Web Ratio Project Sanitary System

Advanced Software Engineering

Bianca Francesco
Prandini Manuel
Trombetta Valeria

Supervised by Prof. Bottoni Paolo

Dipartimento di Informatica

M.Sc. in Computer Science

Universitá degli studi di Roma, "La Sapienza"

2018/2019

SAPIENZA
UNIVERSITÀ DI ROMA

WWW.UNIROMA1.IT

First edition, January 8, 2020

Contents

1

Introduction

We want to design *SanitarySystemProject*, (a web-app/information system) for management and organization of a complex sanitary system composed by hospital and pharmacies. Various figures such as *patients*, *doctors* and *pharmacists* will be able to interact within the system. *Administrators*, on the other hand, will have access to database changes and to viewing some statistics.
To use the services offered by SanitarySystemProject, registration and approval by administrators is mandatory. During the registration phase, the user will be asked for the following information: name, surname, fiscal code, e-mail, date of birth, telephone, password plus a series of additional information depending on the type of user to register. In particular, patients will have to provide personal data like weight and height, doctors will have to provide data on the medical specialization and the examination room where they work (in the case of *'pronto soccorso'* medical specialization, the information on the first aid room where they work) while the pharmacists will have to provide in additional, only the data of the pharmacy where they work.

Each user can be registered in the system under multiple roles (a patient can also be a doctor or a pharmacist and vice versa).

Once the patient is approved by the system administrator who will check the validity of the data entered, he can book visits to any hospital in the system, he can have access to his medical record, check the calendar of visits booked and search for the availability of drugs in all system pharmacies.

The doctors approved by the administrator, on the other hand, will be able to record the data of the medical visits of the patients carried out during the day (having the possibility to prescribe drugs, hospitalizations, medical tests or interventions) and he will be able to access the patient's medical record. If the doctor is an emergency room's one, he will have access to a different area of the system where he will view the emergency room queue and will be able to visit patients sorted by code (red, yellow and green) and arrival time.

In the system the approved pharmcists will be able to consult the resources of the drugs present in their pharmacy, update the catalog of drugs and will be able to interact with a customer (a patient enrolled in the system) by starting a drug sales session. Prescription drugs can be sold by the pharmacist only if the patient has a valid prescription.

The system administrator will take care of approving each individual user when the check about the data entered during the registration phase, will be completed. He will send a confirmation email in case of successful operation. Within his area he will be able to modify hospital structures, to insert pharmacies and drugs into the system, to insert, delete and modify the F.A.Q and to have access to the statistics of any hospital or pharmacy which are present in the system.

The following sections will explain in more detail the requirements, the various use cases which have been implemented, the constraints of these latter, the domain model and how the relational model was structured.

2

Software Requirements Specification (SRS)

2.1 | Actors

1. *Visitor*: A person who has not yet performed a registration for the first time or who is not yet logged.
2. *User*: A registered user.
    2.1. *Patient*: Registered user who can book visits and search for drugs to buy in a specific area.
    2.2. *Doctor*: Registered user who manages his patients. It can be:
        2.2.1. *Specialist doctor*: doctor with a medical specialization other than first aid (e.g. cardiologist, neurologist etc).
        2.2.2. *Emergency doctor*: doctor with a first aid medical specialization. He works in the emergency room of a hospital.
    2.3. *Pharmacist*: Registered user who has access to the pharmaceutical side of their pharmacy within the system.
3. *Administrator*: He is the one who has access to the administrative side of the system. He is aimed to approve user registrations and he can modify the data in the database.

2.2 | Requirements

1. The system must provide access to the following areas:

1.1. Access to the *Login-Registration area*.

1.2. Access to *Patient area*.

1.3. Access to *Specialist-doctor area*.

1.4. Access to *Emergency-doctor area*.

1.5. Access to *Pharmacist area*.

1.6. Access to *Administrator area*.

2. The Login-Registration area, which all actors can access 2.1, must provide the following services:

2.1. User registration:

2.1.1. For each user (whether a patient, doctor or pharmacist) it is necessary to know: *fiscal code, name, surname, gender, date of birth, country of birth, place of birth, country of residence, place of residence, residential address, email, telephone, password*.

2.1.2. Concerning patients it is also required to know (in addition to the data indicated in 2.1.1.): *height, weight*.

2.1.3. About doctors it is also essential to know (in addition to the data indicated in 2.1.1.): *hospital country, hospital city, hospital, medical specialization and work room within the hospital*.

2.1.4. On pharmacists it is also indispensable to know (in addition to the data indicated in 2.1.1.): *pharmacy country, pharmacy city and pharmacy where he works*.

2.2. User login: the system must allow the insertion of the following credentials in order to perform the log in operation:

2.2.1. fiscal code

2.2.2. password

2.3. Password recovery: the system must allow the insertion of the following credentials to recover the password:

2.3.1. fiscal code

2.3.2. email

2.3.3. control code (captcha)

3. The Patient area, can be accesed by actors of type 2.1., and it must provide the following services:

3.1. Visit:

3.1.1. Viewing visit calendar

3.1.1.1. View visit details

3.1.1.2. Cancel visit

3.1.2. Viewing Hospitalization calendar:

3.1.2.1. View hospitalization details

3.1.2.2. View medical check-ups

3.1.2.3. View clinical surgery

3.1.3. Booking of visits

3.2. Search for the drug:

3.2.1. View drug descriptions

3.3. Viewing medical record

3.3.1. View past visits:

3.3.1.1. View medical prescriptions for visits

3.3.2. View past hospitalizations:

3.3.2.1. View medical check-ups made in hospitalization

3.3.3. View past emergency room services

3.3.4. View past clinical surgery

3.4. Profile management:

3.4.1. View personal information

3.4.2. Change personal information

3.5. View F.A.Q.

3.6. Logout

4. The specialist doctor area must provide the following services:

4.1. Access patient data:

4.1.1. See patient's medical record:

4.1.1.1. View past patient visits:

4.1.1.1.1. View medical prescriptions of patient visits

4.1.1.2. View past patient hospitalizations:

4.1.1.2.1. View medical check-ups performed in the patient's hospitalization

4.1.1.3. View past patient emergency room services

4.1.1.4. View past patient clinical surgery

4.1.2. Record medical check-ups results

4.1.3. Record visit:

4.1.3.1. Entering the title and medical descriptions of the visit

4.1.3.2. Entering medical prescriptions

4.1.3.3. Hospitalization prescription

4.1.3.3.1. Prescriptions medical check-ups to be carried out in hospitalization

4.1.3.4. Clinical surgery prescription

4.2. Registration in the clinical surgery room

4.3. View personal information

4.4. View F.A.Q.

4.5. Logout

5. The emergency room doctor area must provide the following services:

5.1. Patient acceptance in the emergency room

5.1.1. View the emergency room queue

5.1.2. Call the next patient

5.1.3. Discharge the patient

5.2. See patient's medical record:

5.2.1. View past patient visits:

5.2.1.1. View medical prescriptions of patient visits

5.2.2. View past patient hospitalizations:

5.2.2.1. View medical check-ups performed in the patient's hospitalization

5.2.3. View past patient emergency room services

5.2.4. View past patient clinical surgery

5.3. View personal information

5.4. View F.A.Q.

5.5. Logout

6. The Pharmacist area must provide the following services:

6.1. Interact with the customer:

6.1.1. View active medical prescriptions

6.1.2. Drug sale:

6.1.2.1. Selling drugs without prescriptions

6.1.2.2. Selling drugs with prescriptions

6.1.2.3. Shopping cart summary

6.1.2.4. Delete from the shopping cart

6.2. Pharmacy management:

6.2.1. Drugs inventory

6.2.2. Search for drugs

6.3. View personal information

6.4. View F.A.Q.

6.5. Logout

7. The Administrator area must provide the following services:

7.1. Pending registration management:

7.1.1. View users pending approval

7.1.2. Approve user

7.1.3. Reject user

7.2. Hospital management:

7.2.1. Register new hospital

7.2.2. Enter hospital wards:

7.2.2.1. Register visit room

7.2.2.2. Register emergency operation room

7.2.2.3. Register hospitalization room

7.2.2.4. Register clinical surgery room

7.3. Pharmacy management:

7.3.1. Register new pharmacy

7.3.2. Register new drug

7.3.3. Add drug description

7.4. Viewing statistics.

7.5. Management of F.A.Q:

        7.5.1. Insert F.A.Q.
        7.5.2. Update F.A.Q.
        7.5.3. Delete F.A.Q.
    7.6. View personal information
    7.7. Logout

In the following sections each use case of the aforementioned areas will be explained more in detail.

3

Implementation of the application and database

The database was implemented using the MYSQL language, while Tomcat was used as a web server.

3.1 | Domain Model

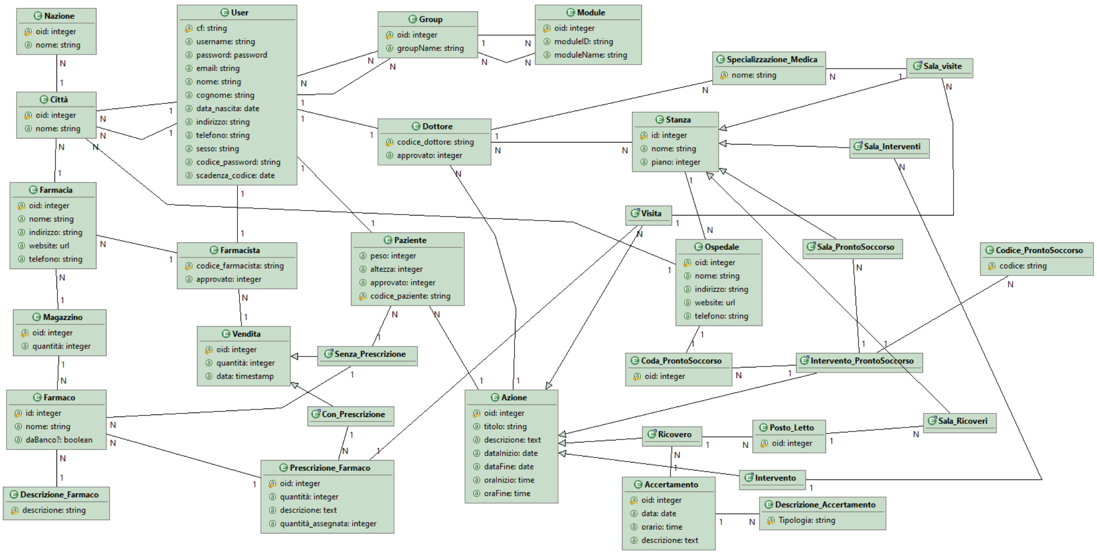


Figure 3.1: Domain Model.

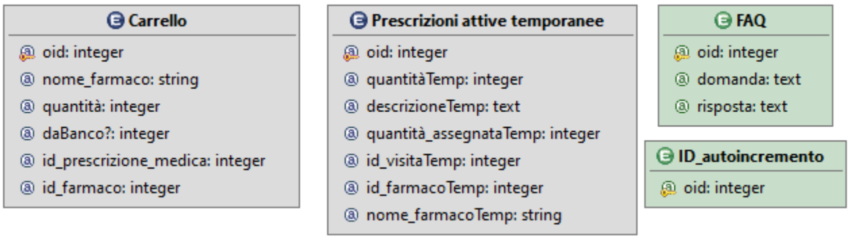


Figure 3.2: Domain Model.

3.1.1 | Data dictionary

3.1.1.1 | Permanent Tables

User		
Attribute name	**Data type**	**Description**
cf	string	user's fiscal code
username	string	is used as foreign key with module Table
password	password	password chosen for the login
email	string	user's email
nome	string	user's given name
cognome	string	user's family name
data nascita	date	user's date of birth
indirizzo	string	user's address
telefono	string	user's phone number
sesso	string	user's gender
codice password	string	is used to set a new password when the user forget the old password
scadenza codice	date	the validity time of 'codice password'

Table 3.1: Primary user entity with all data.

Group		
Attribute name	**Data type**	**Description**
oid	integer	group's id
groupName	string	the name of the user's group, i.e "Patient,etc."

Table 3.2: Default web ratio entity used for the session.

Module		
Attribute name	**Data type**	**Description**
oid	integer	module's id
moduleID	string	the name of the site view
moduleName	string	the name of the user's group

Table 3.3: Default web ratio entity used for the session.

Dottore		
Attribute name	**Data type**	**Description**
codice_dottore	string	id code for the doctor
approvato	int	indicates if the doctor is approved by the administrator system

Table 3.4: Type of system user.

Farmacista		
Attribute name	**Data type**	**Description**
codice_farmacista	string	id code for the pharmacist
approvato	int	indicates if the pharmacist is approved by the administrator system

Table 3.5: Type of system user.

Paziente		
Attribute name	**Data type**	**Description**
codice_paziente	string	id code for the patient
approvato	int	indicates if the patient is approved by the administrator system
altezza	int	patient's height (cm)
peso	int	patient's weight (kg)

Table 3.6: Type of system user.

Nazione		
Attribute name	**Data type**	**Description**
oid	int	country's id
nome	string	country's name

Table 3.7: List of countries present in the system.

Città		
Attribute name	**Data type**	**Description**
oid	int	city's id
nome	string	city's name

Table 3.8: List of cities present in the system.

Farmacia		
Attribute name	**Data type**	**Description**
oid	int	pharmacy's id
nome	string	pharmacy's name
indirizzo	string	pharmacy's address
website	string	pharmacy's website url
telefono	string	pharmacy's telephone number

Table 3.9: Table containing the pharmacies registered in the system.

Magazzino		
Attribute name	**Data type**	**Description**
oid	int	drug supply's id
quantità	int	drug supply's amount

Table 3.10: Quantity of a particular drug present in a certain pharmacy.

Farmaco		
Attribute name	**Data type**	**Description**
id	int	drug's id
nome	string	drug's name
daBanco?	boolean	if true, indicates that the drug can be bought without a prescription

Table 3.11: Drugs in the system.

Descrizione_Farmaco		
Attribute name	**Data type**	**Description**
descrizione	string	description of the drug

Table 3.12: Description of a particular drug

Prescrizione_Farmaco		
Attribute name	**Data type**	**Description**
oid	int	drug prescription's id
quantità	int	indicates the amount of drug prescribed by the doctor
descrizione	text	drug prescription's description
quantità_assegnata	int	amount of drug already sold

Table 3.13: During a visit, a doctor may prescribe drugs. These prescriptions are stored in this table.

Vendita		
Attribute name	**Data type**	**Description**
oid	int	sales id
quantità	int	amount of drug sold to a patient
data	date	date of the sales

Table 3.14: When a pharmacist completes a sale, the sale is saved. Furthermore, it is kept track that the sale concerns a drug without a prescription or with a prescription through a generalization.

Azione		
Attribute name	**Data type**	**Description**
oid	int	action id
titolo	string	the title of the action
descrizione	text	description of the action
dataInizio	date	start date of the action
dataFine	date	end date of the action
oraInizio	time	start time of the action. Is important to know it when the action is a visit or a medical
oraFine	time	end time of the action. Is important to know it when the action is a visit or a medical

Table 3.15: A patient can undergo visits, clinical surgery, hospitalizations and emergency operation. These features are managed through a main entity and a series of generalizations (Ricovero,Intervento,Visita e Intervento pronto soccorso).

Stanza		
Attribute name	**Data type**	**Description**
id	int	room's id
nome	string	room's name
piano	int	the floor of the room inside the hospital

Table 3.16: Each action is tied to a corresponding room. Here we have a main entity and a number of generalizations too.

Specializzazione_Medica		
Attribute name	**Data type**	**Description**
nome	int	medical specialization's name

Table 3.17: A doctor has a medical specialization.

Ospedale		
Attribute name	**Data type**	**Description**
oid	int	hospital's id
nome	string	hospital's name
indirizzo	string	hospital's address
website	url	hospital's website url
telefono	string	hospital's telephone number

Table 3.18: Table containing the hospitals registered in the system.

Coda_ProntoSoccorso		
Attribute name	**Data type**	**Description**
oid	int	medical emergency's queue id of a hospital

Table 3.19: Every emergency room must have a queue.

Codice_ProntoSoccorso		
Attribute name	**Data type**	**Description**
codice	string	indicates the type of medical emergency

Table 3.20: When a patient arrives in the emergency room he must be assigned to a code (green,yellow,red).

Posto_Letto		
Attribute name	**Data type**	**Description**
oid	int	bed place's oid in a specific hospitalization room

Table 3.21: During a hospitalization, a bed is assigned to the patient.

Accertamento		
Attribute name	**Data type**	**Description**
oid	int	medical assessment's id
data	date	medical assessment's date
orario	time	medical assessment's start time
descrizione	text	can indicate both the description of the assessment to be made and the results obtained

Table 3.22: Some checks that can be completed during an hospitalization.

Descrizione_Accertamento		
Attribute name	**Data type**	**Description**
tipologia	string	the type of a specific medical assessment

Table 3.23: Type of check carried out (tac,blood collection etc.)

FAQ		
Attribute name	**Data type**	**Description**
oid	int	FAQ's id
domanda	text	indicates the question
risposta	text	indicates the answer to the question

Table 3.24: Table to store the F.A.Q.

ID_Autoincremento		
Attribute name	**Data type**	**Description**
oid	int	id used to create the patient, doctor and pharmacist codes

Table 3.25: Table that generates auto-increasing ids.

3.1.1.2 | Temporary and Session Tables

Carrello		
Attribute name	**Data type**	**Description**
oid	int	element id inside the shopping cart
nome_farmaco	string	the name of the drug inside the shopping cart
quantità	int	the amount of drug insert inside the shopping cart
daBanco?	int	indicates if the drug inside the shopping cart required the prescription
id_prescrizione_medica	int	the id of visit prescription
id_farmaco	int	the id of the drug inside the shopping cart

Table 3.26: Temporary table to manage the shopping cart. (They cannot be permanently stored in the database before the sale is complete.)

Prescrizioni Attive Temporanee		
Attribute name	**Data type**	**Description**
oid	int	temporary active prescription id
quantitàTemp	int	indicates the amount of drug prescribed by the doctor
descrizioneTemp	text	indicates the description wrote inside the prescription by the doctor
quantità_assegnataTemp	int	indicates the amount of drug already sold
id_visitaTemp	int	indicates the id of the visit of the prescription
id_farmacoTemp	int	indicates the id of the drug prescribed by the doctor
nome_farmacoTemp	string	indicates the name of the drug prescribed by the doctor

Table 3.27: Temporany table to manage the active drug prescriptions in the UI. (They cannot be permanently stored in the database before the sale is complete.)

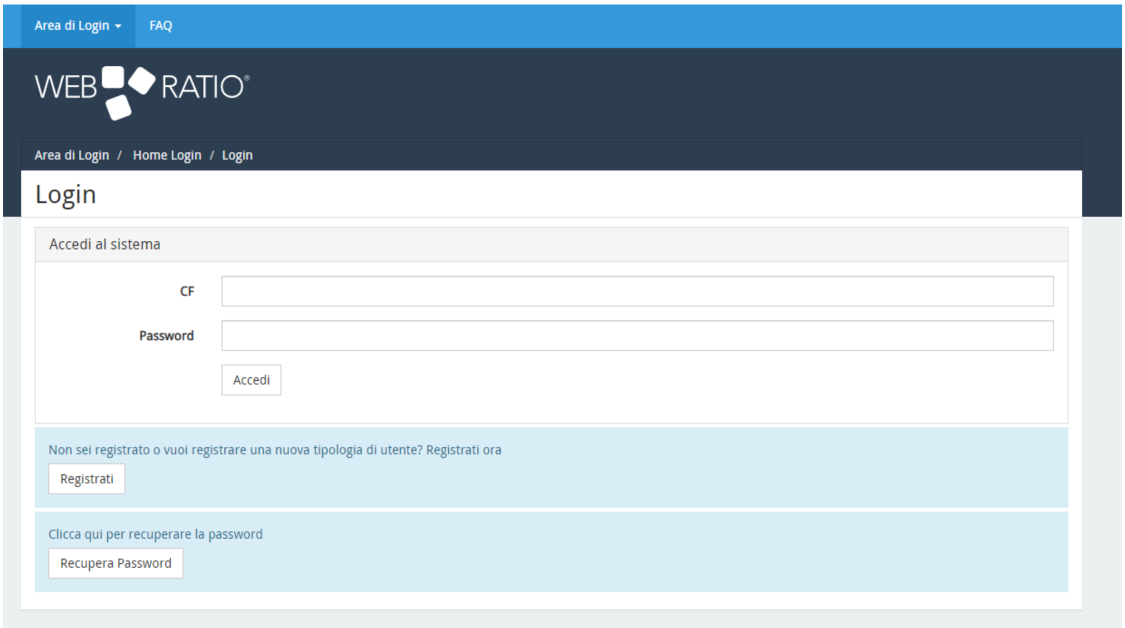
3.2 | Application development

The application is structured as follows:
We find 6 modules that define the macro areas:

- Login area
- Patient area
- Pharmacist area
- Specialist doctor area
- Emergency doctor area
- Administrator area.

Each macro area, inside, is composed of pages and contains the modules necessary to the development of the use-case.

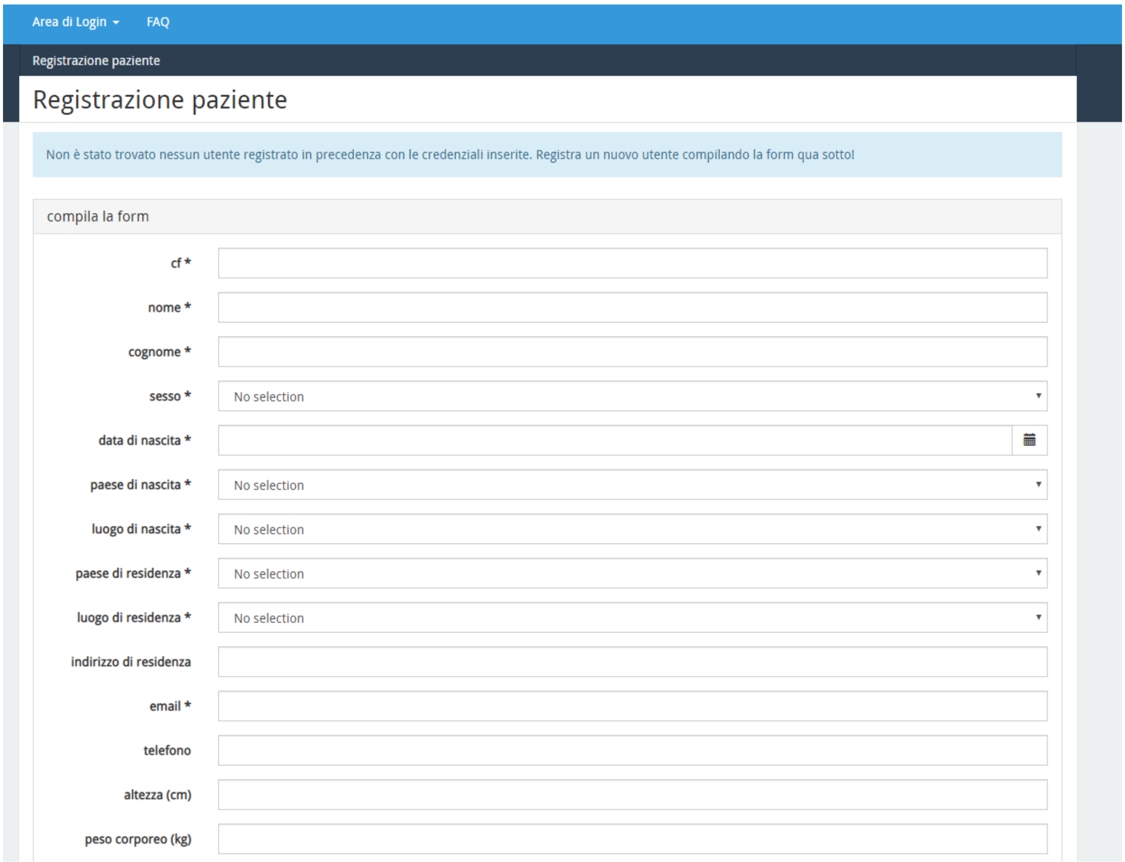
3.2.1 | Login area



Non sei registrato o vuoi registrare una nuova tipologia di utente? Registrati ora

Clicca qui per recuperare la password

Figure 3.3: Home SanitarySystemProject.



Registrazione paziente

Registrazione paziente

Non è stato trovato nessun utente registrato in precedenza con le credenziali inserite. Registra un nuovo utente compilando la form qua sotto!

compila la form

cf *

nome *

cognome *

sesso * No selection

data di nascita *

paese di nascita * No selection

luogo di nascita * No selection

paese di residenza * No selection

luogo di residenza * No selection

indirizzo di residenza

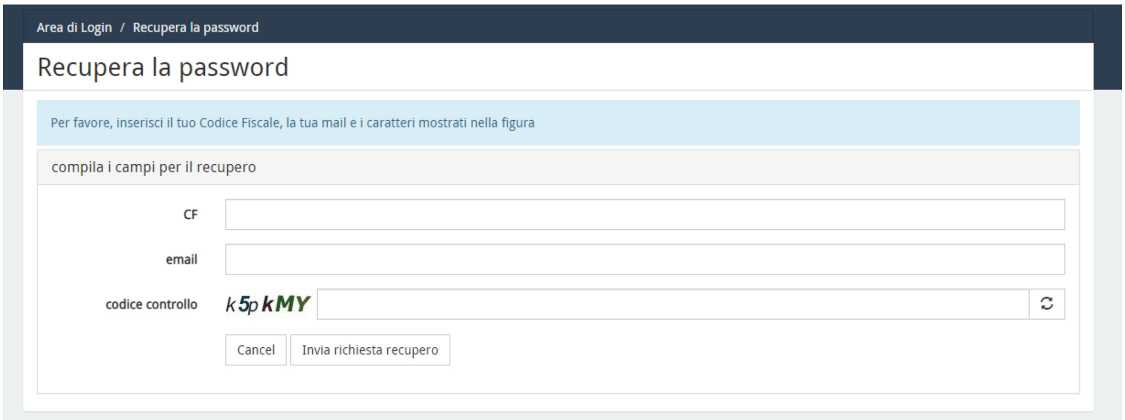
email *

telefono

altezza (cm)

peso corporeo (kg)

Figure 3.4: Example of registration page.



Area di Login / Recupera la password

Recupera la password

Per favore, inserisci il tuo Codice Fiscale, la tua mail e i caratteri mostrati nella figura

compila i campi per il recupero

CF

email

codice controllo k5pkMY

Figure 3.5: Password recovery section.

The Login screen is the one visible to all types of system visitors. Here, the user not yet registered in the system can access the registration form while if he is already registered he can enter his data and go directly to the Login phase. There is also the password recovery function. Let's take a closer look at the following use-cases.

3.2.1.1 | UC - Registration

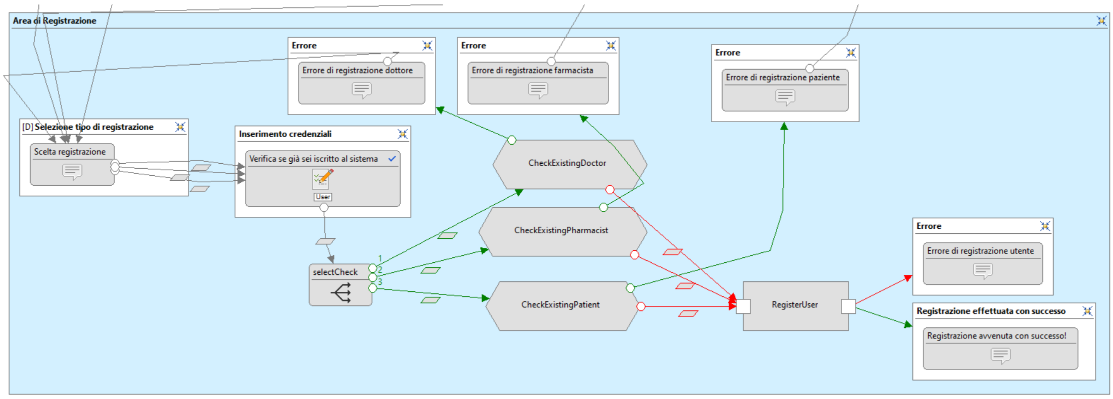

Area di Registrazione
Errore
Errore di registrazione dottore
Errore
Errore di registrazione farmacista
Errore
Errore di registrazione paziente
[D] Selezione tipo di registrazione
Scelta registrazione
Inserimento credenziali
Verifica se già sei iscritto al sistema
User
CheckExistingDoctor
CheckExistingPharmacist
selectCheck
1
2
3
CheckExistingPatient
RegisterUser
Errore
Errore di registrazione utente
Registrazione effettuata con successo
Registrazione avvenuta con successo!

Figure 3.6: Registration area.

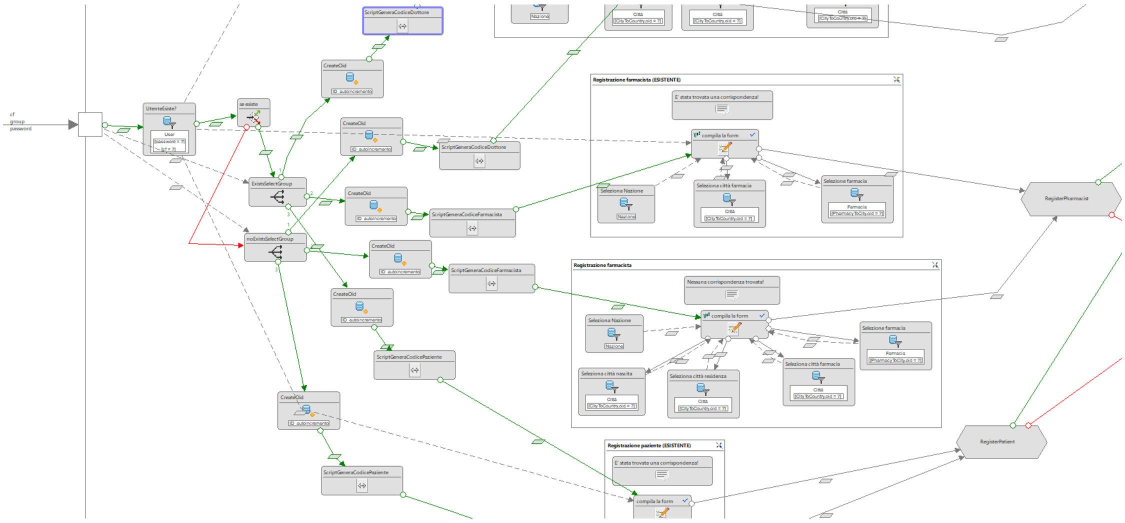

Registrazione farmacista (ESISTENTE)
Registrazione farmacista
Registrazione paziente (ESISTENTE)
RegisterPharmacist
RegisterPatient

Figure 3.7: Registration management extract.

A visitor can initially decide which type of user wants to register, in particular he can choose between Patient, Doctor, Pharmacist. Afterwards, the visitor will be asked if it is already registered to the system. He will enter the login credentials (fiscal code and password) and the system will check if there is a correspondence with the data entered in the system. If it exists, all the fields of the form with personal data (user data 2.1.1.) will be filled with the data recovered from a previous registration, otherwise the visitor will have to fill them out together with the specific data to be known based on the type of registration chosen (patient, doctor or pharmacist).
Specific data of patient are:

- height (cm);
- weight (kg);

Specific data of doctor are:

- hospital where he/she works;
- his/her medical specialization;
- the visit room of the selected hospital and with the selected medical specialization not busy by another doctor. The work room will be an emergency room inside the hospital if the type of medical specialization selected is equal to *'pronto soccorso'*.

Specific data of pharmacist are:

- pharmacy where he/she works;

Some constraints implemented are:

- date of birth cannot be greater than today;
- the fields with asterisk (*) are mandatory;
- can't exist two different people with same fiscal code.

3.2.1.2 | UC - Login

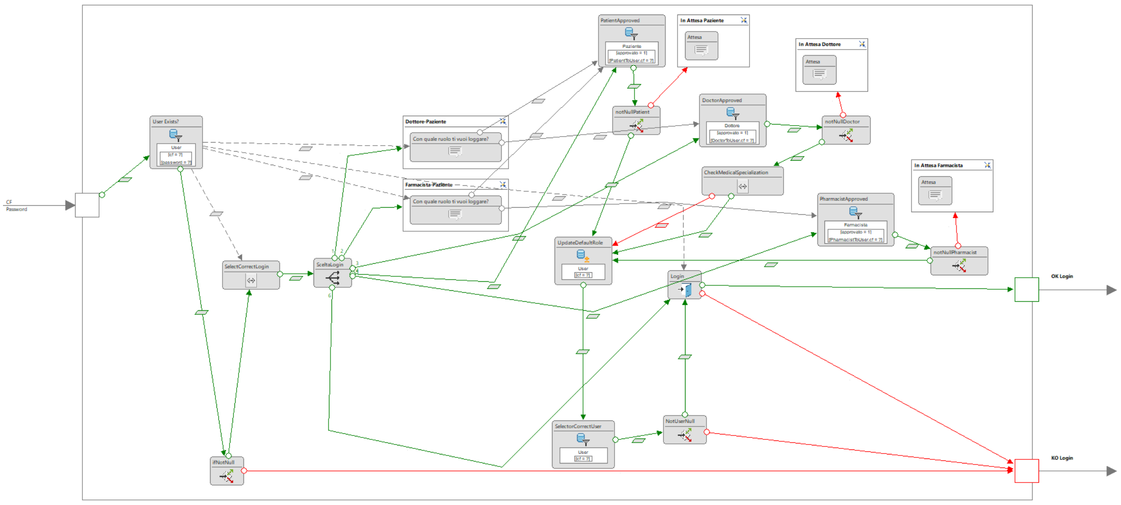


Figure 3.8: Implementation of the Login module.

A visitor will fill the login form with his credentials that are fiscal code and password. The system will check if the user exists and if it approved at least one user typology by the administrator. Here, there are more options:

- If he/she has more user typologies approved, the system will ask him with which kind of user wants to log in;
- If he/she has more user typologies registered but all typologies are not approved, the system will warn the visitor that he wasn't approved yet, so he can't access;
- If it is registered only with a user typology and the credentials are correct, it will log in, else the system will warn the visitor that the credentials aren't correct.

3.2.1.3 | UC - Password recovery

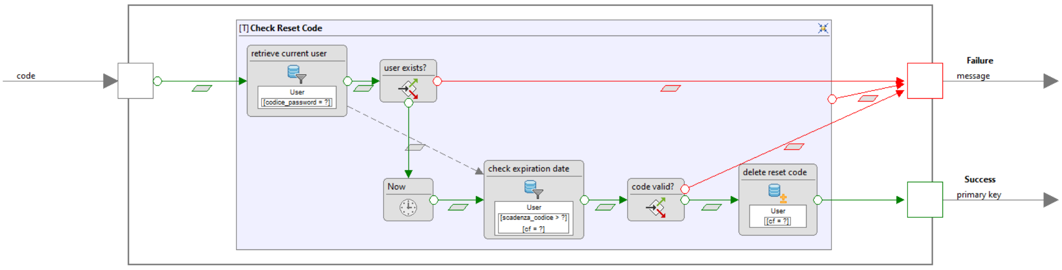


Figure 3.9: Reset Password.

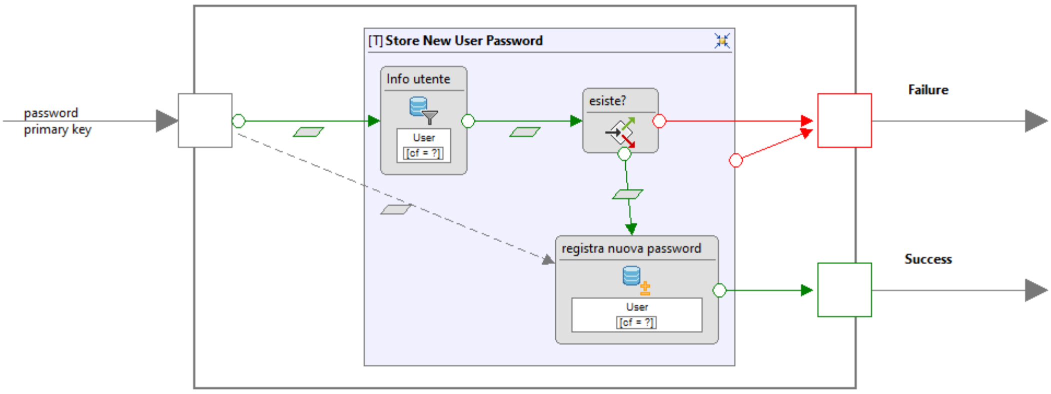


Figure 3.10: Reset Password.

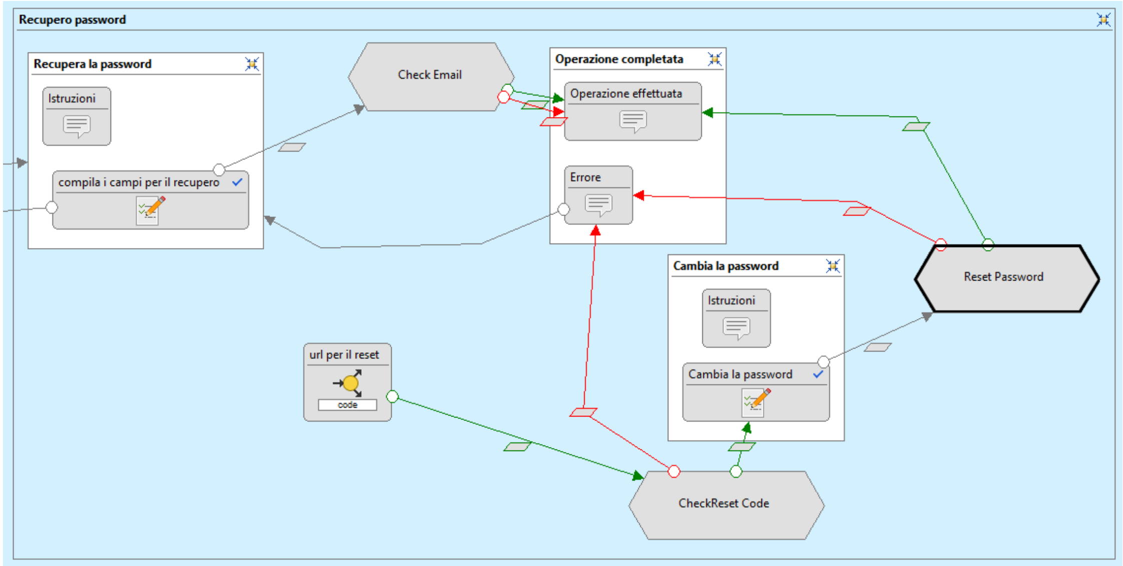


Figure 3.11: Reset Password.

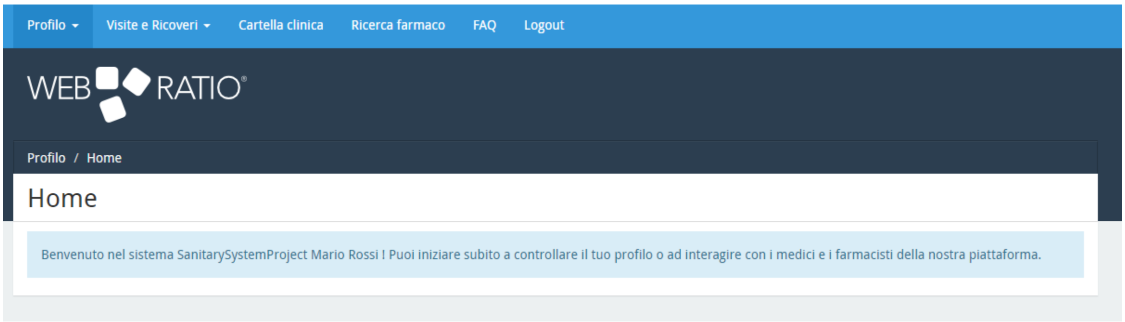
In case that a user forgot the password, he can to reset it. The reset password form requires to the user to fill the fiscal code and email fields and requires to insert a captcha code. If the fiscal code and the email inserted correspond to an user registered and the captcha code is correct, the user will receive an email with a link. This link open a page to reset the password: it has a field password, another field to confirm the password inserted and the preloaded user fiscal code.
Some constrains are implemented:

- when the form requires user fiscal code and email must correspond to the same user;

- the field new password and confirm password must have the same inserted password.

3.2.2 | Patient area

When a visitor access as patient, has the following site view area 3.17 as personal homepage. In this subsection, are shown the actions that he can do.



Profilo / Home

Home

Benvenuto nel sistema SanitarySystemProject Mario Rossi ! Puoi iniziare subito a controllare il tuo profilo o ad interagire con i medici e i farmacisti della nostra piattaforma.

Figure 3.12: Patient home page.

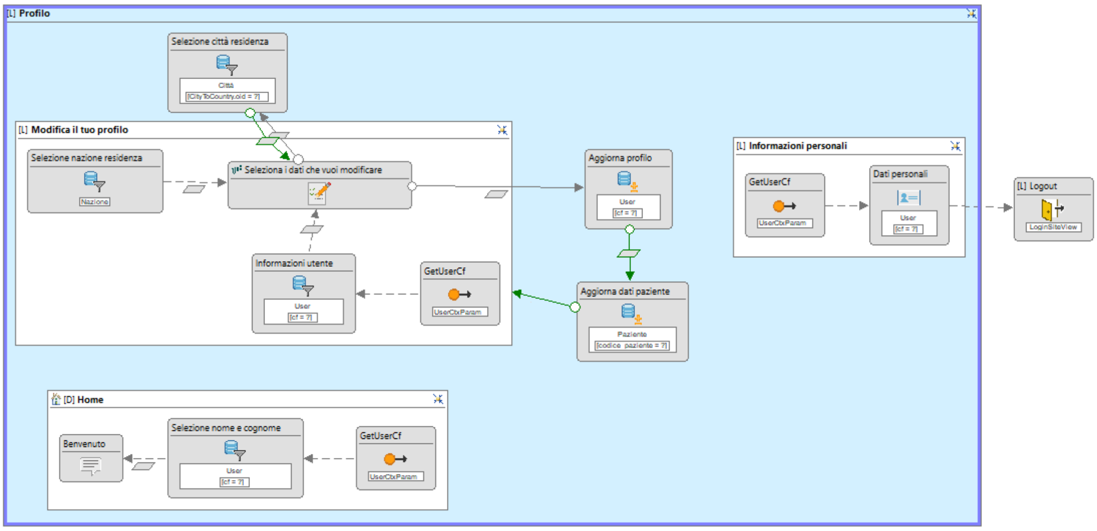


Figure 3.13: Patient profile.

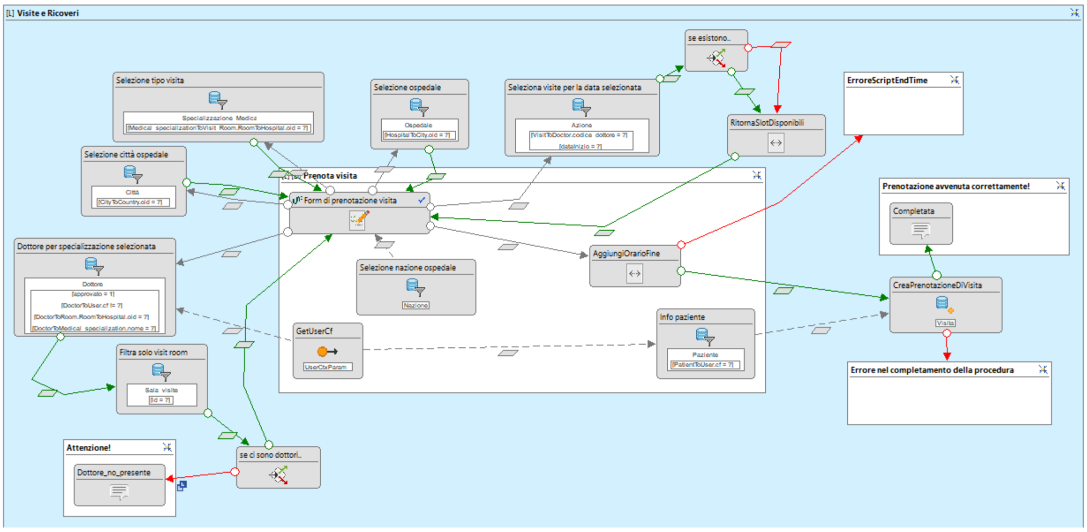


Figure 3.14: Booking of the visit.

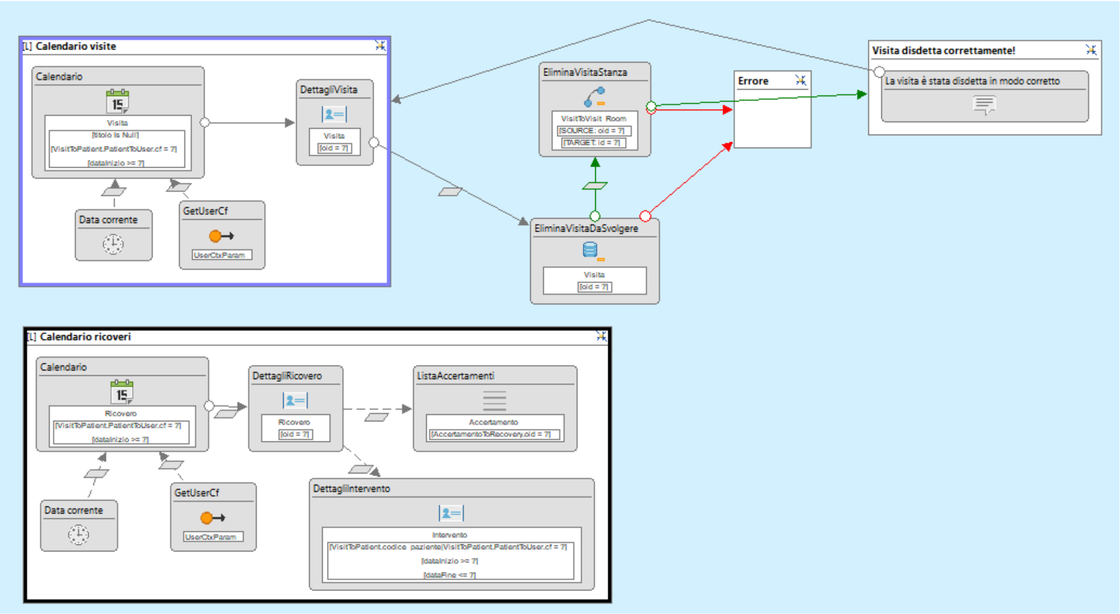


Figure 3.15: Calendar of patient visits.

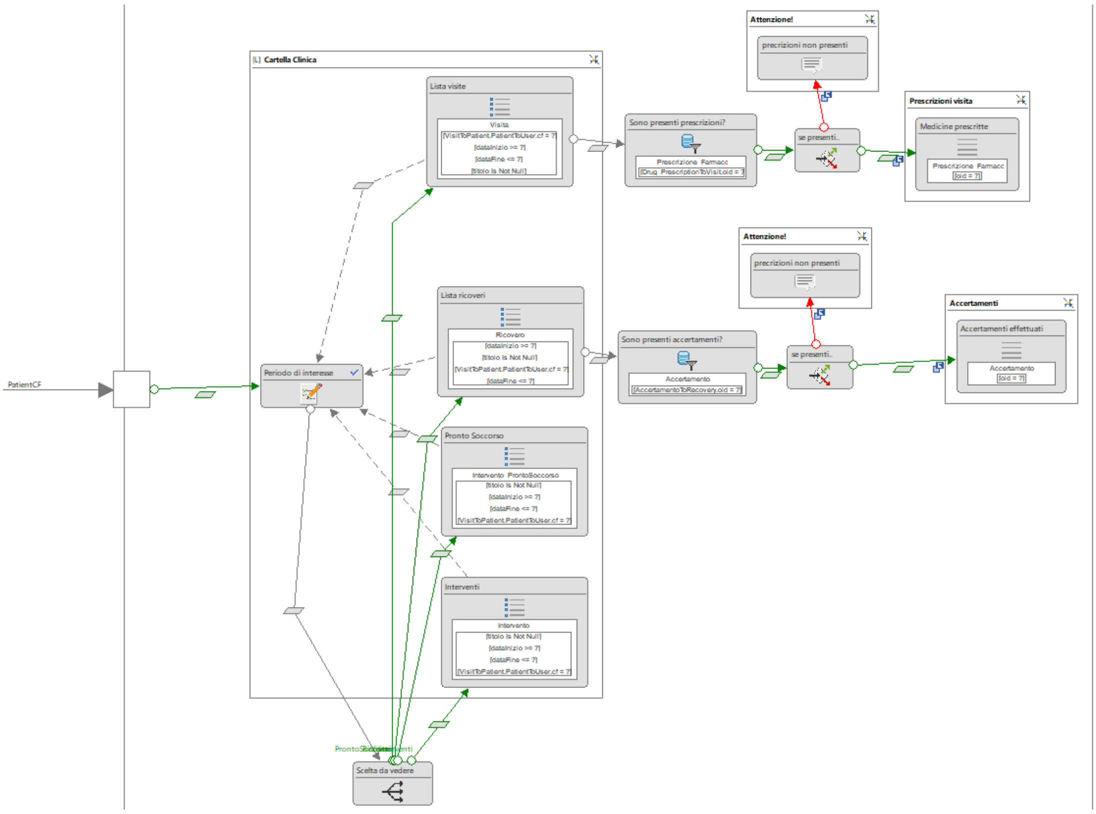


Figure 3.16: Patient’s medical record.

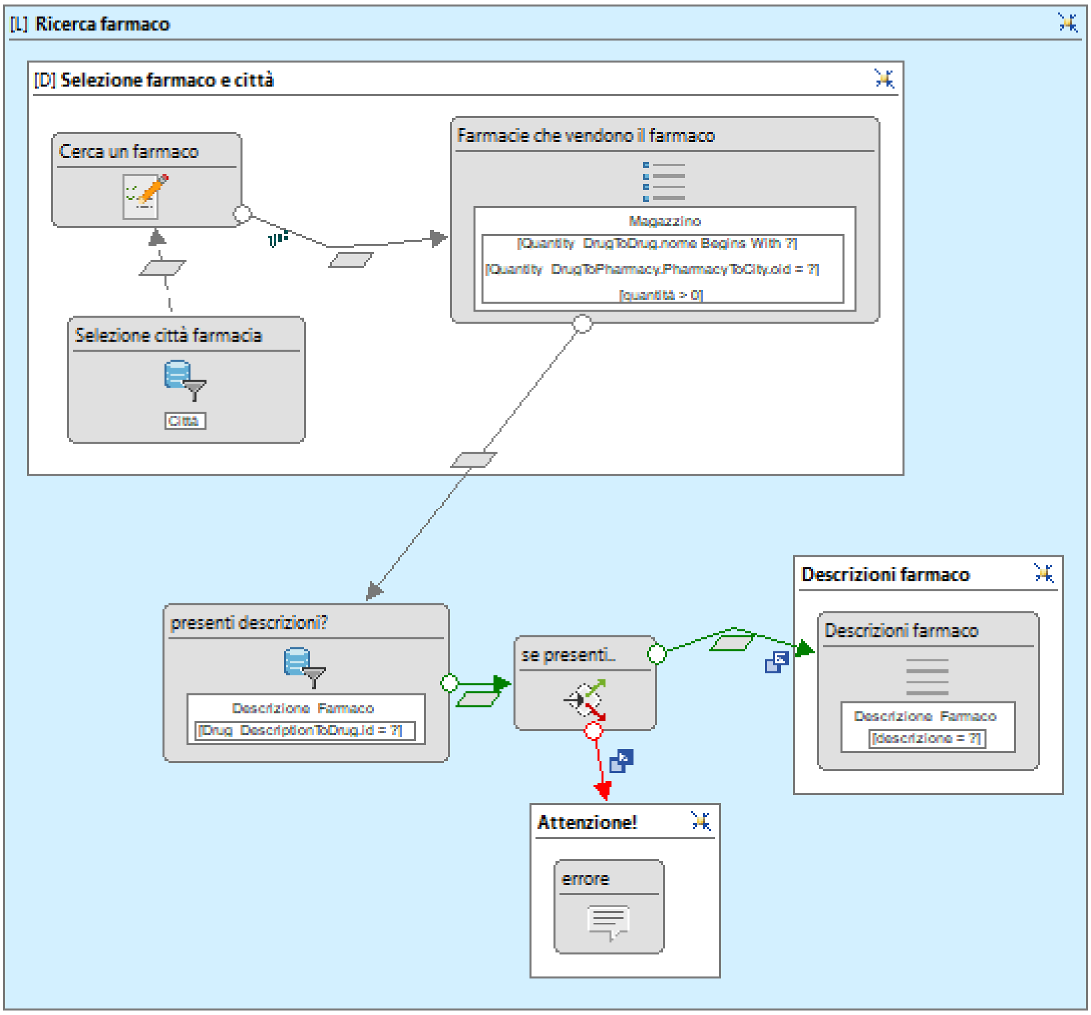
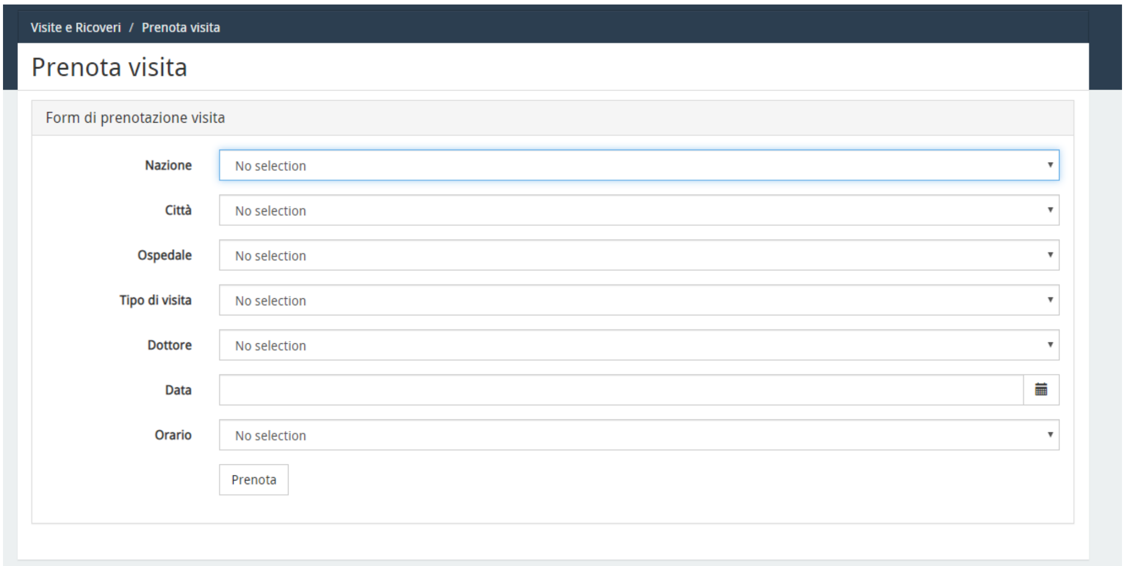


Figure 3.17: Drug search.

3.2.2.1 | UC - Booking of visits



Visite e Ricoveri / Prenota visita

Prenota visita

Form di prenotazione visita

Nazione	No selection ▾
Città	No selection ▾
Ospedale	No selection ▾
Tipo di visita	No selection ▾
Dottore	No selection ▾
Data	
Orario	No selection ▾

Figure 3.18: Visit booking form.

A patient can books a visit through the specific form. The form has the following fields:

- nation of the hospital;
- city of the hospital;
- hospital;
- type of visit;
- doctor for that type of visit;
- date;
- start time.

It was implemented a script to select only the available start times of the visit, checking the other visits with that doctor for that day. For simplicity, each visit has a slot time of 30 minutes and the end time is calculated adding 30 minutes to the selected start time. A patient can books a visit every day of the year, from 8.00 a.m to 6.00 pm. The patient will view a error message if there aren't doctors for the type of visit selected. When the

patient submit the form is going to registered an *'Azione'* of type *'Visita'* in the database, without title and description fields compiled.
Some constrains implemented are:

- all the fields are mandatory;
- if the patient is also a doctor, can't does the visit with himself;
- the patient can't choose a date less or equal than today;
- only doctors approved can be selected;
- all the visits in the same visit room of the selected doctor are disconnected times.

3.2.2.2 | UC - Visit calendar display

From a calendar, the patient can see the booked visits not yet done. Each visit on the calendar shows the doctor surname, the type of visit and the start time. If the patient click on it, the page show the details that are:

- hospital name;
- hospital address;
- visit room name;
- visit room floor.

From here, he can decide to unsubscribe the booked visit.
Some constrains implemented are:

- date of the visit must be greater or equal than today;
- must be shown on the calendar only visits with title and description null, because these are visit not registered yet from the doctor.

3.2.2.3 | UC - Hospitalization calendar display

From a calendar, the patient can see the subscribed hospitalizations registered by the doctors who visited it not yet done. Each hospitalization on the calendar shows the title. If the patient click on it, the page show the details that are:

- title;

- description;
- start date;
- end date;
- hospital name;
- doctor surname;
- hospitalizion room name;
- bed place assigned.

If is programmed a clinical surgery during the hospitalization, then are shown the details that are:

- clinical surgery typology;
- title;
- description;
- hospital name;
- doctor surname;
- date;
- start time;
- end time;

If are programmed medical check-ups during the hospitalization, then is shown a list of them with the following details:

- description;
- date;
- start time;

Some constrains implemented are:

- date of the hospitalization must be greater or equal than today;

3.2.2.4 | UC - See medical record

A patient can check his clinical folder. The search form contains three fields:

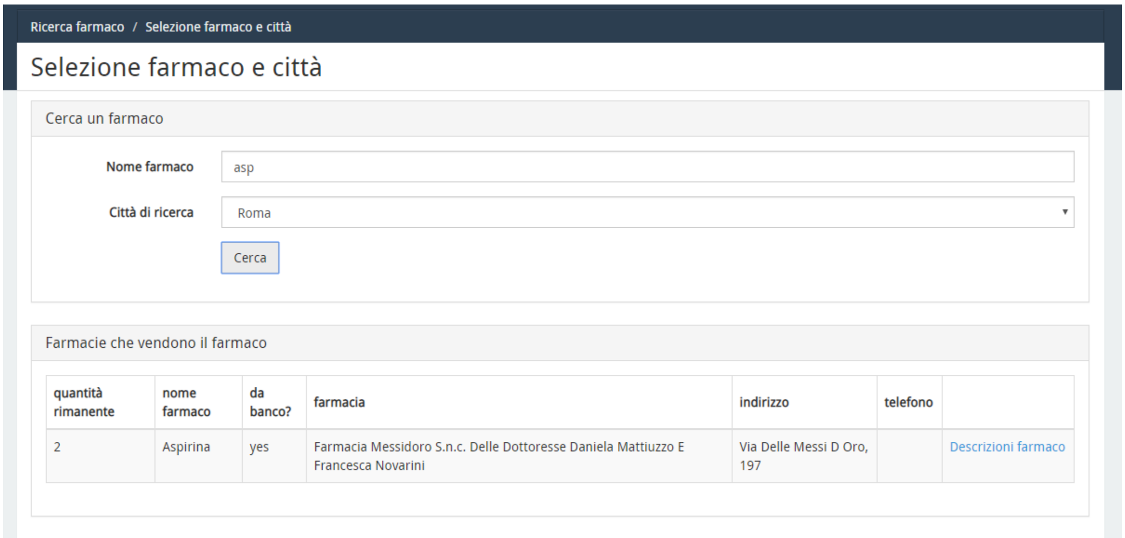
- date of start search;
- date of end search;
- type of search that user wants to see.
  It can be:
  - old registered visits (*"Visite"*);
  - old hospitalizations (*"Ricoveri"*);
  - old clinical surgeries (*"Interventi"*);
  - old emergencies (*"pronto soccorso"*).

If the type of search selected is equal to *'Visite'* and if the patient has old registered visits, they are shown in a list with some details. The patient can also see drug prescriptions details from that visit if are present, else the system will warn the patient that they aren't present . Else if the type of search selected is equal to *'Ricoveri'* and if the patient has old registered hospitalizations, they are shown in a list with some details. The patient can also see the medical check-ups details from that hospitalization if are present, else the system will warn the patient that they aren't present. Else the type of search selected is equal to *'Interventi'* or *'pronto soccorso'* and if the patient has old registered clinical surgeries or emergencies, they are shown in a list with some details.
Some constrains implemented are:

- date of start search and date of end search must be less then today;
- date of start search must be less then date of end search;
- the type of search must be mandatory.

3.2.2.5 | UC - Drug search

A patient can search which are the pharmacies in specific city that sell a specific drug. The search form has the fields city, and drug name, as shown in 3.19. The system executes a query on the drug inventory of the pharmacies in the selected city and return a list with all the drugs that start with the name insert by the patient, showing to it all the details and the remaining quantity in that pharmacy. A patient can clicks on the drug to see the descriptions of that. The search result return only the drugs with a remain quantity greater than 0.



Ricerca farmaco / Selezione farmaco e città

Selezione farmaco e città

Cerca un farmaco

Nome farmaco: asp

Città di ricerca: Roma

Farmacie che vendono il farmaco

quantità rimanente	nome farmaco	da banco?	farmacia	indirizzo	telefono	
2	Aspirina	yes	Farmacia Messidoro S.n.c. Delle Dottoresse Daniela Mattiuzzo E Francesca Novarini	Via Delle Messi D Oro, 197		Descrizioni farmaco

Figure 3.19: Drug search page.

3.2.2.6 | UC - Profile management

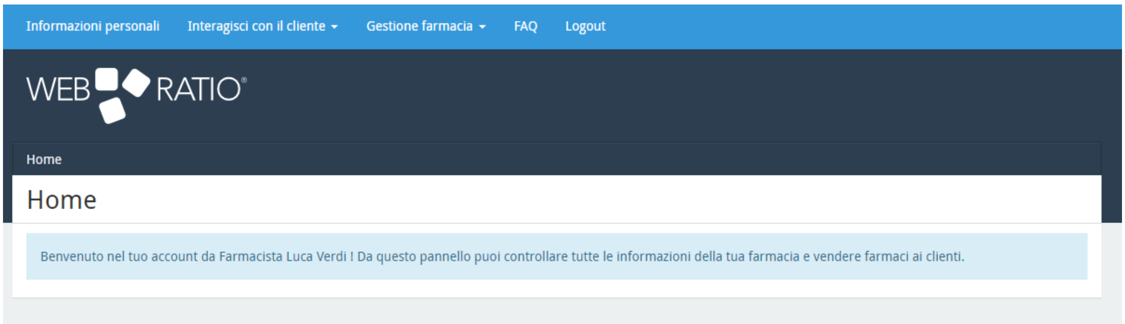
A patient can see the personal data inserted during the registration to the system. He can also edit some data.

3.2.2.7 | UC - Visualization of F.A.Q.

A patient can see the FAQ inserted by the administrators of the system.

3.2.3 | Pharmacist area

When a visitor access as pharmacist, has the following site view area 3.22 as personal homepage. In this subsection, are shown the actions that he can do.



Home

Benvenuto nel tuo account da Farmacista Luca Verdi ! Da questo pannello puoi controllare tutte le informazioni della tua farmacia e vendere farmaci ai clienti.

Figure 3.20: Pharmacist home page.

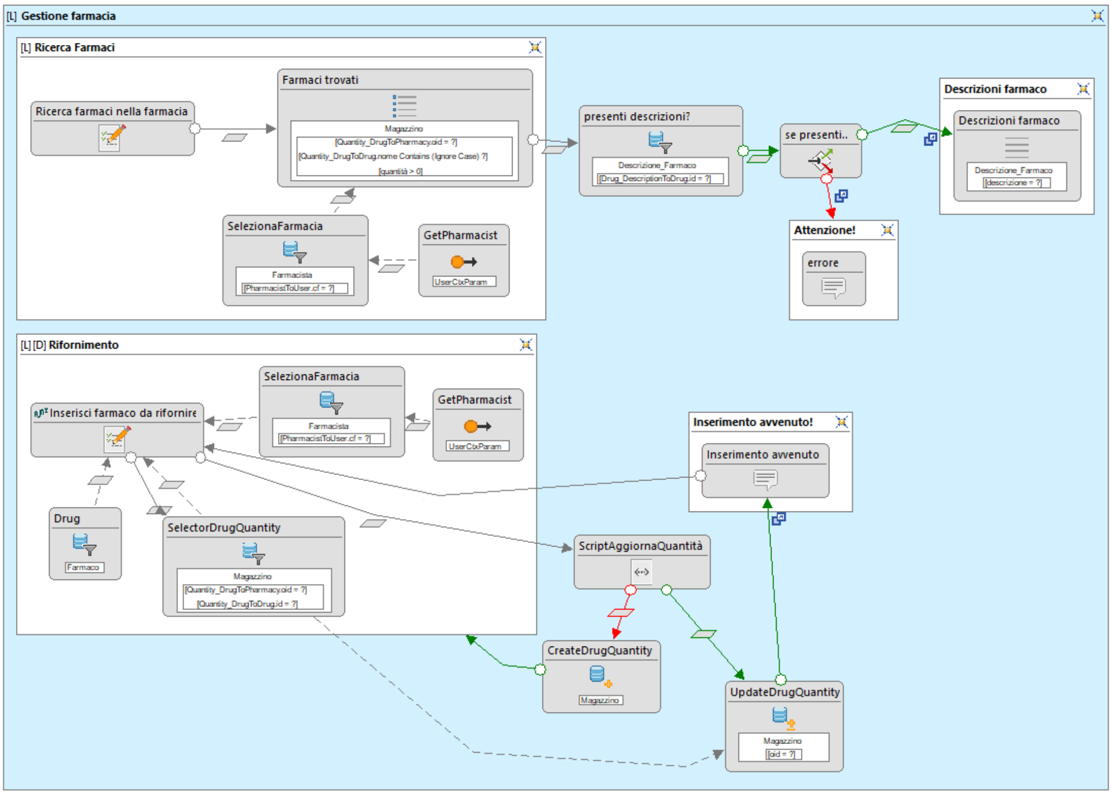


Figure 3.21: Pharmacy functionality.

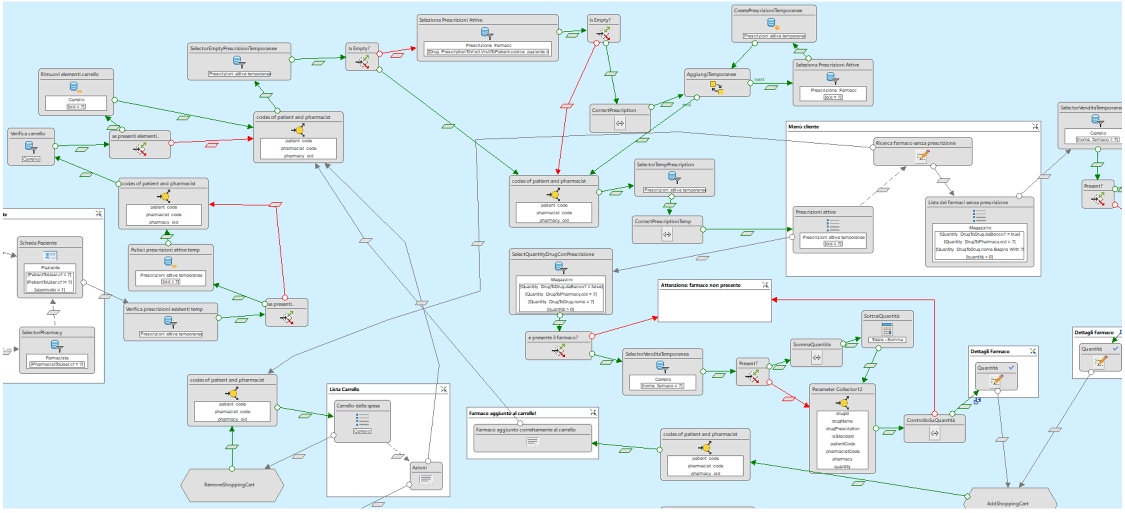

Selezione Prescrizioni Attive
Is Empty?
CreatePrescrizioniTemporanee
SelectorEmptyPrescrizioniTemporanee
Rimuovi elementi carrello
Verifica carrello
se presenti elementi
codes of patient and pharmacist
AggiungiTemporanee
Seleziona Prescrizioni Attive
CorrectPrescription
SelectorVenditaTemporanee
Menù cliente
Ricerca farmaco senza prescrizione
SelectorTempPrescription
CorrectPrescriptionTemp
Prescrizioni attive
Lista dei farmaci senza prescrizione
Scheda Paziente
Pulisci prescrizioni attive temp
Verifica prescrizioni esistenti temp
SelectorPharmacy
SelectQuantityDrugConPrescrizione
Attenzione: farmaco non presente
è presente il farmaco?
SommaQuantità
Present?
Dettagli Farmaco
Parameter Collector12
ControlloSuQuantità
Lista Carrello
Carrello della spesa
Azioni
Farmaco aggiunto al carrello!
Farmaco aggiunto correttamente al carrello
RemoveShoppingCart
AddShoppingCart

Figure 3.22: shopping cart implementation.

3.2.3.1 | UC - Interact with the customer

A pharmacist can interacts with a patient inserting his fiscal code inside a form. If the fiscal code correspond to a patient registered to the system, the pharmacist has the access to the patient data, else the system warn the pharmacist that the fiscal code doesn't exists. When the pharmacist access to patient data is initialized a new sell session and the shopping cart that is represented as session scope table in the database. Then is shown a new page that contains a list with the patient active drug prescriptions, where each prescription has the drug name, the quantity initially prescripted by the doctor, the quantity assigned by others pharmacists. All the drug prescriptions are copied into a session scope database table, because in this way there are no persistent changes to the database before the purchase is completed. Once that the pharmacist select a specific drug prescription, the system verifies if the drug is present inside his pharmacy. If is present, the pharmacist select the quantity to sell to the patient, and the drug quantity will add to the shopping cart with the drug prescription reference. If isn't present, the system warn the pharmacist that can't proceed with the operation. If a pharmacist sell all the remaining quantity indicated on the drug prescription to the patient, the drug prescription will not visible in the list of active drug prescriptions. A pharmacist can sell also drug that don't required drug prescription in the same sell session, simply search them with a field. It's possible to check the shopping cart and remove some drugs from it. When the pharmacist concludes the sell, the system for each drug inside the shopping cart, checks if the drug is sell with or without prescription to create in the

database the right table reference and reference the drug to the correct drug prescription if exists. Then removes the quantity from the pharmacy of the specific drug, update the quantity assigned in correct drug prescription and removes it from the shopping cart.
Some constrains implemented are:

- a pharmacist can't sell drugs to himself;
- a pharmacist can sell drugs only to patients approved by the system;
- a pharmacist can sell drug that require drug prescriptions to the patient only if the patient has active prescriptions;
- the selected quantity of a drug to sell must be less or equal then the quantity present in the pharmacy;
- the selected quantity of a drug to sell must be less or equal then the remain quantity to assign.

3.2.3.2 | UC - Drugs inventory

A pharmacist through a specific form can select a drug present in the database of the system and the quantity to add to the present quantity in his pharmacy. If the drug was added in the past, the quantity of that drug is updated, else is created a new reference of the drug in the table. *'Magazzino'*.

3.2.3.3 | UC - Search for drugs

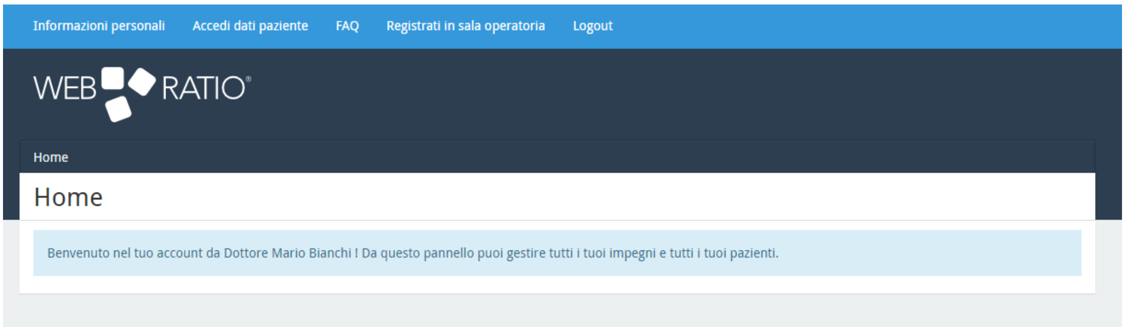
The procedure is similar to 3.2.2.5, but in the search form the pharmacist has to insert only the drug name. If the drug is present, the system show to the patient the drug information and the quantity.

3.2.3.4 | UC - View F.A.Q.

See UC:3.2.2.7

3.2.4 | Specialist doctor area

When a visitor access as doctor with medical specialization different by *"pronto soccorso"*, has the following site view area 3.31 as personal homepage. In this subsection, are shown the actions that he can do.



Home

Benvenuto nel tuo account da Dottore Mario Bianchi ! Da questo pannello puoi gestire tutti i tuoi impegni e tutti i tuoi pazienti.

Figure 3.23: Specialist doctor home page.

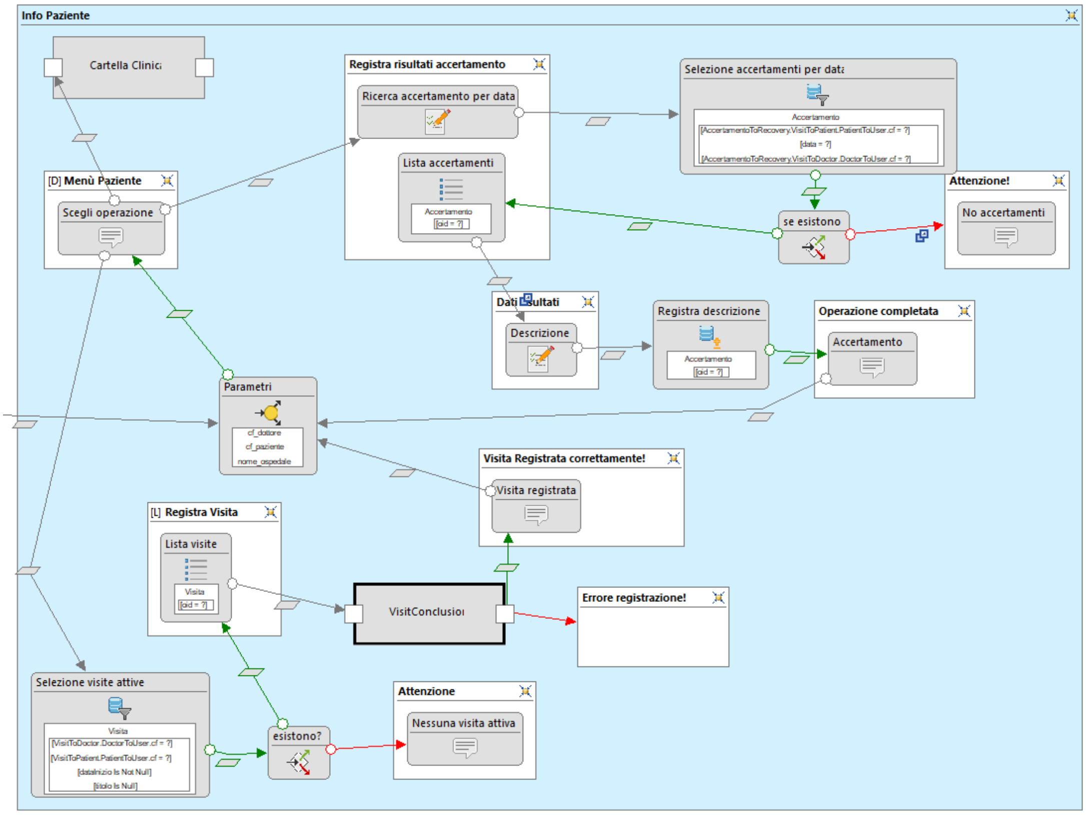
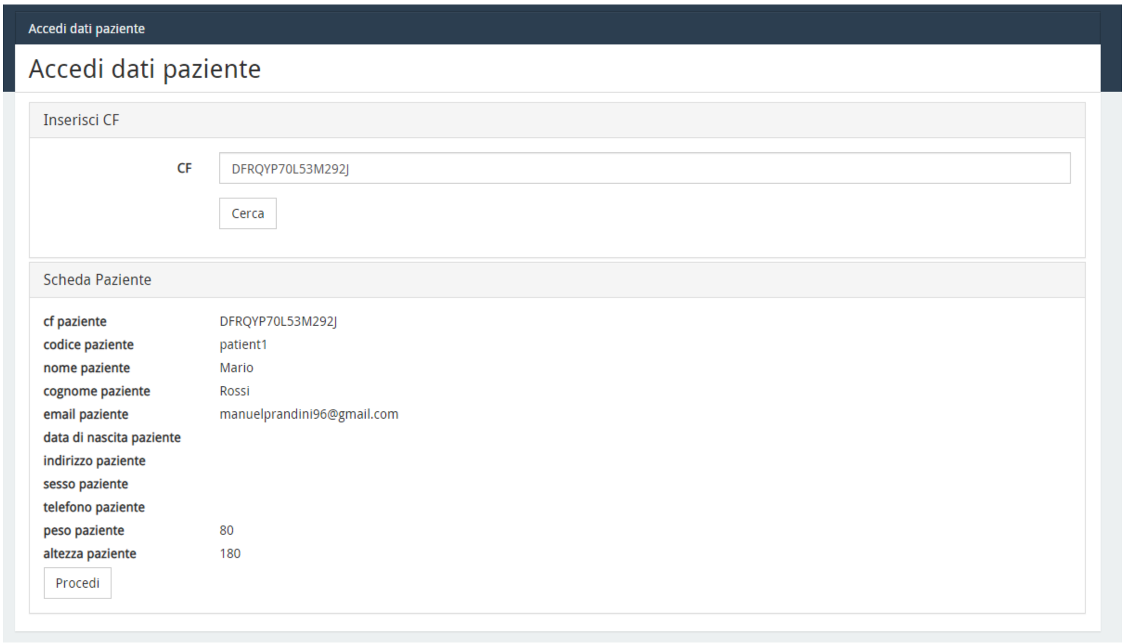


Figure 3.24: Patient info implementation.



Accedi dati paziente

Accedi dati paziente

Inserisci CF

CF DFRQYP70L53M292J

Cerca

Scheda Paziente

cf paziente	DFRQYP70L53M292J
codice paziente	patient1
nome paziente	Mario
cognome paziente	Rossi
email paziente	manuelprandini96@gmail.com
data di nascita paziente	
indirizzo paziente	
sesso paziente	
telefono paziente	
peso paziente	80
altezza paziente	180

Procedi

Figure 3.25: Access to patient data.

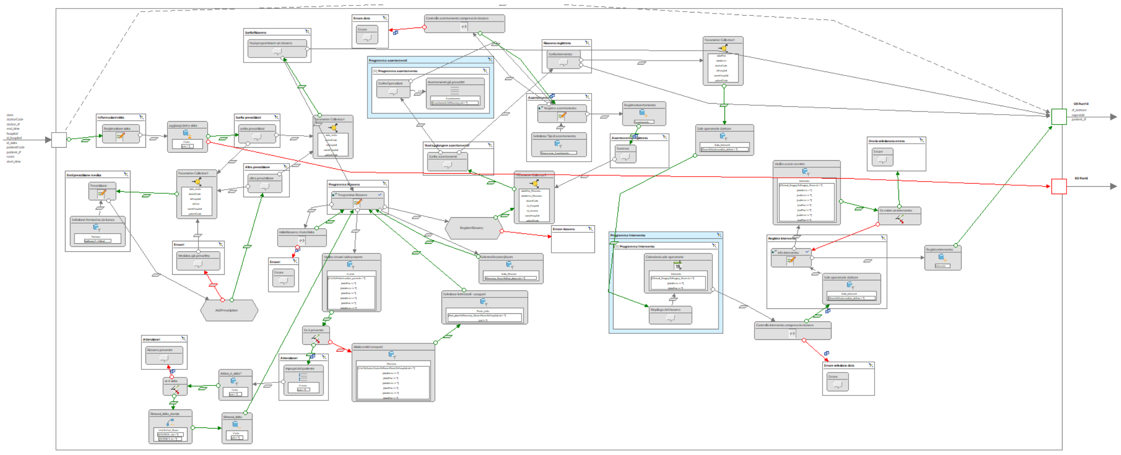
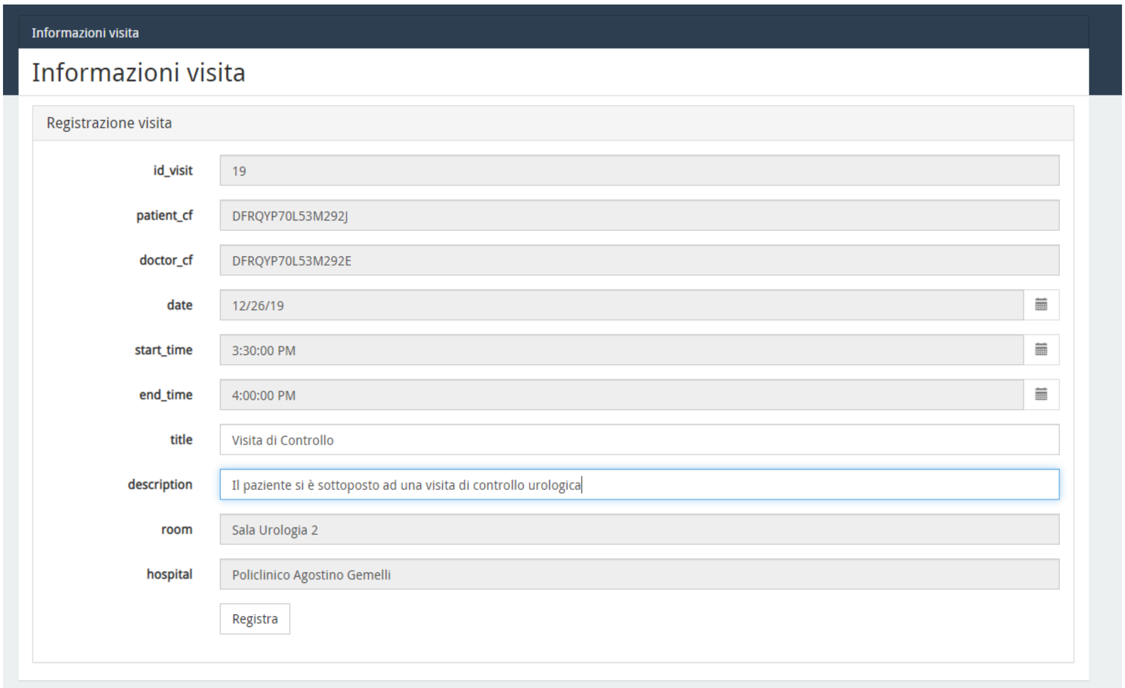


Figure 3.26: Implementation of the conclusion of the visit.



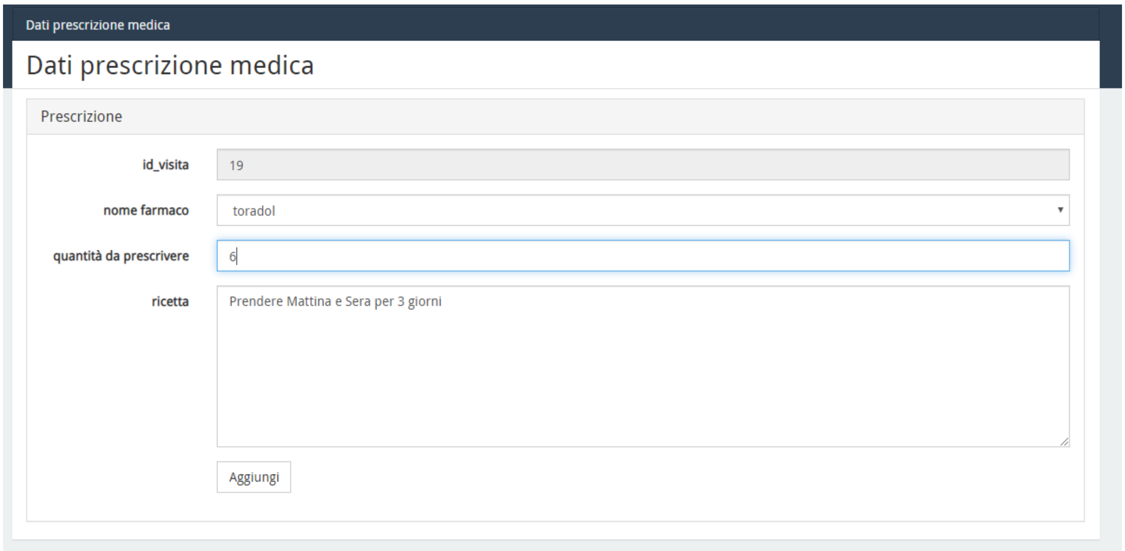
Informazioni visita

Informazioni visita

Registrazione visita

id_visit	19
patient_cf	DFRQYP70L53M292J
doctor_cf	DFRQYP70L53M292E
date	12/26/19
start_time	3:30:00 PM
end_time	4:00:00 PM
title	Visita di Controllo
description	Il paziente si è sottoposto ad una visita di controllo urologica
room	Sala Urologia 2
hospital	Policlinico Agostino Gemelli

Figure 3.27: Visit completion form.



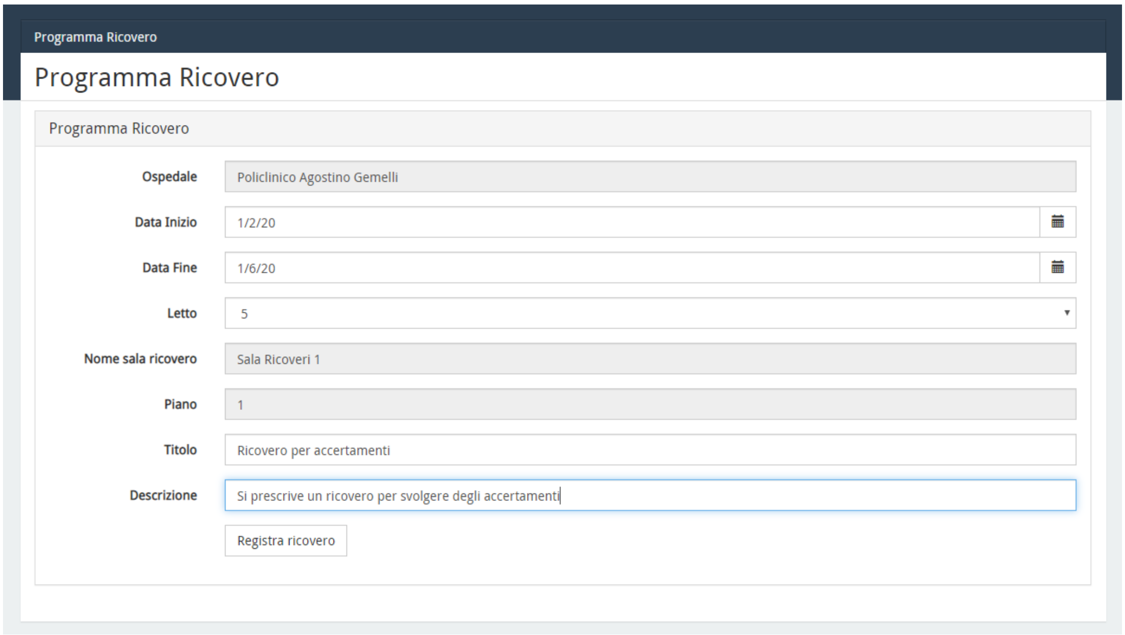
Dati prescrizione medica

Dati prescrizione medica

Prescrizione

id_visita	19
nome farmaco	toradol
quantità da prescrivere	6
ricetta	Prendere Mattina e Sera per 3 giorni

Figure 3.28: Add a prescription to the visit.

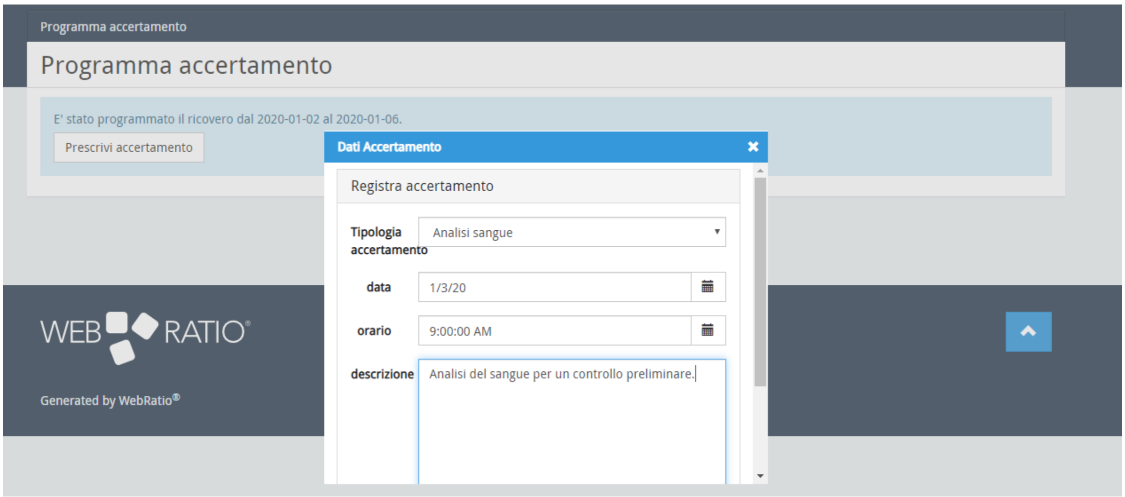


Programma Ricovero

Programma Ricovero

Ospedale	Policlinico Agostino Gemelli
Data Inizio	1/2/20
Data Fine	1/6/20
Letto	5
Nome sala ricovero	Sala Ricoveri 1
Piano	1
Titolo	Ricovero per accertamenti
Descrizione	Si prescrive un ricovero per svolgere degli accertamenti

Figure 3.29: Prescription of a hospitalization.



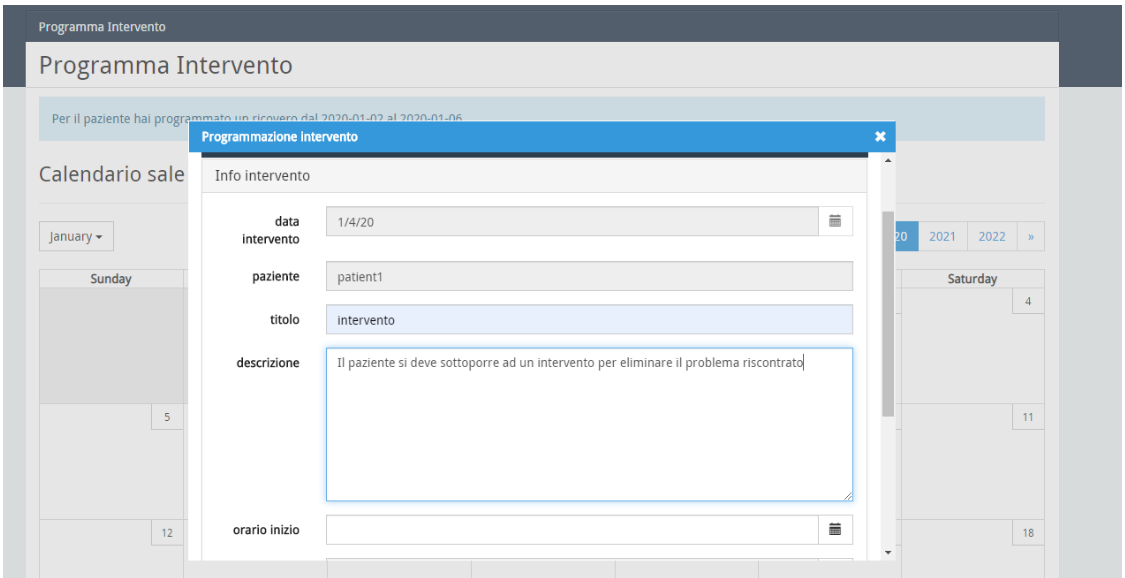
Programma accertamento

E' stato programmato il ricovero dal 2020-01-02 al 2020-01-06.

Registra accertamento

Tipologia accertamento	Analisi sangue
data	1/3/20
orario	9:00:00 AM
descrizione	Analisi del sangue per un controllo preliminare.

Figure 3.30: Prescription of medical check-ups during hospitalization.



Programma Intervento

Programma Intervento

Per il paziente hai programmato un ricovero dal 2020-01-02 al 2020-01-06

Calendario sale

Programmazione intervento

Info intervento

data intervento: 1/4/20

paziente: patient1

titolo: intervento

descrizione: Il paziente si deve sottoporre ad un intervento per eliminare il problema riscontrato

orario inizio:

Figure 3.31: Prescription of clinical surgery.

3.2.4.1 | UC - Access patient data

A doctor can interacts with a patient inserting his fiscal code inside a form. If the fiscal code correspond to a patient registered to the system, the doctor has the access to the patient data, else the system warn the doctor that the fiscal code doesn't exists or is wrong. When a doctor has the access to the patient data, he can do different things: see the clinical folder of the patient 3.2.4.2, register results of medical check-ups 3.2.4.9 or register the patient's visit just made 3.2.4.4.

Some constrains implemented are:

- the patient must be approved by the administrator of the system;
- a doctor cannot access his patient data.

3.2.4.2 | UC - Consult the patient's medical record

A doctor can access to the old patient data like old visits done by him, hospitalizations, clinical surgeries and emergencies. The operation of the use-case is similar to 3.2.2.4.

3.2.4.3 | UC - Record medical check-ups results

A doctor can register the results of some medical check-ups occurred during an hospitalization of a patient. Using a form, he can select the date of the medical check-ups

to register. If they are present, the system returns a list with all the medical check-ups done by the patient in that day, else the system warn the doctor that there aren't medical check-ups for the selected date. So, the doctor can click on the list element and change his description. The system will update the description of the medical check-up.
Some constraints implemented are:

- the medical check-ups can be register only by the doctor who performed it.

3.2.4.4 | UC - Record visit

A doctor can register the data of a patient visit just made. In the registration form there are some fields preloaded like the visit room, the hospital, the visit doctor etc. The doctor can insert the title of the visit and the description. After that, he can choose if prescribe some drugs 3.2.4.5 or jump directly to the choice of a hospitalization prescription 3.2.4.6.
Some constraints implemented are:

- the doctor can register only the visits of a patient just made, so the day of the visit is equal to the current day and the current time is greater or equal than the start time and less or equal than end time. (for the test phase, this constraint wasn't used).

3.2.4.5 | UC - Prescribe drug

Once a doctor has recorded the data of a patient visit, he can make several medical prescriptions. In the prescription form, he must enter the name of the drug, the quantity to be prescribed and the description on the method of intake. The system will create a new drug prescription inside the database related to the patient visit. Once that the registration of prescription is terminated, the doctor can decide if prescribe another prescription or prescribe an hospitalization for the patient 3.2.4.6.
Some constraints implemented are:

- the prescribed quantity must be greater than 0;
- the doctor can't do more drug prescriptions of the same drug;
- the drug prescriptions are related only to the visit just made by the patient;
- the doctor can prescribe only drugs that require a prescription.

3.2.4.6 | UC - Prescribe hospitalization for the patient visited

After the registration of patient visit data, the doctor can decide if prescribe an hospitalization or not. In the hospitalization form he has to insert the period of the hospitalization, the free bed in that period, title and description (for simplicity, there are no start and end times). If the doctor select a period and in this period the patient has other visits or hospitalizations, the system warns him returning a list with details. The doctor can ask to the patient if delete a booked visit but not an other programmed hospitalization. In this way, if the patient has no more booked visits, the doctor can proceed with the registration of the hospitalization. Once that the hospitalization was registered, the doctor can decide if prescribe some medical check-ups 3.2.4.7 or a clinical surgery 3.2.4.8 during the hospitalization or not.
Some constraints implemented are:

- start date of the hospitalization must be greater than the related visit;
- start date of the hospitalization must be less than end date;
- a doctor can't prescribe an hospitalization in a specific period to a patient if this latter has other visits or hospitalizations during that period;
- a doctor cannot assign a bed already assigned to another hospitalization during the scheduled hospitalization period.
- a doctor can assign to the hospitalization only a bed that is in an hospitalization room inside the doctor hospital.

3.2.4.7 | UC - Prescribe medical check-ups during hospitalization

During an hospitalization, a doctor can prescribe some medical check-ups to the patient inserting the type of medical check-up (for example blood analysis etc), date, time and description. Once that the system register the medical check-up to the patient hospitalization, this will be added to a list of medical check-ups just prescribed for that hospitalization by the doctor and visible by this latter. After that the doctor has prescribed the medical check-ups, can decide if prescribe a clinical surgery to the patient 3.2.4.8 or not.
Some constraints implemented are:

- the date of the medical check-ups must be greater or equal than the start date of the hospitalization and less or equal than the end date of the hospitalization.

3.2.4.8 | UC - Prescribe clinical surgery

During an hospitalization, a doctor can prescribe a clinical surgery to a patient. Through a calendar the doctor can see the information of other clinical surgeries if during the period of hospitalization prescribed to the patient there are other clinical surgeries in the surgery rooms where he is enabled to operate. After choosing the day, the doctor has to enter the start and end times of the surgery, a surgery room where he works, the title and description. Only one clinical surgery can be prescribed during the hospitalization period.

Some constraints implemented are:

- the date of the clinical surgery must be greater or equal than the start date of the hospitalization and less or equal than the end date of the hospitalization.
- the start time of the clinical surgery must be less than the end time.
- the doctor can operate only in surgery room where is registered.
- the doctor can register a clinical surgery on a certain day and at a certain time in an surgery room only if the surgery room is not occupied by another clinical surgery.

3.2.4.9 | UC - Registration in the surgery room

A doctor, to do clinical surgeries on the patients, must be registered in a surgery room. Through a form, he can choose to register by a surgery room inside the hospital where he works.

Some constraints implemented are:

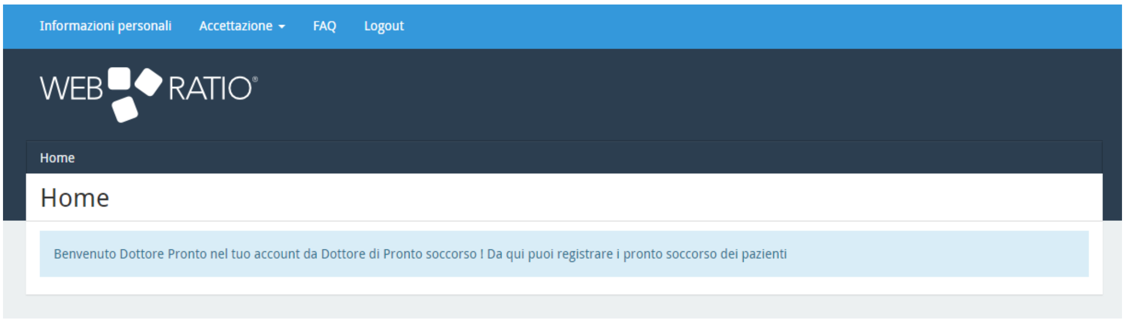
- the doctor can register himself only in surgery rooms that are inside the hospital where he works;
- a doctor can't register in a surgery room where is already registered.

3.2.4.10 | UC - View F.A.Q.

See UC:3.2.2.7

3.2.5 | Emergency doctor area

When a visitor access as doctor with medical specialization equal to *"pronto soccorso"*, has the following site view area 3.36 as personal homepage. In this subsection, are shown the actions that he can do.



Home

Benvenuto Dottore Pronto nel tuo account da Dottore di Pronto soccorso ! Da qui puoi registrare i pronto soccorso dei pazienti

Figure 3.32: Emergency doctor home page.

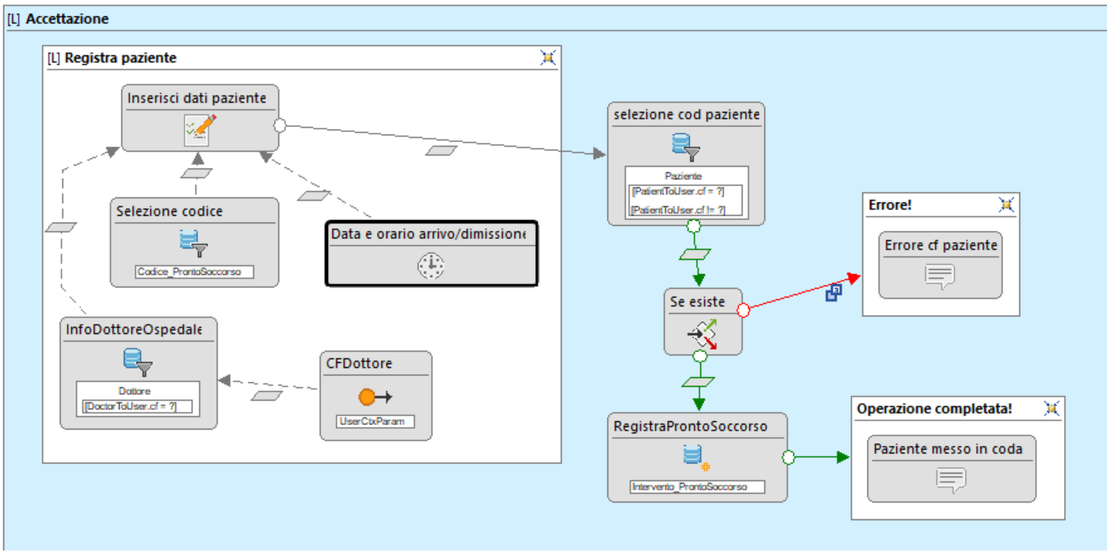


Figure 3.33: Implementation of acceptance.

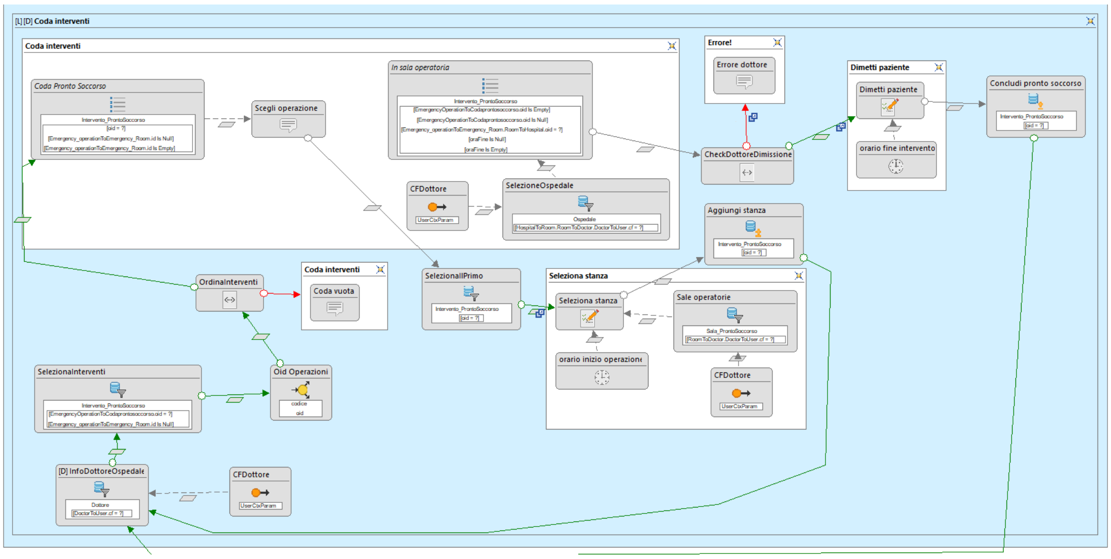
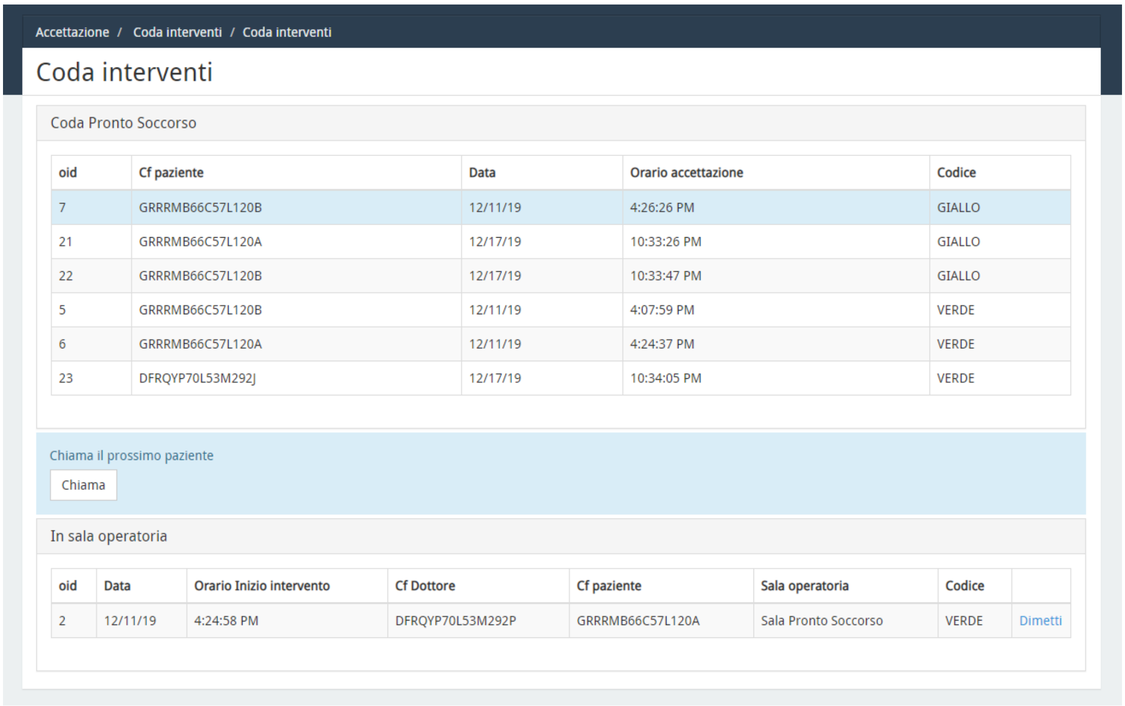


Figure 3.34: Implementation of the queue.



Coda interventi

Coda Pronto Soccorso

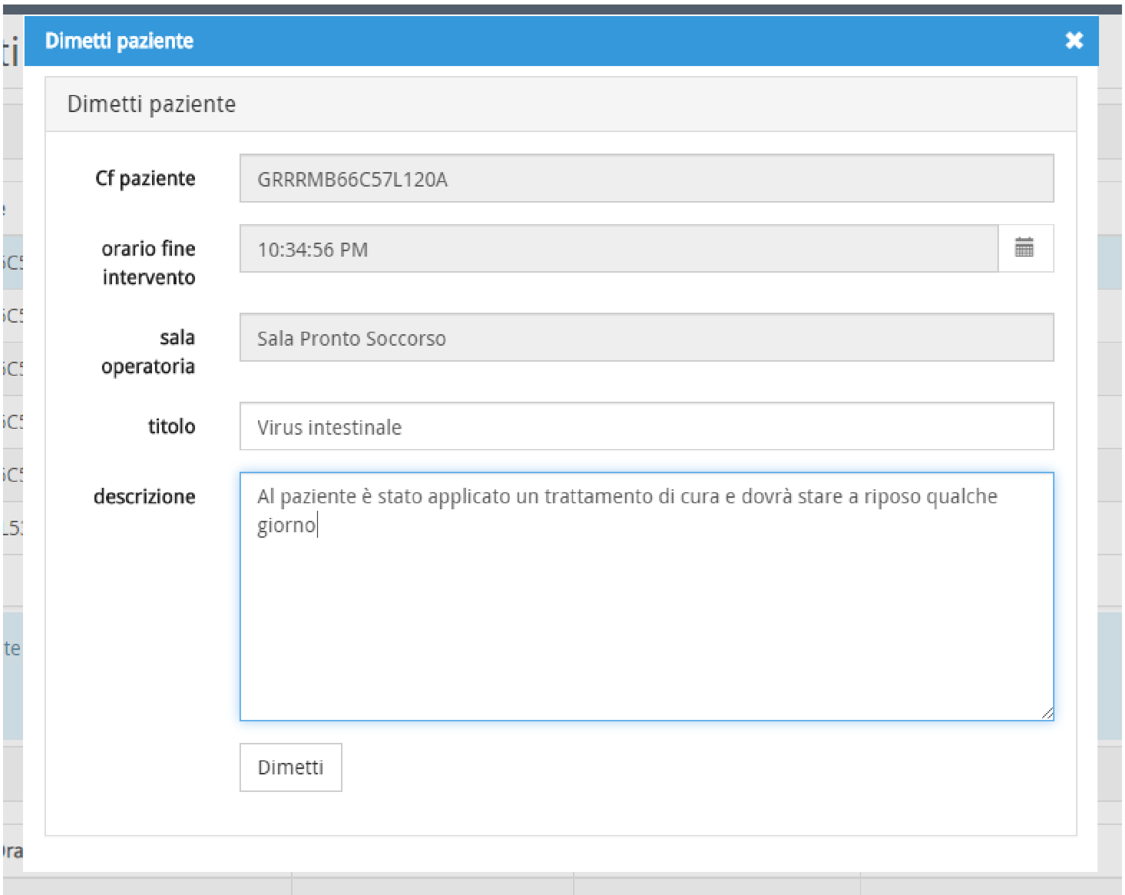
oid	Cf paziente	Data	Orario accettazione	Codice
7	GRRRMB66C57L120B	12/11/19	4:26:26 PM	GIALLO
21	GRRRMB66C57L120A	12/17/19	10:33:26 PM	GIALLO
22	GRRRMB66C57L120B	12/17/19	10:33:47 PM	GIALLO
5	GRRRMB66C57L120B	12/11/19	4:07:59 PM	VERDE
6	GRRRMB66C57L120A	12/11/19	4:24:37 PM	VERDE
23	DFRQYP70L53M292J	12/17/19	10:34:05 PM	VERDE

Chiama il prossimo paziente

In sala operatoria

oid	Data	Orario Inizio intervento	Cf Dottore	Cf paziente	Sala operatoria	Codice	
2	12/11/19	4:24:58 PM	DFRQYP70L53M292P	GRRRMB66C57L120A	Sala Pronto Soccorso	VERDE	Dimetti

Figure 3.35: Queue display.



Dimetti paziente

Dimetti paziente

Cf paziente	GRRRMB66C57L120A
orario fine intervento	10:34:56 PM
sala operatoria	Sala Pronto Soccorso
titolo	Virus intestinale
descrizione	Al paziente è stato applicato un trattamento di cura e dovrà stare a riposo qualche giorno

Figure 3.36: Form to be completed to discharge a patient.

3.2.5.1 | UC - Acceptance: Register patient in the emergency room

An emergency doctor can register a patient when he arrives to the emergency inserting in a form the fiscal code and the emergency code. Acceptance's date and time are values calculated automatically by the system. The patient will be visible in the waiting list by the doctors 3.2.5.2 of the same emergency of the emergency doctor that register the patient.
Some constraints implemented are:

- the fiscal code must be related to a patient registered and approved inside the system;
- the fiscal code must be different by the fiscal code of the emergency doctor.

3.2.5.2 | UC - Acceptance: View waiting list

An emergency doctor can see all the patients registered by him or by other doctors in the hospital emergency where it work. The list is ordered by first the emergency code (the order is : *RED, YELLOW, GREEN*) and then by the timestamp acceptance. In this way, an emergency doctor can call the next patient from the queue 3.2.5.3.

3.2.5.3 | UC - Acceptance: Call patient from the waiting list

An emergency doctor can call the next patient from the emergency queue, and when this happens, he has to insert one emergency room where works and where the patient will be visited. The start time of the acceptance will update with the start time of the emergency operation. When he does this, the patient will be removed from the waiting emergency queue and will be added to the list of patients called in emergency rooms. The doctor will register the data at the end of the emergency 3.2.5.4.
Some constraints implemented are:

- the doctor can call the patients only in the emergency rooms where is registered.

3.2.5.4 | UC - Acceptance: Discharge patient from emergency room

An emergency doctor, from the list of the patients called in emergency rooms can discharge a patient from it. The doctor has to register the title and the description of the emergency and patient will be removed from the list.
Some constraints implemented are:

- the doctor can remove from the list only patients called by him.

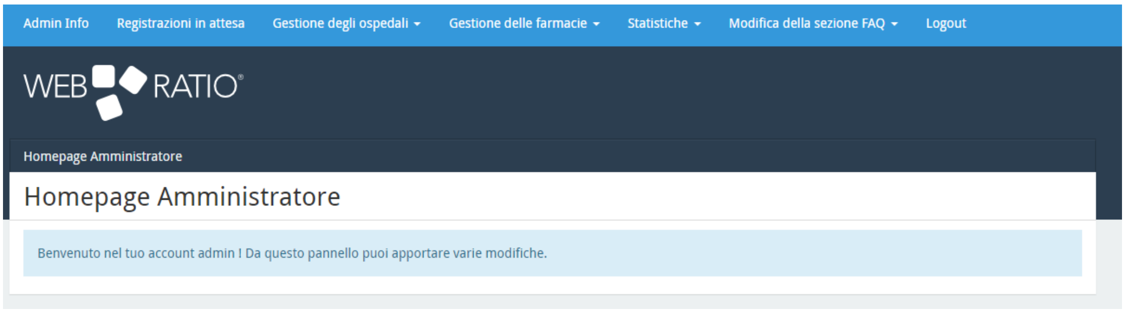
3.2.5.5 | UC - Consult the patient's medical record

The same thing of 3.2.2.4.

3.2.5.6 | UC - View F.A.Q

3.2.6 | Administrator Area

When a visitor access as administrator, has the following site view area 3.40 as personal homepage. In this subsection, are shown the actions that he can do.



Homepage Amministratore

Benvenuto nel tuo account admin ! Da questo pannello puoi apportare varie modifiche.

Figure 3.37: Home Page Admin.

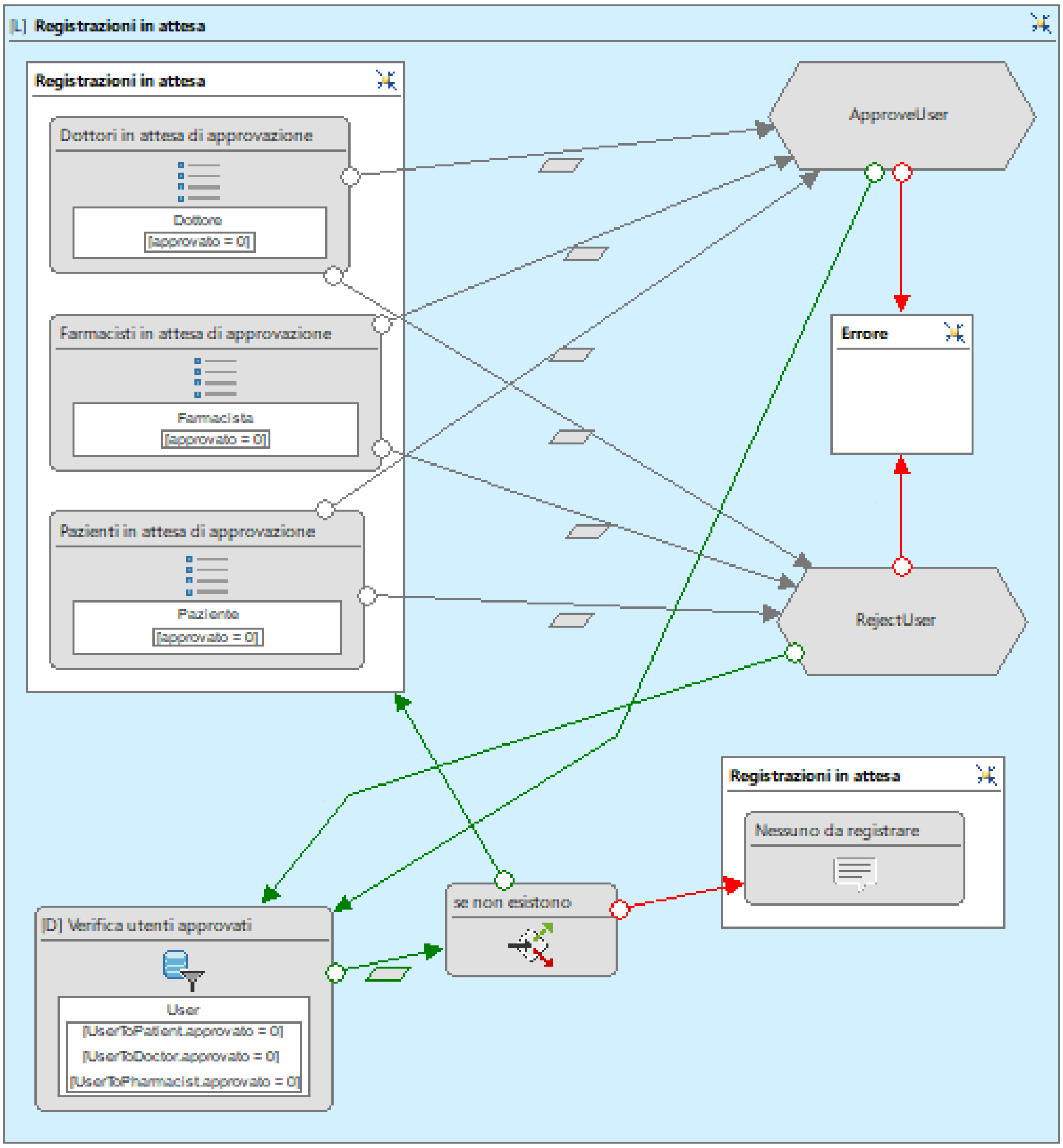

[L] Registrazioni in attesa
Registrazioni in attesa
Dottori in attesa di approvazione
Dottore
[approvato = 0]
Farmacisti in attesa di approvazione
Farmacista
[approvato = 0]
Pazienti in attesa di approvazione
Paziente
[approvato = 0]
ApproveUser
Errore
RejectUser
Registrazioni in attesa
Nessuno da registrare
se non esistono
[D] Verifica utenti approvati
User
[UserToPatient.approvato = 0]
[UserToDoctor.approvato = 0]
[UserToPhamacist.approvato = 0]

Figure 3.38: Users awaiting approval.

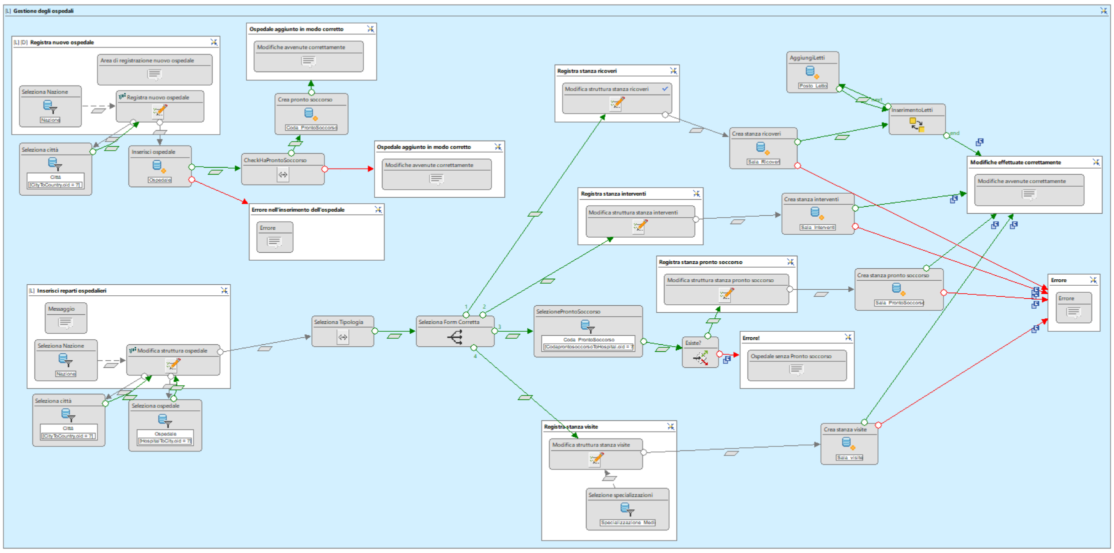

[L] Gestione degli ospedali
[L] [D] Registra nuovo ospedale
Area di registrazione nuovo ospedale
Seleziona Nazione
Nazione
Registra nuovo ospedale
Seleziona città
Città
[CityToCountry.oid = ?]
Inserisci ospedale
Ospedale
Ospedale aggiunto in modo corretto
Modifiche avvenute correttamente
Crea pronto soccorso
Coda_ProntoSoccorso
CheckHaProntoSoccorso
Ospedale aggiunto in modo corretto
Modifiche avvenute correttamente
Errore nell'inserimento dell'ospedale
Errore
Registra stanza ricoveri
Modifica struttura stanza ricoveri
AggiungiLetti
Posto_Letto
next
InserimentoLetti
end
Crea stanza ricoveri
Sala_Ricoveri
Modifiche effettuate correttamente
Modifiche avvenute correttamente
Registra stanza interventi
Modifica struttura stanza interventi
Crea stanza interventi
Sala_Interventi
Registra stanza pronto soccorso
Modifica struttura stanza pronto soccorso
Crea stanza pronto soccorso
Sala_ProntoSoccorso
Errore
Errore
[L] Inserisci reparti ospedalieri
Messaggio
Seleziona Nazione
Nazione
Modifica struttura ospedale
Seleziona città
Città
[CityToCountry.oid = ?]
Seleziona ospedale
Ospedale
[HospitalToCity.oid = ?]
Seleziona Tipologia
Seleziona Form Corretta
1
2
3
4
SelezioneProntoSoccorso
Coda_ProntoSoccorso
[CodaprontosoccorsoToHospital.oid = ?]
Esiste?
Errore!
Ospedale senza Pronto soccorso
Registra stanza visite
Modifica struttura stanza visite
Selezione specializzazioni
Specializzazione_Medi
Crea stanza visite
Sala_visite

Figure 3.39: Implementation of hospital management.

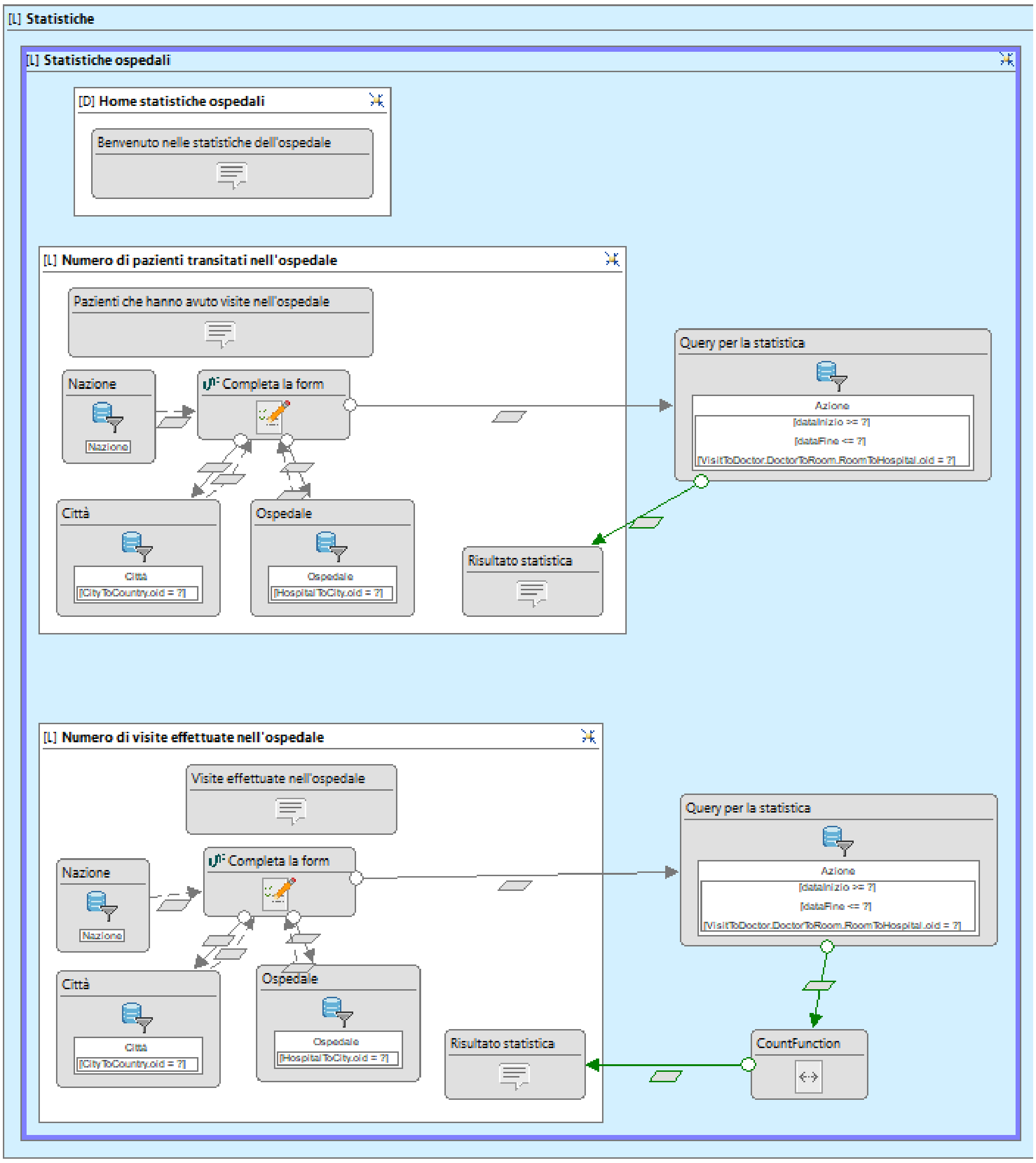


Figure 3.40: Implementation of statistics.

3.2.6.1 | UC - Management of pending recordings

a system administrator can approve recordings previously made by patients, doctors and pharmacists. The system returns three lists (one for doctors, one for pharmacists and one for patients) containing some data of users who have made a registration and

from here, the administrator can decide who accept or reject. The selected user will be removed from the correct list (based on his role) and will be notified of the operation chosen by the administrator by email. If a user is rejected by the administrator and has only one role (doctor or patient or pharmacist) the data will also be removed from the user table in the database.
Some constraints implemented are:

- in the list are shown only the patients or the doctors or the pharmacists that have the attribute *'approved'* equal to zero.

3.2.6.2 | UC - Hospital Management: Register new hospital

A system administrator can register a new hospital entering in a form some data like the country, city, name, address, website and telephone and if has the emergency room.

3.2.6.3 | UC - Hospital Management: Enter hospital wards

A system administrator can register new hospital room of an hospital already present in the database selecting it from a form and the type of room that he wants to add that are *'Stanza visite'*, *'Stanza ricoveri'*, *'Stanza pronto soccorso'*, *'Stanza interventi'*. If the room is of type *Stanza visite*, the next form will asks to administrator to enter the medical specialization of the room, the name and the floor. If the room is of type *Stanza ricoveri*, the next form will asks to administrator to enter the name, the floor inside the hospital and the number of beds to place inside. If the room is of type *Stanza pronto soccorso* or *Stanza interventi*, the next form will asks to administrator to enter only the name and the floor of the room inside the hospital. The system will update the selected hospital inserting a new room to it.
Some constraints implemented are:

- the administrator can register a room of type 'Stanza pronto soccorso' only if the hospital selected has the emergency room;

- the number of beds must be greater than zero.

3.2.6.4 | UC - Pharmacy Management: Register new pharmacy

A system administrator can register a new pharmacy entering in a form some data like the country, city, name, address, website and telephone.

3.2.6.5 | UC - Pharmacy Management: Add drug

A system administrator can register a new drug entering in a form some data like the name, if the drug require a prescription or not and optionally, it is possible to insert one or more drug descriptions to this drug.

3.2.6.6 | UC - Viewing statistics

A system administrator can view some statistics of the system like:

- drugs sold in a particular pharmacy during a specific time period.
- visits to a hospital during a specific time period.

3.2.6.7 | UC - F.A.Q management: Insert new FAQ

A system administrator can register a new FAQ entering in a form the question and the answer.

3.2.6.8 | UC - F.A.Q management: Update FAQ

A system administrator can update a FAQ already registered selecting it from a list and changing the answer related to the question.

3.2.6.9 | UC - F.A.Q management: Remove FAQ

A system administrator can remove a FAQ already registered selecting it from a list.

4

Conclusions and future developments

4.1 | Conclusions

The aim of the project is to analyze a case study of a real system and using a different implementation approach. We became familiar with the IFML (Interaction Flow Modeling Language) in this way. The initial approach was a bit complicated as we had to understand how to use this type of development. Afterwards, moving forward with the work, it was easier to implement the following features.

4.2 | Future developments

In the future it would be interesting to add an interactive map for each hospital. Furthermore, to make the system more usable it would be useful to be able to develop the mobile version.