

# Software Requirement Specification Document

Ayman, Belal, Khaled, Karim, Mostafa

March 27, 2017

## 1 Introduction

### 1.1 Purpose of this document

Is identify the project boundaries and describe functionality of project and to the intended audiences and anyone who wants to participate in the future development.

### 1.2 Scope of this document

Provide a short description of the software being specified and its purpose, including relevant benefits, objectives, and goals.

### 1.3 Overview

provide the user with the nearest pharmacies according to the desired medicine and providing the pharmacy an access to the user profile

### 1.4 Business Context

Advertisements  
yearly subscription of pharmacies

## 2 General Description

### 2.1 Product Functions

Our project (ADwytee) helps people to reach the medicine they seek in the fastest way possible, help pharmacists to find more customers by connecting as many as pharmacies to our database server, help the customer to find the nearest pharmacy has the desired medicine and find more info about the medicine , the pharmacy ,and help pharmacies to keep track of patient history .. etc

## 2.2 Similar System Information

Standalone project

## 2.3 User Characteristics

Normal people ,pharmacies and companies(normal users) developer(we will provide an API that allows the developers to create applications that it compatible with our servers for future developments)

## 2.4 User Problem Statement

Medicine lack, high search time for medicine , the difficulty of finding alternatives for certain medicine

## 2.5 User Objectives

Find medicine in nearest pharmacy with highest precision with least effort and possible alternatives

## 2.6 General Constraints

Provide easy user interface with least complexity provide API compatible with multiple platforms

# 3 Functional Requirements

	Functional Requirement
Code	AL1
Name	Register
Type	Functional requirement
Criticality	High
Input	Guest info
Output	Boolean value accepted or rejected
Description	User should fill some data such as:user name,password and e-mail and An activation message will be sent to user's e-mail
Priority	10/10
Risks	User may Spam or send unwanted data
Precondition	Open register page
Post-condition	An activation message will be sent to user's e-mail
Dependency	-----

	Functional Requirement
Code	AL2
Name	Login
Type	Functional requirement
Criticality	High
Input	Login info (email/user name,password)
Output	Boolean value accepted or rejected and id or key
Description	User should fill :user name,password and e-mail and boolean value will be sent to him with his id
Priority	10/10
Risks	User may Spam or send wrong user name and email or pass
Precondition	Open login page
Post-condition	User will log in into his main page as a pharmacy or user
Dependency	Register

	Functional Requirement
Code	AL3
Name	Logout
Type	Functional requirement
Criticality	High
Input	
Output	Message to user that he logged out
Description	If the user wanted to log out from current session the session will close from this browser
Priority	10/10
Risks	
Precondition	User must be registered and logged in
Post-condition	User will logout from website
Dependency	Login

	Functional Requirement
Code	AL4
Name	Search
Type	Functional requirement
Criticality	High
Input	Medicine name/location
Output	List of pharmacies with searched medicine from near to far
Description	The user will search desired medicine while inputing his location and medicine name and he will be able to see where is the nearest pharmacy with desired medicine and will return empty list if medicine isn't found
Priority	10/10
Risks	The search will consume a lot of time and be useless
Precondition	Open search page
Post-condition	All pharmacies which have the medicine will be appear
Dependency	Choose location

	Functional Requirement
Code	AL5
Name	View alternatives
Type	Functional requirement
Criticality	Medium
Input	Medicine name
Output	List of medicine alternatives
Description	The user will search desired medicine the alternative medicines will appear in list beside his search
Priority	7/10
Risks	The search for alternatives will consume a lot of time and be useless no registered medicine alternatives by pharmacy
Precondition	
Post-condition	All medicine alternatives will appear with pharmacy name that suggested alternatives
Dependency	Search medicine

	Functional Requirement
Code	AL6
Name	Choose location
Type	Functional requirement
Criticality	High
Input	Location in map
Output	
Description	The user will choose location
Priority	10/10
Risks	User can choose location wrong
Precondition	User must be registered and logged in
Post-condition	Identify user location correctly
Dependency	

	Functional Requirement
Code	U7
Name	Track
Type	Functional requirement
Criticality	Medium
Input	Name of not found medicine
Output	Notification sent to user when tracked medicine is in pharmacy
Description	The user if he can't find medicine he will be able to track medicine and it will send him an email when medicine is available
Priority	6/10
Risks	The medicine is no longer produced so large waiting time for user
Precondition	User must be registered and logged in
Post-condition	A notification will be appeared in user's profile informing him about the details of the medicine and in which pharmacy it's found or sending user an email
Dependency	Register /login

	Functional Requirement
Code	AL8
Name	Select pharmacy
Type	Functional requirement
Criticality	High
Input	Name of pharmacy
Output	Info about pharmacy and order option
Description	The user will search the medicine he desire then select pharmacy from list of pharmacies and then he can view info about this pharmacies
Priority	9/10
Risks	No info registered by pharmacy
Precondition	User must be registered and logged in and searched for <u>medicine</u>
Post-condition	Pharmacy page
Dependency	Accept pharmacies register

	Functional Requirement
Code	A1
Name	Accept pharmacies register
Type	Functional requirement
Criticality	High
Input	Name of pharmacy and info
Output	Boolean with <u>admin</u> decision
Description	The <u>admin</u> will decide neither to accept or decline the register
Priority	10/10
Risks	
Precondition	The pharmacy was registered
Post-condition	Create account to pharmacy
Dependency	Register

	Functional Requirement
Code	A2
Name	Delete pharmacies
Type	Functional requirement
Criticality	High
Input	Name of pharmacy and info
Output	
Description	The <u>admin</u> may delete pharmacy from database
Priority	10/10
Risks	
Precondition	Pharmacy has account
Post-condition	The pharmacy is informed with <u>admin</u> decision
Dependency	Accept pharmacies register

	Functional Requirement
Code	A3
Name	Hung pharmacies
Type	Functional requirement
Criticality	High
Input	Name of pharmacy and info
Output	
Description	The <u>admin</u> may hung pharmacy activities
Priority	10/10
Risks	
Precondition	Open page control
Post-condition	The pharmacy is informed with <u>admin</u> decision
Dependency	Accept pharmacies register

	Functional Requirement
Code	A4
Name	<u>Un</u> hung pharmacies
Type	Functional requirement
Criticality	High
Input	Name of pharmacy and info
Output	
Description	The <u>admin</u> may <u>Un</u> hung pharmacy and re support pharmacy with our service
Priority	10/10
Risks	
Precondition	Open page control
Post-condition	The pharmacy is informed with <u>admin</u> decision
Dependency	Accept pharmacies register

	Functional Requirement
Code	A5
Name	Show control page
Type	Functional requirement
Criticality	High
Input	.....
Output	List of pharmacies with status (hung or <u>un</u> hung) with show report option
Description	The <u>admin</u> may order a list of pharmacies to show status
Priority	10/10
Risks	.....
Precondition	<u>admin</u> must be registered and logged in
Post-condition	List of Pharmacies and their status will appear
Dependency	Open page control

	Functional Requirement
Code	A6
Name	Show Report
Type	Functional requirement
Criticality	Medium
Input	-----
Output	List of orders in every region is displayed with most region and least region of orders
Description	The <u>admin</u> may show report to view sales activities
Priority	6/10
Risks	-----
Precondition	<u>admin</u> must be registered and logged in
Post-condition	List of orders in every region is displayed with most region and least region of orders
Dependency	Register /login

	Functional Requirement
Code	U1
Name	Rate
Type	Functional requirement
Criticality	High
Input	Rate value and pharmacy name
Output	Boolean value
Description	The user will be able to rate the pharmacy after he ordered
Priority	8/10
Risks	-----
Precondition	User must be registered and logged in
Post-condition	The rate will be displayed
Dependency	Register

	Functional Requirement
Code	U2
Name	Set schedule
Type	Functional requirement
Criticality	High
Input	The day and medicine
Output	Return notification to user
Description	The user will be able to set seclude of his medicine and server will order automatically the medicine he need
Priority	8/10
Risks	-----
Precondition	User must be registered and logged in
Post-condition	The schedule will be displayed
Dependency	view user profile

	Functional Requirement
Code	U3
Name	View profile
Type	Functional requirement
Criticality	High
Input	The id of user
Output	
Description	The user will be able to view their profile according to their identity
Priority	8/10
Risks	
Precondition	User must be registered and logged in
Post-condition	The profile will be displayed
Dependency	Login

	Functional Requirement
Code	U4
Name	Order medicine
Type	Functional requirement
Criticality	High
Input	The name of medicine the no of packets
Output	Boolean value neither order is posted or not
Description	The user will be able to order the desired medicine after searching and selecting pharmacies
Priority	8/10
Risks	Pharmacies doesn't support ordering
Precondition	medicine search pharmacy select
Post-condition	The order notification will be sent to pharmacy
Dependency	pharmacy select

	Functional Requirement
Code	U5
Name	Edit Order
Type	Functional requirement
Criticality	High
Input	The name of medicine the no of packets and id of order
Output	Boolean value neither order is editable or not
Description	The user will be able to edit the order the desired medicine after searching and selecting pharmacies
Priority	8/10
Risks	Pharmacies doesn't support ordering or exceed the time limit
Precondition	select order
Post-condition	The User new order will be displayed and notification will be sent to pharmacy
Dependency	Order medicine



	Functional Requirement
Code	U6
Name	delete Order
Type	Functional requirement
Criticality	High
Input	The id of order
Output	Boolean value neither order is deleted or not
Description	The user will be able to delete order the desired medicine after searching and selecting pharmacies
Priority	8/10
Risks	exceed the time limit of edit or delete
Precondition	select order
Post-condition	The User new order will be deleted and notification will be sent to pharmacy
Dependency	Order medicine

	Functional Requirement
Code	P1
Name	Add medicine
Type	Functional requirement
Criticality	Medium
Input	Medicine <u>barcode</u>
Output	Boolean value neither medicine is added or not
Description	The pharmacy will be able to add medicines to their database if they are on system
Priority	7/10
Risks	Hosting issue due to large data size
Precondition	Login
Post-condition	The medicine will be added in their database and will searched by user
Dependency	Register medicine

	Functional Requirement
Code	P2
Name	remove medicine
Type	Functional requirement
Criticality	Medium
Input	Medicine name
Output	Boolean value neither medicine is deleted or not
Description	The pharmacy will be able to delete medicine from their database
Priority	7/10
Risks	.....
Precondition	Medicine was added
Post-condition	The medicine will be removed from their DB
Dependency	Order medicine

	Functional Requirement
Code	P3
Name	Register medicine
Type	Functional requirement
Criticality	High
Input	Medicine name and medicine info
Output	Boolean value neither medicine is Registered or not
Description	The pharmacy will be able to register new medicine and add them to their Db
Priority	10/10
Risks	Hosting issue due to large data size
Precondition	logged in
Post-condition	The medicine will be registered to system and available in this pharmacy
Dependency	Accept register pharmacy

	Functional Requirement
Code	P4
Name	Set alternatives
Type	Functional requirement
Criticality	Low
Input	Medicine name and medicine info and main medicine
Output	Boolean value neither medicine has added an alternative or not
Description	The pharmacy will be able to set alternatives to a <u>medicine</u>
Priority	4/10
Risks	Hosting issue due to large data size
Precondition	the medicine in Database
Post-condition	The alternative is added to medicine
Dependency	Register medicine

	Functional Requirement
Code	P5
Name	Add expire date
Type	Functional requirement
Criticality	Low
Input	Medicine name and expire date
Output	Boolean value neither expire date is added or not
Description	The pharmacy will be able to track their medicines expire date
Priority	5/10
Risks	*****
Precondition	the medicine was added
Post-condition	Send notification to pharmacy when the medicine is expired
Dependency	Added medicine

	Functional Requirement
Code	P6
Name	Bind User
Type	Functional requirement
Criticality	Medium
Input	Patient key
Output	Boolean value neither patient is added or not
Description	The pharmacy will be able to bind their customer with their keys
Priority	7/10
Risks	-----
Precondition	logged in
Post-condition	The pharmacy will be able to edit user history
Dependency	View user profile

	Functional Requirement
Code	P7
Name	Edit User history
Type	Functional requirement
Criticality	High
Input	Patient key and pharmacy password
Output	Patient history
Description	The pharmacy will be able to change,view and edit patient history
Priority	9/10
Risks	-----
Precondition	View user profile
Post-condition	The pharmacy will be able to view user history page and change it
Dependency	Bind user

	Functional Requirement
Code	P8
Name	Accept order
Type	Functional requirement
Criticality	Medium
Input	Boolean neither accept nor decline
Output	-----
Description	The pharmacy will accept the order
Priority	6/10
Risks	-----
Precondition	Open order page
Post-condition	Notification will be sent to User who ordered the order
Dependency	Order medicine

	Functional Requirement
Code	P9
Name	Decline order
Type	Functional requirement
Criticality	Medium
Input	Boolean neither decline nor not
Output	-----
Description	The pharmacy will decline the order
Priority	6/10
Risks	-----
Precondition	already the medicine ordered
Post-condition	Notification will be sent to User who ordered the order
Dependency	Order medicine

	Functional Requirement
Code	S1
Name	Warning expire
Type	Functional requirement
Criticality	Low
Input	The expire date of medicine
Output	-----
Description	The server will send notification to pharmacy when medicine expires
Priority	3/10
Risks	-----
Precondition	Medicine expired in DB The pharmacy is logged in not in hung mode
Post-condition	Notification will be sent to pharmacy
Dependency	Expire time

	Functional Requirement
Code	S2
Name	Hang pharmacy
Type	Functional requirement
Criticality	Low
Input	Date of end of subscription
Output	-----
Description	The server will send notification to pharmacy when subscription expires
Priority	3/10
Risks	-----
Precondition	No resubscribe done by pharmacy
Post-condition	Notification will be sent to pharmacy
Dependency	Pharmacy subscription time

	Functional Requirement
Code	S3
Name	Notify for schedule
Type	Functional requirement
Criticality	Low
Input	Date of schedule
Output	-----
Description	The server will send notification to pharmacy when subscription expires
Priority	3/10
Risks	-----
Precondition	Time of medicine in schedule has come
Post-condition	Send notification to user and ask for his permission to order
Dependency	User schedule

## 4 Interface Requirements

Our software will provide an API so it will provide an access for any other software development

### 4.1 User Interfaces

The interface with user will be easy and simple to insure that he will find the medicine

#### 4.1.1 GUI

This is the home page, It can navigate you to register page or to contact us, or even you can search for a desired medicine and get the nearest pharmacies that having the searched medicine.



#### 4.1.2 CLI

...

#### 4.1.3 API

#### 4.1.4 Diagnostics or ROM

Apache web server

### 4.2 Hardware Interfaces

User may use Barcode reader for easier accessibility.

### 4.3 Communications Interfaces

Will mainly relay on network connection

### 4.4 Software Interfaces

...

## 5 Performance Requirements

Provide search results with max delay 0.2 sec and provide the pharmacy location with the highest precision with radius of default value 2.5km.

## 6 Design Constraints

Must be easy to use interface with simple ads included.

### 6.1 Standards Compliance

### 6.2 Hardware Limitations

### 6.3 others as appropriate

## 7 Other non-functional attributes

### 7.1 Security

Non-functional requirements/NFR.11	
<b>Code</b>	NFR.11
<b>Name</b>	Encrypted passwords
<b>Type</b>	Security Requirement
<b>Description</b>	All passwords will be hashed in the Database
<b>Priority</b>	10/10

## 7.2 Binary Compatibility

Non-functional requirements/NFR.04	
<b>Code</b>	NFR.04
<b>Name</b>	Compatible Devices
<b>Type</b>	Compatibility Requirement
<b>Description</b>	The system will be compatible with all web browsers, computers and mobile phones with our API
<b>Priority</b>	8/10

## 7.3 Reliability

Non-functional requirements/NFR.03	
<b>Code</b>	NFR.03
<b>Name</b>	Availability
<b>Type</b>	Reliability Requirement
<b>Description</b>	The system will be available all the time, with no crash
<b>Priority</b>	9/10

## 7.4 Maintainability

Non-functional requirements/NFR.06	
<b>Code</b>	NFR.06
<b>Name</b>	Service
<b>Type</b>	Maintainability Requirement
<b>Description</b>	The system will be maintainable, any known defect will be able to be corrected
<b>Priority</b>	10/10

## 7.5 Portability

Non-functional requirements/NFR.02	
<b>Code</b>	NFR.02
<b>Name</b>	Portable System
<b>Type</b>	Portability Requirement
<b>Description</b>	The system will run on all computers with any web browser
<b>Priority</b>	7/10

## 7.6 Re-usability

Non-functional requirements/NFR.10	
<b>Code</b>	NFR.10
<b>Name</b>	Familiar Interface
<b>Type</b>	Usability Requirement
<b>Description</b>	The UI must be user-friendly
<b>Priority</b>	10/10

## 7.7 Serviceability

Non-functional requirements/NFR.08	
<b>Code</b>	NFR.08
<b>Name</b>	Test use case
<b>Type</b>	Testability Requirement
<b>Description</b>	All major use cases must be tested well
<b>Priority</b>	10/10

## 7.8 others as appropriate

Non-functional requirements/NFR.07	
<b>Code</b>	NFR.07
<b>Name</b>	Performance
<b>Type</b>	Response Time Requirement
<b>Description</b>	Query response time must be fast, all queries must return a response in < 0.2 seconds
<b>Priority</b>	8/10

Non-functional requirements/NFR.05	
<b>Code</b>	NFR.05
<b>Name</b>	Language Switch
<b>Type</b>	Localizability Requirement
<b>Description</b>	It's possible to operate the system in different language, User can switch between English and Arabic
<b>Priority</b>	10/10

Non-functional requirements/NFR.01	
<b>Code</b>	NFR.01
<b>Name</b>	Accurate location
<b>Type</b>	Accuracy Requirement
<b>Description</b>	It's necessary to get the pharmacies and users location accurate
<b>Priority</b>	10/10

Non-functional requirements/NFR.09	
<b>Code</b>	NFR.09
<b>Name</b>	System understandable
<b>Type</b>	Understandability Requirement
<b>Description</b>	The system must be easy to understand and easy to use
<b>Priority</b>	10/10

## 8 Preliminary Object-Oriented Domain Analysis

...



## 8.1 Inheritance Relationships

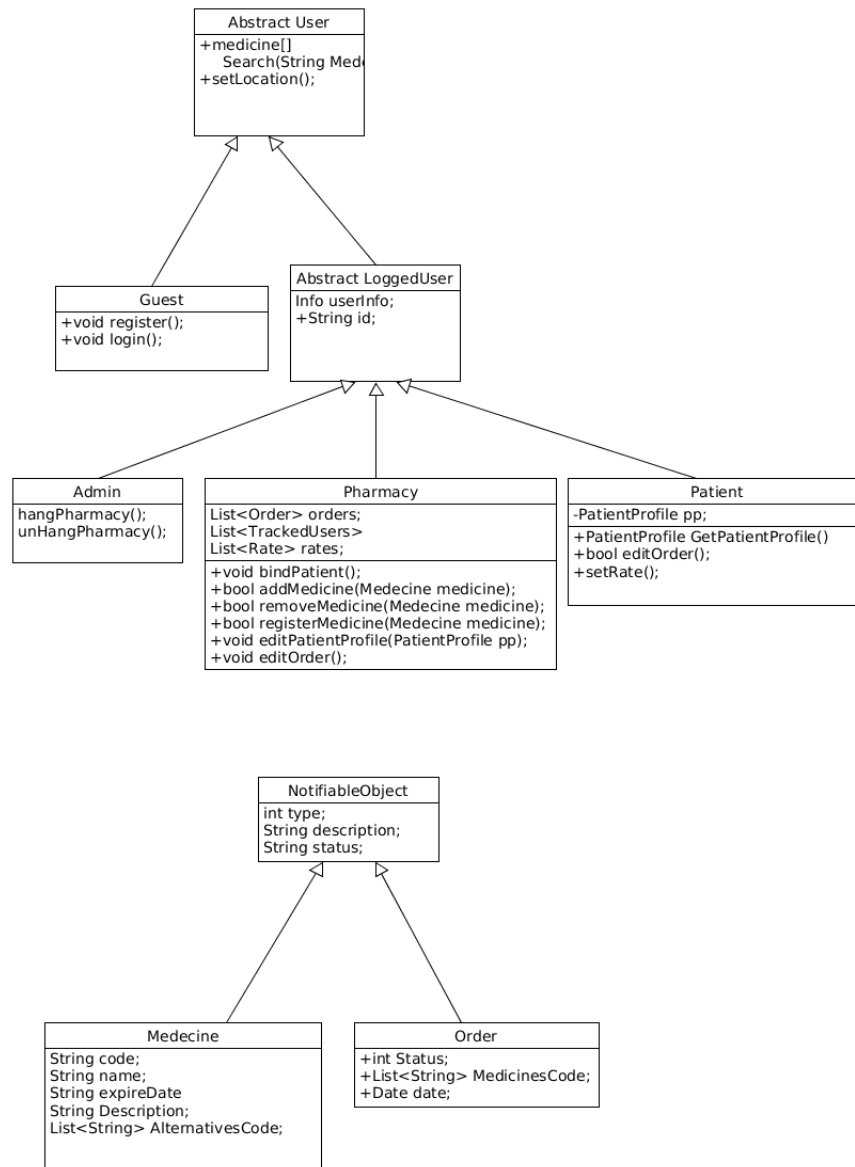


Figure 1: Inheritance Relations

## 8.2 Class descriptions

User	
Name	Abstract
Super classes	-
Sub classes	Guest, LoggedUser
purpose	Provide the main functionality for all users like searching
Collaborations	InteractionManager
Attributes	-
Operations	Search, and set the location
constrains	-

Guest	
Name	Concrete
Super classes	User
Sub classes	-
purpose	It has a limited access to some features till he registers or login
Collaborations	LoginInfo, RegisterInfo
Attributes	-
Operations	Void Register() ,void login()
constrains	Cant register as pharmacy till he provide a valid credit card

LoggedUser	
Name	Abstract
Super classes	User
Sub classes	Admin, Pharmacy, Patient
purpose	provide the basic info for a logged User
Collaborations	-
Attributes	UserInfo ui,string ID
Operations	-
constrains	-

Patient	
Name	Concrete
Super classes	LoggedUser
Sub classes	-
purpose	Has the main functionality of the patient and interaction with the system
Collaborations	Medicine, Order, PatientProfile
Attributes	PatientProfile Pp
Operations	Edit order ,set rate, track a medicine
constrains	-

PatientProfile	
Name	Concrete
Super classes	-
Sub classes	-
purpose	Has the main info pharmacy need about the user
Collaborations	Transactions, Allergy, BloodPressure
Attributes	Int Weight, int height,string gender,Chronic chronics,RiskFactor rf, Transaction tr
Operations	-
constrains	-

NotifiableObject	
Name	Abstract
Super classes	-
Sub classes	Medicine, Order
purpose	Has the main info a notification require to provide info about certain object can be send in notification
Collaborations	-
Attributes	Type, description, status
Operations	-
constrains	-

notification	
Name	Concrete
Super classes	-
Sub classes	-
purpose	This will have a unique id of notification and responsible to deliver it to the desired user
Collaborations	NotifiableObject, Patient
Attributes	String id , int type;
Operations	Notification nf
constrains	-

Medicine	
Name	Concrete
Super classes	NotifiableObject, ApiObject
Sub classes	-
purpose	Has the main info about medicine
Collaborations	Patient, Pharmacy
Attributes	String code, String name, String expireDate, String Description, List<String> Alternatives;
Operations	-
constrains	-

Order	
Name	Concrete
Super classes	-
Sub classes	-
purpose	Contain the main info about certainOrder
Collaborations	Patient, Pharmacy
Attributes	Int status, List<String> mCodes,Date date
Operations	-
constrains	-

Pharmacy	
Name	Concrete
Super classes	LoggedInUser
Sub classes	-
purpose	Provide the functionality of pharmacy allowing it to access patient profile
Collaborations	Patient Profile, Rate, Order
Attributes	List<Order> orders(0,m), List<PatientProfile> trackedPatients(0,m), list<Rate> rates(0,m)
Operations	+void bindPatient(); +bool addMedicine(Medecine medicine); +bool removeMedicine(Medecine medicine); +bool registerMedicine(Medecine medicine); +void editPatientProfile(PatientProfile pp); +void editOrder();
constrains	-

Admin	
Name	Concrete
Super classes	LoggedInUser
Sub classes	-
purpose	Provide the functionality of admin giving him access to unique features
Collaborations	Patient, Pharmacy
Attributes	-
Operations	Void hangPharmacy(); void unHangPharmacy(); void removePharmacy(); void acceptRegesteredPharmacy();
constrains	-

Info	
Name	Concrete
Super classes	-
Sub classes	PharmacyInfo , patientInfo
purpose	Provide the basic info for each basic user
Collaborations	LoggedInUser
Attributes	String name, String key, String teleNo;
Operations	Void hangPharmacy(); void unHangPharmacy(); void removePharmacy(); void acceptRegesteredPharmacy();
constrains	-

PatientInfo	
Name	Concrete
Super classes	Info
Sub classes	-
purpose	Provide the detailed info for Patient
Collaborations	Patient, Locaction
Attributes	Location[] prefLocations(0,m); Medecine[] trackedMedicines(0,m) String BirthDate;
Operations	-
constrains	-

PharmacyInfo	
Name	Concrete
Super classes	Info
Sub classes	-
purpose	Provide the detailed info for Patient
Collaborations	pharmacy
Attributes	Location location; String Note; String Description Patient[] BindedPatients(0,m);
Operations	-
constrains	-

Location	
Name	Concrete
Super classes	-
Sub classes	-
purpose	Provide the detailed info about the exact location
Collaborations	Pharmacyinfo, PatientInfo
Attributes	int X_axis; int Y_axis; String Details;
Operations	-
constrains	-

InteractionManager	
Name	Concrete
Super classes	-
Sub classes	-
purpose	Provide the interactions between the user and the server and acts as mediator between user and ApiManger
Collaborations	ApiManger, DataBaseManger
Attributes	-
Operations	void sendNotification(); Location provideEstimatedLocation()
constrains	-

ApiManager	
Name	Concrete
Super classes	-
Sub classes	-
purpose	Analyze input from more than one platform and respond according to the privilege
Collaborations	DataBaseManger
Attributes	-
Operations	String ManageMedecine(String req); String ManageUser(String req);
constrains	The user should provide a valid Api key whenever he demand a response

ApiObject	
Name	interface
Super classes	-
Sub classes	-
purpose	Provide the functions needed for object to be able to use the ApiManger
Collaborations	-
Attributes	-
Operations	+String ToJsonString()
constrains	

## 9 Operational Scenarios

Use case	register
Actors	Guest
Precondition	open register window
Basic flow	1-fill the data of the registration forum which is: user name ,password , mail , phone 2-click submit 3-check data validation 4-save data in database
Alternative flow	Data is wrong and data not saved
Post condition	1- An activation message will be sent to user's e-mail 2-open profile page

Use case	Login
Actors	Patient pharmacy admin
Precondition	Register open login window
Basic flow	1-enter user name and password 2-click login 3- search for data in database 4- Fetch user key
Alternative flow	User data not found in database
Post condition	A specific web page will be opened according to each actor

Use case	Logout
Actors	patient pharmacy company admin
Precondition	Login
Basic flow	1-open profile page 2-click logout
Alternative flow	
Post condition	The logout will redirect to guest home page

Use case	Search
Actors	User guest pharmacy
Precondition	Open a search page
Basic flow	2-write name of the medicine or bar code of the medicine in text box 3-click search beside the text box
Alternative flow	The bar code of the medicine is wrong or the name is wrong the medicine is not found in any pharmacies
Post condition	All pharmacies which have the medicine will be appear

Use case	Select alternatives
Actors	patient pharmacy
Precondition	search for medicine
Basic flow	1-click on alternative name
Alternative flow	The medicine has no alternative
Post condition	

Use case	Choose location
Actors	Patient guest pharmacy
Precondition	Search for medicine
Basic flow	1-open google map 2- click on your location
Alternative flow	The web app gets the location automatically without user input
Post condition	Identify user location correctly

Use case	Track medicine
Actor	User
Precondition	Login
Basic flow	1-enter the name or the bar code of the most searched medicine 2-click track button
Alternative flow	1-The medicine is not found yet in any pharmacies 2-the name of the medicine is wrong 3-the bar code of the medicine is wrong
Post condition	A notification will be appeared in user's profile informing him about the details of the medicine and in which pharmacy it's found

Use case	Select pharmacy
Actors	User/guest/pharmacy
Precondition	logged in
Basic flow	1-select pharmacy
Alternative flow	The location is not exact the name or the bar code of the medicine is wrong
Post condition	All the pharmacies which have the medicine will be appear and the user will choose

Use case	Accept pharmacies register
Actors	Admin
Precondition	logged in
Basic flow	1-open control page 2-click on accept button
Alternative flow	
Post condition	The pharmacy is informed with admin decision

Use case	delete pharmacies
Actors	Admin
Precondition	logged in
Basic flow	1-open control page 2- select pharmacy 3-click on delete
Alternative flow	
Post condition	The pharmacy is informed with admin decision

Use case	Hang pharmacy
Actors	Admin
Precondition	logged in
Basic flow	1-open control page 2- select pharmacy 3-click hang button
Alternative flow	
Post condition	Inform the pharmacy that it is no longer supported by our services

Use case	UnHang pharmacy
Actors	Admin
Precondition	logged in
Basic flow	1-open control page 2- select pharmacy 3-click Unhang button
Alternative flow	
Post condition	the pharmacy is informed that it is back to be supported by our services

Use case	Show control page
Actors	admin
Precondition	logged in
Basic flow	1-open control page
Alternative flow	
Post condition	Show pharmacies status and names

Use case	Show report
Actors	Admin
Precondition	logged in
Basic flow	1-open control page 2- select show report
Alternative flow	
Post condition	there is orders that is already in <u>orderlist</u> and show most ordered regions

Use case	Rate
Actors	User
Precondition	logged in
Basic flow	1-select pharamcy 2-click rate
Alternative flow	
Post condition	Show rate result

Use case	Set schedule
Actors	User
Precondition	logged in
Basic flow	1-open profile page 2-click set schedule
Alternative flow	
Post condition	The conform message

Use case	View profile
Actors	User
Precondition	logged in
Basic flow	1-open profile
Alternative flow	The user name and pass is wrong so user can't login
Post condition	The profile will be displayed



Use case	Order medicine
Actors	User Pharmacy
Precondition	logged in
Basic flow	1- search for medicine 2-select pharmacy 3-click order
Alternative flow	The medicine is not available so no pharmacies will appear or pharmacy does not support ordering
Post condition	The order notification will be sent to pharmacy

Use case	Edit Order
Actors	User pharmacy
Precondition	logged in medicine ordered
Basic flow	1- open orders pages 2-select order 3-click edit order
Alternative flow	The order already passed editing time so no re-fonded orders
Post condition	The User new order will be displayed and notification will be sent to pharmacy

Use case	delete Order
Actors	User pharmacy
Precondition	logged in medicine ordered
Basic flow	1- open orders page 4-select order 5-click delete order
Alternative flow	The order already passed deleting time so no re-fonded orders
Post condition	The order will be deleted and notification send to pharmacy

Use case	Add medicine
Actors	Pharmacy
Precondition	logged in
Basic flow	1-open add page 2-enter medicine barcode 3- enter expiry date
Alternative flow	Enter wrong barcode
Post condition	The medicine will be added in their database and will be searched by user

Use case	remove medicine
Actors	Pharmacy
Precondition	logged in
Basic flow	1-open medicine control page 2- click on remove button 2- enter barcode for medicine need remove
Alternative flow	The medicine isn't already added so it will return an error message to pharmacy
Post condition	The medicine will be removed from their DB

Use case	Register medicine
Actors	Pharmacy
Precondition	logged in
Basic flow	1-open medicine control page 2- click on Register medicine button 2- enter barcode for medicine need remove
Alternative flow	The medicine is already registered
Post condition	The medicine will be registered to system and available in this pharmacy

Use case	Set alternatives
Actors	Pharmacy
Precondition	logged in the medicine in Database
Basic flow	1-select medicine 2-open medicine info 5-add alternatives
Alternative flow	The pharmacy in hung mode so it can't edit medicine alternatives
Post condition	The alternative is added to medicine

Use case	Edit expire date
Actors	Pharmacy
Precondition	logged in
Basic flow	1-add new medicine 2-enter expire date
Alternative flow	The pharmacy in hung mode so it can't access their profile
Post condition	Send notification to pharmacy when the medicine is expired

Use case	Bind patient
Actors	Pharmacy
Precondition	logged in
Basic flow	1-search for patient 2- click on Bind button
Alternative flow	The pharmacy enter wrong key
Post condition	The pharmacy will be able to edit user history

Use case	Edit user history
Actors	Pharmacy
Precondition	logged in the pharmacy Banded user
Basic flow	1-open user profile 2-edit user data
Alternative flow	
Post condition	The user data is edited

Use case	Accept order
Actors	Pharmacy
Precondition	already the medicine ordered
Basic flow	1-view orders 2- click accept order
Alternative flow	The pharmacy in hung mode so it can't access their profile the amount of medicine isn't enough in this pharmacy
Post condition	Notification will be sent to User who ordered the order

<i>Use case</i>	<i>Decline order</i>
<i>Actors</i>	Pharmacy
<i>Precondition</i>	already the medicine ordered
<i>Basic flow</i>	1-view orders 2- click decline order
<i>Alternative flow</i>	The pharmacy in hung mode so it will automatically lose all active orders
<i>Post condition</i>	The pharmacy is logged in and not in hung mode and have already the medicine ordered

<i>Use case</i>	<i>Warning expire</i>
<i>Actors</i>	Server
<i>Precondition</i>	Medicine expired in DB The pharmacy is logged in not in hung mode
<i>Basic flow</i>	Send notification to pharmacy
<i>Alternative flow</i>	The pharmacy in hung mode so no notification is not available
<i>Post condition</i>	Notification will be sent to pharmacy

<i>Use case</i>	<i>Hang pharmacy</i>
<i>Actors</i>	Server
<i>Precondition</i>	No resubscribe done by pharmacy
<i>Basic flow</i>	Send notification to pharmacy and close their account
<i>Alternative flow</i>	The pharmacy resubscribe after
<i>Post condition</i>	Notification will be sent to pharmacy

<i>Use case</i>	<i>Notify for schedule</i>
<i>Actors</i>	Server
<i>Precondition</i>	Time of medicine in schedule has come
<i>Basic flow</i>	Send notification to user informs him about the scheduled medicines
<i>Alternative flow</i>	No medicine is found in pharmacies
<i>Post condition</i>	Ask the user for his permission to order the medicines

<i>Use case</i>	<i>register</i>
<i>Actors</i>	Guest
<i>Precondition</i>	open register window
<i>Basic flow</i>	1-fill the data of the registration forum which is: user name ,password , mail , phone 2-click submit 3-check data validation 4-save data in database
<i>Alternative flow</i>	Data is wrong and data not saved
<i>Post condition</i>	1- An activation message will be sent to user's e-mail 2-open profile page

## 10 Preliminary Schedule Adjusted

...

## 11 Preliminary Budget Adjusted

...

## **12 Appendices**

...

### **12.1 Definitions, Acronyms, Abbreviations**

...

### **12.2 Collected material**

## **13 References**