Specification of software requirements

Draft: del proyecto]
Review

de año]

Instructions for using this format

This format is a standard template for software requirements documents.

It is based on and complies with the IEEE Std 830-1998 standard.

The sections that are not considered applicable to the system described may justifiably be indicated as not applicable (NA).

Notes:
Texts in blue are indications that must be eliminated and, where appropriate, replaced by the contents described in each section.

Texts in square brackets such as “aquí el texto]”Allow the direct inclusion of text with the appropriate color and style for the section, when clicking on them with the mouse pointer.

The titles and subtitles of each section are defined as MS Word styles, so that their consecutive numbering is automatically generated according to the “Title1, Title2 and Title3” styles.

The indentation of the texts within each section is automatically generated when you press Enter at the end of the title line. (Styles Normal indented1, Normal indented 2 and Normal indented 3).

The document index is a table of contents that MS Word updates based on the document titles.
Once your writing is finished, Word should be instructed to update all its content to reflect the final content.

.

Revision history

Date	Review	Description	Author
dd / mm / yyyy	1.0	"Interface Requirements"	<name>

Document validated by the parties on date:

For the client	By the supplying company
Signed. Mr. / Mrs.	Signed. Mr / Mrs

Content

1 Introduction

aquí el texto]

The introduction to the Software Requirements Specification (SRS) should provide an overview of the SRS. It should include the purpose, scope, definitions and acronyms, references, and overview of the SRS.

1.1 Purpose

aquí el texto]

- *Purpose of the document*
- *Target audience*

1.2 Scope

aquí el texto]

- *Identification of the product (s) to be developed by a name*
- *Consistency with similar definitions of higher level documents (eg system description) that may exist*
- ***[A description of the affected environment; which projects are affected or influenced by this Software Requirements Specification.]***

1.3 Involved personnel

Name	aquí el texto]
Role	aquí el texto]
Professional category	aquí el texto]
Responsibilities	aquí el texto]
Contact information	aquí el texto]
Approval	aquí el texto]

List of people involved in the development of the system, with contact information. This information is useful so that the project manager can locate all the participants and gather the necessary information to obtain requirements, follow-up validations, etc.

1.4 Definitions, acronyms and abbreviations

aquí el texto]

Definition of all terms, abbreviations and acronyms necessary to properly interpret this document. In it you can indicate references to one or more appendices, or to other documents.

1.5 References

Reference	Title	Route	Date	Author

Complete list of all documents related to the software requirements specification, identifying each document's title, reference (if applicable), date and organization that provides it.

1.6 Summary

aquí el texto]

- *Description of the content of the rest of the document*
- *Explanation of the organization of the document*

2 General description

[The description of the main factors that affect the solution space is considered in this part. Include such items as product perspective, product features, user characteristics, limitations, assumptions, and dependencies. The description of the requirements is not included in this section.]

2.1 Product perspective

aquí el texto]
Indicate if it is a standalone product or part of a larger system. In the case of a product that is part of a larger system, a diagram that places the product within the system and identifies its connections facilitates understanding.

2.2 Product functionality

aquí el texto]
Summary of the main functionalities that the product must perform, without entering into detailed information.
Sometimes the information in this section can be taken from a higher level system specification document (eg System Requirements).
The functionalities must be organized in such a way that the client or any interlocutor can understand it perfectly. For this, textual or graphic methods can be used.
[If you are using use case modeling, this section should contain its reference, and a description or summary of the model or the most representative subset of it. This includes a list of names and brief descriptions of the use cases, actors, applicable diagrams, and relationships.

If there is no use case model, all the existing descriptions of the functionalities must be referenced, whether they are meeting minutes, emails, etc. It is necessary to add those descriptions in this section and in the References section of the document all sources of the requirements need to be mentioned.

2.3 User characteristics

Type of user	aquí el texto]
Training	aquí el texto]
Skills	aquí el texto]
Activities	aquí el texto]

Description of the users of the product, including educational level, experience and technical experience.

2.4 Restrictions

aquí el texto]
Description of those limitations to take into account when designing and developing the system, such as the use of certain development methodologies, programming languages, particular standards, hardware restrictions, operating system restrictions, etc.

2.5 Assumptions and dependencies

aquí el texto]
Description of those factors that, if they change, may affect the requirements. For example an assumption might be that a certain operating system is available for the required hardware. In fact, if the operating system was not available, the SRS would have to be modified.

2.6 Predictable evolution of the system

aquí el texto]

Identification of future improvements to the system, which may be analyzed and implemented in the future.

3 Specific requirements

This is the longest and most important section of the document.
It must contain a detailed and complete list of the requirements that the system to be developed must meet. The level of detail in the requirements must be sufficient so that the development team can design a system that meets the requirements and that testers can determine whether they are met.

The requirements will be arranged in the form of numbered lists for identification, monitoring, traceability and validation (eg RF 10, RF 10.1, RF 10.2, ...).

For each requirement, the following table must be completed:

Requirement number	aquí el texto]		
Requirement name	aquí el texto]		
Type	☐ Requirement	☐ Restriction	
Requirement source	aquí el texto]		
Requirement priority	☐ High / Essential	☐ Average / Desired	☐ Low / Optional

and make the description of the requirement

The distribution of the paragraphs that make up this point may differ from the one proposed in this template, if the characteristics of the system advise another distribution to offer greater clarity in the exposition.

3.1 Common interface requirements

aquí el texto]
Detailed description of all inputs and outputs of the software system.

3.1.1 User interfaces

aquí el texto]
Describe the user interface requirements for the product. This can be in the form of text descriptions or interface screens. For example, the customer may have specified the style and colors of the product. Describe precisely how the product will appear to its intended user.

3.1.2 Hardware interfaces

aquí el texto]
Specify the logical characteristics for each interface between the product and the hardware components of the system. Configuration features will be included.

3.1.3 Software interfaces

aquí el texto]
Indicate whether to integrate the product with other software products.
The following must be specified for each software product:
- *Description of the software product used*
- *Interface purpose*
- *Interface definition: content and format*

3.1.4 Communication interfaces

aquí el texto]

Describe the communication interface requirements if there are communications with other systems and what the communication protocols are.

3.2 Functional requirements

aquí el texto]
Definition of fundamental actions that the software must take when receiving information, processing it and producing results.
They include:

- *Validity check of tickets*
- *Exact sequence of operations*
- *Response to abnormal situations (overflows, communications, error recovery)*
- *Parameters*
- *Output generation*
- *Relationships between inputs and outputs (sequences of inputs and outputs, formulas for information conversion)*
- *Specification of the logical requirements for the information to be stored in the database (type of information, required)*

Functional requirements can be divided into subsections.

3.2.1 Functional requirement 1

3.2.2 Functional requirement 2

3.2.3 Functional requirement 3

3.2.4 Functional requirement n

3.3 Non-functional requirements

3.3.1 Performance requirements

aquí el texto]
Specification of the requirements related to the load that the system is expected to bear. For example, the number of terminals, the expected number of simultaneously connected users, the number of transactions per second that the system must support, etc.
All of these requirements must be measurable. For example, stating "95% of transactions must be completed in less than 1 second", instead of "operators should not wait for the transaction to complete".

3.3.2 Security

aquí el texto]
Specification of elements that will protect the software from malicious access, use and sabotage, as well as from malicious or accidental modifications or destruction.
Requirements may specify:

- *Use of cryptographic techniques.*
- *Record of files with activity logs.*
- *Assignment of certain functionalities to certain modules.*
- *Communication restrictions between certain modules.*
- *Integrity checks of critical information.*

3.3.3 Reliability

aquí el texto]

Specification of the necessary reliability factors of the system. This is generally expressed as the time between permissible incidents, or the total permissible incidents.

3.3.4 Availability

aquí el texto]
Specification of the final availability factors required of the system. Normally expressed in% of time in which the software has to show availability.

3.3.5 Maintainability

aquí el texto]
Identification of the type of maintenance required for the system.
Specification of who should perform maintenance tasks, for example users, or a developer.
Specification of when maintenance tasks should be performed. For example, generation of weekly and monthly access statistics.

3.3.6 Portability

aquí el texto]
Specification of attributes that the software must present to facilitate its transfer to other platforms or environments. They may include:

- *Percentage of server-dependent components.*
- *Percentage of code dependent on the server.*
- *Use of a certain language for its portability.*
- *Use of a specific compiler or development platform.*
- *Use of a specific operating system.*

3.4 Other requirements

aquí el texto]
Any other requirement that does not fit in any of the previous sections.

For example:
Cultural and political requirements
Legal requirements

4 Appendices

aquí el texto]
They can contain all kinds of information relevant to the SRS but which, properly speaking, is