
Specification of software requirements

Draft: Prototype of virtual ID for first level student of
the Universidad de las Fuerzas Armadas “ESPE”

Review

Logo

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

DOCUMENT FILE3

CONTENTS4

1 INTRODUCTION6

1.1 Purpose6

1.2 Scope6

1.3 Involved personnel6

1.4 Definitions, acronyms and abbreviations6

1.5 References6

1.6 Summary6

2 OVERVIEW7

2.1 Product perspective7

2.2 Product functionality7

2.3 User characteristics7

2.4 Restrictions7

2.5 Assumptions and dependencies7

2.6 Predictable evolution of the system7

3 SPECIFIC REQUIREMENTS7

3.1 Common interface requirements8

3.1.1 User interfaces8

3.1.2 Hardware interfaces8

3.1.3 Software interfaces

3.1.4 Communication interfaces8

8

3.2 Functional requirements8

3.2.1 Functional requirement 19

3.2.2 Functional requirement 29

3.2.3 Functional requirement 39

3.2.4 Functional requirement n9

3.3 Non-functional requirements9

3.3.1 Performance requirements9

3.3.2 Safety9

3.3.3 Reliability9

3.3.4 Availability9

3.3.5 Maintainability10



3.3.6 Portability10

3.4 Other requirements10

4 Appendices10

1 Introduction

1.1 Purpose

The objective of this file is to present a descriptive specification of the first identification model for first grade students, where the properties of the system, system interfaces, as well as the limitations under which it should operate will be explained. This file is intended for interested users and also for system developers.

1.2 Scope

This prototype is going to be a virtual certification system for first grade students. This system will be developed to facilitate both mobility and the use of technologies, otherwise it would have to be developed manually.

More especially, this system is designed so that first-grade students obtain this virtual Canet and in this way streamline the procedures that all said users have, both at the entrance of the institution and in the polyclinic, in addition to having the services of the polyclinic. This system will have the data entry of each student in order to create a code that allows them to enter the institution and also the polyclinic.

1.3 Involved personnel

Name	Moromenacho Tipan, Mishell Estefania
Role	Developer
Professional category	IT Student
Responsibilities	editor and developer
Contact information	memoromenacho@espe.edu.ec
Approval	aquí el texto]

Name	Navarro Zambrano, Johny Nicolay
Role	Developer
Professional category	IT Student
Responsibilities	editor and developer
Contact information	jnnavarro@espe.edu.ec
Approval	aquí el texto]

Name	Ocaña Bolaños, Francisco Javier
Role	Developer
Professional category	IT Student
Responsibilities	editor and developer
Contact information	fjocana@espe.edu.ec
Approval	aquí el texto]

Name	Pallasco Mancero, Oscar Fernando
Role	Developer
Professional category	IT Student
Responsibilities	editor and developer
Contact information	ofpallasco@espe.edu.ec
Approval	aquí el texto]

Name	Pazmiño Gordon, Bryan Alexander
Role	Developer
Professional category	IT Student
Responsibilities	editor and developer
Contact information	bapazmino3@espe.edu.ec
Approval	aquí el texto]

1.4 Definitions, acronyms and abbreviations

VR: Virtual Card

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
IEEE	Standard IEEE 830 - 1998			

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

In the next chapter, there is the general explanation. Explains the requirements used in this system, an environment for the description of technical requirements.

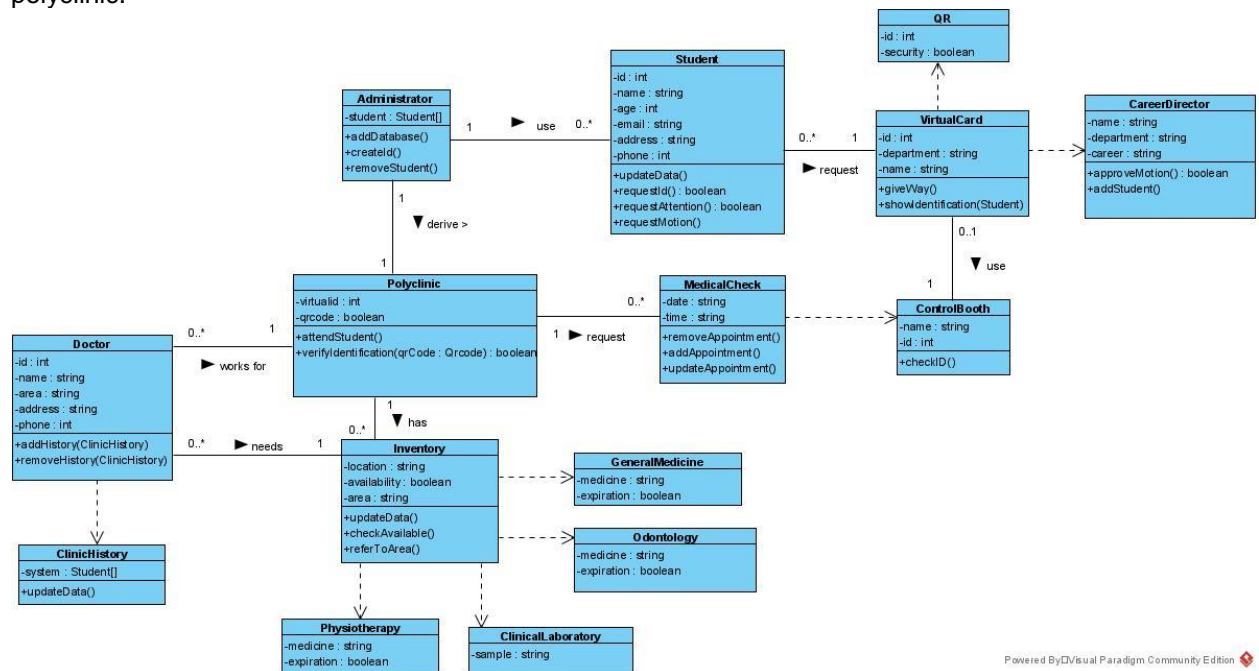
The third chapter, Requirements Description part, of this file is written primarily for developers and explains in technical terms the details of the system's functionality.

2 General description

2.1 Product perspective

The program depends on the implemented system of the polyclinic that will be connected with different systems such as the sentry box that will be in charge of giving access to the institution to the student.

It will also depend on the polyclinic database since that way the student can be identified and all their data can be accessed. The career director also plays an important role since he is in charge of giving said information from the student to the polyclinic.



Powered By/DI/visual Paradigm Community Edition

2.2 Product functionality

Virtual identification is based on the student being able to enter their data virtually from a device, in this way they can generate their identification that will be used to access different areas of the campus such as:

- Access to the polyclini
- Access to laboratories
- Access to the library
- Access to parking
- Others.

In this case we will focus more on access to polyclinic.

Once the student enters their data, they will be saved in the ESPE's database and their virtual ID will be created, that information will reach each area of the campus once the Director of the program has authorized it. In this way, the student will be able to access the free campus services.

In the polyclinic area, the first thing that will be done before assisting the student is to verify that they belong to the institution, this is achieved thanks to the QR code that each student has in their virtual ID. Once the information is corroborated, the student will be served in the area that he wishes to be served.

If the student needs any medication, the polyclinic will deliver it without any problem and will register it in the database.

The virtual ID has the same appearance as a normal card that the ESPE grants to all students with the minimum difference that it is electronic and it is more feasible to obtain it.



2.3 User characteristics

Type of user	First level students
Training	IT engineering
Skills	study
Activities	Student

2.4 Restrictions

The program, being a prototype, is limited to many things when designing such as the number of users, the types of uses in different areas, programming languages, etc.

In the type of user we limit ourselves to only new students of the IT career and with respect to the types of areas we limit ourselves to only the area of medicine and the entrance of the student to the institution.

2.5 Assumptions and dependencies

As the program is a prototype, its dependencies are not defined, however at the time of terminating the program it would depend on the operating system of the device since it would have to have several versions of the program depending on the device it is on.

2.6 Predictable evolution of the system

In future versions it is desired to expand the types of uses in different areas of the institution such as parking lots or laboratories.

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	<input type="checkbox"/> Requirement	<input type="checkbox"/> Restriction	
Requirement source	aquí el texto]		
Requirement priority	<input type="checkbox"/> High / Essential	<input type="checkbox"/> Average / Desired	<input type="checkbox"/> 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 not part of the SRS.