

System Requirements Specification

Instructions

Version 1.2 • 14 JAN 2008



Version History

This and other Framework Extension tools are available on the Framework Web site.

Release Date	Description
14-Jan-2008	Version 1.2 released. Modified "Using this Template" section of the Template and italicized all section instructions to align with the Framework and Change Request (CR) #34. CR #34 was recommended by the Framework Change Advisory Board (CAB) and approved by DIR.
13-Mar-2007	Version 1.1 released. Made minor modifications to indicate Framework Extension.
30-Jun-2006	Version 1.0 – Instructions and Template Released.

Contents

Introduction	1
Use of the System Requirements Specification	1
Section 1. Introduction	2
1.1 Purpose.....	2
1.2 Business Context	2
1.3 Scope	2
1.4 User Characteristics.....	2
Section 2. General System Description	2
2.1 System Context.....	2
2.2 System Modes and States	3
2.3 Major System Capabilities	3
2.4 Major System Conditions	3
2.5 Major System Constraints.....	3
2.6 Assumptions.....	3
2.7 Dependencies	3
2.8 Operational Scenarios	3
Section 3. System Capabilities, Conditions, and Constraints	4
3.1 Business Requirements	4
3.2 Functional Requirements	4
3.3 Physical Requirements	6
3.4 Logical Data Requirements.....	6
3.5 User Requirements	7
3.6 Information Management Requirements	7
3.7 Systems Requirements.....	7
3.8 Policy and Regulation Requirements.....	9
3.9 System Life Cycle Sustainment Requirements.....	9
Section 4 System Interfaces	10
Section 5 Requirements Traceability Matrix	10
Section 6. References.....	11
Section 7. Glossary	11
Section 8. Revision History	11
Section 9. Appendices	11
Appendix A. Sample Requirements Traceability Matrix	12

Introduction

The System Requirements Specification (SyRS) Template and Template Instructions are included within the System Development Life Cycle (SDLC) Extension of the Texas Project Delivery Framework (Framework) to establish a consistent method for documenting system requirements for technology projects. System requirements describe what the system's sponsor expects the system to do, the system's expected environment, the system's usage profile, its performance parameters, and its expected quality and effectiveness. The SyRS is a structured collection of information that embodies the requirements of the system. The SyRS completely describes all inputs, outputs, and required relationships between inputs and outputs.

Documenting system requirements can reduce project risk by reducing uncertainty in implementation of the system. Documentation of detailed and accurate system requirements contributes to the success of IT systems by establishing and communicating expectations for all aspects of the system's features and performance. In addition, the documented system requirements provide the basis for ensuring that requirements are addressed during system design and testing.

Use of the System Requirements Specification

Within the Framework, the System Requirements Specification (SyRS) is completed, reviewed, and approved in the Project Planning Review Gate. The SyRS documents and communicates the requirements of the customer to the technical community who will specify and build the system. The collection of requirements in the SyRS acts as the bridge between the two groups and must be understandable by both the customer and the technical community.

Approval of the System Requirements Specification constitutes agreement that the system satisfying the specifications within will be accepted. Once approved, changes can be made to the specifications in the SyRS only through the change management process.

The SyRS Template should be used to develop a SyRS for each project. The SyRS should be developed in coordination with and be accessible by appropriate project team and stakeholder entities. In addition, all information in the SyRS should be consistent with the Project Plan and the related project documents. Documented system requirements should be addressed in all appropriate project requirements, design, and testing documentation throughout the life of the project.

Note: The System Requirements Specification should contain descriptive labels for and references to every figure, table, and diagram included within the document.

Section 1. Introduction

In the following subsections, provide an overview of the entire System Requirements Specification (SyRS). This section should stand alone as an executive summary.

1.1 Purpose

Specify the purpose the SyRS and its intended audience.

1.2 Business Context

Provide an overview of the business organization sponsoring the development of the system, including the mission statement and organizational objectives of the business unit.

1.3 Scope

Describe the scope of the system to be produced. Within the description, include:

- The identity of the system
- A brief description of the system's functionality
- An explanation of what the system will and will not do
- A description of the application of the system
- A description of the relevant benefits, objectives, and goals of the system

This description should be consistent with similar statements in preceding project documents.

1.4 User Characteristics

Identify each type of user of the system by function, location, and type of device. Specify the number of users in each group and the nature of their use of the system. Characteristics of the users of the system product will affect the specific requirements. Many people interact with a system during the operation and maintenance phases of the life cycle of the system. Some are users, operators, and maintenance and systems personnel. Certain characteristics of these people, such as educational level, experience, and technical expertise may impose important constraints on the system's operating environment.

Section 2. General System Description

2.1 System Context

Provide appropriate diagrams and accompanying narratives to provide an overview of the context of the system, defining all significant interfaces crossing the system's boundaries.

2.2 System Modes and States

Describe the system's various modes of operation (e.g., administrative, batch, debug, embedded training capability, fully operational, online, idle offline, or safe), and the conditions (e.g., environmental, configuration, or operational) that determine the modes of operation.

2.3 Major System Capabilities

Provide diagrams and accompanying narratives to depict major capability groupings of the requirements.

2.4 Major System Conditions

Describe the major system conditions and their associated capabilities. Conditions may limit the options available to a designer. It is important to identify conditions as attributes of capabilities, not as primary capabilities, to ensure that the requirements clearly define the need without imposing unnecessary bounds on the solution.

2.5 Major System Constraints

Describe the major system constraints. Constraints limit the scope and functionality of the system including, but not limited to, regulatory policies, infrastructure limitations, resources, and licensing.

Constraints are requirements that are imposed on the solution by circumstance, force, or compulsion. They limit, absolutely, the options available to a designer of a solution by imposing immovable boundaries and limits.

2.6 Assumptions

Describe the assumptions that can affect the requirements specified in the SyRS. Assumptions are factors that are believed to be true during the life cycle of the project that, if changed, may affect the project outcomes negatively including, but not limited to, end-user characteristics, known technology infrastructure, resource availability, and funding availability.

2.7 Dependencies

Describe the dependencies that can affect the requirements specified in the SyRS. Dependencies are outside of project scope and control and must remain true for the project to succeed.

2.8 Operational Scenarios

Provide descriptive operational scenarios for the system. An operational scenario is a step-by-step description of how the system should operate and interact with its users and its external interfaces under a given set of circumstances. Scenarios may also be used to describe what the system should not do.

Section 3. System Capabilities, Conditions, and Constraints

In the following subsections, specify all the system requirements to a level of detail sufficient to enable developers to specify and build the system. Every stated requirement should be understandable by users, developers, operators, and external systems staff.

Specify at a minimum the transformation of inputs of the system into outputs of the system product and all functions performed by the system in response to an input or in support of an output. This description may consist of a model of the requirements. For example, the model may contain data flow diagrams, entity relationship diagrams, Product Network Diagram, and a data dictionary.

Requirements should be:

- Correct, unambiguous, complete, consistent, ranked for importance and/or stability, verifiable, modifiable, and traceable
- Cross-referenced to earlier documents that relate
- Uniquely identifiable
- Organized for maximum readability

Note: Each requirement documented in this section must have a unique identifier for requirements traceability and should be ranked for importance and/or stability.

3.1 Business Requirements

Describe all business requirements for the system. Business requirements are the parts of the fully defined business process that will be automated by the system. Business requirements may be defined as use cases.

3.2 Functional Requirements

Customize this subsection to contain the subsections necessary to comprehensively define the fundamental actions that must take place within the system to accept and process the inputs and to process and generate the outputs.

Fundamental actions in functional requirements include:

- Validity checks on the inputs
- Exact sequence of operations
- Responses to abnormal situations, including overflow and communication facilities
- Effect of parameters

- Relationship of outputs to inputs, including
 - Input/output sequences
 - Formulas for input to output conversion
 - Definitions of the responses of the system to all realizable classes of input data in all realizable classes of situations

Functional requirements should include specific requirements for business rules. Business rules describe and document the steps in a business process.

It may be appropriate to partition the functional requirements into subfunctions or subprocesses. This does not imply that the architecture or system design will also be partitioned in that way.

The two common means of specifying functional requirements are functional decomposition and use cases. Subsection templates for each of the means of specifying functional requirements are provided below.

3.2.XF Function X

When functional decomposition is used as the means of specifying the functional requirements, provide a 3.2.xf subsection for each function. Each 3.2.xf subsection should be labeled and titled appropriately for a specific function, where *xf* is the appropriate sequential subsection number and *X* is the name of the specific function.

3.2.xf.1 Function X Purpose

Describe the intent of the function.

3.2.xf.2 Function X Inputs

Describe the inputs to the function, including sources, valid ranges of values, timing considerations, operator requirements, and special interfaces.

3.2.xf.3 Function X Operations

Describe the operations to be performed within the function, including validity checks, responses to abnormal conditions, and types of processing required.

3.2.xf.4 Function X Outputs

Describe the outputs from the function, including output destinations, valid ranges of values, timing considerations, and considerations for handling of illegal values, error messages, and interfaces required.

3.2.XU Use Case Y

When use cases are used as the means of specifying the functional requirements, provide a 3.2.xu subsection for each use case. Each 3.2.xu subsection should be labeled and titled

appropriately for a specific use case, where *xu* is the appropriate sequential subsection number and *Y* is the name of the specific use case.

Within each use case subsection, specify the use case information, including the actor, pre-conditions, post-conditions, scenarios, and alternate scenarios.

3.3 Physical Requirements

3.3.1 Construction

Specify the environmental (e.g., mechanical, electrical, chemical) characteristics of where the system will be installed. For example, the weight limits of the system, dimensional and volume limitations, operator station layout, and access for maintenance should be specified.

3.3.2 Durability

Specify the durability characteristics of the system.

3.3.3 Adaptability

Specify the growth, expansion, capability, and contraction characteristics. For example, if the system will require future network bandwidth, the hardware rack should be specified with extra slots to accommodate new network cards, as demand increases.

3.3.4 Environmental Conditions

Specify the environmental conditions to be encountered by the system. The following subjects should be considered for coverage: natural environment (e.g., wind, rain, and temperature), induced environment (e.g., motion, shock, noise), and electromagnetic signal environment.

3.4 Logical Data Requirements

Describe the logical data requirements for the system. Logical data requirements may include:

- Types of information used by various functions
- Frequency of use
- Accessing capabilities
- Data entities and their relationships
- Integrity constraints
- Data retention requirements

3.5 User Requirements

Describe the user requirements for the system. User requirements capture the intended behavior of the human interface for the application. For example, if the user operates through a display terminal, specify the required screen content, content of any reports or menus, and relative timing of inputs and outputs. User requirements may include example screen or report formats, as prototypes to illustrate requirements.

3.6 Information Management Requirements

Describe the requirements for managing the creation, capture, organization, maintenance, use, protection, and disposition of information in accordance with applicable laws, regulations, policies, and standards.

3.7 Systems Requirements

3.7.1 Performance Requirements

Describe the performance conditions and their associated capabilities. Include such considerations as:

- Dynamic actions or changes that occur (e.g., rates, velocities, movements, and noise levels)
- Quantitative criteria covering endurance capabilities of the equipment required to meet the user needs under stipulated environmental and other conditions, including minimum total life expectancy. Indicate required operational session duration and planned utilization rate.
- Performance requirements for the operational phases and modes
- The number of terminals to be supported
- The number of simultaneous users to be supported
- The numbers of transactions and tasks and the amount of data to be processed within certain time periods for both normal and peak workload conditions
- Acceptable performance under atypical stress

State these requirements in measurable terms. For example, 95% of the transactions shall be processed in less than one second, rather than, operator shall not have to wait for the transaction to complete.

Performance characteristics unique to a specific function (see Functional Requirements subsection) and outside the general performance characteristics of the system should be specified as part of the processing description of that function.

3.7.2 Quality Requirements

Describe requirements for the quality characteristics of the system. Specify the requirements in measurable and verifiable terms. Describe any trade-offs between the characteristics (e.g., security versus portability). Definitions of the quality characteristics include:

- Correctness – extent to which program satisfies specifications and fulfills user’s mission objectives
- Efficiency – amount of computing resources and code required to perform function
- Flexibility – effort needed to modify operational program
- Integrity/security – extent to which access to software or data by unauthorized people can be controlled. Security requirements relate to both the facility that houses the system and operational security requirements. Examples of security requirements might be to specify the security and privacy requirements, including access limitations to the system, such as existence of log-on procedures and passwords, and of data protection and recovery methods. This could include the factors that would protect the system from accidental or malicious access, use, modification, destruction, or disclosure.
- Interoperability – effort needed to couple one system to another
- Maintainability – effort required to locate and correct an error during operation. Maintainability requirements apply to maintenance in the planned maintenance and support environments. Examples of quantitative maintainability requirements are time (e.g., mean and maximum downtime, reaction time, turnaround time, mean and maximum times to repair, mean time between maintenance actions), rate (e.g., maintenance staff hours per specific maintenance action, operational ready rate, maintenance time per operating hour, frequency of preventative maintenance), maintenance complexity (e.g., number of people and skill levels, variety of support equipment), and maintenance action indices (e.g., maintenance costs per operating hour, staff hours per overhaul).
- Portability – effort needed to transfer from one hardware or software environment to another
- Reliability – extent to which the system performs with required precision
- Reusability – extent to which software and associated artifacts can be reused in another application
- Testability – effort needed to test to ensure that the software performs as intended
- Usability – effort required to learn, operate, prepare input, and interpret output. Usability requirements include any special or unique requirements that identify and define human-factor considerations and constraints (e.g., design space limits, climatic limits, eye movement, reach, ergonomics, cognitive limits, and usability) that affect operation of the system. Examples include those specific areas, stations, or equipment that require

concentrated human engineering attention due to the sensitivity of the operation or criticality of the task (e.g., those areas where the effects of human error would be particularly serious).

3.8 Policy and Regulation Requirements

Specify relevant applicable laws, regulations, policies, and standards that will affect the operation or performance of the system, as well as any relevant external regulatory requirements, or constraints imposed by normal business practices.

Examples of policy and regulation requirements include:

- Multilingual support
- Labor policies
- Protection of personnel information
- Reports to a regulatory agency
- Health and safety criteria, including those basic to the design of the system, with respect to equipment characteristics, methods of operation, and environmental influences

3.9 System Life Cycle Sustainment Requirements

Describe the review, measurement collection, and analysis quality activities to be executed during the life cycle of the system.

The capacity to change or enhance the product and life cycle processes can be designed into the system architecture to enable the cost-effective sustainment of the system throughout its life cycle. This design attribute should be established early in the system's development to provide a basis for planning each incremental development effort. Evolutionary development strategies should address approaches for managing the introduction of new technologies, evolving requirements, or enhancing product capabilities.

The life cycle of the system includes:

- Development
- Testing
- Manufacturing
- Distribution
- Operation
- Support
- Training

- Disposition

Section 4. System Interfaces

Specify in detail, requirements for interfaces among different components and their external capabilities, including all its users, both human and other systems. The characteristics of interfaces to systems under development, or future systems, should also be included. Any interdependencies or constraints associated with the interfaces should also be identified (e.g., communication protocols, special devices, standards, fixed formats). Each interface may represent a bidirectional flow of information. A graphic representation of the interfaces should be used when appropriate for the sake of clarity.

Note: Each interface requirement must have a unique identifier for requirements traceability and should be ranked for importance and/or stability.

Section 5. Requirements Traceability Matrix

In this section, provide a reference to the location of or provide the actual Requirements Traceability Matrix (RTM) that will be completed during the life of the project.

The RTM is initiated in the SyRS and is updated appropriately during the life of the project to indicate traceability to the design elements documented in the System Design Description, the software requirements documented in the Software Requirements Specification, and the design elements documented in the Software Design Description. The completed RTM assures that every requirement has been addressed in the design and that every design element addresses a requirement. The RTM also provides the necessary traceability for integration, acceptance, regression, and performance testing.

The Requirements Traceability Matrix in the SyRS should:

- Contain the columns that will be used to illustrate traceability of the system requirements to the system design elements, software requirements, and software design elements
- Contain the columns necessary to illustrate traceability for integration, acceptance, regression, and performance testing
- Be populated with all requirements documented in the SyRS
- Indicate the source or origin of each requirement

A sample RTM template is provided as an additional tool in the Appendix.

Note: Maintaining the RTM as a separate document and performing appropriate updates in a controlled fashion—rather than including it within the SyRS and requiring that the SyRS be revised each time the RTM is modified—is more efficient.

Section 6. References

Identify the information sources referenced in the System Requirements Specification (SyRS) and utilized in developing the System Requirements Specification. Include for each the document number, title, date, and author.

Section 7. Glossary

Define all terms and acronyms required to interpret the System Requirements Specification properly.

Section 8. Revision History

Identify changes to the System Requirements Specification.

Section 9. Appendices

Include any relevant appendices.

Appendix A. Sample Requirements Traceability Matrix

Requirement Source	Requirement Description	Allocation	SyRS Reference	SyDD Element	SRS Reference	SDD Element	Test Scenario Reference	Test Description Reference	Test Procedure Reference