UNIVERSITAT DE BARCELONA

Final Degree Project

**Biomedical Engineernig Degree**

# Development of a software for radiopharmaceutical management in nuclear medicine

Barcelona, 8 de juny de 2022

Author: Beatriz Martínez Ruiz

Director/s: Aida Niñerola Baizán

Tutor: Aida Niñerola Baizán

## Acknowledgements

Transmetre el meu sincer agraïment a totes aquelles persones que han estat al meu costat i m'han donat un cop de mà al llarg d'aquesta etapa.

En primer lloc, a la meva tutora, Aida Niñerola, per haver comptat i confiat en mi per la realització d'aquest treball de fi de grau i per haver-me proporcionat totes les eines i el seu temps per dur-lo a terme. També, vull agrair a la Unitat de Radiofarmàcia de l'Hospital Clínic per oferir-me les seves instal·lacions per realitzar proves. En especial, Francisco Campos, Inmaculada Romero i Manuel Santos per dedicar-me el seu temps en explicar-me el funcionament del seu departament per poder plasmar-ho al programa desenvolupat i finalment, avaluar-lo.

D'altra banda, a tots els professionals de l'hospital que han dedicat una petita part del seu temps per resoldre'm dubtes.

En segon lloc, als meus amics que m'escoltaven i mostraven tot el seu interès al llarg del seu desenvolupament malgrat que no acabessin d'entendre la meitat de conceptes emprats.

Finalment, vull agrair a la meva família per la seva dedicació. Sense ells i el seu suport incondicional no hauria sigut el mateix. A la meva mare, que ha estat al dia sobre cada avanç del treball i m'ha recordat contínuament que tot l'esforç té la seva recompensa. Al meu pare, per aconsellar-me i recomanar-me en aspectes informàtics. I a la meva germana, per amenitzar-me sobretot les últimes setmanes d'aquesta etapa.

## Abstract

Radiopharmaceuticals are radioactive drugs injected intravenously into patients for diagnostic imaging and therapeutic treatments in nuclear medicine. To achieve an accurate interpretation of the results or a correct therapeutic effect, these medicines must be strictly supervised by the radiopharmacy department from the moment their raw material arrives at the hospital until its administration to the patient.

This project aims to develop an user-friendly and efficient software that integrates all processes involved in the radiopharmaceutical management of the *Unitat de Radiofàrmacia de l'Hospital Clínic* which are currently stored in independent Excel documents leading to human errors and complex dose traceabilities. At present, some software is available on the market, but they are too expensive and none of them fulfils all their requested demands. To develop the application, several programming options were evaluated. Finally, the software was developed from scratch in an open-source Python environment, Anaconda. It is mainly based on Tkinter and SQLite packages for creating graphical interface and database communications, respectively. Unfortunately, the external connections with activimeters and the SAP platform did not succeed due to time limitations, lack of knowledge and incompatibility. At last, the prototype was evaluated and the feedback from three radiopharmacists was collected after testing it.

As the software was built from zero, its main advantage is being modelled based on all the radiopharmacy's demands. Although it fulfils currently most of their requirements, there are still necessary improvements to reach its full potential. Besides, it could encourage other hospital departments to introduce hospital information systems.

## TABLE OF CONTENT

## LIST OF FIGURES

**LIST OF TABLES**

## LIST OF EQUATIONS

## LIST OF ABBREVIATIONS

| | |
|---|---|
| **SAP** | Systems, Applications, Products in Data Processing |
| **HIS** | Hospital Information System |
| **GUI** | Graphical User Interface |
| **EHR** | Electronic Health Record |
| **HCUP** | Health Cost and Utilization Project |
| **SPECT** | Single Photon Emission Computed Tomography |
| **PET** | Positron Emission Tomography |
| **DBMS** | Database Management System |
| **AI** | Artificial Intelligence |
| **PM** | Patient Monitor |
| **MMS** | Military Medic Smartphone |
| **SQL** | Structured Query Language |
| **ML** | Machine Learning |
| **macOS** | Macintosh Operating System |
| **MATLAB** | MATrix LABoratory |
| **DTE** | Data Terminal Equipment |
| **USB** | Universal Serial Bus |
| **CSV** | Comma-Separated Values |
| **tk** | Tkinter |
| **Mo** | Molybdenum |
| **WBS** | Work Breakdown Structure |
| **MDCG** | Medical Device Coordination Group |

## 1. INTRODUCTION

When a patient comes to a hospital, a wide variety of data is collected such as the patient's identification, claim data, electronic health records, administrative data, medical records, clinical trials, and research data, among others [1]. Then, it is processed, reviewed and analyzed for many diverse purposes. As information plays an important role in healthcare, good management can provide benefits for patients, medical research, healthcare systems and providers.

Despite the incredible progression in technology, paper documentation is still present in the workplace and is still a habit in enterprises like healthcare. However, during the latest years, we could assist in the progressive computerization of medical information called Hospital Information System (HIS). This concept is related to the component of health informatics which is focused on solving the administrative, financial, and clinical demands of hospitals. So, HIS is essentially created to manage patients and their corresponding information in a centralized way [2]. However, its first application was to cover financial aspects of business around the 1960s. [3]

As the volume of data related to each patient has significantly increased, hospital information management has required the introduction of a clinical software called database. This term is defined as a combination of different types of information such as patient identification, finances, labs, investigations of uncommon diseases, etc., which are set up in a system for easy access, management and updating [4]. These details must remain privy to the public but easily reachable to the healthcare professionals [5].

Depending on which hospital sector we are working on, the features selection will change. For instance, the data stored in an external consultation is quite different from the one obtained in a diagnostic imaging test. In the first scenario, a physician or a health care professional establishes an in-depth conversation with a patient. As a result, the consultant obtains the patient's basic information, symptoms, medical records, etc., which are useful for arranging a future doctor's visit, a blood test, among other medical procedures. On the other hand, the main data acquired in a medical imaging test is a visual presentation of the anatomy or physiology of some organs or tissues which is necessary to reveal some internal structures as well as to diagnose and treat illness [6].

Nowadays, there is a wide range of different databases and each one serves a particular function in a hospital. Some examples are Electronic Health Records (EHR), Health Cost and Utilization Project (HCUP) and PubMed.

Along with assisting in the daily making decisions of healthcare professionals, databases can drive efficiency by providing filters that select a specific type of data to the doctors in order to get easily and quickly access to relevant information. It also promotes an efficient transfer of individual patient data with no variation between different hospital services or establishments. Another important advantage is the ability to track and monitor medical operations with the purpose of offering a better quality in patient care. In addition, some doctors research external databases to enrich their knowledge and determine better diagnoses and treatments [5] On the other hand, the implementation of databases in healthcare also involves some drawbacks. For example, there are concerns related to privacy as they are working with patient information and technical problems due to a possible software failure and cost. [7]

Nuclear medicine is a medical discipline that uses radioactive substances, radiopharmaceuticals, which are radioactive drugs used mainly for diagnostic imaging and therapy in nuclear medicine. They are administrated by intravenous injection, orally or by inhalation and their gamma and positron radiation emitted can be detected outside the body using dedicated equipment to obtain images by Single Photon Emission Computed Tomography (SPECT) or Positron Emission Tomography (PET) techniques [8]. Radiopahrmaceuticals can also be administrated for a therapeutic objective.

The radiopharmacist is the responsible to ensure the correct radiopharmaceutical preparation and guarantee its safety and efficiency [9]. This is of paramount importance as the accurate interpretation of the obtained images or the delivery of the right therapeutic dose, depends on the radiopharmaceutical preparation and the fact that its radioactivity decays over time. In addition, it's also important to correctly document each radiopharmacy operation starting from the moment a radiopharmaceutical is prescribed to its administration to a patient in order to have efficient and fast traceability of each dose [10].

## 1.1 Objectives

The main objective of this project is to develop a medical database integrated with an user-friendly graphical interface in order to be able to track any radiopharmaceutical from the moment its raw material arrives at the radiopharmacy to the patient's administration and the storage of its residues.

This DataBase Management System (DBMS) is for the *Unitat de Radiofàrmacia de l'Hospital Clínic* and it will provide better supervision of its workflow in a centralized electronic system by removing duplication and unnecessary procedures. In addition, this software will be used by all medical professionals involved, such as technicians, radiopharmacists and nurses. Therefore, the more attractive and easier the graphical interface is, the better performance will be obtained.

The following secondary goals have been identified with the purpose of improving the principal aim:

- To establish a connection with the dose calibrators of the radiopharmacy department.
- To establish efficient communication with the HIS of the *Hospital Clinic*.

The last objective is to acquire knowledge and formation of the radiopharmaceutical preparation processes, backend and frontend programming and external connections to accomplish the previous goals mentioned.

### 1.2 The situation of the implementation

My project is focused only on the department of radiopharmacy located in the Nuclear Medicine department of the Hospital Clinic, Barcelona.

Nowadays, paper documentation is still present in the *Unitat de Radiofàrmacia de l'Hospital Clínic.* All the radiopharmacy's workflow is stored in paper and different Excel documents which are classified into four main folders: product orders, product receptions, radiolabelling and prescription. Based on the radiopharmacy stock, orders are executed and saved in an independent unique file in which information about suppliers, workers and stock can be consulted. Regarding the product arrival registration, the data of each different kit, ready-to-use radiopharmaceutical and molybdenum generator are located in distinct sheets meaning more time invested.

Once radiopharmaceutical preparations are required, technicians write down all data involved such as elution used, activity, and volume to quote a few, in a textbook as it is more accessible. Then, it is transcribed into computers as soon as possible meaning hours or days. Finally, patients' doses are dispensed and identified with labels that are printed the previous day and based on SAP information instead of the real one.

In the end, as there is no communication between files, the radiopharmacists need to write data that have been previously introduced in another document so that, errors are much more liable, and traceability of each radiopharmaceutical dose is complex and slow to be performed.

### 1.3 Methods and structure of the project

The elaboration of this project has lasted about six months from December 2021 to June 2022; when it is presented in court. Given the current tendency of telematic work motivated by the situation of COVID-19, it has been developed from home. It has been possible as the elaboration of the medical software is mainly based on the Anaconda software which is an open-source Python language toolkit. However, most of the weekly meetings have been done physically in which a project update has been made. In addition, there have been meetings with the *Unitat de Radiofàrmacia de l'Hospital Clínic* in order to exchange opinions and doubts about the software while it was being modelled.

Through this project memory, you will be familiarized with background, market analysis, conception engineering, detailed engineering, execution plan, technical and economic feasibility, regulation and legal aspects, and conclusion and future directions.

### 1.4 Scope and limitations

The project consists of developing and designing a software to computerize the management of the *Unitat de Radiofàrmacia de l'Hospital Clínic.* It will provide an electronic and centralized data management of the radiopharmaceutical preparation process.

The project's development has been divided into three time periods. The first period was from December 2021 to February 2022, wherein the necessary knowledge for radiopharmacy workflow and Python programming language was acquired. The second period was from February 2022 to May 2022, wherein the radiopharmacy database was developed. The last period was from May 2022 to June 2022, wherein the database was tested, and the final corrections were carried out.

The main limitation of the project is the duration determined by the thesis deadlines, which are set up for June 2022. This implies that some external implementations are not mandatorily considered for being finished at the end of the timeline. For example, the communication between the database and the HIS of the *Hospital Clinic* will be difficult to be established as it depends on third-party help.

Finally, this project doesn't include performing statistical analysis of the data inserted in the database with the purpose of identifying common patterns and trends. In addition, this software is aimed to be manipulated by the *Unitat de Radiofàrmacia de l'Hospital Clínic.* At present, being used in other radiopharmacy units is out of reach.

# 2. BACKGROUND

The creation of computerized databases was driven by the cost drop of purchasing and maintaining computers in the 1960s [11]. As was mentioned before, its principal aim is to be able to allow users an easy data and details extraction by taking and storing information in an organized way. It was not until the late 1970s that the first prototype of the graphical user interface (GUI) was developed by Xerox Palo Alto Research Center [12]. This computer program provides a platform with visual indicator representations such as icons, symbols or images, for creating an attractive interaction between people and computers. As a result of a combination of both technologies, an efficient and user-friendly application is built.

## 2.1 State of art

For the purpose of learning more about their impact on hospital administration and decision-making, some current technologies are described below in chronological order.

### 2.1.1 Database software

In the past few years, the amount of information generated, collected and stored in healthcare is growing day by day. As a result, this medical industry is one of the most important big data's applications. Medical big data daily comes from a huge number of different sources such as administrative data, EHR, clinical registration, etc [13]. Faced with a wide variety of electronic data types, a new technology for searching for potential valuable knowledge was starting to become established in the 1980s. This technique, called data mining, provides a group of tools that explores enormous databases with the objective of finding repeated patterns, tendencies or rules which give us information about the data behaviour. Its main advantages are an easy results interpretation, a cost decrease, better customer support and extraction of valid predictions [14].

A fundamental requisite of any application is security, especially, if the information is related to healthcare. As the hospital is producing a large amount of data every day, database software vulnerabilities are also increasing. For example, we can face insider threats, human error, excessive privilege, etc [15]. In order to improve database security, transparent database encryption was introduced in the late 2000s. It uses an encryption algorithm which converts data from a readable state into a ciphertext of unreadable characters and stores it in a disk. [16]

Another security method, called blockchain, was introduced by Satoshi Nakamoto in 2008. It is defined as an advanced data structure that includes a large list of records called blocks. In addition, this technology has been demonstrated to provide some important benefits in healthcare. For instance, these blockchains can reduce the occurrence of a data breach, improve the safety of medical information and provide an easy patient data tracking by using timestamps. [17]

Related to the abundance of data in a hospital, Cloud Native Computing Foundation was formed in 2015 in response to the huge quantity of organizations and services adopting cloud-native systems [18]. The concept of cloud native is defined as a software architecture pattern based on the main principles of cloud computing such as scale, elasticity and resiliency to quote a few. Its fundamental advantages are independence, resiliency, standards-based, business agility, automation and no downtime [19]. Some examples used in healthcare are Amazon Web Services, Microsoft Azure and Google Cloud Platform [20].

The last innovative technology related to the database concept is the introduction of Artificial Intelligence (AI) motivated by getting access to an abundance of information for learning processing, effective data management and intelligent processing of data [21]. Although the first AI application in healthcare was in the early 1970s which was a program whose objective was to help recognise blood infectious treatments [22], the integration of AI in DBMS has started to be studied in the last years with the purpose of providing help for data-centric decision-making. AI including Machine Learning and Deep Learning has the capability to enhance database query precision and performance, and to optimize system resources [23].

Despite the remarkable evolution of databases from their origin such as journals, libraries or filing cabinets, there are still certain challenges that don't allow DBMS to reach its full potential [5]. For example, the lack of integrated information, the exponential growth of data collected and the limitations related to mitigation [24].

### 2.1.2 Graphical User Interface

As was mentioned before, the first introduction of the GUI was in the 1970s and it has been evolving into more sophisticated user-friendly systems. Nowadays, this technology provides the following benefits: easy use with symbols, icons and images, an easy-to-learn interface due to it doesn't require computer knowledge; attractiveness, shortcuts and multitasking [25].

One of the most worldwide commonly used medical devices in hospitals is patient monitor (PM). They are used to measure, record and display various patient parameters such as heart rate, blood pressure, temperature and respiratory to quote a few. A surprising observation is that the interfaces of PM haven't significantly changed over the last two decades. However, new design proposes have been presented to improve user interaction and prevent potential risks related to human errors. For instance, the use of graphical displays instead of a unique numerical display to represent the discrete vital signs into one or more multidimensional objects. This innovation was designed to make easier the assimilation of the patient's current state to anaesthesiologists. [26]

Another example of a good user-friendly GUI that can save lives and reduce recovery time from injuries sustained is a prototype of a Military Medic Smartphone (MMS). Different from current mobile technologies to support combat casualty care, this MMS is connected wirelessly to physiologic sensors, and it can be operative at the same time in different electronic devices. [27]

Finally, an important challenge in any application is the protection from hostile attacks. According to Kumar (2020), security can be improved by using biometrics identifiers such as fingerprint, face and retina features, etc. Kumar and his team proposed a GUI based on a multimodal authentication technique using face and retina characteristics. They used the Harris corner detector to extract physiological features which later are encoded and decoded by the stenography method. Finally, for authentication, they applied the Multiclass Support Vector Machine technique. [28]

# 3. MARKET ANALYSIS

## 3.1 Sectors to which it is directed

The principal promise of a database system is to provide a better way to access large volumes of data where someone could do collection, searching, analysis, manipulation, etc. Businesses use this information for better decision making, client satisfaction, excellent management, and strategy implementation [29]. Some examples of the fields of end-users companies are banking, healthcare, education, life science, airlines, service, telecommunication, entertainment, and more. In addition, it is expected that database management systems will be introduced in more areas.

Concerning this end-of-degree project, it is directly dedicated to the *Unitat de Radiofàrmacia de l'Hospital Clínic* in order to optimize its management reducing possible errors resulting from manual data entry and use of paper documentation and to accelerate the transmission of more accurate and complete data between different hospital departments. As the development of this DMBS is based on fulfilling the demands of this radiopharmacy department, it can only be used by its workers. However, if the assessment of this database software is positive, it could be introduced to other radiopharmacy departments and adapted to its radiopharmacy needs. For example, possible hospital candidates in Catalonia are:

- Hospital de Bellvitge, Hospitalet de Llobregat [30]
- Hospital Vall d'Hebron, Barcelona [31]
- Hospital Germans Trias, Badalona [32]
- Hospital Universitari de Girona Doctor Josep Trueta, Girona
- Hospital del Mar, Barcelona
- Hospital de la Santa Creu i Sant Pau, Barcelona

## 3.2 Historical evolution of the market

In the early 1960s, the Integrated Data Store (IDS) was the first access-direct DBMS developed by Charles W. Bachman in General Electric. The main reason for the IDS creation was the need of improving the information management in the company where Charles was working. IDS is a database system based on the storage of information in virtual memories. At the present, IDS and its latest versions are still in service, involving a thousand mainframe installations. [33]

At the end of the 60's decade, a new database model was developed by the International Business Machines Corporation [34]. It is called the hierarchical database model and it consists of arranging data with levels and segments whose connections are based on one-to-many relationships [35].

Another example of a hierarchical database is Raima Database Manager Mobile, by Raima, which provides support for making an intelligent decision in microseconds [36].

As the hierarchical model was having some limitations such as flexibility, Charles Bachmann developed the network model in 1969. It is characterized by allowing multiple records to be linked to multiple files and vice versa [37]. For instance, Integrated Database Management System by CA Technologies is an example of this database model combined with the following model. [38]

Although Edgar Frank Codd created the concept of "relational database" in 1970 at IBM, the first relational database management system, Multics Relational Data Store, was published in 1976 by Honeywell Information [39]. Nowadays, it is the most common model employed and it consists of storing data in rows and columns of multiple tables that are connected. Most of these database models are managed by applying Structured Query Language (SQL). This standardized programming language is a useful and easy tool for manipulating, modifying and extracting information from database tables. Another example of this database design is Ingres released by the University of California in Berkeley in 2017 [40].

Finally, the last principal database model is called object-oriented, and it was developed in 1985. Different from the previous designs, it offers more manageable code bases since data is dealt with as one instantly available object [41]. For instance, ObjectStore is a commercial object database created by Ignitetech [42].

In addition, further improvements have been introduced over the past six decades as the technology has been evolving in the fields of computer storage, memory and networks and processors.

For example, the development of a GUI had a relevant impact on the creation of an automatic database. It was promoted by the inefficiency of text-based command-line interfaces for habitual use, in the late 1970s [43]. It has evolved from basic and simple systems to more elaborated ones. Nowadays, GUI is used for a wide range of applications through numerous fields involving sending an email as well as showing the vital constants of a patient. All the current database systems have integrated a GUI.

### 3.3 Future market perspectives

As the use of database management system applications have been significantly increased, the number of challenges has also grown. For example, an important use of database in healthcare is big data analytics which involves investigation and integration of a large number of heterogenous information [44]. These data analysis lead to current challenging issues about capacity and privacy.

In order to overcome these problems, some companies are looking for cloud storage or hosting and an implementation of stronger security layers.

In addition, as machine learning (ML) has been demonstrating some significant benefits, some companies are trying to integrate ML algorithms to aspects of database design and query optimization [45].

### 3.4 Product environment

Most hospital administrations depend on the efficiency and accuracy of healthcare databases. They involve information about labs, finances, billing, payments, and patient identification, among others [5]. Furthermore, some computer companies had developed software that offers a platform in which you can create your database from scratch or a customizable database application in which a few changes must be performed to adapt it for your own needs.

Radiolab is an example of a database software application intended to be applied for the management and traceability of hospital radiopharmacies [46]. It has been developed by Jesús Luis Gómez Perales, chemist specialized in radiopharmacy. This software was created using Microsoft (MS) Access and it provides a large range of functions and modules to achieve an excellent radiopharmacy workflow. In addition, this chemist has also developed more database software for different nuclear medicine applications such as operations of radiation dosimetry (Dosisrad), calculations related to medical diagnostics tests (Nucleolab) and interactions with radiopharmaceuticals and their adverse effects (Datinrad) [47].

Similar to the future result of the end-of-degree project, SiiMA Radiopharmacy is an application that manages the radiopharmacy workflow from the registration of products until their administration to patients. It provides tools for controlling the stock, computing radionuclide's activity and elaborating accurately the final radiopharmaceutical. It was developed by First, a company that was established in 2001 and is specialised in the Information Technology Healthcare market. Moreover, SiiMA offers more medical platforms for workflow management of different hospital departments such as cardiology, radiology, pneumology and physical rehabilitation to quote a few. [48]

Another software applied for hospital management is called "Salus" and it was created by QSOFT, an enterprise created in 1995. It is widely used around the world such as in Spain, Mexico, Morocco, Ecuador, Costa Rica, etc. This software also provides versions for different hospital departments such as gynaecology, physiotherapy, ophthalmology, and more. [49]

Different from the two previous mentioned, MS Access is a program that provides tools with the purpose of designing your database application based on a relational model. It was developed by

Microsoft Office and its first launch was in 1992. Since that moment, there have been eleven versions published in order to offer better characteristics and possibilities. Its main advantage is its lack of complication, in other words, it can be programmed by anyone independently of its knowledge of programming. [50]

## 4. CONCEPTION DESIGN

The initial idea of this end-of-degree project is to develop a software for the radiopharmaceuticals management of the Hospital Clinic which should fulfil the following requirements: user-friendly, multi-user, intuitive, aesthetic and attractive, efficient and error preventive to quote a few. The most important requisite is the fact that it should reflect the actual radiopharmacy workflow.

This section outlines the potential methods for the main block and the two additional incorporations of the project which are represented in Figure 1. Following an analysis of their advantages and drawbacks, the chosen solutions which will be executed are given.

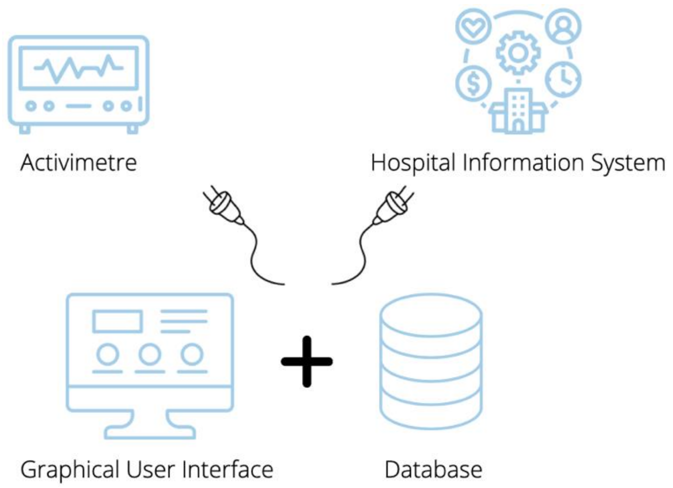



***Figure 1***. *Initial diagram of the project*

### 4.1 Solutions study

Before going in-depth into the study of the solutions, it's relevant to comment that several programs that provide these functions are currently available on the market, some of which have been introduced to the *Unitat de Radiofarmàcia de l'Hospital Clínic*. However, none fulfilled all their requirements and involved a high cost. Furthermore, a few options have been evaluated.

#### 4.1.1 Database and GUI development

The development of a DBMS is based on writing programming sentences that establish relations between organized data within the database and present them to users in order to collect, modify or delete records. With the purpose of achieving this final software program, two different pathways should be considered.

On the one hand, some companies such as Microsoft and Amazon have built some database software which provides the necessary tools for the creation and management of data. They are inspired by three principal elements: data itself, schema and engine. While the logical structure depends on the database schema, the access and modification of data are ensured by the database engine [51]. For this project, the following programs were contemplated.

**Microsoft Access** is an open-source platform which provides user-friendly and intuitive tools so, users are able to design their database applications based on a relational model without any programming knowledge. In addition, it is compatible with any product of Windows Office and SQL Server or Microsoft Azure SQL whose function is storing a large amount of data in a reliable, safe, scalable and easy way. However, it is only available for Windows computers, its capacity of storage is limited to 2GB which is easily achieved leading to a slowing down of its performance and all the database information is stored in a centralized file. [50]

**LibreOffice Base** is also a free and open-source information technology system which is known as the main alternative of Microsoft Access for Macintosh Operating System (macOS) personal computers, and it is compatible with most Microsoft Office documents. While the database application is being built, not more than a unique user can design and modify its structure at the same time producing an important disadvantage. Moreover, users don't need programming knowledge just like Microsoft Access, in order to be able to build data management and this platform also provides a high speed and flexibility [52].

**Power Apps** is an open-source platform, created by Microsoft, which provides an environment to develop a customized application dedicated to collecting business information from different local or online origins [53]. Different from the previous options, it offers a dynamic design suitable for mobile devices or computers and a guaranteed security by allowing user control and a political administration of records and authorization. Besides, this database platform allows establishing automatic functions which reduce the number of manual entries of some known information leading to decreasing the possibility of producing errors. [54]

On the other hand, a database management system can also be created without using specialized software instead from scratch. This option is based on using a determined programming language such as Python, R, C#, SQL and PHP and working with their supplied libraries.

**Anaconda Navigator** is an open-source desktop GUI of R and Python programming languages for performing data science and machine learning applications [55]. It offers a wide range of libraries for different programming functions such as mathematical, artistic, graphical and data management, among others. These packages are compatible with the three following operating systems: Windows, Linux and macOS. In addition, there are available applications whose main role is to work on the general programming process involving the development and representation of the executable code. For example, **Spyder** is a potential scientific environment written in Python whose components are editor, IPyhton console, variable explorer, in which users can consult defined variables, plots, debugger and help [56].

**MATLAB** is a programming and numeric computing platform used by millions of people in order to examine data, develop algorithms and make models. Within its many available packages, the most relevant applications for DBMS development are Database Explorer and App Designer. A user can create their final result, relational database and graphical user interface, respectively, without typing a command. These platforms provide a point-and-click control instead. However, they are characterized by some problems related to scalability, slowness and lack of features to quote a few. [57]

Going deeply in this second pathway, an important parameter whose typology must be analysed is the programming language. It is defined as a group of rules for instructing a computer to perform tasks [58]. Based on the previous options, the two main programming languages considered are Python and R.

**Python** is a programming language whose main function is to develop any type of application. An important advantage is its easiness to be learned and working with because it presents more structure and support for large programs and more error checking. In addition, it is compatible with Windows, macOS and Unix operating systems. [59]

**R** is a programming language mainly used for statistical computing and graphics. It is known for its good performance and available tools in data analysis and its storage facility. Different from Python, it only includes eight packages, but more can be obtained through the CRAN family. [60]

At the beginning of the DBMS development, it is important to be aware of the wide range of the libraries that Python provides for building and manipulating a database and designing a GUI. So, I've searched and asked professionals which have experience with Python about the best options. Once I have all my possible candidates, I've conducted an analysis of their advantages and drawbacks.

**Tkinter** is a Python package destined to be applied to create a graphical user interface. It is available in more operating systems such as Windows and Unix as it is maintained by ActiveState which is an independent software enterprise. Moreover, the most relevant component is the widgets. They are represented as objects, can be configured and follow a hierarchy. [61]

**PySimpleGUI** is also a Python library whose main application is designing a basic GUI. It is an easy way to develop it and it is composed of parts of the four following different packages: Tkinter, pyQt, wxPython and Remi. [62]

**Panda** is a Python library that provides flexible, powerful and easy to manipulate tools mainly used for data analysis [63]. It was created in 2008 by Wes McKinney and it has been spreading and turning into the most popular Python open source. It is well-known for its speed and its extensive collection of functions for almost any data manipulation.

SQL is a programming language which is applied to relational databases for storing data in a group of tables. These tables are connected by foreign keys which are unique identifications from one row in a relational table to another row in a table [64]. The most well-liked libraries from Python are:

**SQLite** is an open-source module which integrates SQL language. It offers the possibility of creating a temporal or a permanent database. However, its manipulation is limited by a single connection time as it is saved in a unique file on a disk [64].

**PostgreSQL** is a potent open-source object-relational database system whose application is based on SQL language fused with many complex features [ad]. It was created as a section of a Berkeley POSTGRES Project whose leader was Professor Michael Stonebreaker from the University of California in 1986 [ae]. Its main potential benefit is the ability to store Python web application data. However, not all types of documents are compatible with this library which implies more software transformation leading to wasting more time.

### 4.1.2 Connection with activimeters

An activimeter is an instrument whose function is to measure the activity of radiopharmaceuticals that will be administrated to patients. Therefore, it's important to present great accuracy and precision when this variable is being manipulated in order to guarantee patient safety. Establishing a connection between this end-of-degree DBMS and dose calibrators of the Hospital Clinic might reduce the possibility of producing human errors in the moment of writing the radioactivity obtained.

For this reason and knowing that the activimeters have different serial ports as can be observed in Figure 2, an analysis of three options for connection has been conducted.


***Figure 2**. Serial ports of one of the activimetre*

**RS232** is a type of serial data transmission called a serial connection. It is a standard 9 pin DB9 cable which transforms positive or negative voltage into a binary language (1 or 0, respectively). There are two types of RS232 devices:

- Data Terminal Equipment (DTE) such as a computer and activimeter.
- Data Communications Equipment such as a modem.

In the case of dealing with two DTE or two DCE, it's necessary to know that they cannot communicate with each other so, a reverse (null-modern) cable is needed to connect the devices. As computers and activimeters are DTE, their RS232 port is male so, a male-to-male adaptor should be considered [67]. In Figure 3, we have a DB9 female (left) to male (right).

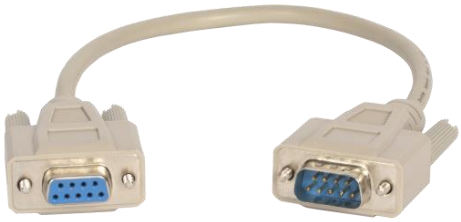


***Figure 3**. RS232 Cable, DB9 female to DB9 male*

RS232 is usually used as the connector between computers and printers as it's widespread and inexpensive. However, the speed at which data can be transferred is slow, around 20 kilobytes per second, and the maximum length of a cable is about 50 feet as wire resistance and voltage drops become an issue with a cable longer than this. [68]

**USB (Universal Serial Bus)** is the most common serial port whose main function is to establish communication between different devices through a single type of port and cable. Its original purpose is to exchange information between two devices. Nowadays, it is also used to charge devices. There are many types of USB connectors, but we are not going to analyze them in-depth. [69]

**USB Type B**, also known as Standard-B, is a type of USB whose port shape is rectangular on the bottom and round on the top. Its main application is linking computers to large devices such as printers and scanners. [70]

### 4.1.3 Communication with the hospital information system

A HIS refers to a centralized source of information related to a patient's electronic medical record and the hospital's operational management. This hospital database software collects, stores, manages and sends healthcare data to each department within a hospital [71].

In this end-of-degree project, one of the secondary goals is to establish a communication with the SAP of the *Hospital Clinic* with the objective of extracting fundamental data of patients who will betreated in the Nuclear Medicine department. Two options have been taken into account in order to achieve this aim.

**Pynetdicom** is a Python library focused on data access within a worklist from a HIS. The concept of a worklist is defined as the structure of information associated with a particular task. Based on the hospital server information and the Application Entity title, and using the DIMSE C-FIND service, we are able to create a connection and a query in order to obtain the necessary information from SAP. [72]

**Comma-Separated Values (CSV)** is a text file that stores data in a tabular structured format [73]. As it is compatible with Python, it will be used as a DBMS input if the connection with SAP is not achieved. This method is based on the actual procedure executed in the *Unitat de Radiofàrmacia de l'Hospital Clínic* which consists of copying the patient's information and pasting it into a CSV file. The CSV data is filtered, and the result gives information about the number of radiopharmaceuticals that need to be prepared.

### 4.2 Proposed solution

Based on the main goal of this project and its further implementations, the best choice is to develop a Database Management System from scratch. This option will provide us more freedom in the fields of introducing connections to external devices or applications, the possibility of scaling it into another platform and the compatibility with any operating system, such as Windows, macOS and Linux, on which the software will be manipulated, to quote a few.

Following this decision, the programming language selected was Python mainly for the same reasons as before. Its wide range of functions eases the incorporation of any improvement such as USB communication or a DICOM reader. In addition, most of the errors or doubts related to some Python code or function can be found by asking accurate and correct questions as it has been one of the most popular coding languages over the last years. The last reason which concludes with this dilemma was based on self-experience and professional recommendation. Consequently, Spyder will be the environment where the Python code will be run.

Among the different Python libraries analyzed, Tkinter will be used to design the GUI of the DBMS because it offers a wide variety of widgets such as labels, buttons, combobox, etc. In addition, as it was briefly taught in previous degree classes, the training before the DBMS development will be easier and shorter. On the other hand, we will use SQLite3 in order to access DBMS data. This choice was motivated by a large number of available functions, its intuitive and simple code and its popularity in the field of database management systems.

As a connector between the DBMS and the activimeters, we have chosen a USB to RS232 Converter-DB9 which allows us to connect the activimeters to a laptop or computer. It is composed of a USB port destined to be connected to a computer at one end, and a DB9 RS232 port at the other end which will be coupled to the activimeters of the *Unitat de Radiofàrmacia*.

Related to the communication between the end-of-degree DBMS and the SAP of the *Hospital Clinic*, the selected option has been the utilization of the Python library called Pynetdicom. It will help to reduce the amount of unnecessary work for the employees of the radiopharmacy department and it is expected also to decrease the possibility of human errors.

### 4.3 Alternative solution

In the case that the development of the Database Management System couldn't be executed from scratch and by using Python and its corresponding packages, an alternative solution will be the utilization of database software, specifically, Power Apps. This decision is based on the study and balance of the benefits and drawbacks that each option can provide.

This Microsoft platform has a huge potential in the field of DBMS construction as it provides an environment compatible with any operating systems such as Windows, macOS or Linux which can be computers or mobile devices. Another advantage is that it doesn't require any professional knowledge of programming because it offers predesigned templates, simple functions and tutorials about how it works. In addition, several users can manipulate the application in play mode but only one can modify it at the same moment.

With respect to the connection with activimeters, there is no alternative candidate as they don't have more serial ports which could allow us to establish a bridge between them and the DBMS. The main reason is that they were bought around ten and fifteen years ago so, they were created based on that period's demands and trends.

Finally, if the hospital server connection and the Pynetdicom query aren't accomplished due to external factors such as time and third-party help, the radiopharmacy department will copy and paste the patient's information from SAP into a CSV document which will be an input of the DBMS. Therefore, the application will filter the necessary data and execute some required modifications in order to be able to manipulate it within the software.

# 5. DETAILED ENGINEERING

Before starting to describe how this end-of-degree software has been developed, a deeper and detailed explanation of the radiopharmacy workflow is made starting from the moment a product is ordered to a single dose is injected into the patient. Once understood and assimilated all its requirements and its workflow, the way of expressing them into the project DBMS is managed with the objective of achieving the best-customized application which fulfils all the radiopharmacy's demands.

## 5.1 Radiopharmacy's state of art

A radiopharmacy workflow starts when a patient needs a diagnostic or therapeutic imaging test and automatically, a medical prescription for a radioactive tracer is ordered. Then, a radiopharmacist looks at the stock and executes an order based on the products needed for the exam. These products can be classified into three groups depending on their properties: ready-to-use radiopharmaceuticals, generators and non-radioactive kits. Some days later, the merchandise arrives at the hospital, and it is registered. Once the patient has arrived at the nuclear medicine department, a nurse orders his radiopharmaceutical dose.

Depending on his exam, the radiotracer elaboration changes. On the one hand, some radiopharmaceuticals received are ready-made, so that a dose extraction is the only step needed. However, they are sometimes single-dose vials. On the other hand, other radiotracers are prepared inside the radiopharmacy. They are composed of a non-radioactive kit and a radionuclide, and in autologous preparations, additionally, cellular components such as leukocytes, erythrocytes and platelets. Most kits are combined with $^{99m}Tc$ elutions that are previously extracted from a Molybdenum generator. Finally, a single radiopharmaceutical dose is extracted which will be administrated to the patient by a nurse.

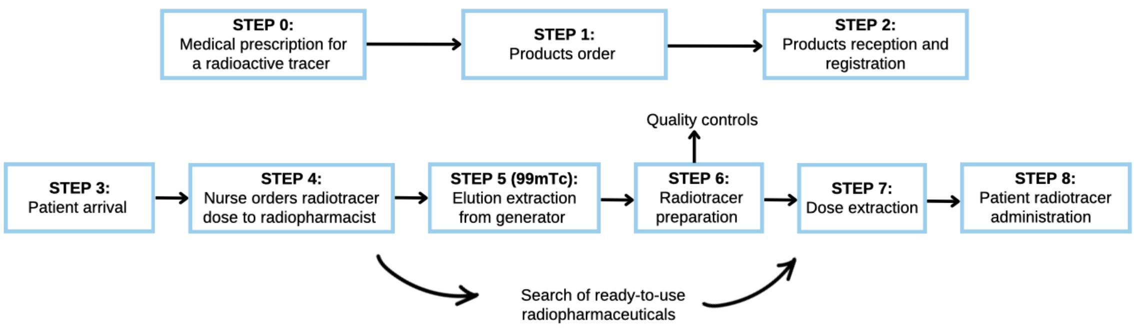



***Figure 4**. Unitat de radiofàrmacia's workflow*

## 5.2 Database identification and development

Within a database management system, data is organized as independent or dependent tables in which end-users can extract, modify, add and delete registers with the help of the GUI. Based on the documentation that is currently used in the radiopharmacy department, these database tables can be divided into two main groups depending on their utilization: master and transactions. The identification of them is relevant to comprehend how the system communicates with the database.

Regarding the master tables, they are firstly designed and include all the general information that will feed transactions tables. In this end-of-degree DBMS, the entities described are products which will be ordered, suppliers, workers that are currently in the radiopharmacy department, kits, generators and radioactive materials which will be manipulated, examinations performed in nuclear medicine and the relation between products requested and received.

On the other hand, while using the DBMS application, the information of the different tasks performed is recorded in transaction tables whose entries usually have a foreign key to master records. A foreign key is a field which is used to link two tables together. Related to the DBMS software, the activities involved in this type of table are orders, registrations of kits, generators and radioactive materials which have arrived at the hospital, elution, stock, quality controls, patient prescriptions extracted from SAP, preparations of any type of radiopharmaceuticals and dispensations.

### 5.2.1 Python code

As was mentioned in the Proposed solution, the development of the DBMS software has been performed in the Spyder environment by using Python as the programming language. To create all these database tables, the *Pandas* library was firstly used upon external recommendation but, as it didn't fulfil all the required needs, the *SQLite3* package was finally chosen and specifically, the following python functions were only utilized for creating the database as a connection object [74]:

- *conn=Sqlite3.connect():* to create a connection object that is the database, as conn.
- *c= conn.cursor():* to generate a cursor that will help us to execute SQL statements, as c.
- *c.execute():* technique to perform SQL commands.
- *conn.commit():* to carry out changes.
- *conn.close():* to close the database connection.

In addition, a SQL CREATE TABLE Statement was elaborated for each database table. Within the SQL command, the name of the table was first specified and later, all its columns were described as can be seen in Figure 5. In addition, the characteristics of each column parameter are defined on its right side whose first variable corresponds to the datatype and it could be text, integer, varchar, DateTime and float to quote a few, and its second variables are constraints as primary key, not null and unique. Finally, the foreign keys are specified at the end of the SQL Statement, in which the column parameter involved is mentioned and then, the name of the implicated column and table to which is linked is indicated.

```
c.execute ("""CREATE TABLE IF NOT EXISTS kitsreg (
        id text PRIMARY KEY,
        id_order text NOT NULL UNIQUE,
        id_kit text NOT NULL,
        arrivalexpected text,
        arrivaldate text,
        Numbatch text NOT NULL,
        expirationdate text,
        Numvials text,
        Numpickupnote text,
        observation text,
        FOREIGN KEY (id_order) REFERENCES orders(id),
        FOREIGN KEY (id_kit) REFERENCES kit (id))""")
```

***Figure 5****. SQL CREATE TABLE Statement of the table corresponding to the registration of kits*

Regarding the master tables, as they were generated and filled before manipulating the DBMS software, they are CSV documents which are introduced in the database by using the next python functions: *open()* and *csv.reader()*. Moreover, they can be modified by only administrators, and it is explained in the Start-up screen.

### 5.2.2 Relational schema

As all the tables mentioned are connected with each other, a relation schema has been carried out by using LibreOffice software since it is compatible with macOS operating system. It gives us a general visual representation of the design and structure of the database in which each table is symbolized by boxes and their respective associations by lines. This relation schema can be observed in Figure 6.

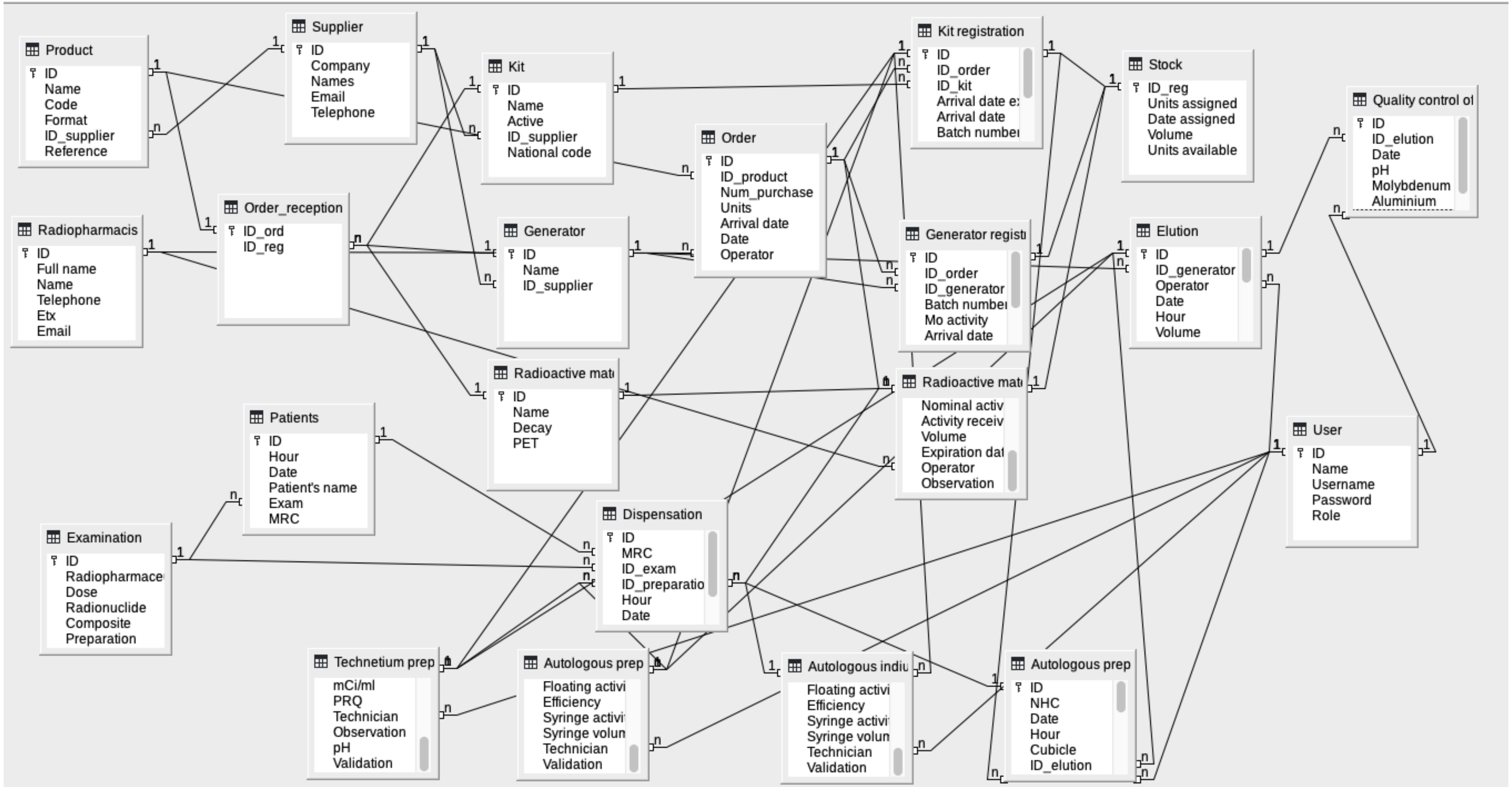



***Figure 6.*** *Database schema in LibreOffice*

### 5.3 Graphical user interface development

The DBMS application for radiopharmaceuticals management has been developed based on the radiopharmacy's workflow that was explained in Radiopharmacy's state of art . Spyder from the Anaconda platform have been used as the environment where the Python code has been elaborated and run.

Moreover, Tkinter (tk) package form Python was finally selected as it offers a wide range of functions for different components in order to create a fully equipped GUI. Among all the available widgets, the most common are [75]:

- *tk.Label():* to show a static text.
- *tk.Entry():* to allow user to introduce information.
- *tk.Button():* to head user for another window or to execute an action.
- *tk.Combobox():* to generate a drop-down list.
- *tk.Treeview():* to show information in a tabular way.
- *tkr.CheckButton()*
- *ImageTk.PhotoImage()*
- *tk.messagebox():* to create a dialog box in order to show some specific information.

In order to access the functions offered by the application, a worker of the radiopharmacy department must sign in which will give him a software role: administrator or basic user. Then, he is able to perform any task from the first page such as orders, receptions, elution, preparation of radiopharmaceuticals, quality controls, dispensations and traceability.

Following the process of a radiopharmaceutical preparation, Figure 4, first a product order is performed followed by its arrival registration. When the patient arrives, depending on the radiotracer needed, it could be necessary to eludate the molybdenum generator, combined $^{99m}$Tc with a radioactive isotope or directly use a ready-to-use radiopharmaceutical. Regarding radiopharmaceutical preparation, there are two types: autologous and technetium. Once a single dose is injected into a patient, traceability of patients involved with a specific kit, generator or radioactive material can be carried out. Moreover, additional activities that can be executed are performing quality controls of molybdenum generators or radiopharmaceutical preparations, stock inquiries and generating reports of activity received. This process is summarized in the following diagram:

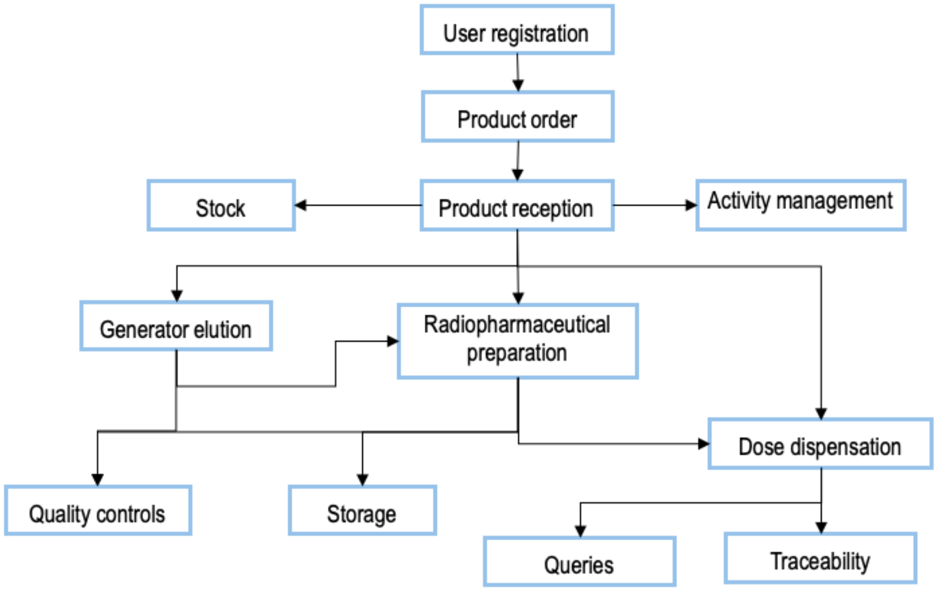



***Figure 7.*** *Workflow diagram of the DBMS software*

This application is divided into 13 blocks as can be observed in Figure 7 and each one is destined for a specific task in the radiopharmacy's workflow. Along with this chapter, all the different blocks will be explained in terms of front-end and back-end. For more details, an in-depth description and a visual representation of all the application elements can be found in User Manual.

### 5.3.1 User sign-in

- **Front-end**

Once the application is opened, the first window was created by using *tkinter.Tk* whose function Config allow modifying certain characteristics of widgets. It consists of two *entry boxes* that are dedicated to introducing the username and its password, two *labels* which inform users where they should write down their login data and a *button* which checks if the user has been previously registered. When the button is clicked, a message box will show up telling if both parameters are correct or not. As same as all software screens, there are two images that represent the logos of the *Universitat de Barcelona* and the *Hospital Clinic*, and today's date at the top.


***Figure 8.*** *Sign-in window and a message box informing if the login data isn't correct*

- **Back-end**

To check if the information introduced in both entries corresponds to the login data of a register of any radiopharmacy's worker, a for loop was generated in which there is a comparison between the parameters introduced and the records of the SQL table named users. In addition, there is a condition established that if both data coincide, the user is able to manipulate the application. Finally, depending on the software role, a specific menu widget will show up in the next window.

### 5.3.2 Start-up screen

- **Front-end**

This second block consists of a visual summary of the different tasks that can be performed in this DBMS software, Figure 9. As depends on the user sign-in screen, a Toplevel() function was used whose input is the name of the user sign-in screen. Each function is represented as a *Button* that will head users to its corresponding window which will be explained in detail in the next sections.

***Figure 9.*** *Start-up screen*

As was introduced in the previous section, depending on the user role, a widget Menu is generated at the screen top. For administrators, the different serviceable options are summed up in Figure 9. It is composed of 5 different submenus generated with the help of the function *Tkinter.Menu()* and added to the Menu bar by using *add.cascade().* By clicking each submenu, a list of different options will dropdown.

Following the right-to-left order, the function of *BBDD* is to allow the user to add or modify records within the SQL tables. For both cases, the names of all the SQL tables are obtained and displayed in a *tk.Combobox*. Once the table is chosen, its column parameters are also displayed in a *tk.Label* and *tk.Combobox*. Finally, the new value or record is written down and imported to the SQL table.

As same as before, *Registres* submenu give administrators permission to eliminate records that had been wrong introduced. So, once the table and column names are selected, the record involved will be eliminated.

Next, the *Worklist* submenu has an *tk.Entry* and *tk.Combobox* in which users can upload and correct the document containing patient information from SAP. Besides, in the *Validació* submenu administrator can validate the preparation elaborated but not verfied in the radiopharmacy from the different preparation SQL tables. They are displayed in a *tk.Combobox.* Finally, the last submenu, Configuració, is destined to quit the application and is the only one available for the basic user.

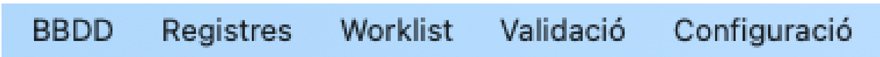



***Figure 10.*** *Menu widget for administrators*

- **Back-end**

To link each button to a specific function within Python script, there are two methods: the parameter *command* and the Binding function. As it didn't need to be complex, the first function was used since it is simpler and easier.

Regarding the widget Menu, most of the submenus demand to communicate with the SQL database. Therefore, SQL statements were written down such as SELECT, UPDATE, INSERT INTO and DELETE. For example, in order to extract the names of all SQL tables, the following command was used: c.execute("SELECT name FROM sqlite_master WHERE type='table';"), in which we ask for the names of the components from *sqlite_master* that are tables. On the other hand, the *data.description* function is utilized to obtain the column's names.

With respect to the UPDATE SQL command, it is useful for modifying records of SQL tables once the names of the column and table involved and the new value are defined. Different from the previous statement, the objective of INSERT INTO is to introduce a new complete record into SQL tables instead of a unique value. On the contrary, the DELETE statement is utilized to eliminate a record, usually, through its identification.

In addition, the *Worklist* submenu exclusive is for CSV documents that are saved in a specific folder and follow a determined format as they feed a SQL table that is already established. It is explained in more detail in the User Manual.

### 5.3.3 Product order module

- **Front-end**

This block consists of a form, Figure 12, which is filled when there is no stock of a product required for a radiopharmaceutical elaboration and whose fields are: product, supplier, code, reference, format, number of the purchase, units, delivery date and petitionary. Starting from the upper side, there are a *tk.Combobox()* which will display all the products that can be ordered and then, four empty *tk.Label()*, which will be filled once the product is selected and the Button *Actualitzar* is clicked. Two *tk.Entry()* are located below in which the user can introduce information about the product order such as the number of the purchase and units needed. Moreover, the delivery date is chosen with the help of *tk.Calendar()* module which provides a Calendar widget as Figure 11. The last field corresponds to a *tk.Combobox()* in which a list of the allowed radiopharmacy workers.


***Figure 11****. Calendar widget*

Regarding the widgets below, there are three buttons: one for saving the information of the form, the other one for generating an order report and the last one for creating an Excel summary of the orders executed between the dates that are selected with the help of the Calendar widget.


***Figure 12.*** *Order form*

- **Back-end**

To create a Combobox, there were two options studied: *tk.Combobox()* and *Gtk.Combobox()*. As the second widget needed an extra Python package whose installation didn't succeed, Combobox from the Tkinter library was finally used. The information displayed is extracted from the product and radiopharmacist SQL tables by using the SELECT statement. Moreover, for the fields that are automatically filled, the SELECT statement is also written down but with an additional condition: WHERE, which restricts the output information.

Once the order questionnaire was filled, all data is inserted into a specific SQL table by using the INSERT INTO statement but there is one requirement: the petitionary field needs to be chosen. Automatically an ID number is generated based on the last order record. Besides, an order report can be also developed by importing the openpyxl package which let us use an Excel template and only fill specific cells. Firstly, the template file needs to be called and activated by using the *load_workbook()* and *.active* functions, respectively. Then, each record value is assigned to a specific Excel cell and finally, a new document is created with the *.save()* function.

Finally, the order summary is generated by incorporating a SELECT statement coupled with a JOIN statement which allows combining rows from different tables based on a common parameter, and the WHERE + BETWEEN condition for limiting the products by the delivery dates.

### 5.3.4 Product reception module

- **Front-end**

This block consists of *Toplevel* window which is composed of three images that represent each product type and three buttons that will head users for the following reception forms:

1. Registration of kits
2. Registration of generators
3. Registration of radioactive material

All three forms use a *tk.Combobox()* for dropping down a list of the different products that should be received, several *tk.Entry()* widgets so that users can introduce information about the product reception and some *tk.Calendar()* buttons destined to choose a date.

In addition, two buttons are located at the bottom of each window whose main function is to save the form data into its respective SQL table and to generate a summary of the received kits, generators or radioactive materials between two specific days.

- **Back-end**

To restrict the selection of some specific products ordered that will be displayed in the first drop-down list, the SELECT statement was combined with INNER JOIN statements that filter those rows that have common characteristics with determined tables and the WHERE NOT EXISTS statement which only extract the records that don't coincide with the secondary SELECT statement. In Figure 13, we can observe an example.

```
c.execute("""SELECT orders.id, orders.arrivaldate, product.name FROM orders
          INNER JOIN ordreg ON orders.id_product=ordreg.id_ord
          INNER JOIN product ON orders.id_product=product.id
          INNER JOIN kit ON ordreg.id_reg=kit.id
          WHERE NOT EXISTS (SELECT kitsreg.id_order FROM kitsreg WHERE kitsreg.id_order = orders.id)
          """)
```

***Figure 13.*** *SQL statement for asking for product that should be received*

Different from the two first reception forms, the field of the received activity in the radioactive material acquisition forms is an automatic parameter. It is computed by following the equation of the radioactivity decay, Equation 1. Each material has a unique decay constant, also known as a half-life, that is saved in the SQL table associated with the radioactive material.

$$N = N_0 * e^{\left(-\frac{ln(2)}{\lambda}\right)*t}$$

***Equation 1.*** *Equation of the radioactivity decay*

Related to the storage button of the data filled with the form, an ID number for each reception is created based on the last record and the type of product acquired. The letter K is used for kits, G for generators and MR for radioactive material. This discrimination let users visually differentiate which product they are working with. In addition, each reception row is connected with a particular order so, it will help us to execute fast and east traceability.

Finally, as done in the order screen, a summary report can be created that will come up within the folder Reports in which all the products received between two dates are introduced.

### 5.3.5 Stock module

- **Front-end**

As same as the previous block, this *Toplevel()* window is divided into three distinct sections for each product type reception as their corresponding information filled is different from each other. All three follow the same screen structure except for small details, Figure 14. Therefore, it is composed of a *tk.Treeview()* widget which shows all the relevant data of each available kit, generator or radioactive material and a button whose function is to update the Treeview data.


***Figure 14.*** *Ready-to-use radiopharmaceutical stock's screen*

Regarding the visual dissimilarities, there are two principal aspects. The first one is related to the process of subtracting empty or expired products as some of them are single-dose and the other are multi-dose vials so, it's difficult to keep a good track of all the dispensed volume and activity. On the other hand, the number of *tk.Treeview()* tables also differs as in the case of radioactive material, there are two subgroups depending on the medical imaging technique applied: PET or, SPECT and gammacamera.

In addition, the column of available activity is an additional and important data provided by the software that the current Excel documents don't have. It gives us information about how much activity is left from a radioactive isotope.

- **Back-end**

To filter the required information which will be displayed, the WHERE condition was established that consists of accepting only the products that have available units in the stock SQL table.

Regarding the kit stock, the subtracting process is executed manually by entering a number that will be removed from the current value. On the other hand, the columns of the available activity and the number of elutions from the generator stock are computed automatically by following a mathematical equation. For the second parameter, a sum of all the elutions that share the same generator ID is performed. As radioactivity decays over time and it is important to subtract the elution activity at its precise moment, a for loop was created in which the Equation 1 is applied from the calibration instant until the first elution is executed and then, the technetium activity is subtracted. So, this process is repeated for the following elution performed until there is no more activity in the Molybdenum generator or elution left.

Concerning the last product type, once the dose is dispensed, a unit of this radiopharmaceutical is automatically removed except for PET radiopharmaceuticals. These last ones are treated as generators since they are vials which contain more than one dose, and their retirement variable is their activity. Therefore, once the radioactivity is null, they are immediately removed from service.

### 5.3.6 Activity management module

- **Front-end**

This block is composed of two *tk.Combobox()* in which all the radionuclides and generators used are summarized. In addition, there are two *tk.Calendar()* buttons with which users can choose the days between a report of the activity management associated with the radionuclide picked is developed. Finally, the button which generates the reports is located at the bottom of the window.

- **Back-end**

Since some products share the same radioisotope, they are linked to their respective radioactive atom within a master table that is created before manipulating the radiopharmaceutical management software. Once the radionuclide is specified and the *"Informe"* button is clicked, a SELECT + WHERE statement is run whose objective is to filter the products received by the dates and radioisotope column.

### 5.3.7 Generator elution module

As the technetium radioactive half-life is too short, its storage and transport are nearly impossible and too expensive. Instead of that, a molybdenum (Mo) generator is supplied to the hospital as its decay constant is around 66 hours. Therefore, technetium is extracted by passing the saline solution through the molybdenum generator.

Regarding its graphical interface, it consists of a *Toplevel()* window in which two *tk.Combobox()* are located at the top of the questionnaire and correspond to available generators and radiopharmacy workers' names. Moreover, a Calendar widget and three *tk.Entry()* are placed below on which the following fields are based. These last items will show up once the previous parameters are filled and the *"Càlculs"* button is clicked.

On the right side, there is a section designed for the molybdenum generator that has been previously selected in order to give some information to users. These fields will manifest by pressing the button *"A"*. Besides, a quality control form is provided in which users can introduce the information that they just measured and next to it, there are some guides about the values that should be acquired.

- **Back-end**

Different from the previous forms, some numbers that show up on the window are based on mathematical equations. First, the molybdenum activity is converted from GBq to mCi with Equation 2. To compute each variable, the following formulas have been used:

1. Theoretical activity corresponds to the activity that should be eluted from the Mo generator. Before calculating Equation 3, a factor is estimated depending on the elution volume.

$$\begin{Bmatrix} v < 0{,}19ml & Factor = 0 \\ v > 0{,}19ml & Factor = 0{,}00156 * v^3 - 0{,}0419 * v^2 + 0{,}3753 * v - 0{,}1 \end{Bmatrix}$$

2. Elution efficiency is the ratio between the theoretical and the real activity eluted, Equation 4.
3. Weight ($^{99}$Tc + $^{99m}$Tc), Equation 5.
4. Hours since the calibration process represents the difference in hours between the elution execution and calibration process, *HCal.*
5. Hours since the previous elution, *Hel.* By default, this parameter is equal to 48 for the first elution and for the result, it is the period time between consecutive elutions.
6. Molybdenum activity, Equation 6.
7. Specific activity (mCi/µg) gives the relation between eluted activity and weight, Equation 7.
8. Mole fraction corresponds to the technetium concentration, Equation 8.

$$A_{theo} = Factor * 0{,}87 * A_{Mo}(mCi) * e^{-0{,}0105*Hcal} * 1{,}1 * (1 - e^{-0{,}106*Hel})$$

***Equation 2.*** *Theorical activity eluted (mCi)*

$$\mathrm{A_{Mo}(mCi)} = \mathrm{A_{Mo}(GBq)} * \frac{1000}{37}$$

**Equation 3**. Units conversion from GBq to mCi

$$Eff = \frac{A_{theo}}{A_{eluted}}$$

**Equation 4**. Elution efficiency (%)

$$W = 2{,}209 * 10^{-3} * A_{theo} * \frac{(1 - e^{-0{,}106*Hel})}{e^{-0{,}0105*Hel} - e^{-0{,}1155*Hel}}$$

***Equation 5.*** *Technetium weight (µg)*

$$A_{mob} = A_{Mo}(mCi) * e^{-0,0105*Hcal}$$

**Equation 6.** Molybdenum activity (mCi)

$$A_{spec} = \frac{A_{theo}}{W}$$

**Equation 7.** Specific activity (mCi)

$$FM = 0,86 * 0,0105 * \frac{e^{-0,105*Hel} - e^{-0,1155*Hel}}{(1 - e^{-0,0105*Hel}) * (0,1155 - 0,0105)}$$

***Equation 8.*** *Molar fraction*

Regarding the quality control window, its SQL table is linked to the elution table in order to perform good traceability of these tests.

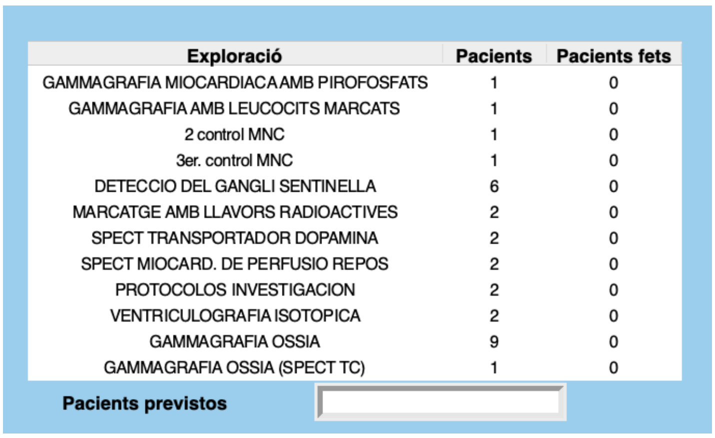
### 5.3.8 Radiopharmaceutical preparation module

- **Front-end**

This block consists of a visual summary of the different radiopharmaceutical preparations that can be performed:

- Preparations with technetium
- Autologous preparations with technetium and leukocytes
- Autologous preparations with indium
- Autologous preparations with technetium and erythrocytes.

On the left side, there is a brief list of the medical exams that will be performed at that day, Figure 15. As these preparations are multi-doses, a *tk.Entry()* is provided to introduce the number of patient expected to be included. Consequently, future users will be able to make a query about how many patients are left.



| Exploració | Pacients | Pacients fets |
|---|---|---|
| GAMMAGRAFIA MIOCARDIACA AMB PIROFOSFATS | 1 | 0 |
| GAMMAGRAFIA AMB LEUCOCITS MARCATS | 1 | 0 |
| 2 control MNC | 1 | 0 |
| 3er. control MNC | 1 | 0 |
| DETECCIO DEL GANGLI SENTINELLA | 6 | 0 |
| MARCATGE AMB LLAVORS RADIOACTIVES | 2 | 0 |
| SPECT TRANSPORTADOR DOPAMINA | 2 | 0 |
| SPECT MIOCARD. DE PERFUSIO REPOS | 2 | 0 |
| PROTOCOLOS INVESTIGACION | 2 | 0 |
| VENTRICULOGRAFIA ISOTOPICA | 2 | 0 |
| GAMMAGRAFIA OSSIA | 9 | 0 |
| GAMMAGRAFIA OSSIA (SPECT TC) | 1 | 0 |



***Figure 15.*** *Treeview of medical exams within the preparation window*

Regarding the different preparation buttons, they will head users to its corresponding form that needs to be filled. All of them follow more or less the same graphic design as they consist of mixing two or more components such as non-radioactive kits, radionuclides and cellular components. They are mainly composed of *tk.Entry()* which is employed to insert new information, Calendar widgets for choosing dates, *tk.Combobox()* useful for connecting new radiopharmaceutical elaborations with products from the radiopharmacy's stock, and a button that will save all the questionnaire data in the SQL database. Moreover, the font color is changed once the last two columns coincide.

As is commented in the next section, concerning the quality controls performed in radiopharmaceutical preparations, they are represented as Radiochemical purity (PRQ) in technetium mixtures and efficiency in autologous.

- **Back-end**

To generate the medical exams' list, two SQL statements were written, SELECT and SELECT DISTINCT, whose main difference is that unique values are only obtained from the second command. Then, two for loops were created in order to evaluate each register and an if condition was established to sum units when both explorations match. Moreover, to change the number of the third column, the Bind method was used whose input event is "<<TreeviewSelect>>" which returns the position of the row selected and then, the new value is added by using the UPDATE statement.

On the other hand, a brief explanation about the preparation forms is that the SELECT + WHERE statement is utilized to restrict only the kits with available stock units and the elutions with radioactivity, and the INSERT INTO command applied to save the form data into the SQL database.

### 5.3.9 Quality controls module

- **Front-end**

This block is aimed to allow users to make a consult about the quality controls that have been performed on molybdenum generator's elutions and radiopharmaceutical elaborations. All this information is displayed in a *tk.Treeview()* within a specific window for each category.

- **Back-end**

As was explained before, these tests about product quality aren't executed in all the samples, so there are some parameters or tables that determine which product has been evaluated. By assuming this discrimination and using the WHERE SQL statement, the Treeview tables are filled.

### 5.3.10 Storage module

- **Front-end**

This block consists of showing the active elutions, in other words, which still have radioactivity, and the preparations done. Regarding the preparation storage, they are revealed according to a specific day which is chosen with the help of the Calendar widget and the update button. However, a default condition is established that displays the preparation elaborated on the same day as the consult is being made.

- **Back-end**

Different from the stock window, it offers the possibility of looking up for radiopharmaceuticals that have been prepared in previous days by creating a new function which clears the *tk.Treeview()* and inserts the data related to the new day selected.

### 5.3.11 Dose dispensation module

- **Front-end**

Once a patient has arrived at the nuclear medicine floor and its corresponding radiopharmaceutical preparation has been executed, a nurse can extract a single dose and fill the respective questionnaire. As a consequence, the following label will be printed, Figure 16.

| | | | | |
|---|---|---|---|---|
| HOSPITAL CLÍNIC | | UNITAT DE RADIOFARMÀCIA | | |
| 13/05/2022 | | *RADIOACTIU* | | |
| **RENOGRAMA AMB DIURETIC** | | | | |
| | | | | |
| RADIOFÀRMAC: | | 99mTc-MAG-3 | | |
| DOSI (mCi) | 5 | H. Calibració: | 07:40 | |
| | | H. Caducitat: | 9:12:00 | |
| | | | *Tecnic:* | |
| | | | | |

***Figure 16.*** *Dose label*

Concerning PET doses, this end-of-project software is able to compute and guide nurses about the volume and activity that should be dispensed depending on the patient's weight and the hours of the extraction and injection. At the end of the form, the real information about the dose dispensation must be introduced.

- **Back-end**

In order to assign the preparation used to the patient involved, the last column of the examination SQL table links the medical exam to the type of radiopharmaceutical that should be injected.

Thanks to this connection, the register extraction is easier by only running the SELECT+INNER JOIN statement.

As was mentioned before, this software provides a dose estimation for patients who are prescribed to have a PET scan. This prediction is based on three consecutive mathematical formulas. Firstly, the final activity that should be injected into a patient is computed following Equation 9. Secondly, a nurse needs to know the amount of radioactivity required to be dispensed as we all know, that radioactive material decays over time, Equation 1. Finally, the last step is to calculate the volume according to this activity due to the previous assumption, Equation 10.

$$A = W(kg) * 0{,}1\ (mCi)$$

***Equation 9.*** *Weight-activity conversion*

$$V_{ex} = \frac{A * V_0}{A_0}$$

***Equation 10.*** *Activity-volume relation*

### 5.3.12 Traceability module

- **Front-end**

This block is planned to provide good traceability issued by patient name, kit batch, generator batch, ready-to-use radiopharmaceutical batch and preparations. Once the selection is done, the result information can be displayed on the screen below its respective *tk.Combobox()* or summarized in an Excel report depending on which button is pressed.

- **Back-end**

To perform faster traceabilities without much computational cost, four different python functions were defined since each one explores different SQL tables. Regarding the patient research, a basic SELECT statement was combined with the LEFT JOIN command because we want to keep the principal rows and add values from the secondary ones as can be analyzed in Figure 17. In addition, the WHERE condition was specified to obtain the records with a specific medical record (MRC).

```
c.execute ("""SELECT id_preparat, id_exam, dispensation.hour, dispensation.date, dispensation.activity,
           dispensation.volume FROM dispensation
            LEFT JOIN mradioactivereg ON dispensation.id_preparat=mradioactivereg.id
            LEFT JOIN preparationtec ON dispensation.id_preparat=preparationtec.id
            LEFT JOIN preprationautWC ON dispensation.id_preparat=preprationautWC.id
            LEFT JOIN preprationautRC ON dispensation.id_preparat=preprationautRC.id
            LEFT JOIN preprationautIn ON dispensation.id_preparat=preprationautIn.id
            WHERE MRC='{}'""".format(MRC))
```

***Figure 17.*** *SQL statement for extracting patient data*

Concerning kit and generator functions, they are based on a similar programming code in which a first scan of the three preparation SQL tables involved is executed. As a result, a list of ID and Nan values is acquired that is reduced to the ones that provide quality information. After, this series of items are compared to the dispensation SQL table with the purpose of getting patients data.

On the other hand, to generate a list which includes all the ID of the diverse preparations elaborated for their traceability, four combinations of the SELECT statement and the extend list function are carried out, each item of which goes through an evaluation to obtain a group of patient records related.

#### 5.3.13 Queries module

- **Front-end**

This last block is composed of an almost identical graphic design associated with the storage and stock windows. Unlike, it is aimed to show all the dispensation records in a single day which can be chosen with the Calendar widget.

- **Back-end**

In order to fill the Treeview, a SELECT statement is performed, and the outcome records are placed inside by using the function *treeview.insert().*

### 5.4 Communication with activimeters

Activimeters are necessary instruments to measure the radioactivity of radiopharmaceuticals after their preparations and before the dispensation of doses. The initial idea was to be able to establish a communication between the software and the activimeters of the *Unitat de Radiofàrmacia de l'Hospital Clínic* as some fields of several questionnaires within the application ask for this measure.

The first step consisted of analyzing the user's manuals of the three activimeters and the devices themselves since they were acquired at different moments. Regarding their serial ports, in Figure 2, three different options were observed which then, were studied in the Solutions study. As a result, the best option was the RS232-USB cable.

While the purchasing process was being carried out, the Python code was being developed based on the Serial package which offers functions to create a connection with the external device, read and send data to quote a few. Once the cable arrived at the hospital, the driver provided wasn't compatible with macOS operating system so, it was installed in a Windows laptop lent by the end-of-project's tutor, Aida.

After several attempts and rewriting the Python code, a connection was accomplished but any data was received. Therefore, the latest versions of user manual of the three activimeters were requested and analyzed. As any solution was found and radiopharmacy's workers told us that it was not very relevant, this secondary objective was left behind as further work.

### 5.5 Connection to SAP platform

Radiopharmaceutical preparations depend mainly on the number of medical imaging tests performed. All this information is located in the HIS of the *Hospital Clinic*, called SAP. Therefore, it would be ideal to create a communicative bridge between the end-of-degree software and SAP platform in order to reduce the time invested in copying patients' data from SAP and pasting it into an Excel document.

However, a relevant drawback of this platform is that it is required to be directly connected to the hospital WI-FI network, in other words, it's mandatory to use a computer that belongs to the hospital. This disadvantage affects the development of the Python code, whereas it won't matter in its implementation.

As it is an unknown and complex informatics field, we made contact with the hospital department specialized in this discipline. Based on their advice, a Python code was developed by using the Pydicom and Pynetdicom packages. After some endeavour, a connection was established. Nevertheless, all fields couldn't be examined because of time limitations so, the accurate values required to ask for a worklist weren't identified. Finally, the alternative solution was performed, and it is explained in detail in the Start-up screen and User Manual sections.

### 5.6 Application

Once the Python script was completely developed, the following step consisted of making it executable in order to be used on another computer without installing a Python environment. The package utilized to achieve this result was *pyinstaller* which is aimed to build a Python application and all its dependencies into a single package [78]. As the library tkcalendar involved some problems with the detection of babel dependency, an additional function, *babel.numbers*, was included leading to the next and final command:

```
pyinstaller --onefile --windowed --hidden-import babel.numbers TFG_DBMS.py
```

### 5.7 Technical specifications

A user manual has been designed by using an online LaTeX editor, known as Overleaf. This document includes specifications related to installation so that the software will perform correctly because additional files are required such as Excel templates and the original database in which master tables have already been uploaded. Moreover, clinicians can follow the system instructions and be aware of some relevant requirements in several questionnaire fields while the application is being manipulated.

It can be found in User Manual.

### 5.8 Discussion. Limitations and further improvements

Concerning the main goal of this end-of-degree project explained in detail in Objectives, the application should cover all the steps performed within the radiopharmacy department in a centralized way. Moreover, its graphical interface should be easy-to-use and efficient.

In order to evaluate if this objective has been accomplished, a questionnaire, found in Software opinion survey, was created and sent to the final users who are radiopharmacists and technicians of the *Unitat de Radiofarmàcia de l'Hospital Clínic*. They could fill it out after manipulating the new software, which is located in a specific folder of the hospital cloud, and reading the user manual, found in User Manual.

The first part of the survey form is aimed to examine the general characteristics and usefulness of the application. In addition, there is one question, related to the user manual generated, which is about if the information provided is clear and helpful. Related to the results obtained, this software mimics the basic functions of the radiopharmacy's workflow, but it is still not ready to replace the current method as several modifications are required.

While the software was being developed, several meetings took place in which the different application functions were tested and adapted to their demands. Three radiopharmacists from the Radiopharmacy Department of Nuclear Medicine in Hospital Clínic, Francisco Campos, Immaculada Romero and Manuel Santos Virosta, participated showing great interest in the evolution of the software. Therefore, they tested the prototype without my supervision and answered the second part of the opinion questionnaire whose queries were related to the aesthetic and useful features of each window. All of them were able to simulate the work of that day without any problem. Thanks to these tests, small essential changes were detected in order to achieve a ready-to-use version in a clinical routine. Some of these recommendations weren't initial requisites of this end-of-degree project but, they have been coming up as the software has been evolving. In addition, these final users have verified the potential of this program.

For example, a header that could give users clear and concise information about the utility of each window is missing. It could allow users to be easily aware of which screen is open in case of unknowing the previous process followed. Besides, the introduction of more automatic fields could involve a reduction of the possibility of making errors while known numbers or words are being introduced.

Referring to more software upgrades, the dispensation window is the target of some functionality and visual changes. At present, the Combobox widget is aimed to drop down a list of all patients involved in today's imaging tests. Some recommendations received are filtering patients by imaging technique implicated and deleting patients that have been dispensed with the purpose of producing satisfaction from a job done. A further implementation related to this step of the radiopharmacy's workflow is the installation of a label printing which will be connected to the software so that, a label will be released once a specific dose is dispensed.

Furthermore, technicians follow radiopharmacy rules when combinations of technetium elutions and non-radioactive kits are being carried out. This end-of-degree software could introduce the mathematical equation required in such a way that it could show a message box about the number of vials that should be essential for a determined number of patients.

Overall, the feedback of three final users related to the software evaluation was very optimistic and useful for achieving the final software that fulfils all their initial demands. In addition, they expressed that the software evolution is on the right track so that, by making some changes, its final implementation will be carried out soon.

Regarding further directions, as this software currently runs locally and natively on the operating system of a computer, it would be such a great idea to convert it to a web application. As a result, it would allow being used by several users simultaneously without decreasing safety. In addition, as the connection with the SAP platform didn't succeed, its research can be continued and achieved.

# 6. PROJECT PLANNING

The main purpose of this section is to plan time and resources to determine suitably the activities required and to effectively control the project development.

## 6.1 Task and time definition

### 6.1.1 Work Breakdown structure (WBS)

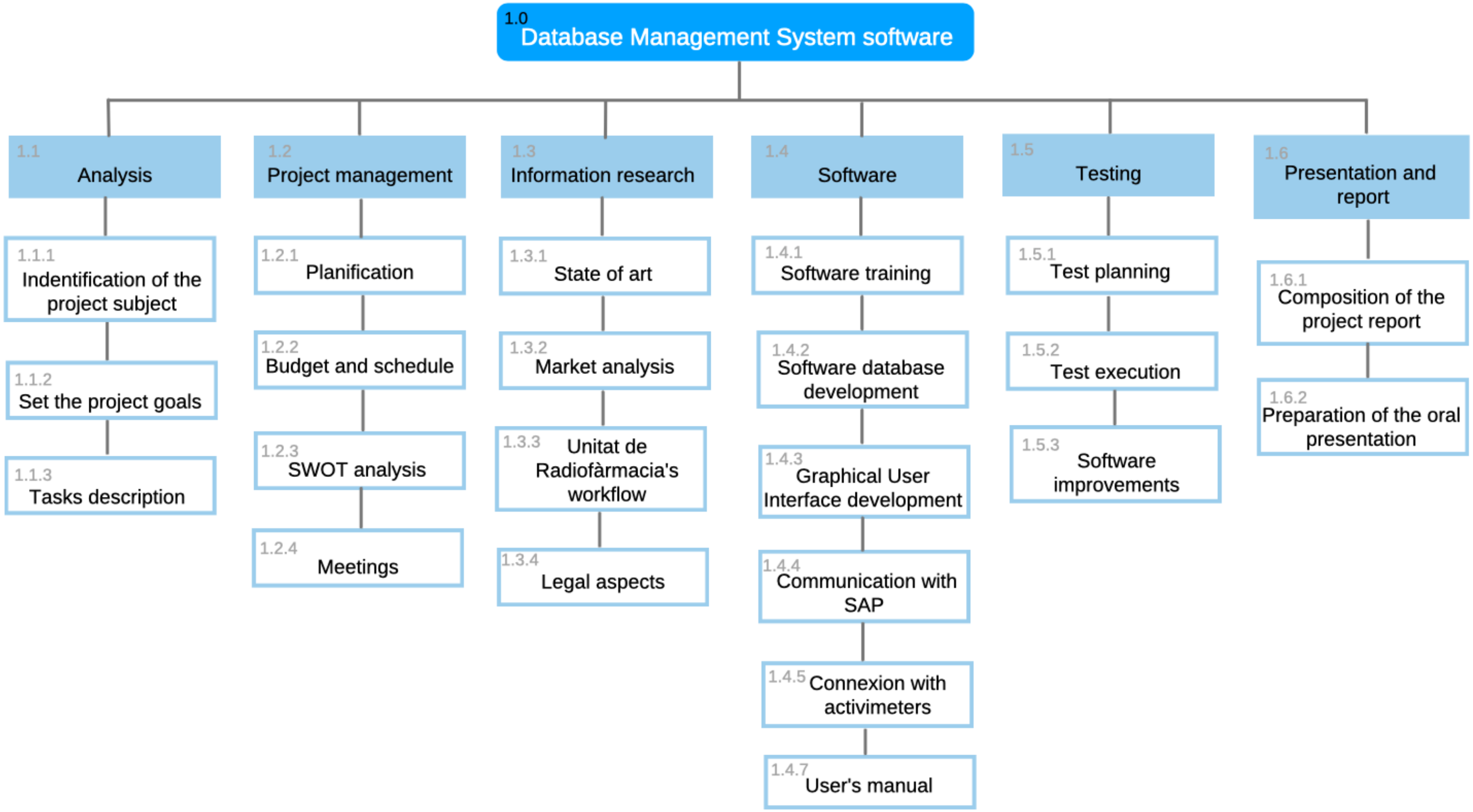



***Figure 18.*** *Work-breakdown structure diagram*

### 6.1.2 WBS dictionary

| Level | Task name | Description |
|---|---|---|
| **1. Analysis** | | |
| **1.1.** | Identification of the project subject | The project tutor offers some possible themes, and the final topic is selected. |
| **1.2.** | Set the goals | Establishing the main objectives which are expected to be achieved at the end. |
| **1.3.** | Tasks description | Brief information about each activity. |
| **2. Project management** | | |
| **2.1.** | Planification | Organization of activities. |
| **2.2.** | Budget and schedule | Determine the cost and the agenda to complete the project. |
| **2.3.** | SWOT analysis | Analysis of the current status of the DBMS to predict its performance, |
| **2.4.** | Meetings | Exchange of information with the tutor, external professional and the radiopharmacy department. |
| **3. Information research** | | |
| **3.1.** | State of art | Research of the latest improvements associated with DBMS technology |
| **3.2.** | Market analysis | Description of the current product related to the end-of-project Database Management System. |
| **3.3.** | *Unitat de Radiofàrmacia del Hospital* Clínic's workflow | Acquisition of knowledge about how the radiopharmacy department works. |
| **3.4.** | Legal aspects | Investigation of rules involved in the DBMS implementation. |
| **4. Software** | | |
| **4.1.** | Software training | Execution of a brief course in Python |
| **4.2.** | Software database development | Definition and creation of the master tables. |
| **4.3.** | Graphical User Interface development | Design of the graphical user interface of the DBMS. |
| **4.4.** | Communication with SAP | Creating a bridge between the DBMS and SAP in order to be able to extract information from the hospital information system. |

| | | |
|---|---|---|
| **4.5.** | Connexion with activimeters | Research an appropriate connector which will allow us to receive information from the activimeters. |
| **4.6.** | User's manual elaboration | Development of a document which explains how the software works. |
| **5. Testing** | | |
| **5.1.** | Test planning | Organization and elaboration of the documentation necessary for the test development. |
| **5.2.** | Test execution | The Radiopharmacy department checks and evaluates the end-of-degree DBMS. |
| **5.3.** | Software improvements | Summary of the test opinions and execution of upgrade of the software weakness. |
| **6. Presentation and report** | | |
| **6.1.** | Composition of the project report | Documenting all the end-of-degree processes. |
| **6.2.** | Preparation of the oral presentation | Elaborating a plan for the explanation of each project section. |

***Table 1**. Work-breakdown Structure dictionary*

## 6.2 Program evaluation and review techniques, and critical path method

In the following table, the Eisenhower matrix is represented. The activities' names and durations are assigned, and their precedent activity is also determined in order to be able to realize them.

| Task name | Identification | Precedent | Duration (hours) |
|---|---|---|---|
| Identification of the project subject | A | - | 1 |
| Set the goals | B | A | 2 |
| Tasks description | C | B | 3 |
| Planification | D | C | 4 |
| Budget and schedule | E | D | 4 |
| SWOT analysis | F | E | 3 |
| Meetings | G | A | 30 |
| State of art | H | C | 3 |
| Market analysis | I | H | 3 |
| Unitat de Radiofàrmacia del Hospital Clínic's workflow | J | C | 4 |
| Legal aspects | K | I | 3 |
| Software training | L | C | 20 |

| | | | |
|---|---|---|---|
| Software database development | M | J, L | 7 |
| GUI development | N | M | 100 |
| Communication with SAP | O | L | 15 |
| Connexion with activimeters | P | L | 15 |
| User’s manual elaboration | Q | N, O, P | 10 |
| Test planning | R | Q | 15 |
| Test execution | S | R | 6 |
| Software improvements | T | S | 10 |
| Composition of the project report | U | A | 60 |
| Preparation of the oral presentation | V | U, K, I, U | 20 |

***Table 2**. Project activities with their duration and precedent activity*

Based on Table 2, a PERT diagram has been developed, Figure 19, in order to obtain which is the Critical Path for avoiding delays due to unexpected events. The critical path of this project is A, B, C, L, O, P, N, Q, R, S, T and V.

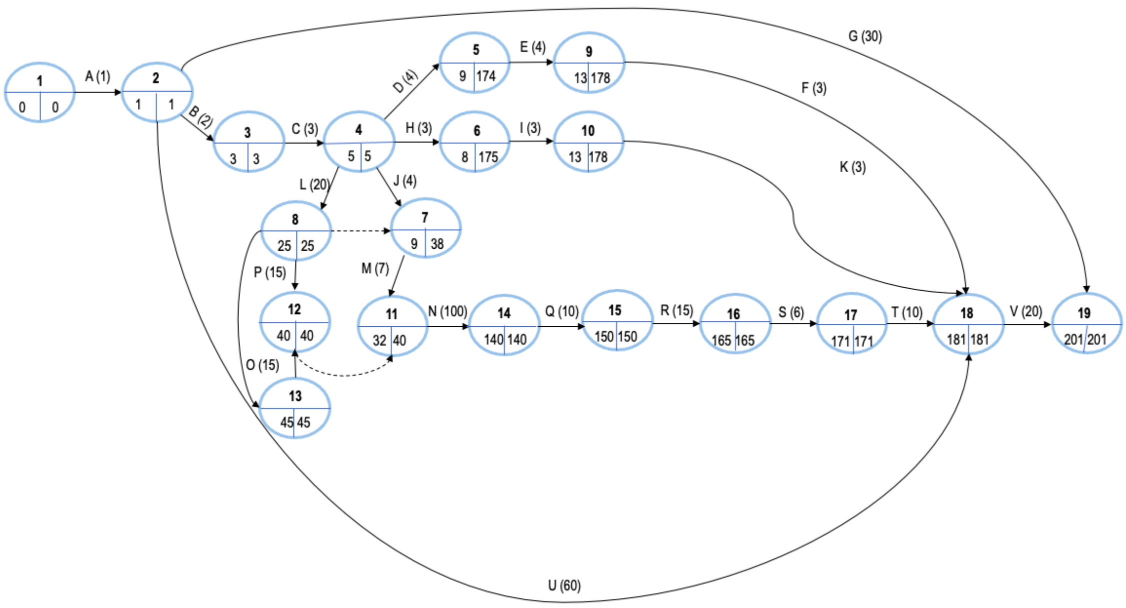



***Figure 19**. Program Evaluation and Review Technique's diagram*

### 6.3 Schedule (timing)

In order to organize the elaboration of this project, a GANTT chart was created which gives us some ideas of how this project will be developed and it is represented in the next chronological axis, Figure 20.

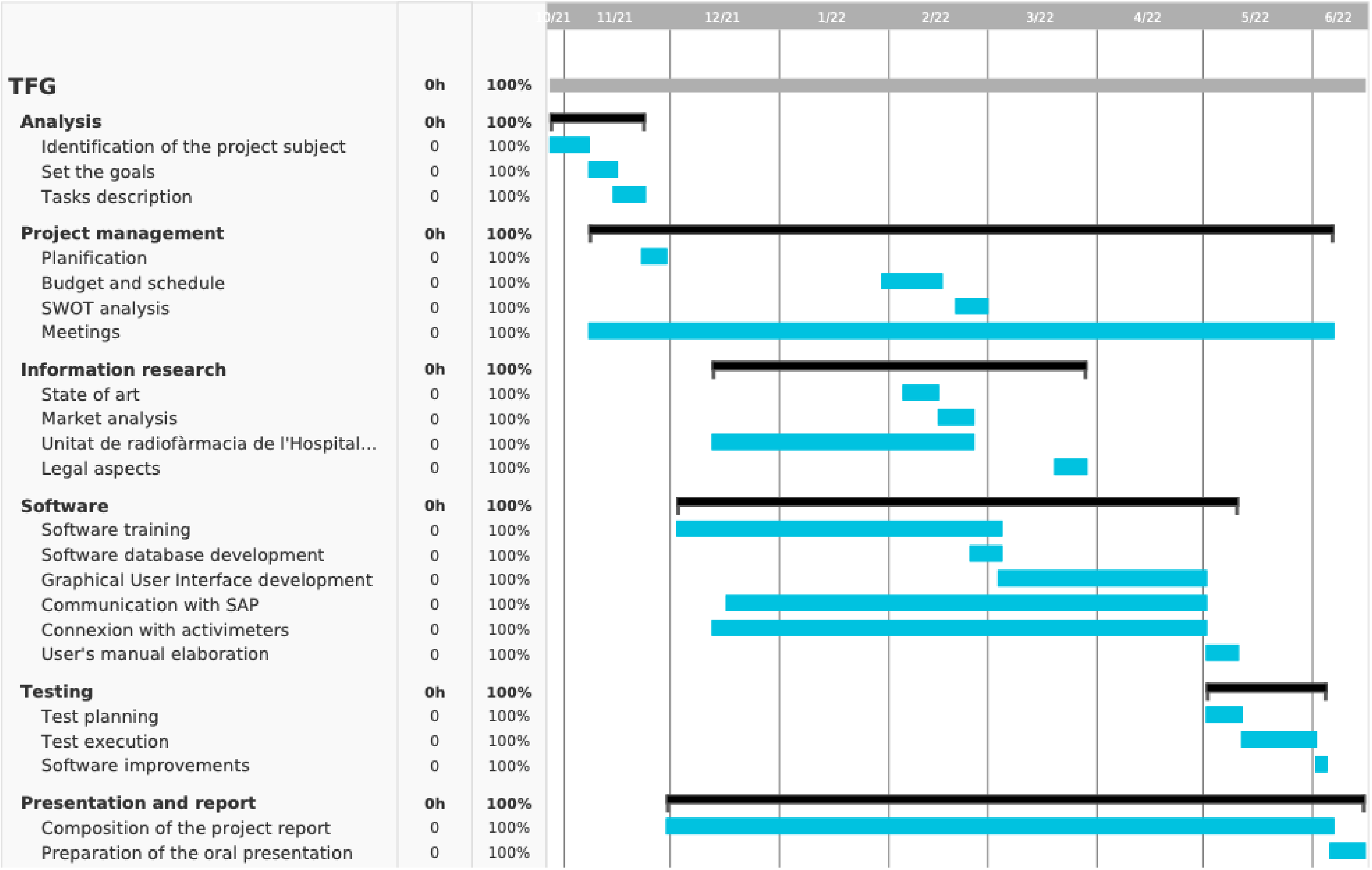



***Figure 20.*** *GANTT chart*

## 7. TECHNICAL FEASIBILITY

In order to carry on with this project of radiopharmaceutical management software, a SWOT analysis is performed. It is an evaluation of the strategic position of this work which takes account of internal and external, and positive and negative factors, Figure 21.

Regarding the project strengths, the development of this software means that all the tasks of the radiopharmacy's workflow will be centralized in a unique platform that will be accessible to any radiopharmacy worker. Therefore, radiopharmaceuticals could be tracked in an easy and fast manner. In addition, its graphical interface is user-friendly as users can accomplish tasks without previous programming knowledge. While the software was being built, it has been constantly reviewed in order to obtain the result that fits better with the demands of the *Unitat de Radiofàrmacia de l'Hospital Clínic*. As part of the main project objective, the implementation of this software would entail the removal of duplication and unnecessary procedures leading to a decrease in human error and time. However, it's a local application which means that only can be used by one person at a time. Due to the short duration of the project execution, some secondary goals haven't been completely achieved as they involve third-party help. Besides, some informatics fields have come up with which I wasn't familiar so, this has made it more difficult and drawn out the software development.

On the other hand, as technology is evolving over years, further improvements to this radiopharmaceuticals management software could be implemented in the near future. As a consequence, additional radiopharmacy departments from other hospitals may be interested in it. Nevertheless, similar solutions have been launched to the market by big enterprises such as Microsoft, Oracle and First, among others. Finally, working with a medical database involves storage and security problems due to the fact that several patients are daily attended to, and this application could be a victim of corruption or data theft.

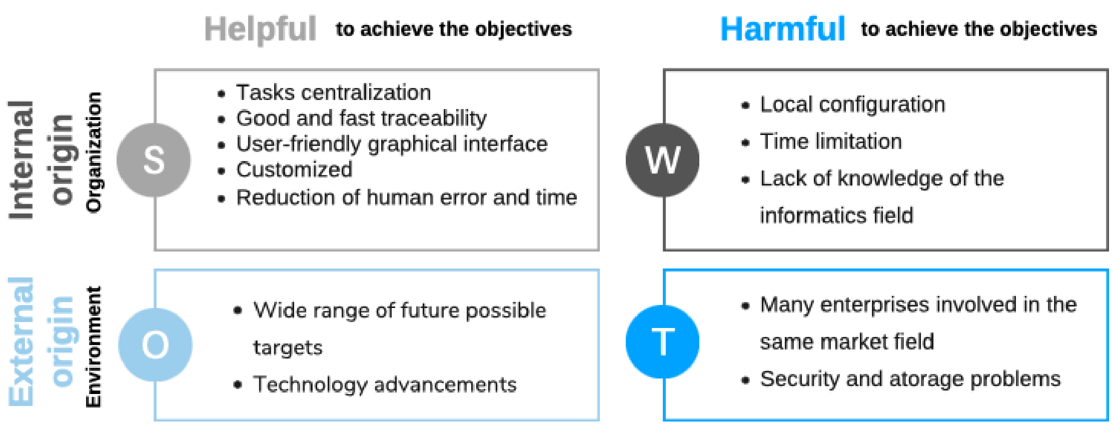



***Figure 21***. *SWOT analysis matrix*

## 8. ECONOMIC FEASIBILITY

In this chapter, all financial movements associated with executing the end-of-degree project from the moment that idea came up to the tune-up of the final product have been considered. They are classified into two main parts:

1. Material and service

To establish a communicative bridge between the project software and the activimeters of the *Unitat de Radiofàrmacia de l'Hospital Clínic*, an RS232-USB cable has been acquired. In addition, some informatics services have been hired and bought.

| | Company | Unit price (€) | Total (€) |
|---|---|---|---|
| Material | | | |
| **RS232 -USB cable** | Amazon | 14.99 | 14.99 |
| Service | | | |
| **Office 365 license** | Microsoft | 7.00€/month | 49 |
| **MacBook Air (13-inch, 2017)** | Apple | 858 | 858 |
| | | | **921.99 €** |

***Table 3.*** *Material and service costs*

2. Personnel

Besides material and service costs, the time spent by the physical team in order to develop this project should be also considered. The principal members are a junior engineer (who just graduated and has less than two years of work experience) and a biomedical engineer with phD.

Based on the average annual salary of a junior software engineer with python and SQL skills in Spain which is around 24,500€, it leads us to approximate 13.6€/hour according to "Estatuto de Trabajadores" which estimates that the work hours fluctuate around 1800h/year. Since the end-of-project is expected to last about 300 hours, the junior salary is guessed to be around 4080€.

Regarding the labour of the biomedical engineer with a PhD, its average annual salary is considered to be approximately around 76,500 €. Therefore, it leads to 42.5€/h and as it is expected to work about 75 hours, her final personal cost is estimated to be 3187.5€.

| | Salary | Hours | Total (€) |
|---|---|---|---|
| **Junior software engineer** | 13.6 €/h | 300 | 4080 |
| Biomedical engineer | 42.5€/h | 75 | 3187.5 |
| | | | **7267.5 €** |

***Table 4***. *Personal cost*

**Final budget**

By taking into account both costs, the final budget needed to develop this end-of-degree project is guessed to be 8189.49€.

## 9. REGULATION AND LEGAL ASPECTS

In order to know which legalization is involved with our product, we should highlight that this end-of-degree software is aimed to manage radiopharmacy workflow excluding recommendations for diagnosis, prognosis, monitoring and treatment of patients. In addition, this chapter is only focused on European regulations since the software's scope is restricted to Spain.

The Medical Device Coordination Group (MDCG) from the European Commission developed the software classification in the Medical Device Regulation and In Vitro Diagnostic Regulation known as MDCG 2019-11. This document provides guidance on the qualification and classification of software under Regulation (UE).

According to the MDCG 2019-11 [76], software used for medical data management in the healthcare environment is defined as: "Information systems that are intended only to store, archive and transfer data are not qualified as medical devices in themselves. "

Given that this end-of-project software doesn't fall under the definition of medical device, other legislation should be taken into account.

As clinical data is also manipulated in this radiopharmaceutical management application, the General Data Protection Regulation of the European Commission must be contemplated. This rule strengthens the fundamental rights and freedom of people and protects their personal data independently from where its manipulation is being performed [77].

In addition, since the software is involved with radiopharmaceutical management, the law 1/2015 and the RD 1345/2007 need to come into effect. On the one hand, the first one summarized the standards of guarantees and rational use of medicine and healthcare products. On the other hand, the last standardizes the procedure of authorization, registration and dispensation conditions of drugs of human use which are industrially fabricated. [79] [80]

## 10. CONCLUSIONS AND FURTHER DIRECTIONS

The principal aim of this project was to develop a user-friendly and efficient software which groups together all processes involved in the *Unitat de Radiofàrmacia del Hospital* Clínic's workflow in a centralized electronic system. In addition, supplementary objectives were considered with the purpose of improving the main goal such as establishing a connection with dose calibrators and creating communication with the HIS.

For this project, this radiopharmacy's workflow was extensively studied which helped me to choose the programming solution that fitted better with it among all the options analysed. Besides drawbacks and benefits, self-experience and professional recommendations were relevant reasons for this choice. This software was developed from scratch in the Anaconda environment. The most Python packages used were *Tkinter* for the graphical interface, *SQLite* for the database creation and communications and *openpyxl* for the necessary reports.

Regarding the graphical interface development, it was initially inspired by the current documents utilized in this Nuclear Medicine Department. Then, it was modelled based on the radiopharmacists' opinions and demands, and also on the design which provided the best usefulness and performance..

Once the prototype was built and its user manual elaborated, three radiopharmacy workers tested it before filling out the opinion questionnaire. Although being the early model, their feedback was generally positive and useful for future improvements.

Concerning the future potential it has, the implementation of this software in the *Unitat de Radiofàrmacia del Hospital* Clínic will mean good radiopharmaceutical management as duplication and unnecessary procedures will be removed and human errors will be less susceptible to happen. Furthermore, the time invested in those actions can be dedicated to more relevant processes to ensure patient safety and radiopharmaceutical efficiency. Moreover, the implementation of this software in this Unit is expected to be performed in the near future.

Additionally, as paper-based mediums are still present in healthcare, this end-of-degree project can encourage other hospital departments to introduce centralized electronic management which provides effectiveness, timeliness, communication, among others.

## 11. REFERENCES

[1]: How Are Different Types Of Data Used In The Healthcare Industry? | Smart Data Solutions. (n.d.). Retrieved April 11, 2022, from https://sdata.us/2021/01/26/how-are-different-types-of-data-used-in-the-healthcare-industry/

[2]: What is Hospital Information Systems (HIS)? (n.d.). Retrieved April 11, 2022, from https://www.talkinghealthtech.com/glossary/hospital-information-systems-his

[3]: The evolution of Hospital Information Systems - OH Blog. (n.d.). Retrieved April 11, 2022, from https://blog.orionhealth.com/the-evolution-of-hospital-information-systems/

[4]: What is a Database? Definition from SearchDataManagement. (n.d.). Retrieved April 13, 2022, from https://www.techtarget.com/searchdatamanagement/definition/database

[5]: How Important Are Databases in the Healthcare System? - DNSstuff. (n.d.). Retrieved April 11, 2022, from https://www.dnsstuff.com/how-important-database-healthcare

[6]: Kreel L. (1991). Medical imaging. Postgraduate medical journal, 67(786), 334–346. https://doi.org/10.1136/pgmj.67.786.334

[7]: The Pros and Cons of Healthcare Database Systems | The Healthcare Guys. (n.d.). Retrieved April 11, 2022, from https://www.healthcareguys.com/2019/10/27/the-pros-and-cons-of-healthcare-database-systems/

[8]: Nuclear Medicine. (n.d.). Retrieved April 11, 2022, from https://www.nibib.nih.gov/science-education/science-topics/nuclear-medicine

[9]: What does a Radiopharmacist do? - British Nuclear Medicine Society. (n.d.). Retrieved April 11, 2022, from https://www.bnms.org.uk/page/WhatdoesaRadiopharmacistdo

[10]: Human Health Campus - Documentation. (n.d.). Retrieved April 11, 2022, from https://humanhealth.iaea.org/HHW/Radiopharmacy/VirRad/Documentation/index.html

[11]: A Timeline of Database History | Quickbase. (n.d.). Retrieved April 14, 2022, from https://www.quickbase.com/articles/timeline-of-database-history

[12]: Web 101: A History of the GUI | WIRED. (n.d.). Retrieved April 14, 2022, from https://www.wired.com/1997/12/web-101-a-history-of-the-gui/

[13]: Yang, J., Li, Y., Liu, Q., Li, L., Feng, A., Wang, T., Zheng, S., Xu, A., & Lyu, J. (2020). Brief introduction of medical database and data mining technology in big data era. Journal of evidence-based medicine, 13(1), 57–69. https://doi.org/10.1111/jebm.12373

[14]: ¿Qué es el minado de Datos o Data Mininig? Técnicas y pasos a seguir. (n.d.). Retrieved April 12, 2022, from https://www.iebschool.com/blog/data-mining-mineria-datos-big-data/

[15]: Importance of Database Security in 2021 – GA Systems. (n.d.). Retrieved April 12, 2022, from https://www.gasystems.com.au/blog/database-security/

[16]: Types of Database Encryption Methods | N-able. (n.d.). Retrieved April 12, 2022, from https://www.n-able.com/blog/types-database-encryption-methods

[17]: Abu-Elezz, I., Hassan, A., Nazeemudeen, A., Househ, M., & Abd-Alrazaq, A. (2020). The benefits and threats of blockchain technology in healthcare: A scoping review. International journal of medical informatics, 142, 104246. https://doi.org/10.1016/j.ijmedinf.2020.104246

[18]: Cloud Native Computing Foundation. (n.d.). Retrieved April 12, 2022, from https://www.cncf.io/

[19]: What is Cloud Native? | Oracle. (n.d.). Retrieved April 12, 2022, from https://www.oracle.com/cloud/cloud-native/what-is-cloud-native/

[20]: What is Cloud Native? | Oracle. (n.d.). Retrieved April 12, 2022, from https://www.oracle.com/cloud/cloud-native/what-is-cloud-native/

[21]: Nihalani, N., Silakari, S., & Motwani, M. (2009). Integration of artificial intelligence and database management system: An inventive approach for intelligent databases. 2009 1st International Conference on Computational Intelligence, Communication Systems and Networks, CICSYN 2009, 35–40. https://doi.org/10.1109/CICSYN.2009.54

[22]: The Evolution of AI in Healthcare. (n.d.). Retrieved April 13, 2022, from https://www.xsolis.com/blog/the-evolution-of-ai-in-healthcare

[23]: AI and Data Management Delivering Data-Driven Business Transformation and Operational Efficiencies The Importance of AI and Machine Learning to Data Platform and Analytics Initiatives. (n.d.).

[24]: Techreviewer Blog: What Are the Challenges for Database Management in 2021? (n.d.). Retrieved April 13, 2022, from https://techreviewer.co/blog/what-are-the-challenges-for-database-management-in-2021

[25]: What Does GUI Mean? Definitions, Elements and Benefits of Graphical User Interface | Indeed.com. (n.d.). Retrieved April 14, 2022, from https://www.indeed.com/career-advice/career-development/gui-meaning

[26]: Andrade, E., Quinlan, L., Harte, R., Byrne, D., Fallon, E., Kelly, M., Casey, S., Kirrane, F., O'Connor, P., O'Hora, D., Scully, M., Laffey, J., Pladys, P., Beuchée, A., & ÓLaighin, G. (2020). Novel Interface Designs for Patient Monitoring Applications in Critical Care Medicine: Human Factors Review. JMIR human factors, 7(3), e15052. https://doi.org/10.2196/15052

[27]: Wong, K. H., Bayarsaikhan, S., Levine, B. A., & Mun, S. K. (2020). Prototype of a Military Medic Smartphone Medical Graphical User Interface for Use by Medics in Deployed Environments. Military medicine, 185(Suppl 1), 536–543. https://doi.org/10.1093/milmed/usz225

[28]: Gopika S Kumar. (2020). GUI based Multi-Modal Biometrics Authentication. International Journal of Engineering Research And, V9(06). https://doi.org/10.17577/IJERTV9IS060849

[29]: Database Management System Market Size | Industry Report, 2025. (n.d.). Retrieved April 17, 2022, from https://www.grandviewresearch.com/industry-analysis/database-management-systems-dbms-market

[30]: Medicina Nuclear | Hospital de Bellvitge. (n.d.). Retrieved April 17, 2022, from https://bellvitgehospital.cat/es/especialidades/medicina-nuclear

[31]: Residència en Radiofarmàcia | Vall d'Hebron Barcelona Hospital Campus. (n.d.). Retrieved April 17, 2022, from https://www.vallhebron.com/docencia/unitats-docents/radiofarmacia

[32]: Medicina Nuclear - Germans Trias. (n.d.). Retrieved April 17, 2022, from http://germanstriashospital.cat/es/servicio-medicina-nuclear

[33]: Bachman, C. W. (2009). The Origin of the Integrated Data Store (IDS): The First Direct-Access DBMS. IEEE Annals of the History of Computing, 31(04), 42–54. https://doi.org/10.1109/MAHC.2009.110

[34]: Information Management System (IMS) Software | IBM. (n.d.). Retrieved April 20, 2022, from https://www.ibm.com/it-infrastructure/z/ims

[35]: The evolution of Database Management system – Track2Training. (n.d.). Retrieved April 20, 2022, from https://track2training.com/2021/07/26/the-evolution-of-database-management-system/

[36]: Administrador de bases de datos Raima (RDM) | Base de datos integrada de alta velocidad. (n.d.). Retrieved April 20, 2022, from https://raima.com/es/

[37]: What is a Network Database? - Definition from Techopedia. (n.d.). Retrieved April 20, 2022, from https://www.techopedia.com/definition/20971/network-database

[38]: Broadcom. (2020). Product Brief CA IDMS.

[39]: Multics Relational Data Store (MRDS). (n.d.). Retrieved April 20, 2022, from https://www.mcjones.org/System_R/mrds.html

[40]: Database of Databases - Ingres. (n.d.). Retrieved April 20, 2022, from https://dbdb.io/db/ingres

[41]: Object-Oriented Database {Concepts, Examples, Pros and Cons}. (n.d.). Retrieved April 20, 2022, from https://phoenixnap.com/kb/object-oriented-database

[42]: Software Library :: IgniteTech. (n.d.). Retrieved April 20, 2022, from https://ignitetech.com/softwarelibrary

[43]: What is a Graphical User Interface? Definition and FAQs | HEAVY.AI. (n.d.). Retrieved April 20, 2022, from https://www.heavy.ai/technical-glossary/graphical-user-interface

[44]: Ristevski, B., & Chen, M. (2018). Big Data Analytics in Medicine and Healthcare. Journal of integrative bioinformatics, 15(3), 20170030. https://doi.org/10.1515/jib-2017-0030

[45]: Roberts, G., Collins, D., Roberts, G., Collins, D., & Roberts, G. (2021). by with The Use of Machine Learning in Database Systems.

[46]: Gómez Perales J. L. (2013). Development of a new software for comprehensive management and traceability of hospital radiopharmacies. Computer methods and programs in biomedicine, 112(1), 166–172. https://doi-org.sire.ub.edu/10.1016/j.cmpb.2013.06.016

[47]: Radiolab - software de gestión de radiofarmacias hospitalarias. (n.d.). Retrieved April 21, 2022, from https://www.radiopharmacy.net/radiolab-es.html

[48]: Radiopharmacy. (n.d.). Retrieved April 21, 2022, from http://www.first-global.com/en-us/Solutions/SiiMA/Radiopharmacy

[49]: QSOFT - Software para Hospitales, Clinicas y Centros Medicos. (n.d.). Retrieved April 21, 2022, from https://q-soft.net/

[50]: Microsoft Access - Descargar. (n.d.). Retrieved April 21, 2022, from https://microsoft-access.softonic.com/

[51]: Top 25 Best Database Management Software in 2022. (n.d.). Retrieved April 29, 2022, from https://improvado.io/blog/top-25-best-database-management-software

[52]: LibreOffice - Descargar. (n.d.). Retrieved April 29, 2022, from https://libreoffice.softonic.com/

[53]: ¿Qué es Power Apps? - Power Apps | Microsoft Docs. (n.d.). Retrieved April 29, 2022, from https://docs.microsoft.com/es-es/power-apps/powerapps-overview

[54]: 5 beneficios de la integración de Power Apps para su empresa. (n.d.). Retrieved April 29, 2022, from https://fwpr.com/es/5-beneficios-de-la-integracion-de-power-apps-para-su-empresa/

[55]: Anaconda | The World's Most Popular Data Science Platform. (n.d.). Retrieved April 29, 2022, from https://www.anaconda.com/
[56]: Home — Spyder IDE. (n.d.). Retrieved April 29, 2022, from https://www.spyder-ide.org/
[57]: MATLAB - El lenguaje del cálculo técnico - MATLAB & Simulink. (n.d.). Retrieved April 29, 2022, from https://es.mathworks.com/products/matlab.html
[58]: 1.2 Elements of Programming. (n.d.). Retrieved May 3, 2022, from https://composingprograms.com/pages/12-elements-of-programming.html#expressions
[59]: 1. Whetting Your Appetite — Python 3.10.4 documentation. (n.d.). Retrieved May 3, 2022, from https://docs.python.org/3/tutorial/appetite.html
[60]: R: What is R? (n.d.). Retrieved May 3, 2022, from https://www.r-project.org/about.html
[61]: tkinter — Interface de Python para Tcl/Tk — documentación de Python - 3.10.4. (n.d.). Retrieved May 3, 2022, from https://docs.python.org/es/3/library/tkinter.html
[62]: PySimpleGUI: The Simple Way to Create a GUI With Python – Real Python. (n.d.). Retrieved May 3, 2022, from https://realpython.com/pysimplegui-python/
[63]: (2) New Messages! (n.d.). Retrieved May 3, 2022, from https://mode.com/python-tutorial/libraries/pandas/
[64]: Databases - Full Stack Python. (n.d.). Retrieved May 3, 2022, from https://www.fullstackpython.com/databases.html
[65]: PostgreSQL: About. (n.d.). Retrieved May 3, 2022, from https://www.postgresql.org/about/
[66]: PostgreSQL: Documentation: 14: 2. A Brief History of PostgreSQL. (n.d.). Retrieved May 3, 2022, from https://www.postgresql.org/docs/current/history.html
[67]: Understanding DCE & DTE connections | Digi International. (n.d.). Retrieved April 30, 2022, from https://www.digi.com/support/knowledge-base/understanding-dce-dte-connections
[68]: (4) What is RS232 and What is it Used for? - YouTube. (n.d.). Retrieved April 30, 2022, from https://www.youtube.com/watch?v=eo9dbnrpspM
[69]: What are the types of USB cables. (n.d.). Retrieved April 30, 2022, from https://www.blackbox.cl/cl-cl/page/44082/Recursos/Recursos-Tecnicos/black-box-explica/usb-universal-serial-bus/Cuales-son-los-tipos-de-cables-USB
[70]: Identifying your USB connector | CMD. (n.d.). Retrieved April 30, 2022, from https://www.cmd-ltd.com/advice-centre/usb-chargers-and-power-modules/usb-and-power-module-product-help/identifying-usb-connector/#identifyingusb3
[71]: What is a Health Information System? | Digital Guardian. (n.d.). Retrieved May 2, 2022, from https://digitalguardian.com/blog/what-health-information-system
[72]: K Basic Worklist Management Service (Normative). (n.d.). Retrieved May 2, 2022, from https://dicom.nema.org/medical/dicom/current/output/chtml/part04/chapter_K.html
[73]: CSV file: Definition - Google Ads Help. (n.d.). Retrieved May 2, 2022, from https://support.google.com/google-ads/answer/9004364?hl=en

[74]: sqlite3 — DB-API 2.0 interface for SQLite databases — Python 3.10.4 documentation. (n.d.). Retrieved May 23, 2022, from https://docs.python.org/3/library/sqlite3.html

[75]: tkinter — Interface de Python para Tcl/Tk — documentación de Python - 3.10.4. (n.d.). Retrieved May 24, 2022, from https://docs.python.org/es/3/library/tkinter.html

[76]: *MEDICAL DEVICES: Guidance document-Qualification and Classification of stand alone software - GUIDELINES ON THE QUALIFICATION AND CLASSIFICATION OF STAND ALONE SOFTWARE USED IN HEALTHCARE WITHIN THE REGULATORY FRAMEWORK OF MEDICAL DEVICES*. (n.d.).

[77]: *La protección de datos en la UE | Comisión Europea*. (n.d.). Retrieved March 26, 2022, from https://ec.europa.eu/info/law/law-topic/data-protection/data-protection-eu_es

[78]: Create Executable from Python Script using Pyinstaller - Data to Fish. (n.d.). Retrieved June 1, 2022, from https://datatofish.com/executable-pyinstaller/

[79]: Ministerio de Educación y Formación Profesional, G. de E. (2019). Boletón Oficial del Estado. *Boletín Oficial Del Estado*, *11*, 2260–2268. https://www.boe.es/boe/dias/2019/01/12/pdfs/BOE-A-2019-317.pdf

[80]: España. (2007). Real Decreto-ley 1345/2007, de 11 de octubre de 2007, por el que se regula el procedimiento de autorización, registro y condiciones de dispensación de los medicamentos de uso humano fabricados industrialmente. *Boletin Oficial Del Estado*, 45652–45698.

# 12. ANNEXES

## 12.1 Software opinion survey

To acquire the opinion of the final users of this end-of-degree application, a survey form has been developed. An important objective established in this project is the ability to build an application whose graphic interface should be attractive, user-friendly, efficient and simple since it will be manipulated by people not specialized in informatics.

The following questionnaire is aimed to be filled by radiopharmacy workers while the radiopharmaceutical management software is being tested. It has been divided into two main parts:

- An evaluation of general characteristics such as efficiency, fulfilment of the customer's demands and usefulness.
- Individual exams about design features for each application windows.

This survey form has been provided to each final user through the Google Forms platform consequently, the results are automatically received. The corresponding link sent was:

https://docs.google.com/forms/d/e/1FAIpQLScsowlVOrdhXYoK-ZxnzTc33Tfql68rUhNjjd50MVLsMfoorw/viewform?usp=sf_link

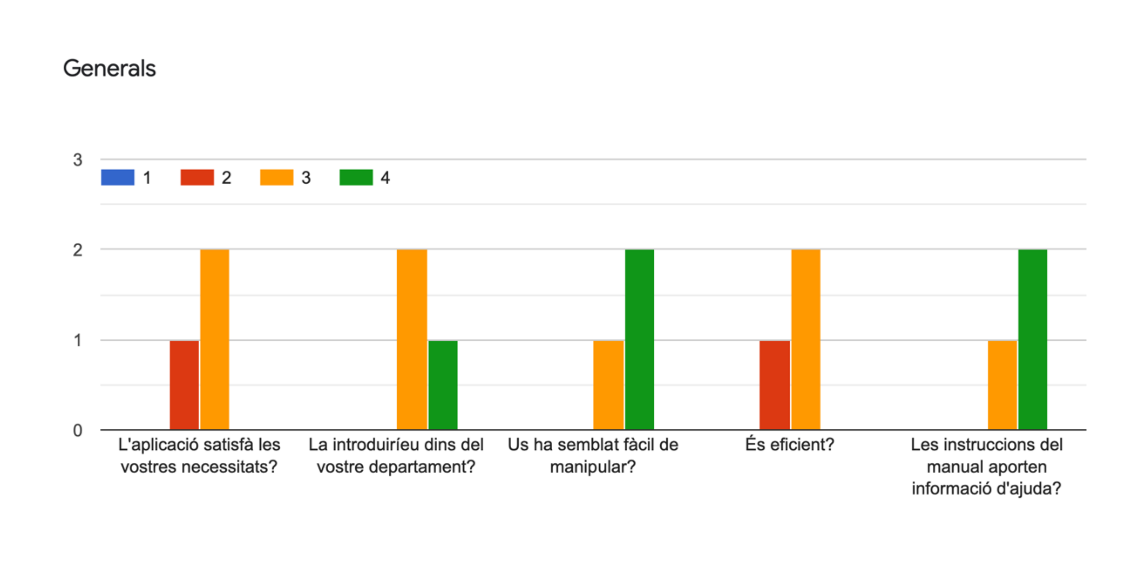
| ***Preguntes*** | **Puntuació** | | | |
|---|---|---|---|---|
| *Generals* | 1 | 2 | 3 | 4 |
| *L'aplicació satisfà les vostres necessitats?* | | | | |
| *La introduiríeu dins del vostre departament ?* | | | | |
| *Us ha semblat fàcil de manipular?* | | | | |
| *És eficient?* | | | | |
| *Les instruccions del manual aporten informació d'ajuda?* | | | | |
| | | | | |
| *Finestra* | 1 | 2 | 3 | 4 |
| *Distribució dels elements* | | | | |
| *Mida de la lletra* | | | | |
| *Claredat de les frases* | | | | |
| *Vocabulari adequat* | | | | |
| *Automatització* | | | | |

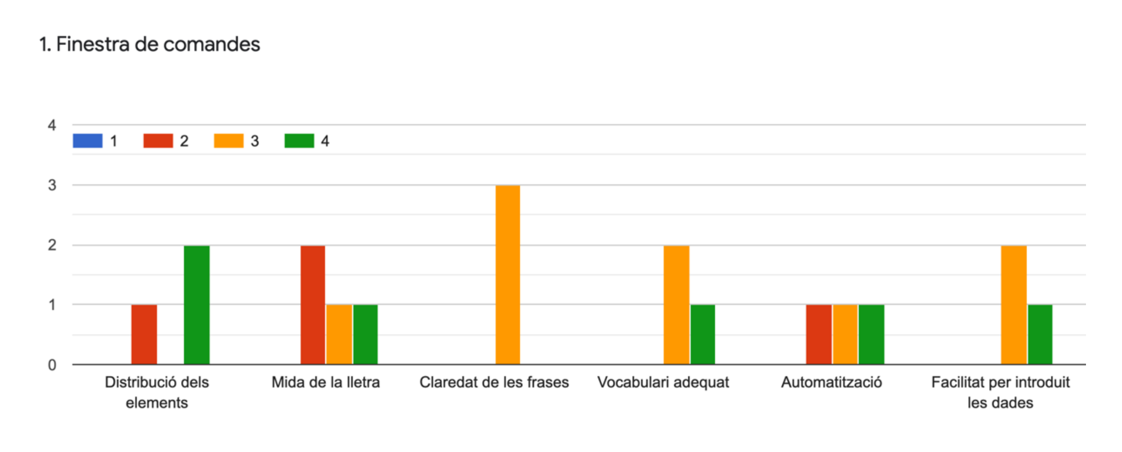
| | | | | |
|---|---|---|---|---|
| *Facilitat per introduir les dades* | | | | |
| *Validació de les dades* | | | | |
| *Comentari* | | | | |
| | | | | |

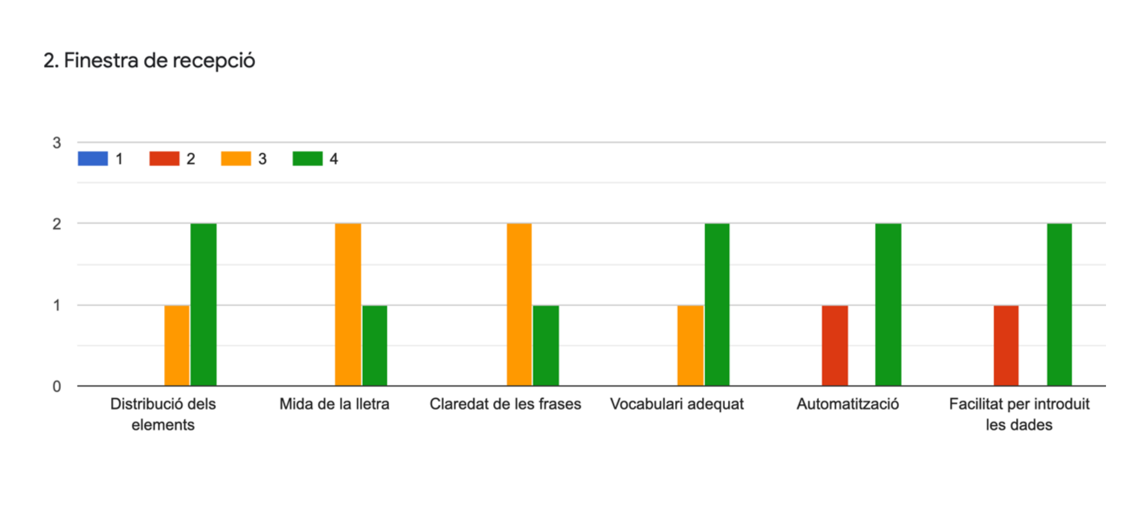
***Table A 1.*** *Opinion survey about the software*

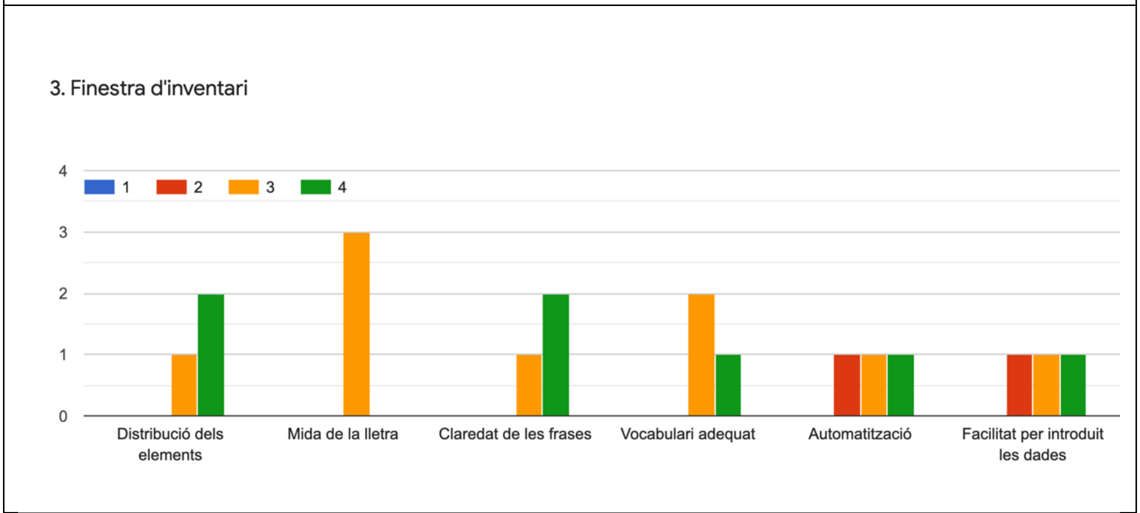
## 12.2 Result of the questionnaire

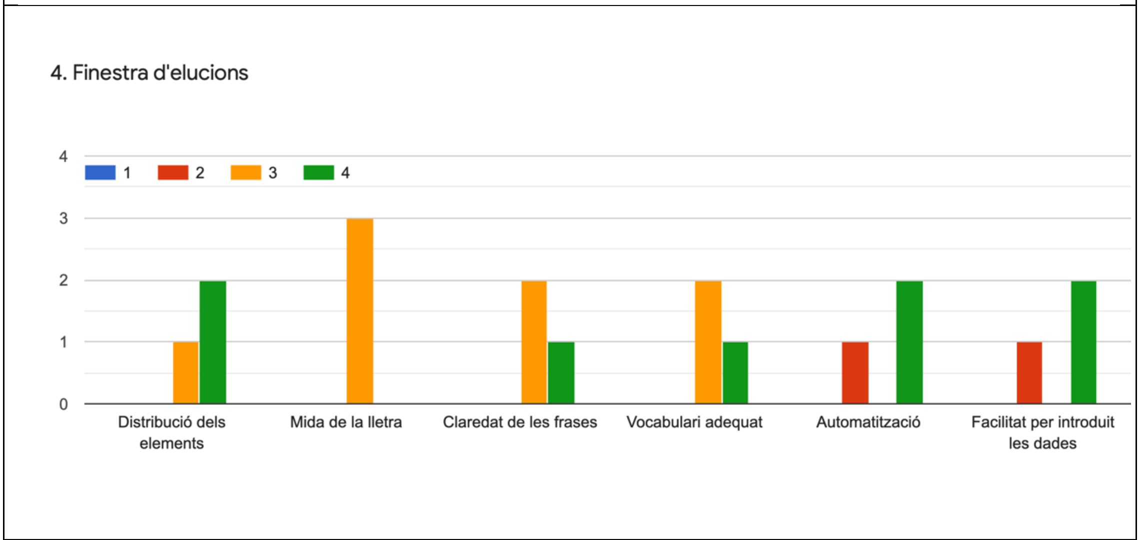
The next table summarizes the result obtained from the opinion questionnaire which has been answered by three radiophamarcists.



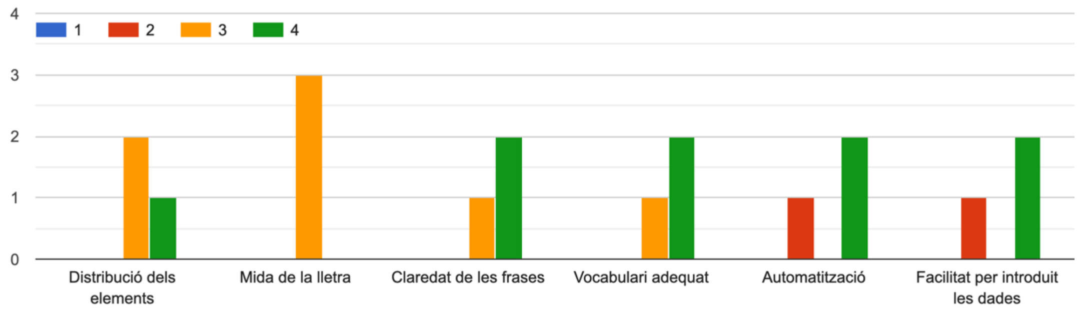




2. Finestra de recepció
1
2
3
4
0
1
2
3
Distribució dels elements
Mida de la lletra
Claredat de les frases
Vocabulari adequat
Automatització
Facilitat per introduit les dades

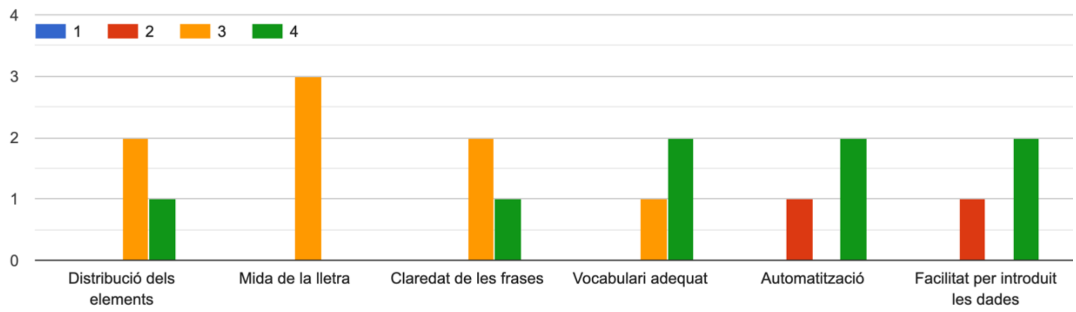

3. Finestra d'inventari
1
2
3
4
0
1
2
3
4
Distribució dels elements
Mida de la lletra
Claredat de les frases
Vocabulari adequat
Automatització
Facilitat per introduit les dades


4. Finestra d'elucions
1
2
3
4
0
1
2
3
4
Distribució dels elements
Mida de la lletra
Claredat de les frases
Vocabulari adequat
Automatització
Facilitat per introduit les dades

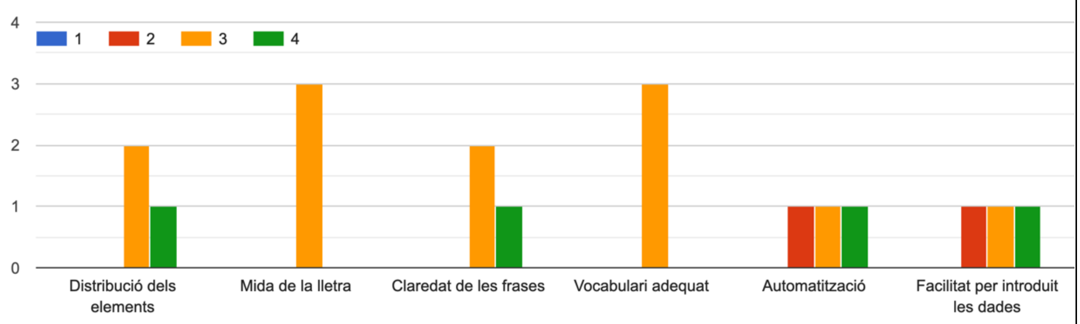
5. Finestra de preparacions



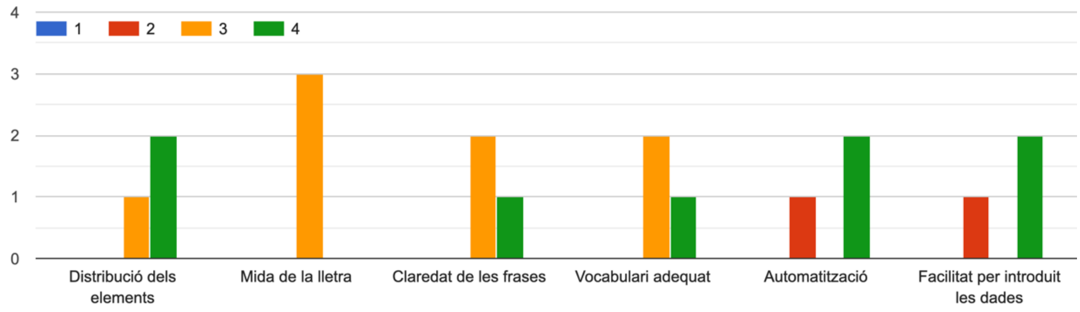

6. Finestra de magatzem



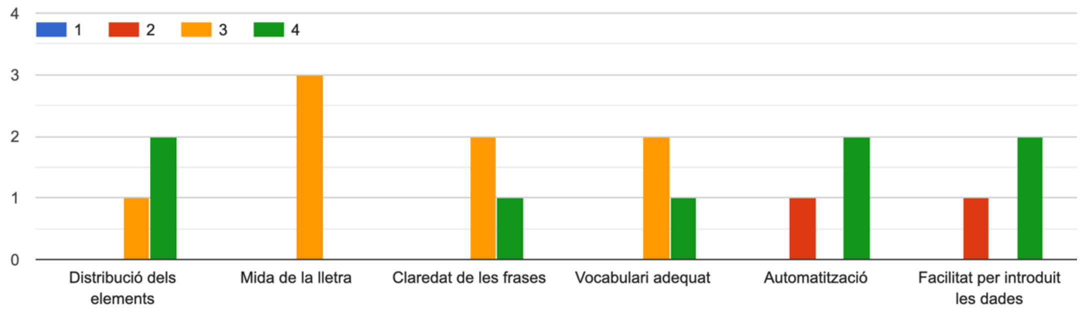

7. Finestra de dispensació



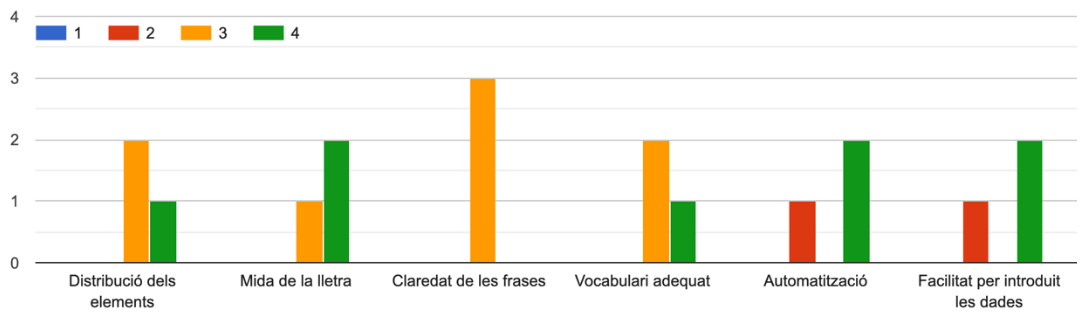
8. Finestra de gestió d'activitat

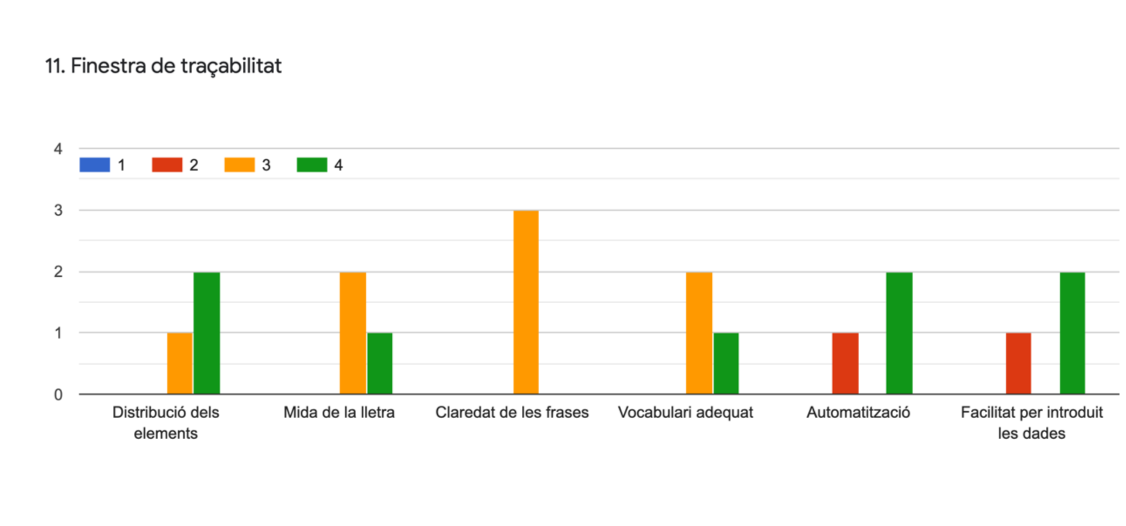



9. Finestra de controls de qualitat




10. Finestra de consultes





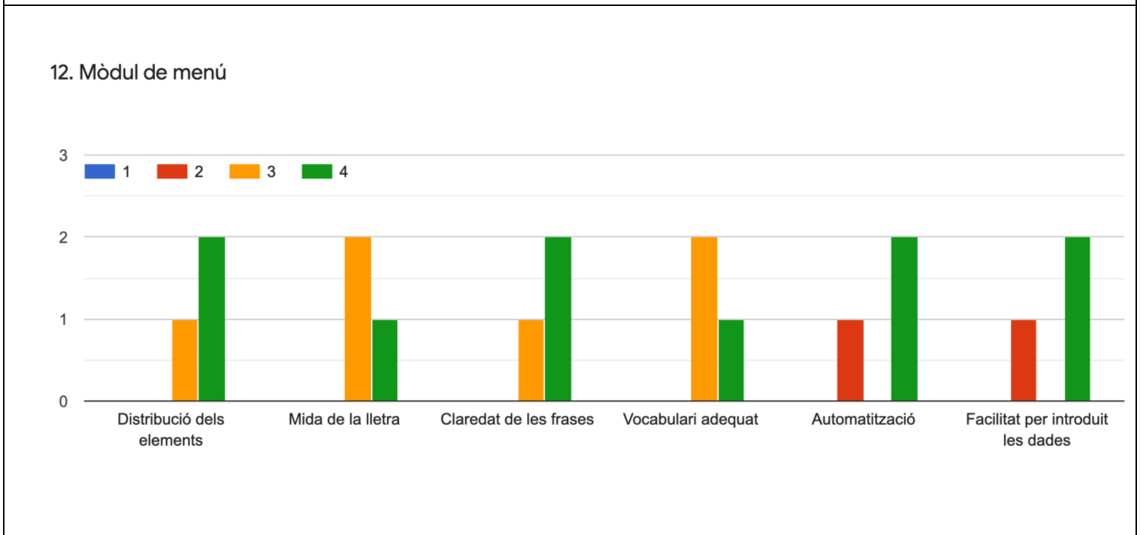




***Table A 2***. *Results of the software questionnaire*

## 12.3 User Manual

# MANUAL D'USUARI

Software director integrat per la gestió dels radiofàrmacs

AUTORA: Beatriz Martínez Ruiz

Barcelona, juny del 2022

# Índex

# 1. Introducció

Aquesta aplicació està desenvolupada per gestionar els radiofàrmacs des del moment que arriben al departament de radiofarmàcia fins que és administrat als pacients. És un sistema intuïtiu i fàcil d'utilitzar. A continuació, s'ha elaborat una explicació detallada dels documents complementaris de l'aplicació i seguidament, de cada pas a seguir per treballar amb les diferents finestres. Respecte a la segona part, es descriu cada component amb els seus corresponents requeriments que componen les següents pantalles:

- Comandes
- Recepcions
- Inventari
- Elucions
- Preparacions
- Magatzem
- Dispensació
- Gestió d'activitat
- Controls de qualitat
- Consultes
- Traçabilitat

# 2. Descripció de l'aplicació

Per accedir a la carpeta on es troba l'aplicació, us haureu de situar dins de la carpeta Nuclear. Un cop dins, haureu de prémer la carpeta Unitat de Radiofarmacia i finalment, la carpeta TFG.

A continuació, observareu un fitxer .db on es guardarà tota la informació, però no es pot visualitzar, i quatre carpetes:

- **CSV i Images:** són necessàries per a la creació de la base de dades i la manipulació de l'aplicació.
- **Templates:** es troben les plantilles d'Excel que s'ompliran quan es vulgui generar un informe. S'aconsella no modificar-les sense cap supervisió.
- **Reports:** la carpeta destinatària de tots els informes creats amb la informació de la base de dades.

A més, trobareu un fitxer .exe que consisteix en l'aplicació que haureu d'obrir.

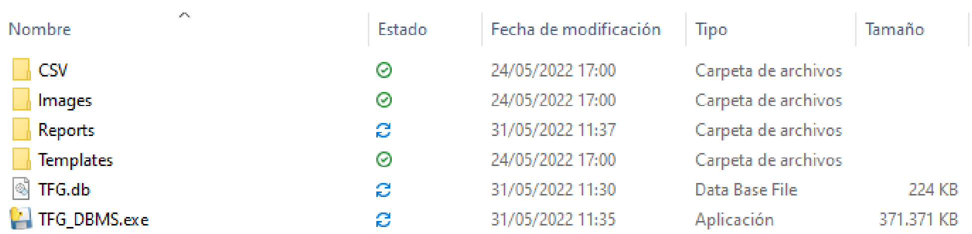



Figura 2.1: Carpeta TFG

# 3. Registre de l'usuari

Un cop oberta l'aplicació, apareixerà aquesta finestra en la qual introduint el nom de l'usuari i la seva contrasenya, permetrà al treballador manipular l'aplicació. En el cas que sigui correcta la informació introduïda, apareixerà una finestra informant que són correctes, en canvi, en el cas contrari, no deixaria continuar endavant amb l'aplicació i sortiria amb el següent text: "Usuari o contrasenya incorrecta".

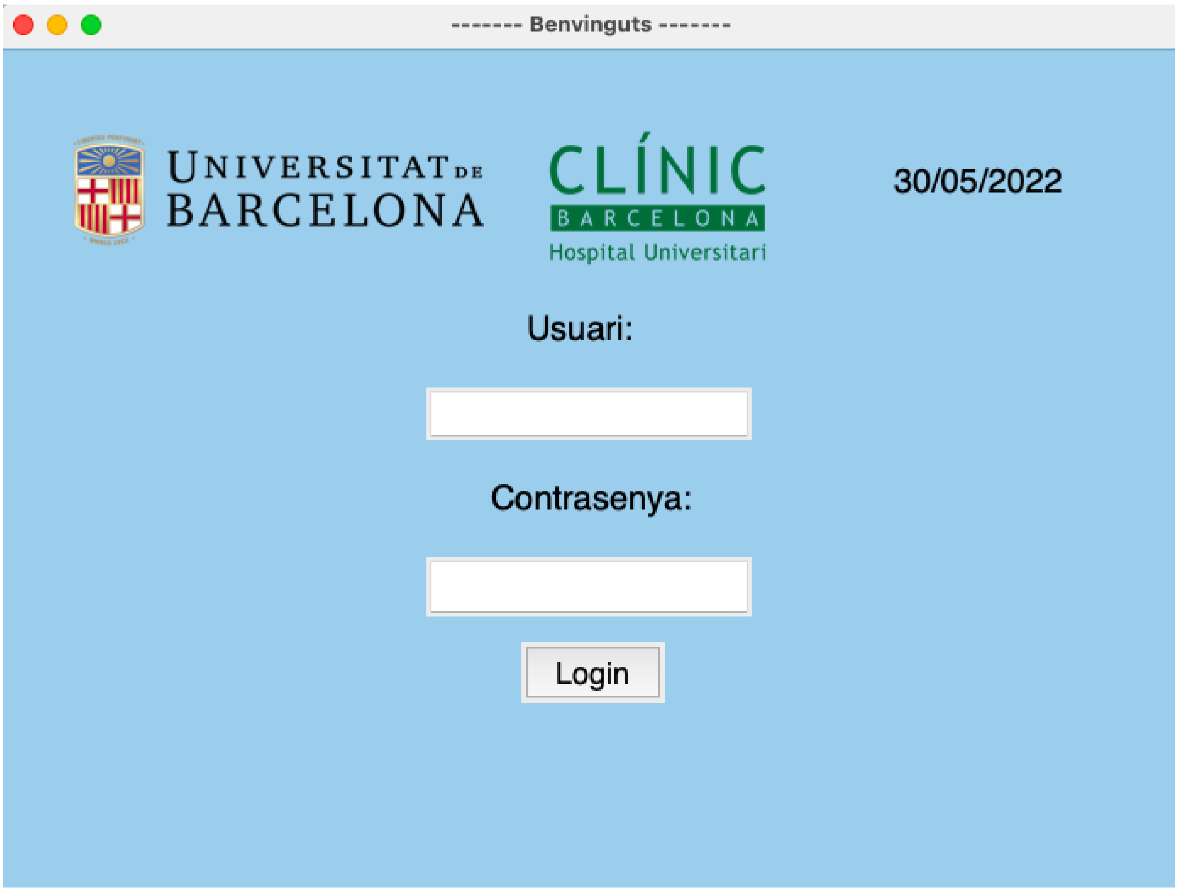



Figura 3.1: Finestra de registre

# 4. Menú

Depenent del sistema operatiu el menú apareixerà d'una manera o una altra. Aquest mòdul ens ofereix les següents funcions relacionades amb l'aplicació.

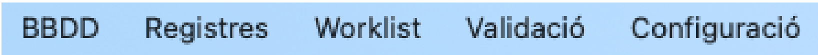



Figura 4.1: Barra de menú

## 4.1. Bases de dades

Aquesta opció ens ofereix dues funcions associades amb la base de dades.

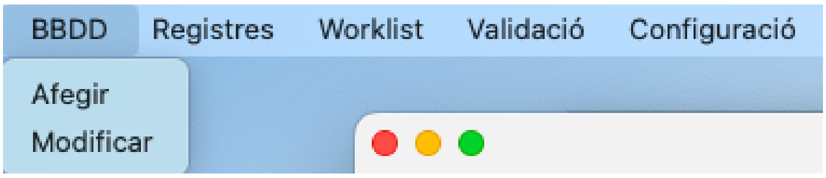



Figura 4.2: Opció de BBDD del menú

### 4.1.1 Afegir

En aquesta finestra, s'hauria d'escollir primer la taula de la qual es vol afegir un registre i a continuació, prement els dos botons $A$, apareixerà els noms de les columnes de la taula i l'últim registre d'aquesta per tenir una petita orientació de com es faria. Per separar cada paràmetre, s'ha de posar el signe de puntuació: ;.

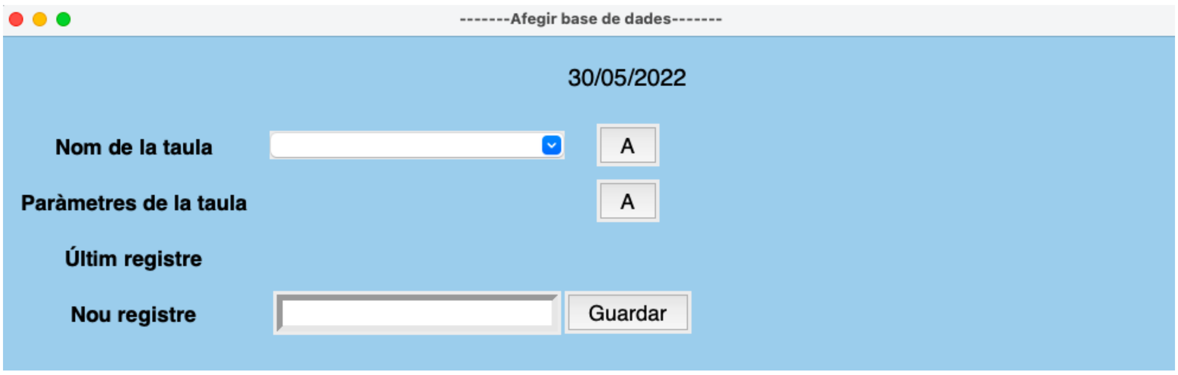



Figura 4.3: Finestra per afegir nous registres

### 4.1.2 Modificar

En aquest apartat, podem modificar les dades de les taules. Amb les llistes desplegables, podem escollir el nom de la taula, columna després de prémer el primer botó $A$. Després, amb el segon botó $A$, es mostrarà una llista amb el número d'identificació del registre i el paràmetre que es vol modificar. En l'entrada de sota, s'ha d'introduir el nou valor del registre i amb el botó Modificar, es guardarà el canvi.

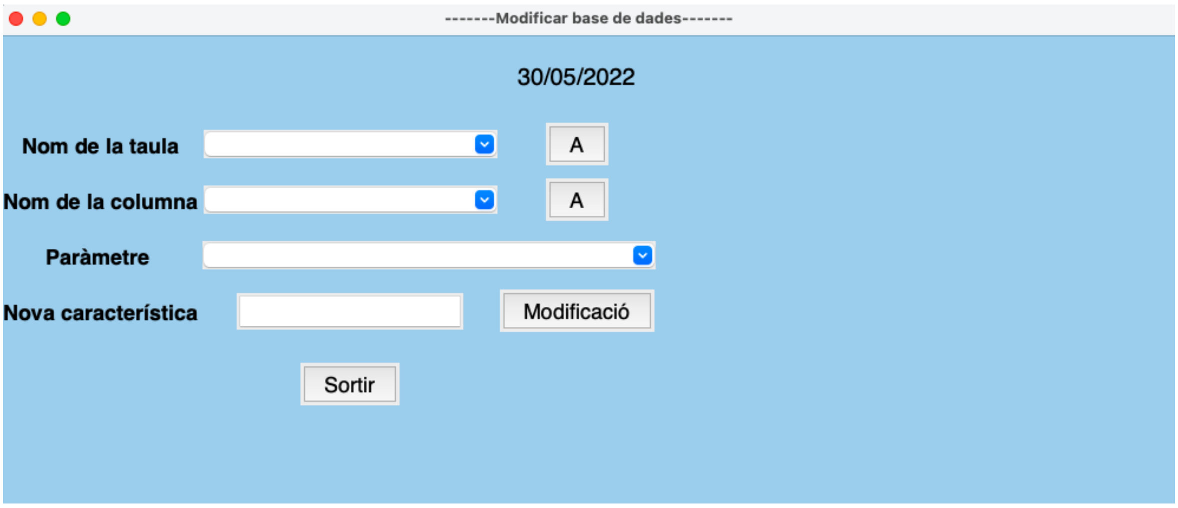



Figura 4.4: Finestra per modificar registres existents

## 4.2. Registres

Aquesta opció ens ofereix una funció relacionada amb els registres introduïts en l'aplicació.

### 4.2.1 Eliminar

En aquesta finestra, es pot eliminar qualsevol registre introduït en la base de dades. Els camps necessaris a omplir són:

- **Nom de la taula:** llista desplegable amb totes les taules de la base de dades. Important prémer el botó $A$, per obtenir la informació del camps següent
- **ID del registre:** llista desplegable de tots els números d'identificació de cada registre en la taula seleccionada.
- **Característiques del registre:** tots els paràmetres del registre escollit que es vol eliminar. Apareixerà amb el botó $A$, del camp anterior un cop seleccionat el registre.

Un cop escollit el registre de la taula determinada, amb el botó *Eliminació*, aquesta fila desapareixerà de la base de dades.


Figura 4.5: Finestra per eliminar registres existents

## 4.3. Worklist

Aquesta funció està dissenyada per introduït la llista de treball del SAP dins de la base de dades d'aquesta aplicació amb l'objectiu de poder manipular-la i consultar-la a l'hora de realitzar les comandes, preparacions, etc.

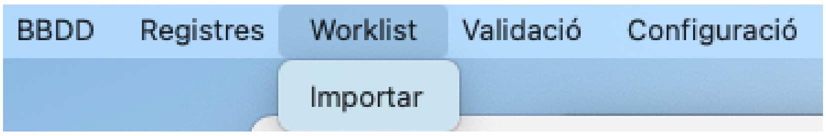



Figura 4.6: Opció de Worklist del menú

### 4.3.1 Importar

En aquesta finestra, s'introduirà un CSV document el qual conté informació sobre les exploracions que es realitzaran dins del departament de medicina nuclear. Només cal que s'escrigui el nom del fitxer amb el seu format i prémer a continuació, el botó Importar *worklist*.

**Informació important:** aquest fitxer és imprescindible que sigui en format CSV i el nom no pot portar cap barra lateral (/). Per exemple, *"WL1305.csv"*. Les dades han d'estar col·locades dins del document d'una manera determinada de manera que l'aplicació pugui interpretar la informació.

- En la primera columna, el dia que s'ha programat les proves mèdiques.
- En la segona columna, tindríem les hores programades d'injecció de la dosi al pacient.
- En la tercera columna, el nom del pacient.
- En la quarta columna, prova diagnòstica o terapèutica que està programada per dur-se a terme.

- En la cinquena columna, el número d'història clínica del pacient.

| | | | | |
|---|---|---|---|---|
| 30/5/22 | 8:00 | ENSALADA TOMATE, PEPINO ( 99 , H ) | GAMMAGRAFIA OSSIA | 0000000 |
| 30/5/22 | 8:30 | | GAMMAGRAFIA AMILOIDOSIS CARDIACA | XXXXXXX |
| 30/5/22 | 13:00 | | GAMMAGRAFIA PULMONAR DE PERFUSIO | XXXXXXX |
| 30/5/22 | 9:45 | | DETECCIO X2 GANGLI SENTINELLA | XXXXXXX |
| 30/5/22 | 10:00 | | DETECCIO DEL GANGLI SENTINELLA | XXXXXXX |
| 30/5/22 | 10:00 | | RENOGRAMA AMB DIURETIC | XXXXXXX |

Figura 4.7: Exemple de l'Excel de la llista de treball

Un cop s'ha importat la llista de treball, es pot modificar els noms de les exploracions de dues maneres. La primera es faria amb les dues llistes desplegables. En la d'exploració del SAP, apareixeran les proves mèdiques de la llista de treball del SAP que no tenen correspondència amb els noms de la llista proporcionada per radiofàrmacia. Del segon desplegable, s'escull una exploració i es guarda el canvi amb el primer botó *Guardar canvi.*

La segona manera consisteix en seleccionar el nom del pacient que volem i amb el botó *A*, sortirà la seva respectiva exploració i com l'altre cop, s'escull el nom de la prova de la llista de radiofàrmacia. Per finalitzar el canvi, s'ha de prémer el botó *Guardar canvi.*

És important que un cop realitzat tots els canvis necessaris sobre els noms de les exploracions, es premi el botó *Guardar llista de treball* per poder treballar amb ella dins de l'aplicació.


Figura 4.8: Finestra per importar i modificar la llista de treball

## 4.4. Validació

En aquesta funció del menú, l'usuari administrador té la possibilitat de validar les preparacions elaborades dins del departament de radiofarmàcia.

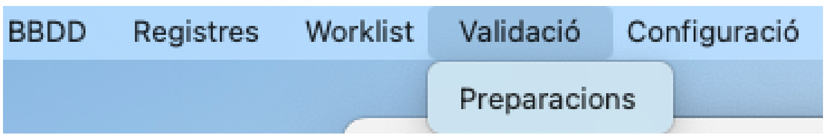



Figura 4.9: Opció de Validació del menú

### 4.4.1 Preparacions

Per validar les preparacions, se segueix el mateix procediment per a cada tipus. Primer de tot, escollir la data amb el botó *C*. A continuació, s'ha de prémer el botó *A* del costat del tipus de preparacions. S'escull el número d'identificació de la preparació que es vol i es prem el botó de control.

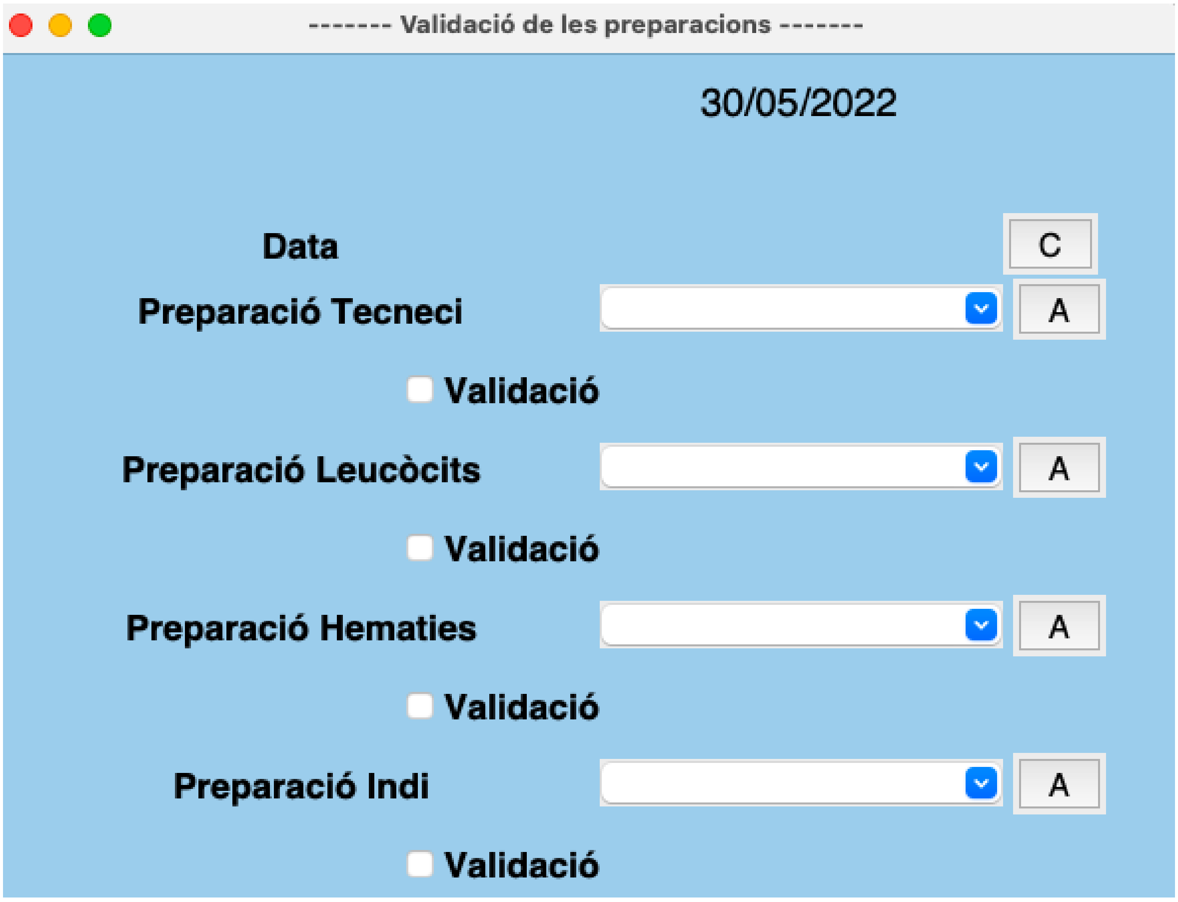



Figura 4.10: Finestra per validar les preparacions elaborades

## 4.5. Configuració

Aquesta opció permet tancar l'aplicació.

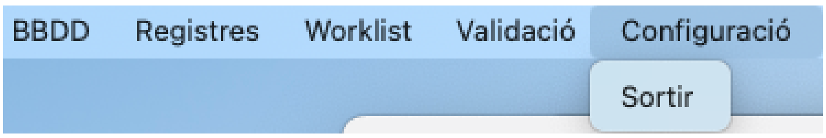



Figura 4.11: Opció de Configuració del menú

# 5. Pàgina inicial

Un cop us heu registrat, us apareixerà la següent imatge en la qual podeu observar diferents botons que cadascun portarà a una finestra nova per executar una tasca diferent. A més, hi ha un altre botó amb el qual podràs tancar tot el programa.


Figura 5.1: Finestra de la pàgina d'inici

# 6. Mòdul de comandes

L'apartat de Comanda està dissenyat per facilitar la gestió de comanda dins del departament de radiofarmàcia. En aquesta especialitat de farmàcia, es treballen amb determinats productes que es classifiquen en tres principals grups:

- Kits
- Radiofàrmacs
- Generadors de molibdè

Quan es prem el botó de *Comanda* de la pàgina inicial de l'aplicació, s'obre el formulari de Comanda que conté la següent informació:


Figura 6.1: Formulari de comandes

La finestra de comandes està composta pels següents camps:

- **Producte:** en aquest apartat, es desplega una llista de tots els productes que es poden demanar. Un cop seleccionat un determinat, es prem el botó del costat *Actualitzar* la funció del qual és emplenar automàticament els següents camps associats al producte escollit.

**Informació addicional:** en el cas que es vulgui modificar, afegir o eliminar la informació de la taula mare dels productes, s'ha d'anar a la secció de Bases de dades 4.1 i Registres 4.2

- **Proveïdor:** aquest camp apareixerà de manera automàtica un cop s'ha seleccionat un producte en l'apartat anterior. Ens dona el nom de la companyia del proveïdor.
- **Codi:** aquesta secció es mostrarà de la mateixa forma que l'apartat anterior, ja que són característiques úniques de cada producte. Fa referència al codi propi del producte.
- **Referència:** aquest apartat s'omplirà com la informació anterior del producte.
- **Format:** aquesta última informació automàtica del producte seleccionat ens informa sobre la manera com el producte arribarà que pot ser en activitat (mCi), nombre de vials, unitats, etc.
- **Núm. de compra:** aquesta entrada fa referència al número de compra.
- **Unitats:** nombre d'unitats del producte sol·licitat.
- **Data d'entrega:** dia en el qual s'espera que arribarà el producte. És un camp on no es permet que sigui anterior ni igual a la data actual.
- **Peticionari:** consta d'una llista desplegable dels treballadors habilitats a fer aquesta tasca.

Un cop s'ha omplert tots els camps relacionats amb la comanda d'un producte determinat, és important clicar el botó de *Guardar* per introduir aquesta informació dins de la base de dades. Addicionalment, a les dades prèviament comentades, es genera un número d'identificació de la compra compost per la lletra C + l'any quan s'ha dut a terme la comanda i per últim un número seqüencial. Per exemple, per la primera comanda de l'any 2022, tindríem: C2022/001.

A més, al costat es troba el botó *Informes* la funció del qual és omplir el document que està a continuació amb la informació del producte generat en aquell moment. Un cop generat aquest document, es guarda en format Excel a la memòria dins de la carpeta Reports amb el nom del producte comprat i la data en què s'ha efectuat aquesta acció.

SERVEI DE MEDICINA NUCLEAR
**UNITAT DE RADIOFARMÀCIA**

CLINIC BARCELONA Hospital Universitari

**SOL·LICITUD DE RADIOFÀRMAC**

| | |
|---|---|
| Per: | De: |
| Organització: | Data: |
| Correu electrònic: | Nº total de pàgines, porta inclosa: 1 |
| Assumpte: | **COMANDA DE RADIOFÀRMACIA** |

**Prego que ens subministrin el següent radiofàrmac per la data d'entrega indicada:**

| PRODUCTE | CODI | REFERENCIA | FORMAT | Nº DE SOL·LICITUD | UNITATS | DATA D'ENTREGA |
|---|---|---|---|---|---|---|
| | | | | | | |

Prego que confirmin el processat de la comanda

NOTA: FER CONSTAR AL ALBARA:
Nº CODI HCP
Nº COMANDA

GRÀCIES

Servei de medicina nuclear- IRA 0017. Hospital Clínic i Provincial.
Villaroel, 170. 08036 Barcelona

Figura 6.2: Informe de la comanda

Finalment, l'última acció que es pot realitzar en aquesta finestra és crear un fitxer com a resum de totes les comandes realitzades dins del període que s'ha establert amb el botó $C$ del costat esquerre i es guardarà a la carpeta Reports. A continuació, trobem un exemple.

SERVEI DE MEDICINA NUCLEAR

**UNITAT DE RADIOFARMÀCIA**

CLÍNIC BARCELONA Hospital Universitari

**COMANDES EFECTUADES**

Des de: Fins:

| PRODUCTE | PROVEÏDOR | CODI | FORMAT | Nº COMPRA | UNITATS | DATA ENTREGA | PETICIONARI | DATA |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |

Figura 6.3: Resum de comandes

# 7. Mòdul de recepció

En aquest mòdul, s'introduirà tota la informació relacionada amb els productes que hem rebut amb relació a les comandes executades. Aquest apartat està dividit en tres diferents grups d'acord amb les característiques de cadascun:

- Kits
- Generadors
- Material radioactiu

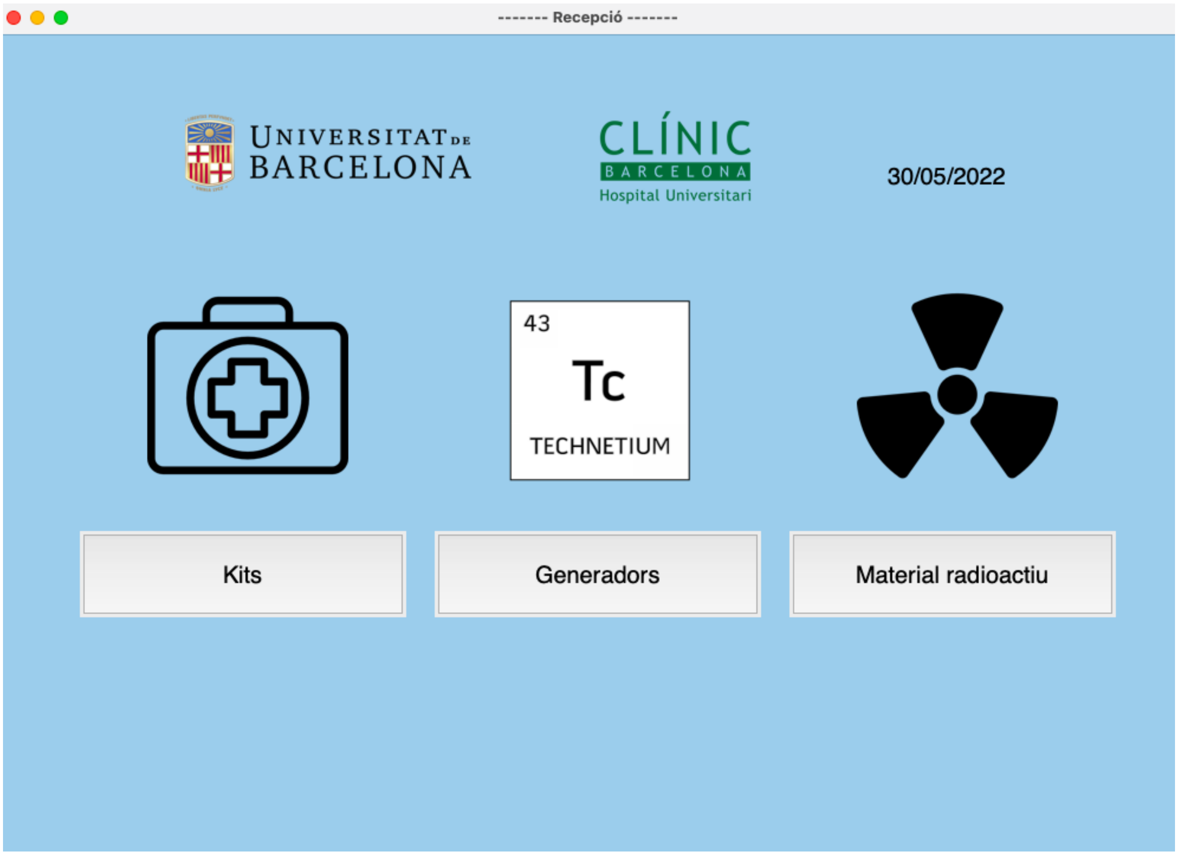



Figura 7.1: Pàgina principal de la recepció

## 7.1. Registre de kits

Introduint-nos en la recepció de kits, ens trobarem amb els següents camps:

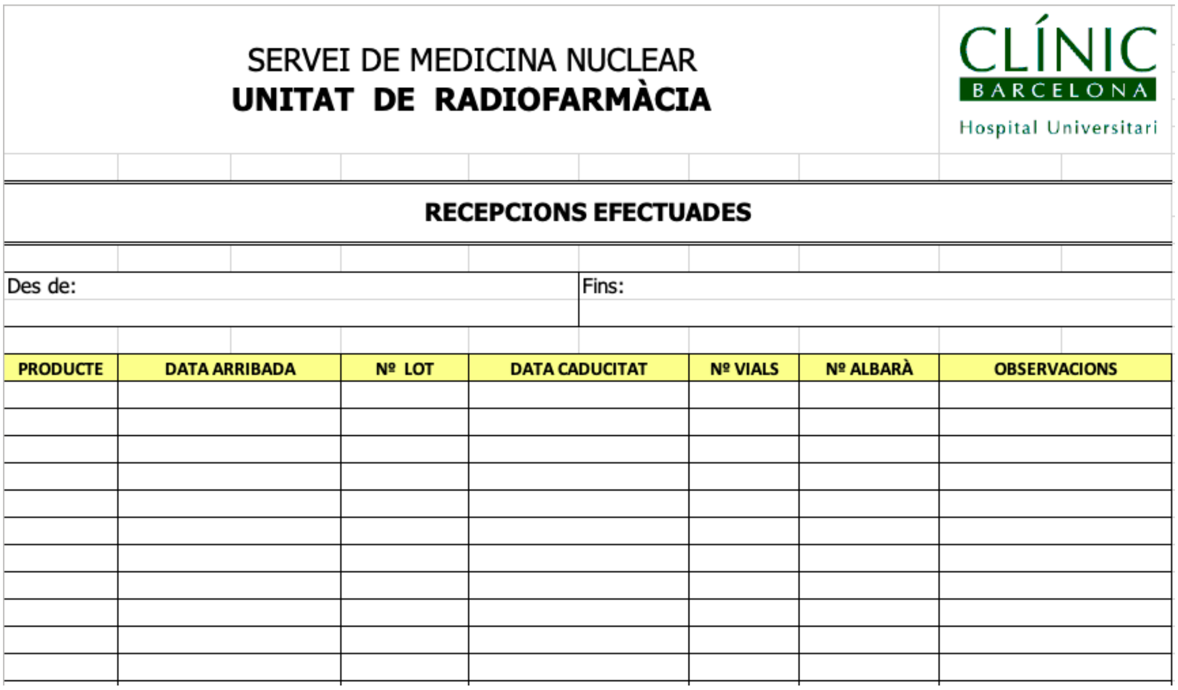
- **Kit:** consta d'una llista que desplega tots els kits que s'han demanat a excepció dels que ja s'han registrat. Les característiques que surten són el número d'identificació de la comanda, la data que s'espera d'arribar i el nom del kit. Per mostrar la informació relacionada al kit escollit, s'ha de prémer el botó *Actualitzar*.

- **Data esperada d'arribada:** és un camp automàtic que va relacionat amb l'anterior apartat i ens informa del dia quan en un principi, arribaria el kit.
- **Data de recepció:** aquesta secció és el dia quan realment ha arribat el kit i s'introdueix amb l'ajuda del botó de la dreta. Prement aquest botó, s'obre un calendari per escollir la data exacta. Només es poden introduir dates del mateix dia o dies posteriors.
- **Número de lot:** aquesta entrada fa referència al número del lot del fabricant de kit rebut.
- **Data de caducitat:** és un apartat que funciona de la mateixa manera que la data esperada d'arribada. Amb el botó, s'escull la data que hi ha restricció de manera que només deixa introduir dates del mateix dia o dies posteriors.
- **Número de vials:** la quantitat de vials que s'han rebut en aquesta recepció. Important que siguin nombre enter.
- **Número d'albarà:** correspon a la identificació de l'albarà.
- **Proveïdor:** és un apartat automàtic que es mostra un cop premut el botó *Actualitzar* al costat de la selecció dels kits. Ens informa del nom de la companyia relacionada amb la recepció del kit.
- **Codi nacional:** correspon al número de l'autorització de la comercialització específic de cada kit. És un apartat que surt de manera automàtica amb l'elecció del kit que es vol registrar.
- **Observacions**

Un cop omplert tots els apartats necessaris, per guardar aquesta informació s'ha de prémer el botó de *Guardar*. Realitzant aquesta acció, es crea un número d'identificació únic d'aquesta recepció que està compost per les lletres 'KR' amb l'any de la recepció i un número seqüencial. Per exemple: KR2022/001. A més, es guarda totes les dades introduïdes en el formulari.


Figura 7.2: Formulari pel registre dels kits

Finalment, l'última acció que es pot realitzar en aquesta finestra és crear un fitxer com a resum de totes les recepcions realitzades dins del període que s'ha establert amb el botó *C* del costat esquerre i es guardarà a la carpeta Reports. A continuació, trobem un exemple.



SERVEI DE MEDICINA NUCLEAR
**UNITAT DE RADIOFARMÀCIA**

CLÍNIC BARCELONA Hospital Universitari

**RECEPCIONS EFECTUADES**

Des de: Fins:

| PRODUCTE | DATA ARRIBADA | Nº LOT | DATA CADUCITAT | Nº VIALS | Nº ALBARÀ | OBSERVACIONS |
|---|---|---|---|---|---|---|
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

Figura 7.3: Informe de les recepcions de kits

## 7.2. Registre de generadors

Respecte a la recepció dels generadors de molibdè, ens trobem amb els camps següents:

- **Generador:** en aquest camp, es desplegarà una llista amb els corresponents generadors que s'han demanat i que encara no s'han registrat. La informació proporcionada correspon al número d'identificació de la demanda, data que s'espera l'arribava del generador i el tipus del generador.
- **Núm. del lot:** correspon al número d'identificació específic que cada producte proporcionat pel fabricant.
- **Activitat Mo (GBq):** activitat del generador proporcionada pel fabricant en data de la calibració.

  **Important:** si el número és decimal, és important que per separar les unitats amb els decimals es faci amb un punt.
- **Data d'arribada:** dia en el qual el generador s'ha rebut. Per introduir-la, s'ha de fer amb el botó *C* del costat dret. No ha de ser anterior a la data quan s'omple el formulari.
- **Hora d'arribada**
- **Data de calibració:** dia en el qual el generador s'ha calibrat amb el botó *C*.
- **Hora de calibració:** hora quan la calibració del generador s'ha efectuat i és important que segueixi un format estricte.
- **Data de caducitat:** dia en el qual el generador ja no és vàlid. Té una restricció que ha de ser al voltant de nou dies posteriors a la data de calibració.

Per guardar aquesta informació, s'ha de clicar el botó *Actualitzar* el qual automàticament genera un número d'identificació compost per la lletra 'G' més l'any i un número seqüencial de tres xifres.


Figura 7.4: Formulari per la recepció de generadors

A més, es pot crear un fitxer com a resum de totes les recepcions realitzades dins del període que s'ha establert amb el botó *C* del costat esquerre i es guardarà a la carpeta Reports. A continuació, trobem un exemple.

SERVEI DE MEDICINA NUCLEAR
**UNITAT DE RADIOFARMÀCIA**

CLÍNIC BARCELONA Hospital Universitari

**RECEPCIONS EFECTUADES**

Des de: | Fins:

| PRODUCTE | Nº ALBARÀ | Nº LOT | CALIBRACIÓ | ACTIVITAT NOMINAL | ARRIBADA | ACTIVITAT REBUDA | VOLUM | CADUCITAT | OPERADOR |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |

Figura 7.5: Informe de les recepcions de generadors

## 7.3. Registre de material radioactiu

En aquesta finestra, el formulari que s'ha d'emplenar posseeix els següents camps:

- **Material radioactiu:** en aquest apartat, es desplega una llista amb tots els materials radioactius que s'ha demanat i encara no s'han registrat. Aquesta informació està composta pel número d'identificació de la comanda, la data d'entrega i el nom de producte.
- **Número d'albarà:** el número d'acreditació de l'entrega del producte.
- **Número de lot:** el número particular del producte rebut proporcionat pel fabricant.
- **Data de calibració:** el dia que s'ha calibrat el producte.
- **Hora de calibració:** hora exacta de la calibració que ha d'introduir-se en un format específic (hh:mm).
- **Activitat nominal (mCi):** activitat de cada producte en mCi adquirida el dia i hora de calibració.
- **Data de recepció:** amb l'ajuda del botó *C* es pot introduir la data específica que ha de ser igual o posterior a l'actual data.
- **Hora de recepció:** hora en la qual s'ha rebut el producte per poder calcular l'activitat que ha arribat, en hh:mm.
- **Activitat rebuda (mCi):** és un camp automàtic que prové del càlcul relacionat amb la desintegració del material radioactiu.

- **Volum (ml):** no és un camp obligatori menys pels productes que s'utilitzaran per a les proves del PET.
- **Data de caducitat:** dia i hora quan el producte ja no serà efectiu per emprar-se.
- **Operador:** nom del treballador que està introduint la informació de la recepció.
- **Observacions**

**Important:** en els camps d'activitat nominal i volum, si el número és decimal, és important que per separar les unitats amb els decimals es faci amb un punt.

El botó de *Guardar* ens ajudaria a introduir les dades omplertes al formulari dins de la base de dades. Automàticament es genera un número d'identificació per la recepció del producte que està compost per les lletres 'MR', l'any i un número seqüencial de tres xifres.


Figura 7.6: Formulari per la recepció de material radioactiu

Finalment, també es pot crear un fitxer com a resum de totes les recepcions realitzades dins del període que s'ha establert amb el botó $C$ del costat esquerre i es guardarà a la carpeta Reports. A continuació, trobem un exemple.

SERVEI DE MEDICINA NUCLEAR
**UNITAT DE RADIOFARMÀCIA**

CLÍNIC BARCELONA Hospital Universitari

**RECEPCIONS EFECTUADES**

Des de: Fins:

| PRODUCTE | Nº LOT | ACTIVITAT MOLIBDÈ | DATA ARRIBADA | DATA CALIBRACIÓ | HORA CALIBRACIÓ | DATA CADUCITAT |
|---|---|---|---|---|---|---|
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

Figura 7.7: Informe de les recepcions de material radioactiu

# 8. Mòdul d'inventari

El mòdul d'inventari ens informa dels productes que estan disponibles i es divideix en tres apartat:

- Kits
- Generadors
- Material radioactiu

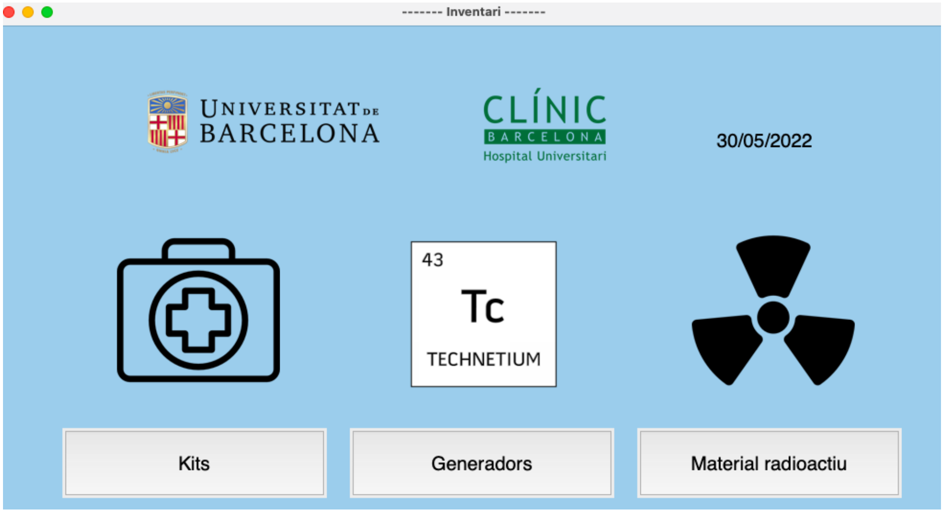



Figura 8.1: Pàgina principal de l'inventari

## 8.1. Inventari de kits

En aquesta finestra, es mostren tots els kits disponibles i els que li queden menys de trenta dies a caducar-se apareixeran en color taronja. En el moment que s'ha acabat un kit, el treballador pot introduir la quantitat de kits que s'han esgotat a l'entrada i ha de seleccionar el kit implicat. Finalment, s'ha de prémer el botó *Actualitzar* per visualitzar la informació de la taula.


Figura 8.2: Taula per visualitzar l'inventari de kits

## 8.2. Inventari de generadors

En aquesta taula, es pot visualitzar la informació relacionada amb els generadors disponibles amb la seva respectiva activitat disponible. En la columna de l'activitat pendent, apareix l'activitat en mCi que es pot eluir de tecneci. En aquest cas, quan es retiren generadors pel fet que ja no tenen activitat, s'ha de clicar el quadre al costat de la paraula "Retirat" i seleccionar el generador implicat.


Figura 8.3: Taula per visualitzar l'inventari de generadors

# 8.3. Inventari de materials radioactius

En aquesta finestra, es pot dividir en dos subgrups d'acord amb les seves aplicacions mèdiques. Per una banda, tenim tots els materials radioactius que venen preparats destinats per a cada pacient. D'altra banda, els materials radioactius utilitzats en la prova diagnòstica PET es troben en la segona taula. Per saber, la informació de cada producte a temps real d'acord amb la desintegració del radionúclid i l'extracció de dosis individuals per a cada pacient s'ha de prémer el botó *Actualitzar*.


Figura 8.4: Taula per visualitzar l'inventari de materials radioactius

# 9. Mòdul d'elucions

Aquest mòdul recull un formulari amb tota la informació necessària a l'hora de fer una elució:

- **Generador:** aquest camp correspon a una llista desplegable que dona informació sobre els generadors disponibles com el número d'identificació de la seva recepció, data d'arribada del generador i el nom del producte.
- **Operador:** és una llista que s'expandeix amb els noms dels tècnics.
- **Data d'elució:** data en què s'ha realitzat l'elució.
- **Hora:** moment exacte que s'ha portat a cap l'elució.
- **Volum (ml):** quantitat de tecneci extret, en ml.
- **Activitat eluïda A (mCi):** activitat real que s'ha eluït, en mCi.
- **Activitat teòrica A' (mCi):** activitat que hauria d'eluir el generador, paràmetre automàtic.
- **Rendiment elució:** relació amb l'activitat eluïda i la teòrica, paràmetre automàtic.
- **Pes $^{99}Tc + ^{99m}Tc$(µg):** pes present en l'elució, en µg, paràmetre automàtic.
- **Hora calibració:** nombre d'hores des que s'ha fet la calibració, paràmetre automàtic.
- **Hora des d'elució:** nombre d'hores des de l'última elució, paràmetre automàtic.
- **Activitat molibdè:** activitat que s'ha extret del generador de molibdè, paràmetre automàtic.
- **Activitat específica (mCi/µg):** relació entre el pes i l'activitat real eluïda, paràmetre automàtic.
- **Fracció molar $^{99m}Tc$:** la concentració de tecneci, paràmetre automàtic.

**Important:** en els camps de volum i activitat eluïda, si el número és decimal, és important que per separar les unitats amb els decimals es faci amb un punt.

Per saber la informació relacionada amb el generador escollit, es pot prémer el botó A del costat de la llista dels generadors disponibles. D'aquesta manera, apareixerà de manera automàtica les dades següents: el número d'identificació del generador obtingut a la seva recepció, el número de lot i l'activitat en GBq. D'altra banda, els últims camps associats amb l'elució executada apareixeran un cop s'ha premut el botó *Càlculs* i es guardaran dins de les bases de dades amb l'altre botó *Guardar* que automàticament es genera un número d'identificació específic per cada elució.

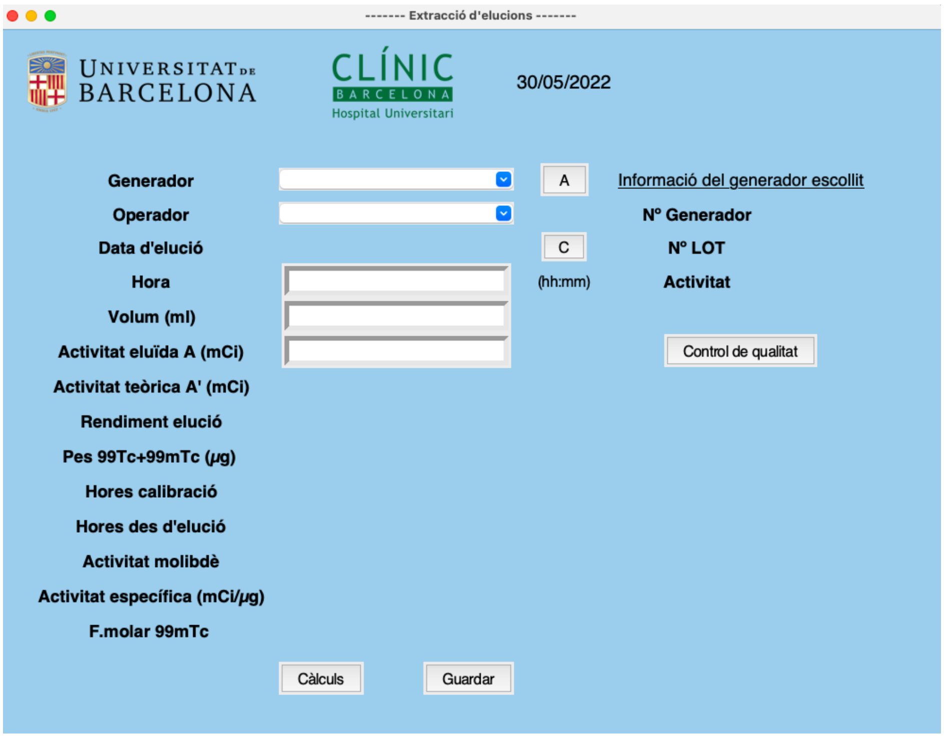



Figura 9.1: Formulari d'elucions

Per acabar, el formulari corresponent al control de qualitat s'accedeix amb el botó *Control de qualitat* i els camps que s'han d'omplir són els següents:

- **Elució:** s'estén un catàleg de les elucions que s'han realitzat del magatzem.
- **Data:** dia que s'ha dut a terme el test.
- **pH:** número relacionat amb l'acidesa i alcalinitat de la solució
- **Molibdè:** en µCi/mCi
- **Nivell d'alumini (ppm):** no és necessari que sigui un número.
- **Operador:** és una llista desplegada amb tots els noms dels treballadors.

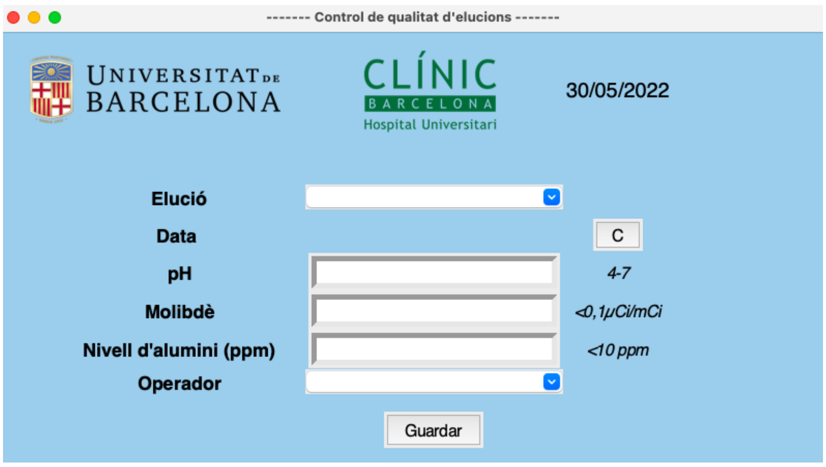



Figura 9.2: Formulari del control de qualitat d'elucions

# 10. Mòdul de preparacions

Aquest mòdul consisteix en la documentació de tota la informació relacionada amb la preparació de les radiofàrmacs. A l'esquerra, es troben els botons que us portarà al respectiu formulari per omplir:

- Preparacions amb tecneci
- Preparacions autòlogues amb tecneci i leucòcits
- Preparacions autòlogues amb indi
- Preparacions autòlogues amb tecneci i hematies.

A la dreta, tenim una taula amb les exploracions que s'han de fer aquell dia i que s'han importat en la secció llista de treball 4.3. Un cop s'ha executat una preparació, es pot introduir el nombre de pacients els quals rebran una dispensació i clicant la fila de l'examen implicat, apareixerà a la tercera columna. Quan els números de les columnes pacients i pacients fets coincideixen, el color de la lletra canviarà a verd.

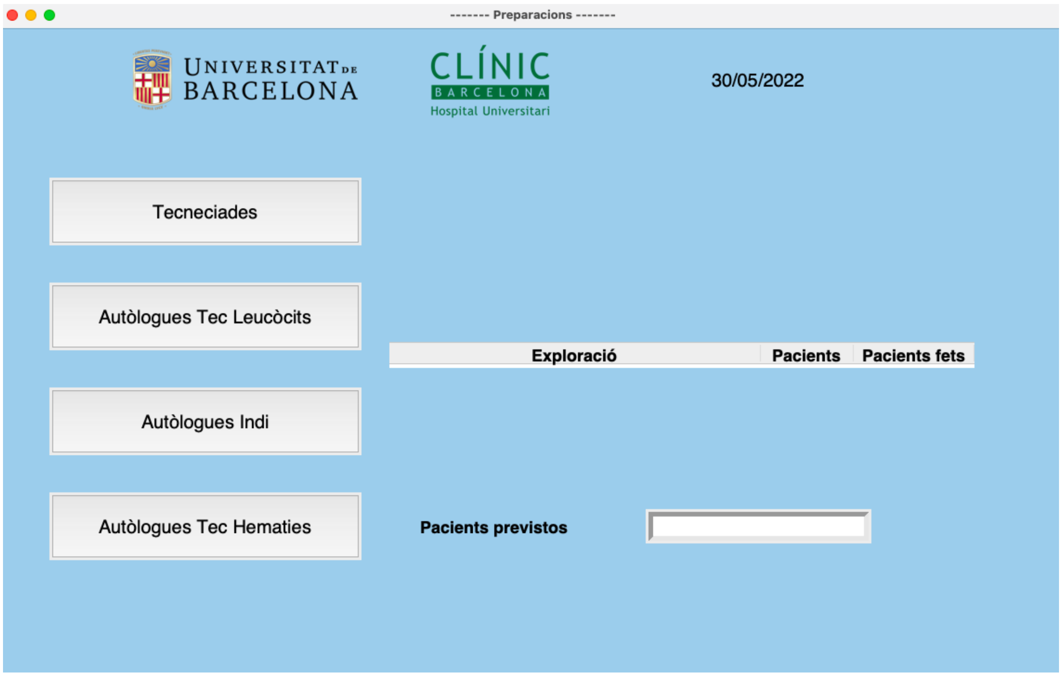



Figura 10.1: Pàgina inicial de les preparacions

## 10.1. Preparacions amb tecneci

Respecte a la preparació de radiofàrmacs amb tecneci, trobem els camps següents:

- **Data:** amb l'ajuda del botó $C$, s'introdueix la data de l'elaboració de la preparació.
- **Kit:** apareixerà una llista amb tots els kits que es localitzen en l'inventari amb unitats. La informació que es mostra consta d'un número d'identificació corresponent a la recepció del kit, el tipus i el lot.
- **Data de caducitat:** aquest camp ens dona informació del dia de la caducitat del kit i es mostrarà de manera automàtica amb el botó $A$.
- **Elució:** s desplega un catàleg amb les elucions amb activitat disponible. Per saber quina és la seva activitat en temps real, aneu a l'apartat de magatzem 11.1.
- **Activitat (mCi):** activitat eluïda en mCi.
- **Volum $^{99m}Tc$:** en ml.
- **Volum final:** en ml.
- **Hora:** hora exacta que s'executa l'elució seguint el format especificat al costat.
- **mCi/ml:** relació entre l'activitat i el volum de tecneci que apareixerà de manera automàtica amb el botó $A$.
- **PRQ:** en percentatge.
- **Tècnic:** llista desplegable amb els noms dels treballadors.
- **Observacions**
- **pH:** mesura quantitativa de l'acidesa o basicitat d'una dissolució.

**Important:** en els camps de volum, activitat i pH, si el número és decimal, és important que per separar les unitats amb els decimals es faci amb un punt.

Per introduir les dades a la base de dades, s'ha de prémer el botó *Guardar* que automàticament generarà un número d'identificació de la preparació executada.

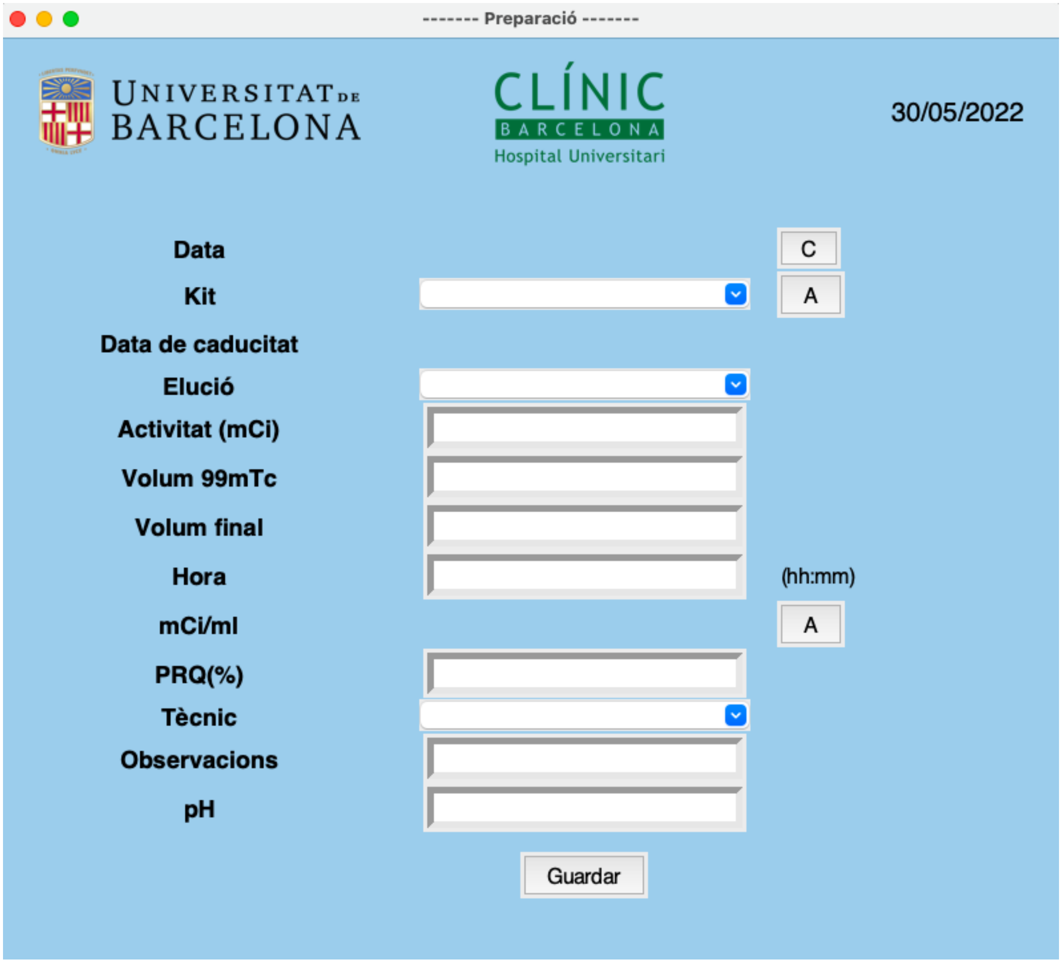



Figura 10.2: Formulari de les preparacions amb tecneci

## 10.2. Preparacions autòlogues amb tecneci i leucòcits

En aquesta finestra, podem observar els següents camps:

- **NHC i inicials:** Número d'Història Clínica i inicials del pacient.
- **Data:** amb l'ajuda del botó *C*, es pot introduir el dia exacte de l'elaboració de la preparació.
- **Hora:** en format hh:mm.
- **Cabina:** sigles de les dues cabines: CFL3 o CFL4.
- **Volum PRL:** volum de plasma ric en leucòcits, en ml.
- **Elució:** llista desplegable amb la informació de les elucions disponibles. Per saber quina és la seva activitat en temps real, aneu a l'apartat de magatzem 11.1.
- **HMPAO kit:** llista desplegable dels kits HMPAO disponibles amb el seu número d'identificació de la recepció, el dia de recepció i el lot.
- **Volum HMPAO kit:** en ml.

- **Activitat Tc-HMPAO:** en mCi
- **Activitat leucòcits:** en mCi
- **Activitat sobrenedant:** en mCi
- **Rendiment:** paràmetre automàtic sobre la relació entre l'activitat dels leucòcits i l'activitat sobrenedant en
- **Activitat de la xeringa:** en mCi
- **Volum de la xeringa:** en ml.
- **Tècnic:** llista desplegable dels noms dels treballadors.

**Important:** en els camps de volum i activitat, si el número és decimal, és important que per separar les unitats amb els decimals es faci amb un punt.

El botó *Guardar* ens ajuda a preservar la informació introduïda dins de les bases de dades amb la creació d'un número d'identificació de la preparació elaborada.

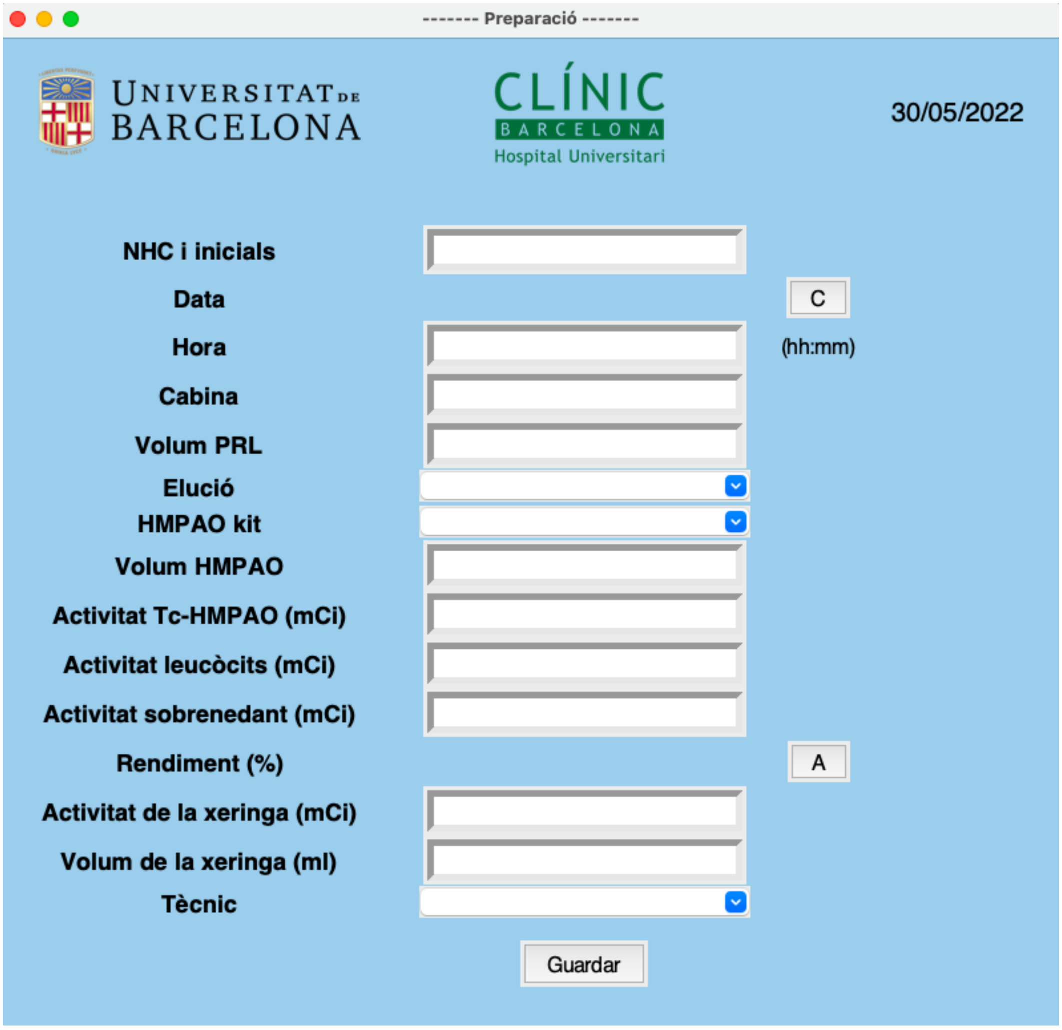



Figura 10.3: Formulari de les preparacions amb tecneci i leucòcits

## 10.3. Preparacions autòlogues amb indi

En aquest apartat, es troben els camps següents:

- **NHC i inicials:** Número d'Història Clínica i inicials del pacient.
- **Data:** es selecciona amb l'ajuda del botó *C* on apareixerà un calendari.
- **Hora:** en format hh:mm,
- **Cabina:** sigles de les dues cabines: CFL3 o CFL4.
- **Volum PRC:** volum de plasma ric en cèl·lules.
- **Cèl·lules:** llista desplegable amb els dos tipus de cèl·lules que s'utilitzen.
- **111In-Oxina kit:** llista desplegable de kits de 111In-Oxina disponibles.
- **Lot tampó:** número del tampó fet servir en aquesta preparació.
- **Activitat Indi:** en mCi.
- **Activitat cèl·lules:** en mCi.
- **Activitat sobrenedant:** en mCi
- **Rendiment:** paràmetre automàtic que informa sobre la relació entre l'activitat de les cèl·lules i sobrenedant en percentatge.
- **Activitat de la xeringa:** en mCi
- **Volum de la xeringa:** en ml.
- **Tècnic:** llista desplegable dels noms dels treballadors.

**Important:** en els camps de volum i activitat, si el número és decimal, és important que per separar les unitats amb els decimals es faci amb un punt.

Per guardar la informació que s'ha omplert en el formulari, es prem el botó *Guardar* i automàticament es crearà un número d'identificació associat amb la preparació.

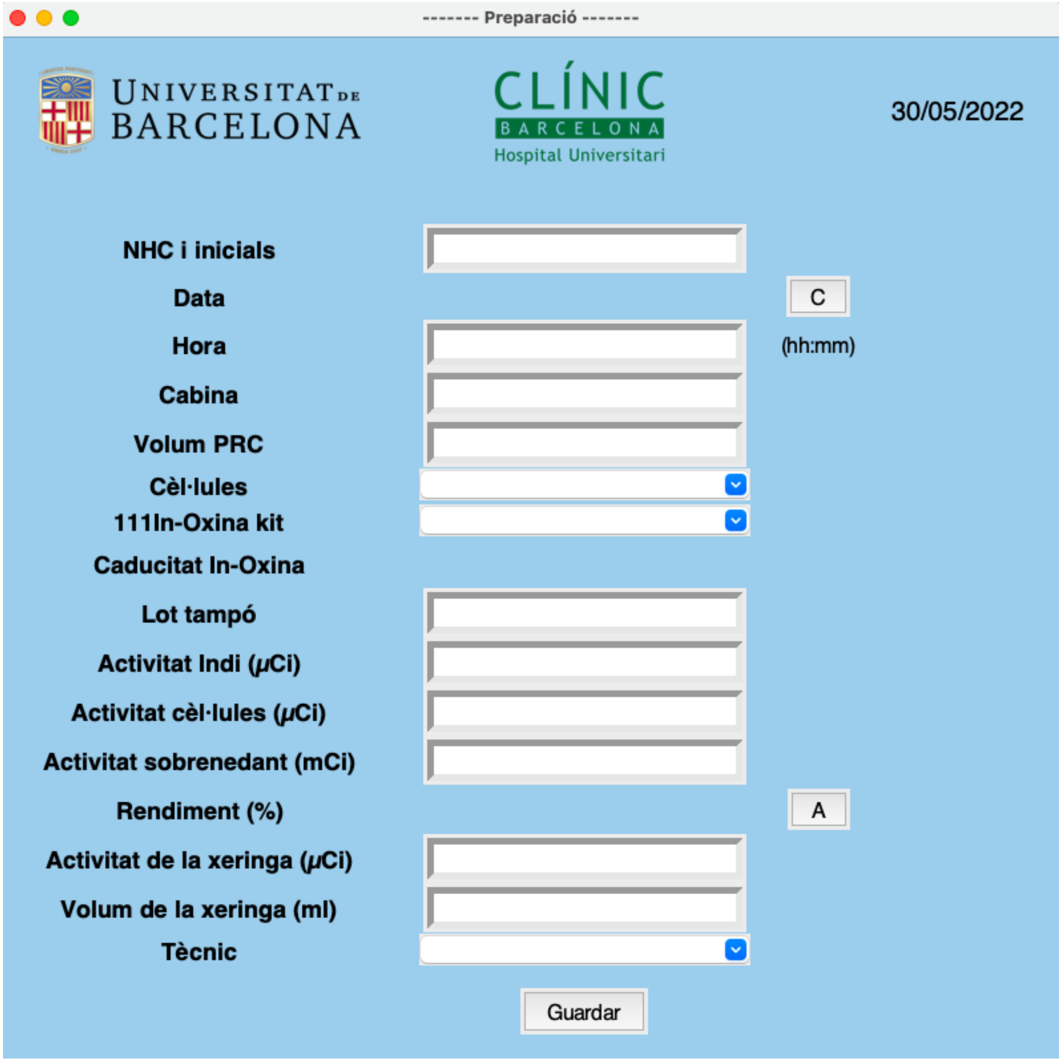



Figura 10.4: Formulari de les preparacions amb indi

## 10.4. Preparacions autòlogues amb tecneci i hematies

En aquesta finestra, trobarem els següents camps:

- **NHC i inicials:** Número d'Història Clínica i inicials del pacient.
- **Data:** amb l'ajuda del botó *C*.
- **Hora:** en format hh:mm.
- **Cabina:** sigles de les dues cabines: CFL3 o CFL4.
- **Elució:** llista desplegable amb totes les elucions disponibles. Per saber quina és la seva activitat en temps real, aneu a l'apartat de magatzem 11.1.
- **PYP kit:** : llista amb tots els kits disponibles de PYP.
- **Caducitat PYP:** paràmetre automàtic relacionat amb l'anterior apartat i apareix quan es prem el botó *A*.

- **Activitat tecneci:** en mCi.
- **Activitat eritròcits:** en mCi.
- **Activitat sobrenedant:** en mCi.
- **Rendiment:** paràmetre automàtic que relaciona l'activitat dels eritròcits amb la sobrenedant.
- **Activitat de la xeringa:** en mCi
- **Volum de la xeringa:** en ml.
- **Tècnic:** llista desplegable dels noms dels treballadors.

**Important:** en els camps de volum i activitat, si el número és decimal, és important que per separar les unitats amb els decimals es faci amb un punt.

Un cop omplert tots els camps, es prem el botó *Guardar* per preservar aquesta informació dins de la base de dades.

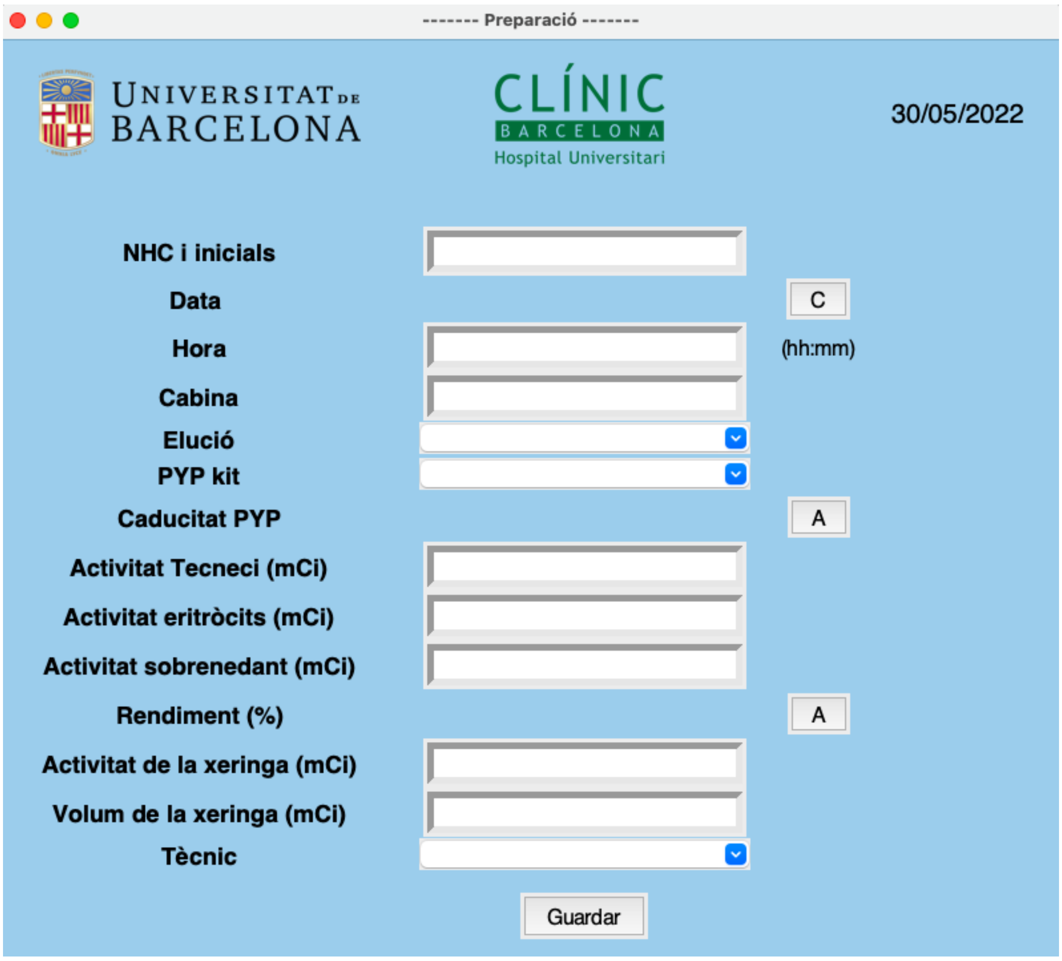



Figura 10.5: Formulari de les preparacions amb tecneci i hematies

# 11. Mòdul de magatzem

Aquest mòdul està dissenyat a introduir tota la informació necessària per extreure la dosi requerida per a cada pacient de manera que facilitaria saber en un hipotètic cas quins lots han sigut utilitzats per un pacient. Està dividit en dos apartats depenent de la màquina d'adquisició d'imatges que s'emprarà.


Figura 11.1: Pàgina principal del magatzem

## 11.1. Magatzem d'elucions

En el cas d'elucions, per mostrar les seves dades, s'ha de prémer el botó *Actualitzar*, ja que l'activitat pendent és a temps real.


Figura 11.2: Taula de magatzem d'elucions

## 11.2. Magatzem de preparacions amb tecneci

Taula amb tota la informació sobre les preparacions amb tecneci del dia. En el cas que es vulgui consultar les preparacions amb tecneci elaborades un altre dia, s'ha d'escollir un dia del calendari que apareixerà prement el botó *C* i a continuació, actualitzar la taula amb el botó *Actualitzar*.


Figura 11.3: Taula de magatzem de les preparacions amb tecneci

## 11.3. Magatzem de preparacions amb tecneci i leucòcits

Taula amb tota la informació sobre les preparacions amb tecneci i leucòcits del dia. En el cas que es vulgui consultar les preparacions amb tecneci i leucòcits elaborades un altre dia, s'ha d'escollir un dia del calendari que apareixerà prement el botó *C* i a continuació, actualitzar la taula amb el botó *Actualitzar*.


Figura 11.4: Taula de magatzem de les preparacions amb tecneci i leucòcits

## 11.4. Magatzem de preparacions amb indi

Taula amb tota la informació sobre les preparacions amb indi del dia. En el cas que es vulgui consultar les preparacions amb indi elaborades un altre dia, s'ha d'escollir un dia del calendari que apareixerà prement el botó *C* i a continuació, actualitzar la taula amb el botó *Actualitzar*.


Figura 11.5: Taula de magatzem de les preparacions amb indi

## 11.5. Magatzem de preparacions amb tecneci i hematies

Taula amb tota la informació sobre les preparacions amb tecneci i hematies del dia. En el cas que es vulgui consultar les preparacions amb tecneci i hematies elaborades un altre dia, s'ha d'escollir un dia del calendari que apareixerà prement el botó *C* i a continuació, actualitzar la taula amb el botó *Actualitzar*.


Figura 11.6: Taula de magatzem de les preparacions amb tecneci i hematies

# 12. Mòdul de dispensació

Aquest mòdul està dissenyat a introduir tota la informació necessària per extreure la dosi requerida per a cada pacient de manera que facilitaria saber en un hipotètic cas quins lots han sigut utilitzats per un pacient. Està dividit en dos apartats depenent de la màquina d'adquisició d'imatges que s'emprarà.


Figura 12.1: Pàgina principal de les dispensacions

## 12.1. Dipensació

En el cas de les dispensacions destinades a proves de SPECT i gammacàmeres, els camps del formulari són els següents:

- **Història clínica del pacient:** llista desplegable amb els pacients. Per poder escollir el pacient determinat, la informació que es mostrarà és el número d'identificació dels pacients dins de la base de dades, el número d'història clínica i el nom.
- **Exploració:** és un paràmetre automàtic que es mostrarà amb el botó A un cop seleccionat el pacient.
- **Preparació:** d'acord amb l'exploració que s'ha d'executar al pacient, els elements de la llista desplegable correspondran als marcatges específics.
- **Hora d'extracció:** en format hh:mm.
- **Activitat:** en mCi.

  A més, per preservar la informació introduïda, s'utilitzarà el botó *Guardar*.

**Important:** en els camps de volum i activitat, si el número és decimal, és important que per separar les unitats amb els decimals es faci amb un punt.


Figura 12.2: Formulari per les dispensacions

Finalment, el botó *Etiqueta* està dissenyat per generar l'etiqueta corresponent com la següent imatge:

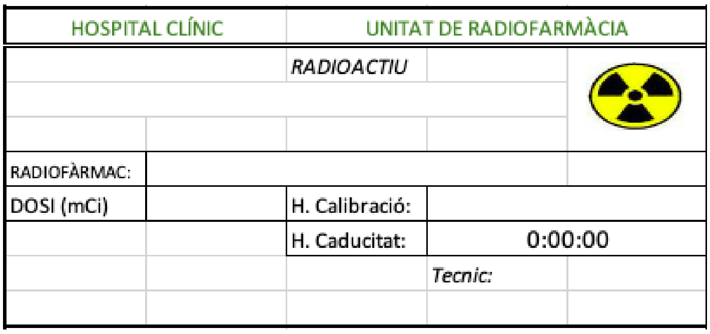

HOSPITAL CLÍNIC | UNITAT DE RADIOFARMÀCIA
RADIOACTIU
RADIOFÀRMAC:
DOSI (mCi) | H. Calibració:
H. Caducitat: | 0:00:00
Tecnic:

Figura 12.3: Etiqueta

## 12.2. Dipensació PET

En aquesta finestra, s'introduirà la informació sobre dispensacions de dosis destinades a les proves de PET. Ens trobem amb els camps següents:

- **Història clínica del pacient:** llista desplegable amb el número d'identificació de la prova, història clínica del pacient i el nom.
- **Hora i exploració:** d'acord amb la selecció anterior, apareixerà l'exploració que es farà el pacient i l'hora programada. És un camp automàtic que es mostrarà amb el botó *A*.
- **Preparació:** llista desplegable amb les preparacions disponibles pel PET.
- **Pes:** del pacient en kg.
- **Hora d'extracció:** en format hh:mm.
- **Hora d'injecció:** en format hh:mm.
- **Dosi:** apareixerà la dosi necessària, en mC, que s'hauria d'extraure respecte a l'hora introduïda.
- **Volum:** volum que s'hauria d'extreure, en ml.
- **Activitat extreta:** en mCi.
- **Volum extret:** en ml.
- **Hora final d'extracció:** en format hh:mm.
- **Activitat:** en mCi.

**Important:** en els camps de volum i activitat, si el número és decimal, és important que per separar les unitats amb els decimals es faci amb un punt.

De la mateixa manera que l'altre apartat de dispensacions, es poden guardar les dades amb el botó *Guardar* i generar l'etiqueta amb el botó *Etiqueta*.

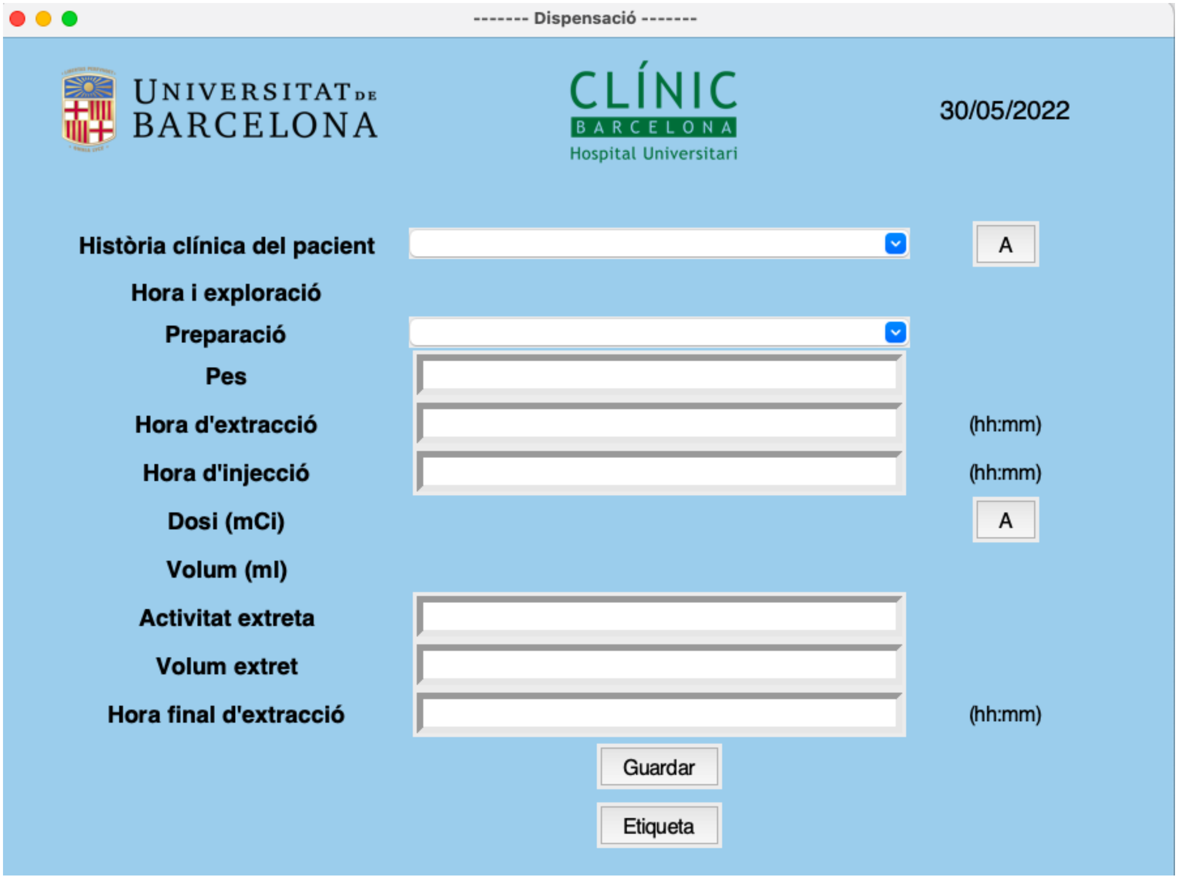



Figura 12.4: Formulari per les dispensacions pel PET

# 13. Mòdul de gestió d'activitat

Aquest mòdul està dissenyat per generar un informe sobre la gestió de l'activitat de cada material radioactiu i generadors que s'utilitza en la radiofarmàcia. Trobem els següents camps:

- **Material radioactiu:** llista desplegable amb els que es treballen, on surt el seu número d'identificació i el seu nom.
- **Generador:** llista desplegable dels diferents generadors emprats, on surt el seu número d'identificació i el tipus.
- **Des de:** orrespon el primer dia que vols la informació i s'escull amb el botó *C*.
- **Fins:**quan vols l'activitat.


Figura 13.1: Pàgina per la gestió de l'activitat ingresas

Amb el botó *Informe*, es crearà un Excel dins de la carpeta Templates amb el nom del material radioactiu, l'any i el més quan s'ha generat.

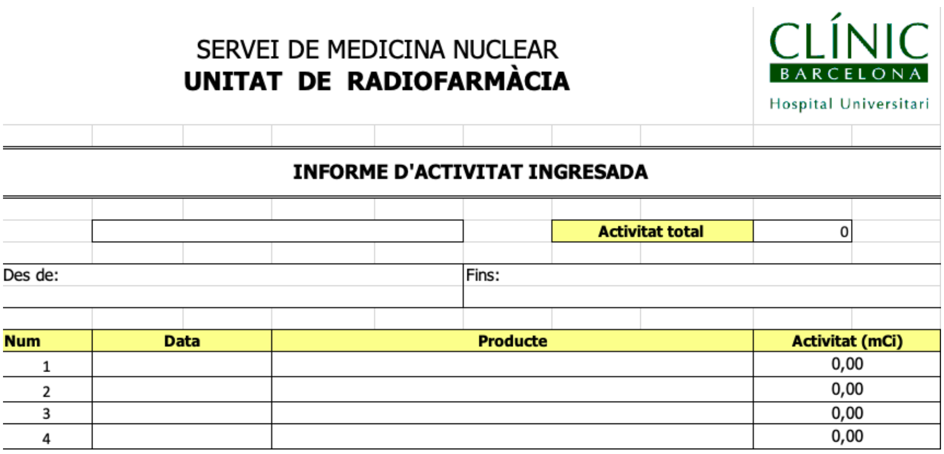



Figura 13.2: Informe sobre l'activitat ingresada

# 14. Mòdul de controls de qualitat

Aquest mòdul està destinat a ensenyar les dades associades al control de qualitat realitzat dins de radiofarmàcia.


Figura 14.1: Pàgina inicial dels controls de qualitat

### 14.0.1 Controls de qualitat de generadors

En aquesta finestra, es pot visualitzar tota la informació dels controls de qualitat realitzats a les elucions del generador de molibdè.


Figura 14.2: Taula dels controls de qualitat dels generadors

### 14.0.2 Controls de qualitat de preparacions amb tecneci

En aquesta finestra, es pot visualitzar tota la informació dels controls de qualitat realitzats a les preparacions amb tecneci i bé determinat amb el paràmetre PRQ.


Figura 14.3: Taula dels controls de qualitat de preparacions amb tecneci

### 14.0.3 Controls de qualitat de preparacions amb tecneci i leucòcits

En aquesta finestra, es pot visualitzar tota la informació dels controls de qualitat realitzats a les preparacions cel·lulars amb tecneci i leucòci.ts


Figura 14.4: Taula dels controls de qualitat de preparacions amb tecneci i leucòcits

### 14.0.4 Controls de qualitat de preparacions amb indi

En aquesta taula, es pot visualitzar tota la informació dels controls de qualitat realitzats a les preparacions cel·lulars amb indi.


Figura 14.5: Taula dels controls de qualitat de preparacions amb indi

### 14.0.5 Controls de qualitat de preparacions amb tecneci i hematies

En aquesta finestra, es pot visualitzar tota la informació dels controls de qualitat realitzats a les preparacions cel·lulars amb tecneci i hematies.


Figura 14.6: Taula dels controls de qualitat de preparacions amb tecneci i hematies

# 15. Mòdul de consultes

Aquest mòdul està dissenyat per realitzar consultes de les dispensacions que s'han realitzat dins de radiofarmàcia. En un primer moment, sortiran les executades en el dia. Per visualitzar les d'un altre dia, s'haurà d'escollir un dia del calendari i prémer el botó d'*Actualitzar*.


Figura 15.1: Taula per les consultes

# 16. Mòdul de traçablitat

Aquest mòdul ens permet realitzar la traçabilitat dels quatre següents paràmetres amb el botó *Buscar*:

- **Pacient:** es mostrarà el número d'identificació de la preparació que s'ha injectat al pacient, l'exploració que s'ha realitzat, l'hora i la data de la injecció, l'activitat i el volum de la dosi.
- **Kit:** es mostrarà el número d'identificació,, el nom i el lot del kit i en l'apartat d'informació addicional, sortirà el número d'identificació de la preparació que s'ha elaborat i l'exploració que s'ha fet, l'hora i la data de la injecció, l'activitat i el volum de la dosi.
- **Generador:** el número d'identificació i el lot del generador i la següent informació relacionada amb les dosis extretes: el número d'identificació de la preparació que s'ha elaborat i l'exploració que s'ha portat a cap, l'hora i la data de la injecció, l'activitat i el volum de la dosi.
- **Radiofàrmac:** el número d'identificació, el nom i el lot del radiofàrmac escollit, el nom del pacient, l'exploració que s'ha realitzat, l'hora i la data de la injecció, l'activitat i el volum de la dosi.
- **Preparacions:** llista desplegable dels números d'identificació de totes les preparacions elaborades dins de radiofarmàcia.

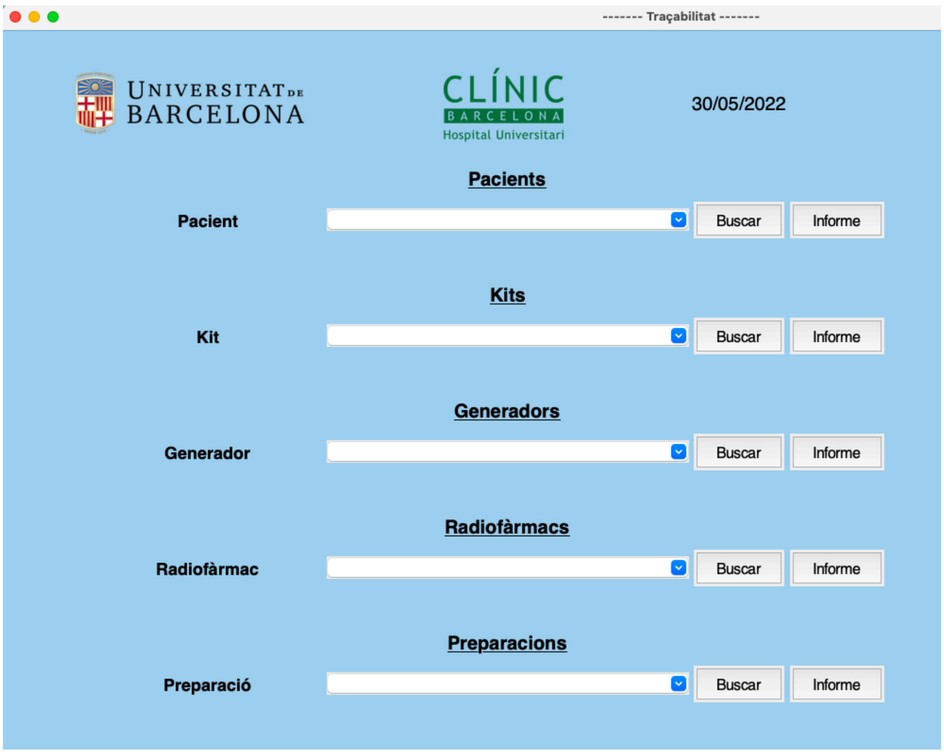



Figura 16.1: Pàgina per realitzar traçabilitats

**Informació addicional:** en cada cas, es pot generar un document on recull tota la informació dels pacients involucrats o les dispensacions realitzades a cada pacient. Quan es prem el botó Informe i es visualitzarà un nou document dins de la carpeta de Reports. En el cas dels pacients, s'anomena Report_traceability + el nom del pacient. En els tres casos posteriors, es diu Report_traceability + el camp seleccionat. Finalment, en el cas de preparacions, es titula Report_traceability + el número d'identificació de la preparació escollida. A continuació, tenim un exemple respecte al tres models de manera ordenada.

SERVEI DE MEDICINA NUCLEAR
**UNITAT DE RADIOFARMÀCIA**

CLÍNIC BARCELONA Hospital Universitari

**TRAÇABILITAT**

Kit/preparació/radiofàrmac

| DATA | PREPARACIÓ (ID) | PACIENT | MRC | EXPLORACIÓ | DOSI |
|---|---|---|---|---|---|

Figura 16.2: Informe Report_traceability_pacient

SERVEI DE MEDICINA NUCLEAR
**UNITAT DE RADIOFARMÀCIA**

CLÍNIC BARCELONA Hospital Universitari

**TRAÇABILITAT**

Pacient:

| DATA | HORA | PREPARACIÓ (ID) | ACTIVITAT | VOLUM | EXPLORACIÓ |
|---|---|---|---|---|---|

Figura 16.3: Informe Report_traceability

SERVEI DE MEDICINA NUCLEAR
**UNITAT DE RADIOFARMÀCIA**

CLÍNIC BARCELONA Hospital Universitari

**TRAÇABILITAT**

Preparació:

| DATA | HORA | PACIENT | ACTIVITAT | VOLUM | EXPLORACIÓ |
|---|---|---|---|---|---|

Figura 16.4: Informe Report_traceability_preparation