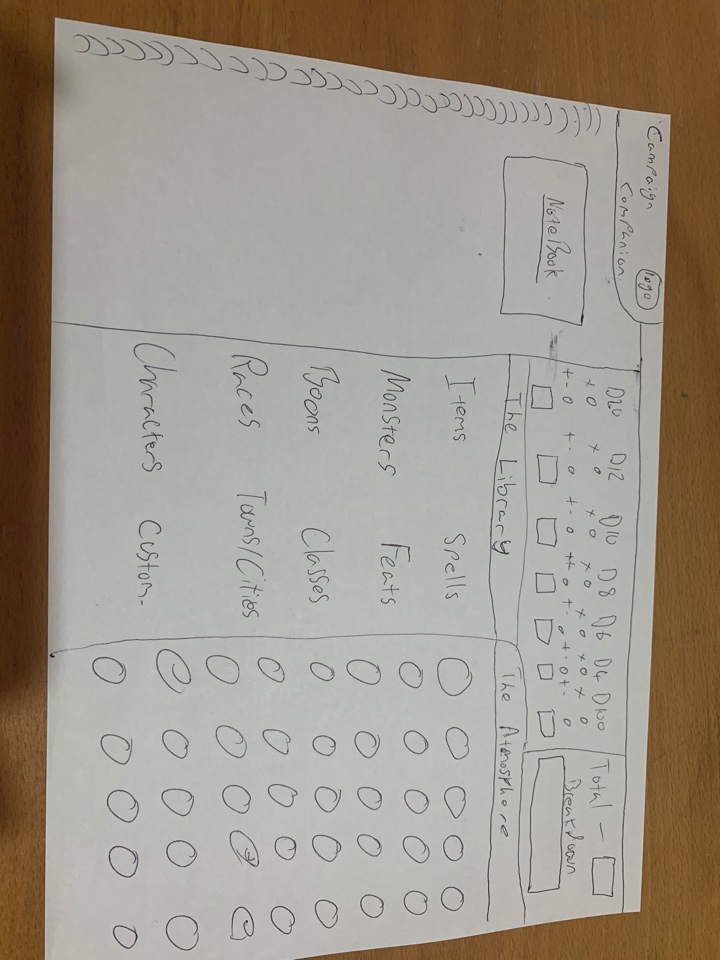
DR-001: The roll, when the user wants to roll some dice but doesn’t have any physically handy, they need to be able to rely on this tool to do just as good of a job.

DR-002: The modifier, when the user rolls the dice they need to be able to rely on the fact that the modifiers that they provide are going to be applied.

DR-003 (optional): The animation, an optional requirement for this feature is for an animated dice tray with the user specified dice being rolled to appear on screen when the ‘Roll Dice’ button is pressed.

# External Interface Requirements

## User Interfaces



With regards to user interface I want to create an immersive and stylish D&D environment for the user with lots of mystical and medieval influence in the colour scheme and visuals. The program will be designed with an efficient workflow in mind where everything the DM needs no more than three clicks away.

I’d like a tab based system to make it feel like the users are navigating a paper binder filled with all their favourite D&D notes and adventures. I’d also like to minimize the separate windows that are produced in the running of the program, therefore the tabs are the most appropriate alternative.

Above is a quick sketch of a possible way that a DM could have their workflow setup. Customization is the ideal goal, in that the DMs ability to customize their own workflow would help to make the user feel like this toolset was designed for them and with their needs in mind.

# Other Nonfunctional Requirements

Another requirement for this software to be viably developed is the addition of a save function on the notebook which allows the session to be saved and then recovered for the next one. This will not be as simple as saving a text file and opening it again due to the real time POI table that is associated with that session, so there are two possible solutions to this problem. The first being that when the session is reloaded it spends a few seconds re-configuring the real time table to contain the required POIs. The second is to find a way to save the tables contents with the contents of the notebook.

Appendix A: Issues List

1. Copyright problems with Wizards of the Coasts if I use the wrong information or too much information.
2. Copyright problems with the owners of ‘Royalty Free’ music and sound effects depending on how they have licensed it.
3. Compatibility issues between the front and back end depending the languages I use
4. The project time scale, depending on how difficult things become and how many people get involved in the project I project between 1 and 3 years’ development time from now to a finished and polished product.

# References

The Orr Group, LLC. (n.a.). *roll20.* Available: https://roll20.net/. Last accessed 03/05/19.

Wizards of the coast. (2019). *dnd beyond.* Available: https://www.dndbeyond.com/. Last accessed 03/05/19.