[Project Name] Requirements Specification

[Document Version Number]
[Date]
Project Team:
[Name] [Role]
[Name] [Role]
[Name] [Role]
[Name] [Role]
Document Author(s):
[Name] [Role]
[Name] [Role]
Customer Representative(s):
[Name]

I. Introduction

The requirements document specifies the services that the system will provide and the constraints under which the system must operate.

II. Functional Requirements

Enumerate all functional requirements in this section. It is generally a good idea to organize the functional requirements according to the modules into which the system has been decomposed by the system architect. Don't forget to provide traceability information such as where the requirement originated and a unique identifier, what the priority is, a definition of different priority levels, how stable the requirement is, and a definition of different stability levels.

Example:

FR1. Communication with Server.

FR1.1. The system shall be able to communicate with the Zephyr server.

Description: Messages, location of other users, and class subscriptions must all be handled through communication with the server.

Origin: Use cases III.2.1., III.2.2., III.2.5., III.2.12., III.2.13., III.2.14., and III.2.15. Customer interview from November 11, 2000.

Priority: 2
Stability: 2

III. Nonfunctional Requirements

Enumerate all nonfunctional requirements in this section. Again, it is best to organize the nonfunctional requirements according to the modules into which the system has been decomposed by the system architect. Don't forget to provide traceability information such as where the requirement originated and a unique identifier, what the priority is, a definition of different priority levels, how stable the requirement is, and a definition of different stability levels.

Example:

NR1. Timing

NR1.1 WindowGrams sent by WinZephyr shall be received at the destination in an amount of time comparable to Unix Zephyr.

Description: The time to receive a WindowGram is described as nearly instantaneous. Messages consisting of 200 characters (roughly three lines of text) shall be sent to and received from the server in an average of two seconds.

Origin: Zephyr on Athena Manual.

Priority: 3
Stability: 1

IV. Constraints

Enumerate all constraints.

V. Requirements Dependency Traceability Matrix

Provide a cross-reference matrix showing related requirements as shown in the example below. The matrix is used to identify dependencies between requirements to identify when one requirement must be completed before another can be implemented.

Is dependent upon requirement				
	FR1.1	FR1.2	FR2.1	NR1.1
FR1.1		X		
FR1.2				
FR2.1				X
NR1.1	X			

VI. Development and Target Platforms

Describe in full detail the expected development and target platforms including software/hardware types, versions, and so on.

VII. Project Glossary

Define any terms that are used throughout the requirements document. There will be many domain terms that have specific meaning in context. It is important to have all these terms defined in one place so that their meaning is clear to all readers.

VIII. Document Revision History

This section includes a list of significant changes that have been made to this document after the 1.0 version has been submitted for assessment. The revision history should contain a dated list of revisions to the document consisting of: the date of each change, the person responsible for the change, and a description of the change. You should be able to trace changes to the individual who completed the modification. Changes are to be listed in reverse chronological order, recording the following information for changes:

Version	File version number
Name(s)	Name of the individual(s) responsible for the change.
Date	Date of committed change.
Description	Description of the changes made to the file.