DATA BASE FOR MEDICAL RECIPIES

Requirements elicitation (Use cases; Use case diagram; Requirements list; Traceability matrix)

TARGETS:

1) Pharmacists
2) Patients
3) Laboratory
4) User

TEMPLATE:

As a Pharmacist I want to be able to have a register where each medical batch is assigned

1. USE CASES:

USE CASE 1; Log in in the database

Actor: User (Patient and Pharmacist)

Goal: The user can log into the app, using its username and password.

Description: Once the user has logged into the app, it will be redirected to the correspondent menu, with the available options for patient or pharmacist.

Preconditions: -

Standard path:

1. Patient logs in.

Alternative scenario:

- The user is not registered. The user and password combination are not correct.

Postcondition:

- The patient is logged in.

USE CASE 2; Sign up in the database

Actor: User

Goal: A new user will create a user profile, with a username and a password. It must choose his role.

Description: Once the user has been signed up in the app, the user can log in into the app.

Preconditions: The username is not already in the database.

Standard path:

1. Patient signs up.

Alternative scenario:

- The patient username is already in the database.

Postcondition:

- The patient is signed up.

USE CASE 3; Identify the patient

Actor: Pharmacist

Goal: Get a specific patient info, in order to access his information.

Description: The pharmacist accesses the information of the patient thanks to its name and id.

Preconditions: The patient its registered in the database.

Standard path:

1. A user with a pharmacist role logs into the app (USE CASE 1).
2. The pharmacist selects the identify patient option.

Alternative scenario:

- The patient is not in the database.

Postcondition:

- The pharmacist can check information about the patient.

USE CASE 4; Check stock

Actor: Pharmacist

Goal: Check the amount of stock of a specific medicine.

Description: The pharmacist can check the stock of a medicine, by providing the medicine's id.

Preconditions: -

Standard path:

1. A user is logged in as a pharmacist (USE CASE 1).
2. The pharmacist selects the option check stock.
3. The pharmacist provides the medicine id.

Alternative scenario:

- The medicine is not in the database.

Postcondition:

- The pharmacist can order new stock if needed.

USE CASE 5; Order new stock

Actor: Pharmacist

Goal: Order more medicine in order to increase the stock.

Description: The pharmacist after checking the stock of a medicine, orders more stock.

Preconditions: The medicine is in the database.

Standard path:

1. A user is logged in as a pharmacist (USE CASE 1).
2. The pharmacist selects the option check stock (USE CASE 4).
3. The pharmacist provides the medicine id.
4. The pharmacist orders new stock.

Alternative scenario:

- The medicine is not in the database.

Postcondition:

- The pharmacist has order new stock, and the amount of medicine has been increased.

USE CASE 6; Mark prescription as used

Actor: Pharmacist

Goal: Mark the prescription as used once the medicine has been given to the patient.

Description: The pharmacist marks prescription as used, selected by its id, and updated by updating the issue date.

Preconditions: The prescription has not been used.

Standard path:

1. A user is logged in as a pharmacist.
2. The pharmacist identifies the patient (USE CASE 3).
3. The pharmacist selects the prescription to be checked as used.

Alternative scenario:

- The patient has not prescriptions.

Postcondition:

- The prescription has been marked as used.

USE CASE 7; Check authenticity

Actor: Pharmacist

Goal: Make sure the prescription has not already been used.

Description: The pharmacist can verify if the prescription can be used, by typing its id.

Preconditions: The patient has a prescription.

Standard path:

1. A user is logged in as a pharmacist.
2. The pharmacist identifies the patient (USE CASE 3).
3. The pharmacist selects the prescription to check its authenticity.

Alternative scenario:

- The patient has not prescriptions.

Postcondition:

- The pharmacist checks if the prescription is valid or not.

USE CASE 8; Check medicine

Actor: Patient

Goal: A user(patient) its able to check all his medications.

Description: The patient can check all the medications he has assigned.

Preconditions: The patient has a medicine assigned.

Standard path:

1. A user is logged in as a patient (USE CASE 1)
2. The patient selects the option search medicine.
3.

Alternative scenario:

- The patient has no medicines assigned.

Postcondition:

- The patient has a list of all his medicines.

USE CASE 9; Check medical history

Actor: Patient

Goal: A user(patient) its able to check all his medical history.

Description: The patient can check all the prescription it has had assigned.

Preconditions: The patient has medical history (previous prescriptions).

Standard path:

1. A user is logged in as a patient (USE CASE 1)
2. The patient selects the option check medical history.

Alternative scenario:

- The patient has no medical history.

Postcondition:

- The patient gets the information about all his medical history

USE CASE 10; Sell medicine

Actor: Pharmacist

Goal: A specific amount of medicine of a pharmacy gets sold to the patient

Description: The pharmacist sells to the patient the medicine on his prescription

Preconditions:

- The patient has a valid prescription
- There are medicines in the prescription
- In the pharmacy there is stock of that medicine

Standard path:

1. Pharmacist checks the authenticity of the prescription
2. Pharmacist checks the stock of the medicine in the prescription
3. Pharmacist gives the medicine to the patient

Alternative scenario:

- The patient has no medicines assigned on his prescription.

Postcondition:

- The patient has his medicines
- The stock of that medicine reduces.

USE CASE 11; Reduce Stock

Actor: Pharmacist

Goal: The stock of a medicine gets reduced after it has been sold.

Description: After selling a medicine, the stock of it gets reduced by the specific number of medicines that have been sold to the patient.

Preconditions:

- A medicine has been sold

Standard path:

1. A medicine gets sold
2. The stock of that medicine gets reduced

Alternative scenario:

- No medicines have been sold

Postcondition:

- The stock of the medicine that has been sold is reduced and therefore is lower than it was before.

USE CASE 12; Insert prescription

Actor: Patient

Goal: The prescription of a patient gets registered in the data base.

Description: After the doctor (not in our data base) has given the patient a prescription, he is able to insert it into the data base.

Preconditions:

- The patient has gone to the doctor and has a prescription.
- The patient has logged in as a patient

Standard path:

1. A user is logged in as a patient
2. The user selects the option insert prescription

Alternative scenario:

- The patient has no prescriptions

Postcondition:

- The prescription is uploaded into the data base

2. REQUIREMENTS LIST:

Functional requirements:

1. The patient should be able to access his medical information on an app
2. The system should allow the pharmacist to access the patient profile at the data base
3. Checking the stock of each medicine
4. Ordering new stock in case of low stock
5. The system must keep a record of the prescriptions that have been used
6. The system should be able to recognize when a prescription is valid or not
7. The data base must keep track of the medicines that have been sold

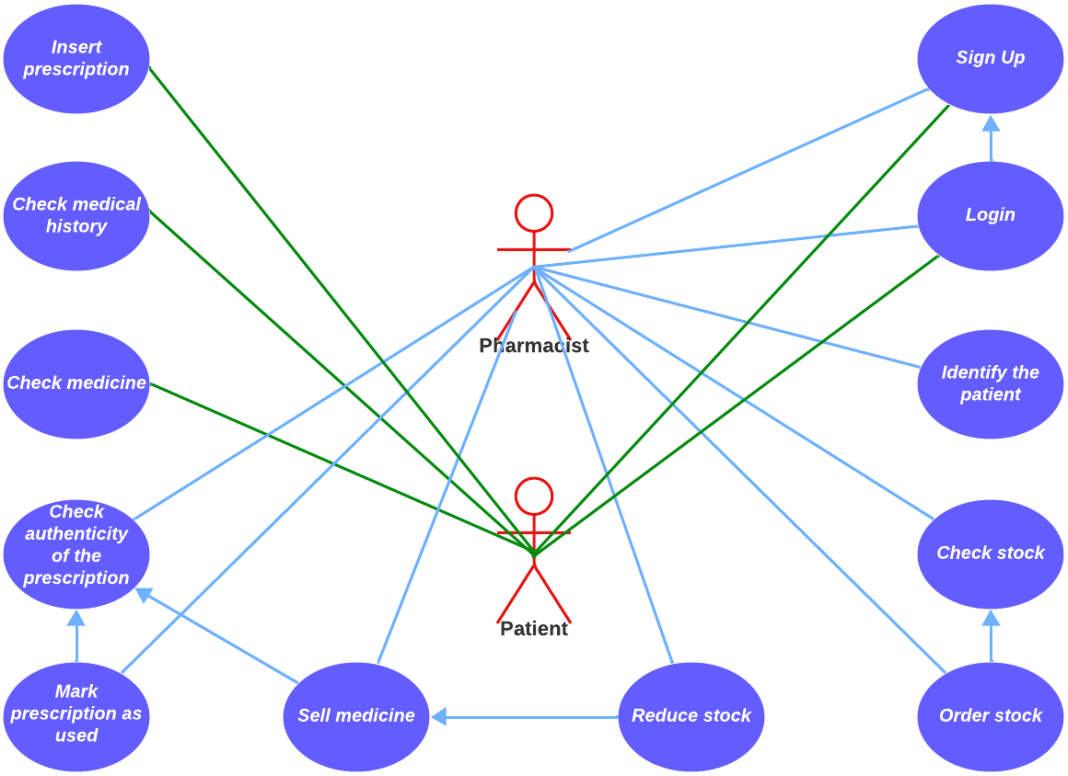
Non-functional requirements:

1. The system is programmed in java
2. The system must run in Linux and windows
3. The patient can be assigned with more than one medicine

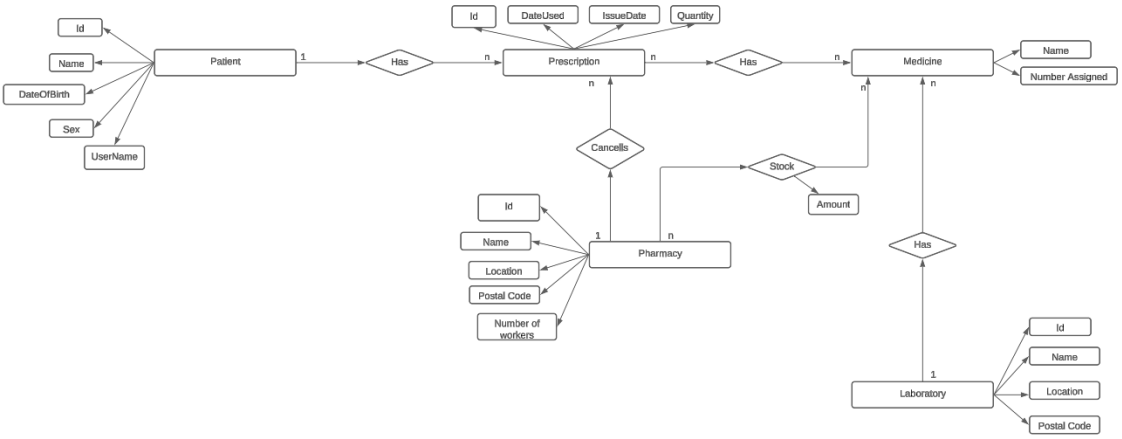
3. TRACEABILITY MATRIX

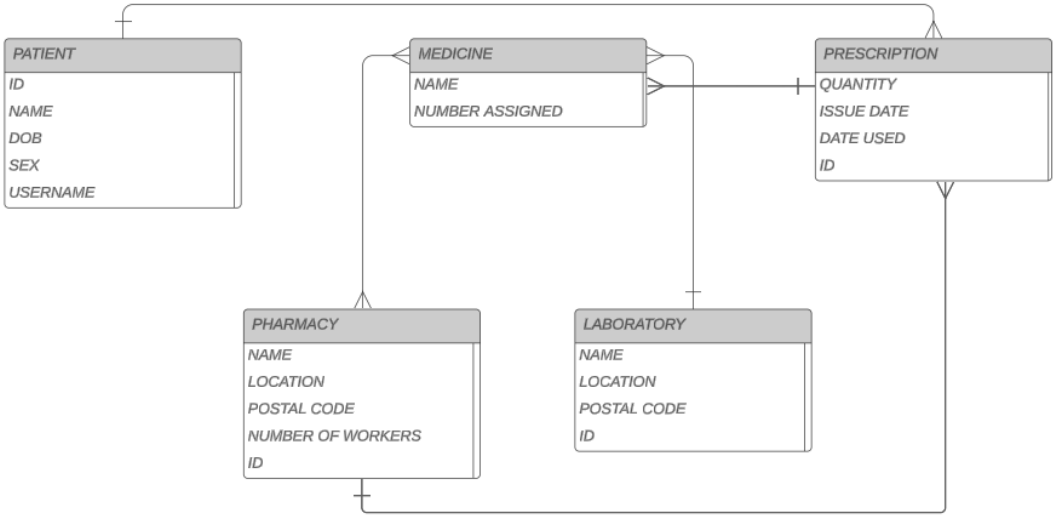
	Req1	Req2	Req3	Req4	Req5	Req6	Req7	NFReq 1	NFReq 2	NFReq 3
UC1	x	x	x	x						
UC2	x	x								
UC3	x	x			x	x				
UC4			x	x			x			
UC5			x	x			x			
UC6		x			x	x	x			
UC7					x	x				x
UC8	x	x			x					x
UC9	x	x			x					x
UC10		x	x	x		x	x			x
UC11			x				x			
UC12	x	x			x	x				

4. USE CASE DIAGRAM

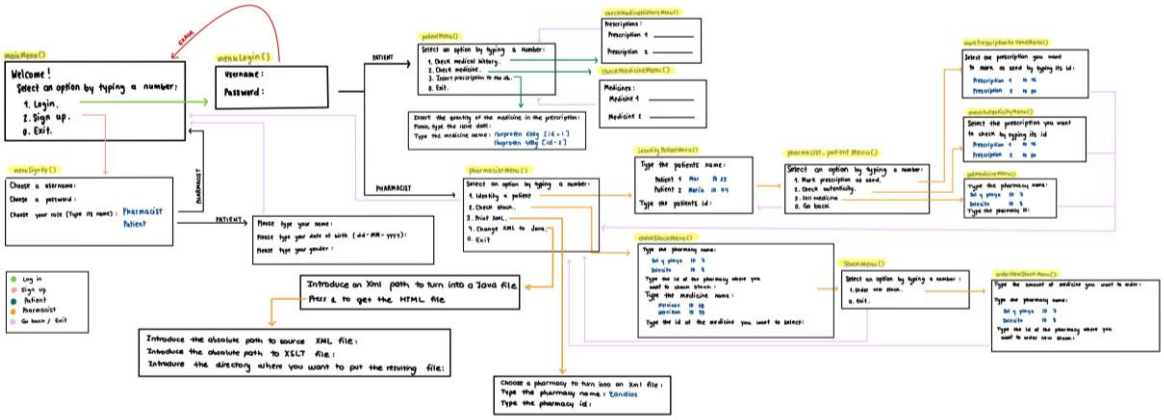


5. ER DIAGRAM


Id
Name
DateOfBirth
Sex
UserName
Patient
1
Has
n
Id
DateUsed
IssueDate
Quantity
Prescription
n
Has
n
Medicine
Name
Number Assigned
n
n
Cancells
Stock
Amount
Id
Name
Location
Postal Code
Number of workers
1
n
Pharmacy
Has
1
Laboratory
Id
Name
Location
Postal Code


PATIENT
ID
NAME
DOB
SEX
USERNAME
MEDICINE
NAME
NUMBER ASSIGNED
PRESCRIPTION
QUANTITY
ISSUE DATE
DATE USED
ID
PHARMACY
NAME
LOCATION
POSTAL CODE
NUMBER OF WORKERS
ID
LABORATORY
NAME
LOCATION
POSTAL CODE
ID

MOCK UP


mainMenu()
Welcome!
Select an option by typing a number:
1. Login.
2. Sign up.
0. Exit.
menuLogin()
Username:
Password:
menuSignUp()
Choose a username:
Choose a password:
Choose your role (Type its name): Pharmacist Patient
PATIENT
PHARMACIST
Please type your name:
Please type your date of birth (dd-MM-yyyy):
Please type your gender:
patientMenu()
Select an option by typing a number:
1. Check medical history.
2. Check medicine.
3. Insert prescription to make an order.
0. Exit.
checkMedicineHistoryMenu()
Prescription 1
Prescription 2
checkMedicineMenu()
Medicines:
Medicine 1
Medicine 2
Insert the quantity of the medicine in the prescription:
Press, type the word done:
Type the medicine name: Ibuprofen 600g (id:1)
Ibuprofen 400g (id:2)
pharmacistMenu()
Select an option by typing a number:
1. Identify a patient
2. Check stock.
3. Print XML.
4. Change XML to Java.
0. Exit
identifyPatientMenu()
Type the patient name:
Patient 1 Mar 23
Patient 2 Maria 19 04
Type the patient id:
pharmacist_patient Menu()
Select an option by typing a number:
1. Mark prescription as used.
2. Check authenticity.
3. Sell medicine
0. Go back
markAsUsedPrescriptionMenu()
Select an prescription you want to mark as used by typing its id:
Prescription 1
Prescription 2
checkAuthenticityMenu()
Select the prescription you want to check by typing its id
Prescription 1
Prescription 2
getMedicineMenu()
Type the pharmacy name:
Set 4 pharm 19 3
Betsalm 19 3
Type the id of the pharmacy:
checkStockMenu()
Type the pharmacy name:
Set 4 pharm 19 3
Betsalm 19 3
Type the id of the pharmacy where you want to check stock:
Type the medicine name:
Ibuprofen 19 04
Aspirin 19 05
Type the id of the medicine you want to select:
StockMenu()
Select an option by typing a number:
1. Order more stock.
0. Exit.
orderMoreStockMenu()
Type the amount of medicine you want to order:
Type the pharmacy name:
Set 4 pharm 19 3
Betsalm 19 3
Type the id of the pharmacy where you want to order more stock:
Log in
Sign up
Patient
Pharmacist
Go back / Exit
Introduce an Xml path to turn into a Java file.
Press 4 to get the HTML file.
Introduce the absolute path to source XML file:
Introduce the absolute path to XSLT file:
Introduce the directory where you want to put the resulting file:
Choose a pharmacy to turn into an Xml file:
Type the pharmacy name: Condias
Type the pharmacy id:

UML CLASS DIAGRAM