Index:

1. SUMMARY

1.1. SYSTEMS INTENT

The primary purpose of this database is to store and manage information of a physical therapy clinic specialized in prothesis engineering.

Furthermore, key features revolve around the creation of a user. The user can create an account in the data base expecting to store, manage and see information according to the role that he/she assumes. This provides **organization** and **privacy** when viewing personal information or the medical history of a patient, inter alia.

This information is protected under an email and password that the user selects. The password will be stored encrypted in the database.

Once the user creates an account can *log in* into the system whenever required. When logged in the user can access all information previously stored and/or change information if necessary.

The main entities of the clinic considered when designing the database were:

1. *Clients*: users who access the clinic to use their services
2. *Physiotherapists*: users who work at the clinic as doctors
3. *Biomedical engineers*: users who work at the clinic as engineers

These roles can interact with the database or between each other. For example, a physiotherapist can create a client. But there are also other inert entities with which these roles can interact with:

1. *Machine*: apparatus that can create an exam
2. *Exam*: created by machines (e.g. MRI, CAT scan, EMG, …)
3. *Prosthetics*: apparatus that mimics a body part

Physiotherapists and engineers can order products if not available and change their availability if used or broken, the orders entities are:

4. *Product*: available stock of a product in the clinic
5. *OrderProductPhysio*
6. *OrderProductEngineer*
7. *OrderExamPhysio*

This system intends to facilitate information management for users while preserving an intuitive functionality and maintaining privacy.

1.2. LIST OF FEATURES

1.2.1. GENERAL FEATURES

- Users can sing up introducing an email, password, and role (1= Client, 2= Physiotherapist, 3= Engineer).
- Users can log into the system providing an email and password.
- Users can change their password.
- Users can exit the system (information altered will be saved in the database).

1.2.2. CLIENT ORIENTED FEATURES

- Clients can search a physiotherapist by their id to view their information.

- Clients can delete an exam through the exam id.

1.2.3. PHYSIOTHERAPIST ORIENTED FEATURES

- Physiotherapists can create a client with all his/her information which will be added to the database.
- Physiotherapists can visualize a list with all the clients in the system.
- Physiotherapists can delete a client from the database and their information.
- Physiotherapist can search a client by their id to view their information.
- Physiotherapist can search an engineer by their id to view their information.
- Physiotherapist can export their own information into an XML file
- Physiotherapists can see all the clients associated to their id from and XML file

1.2.4. ENGINEER ORIENTED FEATURES

- Engineers can change their own phone through their id.
- Engineers can create a machine with all its information which will be added to the database.
- Engineers can visualize a list with all the machines in the system and their information.
- Engineers can change a products availability through the product´s id.
- Engineers can visualize a list with all the prosthetics of the same type and their information.

2. DESIGN

2.1. POJOS

The relationships between entities are represented in the ER diagram and must be implemented in the system through coding.

The entities coexist under the following rules when designing the java system:

2.1.1. One-to-one:

When two entities have one-to-one relationships, the foreign key is chosen where to be put in.

This is represented in the entity that has the FK. Its POJO will contain an attribute as a single object of the class of the entity the FK is pointing to.

2.1.2. One-to-many:

When two entities have a one-to-many relationship, the entity that embodies the one, will have a list of objects of the other entity´s class as one of its attributes. The entity that embodies the many, will contain an attribute as a single object of the class of the entity that embodies the one.

2.1.3. Many-to-many:

Whenever two entities have a many-to-many relationship, in both entities POJOs, one of their attributes will be a list of objects of the other entity´s class.

2.1.4. Byte []:

If in any class, an attribute defined as *byte []* is seen is because in the SQL tables a column is defined as a BLOB.

A BLOB is defined as binary large object. These could be images, pdfs, or videos, for instance.

In the system we can find 2 different types of byte attributes for different entities:

- "*Exam_image*": this attribute in the class Exams corresponds to the image that is obtained from a machine and contains medical information of a patient
- "*License*": this attribute can be seen in the classes Physio and Engineer. It refers to a .pdf document of the diploma obtained in the respective professional´s university.

2.2. JDBC

In the projects' description were required certain actions in the JDCB package.

In "PhysioClinicJDBC" the following can be seen:

- Create all database tables → The class "JDBCManager" was made for this.
- Insert new data into at least 2 tables → the system can: create clients, physiotherapists, engineers, and machines.
- Show all elements of at least 2 tables → if chosen, all clients or machines in the system can be showed.
- Search by 1 column other than the primary key in at least 1 table → all prosthetics of the same type are showed. The primary key of this entity is the prosthetics´ id, type is just an attribute of the type *String*.
- Update data in at least 1 table → the engineers´ telephone can be changed by providing an id. The number available of a certain product can also be modified.
- Delete data from at least 1 table → delete client by id, delete exam by ID
- Insert data into at least 1 table with 1 or more foreign keys → clients, and machines are both tables that have 1 FK.
- Search by 1 primary key → the system can search a client, engineer, and physio by their respective ids.

2.3. ENCRIPTION

For the database an AES algorithm encryption was chosen.

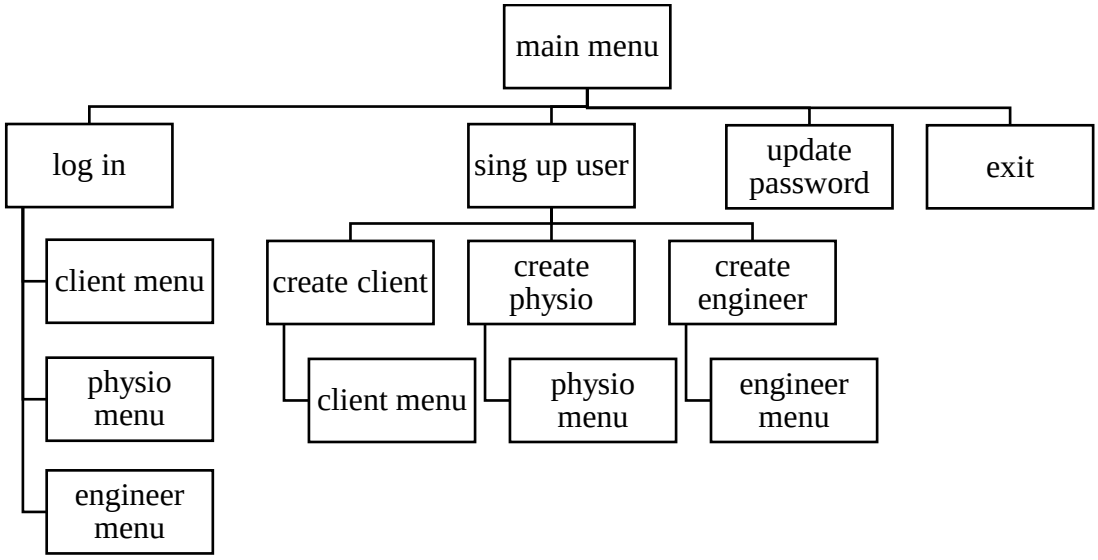
The algorithm basis resides in two strings: the password, that has to be encrypted, and the *secret key*.

- This **secret key** is used in the encryption and decryption of the password. The value was chosen arbitrarily, because with the knowledge of this "word", the passwords of the database could be decrypted.
- The length of secret key could be 128, 192 or 156 bits. A longer secret key provides a larger password margin, meaning there are more possible combinations of bits for the key. This secret key has 192 bits because it has the right balance between a secure system and an efficient one, since the longer the secret key it takes more time for the encyptation to be executed.

For the system has chosen this encriptyon algorithm because of its high security but its efficient performance. This has been widely accepted by its high rate of usage in numerous technological systems of government, military and industrial organizations.

2.4. UI

The menu has the following structure:



In the branch sing up user it can be noted that the different roles menus don't show up directly. This was decided because of how the system is connected to the SQLite tables.

When a new *user* is sign up into the system, the table users is updated, but to the tables *client*, *physiotherapist*, or *engineer* no new users are updated. These tables are the ones implemented in the function that the system carries out. Hence, when the first action that takes place if the database is empty, in case only the user is created, and the system would direct the user to the specific menus, none of the functions would run as the tables *client, physiotherapist* and *engineer* would be empty.

For example, if an engineer wants to create a machine, all its attributes could be saved. But, when having to assign an engineer via his/her id, the table engineer in the database would be empty and the system would throw an error.

Therefore, a new client, physiotherapist or engineer must be created whenever signing up a new user.

2.5. GUI

Even though our GUI is not very extensive it fills its most important purpose. It is giving a more visual first glance that provides a more direct first impression on the client making the system more efficient.

When clicking on an option in the various menus available, it executes the function that the user selected.

4. DESIGN

1. INTRODUCTION

PhysioClinic System has been designed with the aim of integrating advanced technological solutions that support physiotherapists and administrative staff in their daily work. The development of this system is closely linked to the principles of biomedical engineering, where it does not only contribute to the development of rehabilitation equipment and devices, but also to the creation of efficient management systems that ensure high-quality patient care and smooth clinical administration.

The system development involves 3 types of users:

- Physiotherapist
- Engineer
- Client

Depending on the user's role the access to different functions and options will be presented in their own menu.

2. SETUP

2.1. MAIN MENU

The main menu is the primary navigation tool, enabling users to efficiently manage and operate different aspects of the clinic.

2.1.1.LOG IN USER

For the Log-In the user must have an existing account in the system (*view 2.1.2-SIGN IN AS NEW USER)*

When the user enters the password, they are able to see their password by pressing the eye button.

- **IDENTIFICATION**

Once the user has signed in and exists in the database (2.1.2), the system will proceed to the identification process. The user must enter their e-mail address and password, which must match those used when registering.

If the email or password does not match, the system won't let the user access their role menu. The user must introduce the correct email//password in order to access their respective menu [Physiotherapist *(view 2.1.1.2-PHYSIO MENU*), Client (*view 2.1.1.3-CLIENT MENU*) or Engineer (*view 2.1.1.4-ENGINEER MENU)*]

- **PHYSIO MENU**

The physio menu is shown, multiple options are presented which only physiotherapists can perform.

1) CREATE CLIENT

The physio can create a new clients account, in which they must enter their client's information which will be saved in the database as a new clients account for future use (*view 2.1.2.3.1-CLIENTS INFORMATION)*

2) SHOW ALL CLIENTS

The physio has access to all their clients, with this function they are able to see a list of all the clients. In this section, the system will print a list of all their clients without any requirements.

3) DELETE CLIENT

The physio has the power to remove any client from the database. In this section, the system will ask the physio to enter the ID of the client (*view 2.1.2.3.1-CLIENTS INFORMATION)*

4) SEARCH CLIENT

The physio can search up a client in the database. In this section, the system will ask the physio to enter the ID of the client (*view 2.1.2.3.1-CLIENTS INFORMATION)*

Once the ID is entered and the system has verified the format is correct, the information of the client will be shown to the physio.

5) SEARCH ENGINEER

The physio can search up an engineer in the database. In this section, the system will ask the physio to enter the ID of the engineer (*view 2.1.2.3.2-PHYSIO & ENGINEER INFORMATION*) Once the ID is entered and the system has verified the format is correct, the information of the engineer will be shown to the physio.

6) PRINT ME TO XML

The physio can see its assigned clients. In this section, the physio will click the button and be automatically redirected to a file opened on the internet. Once opened, the information will be shown.

7) LOAD CLIENTS FROM XML

8) RETURN

The physio can log out of the system by selecting the return button, which will return them to the main men.

– **CLIENT MENU**

The client menu is shown, multiple options are presented which only clients can perform.

1) SEARCH PHYSIO

The client can search up a physio in the database. In this section, the system will ask the client to enter the ID of the physio (*view 2.1.2.3.1-PHYSIO & ENGINEER INFORMATION)*

Once the ID is entered and the system has verified the format is correct, the information of the physio will be shown to the client.

2) DELETE EXAM

The client can delete an exam from their database. In this section, the system will ask the client to enter the ID of the exam.

- **ID OF EXAM:** Only numbers (int)

Once the ID is entered and the system has verified the format is correct, the exam will be removed.

3) RETURN

The client can log out of the system by selecting the return button which will return them to the main menu.

- **ENGINEER MENU**

The engineer menu is shown, multiple options are presented which only engineers can perform.

1) CHANGE PHONE

The engineer can change their phone, updating the new information in the database. In this section, the system will ask the engineer to enter their ID *(view 2.1.2.3.1-PHYSIO &ENGINEER INFORMATION)*

Once the ID is entered and the system has verified the format is correct, the new phone number will be requested *(view 2.1.2.3.1-PHYSIO &ENGINEER INFORMATION)*

2) CREATE MACHINE

The engineer can create a new machine in which they must enter the machines information which will be saved in the database as a new machine account for future use.

The information required for the correct storage in the database:

- **ID:** Only numbers (int)
- **TYPE OF MACHINE:**
- **DATE BUILD** Following the format yyyy/mm/dd
- **FIRST USE:** Following the format yyyy/mm/dd
- **ID OF ASSIGNED ENGINEER:** Only numbers (int)

3) SHOW ALL MACHINES

The engineer has access to all their machines, with this function they are able to see a list of all the machines. In this section, the system will print a list of all the machines without any requirements.

4) CHANGE PRODUCT AVAILABILITY

The engineer can change the product availability at any given time. In this section, the system will ask the engineer to enter the

- **ID PRODUCT:** Only numbers (int)

Once the ID is entered and the system has verified the format is correct, the engineer must enter a new availability.

- **AVAILABILITY**: only number (int)

5) SHOW ALL PROSTHETICS OF SAME TYPE

The engineer has access to all the prosthetics, with this function they can see a list of all the prosthetics of the same type and all their information. In this section, the system will ask the engineer to introduce the type of prosthetic they want to see.

- **TYPE PROSTHETIC:** String (numbers&letters)

Once the type is entered and the system has verified the format is correct, the engineer will be able to see a list of all the prosthetics of that kind.

6) RETURN

The engineer can log out of the system by selecting the return button which will return them to the main menu.

2.1.2.SIGN UP AS NEW USER

The user must create a new account, this way their information in stored in the database.

- IDENTIFICATION

  In this section, the user will introduce a new email and password which will be their future way of entering the system.

  - **EMAIL**: String (numbers&letters)
  - **PASSWORD**: String (numbers&letters)

If the email or password does not match, the system won't let the user access their role menu.

- ROLE OF USER

After the first process of identification is completed, the system will ask the user to choose their role:

a) PHYSIO
b) CLIENT
c) ENGINEER

- USER INFORMATION

Once the user has entered an email and password and has selected a role, the system will proceed to ask for the users' personal information. Depending on the role they acquire they will have to enter different personal information:

- CLIENT INFORMATION

If the user has acquired the client's role, the information needed will be:

- **ID**: only numbers (int)
- **NAME:** letters (string)
- **PHONE**: only numbers (int)
- **DOB**: following yyyy/mm/dd format
- **CARDNUMBER**: only numbers (int)
- **LARGE FAMILY**: the user must enter 1 if YES and 0 if NO
- **EMAIL**: numbers, letters &special characters (string)
- **ID OF ASSIGNED PHYSIO**: only numbers (int)

- PHYSIO & ENGINEER INFORMATION

If the user has acquired either physio or engineer's role, the information needed will be:

- **ID**: only numbers (int)
- **NAME:** letters (string)
- **PHONE**: only numbers (int)
- **DOB**: following yyyy/mm/dd format
- **CARDNUMBER**: only numbers (int)

- **LARGE FAMILY**: the user must check the box if YES
- **EMAIL**: numbers, letters &special characters (string)
- **SALARY**: Numeric
- **SPECIALITY**: letters, numbers & special characters (string)

When the user's information input is finished, the system will provide a message of confirmation.

2.1.3.UPDATE PASSWORD

For updating the password users must have an existing account in the system *(view 2.1.2-SIGN UP AS NEW USER)*

- IDENTIFICATION & PROCESS

The system will ask the user to introduce their email and current password *(view 2.1.2.1-IDENTIFICATION)*. Once the system has checked the email and password match in the database the user will be able to proceed.

In this section, the system asks the user to enter a new password *(view 2.1.2.1-IDENTIFICATION)*

2.1.4.EXIT

The user can exit the program.