Politecnico di Milano

Medical Informatics

Prof. Anna Maria Bianchi - Medical Informatics

HYPERTENSION LEVEL MONITORING SYSTEM

Part I: UML modeling

Part II: Implementation

Cubeddu Fabio 896052
Mazzotta Gian Marco 896720
Stagno Emanuele 900340
Tranquillo Francesco Raffaele 905980

Academic Year 2017-2018

INDEX

Part I: UML modeling

1 INTRODUCTION

There are strong evidences of the relation between job-related stress level and hypertension. Modern workplaces provide at-work medical offices in which specialized practitioners administer tests and perform measurements such as blood pressure and heart rate. Some of these parameters can be collected either at the workplace or at home, while others might be collected in both.

1.1 Aim of the project

The aim of the Project was to develop a system which allows the in-home and at-work monitoring of several biological parameters with the scope of keeping track of the level of hypertension.

The system includes:

- *Actors authentication*
- *Stress levels questionnaires*
- *Medical visits*
- *Wearable devices*
- *Set of parameters* (divided in “at-work” and “at-home” measured parameters)

Actor authentication allows every user, such as Specialized Practitioners, employees (patients) or technical administrator, to have access to different type of data or to the same data in a different way.

Stress levels questionnaires: Stress is one of the main determinants of health status. Therefore, an instrument to adequately measure stress is of prime interest not only in public health research but also for the examination of educational returns. School and workplace requirements are both essential sources of stress, and stress levels can also be affected by unemployment.

The used Questionnaires were composed by a set of questions regarding the perceived stress level. Each of these questions were related to 5 possible weighted answers in order to compute the stress level perceived by the Employee.

The questionnaire was not designed to be exhaustive but it was used for the only scope to assess the subjective psychological status, in fact, the outcomes were merged to biological

parameters obtaining a more quantitative assessment.

***Medical visits**:* Every 6 months employees were asked to be visited by a specialized practitioner to assess the general health condition.

Sometimes, under special conditions, the employees could be visited more frequently.

***Wearable devices**:* Every employee has one wearable device, appropriately designed to sample data for different biological parameters such as heart rate or blood pressure.

***Set of parameters**:* The parameters sampled and analyzed to assess the stress levels are shown in the following tables.

At-Home Parameters	Insertion mode	Frequency
Heart rate (from wearable)	Automatic from file	Daily
Hours of physical activity (from wearable)	Automatic from file / Manually	Daily
Blood pressure (from wearable)	Automatic from file	Daily
Sleep quality index (from wearable)	Automatic from file	Weekly
Height, weight, BMI	Manually	Weekly
Food calories	Manually	Daily

At-Work Parameters	Insertion mode	Frequency
Heart rate (from clinical device)	Automatic from file	During visit
Blood pressure (sphygmomanometer)	Manually	During visit
Height, weight, BMI	Manually	During visit
Blood sample results	Manually	During visit
Diagnosis from polysomnography	Manually	On request
Perceived stress level (clinical scale)	Computed from inventory	During visit

2. METHODS

Modeling a platform is a complex process composed by specific phases, such as *Requirements Collection* and *Analysis*, *Conceptual Design*, *Logical Design* and *Database Implementation*.
We started to analyze, collect and represent all the requirements and specification of the person involved, such as employees, specialized practitioners and technical administrators. We did this by using *Use Case* and *Activity Diagrams*. Then, to approach the conceptual design, we create *Class* and *E-R Diagram*.
The logical design is the translation of the abstract representation into specifications that can be implemented through a *DBMS*, by using important instruments such as *Queries*.

2.1 Context Analysis

The main actors that we took into consideration are:

- *Employee:* Is the patient who needs to be monitored, so he collects the needed data with the help of the device assigned to him.
- *Specialized Practitioner*: is the specialist who treats the patient, constantly monitor his health condition and evaluates all the data and documents related to him to assess the stress level.
- *Technical Administrator:* Is the only one user capable of setting (and resetting) user data such as user-id and password.
- *GUI*: Is the Graphical User Interface, which links users and intelligence system.
- *System*: Is the software used by the users that includes database, GUI, database management system (DBMS), data analysis and data storage.

3. RESULTS

In this chapter will be explained the diagram we designed and used, their main functions and details. That will explain the principal static and dynamic aspects of our platform.

3.1 Use-Case Diagram

The main aim of the Use Case is to describe the static behavior of the system from the user point of view, and without specifying any details.
It describes the principal functions offered by the system and translates the requirements of the project in a graphical notation.

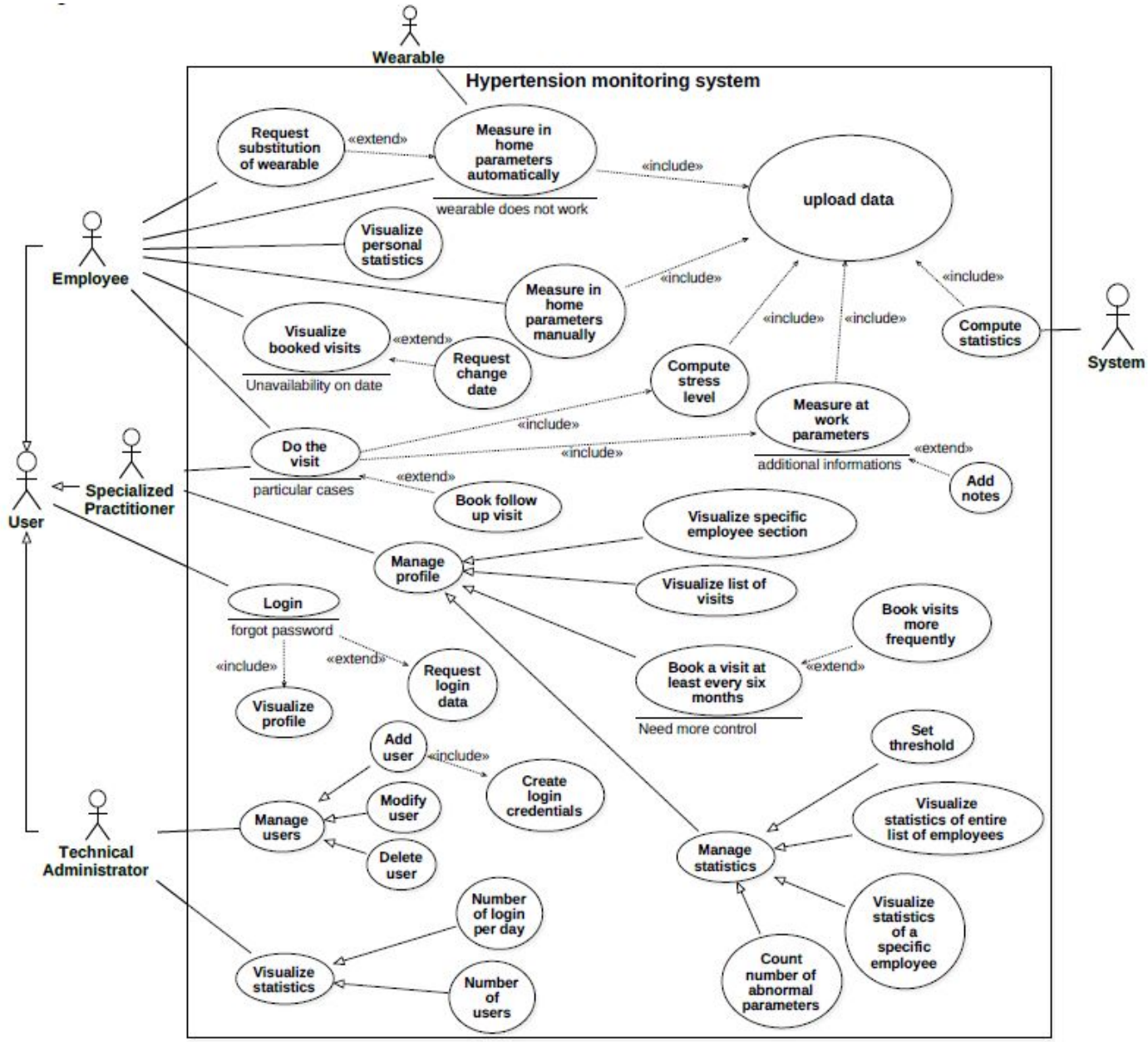


Figure 3.1– use case diagram

The main actors are the Employee, Technical Administrator, Specialized Practitioner, System and Wearable. In order to improve the quality of the overview we decided to make a single Use Case Diagram, and to make the description of each scenario: parameters measuring, Users' interaction with the System and the data. The following table represents the textual description:

TITLE: Interaction with the System of hypertension monitoring

PRE-CONDITION: The Employee is associated to a Specialized Practitioner and to Wearable Devices

POST-CONDITION: The level of hypertension of the Employee is monitored

MAIN SUCCESS SCENARIO (Measuring hypertension level)

1) The Employee does the login
2) The Employee measures in-home parameters manually
3) The Employee uploads the measured parameters in the system
4) The Wearable measures in-home parameters automatically
5) The Wearable uploads the measured parameters in the system
6) The Employee and Specialized Practitioner attend the visit
7) The Specialized Practitioner measures at-work parameters
8) The Specialized Practitioner uploads the measured parameters in the system
9) The Specialized Practitioner computes stress level
10) The System computes statistics

ALTERNATIVE SCENARIO

1.a Forgot password

1. Employee requests new password

4.a Wearable does not work

1. Employee requests the substitution of the Wearable

6.a Particular Cases

1. Specialized Practitioner books a follow up visit

7.a Additional information

1. Specialized Practitioner adds notes

MAIN SUCCESS SCENARIO (Employee's Interaction With The System And GUI)

1) The User does the login
2) The Employee visualizes the personal profile
3) The Employee visualizes booked visits
4) The Employee visualizes the personal statistics

ALTERNATIVE SCENARIO

1.a Forgot password

1. User requests a new password

3.a Employee is not available in a specific date for the visit

1. The Employee requests to change the date

6.a Need more control

1. The Specialized Practitioner books visits more frequently

MAIN SUCCESS SCENARIO (Specialized Practitioner's Interaction With The System And GUI)

1) The User does the login
2) The Specialized Practitioner visualizes the personal profile
3) The Specialized Practitioner visualizes the section of a specific Employee
4) The Specialized Practitioner visualizes the list of visits
5) The Specialized Practitioner books a visit at least every six months
6) The Specialized Practitioner sets threshold
7) The Specialized Practitioner visualizes statistics of entire list of Employees
8) The Specialized Practitioner visualizes statistics of a specific Employee
9) The Specialized Practitioner counts the number of abnormal parameters

ALTERNATIVE SCENARIO

5.a Need more control

1. The Specialized Practitioner books visits more frequently

MAIN SUCCESS SCENARIO (Technical Administrator's Interaction With The System And GUI)

1) The User does the login
2) The Technical Administrator visualizes the personal profile
3) The Technical Administrator adds user
4) The Technical Administrator creates login credentials
5) The Technical Administrator modifies user
6) The Technical Administrator deletes user
7) The Technical Administrator visualizes the number of login per day
8) The Technical Administrator visualizes number of user

3.2 Activity Diagram

The activity diagram describes the sequence of activities including conditional and parallel behaviors. This type of diagram is useful to describe in a more detailed and specific way the workflow of activities for specified actors. The most important and informative activities for the actors have been highlighted and multiple activity diagrams for each of those have been designed. These diagrams will be described in the following paragraphs.

3.2.1 Employee Activity Diagrams

Login process

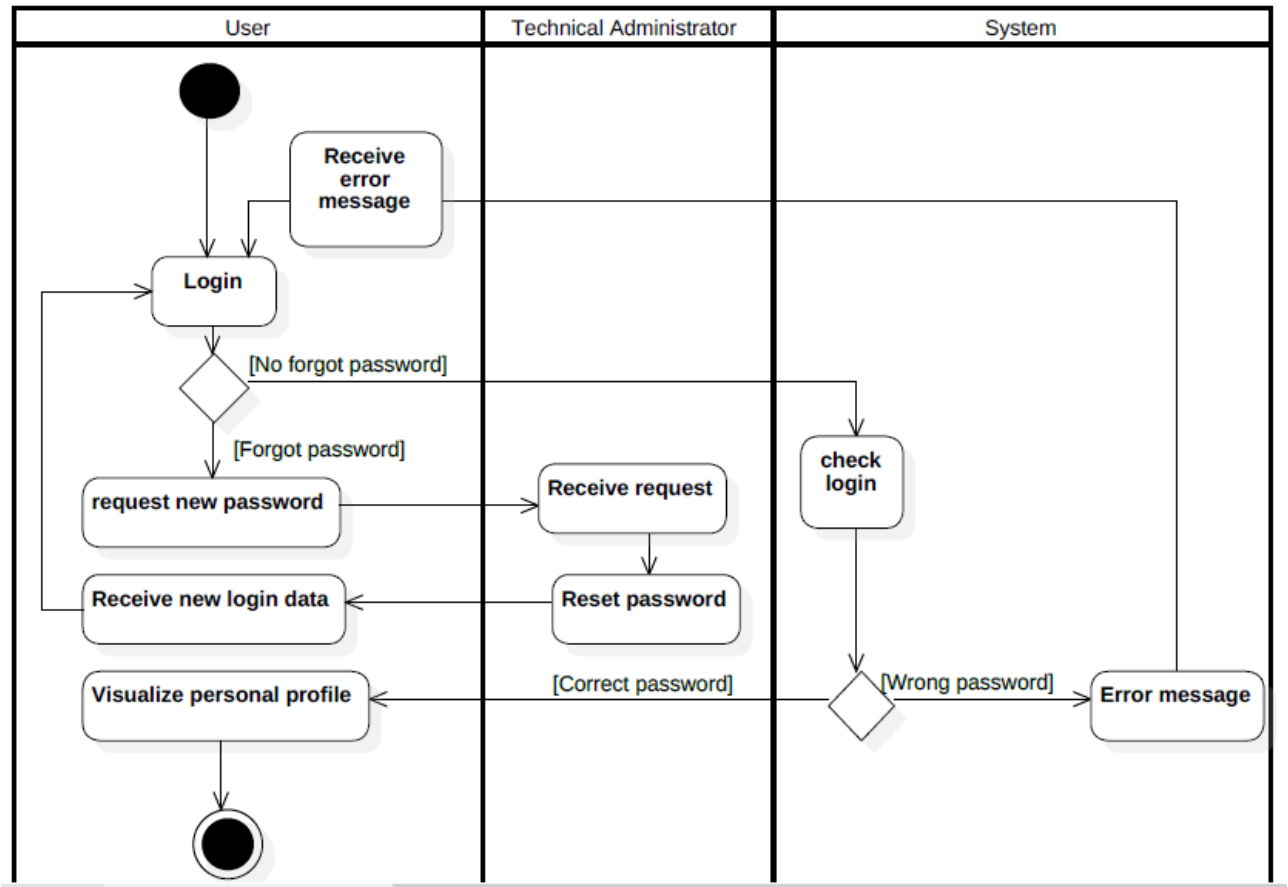


Figure 3.1 – First employee activity diagram

In this first diagram the login process have been modeled: it is the same for all the users.

During the login phase two different situations and respectively two developments can occur:

1) **Forgot password**: If the user doesn't remember the login data, he can request a new password to the Technical Administrator, who receives the request and proceeds to reset the password and to send the user the new credentials. This process is different for the technical administrator, because he can reset password by himself.
2) **Not forgot password:** If the user enters correct username and password, the system allows him to access to his personal profile.

Then the System checks the login credentials and if they are correct grants to the User the access to his personal profile. On the other hand, if it finds a wrong password, it sends to the User an error message, and then the User has to reinsert the credentials.

Visualize personal profile

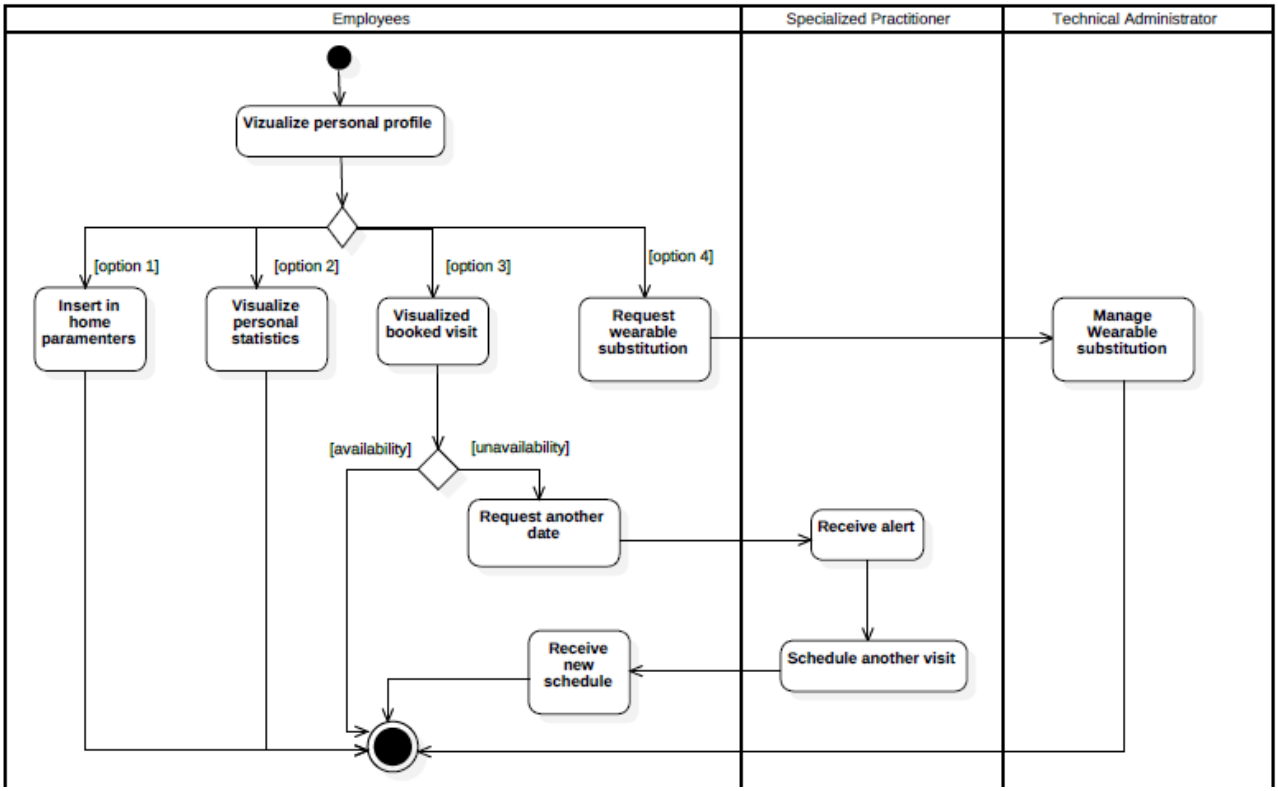


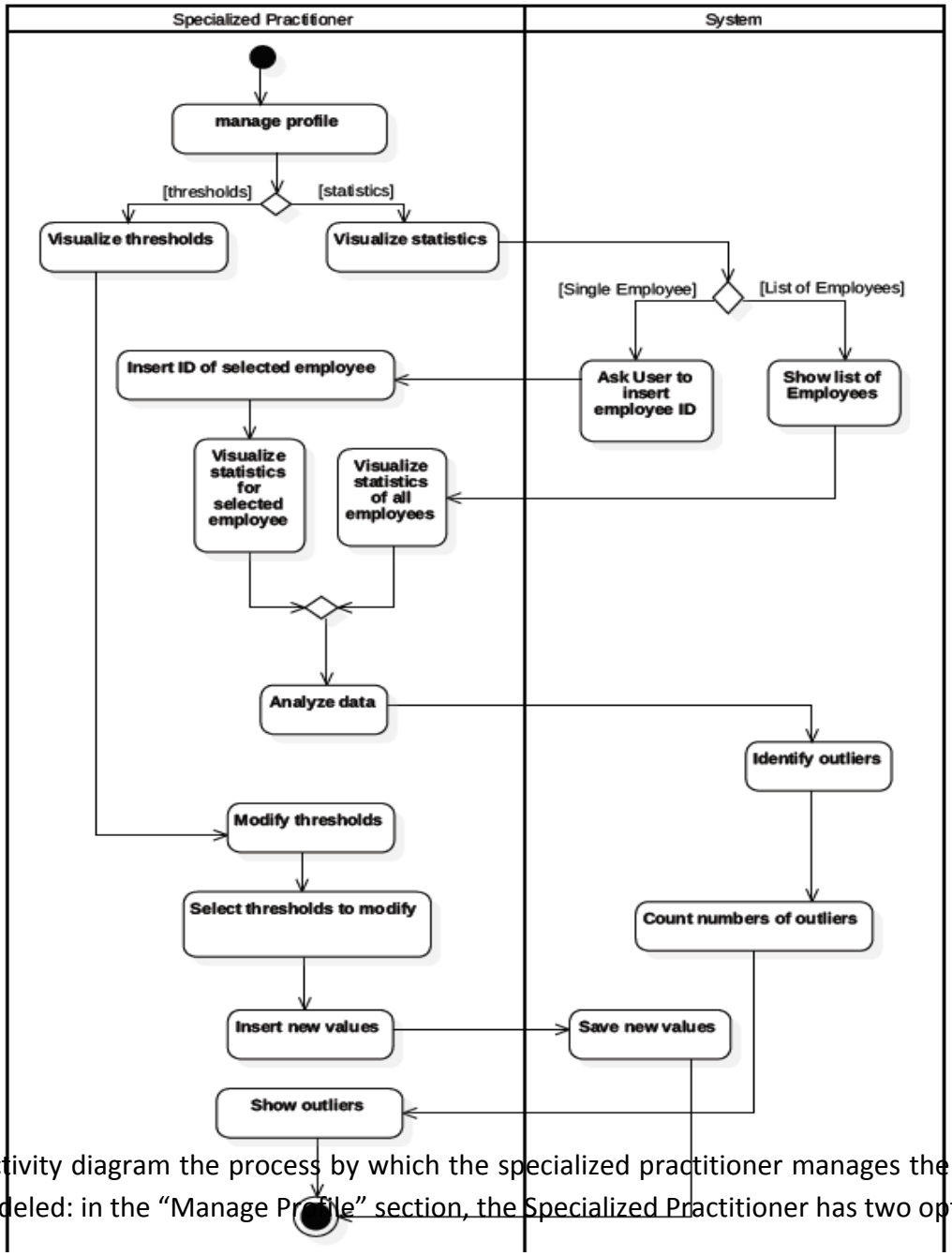
Figure 3.2 – Second employee activity diagram

This diagram represents what the user can do on his profile. There could be four different options:

1) **Insert in-home parameters:** the Employee can upload in-home parameters measured by Wearable (heart rate, blood pressure, sleep quality index, hours of physical activity) or manually (height, weight, BMI, food calories)
2) **Visualize personal statistics:** the Employee can visualize personal statistics about in-home and at-work parameters; statistics can show the temporal trend of a specific parameter or of a set of parameters, and also provide mean values and standard deviations.
3) **Visualize booked visits:** From here, the Employee can decide to request changing of the visit date, depending on his availability on the specific date. Moreover, the Specialized Practitioner receives an alert and is able to schedule a new visit and to communicate it to the Employee
4) **Request Wearable substitution:** The Employee can request a new Wearable to the Technical Administrator if it is not working.

3.2.2 Specialized Practitioner Activity Diagrams

Manage



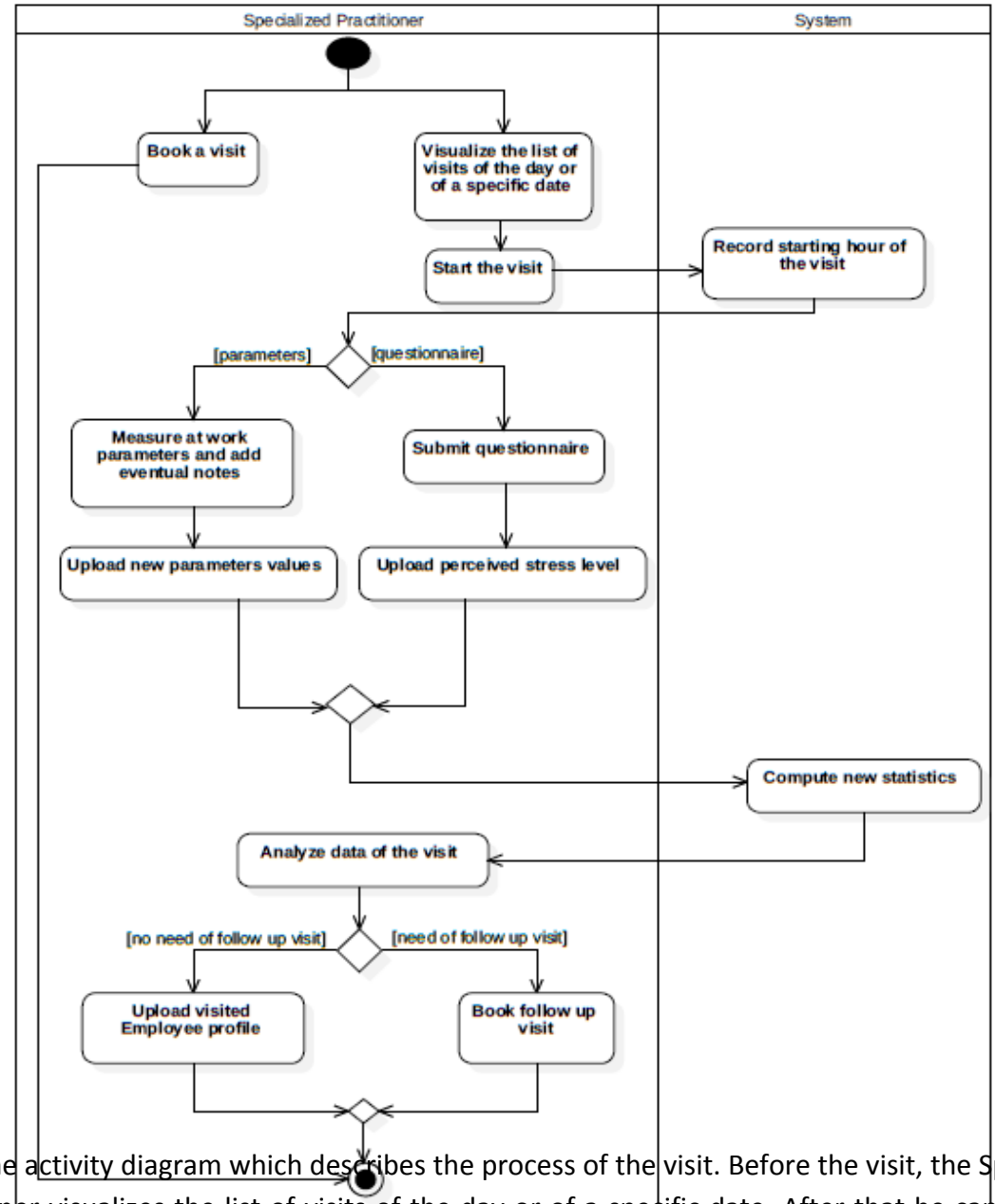
In this activity diagram the process by which the specialized practitioner manages the data have been modeled: in the "Manage Profile" section, the Specialized Practitioner has two options:

1) **Visualize thresholds:** In this section, the Specialized Practitioner can select thresholds to modify, entering their new values. These new values are then stored by the system.
2) **Visualize statistics:** In this section, the specialized practitioner is asked to select the view mode for the statistics:
    a) **single employee information:** The specialized practitioner is asked by the system to insert the specific employee ID. After that, the system provides the specialized practitioner a list of statistics for the selected employee.
    b) **set of Employees information:** Selecting this option, the system retrieves a list of employees with all their statistics.

After he visualized the statistics, the Specialized Practitioner can now analyze these data, with

the help of the System, that is asked to identify and count outliers, and to show them.

Visit



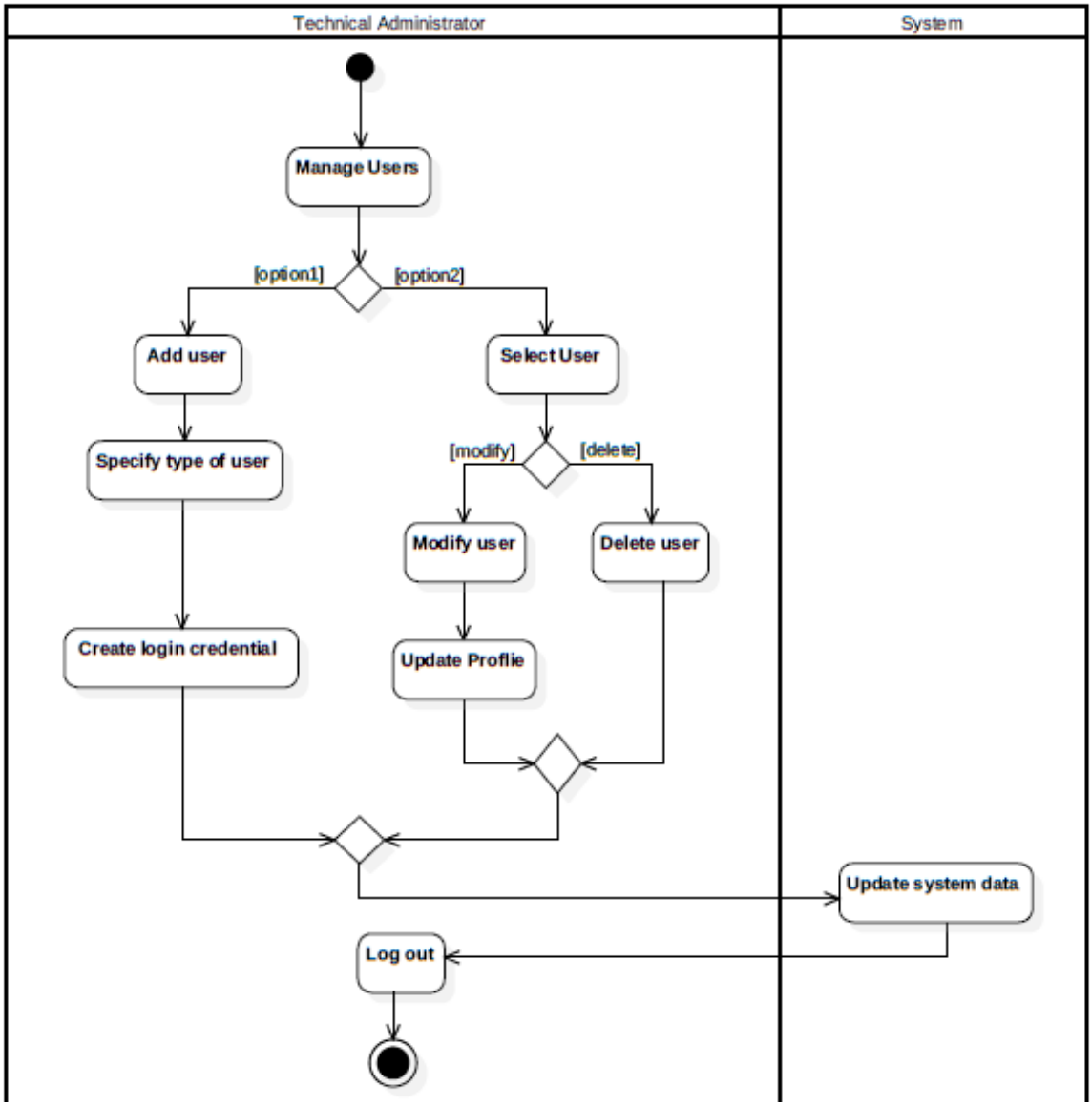
This is the activity diagram which describes the process of the visit. Before the visit, the Specialized Practitioner visualizes the list of visits of the day or of a specific date. After that he can start the visit, entering in the system the starting hour of the visit.

During the visit the Specialized Practitioner measures at-work parameters of the specific Employee and submits him the questionnaire. Then he uploads the new data, about the perceived stress level, provided by the questionnaire, and the parameters, to the system and it computes the new statistics related with the new data.

Then the Specialized Practitioner can analyze the data and, if is necessary, he can book a follow up visit before updating the Employee's profile.

3.2.3 Technical Administrator Activity Diagrams

Manage users



This diagram represents the process by which the Technical Administrator can manage the various users. The Technical Administrator has two options:

1) **Add user**: Technical Administrator has to specify the type of user he wants to add to the system and then he creates login credentials.
2) **Select user:** after selecting a user the Technical Administrator can delete it, or modify it, and then update the profile.

After each activity the Technical Administrator updates the system data.

3.3 Class Diagram

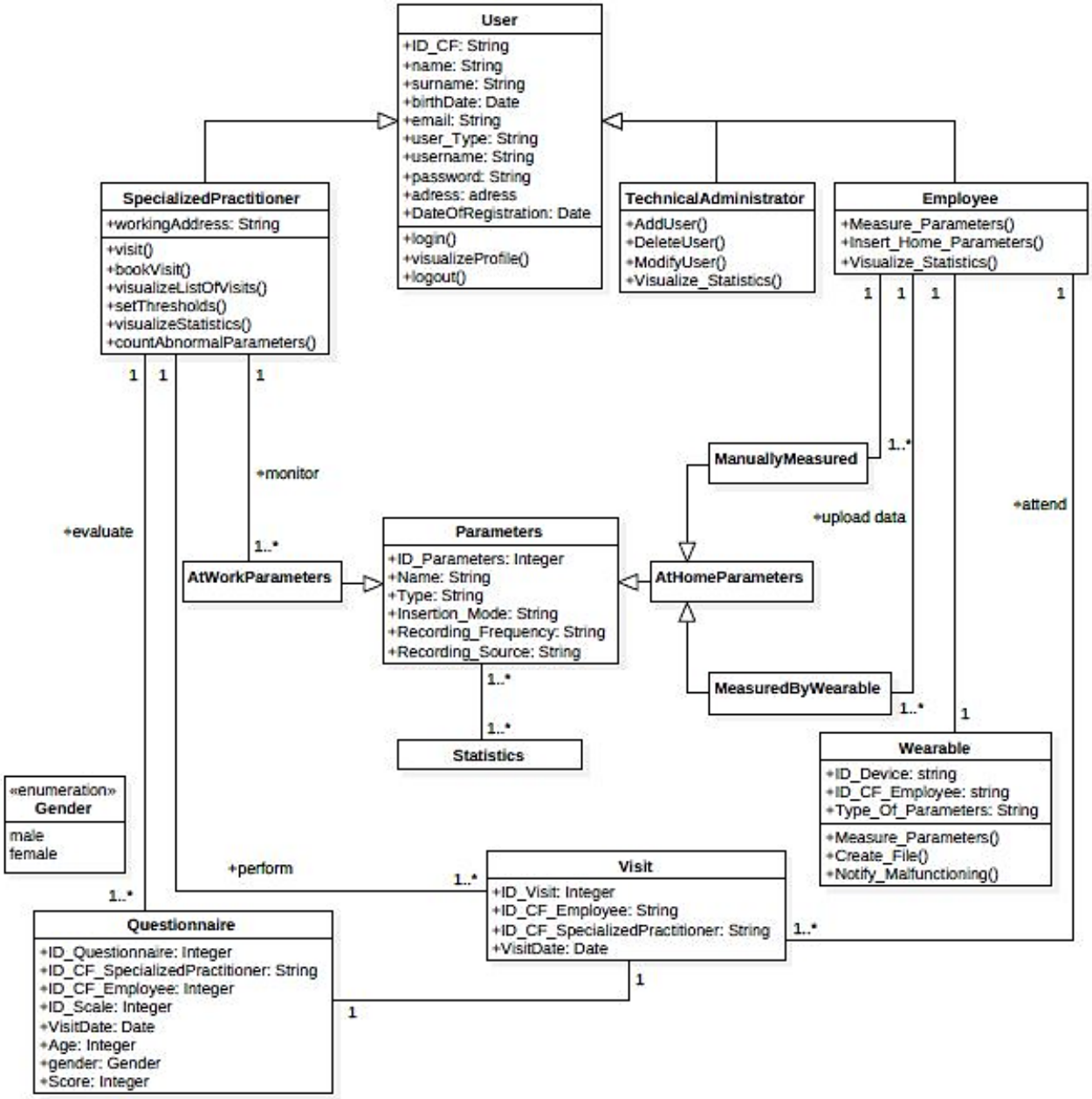


Figure 3.7– Class Diagram

The Class Diagram defines the classes and the relationships among them, and its aim is to describe the static behavior of the system. Each class describes a set of objects that have in common the same attributes, methods and behavior. The attributes are the properties of the class, while methods describe the implementation of procedures and functions.

In the Class Diagram has been created a User generalization of the classes Employee, Specialized Practitioner and Technical Administrator which acquire attributes and methods from the father class, adding some other new ones. All Users can do the login to the system, visualize their personal profile, and log out, and are characterized by a set of general attributes, like CF, that allows to identify univocally the user, User_Type that explicates which kind of User they are. Then each son class has specific attributes and methods related to the specific type of user; for example, Technical Administrator can add, modify and delete a user, Specialized Practitioner can book visits, visualize statistics, visualize the list of visits, while the Employee can measure at-

home parameters and insert them in the system.

Moreover, there is the Visit class that associates Employee to Specialized Practitioner, and has several attributes, ID_visit, Visit_date; it is also linked by one-to-one relationship with the Questionnaire class, that is connected with its compiler, the Specialized Practitioner.

The Questionnaire class, which contains all the answers to the questionnaires, represents the psychophysical state of the employees, that is associated with an important attribute called Score, that is the sum of the scores of each specific question, and is computed by the Specialized Practitioner once the visit is completed.

Employee is connected with a one-to-one relation with Wearable class, that is defined by its attributes, the ID_Device and the type of parameters that it can measure.

Employee can upload different data in the system: for first, they can upload a class of parameters that is called MeasuredByWearable, for second they can upload a class called ManuallyMeasured, relatively to the way of measuring.

These two types of parameters are connected by generalization to their father class, called AtHomeParameters.

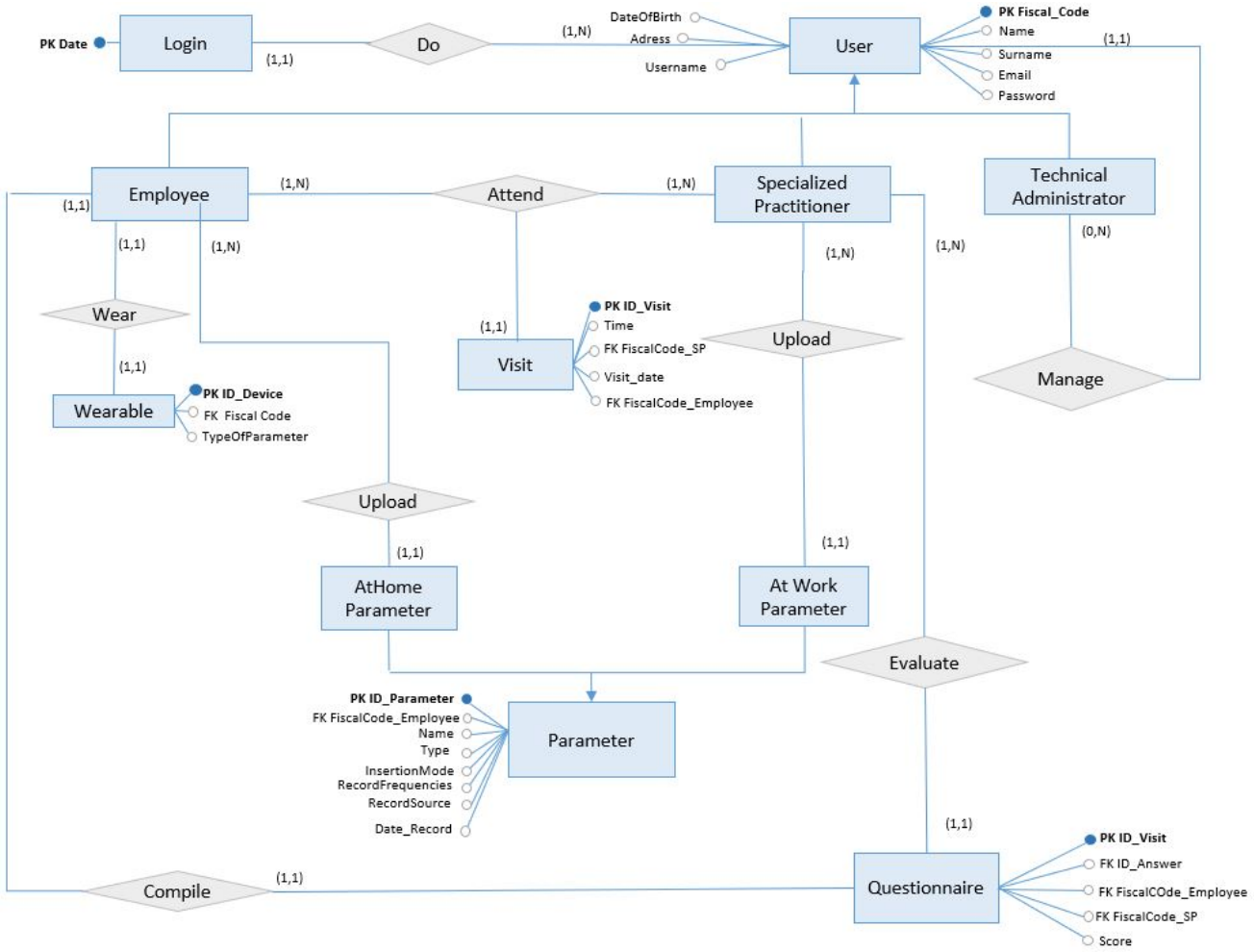
Specialized Practitioner class is connected to AtWorkParameters class because Specialized Practitioners can measure several kinds of parameter during the visit.

In order to distinguish the various sources of the data, AtWorkParameters and AtHomeParameters are connected by generalization to the father class Parameters; this class has several attributes, ID_Parameter, Name, Type, that provide an identification of each parameter, and other ones that give information about the recording frequency, source and insertion mode.
Another important class is Statistics, that is connected with Parameters.

3.4 E-R Diagram

The E-R Diagram explicates the relationships of entity sets stored in a database and illustrates the logical structure of the database.
This kind of representation is very high-level, so it describes the system in a very detailed way, but without reporting methods.

The composition of this diagram and the steps of its realization are analogous to the Class representation.

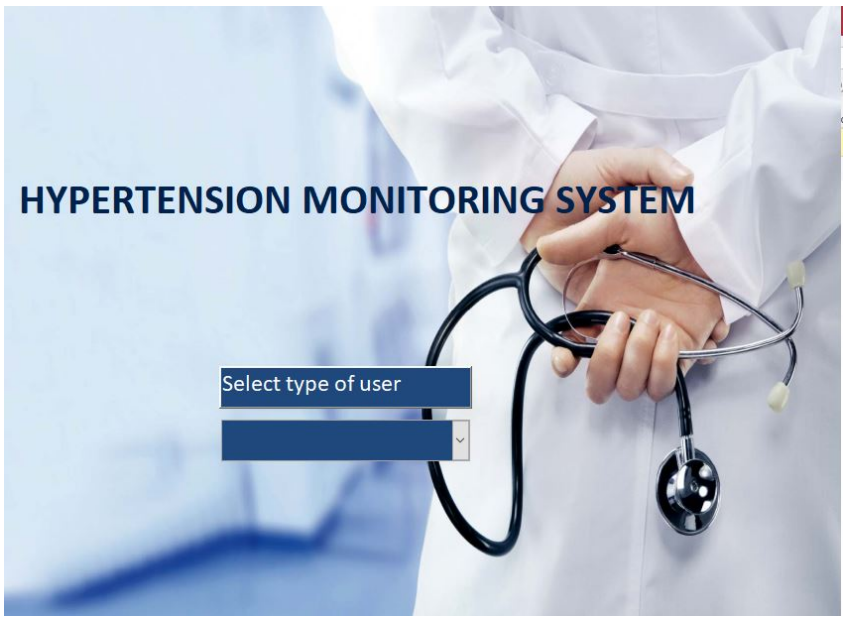


Part II: Implementation

4 IMPLEMENTATION OF THE DATABASE

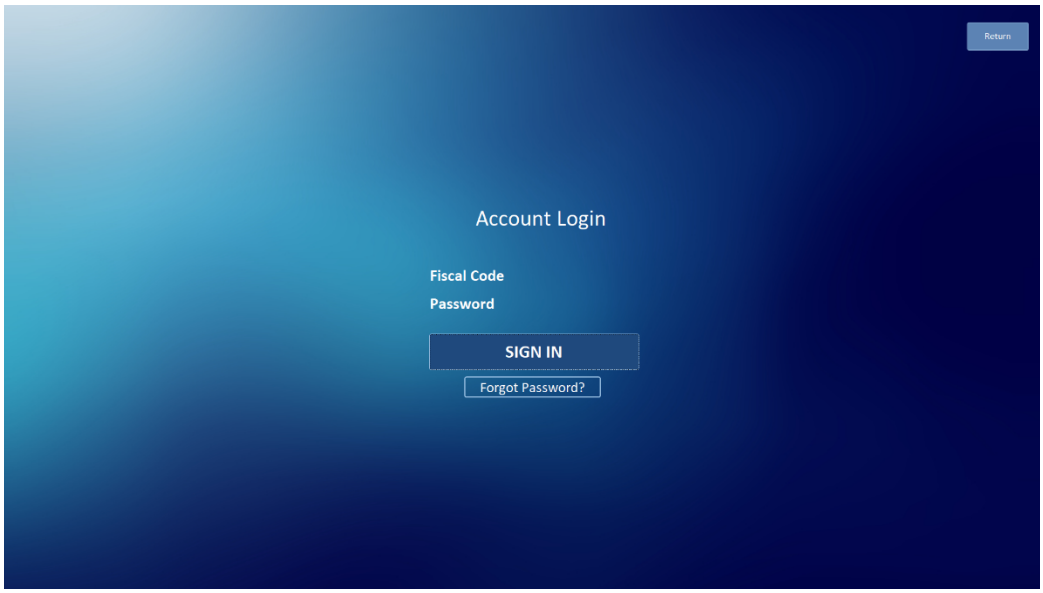
Starting from the logical model built, it was implemented the Database using Microsoft Access.
On the first hand, tables and relations between them have been created following the E-R Diagram

designed in the UML Modeling phase.
On the other hand, the Forms have been designed to facilitate the users' interaction with the system, trying to make it as smart, intuitive and detailed as possible.
In order to allow the access to different type of data, three forms have been created, one for each actor, Employee, Specialized Practitioner and Technical Administrator: first the generic user is asked to specify his User type, by selecting it in a defined form that drives him to the specific User form.



4.1 Login Form

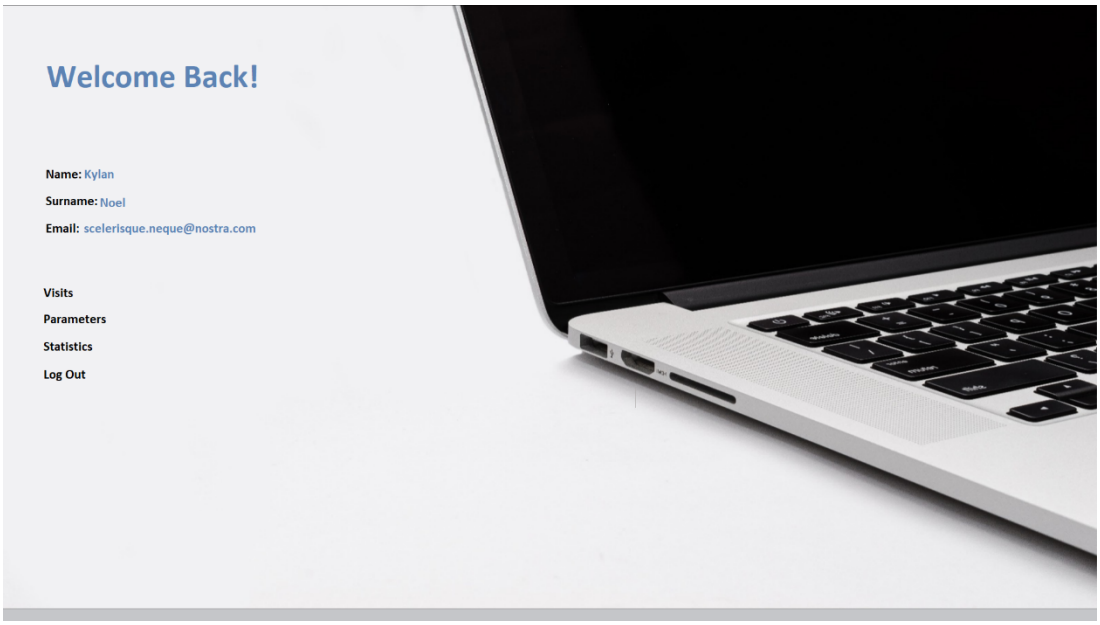
The specific User inserts his credentials, entering the Fiscal Code and the Password.
On the first hand, if the password or the Fiscal code are incorrect a message box appears telling "Username or Password is incorrect", and the user has to insert it again.
On the other hand, if the User does not remember the password, he can request a new one; if he clicks the related field an automatic message will be send to the Technical Administrator, and he remains in his requests until he decides to modify it.
If password is correct, the system automatically opens the Main Menu referred to the specific User.
The following image shows the login form:



Login Form

4.2 Employee Form

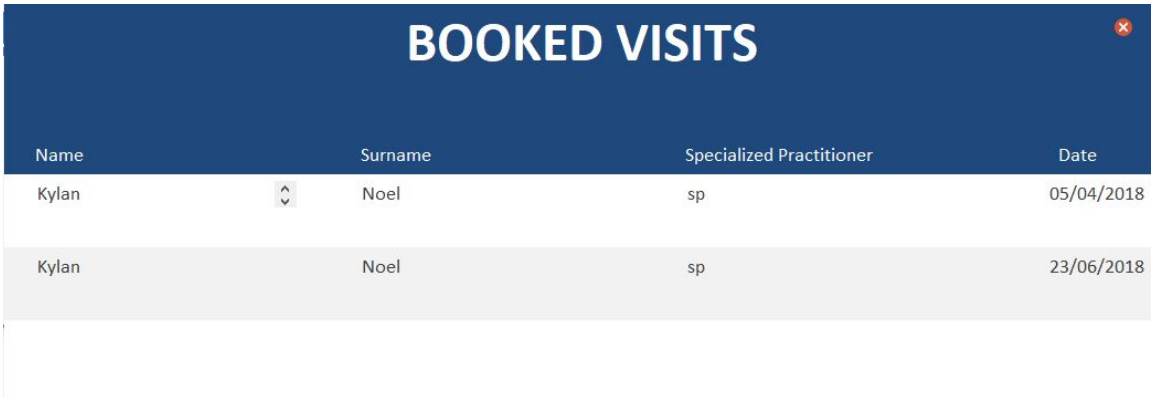
Employee's Form (1) has three main fields, Visits, Parameters, Statistics, followed by a section that allows to logout from the System.



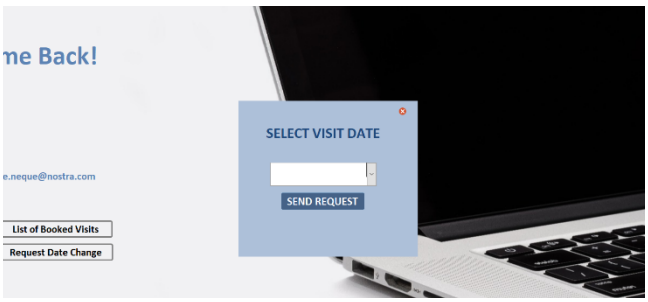
(1) Employee's Form

With the section ***Visits*** (2) the Employee can visualize the *List of Booked Visits (3)* with their dates and the name of the related Specialized Practitioner, or *Request Date Change*, relatively to a specific visit: in this case the Employee is asked to select the date of the visit (4) he wants to change and, then, an automatic message is sent to the Specialized Practitioner (5); the message appears in *Notifications* section of the Specialized Practitioner.

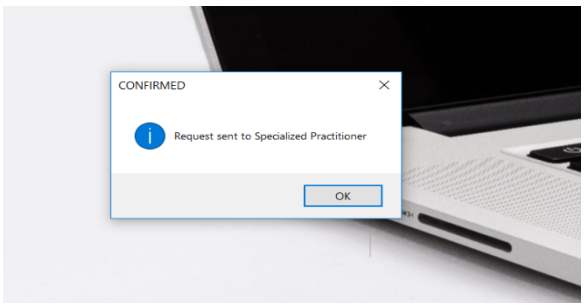
(2) Visits



(3) List of Booked Visits



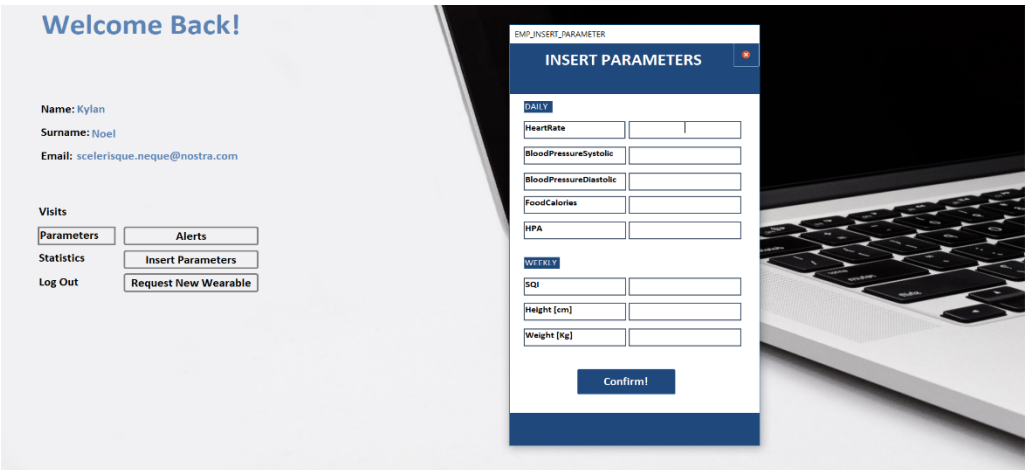
(4) Select the date of the visit to change



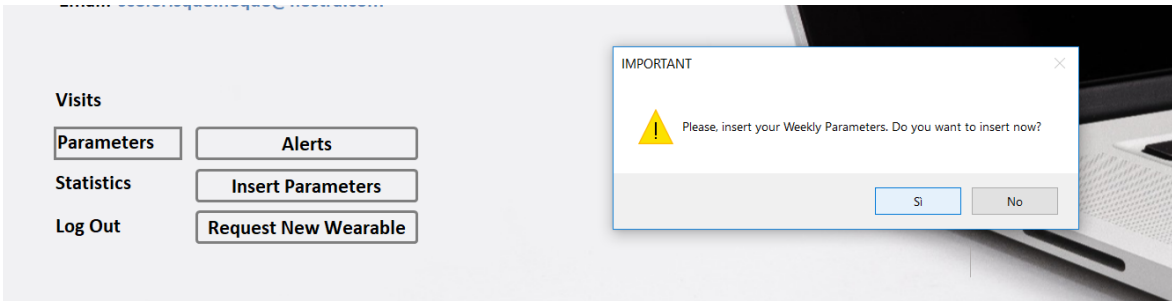
(5) Automatic message sent to the SP

From the section ***Parameters*** (6), the Employee can insert the daily and weekly Parameters (7).
If he clicks on the section *Alerts* and parameters have not been uploaded, he is reminded through a message box to insert his Daily and Weekly Parameters (8); is also given the possibility to decide to upload them in that moment, with an option that drives the Employee to the *Insert Parameters* section.

(6) Parameters

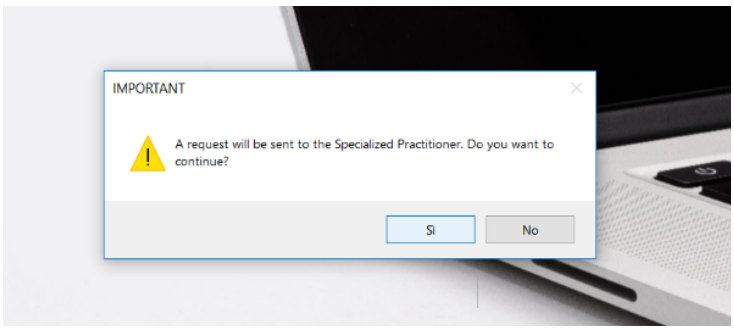


(7) Insert the daily and weekly Parameters



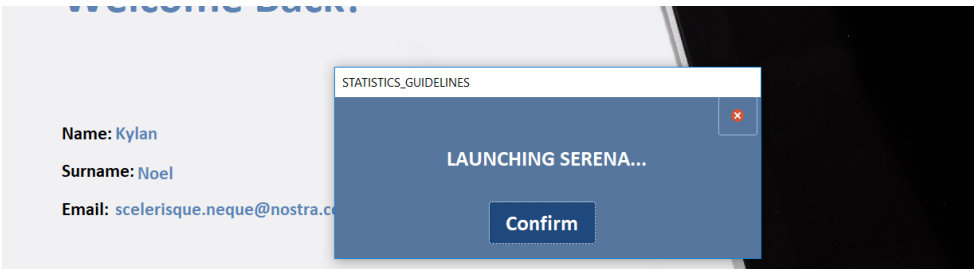
(8) Message box that remind to insert Parameters

The option *Request New Wearable (9)* sends an automatic notification to the Technical Administrator.



(9) Automatic request is sent to the Technical Administrator

From the field ***Statistics*** the Employee is redirected to the external application Serena, that will be explicated in detail in chapter 5.



4.3 Specialized Practitioner Form

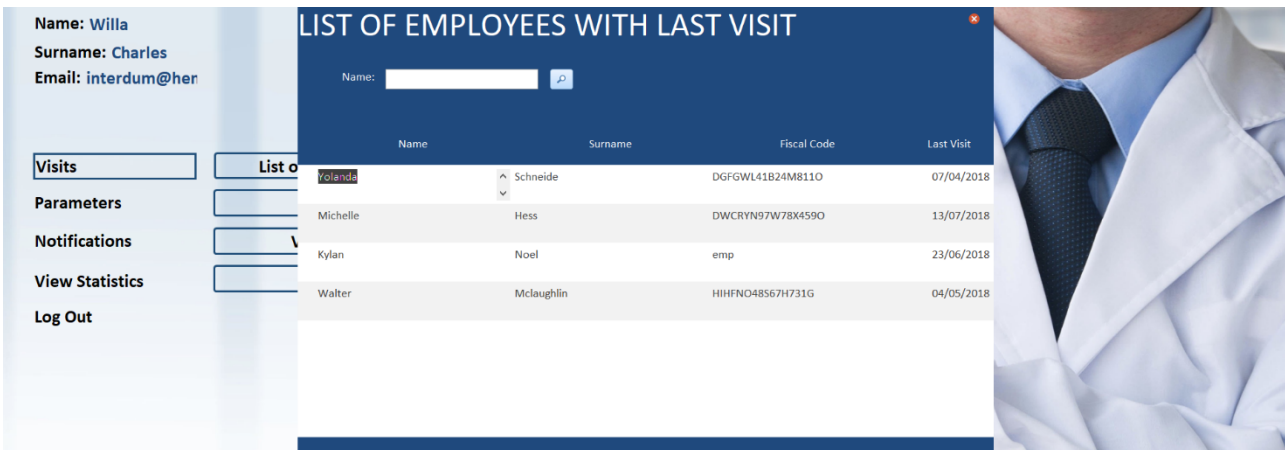
After the login the Specialized Practitioner is driven to his personal profile (10), composed by four main options, Visits, Parameters, Notifications, Statistics.

(10) Specialized Practitioner Form

Visits (11): in this section the Specialized Practitioner visualizes four options.

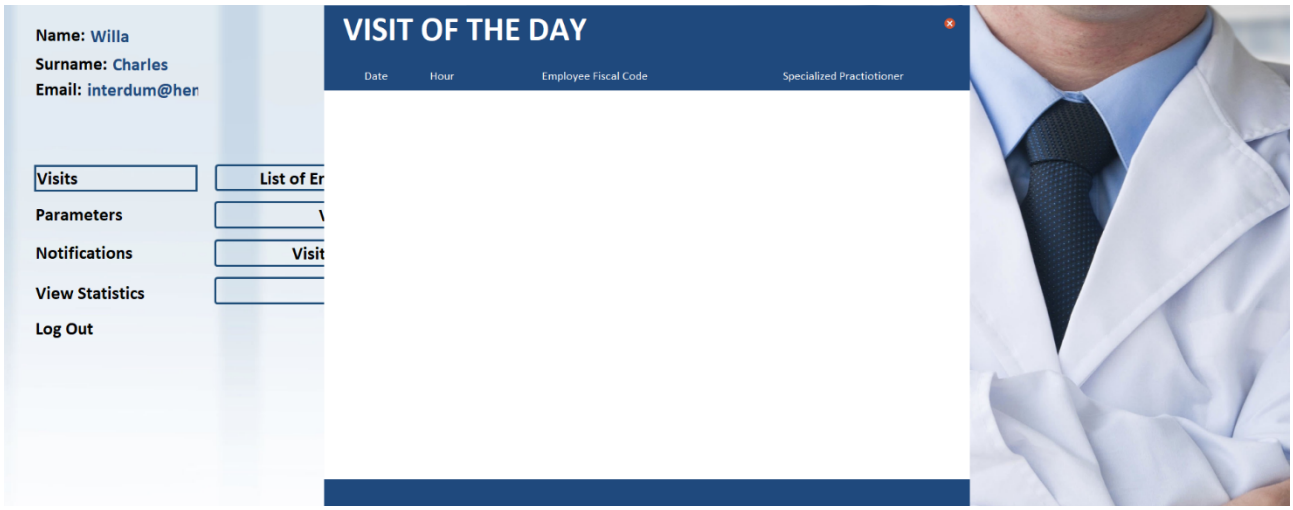
(11) Visits

First one is *List of Employees with Last Visit (12).*

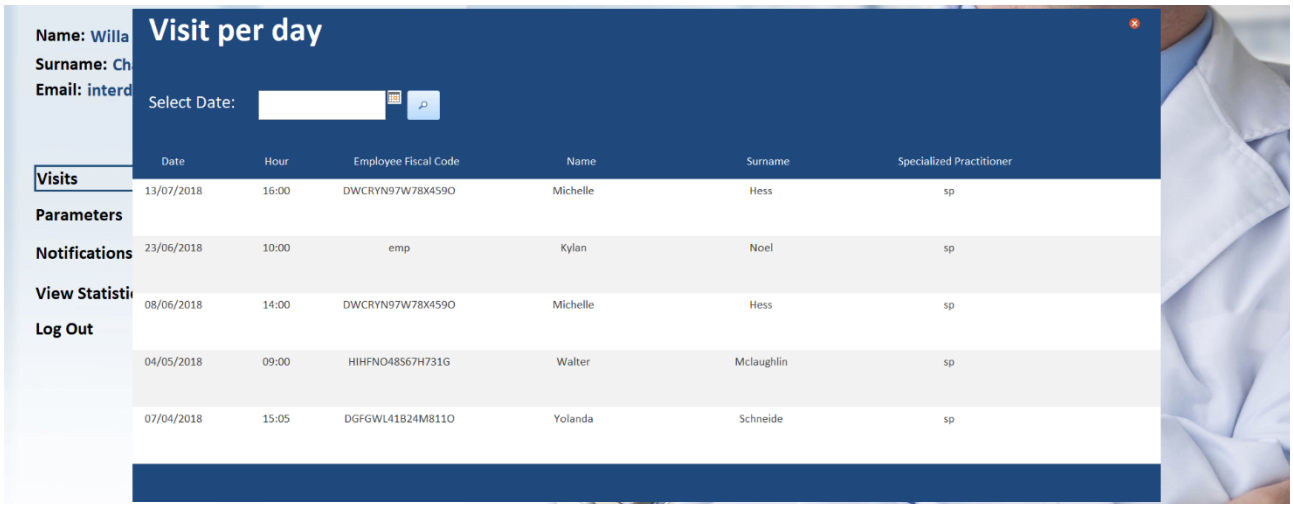


(12) *List of Employees with Last Visit*

Furthermore, the Specialized Practitioner can select *Visits of the day (13)*, obtaining automatically the list of visits of the current day, or *Visits of a Specific Date (14),* where is asked to insert the date of interest.



(13) Visits of the day



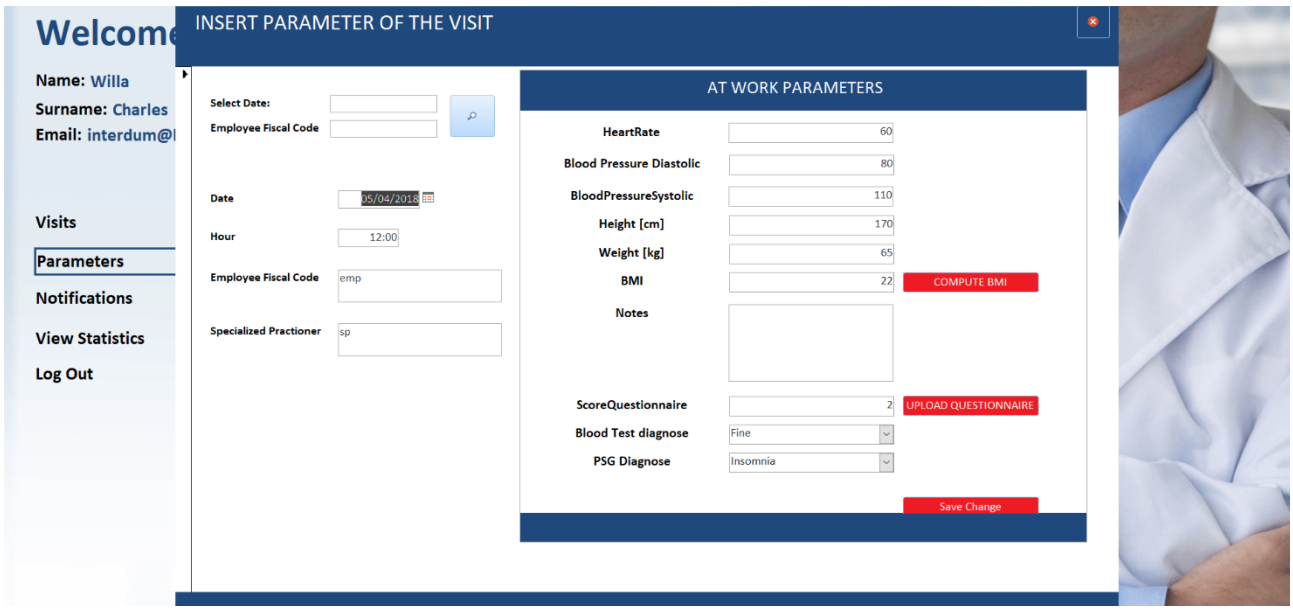
(14) Visits of a Specific Date

Moreover, the Specialized Practitioner can *Book a Visit (15).*

(15) Book a Visit

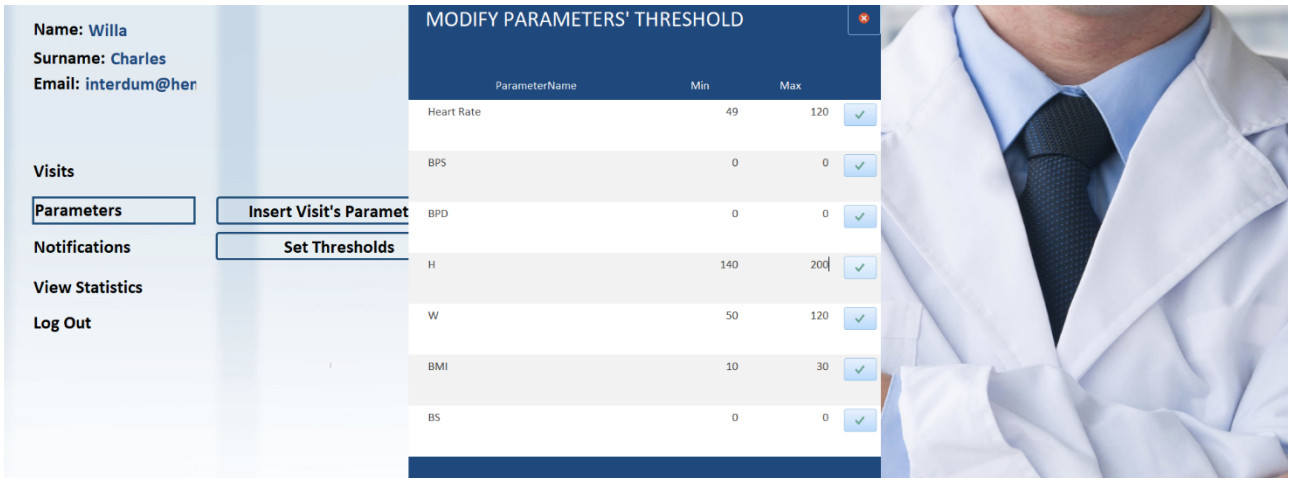
Parameters (16): from this section the Specialized Practitioner can first select *Insert Visit's Parameters (17)*, obtaining a form where is asked to fulfil the needed fields.

(16) Parameters



(17) Insert Visit's Parameters

Secondly, he has the possibility to select *Set the Thresholds (18)*; this form allows him to set or modify the threshold for each parameter:

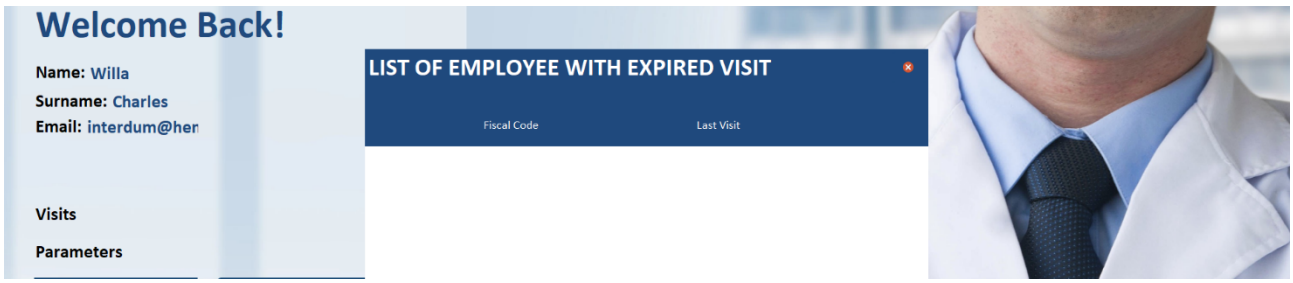


(18) Set the Thresholds

Notifications *(19):* in this section the Specialized Practitioner visualizes the *Requests of Visit Date Changings (20)*.

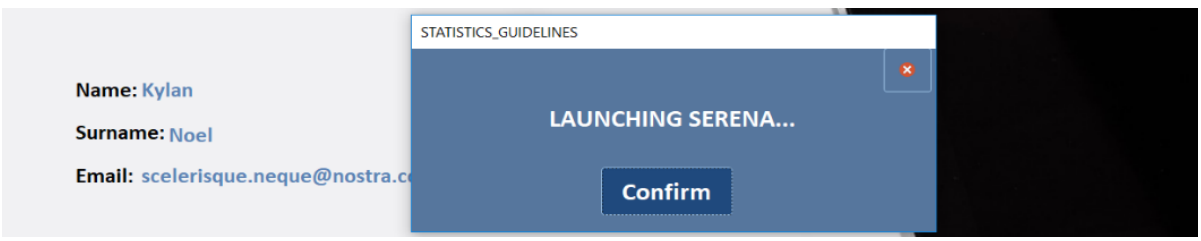
(19) Notifications

Finally, the Specialized Practitioner can visualize the *(20) List of Expired Visits (20);* the system computes the difference between the last visit (of each Employee) and the current day, and if the result is greater than six months, the system adds the Fiscal Code and the date of the last visit to the list.



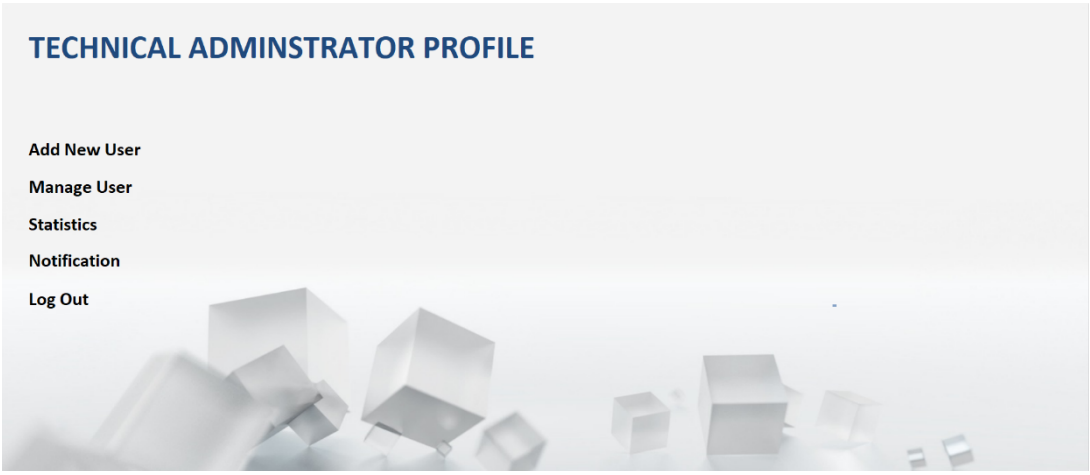
(20) List of Expired Visits

From the field ***Statistics*** the Specialized Practitioner is redirected to the external application Serena, that will be explicated in detail in chapter 5.



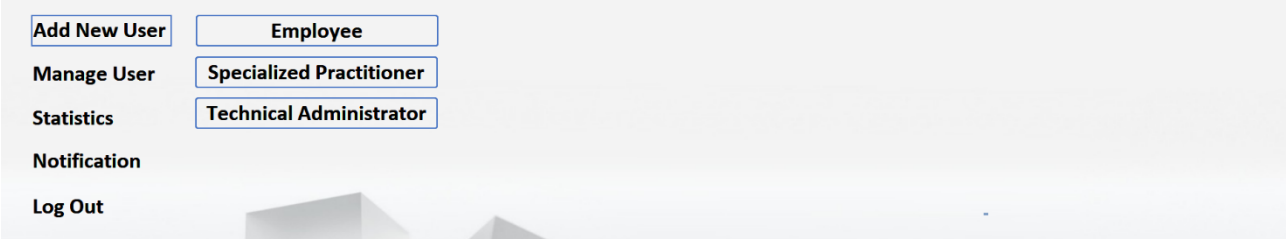
4.4 Technical Administrator Form

The Technical Administrator profile (21) has four main sections, Add New User, Manage User, Statistics, Notification.

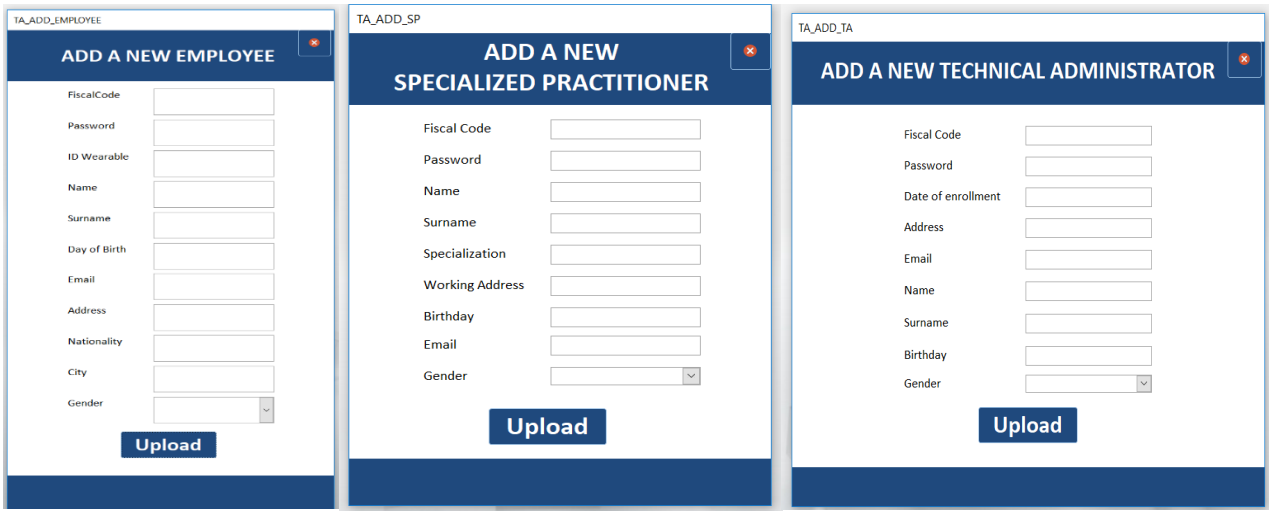


(21) Technical Administrator Form

Add New User *(22)*, allows him to add respectively an *Employee* (23), a *Specialized Practitioner* (24), a *Technical Administrator* (25), where he has to insert the requested information.



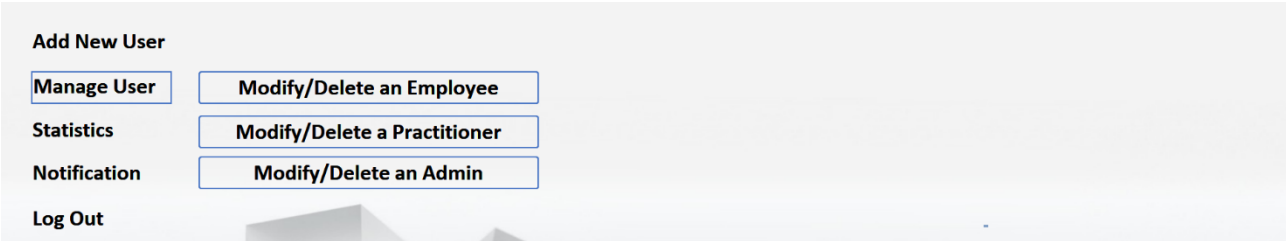
(22) Add new User



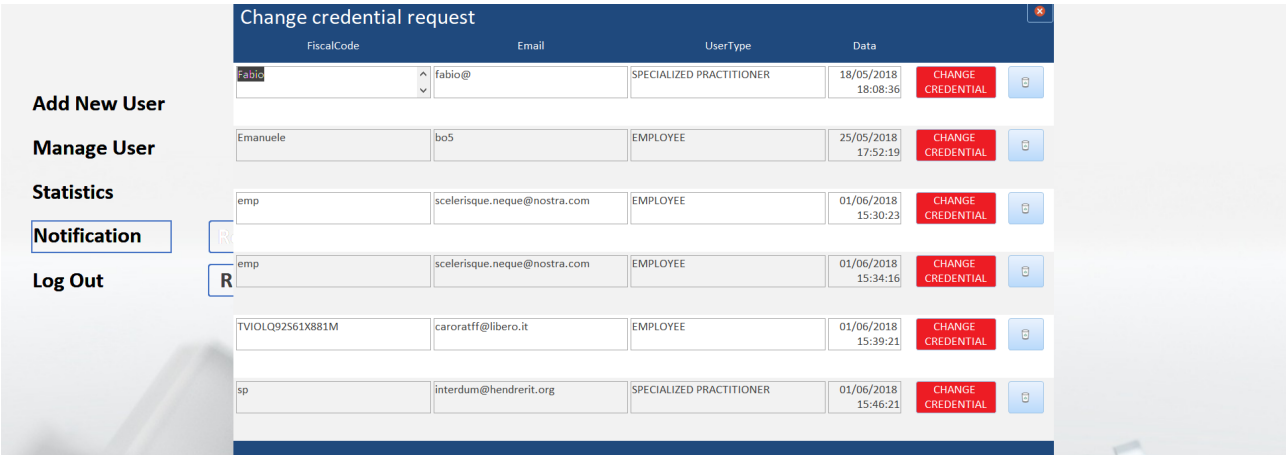
(23) New Employee *(24) New Specialized Practitioner* *(25) New Technical Administrator*

In the section ***Manage User*** *(26)* the Technical Administrator can modify the information about the

users or delete their profile. When he selects the type of user he wants to modify/delete, the list of the specific users appears; then he can modify some data and save or definitively delete the user (27).

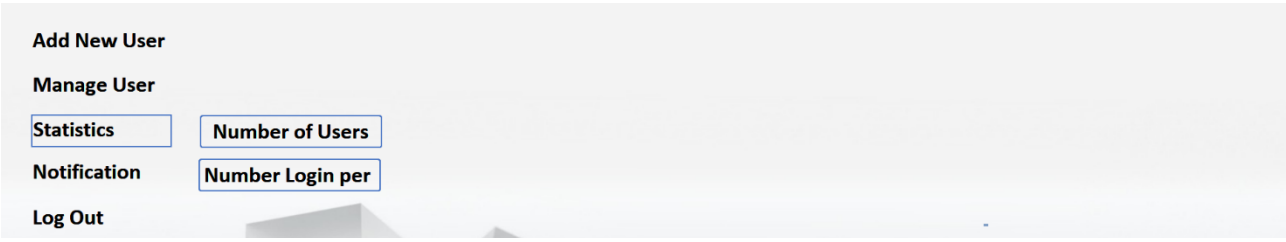


(26) Manage User

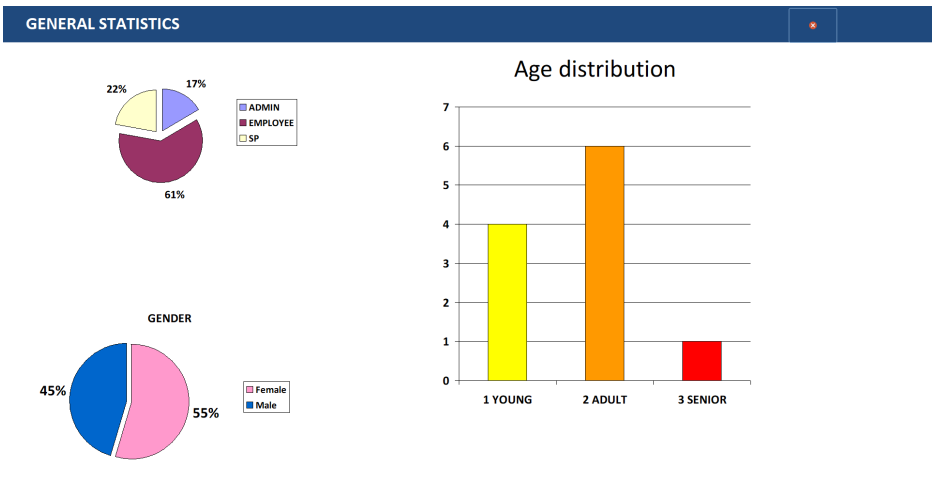


(27) Modify/delete User

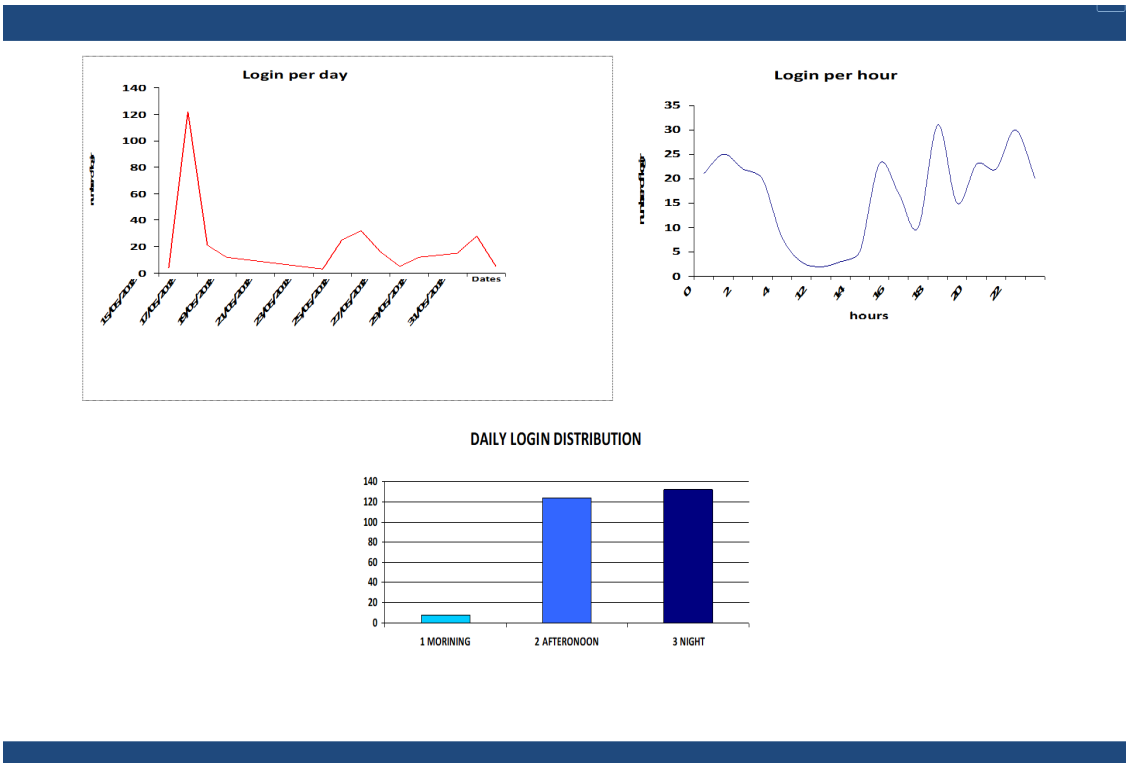
From ***Statistics*** field (28): the Technical Administrator can visualize statistics regarding the *Number of Users* (29), that shows the distribution of the genders, the age, and the type of user, and the *Number of Login per Day* (30), that shows the login per day, per hour and the daily login hour distribution.



(28) Statistics

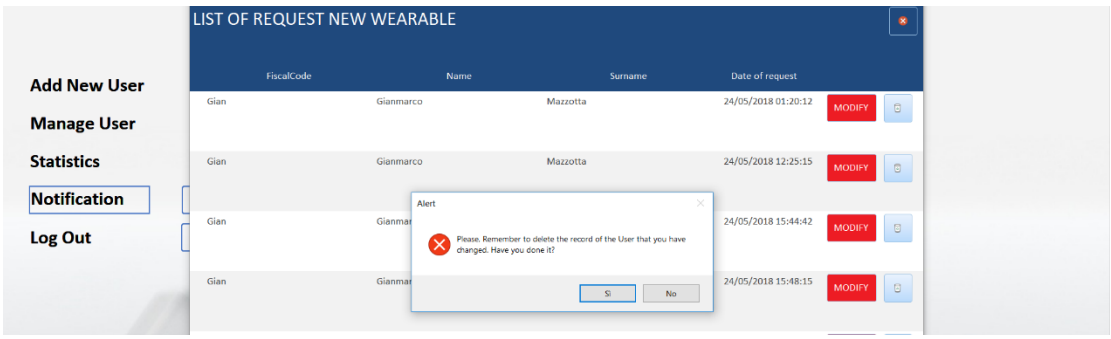


(29) Number of Users

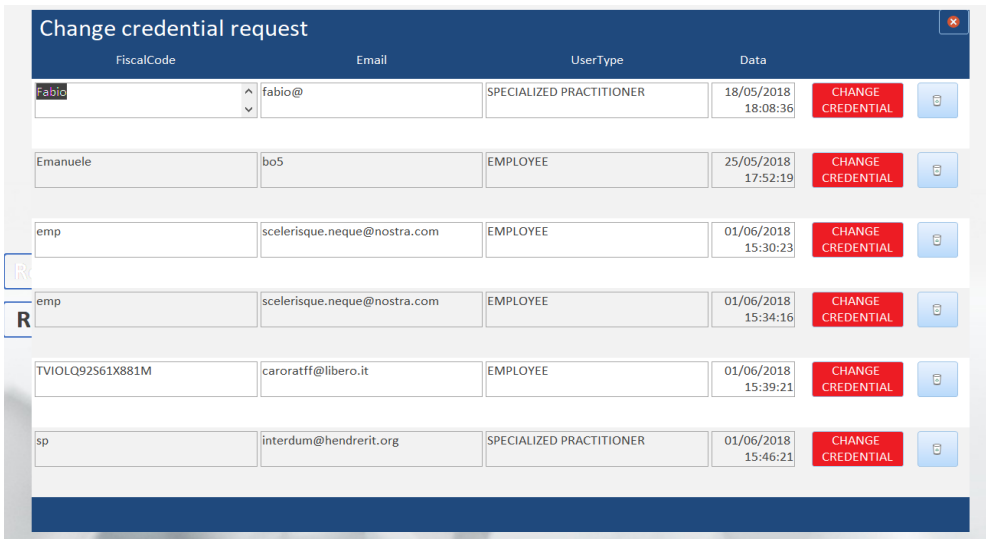


(30) Number of Login per Day

Notifications: in this section the Technical Administrator can visualize the *Request of Wearable Substitution (31)*, and the *Requests of New Credentials (32)*; the last one links him back to the list of Employees.



(31) Request of Wearable Substitution



(32) Requests of new Credentials

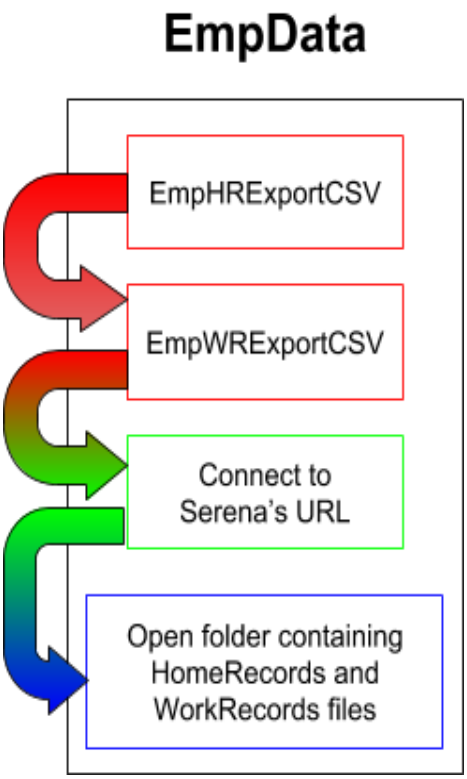
5 STATISTICS MANAGEMENT WITH R TOOL: *SERENA*

"Serena" is a Shiny dashboard application for the visualization of biological data taken from the Company's Database.
The aim of Serena is to provide a highly interactive and user-friendly interface to visualize different type of data giving the users the possibility to control various settings of visualization.
Serena has been developed entirely by our project group (more than 500 lines of code) using many different packages of the software "R". These packages are used within the app during the data preparation for the plotting of the temporal trends. The plots are obtained using mainly "ggplot2" package. The interactivity is given by "*Plotly*" which is a JavaScript package that makes easier for the user to visualize and interact with the plot.
To each user is provided a personal file from the company's Microsoft Access interface.
These files contain records of Employees' biological data in two different contexts:

1. Home records: biological data recorded while the Employee is at home during his every-day life.
2. Work records: biological data recorded while the Employee is undergoing a medical visit at the company.

The User can access to the dashboard using specific buttons on his/her Access profile.
The following description will explain the events for the Employee to visualize his/her data.

1. Employee clicks on "visualize statistics"
2. After clicking on it, one single macro (EmpData) is launched, providing to the user his/her file thanks to this structure:



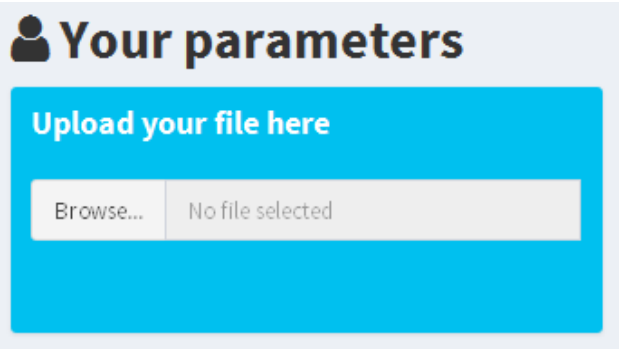
The macro is composed by 4 elements: 2 sub-macros and 2 system's commands.

The first sub-macro is called "EMPHRExportCSV". The first 3 letters stand for the type of user that the macro will gather the data for.
"HR" stand for "Home Records".
ExportCSV define the macro's aim.

Shortly, "EMPHRExportCSV" will firstly create the datatable of interest, through a creation query, and then it will export the newly created table in a textual file (with a .csv extension).
It's important to specify that the creation query is designed to extract all the biological records (of the actual logged employee) from the table "DB_HOME_RECORDS" and save them into a new table called "QEMPEXPORTHR" (Query Employee Export Home Records).
The second sub-macro is similar to the first one. The main difference is that it is designed to gather all the biological records (of the logged employee) from the table "DB_WORK_RECORDS".

After, the EmpData macro will simultaneously connect to Serena's URL and open the computer's folder containing the created csv files.

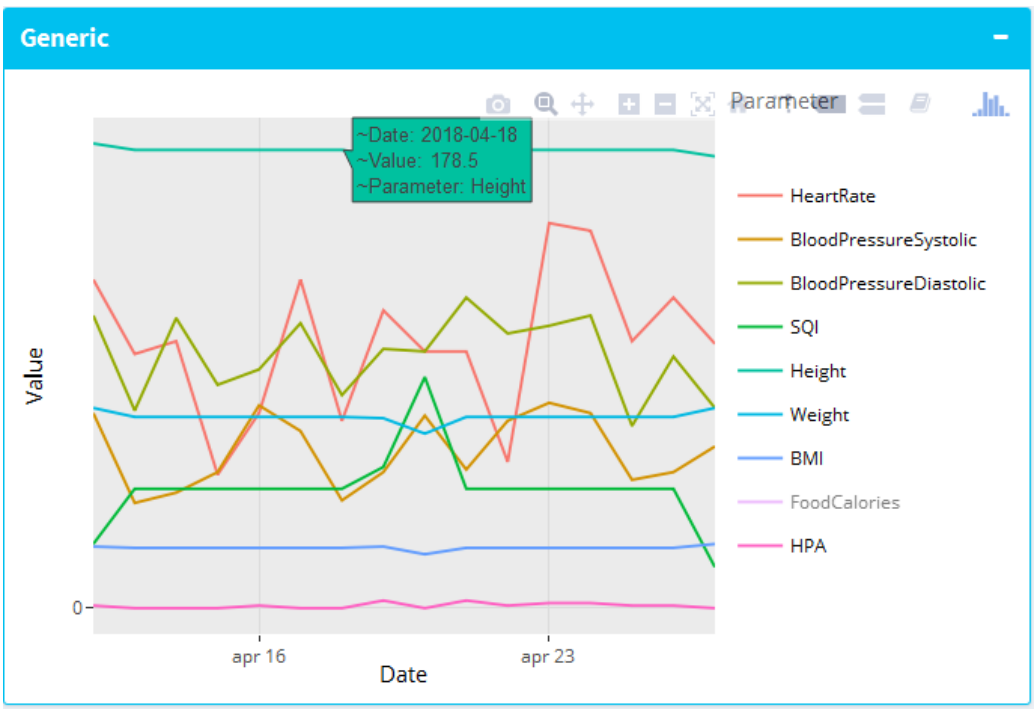
1) The Employee upload his Home Records or, alternatively, Work Records, on Serena through the specific button:



4) Serena will receive the file and will immediately plot all the employee's biological parameters using various R's packages.

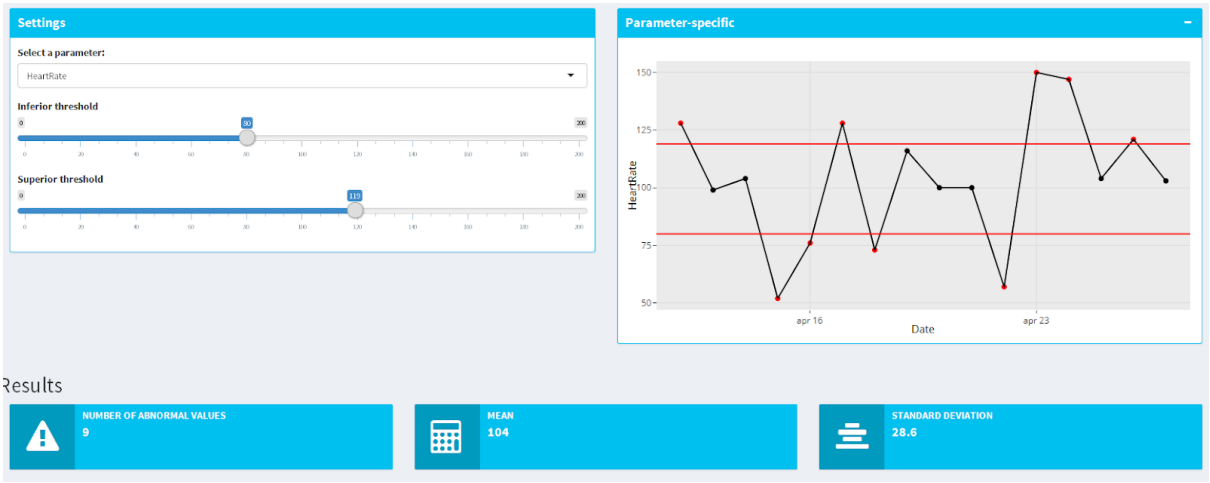
Serena's interface is subdivided in two sections:

1. Generic Overview: This section contains a single plot with all the biological parameters values in a given time frame. The user can freely interact with the plot thanks to the "plotly" R package, which is designed to enhanche the interactivity of a plot.



2. Parameter-specific: In this section, the user is able to select a specific parameter and visualize its temporal trend. Moreover, the user can set, through two sliders, the value of two hypotetycal thresholds (red lines in the plot), in order to have a visual feedback from Serena of

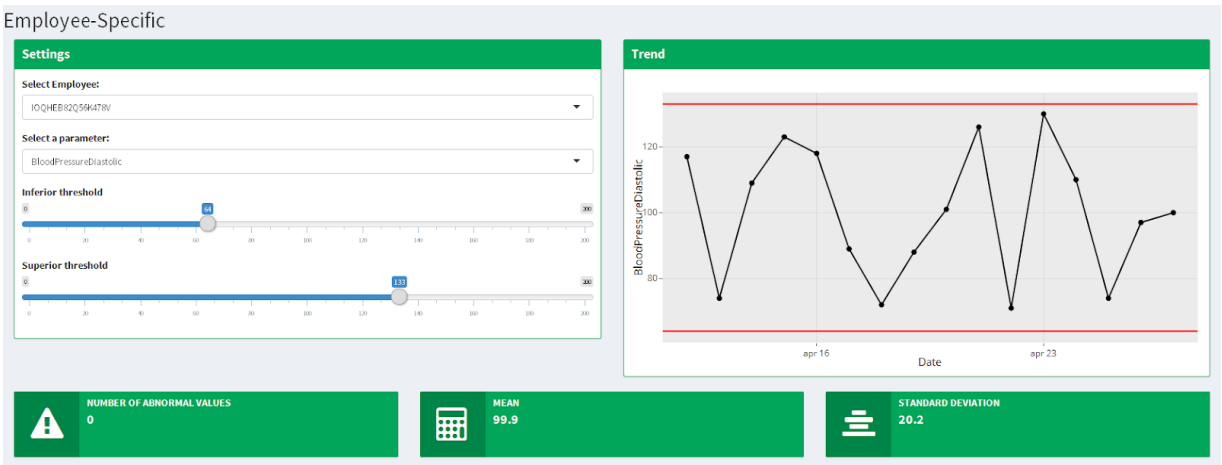
any abnormals values. Lastly, the user can visualize, at the bottom of the page, a set of base statistics regarding the selected parameter:



As already specified, all this procedure is related to an employee. However, the procedure for the Specialized Practitioner is almost the same. The only difference is that he can visualize, in the generic overview, the temporal trends of a (selectable) parameter for all the employees.

This is related to privacy issues: one employee is not allowed to have medical data about any other employees, while the specialized practitioner can.

Furthermore, in the “parameter-specific” section, the specialized practitioner is able to select not only the parameter, but also the specific employee, as highlighted in the following figure:



6. EXAMPLES OF QUERY

6.1 QSP_VisitSpecificDate

This query is used for the selection of the visits of a specific date, chosen by the Specialized Practitioner. It shows the list of the visits with name and surname of the Employees.

QSP_VisitSpecificDate

```
SELECT DB_VISIT.EMP_FiscalCode, DB_VISIT.SP_FiscalCode, DB_VISIT.Date, DB_VISIT.Hour, DB_EMPLOYEE.Name, DB_EMPLOYEE.Surname
FROM DB_SPECIALIZED_PRACTITIONER INNER JOIN (DB_EMPLOYEE INNER JOIN DB_VISIT ON DB_EMPLOYEE.FiscalCode = DB_VISIT.EMP_FiscalCode) ON
DB_SPECIALIZED_PRACTITIONER.FiscalCode = DB_VISIT.SP_FiscalCode
ORDER BY DB_VISIT.Date DESC;
```

6.2 QSP_VisitToday

This query shows the list of the visits of the current day.

QSP_VisitToday

```
SELECT DB_VISIT.EMP_FiscalCode, DB_VISIT.SP_FiscalCode, DB_VISIT.Hour, DB_VISIT.Date
FROM DB_SPECIALIZED_PRACTITIONER INNER JOIN DB_VISIT ON DB_SPECIALIZED_PRACTITIONER.FiscalCode = DB_VISIT.SP_FiscalCode
WHERE (((DB_VISIT.SP_FiscalCode)=[Maschere]![SP_Profile]![Testo62]) AND ((DB_VISIT.Date)=Date()));
```

6.3 QSP_ListEMPLOYEEandLASTVISIT

This query shows the list of the Employees among the last visit.

QSP_ListEMPLOYEEandLASTVISIT

```
SELECT DB_VISIT.EMP_FiscalCode AS [FISCAL CODE], Max(DB_VISIT.Date) AS [Last Visit], DB_EMPLOYEE.Name, DB_EMPLOYEE.Surname
FROM DB_VISIT INNER JOIN DB_EMPLOYEE ON DB_VISIT.EMP_FiscalCode = DB_EMPLOYEE.FiscalCode
GROUP BY DB_VISIT.EMP_FiscalCode, DB_EMPLOYEE.Name, DB_EMPLOYEE.Surname;
```

6.4 Q_CHECK6MONTH

The query shows the dates of the booked visit of a specific employee in order to retrieve the expired visits thanks a “visual basic” code that gives the differences between last visit and the current day. If the difference is greater than six month then the sp will be contacted.

Q_CHECK6MONTH

```
SELECT Max(DB_VISIT.Date) AS ULTIMA
FROM DB_VISIT
GROUP BY DB_VISIT.EMP_FiscalCode
HAVING (((DB_VISIT.EMP_FiscalCode)=[Maschere]![Employee_Profile]![Testo62]));
```

6.5 QEMP_VisualizeVisit

QEMP_VisualizeVisit

```
SELECT DB_EMPLOYEE.FiscalCode, DB_EMPLOYEE.Name, DB_EMPLOYEE.Surname, DB_SPECIALIZED_PRACTITIONER.FiscalCode, DB_VISIT.Date
FROM (DB_VISIT INNER JOIN DB_EMPLOYEE ON DB_VISIT.EMP_FiscalCode = DB_EMPLOYEE.FiscalCode) INNER JOIN DB_SPECIALIZED_PRACTITIONER ON
DB_VISIT.SP_FiscalCode = DB_SPECIALIZED_PRACTITIONER.FiscalCode
WHERE (((DB_EMPLOYEE.FiscalCode)=[Maschere]![Employee_Profile]![Testo62]));
```

6.6 Q_COUNT_USER

This query counts the Users and group by them by their user type.

Q_COUNT_USER

```
SELECT 'EMPLOYEE' as USER, COUNT(*) as TOT
from DB_EMPLOYEE
UNION
SELECT 'SP', COUNT(*)
from DB_SPECIALIZED_PRACTITIONER
UNION SELECT 'ADMIN', COUNT(*)
from DB_TECHNICAL_ADMINISTRATOR;
```

6.7 Fasce_età

This query groups the Employees by their age (Young, Adult, Senior).

Fasce_età

```
SELECT '1 YOUNG' as AGE, COUNT (*) as TOT
FROM DB_EMPLOYEE
Where DB_EMPLOYEE.[Day of Birth] between #10/05/1990# and #12/12/2015#

UNION

SELECT '2 ADULT', COUNT (*) as TOT
FROM DB_EMPLOYEE
Where DB_EMPLOYEE.[Day of Birth] between #10/05/1970# and #11/05/1990#

UNION SELECT '3 SENIOR', COUNT (*) as TOT
FROM DB_EMPLOYEE
Where DB_EMPLOYEE.[Day of Birth] between #10/05/1955#  and #09/05/1970#;
```

6.8 Q_COUNT_GENDER_EMP

This query counts the number of male and female Employees and groups them by gender; it is useful for the statistics available for the Technical Administrator.

Q_COUNT_GENDER_EMP

```
SELECT Gender.Gender, Count(*) AS [Count]
FROM Gender INNER JOIN DB_EMPLOYEE ON Gender.ID = DB_EMPLOYEE.Gender
GROUP BY Gender.Gender, DB_EMPLOYEE.Gender;
```

6.9 TA_HOUR

This query groups the login by their time (Morning, Afternoon, Night).

TA_HOUR

```
Select '1 MORINING', COUNT (*) as TOT
FROM LOGIN
Where Format$(Login.Data,"hh:mm:ss") between #06:00:01# and #14:00:00#

UNION

Select '2 AFTERONOON', COUNT (*) as TOT
FROM LOGIN
Where Format$(Login.Data,"hh:mm:ss") between #14:00:01# and #21:00:00#

UNION Select '3 NIGHT', COUNT (*) as TOT
FROM LOGIN
Where Format$(Login.Data,"hh:mm:ss") between #21:00:00# and #06:00:00#;
```

7. CONCLUSIONS

Overall, the Project Team achieves its goal to build a smart, user-friendly, easy and graphically intuitive Database, composed by many forms that make it reliable and give the possibility to insert new data every time is needed, making it incremental.

The Database can be used by all the three types of users, relating to their access level, that allows them to interact with the system and visualize data guarantying the respect of the privacy. In fact, health data are sensitive data and it is important to guarantee the security and the privacy; for these reasons the Database was created in order to make impossible for an Employee to visualize data of another Employee, while a Technical Administrator does not have the possibility to access to health data. Only the Specialized Practitioners can manage these types of data.

After to an in-depth analysis of the requirements, was started the important phase of UML Modeling, that was useful to visualize and understand the system and the process behind, to facilitate the specification of the system, to facilitate the construction and the documentation of the system.

The diagrams designed with StarUML have been the base of the system, and each one has made the team aware of specifics features: Use Case Diagram was useful to understand and visualize the requirements, the functions and the actors of the system, Activity Diagram to understand the workflow and to describe the algorithms and interactions between actors, while Class Diagram and ER Diagram have been the basis of the Database.

Then, the Database was implemented by using Microsoft Access, with the help of Visual Basic.

The Database have been also enriched by the possibility to send and receive alerts, messages and requests, that facilitate the interaction between actors and between actors and the system.

Finally, Serena was implemented with an R Tool that make possible a better interaction with the data and will also give the possibility to manage a large amount of data in long term, thanks to all the advanced functionalities that R could offer.

The Group believes that the database, as well as the Access interface, should be converted in an online accessible database using, for example, MySQL and PHP.

The company could so have a dedicated website comprehending all the features included in the offline system. The website could also be translated in a mobile application for smartphone, enhancing its accessibility and consultability.