Requirements Analysis Document

STARS Project

15-413 Software Engineering

Fall 1999

Carnegie Mellon University

Pittsburgh, PA 15213

Revision History:

Version R0.1 9/15/98 Robin Loh. Created

Version R0.2 9/14/98 Joyce Johnstone. revised template

Preface:

This document addresses the requirements of the STARS system. The intended audience for this document are the designers and the clients of the project.

Target Audience:

Client, Developers

STARS Members:

(List STARS team members)

INSTRUCTIONS

We encourage you strongly to coordinate your section with other teams via Lotus Notes bboard, discussions in team meetings, architecture integration liaison meetings, documentation liason and on the discuss bboard. Submission (must be a HTML document):

1. Copy this document from the course homepage.
2. The questions provided under each section is to help you brainstorm for ideas and offer

suggestions. You must elaborate on them and produce coherent and descriptive paragraphs.

3. Submit your RAD by posting them on the Review of Documents bboard.

MILESTONES

- 9/15 Release of RAD Template (Project Management)
- 10/7 Team RADs Due (Developers)
- 10/11 Second Draft of team RADs due (Developers)
- 10/17 RAD Review Presentation Deadline

1.0 General Goals

For this section, enter the goals of your subsystem, i.e. what are the objectives of the functions of your subsystem?

2.0 Current System

For this section, describe the current situation that is relevant to your subsystem.

3.0 Proposed System

For this section, describe the proposed solution (i.e. your subsystem) under the following headings.

3.1 Overview

For this section, give a top-level description of your subsystem.

3.2 Functional Requirements

For this section, list out the functional requirements of your subsystem.

3.3 Nonfunctional Requirements

For this section, list out the non-functional requirements of your subsystems in the following headings.

3.3.1 User Interface and Human Factors

For this section, you will have to think about the interaction between the potential users and your subsystem. Consider the following:
What type of user will be using the system (expert, novice, etc.) ? Will more than one type of user be using the system? What sort of training will be required for each type of user? Is it particularly important that the system be easy to learn? Is it particularly important that users be protected from making errors? What sort of input/output devices for the human interface are available, and what are their characteristics?

3.3.2 Documentation

For this section, focus on your plans for future subsystem documents. Consider the following:
What kind of documentation is required? What audience is to be addressed by each document?

3.3.3 Hardware Consideration

For this section, think about the hardware issues that your subsystem will be facing. Consider the following:
What hardware is the proposed system to be used on? What are the characteristics of the target hardware, including memory size and auxiliary storage space?

3.3.4 Performance Characteristics

For this section, consider the performance requirements and limitations of your subsystem. Consider the following:
Are there any speed, throughput, or response time constraints on the system? Are there size or capacity constraints on the data to be processed by the system?

3.3.5 Error Handling and Extreme Conditions

For this section, focus on the possible error occurrences and how your subsytem will deal with them. Consider the following:
How should the system respond to input errors? How should the system respond to extreme conditions?

3.3.6 System Interfacing

For this section, think about the I/O of your subsystem. Consider the following:
Is input coming from systems outside the proposed system? Is output going to systems outside the proposed system? Are there restrictions on the format or medium that must be used for input or output?

3.3.7 Quality Issues

For this section, focus on the possible quality enhancement or compromises. Consider the following:
What are the requirements for reliability? Must the system trap faults? Is there a maximum acceptable time for restarting the system after a failure? What is the acceptable system downtime per 24-hour period? Is it important that the system be portable (able to move to different hardware or operating system environments)?

3.3.8 System Modifications

For this section, think about the current infrastructure of your system which will be extended for future features, incorporated or made obsolete. Consider the following:
What parts of the system are likely candidates for later modification? What sorts of modifications are expected?

3.3.9 Physical Environment

For this section, consider the physical environment in which your susbsystem will exist. Consider the following:
Where will the target equipment operate? Will the target equipment be in one or several locations? Will the environmental conditions in any way be out of the ordinary (for example, unusual temperatures, vibrations, magnetic fields, ...)?

3.3.10 Security Issues

For this section, Focus on all possible security considerations. Consider the following:
Must access to any data or the system itself be controlled? Is physical security an issue?

3.3.11 Resource Issues

For this section, think about data management for your subsystem. Consider the following:
How often will the system be backed up? Who will be responsible for the back up? Who is responsible for system installation? Who will be responsible for system maintenance?
.

3.4 Constraints

For this section, consider all the limitations imposed on your subsystem. Consider the following:
Constraints on the programming language. Constraints on the development environment.

Constraints on the use of libraries. Constraints on the use of legacy systems.

3.5 System Model

You will have to use the UML (Unified Modelling Language) to create the models. If the CASE tools is not installed yet (Together-J), you can use Visio or Powerpoint to produce the models. For more information on the notations of UML, check out the following Rational websites - Notation and Documentation. To make your models more readable, you have to include some texts to guide the reader along the flow of your model. These text are called Navigational Text because they help to move the reader along the models.

3.5.1 Scenarios

For this section, think about all the possible ways which the users will interact with your subsystem. Present them in a "story" format.

3.5.2 Use Case Models

3.5.2.1 Actors
3.5.2.2 Use Cases

3.5.3 Object Models

3.5.3.1 Data Dictionary
3.5.3.2 Class Diagrams

3.5.4 Dynamic Models

3.5.5 User Interface - Navigational Paths and Screen Mockups