

**Team 1:** Jsons

**Members:** Ruben Dario Benavides Macias **NRC:** 9642

Adonny Mateo Calero Argüello

Pablo Esteban Carrera Quinde

Yeshua Amador Chiliquinga Amaya

**System’s requirements**

**1. Introduction**

This document is a Software Requirements Specification (SRS) for the application “OdontoApp”. The content has been created through the collaboration of the members mentioned. This specification has been structured inspired by the guidelines given by the standard "IEEE Recommended Practice for Software Requirements Specification ANSI/IEEE 830 1998".

**1.1. Purpose**

The objective of carrying out the specification is to clearly and precisely define all the functionalities and restrictions of the system to be built. The document is addressed to the system development team. This document will be reviewed by the developer team, inspector team and a record of document versions will be kept on the Git-hub platform, until it is approved by the group involved. Once approved, it will serve as a basis for the development team to build the new system.

**1.2. Scope**

OdontoApp is a dentist consultory platform with the following objective:

Automate the patients data management and the business rules of: services purchase and discounts, appointment assignments, calculation of annual incomes.

This beneficiates these aspects for the dentist:

* + It reduces expenses.
  + It reduces the consumption of paper.
  + It reduces time.
  + Faster and more efficient service in general.

**1.3. Definitions, Acronyms, and Abbreviations**

|  |  |
| --- | --- |
| **Term** | **Description** |
| FR | It refers to a functional requirement. |
| NFR | It refers to a non functional requirement. |
| GitHub | A Version Control Software (VCS) platform. |
| SRS | Software Requirements Specification |
| PL | Programming Language |
| JAVA | A programming language. |
| MONGODB | Database platform. |

**1.4. References**

|  |  |
| --- | --- |
| **Document Title** | **Reference** |
| Specification of Requirements according to the IEEE 830 standard | https://www.fdi.ucm.es/profesor/gmendez/docs/is0809/ieee830.pdf |

**1.5. Overview**

Content organization is structured by 4 layers.

1. A brief view of the system, the purpose of the present document and some details to know before continuing.
2. A broader description of the system and its characteristics, functionalities and a possible future view of the same.
3. Explanation in a specific level description how the software will work describing technical components and requirements the same has to satisfy.
4. Finally, extra no related SRS information that complement the document.

**2. Overall Description**

**2.1. Product Perspective**

In general terms, the system will have access to a clinical history created by the dentist, that will make available the information of each patient who has gone to the dentist, in order to execute certain actions in benefit of the information management. Then, there are some business rules or functions available in the application. The system is not related to a larger product, it is completely independent.

**2.2. Product Functions**

The application will manage patient information through a program that allows the dentist to add, delete, edit and read patients with their respective clinical information in an organized manner and with the ability to obtain the data in the interface of the application. On the other hand, the application allows a dentist to buy services for a determined patient, automatically register the buy and show the buy (billing), also the purchase can include a discount if its cost is more than 200 dollars, the application allows to assign appointments to a specific patient, and finally calculates the annual incomes and read this information.

**2.3. User Characteristics**

|  |  |
| --- | --- |
| **User Type** | **Characteristics** |
| Dentist | She will be able to:  Create a patient entering the respective data she needs.  Edit that patient and the respective data she needs.  Delete completely a patient and its respective data.  Read all the patients available and its data  To have all this data in a database shown in the interface.  —--------------------------------------------------  For a determined patient:  Buy services  Show the bought services (billing)  To have all this data in a database shown in the interface.  —--------------------------------------------------  For a determined patient:  Add an appointment.  To have all this data in a database shown in the interface.  —--------------------------------------------------  She will be able to:  Calculate the daily and/or monthly incomes.  Read the daily and/or monthly incomes.  To have all this data in a database shown in the interface. |

**2.4. Constraints**

* The system is the intellectual property of the aforementioned team, which is part of the Universidad de las Fuerzas Armadas ESPE.
* Use of PL: Java
* The development process will involve the use of GitHub and other less-formal communication methods.
* Developers abilities domain: Basic object-oriented programming in Java, files use (json) in Java, command line management, GitHub, MongoDB, GUI.

**2.5. Assumptions and Dependencies**

**2.5.1 Assumptions**

The functionality of the application is expected under the presumption that the requirements mentioned in this document will not be drastically altered since the goals and impact of the application could be directly affected. In the event that changes are required, the OdontoApp development team will agree on any necessary modifications to the system.

**2.5.2 Dependencies**

The application will function autonomously since there is no observed dependence on external applications and/or systems. The system will correspond between the client and the application, and therefore the effectiveness of the system will be directly related to the devices on which the application is used, in this case, only available in a computer desktop.

**3. Specific Requirements**

|  |  |
| --- | --- |
| **Requirement Identifier** | FR01 |
| **Scope in the system** | System Principal Menu |
| **Requirement Definition** | When executing the distributable, the system will have a simple and easy interface to use. |

|  |  |
| --- | --- |
| **Requirement Identifier** | FR02 |
| **Scope in the system** | System Principal Menu |
| **Requirement Definition** | The system will have a login and password. |

|  |  |
| --- | --- |
| **Requirement Identifier** | FR03 |
| **Scope in the system** | Patient Management Menu |
| **Requirement Definition** | The system will allow the dentist to add, edit, read and delete patients (CRUD). |

|  |  |
| --- | --- |
| **Requirement Identifier** | FR04 |
| **Scope in the system** | Option Go Back |
| **Requirement Definition** | Once the dentist has entered the "go back" option, the system will automatically go back to depending on where part of the system you are in. |

|  |  |
| --- | --- |
| **Requirement Identifier** | FR05 |
| **Scope in the system** | Services Purchase |
|  | The system will be able to purchase services for a determined patient and show the corresponding bill of what he/she bought. |

|  |  |
| --- | --- |
| **Requirement Identifier** | FR06 |
| **Scope in the system** | Services Purchase Print |
|  | The system will be able to print this information physically, and or save on a digital file e.g pdf. |

|  |  |
| --- | --- |
| **Requirement Identifier** | FR07 |
| **Scope in the system** | Services Purchase Discount |
|  | The system will apply a discount of 20% if the purchase is more than 200 dollars. |

|  |  |
| --- | --- |
| **Requirement Identifier** | FR08 |
| **Scope in the system** | Appointments |
|  | The system will be able to schedule appointments for a determined patient. |

|  |  |
| --- | --- |
| **Requirement Identifier** | FR09 |
| **Scope in the system** | Option Change Patient |
|  | The system will allow the dentist to choose a patient in the edit, services purchase and appointments schedule. |

|  |  |
| --- | --- |
| **Requirement Identifier** | FR10 |
| **Scope in the system** | Incomes |
|  | The system will allow the dentist to calculate the daily incomes. |

|  |  |
| --- | --- |
| **Requirement Identifier** | FR11 |
| **Scope in the system** | DataBase |
|  | The system will save all the information in MongoDB. |

|  |  |
| --- | --- |
| **Requirement Identifier** | FR12 |
| **Scope in the system** | GUI |
|  | The system will use a graphical user interface. |

**3.1. External Interfaces**

**3.1.1. User Interface**

**User interface**

The user interface will have a main menu(with login) and other additional ones that contain the functions of the system, these being treated with inputs from the dentist's keyboard, these inputs may have limitations (such as passing a data string to a data int), but will be handled properly.

**3.1.2. Hardware Interface**

Previously we will use Yeshua's computer to run and present the program to the Dentist.

**3.1.3. Software Interface**

The system will use GUI.

**3.1.4. Communication Interface**

The system will have an easy and simple interface to use.

**3.2. Functions**

**3.2.0** Login

**3.2.1** Menus functions

**3.2.2** Add patient

**3.2.3** Edit Patient Information

**3.2.4** Delete Patient

**3.2.5** Read Patient Information

**3.2.6** Exit

**3.2.7** Buy services

**3.2.8** Discounts

**3.2.9** Schedule appointments

**3.3** Calculate daily/monthly incomes

**3.5** Upload data in the database

**3.3. Performance Requirements**

The system will have support for the entry of a person to the application, this will be the client (dentist) and from the application you will be able to manage the data you need about the patients and the relations with the business rules. This data will be saved in a mongodb to provide greater comfort when reading the information. The type of information that is planned to be stored are primitive data types such as string, int, boolean and others. It is expected that this information will be updated as the dentist uses it, so the information will be open to future changes.

**3.4. Logical Database Requirements**

We will use databases in MongoDB.

**3.6.1. Data Validation**

Data validation: ensuring that the data meets the entry requirements by comparing it against a set of pre-established or defined rules. In this case, when entering data of the patients in the database. This process involves performing a series of checks known as validation routines. It also allows us to better understand the scope of data conflicts.

**4. Appendices**



