**[Product]**

**Software Requirements Specification**

*[SRS Version Number]*

*[SRS Version Date]*

**Project Team:** John Boyle, Colton Gerth, Brendan Hagan, Levi Terry

Standard Version Number: 3.1

Standard Version Date: August 2, 2012

Version History

|  |  |  |  |
| --- | --- | --- | --- |
| *Version* | *Date* | *Authors* | *Comment* |
|  |  |  |  |

TABLE OF CONTENTS

[1 Introduction 4](#_Toc331683624)

[1.1 Software Purpose and Scope 4](#_Toc331683625)

[1.2 Document Purpose and Contents 4](#_Toc331683626)

[1.3 Definitions, Acronyms, and Abbreviations 5](#_Toc331683627)

[1.3.1 Definitions 5](#_Toc331683628)

[1.3.2 Acronyms and Abbreviations 5](#_Toc331683629)

[1.3.3 Technical Definitions/Data Dictionary 5](#_Toc331683630)

[1.4 References 6](#_Toc331683631)

[2 General Factors 6](#_Toc331683632)

[2.1 Product Perspective 6](#_Toc331683633)

[2.2 Product Functions 6](#_Toc331683634)

[2.3 Environmental Conditions 6](#_Toc331683635)

[2.4 User Characteristic 7](#_Toc331683636)

[2.5 Dependencies 7](#_Toc331683637)

[2.6 Assumptions 7](#_Toc331683638)

[3 Analysis Use Cases 7](#_Toc331683639)

[4 Explanatory User Interfaces 7](#_Toc331683640)

[5 Specific Requirements 8](#_Toc331683641)

[5.1 Functional Requirements 8](#_Toc331683642)

[5.2 Non-Functional Requirements 8](#_Toc331683643)

[5.2.1 Design Constraints (DC) 8](#_Toc331683644)

[5.2.2 Human Factors (HF) 8](#_Toc331683645)

[5.2.3 External Interface Requirements (XI) 9](#_Toc331683646)

[5.2.4 Security (SC) 9](#_Toc331683647)

[5.2.5 Development Environment (DV) 9](#_Toc331683648)

[5.2.6 Standards (ST) 9](#_Toc331683649)

[5.2.7 Delivery Environment (DL) 9](#_Toc331683650)

[5.2.8 Performance (PR) 9](#_Toc331683651)

[5.2.9 Deliverable Items, Dates and Conditions (DD) 9](#_Toc331683652)

[5.2.10 Cost (CT) 9](#_Toc331683653)

[5.2.11 Quality (QL) 9](#_Toc331683654)

[5.2.12 V&V Activities (VV) 10](#_Toc331683655)

[5.2.13 Database (DB) 10](#_Toc331683656)

[5.2.14 Adaptability 10](#_Toc331683657)

[5.3 Requirements Models 10](#_Toc331683658)

[6 Illustrative Use Cases (IUC) 11](#_Toc331683659)

# Introduction

*[This Software Requirements Specification template is designed to facilitate the definition of processes and procedures relating to software requirements specification activities. This template was developed using IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications.*

Information displayed in brackets is explanatory. Delete the bracketed text items and add your project-specific input. These items are food for thought on the section they address.

*The introduction section should explain the purpose and scope of the project software requirements specification (SRS), as well as, provide clarification of definitions, acronyms, and references. This section should also provide an overview of this document.*

*Place any material here that is not specific to any of the sub-sections below.]*

## Software Purpose and Scope

*[This subsection should:*

1. *Identify the software products to be produced, by name*
2. *Explain what the software products will, and if necessary, will not do*
3. *Describe the application of the software being specified including all relevant goals, objectives, and benefits from producing the software.]*

## Document Purpose and Contents

*[This subsection should explain the purpose for writing an SRS for this project and describe the intended audience for the SRS. This subsection should describe the information that will be presented in each of the subsections from §2 on.]*

## Definitions, Acronyms, and Abbreviations

*[This subsection should provide the definitions of all terms, acronyms, and abbreviations required to fully understand your SRS.]*

### Definitions

|  |  |
| --- | --- |
| software failure | a failure will be attributed to this software product whenever one of the delivered work products does not meet the requirements specified in this SRS, or does not meet ordinary and reasonable customer/user expectations. |

### Acronyms and Abbreviations

|  |  |
| --- | --- |
| AD | Adaptability |
| AL | Availability |
| AUC | Analysis Use cases |
| CM | Communications |
| CT | Cost |
| DB | Database |
| DC | Design Constraint |
| DD | Delivery Data and Conditions |
| DL | Delivery Environment |
| DV | Development Environment |
| EN | Enhanceability/Extendibility |
| FE | Future Enhancements |
| HF | Human Factors |
| HW | Hardware |
| IUC | Illustrative Use Cases |
| ML | Maintainability |
| OP | Operations |
| PR | Performance |
| PT | Portability |
| QL | Quality |
| RL | Reliability |
| SC | Security |
| SDD | Software Design Description |
| SRS | Software Requirements Specification |
| ST | Standards |
| SI | Site |
| SW | Software |
| UB | Usability |
| VV | Verification & Validation |
| XI | External Interfaces |
| XXX | X of X of X |

### Technical Definitions/Data Dictionary

*[This subsection shall list (alphabetically) and briefly describe all data collections and items mentioned in this SRS. If a data base is not involved the technical items names in the requirements are defined here. These names should be chosen with care. The expectation is that these names will be used later in the design and implementation.]*

|  |  |  |
| --- | --- | --- |
| **ItemName** | **Type** | **Brief description of data item** |
| First item | Table | A description |

## References

*[This subsection should list all important references used within the SRS. If there are no pertinent references for this product that fact should be stated here.]*

# General Factors

*[The General Factors section should describe the general factors that affect the product and its requirements. Place any material here that is not specific to any of the sub-sections.*

*In this and each subsequent major section, briefly describe the purpose of this section from the readers perspective.]*

## Product Perspective

*[This subsection should put the product into perspective with other related products or projects. If the product to be produced from this SRS is totally independent, it should be clearly stated here. If the product to be produced from this SRS is part of a larger system, then this subsection should describe the functions of each component of the larger system or project and identify the interfaces between this product and the remainder of the system or project. This subsection should identify all principle external interfaces for this software product (Note: descriptions of the interfaces will be contained in another part of the SRS).]*

## Product Functions

*[This subsection should provide a summary of the functions to be performed by the software produced as a result of this SRS. Functions listed in this subsection should be organized in a way that will make it understandable to the intended audience of the SRS. (Note: this subsection is an overview, details of the specific requirements will be contained in section 4.)]*

## Environmental Conditions

*[This subsection should provide a summary of the environment in which the software must operate. (Note: this subsection is an overview, details of the specific requirements will be contained in section 4.)]*

## User Characteristic

*[This subsection should describe the general characteristics of the eventual users of the product that will affect the specific requirements. Eventual users of the product will include end-product customers, operators, maintainers, and systems people as appropriate. For any users that impact the requirements, characteristics such as education, skill level, and experience levels will be documented within this subsection as they impose constraints on the product.]*

## Dependencies

*[This subsection should list all external system dependencies on which the software resulting from the SRS will depend. This subsection should be the source for recognizing the impact of any changes to systems on the SRS and resulting software depends. This section can highlight unresolved requirement issues that should be recorded on the Project Manager’s Open Issues List.]*

## Assumptions

*[This subsection should list all assumptions that on which the software resulting from the SRS will depend that have not been covered above. This subsection should be the source for recognizing the impact of any changes to these assumptions on the SRS and resulting software.. This section can highlight unresolved requirement issues that should be recorded on the Project Manager’s Open Issues List.]*

# Analysis Use Cases

Analysis Use Cases were not developed for this specification.

# Explanatory User Interfaces

*[This is an optional section that is used when providing the user with information that could be helpful in understanding the specific requirements in the next section.*

*If this section is used, care must be taken that the general descriptions given here are not presented as requirements.]*

# Specific Requirements

*[The Specific Requirements section should contain all the requirements for the subject software. The details within this section should be defined as individual, specific requirements. Each specific requirement should be stated such that its achievement can be objectively verified by observation, inspection, usability testing, functional testing, analysis, or a combination of these. The method verification must be described. Each requirement should be clearly identified for tracking.]*

## Functional Requirements

*[This subsection should specify how the software product will react to every possible input situation. It describes all the actions that must take place in the software in response to every input. Pertinent changes in the environment are considered to be inputs.*

*Care must be taken to avoid dropping into design details. In the user cannot directly experience the effect of a requirement it probably crossed the line into design.*

*Functional requirements should be logically grouped. Each group should have a short, unique (within the SRS) abbreviation and a number. The word processing section number will probably change as the SRS is developed.*

*For each identified requirement an optional rationale for that requirement may be given.*

*Most modern software should provide at least a modicum of user help. For very complex applications in situ help may be supplemented by a user’s manual (or manual page) but for many simple applications comprehensive in situ help is sufficient.]*

## Non-Functional Requirements

*[This subsection should specify both the static and dynamic numerical requirements placed on the software or human interaction with the software. All the identifiers for requirements in this section should begin with the two letter abbreviation shown below]*

### Design Constraints (DC)

There are no design constraints for the program.

### External Interface Requirements (XI)

#### Hardware (HW)

#### Software (SW)

#### Communications (CM)

### Development Environment (DV)

### Standards (ST)

### Delivery Environment (DL)

#### Site (SI)

*[This subsection should specify any requirements for installation or operation of the software that might change the pre-existing configuration of the user site.]*

#### Operations (OP)

*[This subsection should specify normal and special operations required by the user to include:*

* *Various modes of operation within the user organization*
* *Periods of interactive operations and unattended operations*
* *Data processing support functions*
* *Backup and recovery operation.]*

### Performance (PR)

### Deliverable Items, Dates, and Conditions (DD)

### Quality (QL)

#### Reliability (RL)

*[Reliability is specified as mean-time-to failure of an operational item. An operational profile must be specified.]*

#### Availability (AL)

#### Maintainability (ML)

*[Failures can be classified as occurring in either operational or non- operational delivered items Failures in operational items can be classified by the work products that must be changed to eliminate that failure: code only, code and design, code, possibly design, and requirements. For each class of failure what is the maximum estimated effort required to eliminate that failure and what is the rationale for this estimate.]*

#### Usability (UB)

#### Enhanceability/Extendibility (EN)

*[If the future it might be necessary to change the Functional requirements in specified ways, what is the maximum estimated effort required to make such changes and what is the rationale for this estimate?]*

#### Portability (PT)

*[If in the future it might be necessary to change the above Development or Delivery Environments (DV or DL) to other specified environments, what is the maximum estimated effort required to implement such changes and what is the rationale for this estimate]*

### V&V Activities (VV)

### Adaptability (AD)

*[If it is specified that in the future it might be necessary to change any of the above Non-Functional requirements, what is the maximum estimated effort required to implement such changes and what is the rationale for this estimate.]*

## Requirements Models

*[This optional subsection, if present, provides models of the functional requirements to aid in clarifying and validating these requirements. A Z language specification is a good example. This sub-section may be skipped entirely if this SRS does not use any requirements models.]*