**System Requirements Specification**

**for**

***Author(s)***

***Date:***

*Document Version Control Information*

Acceptance by User Group: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Table of Contents**

**1. Introduction**

* 1. **Purpose of this document**  
     Describes the purpose of the document, and the intended audience.
  2. **Scope of this acquisition project**  
     Describes the scope of this requirements definition effort. Introduces the requirements elicitation team, including users, customers, system engineers, and developers. This section also details any constraints that were placed upon the requirements elicitation process, such as schedules, costs, or the software engineering environment used to develop requirements.
  3. **Overview**  
     Provides a brief overview of the product defined as a result of the requirements elicitation process.
  4. **Business Context**  
     Provides an overview of the business organization sponsoring the development of this product. This overview should include the business's mission statement and its organizational objectives or goals.
  5. **Definitions, Acronyms, Abbreviations**

Provides definitions of unfamiliar definitions, terms, and acronyms.

**2. General Description**

* 1. **User Problem Statement**  
     This section describes the essential problem(s) currently confronted by the user group.
  2. **User Objectives**  
     This section describes the set of objectives and requirements for the system from the user's perspective. It may include a "wish list" of desirable characteristics, along with more feasible solutions that are in line with the business objectives.
  3. **User Characteristics**  
     Describes the features of the user community, including their expected expertise with software systems and the application domain.
  4. **Similar System Information**  
     Describes the relationship of this solution with any other installed solutions (if any). Specifies if this solution is intended to be stand-alone, or else used as an integrated component of a solution product. If the latter, this section discusses the relationship of this product to the larger product.
  5. **General Constraints**  
     Lists general constraints placed upon the design team, including speed requirements, industry protocols, hardware platforms, and so forth.

**3. Software Functional Requirements**

This section lists the functional requirements within each of the recommended applications in ranked order. Functional requirements describe the possible effects of a particular software system (such as an accounting system), in other words, *what* the system must accomplish. Other kinds of requirements (such as interface requirements, performance requirements, or reliability requirements) describe *how* the system accomplishes its functional requirements.

Functional requirements will be described in terms of ‘must’, ‘should’, ‘could’, or ‘ideally could’.

Each functional requirement should be specified in a format similar to the following:

**3.1 Functional Requirements of: <application 1>.**

**3.1.1 Scope requirement for <application 1>**

*What is impacted by this application? Describe how this application will address the needs identified. Describe interactions with other requirements.*

**3.1.2 <Functional requirements within application 1)>**

“The application **must** ….”

*Describe what top level functions or modules the application must or should have*

<name of function or module>

3.1.2.1 “The … module must (or ‘should’, etc) be able to …”

…

Second-level (functions within the module):

3.1.2.1.1 “This function must (or ‘should’, etc)…”

3.1.2.1.1 “This function must (or ‘should’, etc)…”

3.1.2.2 “This module must (or ‘should’, etc)…”

…

Second-level (functions within the module):

3.1.2.2.1 “This function must (or ‘should’, etc)…”

3.1.2.2.1 “This function must (or ‘should’, etc)…”

**3.2 Functional Requirements of: <application 2>**

**3.2.1 <Scope requirement for application 2>**

What is impacted by this application? Describe how this application will address the needs identified. Describe interactions with other requirements.

**3.2.2 <Functional requirements within application 2)>**

“The application **must** ….”

*Describe what top level functions or modules the application must or should have*

<name of function or module>

3.2.2.1 “The … module must (or ‘should’, etc) be able to …”

…

NORMALLY YOU WOULD SPECIFY SECOND LEVEL FUNCTIONAL REQUIREMENTS FOR APPPLICAITON 2 (ETC) HOWEVER TO KEEP THINGS BRIEF THIS IS NOT NECESSARY FOR THE ASSIGNMENT UNLESS YOU HAVE BEEN TOLD OTHERWISE

3.2.2.2 “This module must (or ‘should’, etc)…”

…

**3.3 Functional Requirements of: <Selected application>**

**3.3.1 Scope requirement for <Selected application>**

What is impacted by this application? Describe how this application will address the needs identified. Describe interactions with other requirements.

*Here describe the application that you have chosen and supported by the Software Evaluation Matrix included as an Appendix.*

1. **Interface Requirements**

This section describes how the software interfaces with other existing software products or users for input or output.

* 1. **User Interfaces**  
      Describes how this product interfaces with the user.
     1. **GUI**  
        Describes the required interface (GUI or Menu Driven). This section should include a examples of screen layouts or mockups to illustrate user interface features.

If the system is menu-driven, a description of all menus and their components should be provided.

**4.1.2 Ease of Use**

The (minimisation of) effort, difficulty or strain involved in using the system

**4.1.3 Task match**

The degree to which services from the system must match the task needs of the user

**4.1.4 User support**

Assessing the level of help and support necessary, when and where it will be needed, and in the form it is required by the user

**4.1.5 Perceived consequences**

The range of organisational and job related aspects that will be affected by the computer system

**5. Platform Requirements**  
*Describes interfaces to and requirements for hardware devices.*

### **5.1 Hardware**: including peripherals (screens, printers, etc)

### 

### **5.2 Estimated capacities** (CPU speed, memory size, online & offline storage).

**5.3 Operating System and Database Requirements**

### **5.4 Networking, Internet, Communications Requirements**

### **5.5 Data Backup, Recovery**

### **5.6 Security (Virus Protection, Firewall)**

**6. Data Conversion**

The plan for conversion of data from manual records to electronic data, and/or from previous computer system to the new system.

**7. Other non-functional Requirements**

Specifies any other particular non-functional attributes required by the system. Examples are provided below.

**7.1 Security**

**7.2 Reliability**

**7.3 Maintainability**

**7.4 Portability**

**7.6 Application Compatibility**

**7.7 Serviceability**

**7.8 others as appropriate**

**Note: You need NOT complete this section – it is for your information only. Simply remove it.**

**8. Operational Scenarios**

This section should describe a set of scenarios that illustrate, from the user's perspective, what will be experienced when utilizing the system under various situations as follows:

In the broad sense, a scenario is simply a proposed specific use of the system. More specifically, a scenario is a description of one or more end-to-end transactions involving the required system and its environment. Scenarios can be documented in different ways, depending up on the level of detail needed. The simplest form is a use case, which consists merely of a short description with a number attached. More detailed forms are called scripts. These are usually represented as tables or diagrams and involved identifying an action and the agent (doer) of the action. For this reason, a script can also be called an action table.

Although scenarios are useful in acquiring and validating requirements, they are not themselves requirements, because they describe the system's behaviour only in specific situations; a specification, on the other hand, describes what the system should do in general.

**Note: You need NOT complete this section – it is for your information only. Simply remove it.**

**8. Preliminary Schedule**

This section provides an initial version of the project plan, including the major tasks to be accomplished, their interdependencies, and their tentative start/stop dates

**Note: Use a Gantt Chart**

**9. Preliminary Budget**

This section provides an initial budget for the project, itemized by cost factor.

**Note: Keep this section simple, just a cost guesstimate by major component (software, hardware, network, support, training).**

**References**  
Provides complete citations to all documents and meetings referenced or used in the preparation of this document.

**Appendices**

Specifies other useful information for understanding the requirements. All SRS documents should include at least the following two appendices:

**A: Recommended application supporting information/ documents**

**B: Minutes of JAD meetings**