Requirements Specification:

Dungeons & Dragons 5E Character Tools

**Christopher Evans -** [**https://github.com/cevans88**](https://github.com/cevans88)

# Introduction

The introduction of the SRS should provide an overview of the entire SRS. It should contain the following subsections:

## Purpose

This subsection should:

1. Delineate the purpose of the SRS;
2. Specify the intended audience for the SRS.

## Scope

This subsection should:

1. Identify the software product(s) to be produced by name (e.g. Host DBMS, Report Generator, etc.);
2. Explain what the software product(s) will, and, if necessary, will not do;
3. Describe the application of the software being specified, including relevant benefits, objectives, and goals;
4. Be consistent with similar statements in higher-level specifications (e.g. the system requirements specification), if they exist.

## Definitions, acronyms, and abbreviations

This subsection should provide the definitions of all terms, acronyms, and abbreviations required to properly interpret the SRS. This information may be provided by reference to one or more appendixes in the SRS or by reference to other documents.

## References

This subsection should:

1. Provide a complete list of documents references elsewhere in the SRS;
2. Identify each document by title, report number (if applicable), date, and publishing organisation;
3. Specify the sources from which the references can be obtained.

This information may be provided by reference to an appendix or to another document.

## Overview

This subsection should:

1. Describe what the rest of the SRS contains;
2. Explain how the SRS is organised.

# Overall description

This section of the SRS should describe the general factors that affect the product and its requirements. This section does not state specific requirements. Instead, it provides a background for those requirements, which are defined in detail in Section 3 of the SRS, and makes them easier to understand.

This section usually consists of six subsections, as follows:

1. Product perspective;
2. Product functions;
3. User characteristics;
4. Constraints;
5. Assumptions and dependencies;
6. Apportion of requirements.

## Product perspective

This subsection of the SRS should put the product into perspective with other related products. If the product is independent and totally self-contained, it should be so stated here. If the SRS defines a product that is a component of a larger system, as frequently occurs, then this subsection should relate the requirements of that larger system to functionality of the software and should identify interfaces between that system and the software.

A block diagram showing the major components of the larger system, interconnections, and external interfaces can be helpful.

This subsection should also describe how the software operates inside various constraints. For example, these constraints could include:

1. System interfaces;
2. User interfaces;
3. Hardware interfaces;
4. Software interfaces;
5. Communications interfaces;
6. Memory;
7. Operations;
8. Site adaptation requirements.

### System interfaces

This should list each system interface and identify the functionality of the software to accomplish the system requirement and the interface description to match the system.

### User interfaces

This should specify the following:

1. *The logical characteristics of each interface between the software product and its users.* This includes those configuration characteristics (e.g. required screen formats, page or window layouts, content of any reports or menus, or availability of programmable function keys) necessary to accomplish the software requirements.
2. *All the aspects of optimising the interface with the person who must use the system.* This may simply comprise a list of do’s and don’ts on how the system will appear to the user. One example may be a requirement for the option of long or short error messages. Like all others, these requirements should be verifiable, e.g. “a clerk typist grade 4 can do function X in Z min after 1 h of training” rather than “a typist can do function X.” (This may also be specified in the Software System Attributes under a section titled Ease of Use.)

### Hardware interfaces

This should specify the logical characteristics of each interface between the software product and the hardware components of the system. This includes configuration characteristics (number of ports, instruction sets, etc.). It also covers such matters as what devices are to be supported, how they are to be supported, and protocols. For example, terminal support may specify full-screen support as opposed to line-by-line support.

### Software interfaces

This should specify the use of other required software products (e.g., a data management system, an operating system, or a mathematical package), and interfaces with other application systems (e.g. the linkage between an accounts receivable system and a general ledger system). For each required software product, the following should be provided:

* Name;
* Mnemonic;
* Specification number;
* Version number;
* Source;

For each interface, the following should be provided:

* Discussion of the purpose of the interfacing software as related to this software product.
* Definition of the interface in terms of message content and format. It is not necessary to detail any well-documented interface, but a reference to the document defining the interface is required.

### Communications interfaces

This should specify the various interfaces to communications such as local network protocols, etc.

### Memory constraints

This should specify any applicable characteristics and limits on primary and secondary memory.

### Operations

This should specify the normal and special operations required by the user such as:

1. The various modes of operations in the user organisation (e.g. user-initiated operations);
2. Periods of interactive operations and periods of unattended operations;
3. Data processing support functions;
4. Backup and recovery operations.

NOTE – This is sometimes specified as part of the User Interfaces section.

### Site adaptation requirements

This should:

1. Define the requirements for any data or initialisation sequences that are specific to a given site, mission, or operational mode (e.g., grid values, safety limits, etc.);
2. Specify the site or mission-related features that should be modified to adapt the software to a particular installation.

## Product functions

## User characteristics

## Constraints

## Assumptions and dependencies

# Specific requirements

# Appendixes

This may just be a placeholder to be honest, there’s probably a specific way of implementing an appendix using Word itself.

# Index

This may just be a placeholder to be honest, there’s probably a specific way of implementing an index using Word itself.