Software Engineering

REQUIREMENTS SPECIFICATION
IEEE-830

IEEE-830: The Scheme

The various components of the standard should provide the following information:

1) Introduction
2) General description
3) Specific requirements

IEEE-830: (1) Introduction

- This section contains an overview of the complete document

(1) Introduction

1.1) Purpose

1.2) Scope

1.3) Definitions, acronyms and abbreviations

1.4) References

1.5) Overview

return to scheme continue introduction

IEEE-830: General Description

- This section contains a description of matters that concern the overall product and its requirements. It provides a perspective for understanding the specific requirements from section 3 of this document.

(2) General Description

2.1) Product perspectives

2.2) Product functions

2.3) User characteristics

2.4) General constraints

2.5) Assumptions and dependencies

return to scheme

continue to specific description

IEEE-830: Specific Requirements

- This section contains all the details which are relevant for the design phase to follow. The ordering given here is just one way to present the specific requirements in a logical way. Specific requirements should be such that one may objectively determine whether they are fulfilled or not.

(3) Specific Requirements

3.1) Functional requirements

3.2) External interface requirements

3.3) Performance requirements

3.4) Design constraints

3.5) Attributes

3.6) Other requirements

return to scheme

continue to functional requirements

(1) Introduction

(1.1) Purpose

- What is the purpose of this document, and for whom is it written?
- *This document states the requirements of an automated library for a medium-sized library of a research institute. The requirements stated serve as a basis for the acceptance procedure of this system. The document is also intended as a starting point for the design phase.*

Return to introduction

(1.2) Scope

- An identification of the product to be developed, what does it do (and what does it not do), why is the product being developed (including a precise description of its benefits, goals and objectives)?

(1.2) Scope

- *The intended product automates the library functions described in... Its purpose is to provide a more effective service to the library users, in particular through the online search facilities offered. Once the system is installed, the incorporation of new titles will go from an average of 15 minutes down to an average of 5 minutes.*

Return to introduction

(1.3) Definitions, Acronyms and Abbreviations

- This subsection contains definitions of all the terms, acronyms and abbreviations used in the document. Special attention should be paid to the clarification of terms and concepts from the domain of application.

Return to introduction

(1.4) References

- References to all documents that are referred to in the remainder of the requirements specification.

Return to introduction

(1.5) Overview

- This subsection contains an outline of the remainder of the document.
- *Section 2 of this document gives a general overview of the system. Section 3 gives more specific requirements for functions offered to (external) users and library personnel, respectively.*

Return to introduction

continue to general description

(2) General Description

(2.1) Product Perspective

- Does it concern an independent product or is it part of a larger product? In the latter case, the other components should be identified, and the interfaces with those components should be described. In this section, we also give an identification of the hardware to be used.

(2.1) Product Perspective

- *The already installed DB system will be used to store the various catalogues as well as the user administration. There are no interfaces to other systems. The system will be realized on the Y-configuration. System y has a maximum capacity of 32 terminals of type Z. The maximum external storage capacity for the catalogues of the system is 1500 Mb.*

Return to general description

(2.2) Product Functions

- An overview of the functions of the system to be delivered. This should be confined to an overview. A detailed discussion of the functions is given in section 3 of the requirements specification.

(2.2) Product Functions

- *The system provides 3 types of functions:*

  - functions by which library users may search the catalogues. A list of these functions is given in DOC1. A more detailed description is given in sections 3.1.2-3.1.3.

  - functions by which library personnel may update the administration of borrowed titles; see section 3.1.4.

  - functions by which library personnel may update the system's catalogue; see section 3.1.5.

Return to general description

(2.3) User Characteristics

- An indication of general user characteristics, in as far as these are relevant for the requirements specification.

  Experience, training and technical expertise of future users may influence specific requirements of the system to be developed.

(2.3) User Characteristics

- The library users are identical users of the system and have little knowledge of automated systems of this kind. The system therefore has to be self-instructing. Specific requirements are formulated in sections 3.2.1 and 3.3. The library personnel will be trained in the use of the system; see section 3.2.1.

Return to general description

(2.4) General Constraints

- An indication of any other constraints that apply. These may concern government regulations, hardware constraints, security regulations, and so on. Again, we are concerned with the rationale at this point. A further elaboration follows in section 3 of this document.

(2.4) General Constraints

- *Library users may search only the catalogues of books and journal articles; they are not allowed to update a catalogue or the user administration. A password will be needed for these functions. Each library employee will be given his own password; see also section 3.5.2.*

Return to general description

(2.5) Assumptions and Dependencies

- This does not concern constraints on the system to be developed, but things which may influence the requirements specification once they change.

(2.5) Assumptions and Dependencies

- E.g., think of the availability of certain supporting software, such as some given OS or a numeric library. If that OS or library turns out not to be available, the requirements specification will have to be adapted accordingly.

return to general description

continue to specific requirements

(3) Specific Requirements

(3.1) Functional Requirements

- A description is given of how the transformation of inputs to outputs is achieved. The description is given for each class of functions, and sometimes for each individual function. To a certain extent, this description can be seen as a solution to the user. This component of the requirement specification is the main starting point for the design phase.

(3.1) Functional Requirements

(3.1.n) Functional Requirement n

(3.1.n.1) Introduction

- A description of the purpose of this function and the approaches and techniques used. The introduction should include information to clarify the intent of the function.

(3.1.n) n =1, 2, 3… and denotes leveling

(3.1) Functional Requirements

(3.1.1) Functional Requirement 1*:*

Select Feature

(3.1.1.1) Introduction

The main menu appears after the system is started. The user selects one of the options from the main menu. Subsequent actions are described in section 3.1.3-3.1.5. If the option selected is constrained to library employees, the system asks for a password before a switch to return to the main menu.

(3.1) Functional Requirements

(3.1.2) Functional Requirement 2:

Search Book Catalogue

(3.1.2.1) Introduction

Given (part of) a book title or author name, the user may search the book catalogue for titles that match the user input given. The user is offered a screen with two fill-in-the-blank area (title and author), one of which is to be filled in.

(3.1) Functional Requirements

(3.1.n.2) Inputs

- A precise description of the function's inputs (source, quantities, range of acceptable values, and the like).

(3.1.2.2) Inputs (Search Book Catalogue)

The input may contain both upper and lower case letters. Special symbols allowed are listed in DOC1. Any other glyphs entered are discarded and are not shown on the screen.

Input is considered complete when the processing command is issued.

(3.1) Functional Requirements

(3.1.n.3) Processing

- A definition of the operations that must be performed, such as checking for acceptable values, reaction to abnormal situations, or a description of algorithms to be used. As an example of the latter, one may think of the use of some mathematical model for strength computations within a CAD-program.

(3.1) Functional Requirements

(3.1.2.3) Processing

(Search Book Catalogue)

- *All the lower case letters are turned into upper case letters. The string thus obtained is used when quering the DB. A DB entry matches the title string given if the transformed input is a substring of the title-field of the entry. The same holds for the author field if (part of) an author name is input.*

(3.1) Functional Requirements

(3.1.n.4) Outputs

- A precise description of the outputs (destination, quantities, error messages, and the like).

(3.1.2.4) Outputs (Search Book Catalogue)

- *A list of titles that match the input is displayed. Up to four titles are shown on the screen. The user may traverse the list of titles found using the screen scrolling commands provided. A special warning is issued if no title matches the input given.*

Return to specific requirements

(3.2) External Interfaces

- This should be a detailed descriptions of all inputs into and outputs from the software system.
-

(3.2.1) User Interfaces

- A description of the characteristics of the user interfaces, such as screen layout, function keys, help functions. In order to support testing, verifiable requirements regarding learning time for the system functions should be included either here or in some subsection of 3.5 (Attributes).

(3.2.1) User Interfaces

- *(Search Book Catalogue)*
  The screen formats for the different features are specified in Appendix A. Appendix B lists the mapping of commands to function keys. Appendix C contains a list of typical usage scenarios. These usage scenarios will be used as acceptance criteria: 80% of the users must be able to go through them within 10 minutes. An instruction session for library personnel should take at most 2 hours.

(3.2.2) Hardware Interfaces

- A description of the logical characteristics of hardware interfaces, such as interface protocols, or screen-oriented versus line-oriented terminal control.
- *The user interface is screen oriented. The system uses up to 10 function keys.*

(3.2.3) Software Interfaces

- A description of software needed, such as a certain operating system or subroutine package. Interfaces to other application software is also discussed here.
- *The interface with the database system is described in DOC2.*

(3.2.4) Communications Interfaces

- An example is a communication protocol for LANs.
- *Not applicable here.*

Return to specific requirements

(3.3) Performance Requirements

- Static and dynamic requirements
  - Static requirements concern, e.g., the number of terminals to be connected and the number of users that can be handled concurrently.

(3.3) Performance Requirements

- Dynamic requirements concern the operational performance of the system: how frequently will certain functions be called for and how fast should the system's reaction be. It is important that these requirements be stated in measurable terms.

Return to specific requirements

(3.3) Performance Requirements

- *The system will initially support 10 terminals (maximum capacity is 32). The DB holds 25000 book titles and 500 journal subscriptions. The storage capacity needed for these data is 300 MB. On average 1000 books and 2000 journal issues enter the library per year. The average journal issue has six articles. This requires a storage capacity of 15 MB per year.*
  The system must be able to serve 20 users simultaneously. With this maximum load and DB size of 450 MB, user queries as listed in sections 3.1.2 and 3.1.3 must be answered with 5 seconds in 80% of the cases.

(3.4) Design Constraints

- Design constraints may result from such things as the prescribed use of certain standards or hardware.

(3.4) Design Constraints

(3.4.1) Standards compliance

- Which existing standards or regulations must be followed, and what requirements result from these. For example, certain report formats or audit procedures may be prescribed.
- *Title descriptions must be stored in PICA-format. This format is described in DOC3.*

(3.4) Design Constraints

(3.4.2) Hardware limitations

- A description of the characteristics of the hardware environment, in as far as they lead to software requirements. An example of this might be the amount of memory available.
- *See section 2.1.*

Return to specific requirements

(3.5) Attributes

- In this section, particular attention is paid to quality aspects. These requirements must be measurable and verifiable. They must be stated in objective terms.

Note: The subsections that follow by no means comprise a complete list of attributes.

(3.5.1) Availability

- Factors that guarantee a certain level of availability, such as restart procedures. In this subsection we may also enlist requirements regarding fault tolerance (with respect to both hardware failures and software failures).
- *During normal library hours the system must be available 95% of the time. A backup of the system is made at the end of each day.*

(3.5.2) Security

- Requirements regarding unauthorized access and other forms of misuse. Certain cryptographic techniques may be prescribed, and we may put constraints on the communication between different parts of the system.
- *The functions Borrow Title and Modify Catalogue are restricted to library employees and protected by passwords.*

(3.5.3) Maintainability

- Requirements to guarantee a certain level of maintainability of the system, such as a maximum allowable coupling between components.

Return to specific requirements
Return to content

(3.6) Other Requirements

- A description of requirements that are specific to certain software, and which have not been discussed yet.

return