

+Patient():ctor
+addDoctor(Doctor):boolean
+addAppointment(Doctor, Patient, Date):boolean
+sendChangeAppointmentRequest():boolean
+seeAppointmentDates():void
+sendMessages():void
+setName(name):String
+setBloodType( String ): void
-setActivity(int):void
+dischargePatient():void
+reTakePatient():boolean
+createDrug( drugName:String, hungryFull:boolean, startingDate:Sti

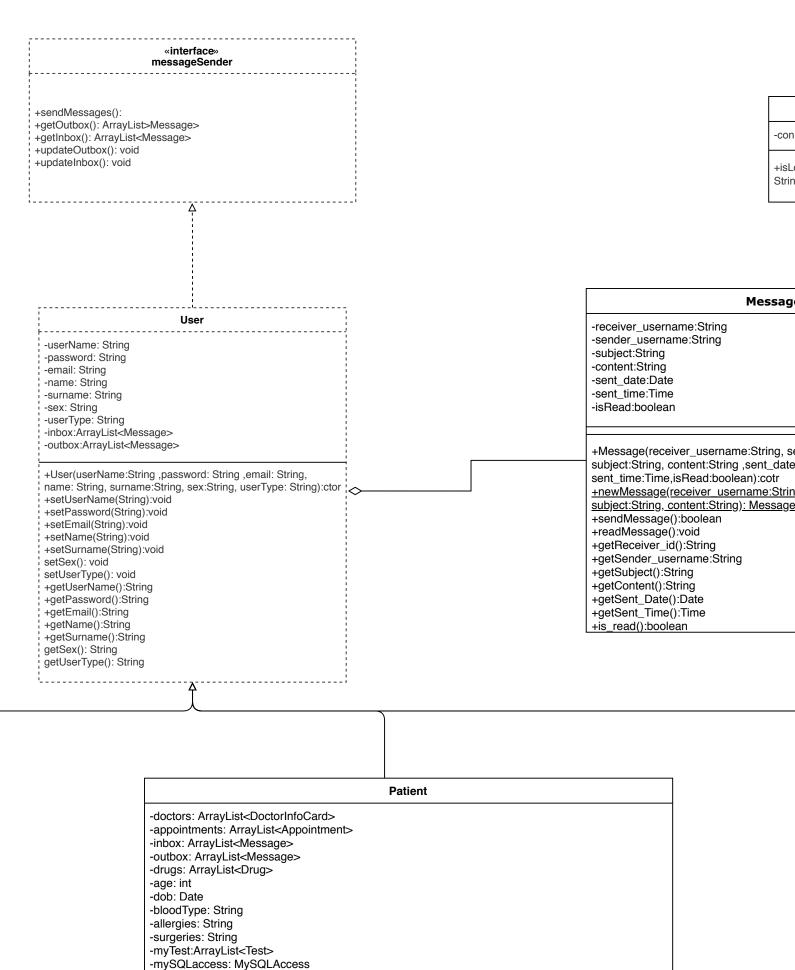
+isR Strin

-con

# RegisterModel : Connection egister(username:String, password: g, email:String, name:String, ame:String sex:String):boolean

<-----

# Admin +addDoctor(name:String, surname: String, email:String sex:String, speciality: String): boolean +addLabTech(name:String, surname: String, email:String sex:String): boolean +searchWorker(name: String): ArrayList<User> seeDoctors() seeLabTechs()



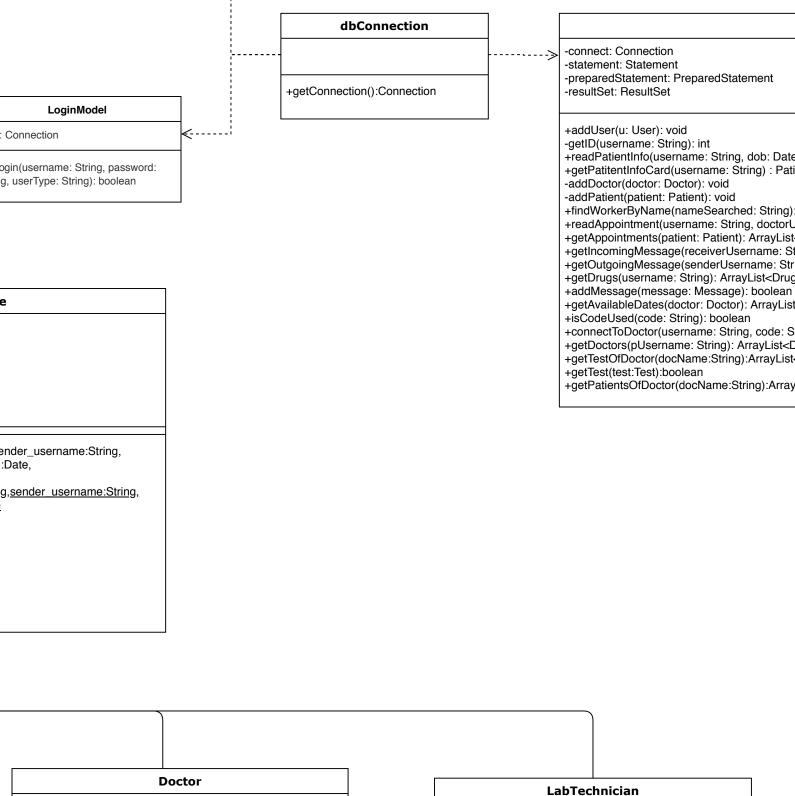
+Patient(username: String, password:

+getDrugs(): ArrayList>Drug

+getAppointmentDates(): ArrayList<Appointment>

+getAge(): int +getDob(): Date +getAllergies(): String +getSurgeries(): String +getBloodType(): String

String, email: String, name: String, surname: String, sex: String):cotr



# -patients : ArrayList<PatientSlot> -appointments: ArrayList<Appointment> -inbox: Arraylist<Message> -outbox: Arraylist<Message> -availableTimes: Arraylist<Timestamp> -mySQLAccess: MySQLAccess -speciality: String -myTests:ArrayList<Test> + Doctor(username:String, password:String, email:String, name:String, surname:String, sex:String, speciality:String):cotr + getAvailableTimes():ArrayList<Timestamp> + sendMessage(username:String, subject:String, content:String):void + getInbox():ArrayList<Message> + getOutbox():ArrayList<Message> + createCodeForPatient():Code + getSpeciality():String + setSpeciality():String + approveAppointment(Appointment): void + updateOutbox():void

+updateInbox():void

- -tests: ArrayList<Test>
- -testRequests:<TestRequest>
- -inbox: Arraylist<Message>
- -outbox: Arraylist<Message>
- +LabTechnician(username: String, password:
- +String, email: String, name: String, surname: String, sex: String):cotr
- +sendTestResult(Test test):void
- +createNewTest(receiver\_username:String,
- test\_name:String, patient\_username:String,file:File):Test
- + sendMessage(username:String, subject:String,
- content:String):void
- + getInbox():ArrayList<Message>
  - + getOutbox():ArrayList<Message>
- +getTestRequests():ArrayList<TestRequest>

-test\_name:String

MySQLAccess
e, bloodType: String, age: int, allergies: String, surgeries: String entInfoCard
: ArrayList <user> Josername: String, date: Date, start_time: Time, end_time: Time): voic  Appointment&gt; ring): ArrayList<message> ring): ArrayList<message> ps&gt;</message></message></user>
<timestamp></timestamp>
tring): boolean ooctor> <test></test>
List <patientinfocard></patientinfocard>

- String

## Code

-code\_id:String -doctor\_username:String -boolean:isUsed

+Code(code\_id:String,doc\_username:String is\_used:boolean)

+newCode(doctor:Doctor): Code

# **DoctorInfoCard**

-doctorUsername: String -doctorEmail: String - doctorName: String -doctorSurname: String -doctorSex: String -doctorSpeciality: String

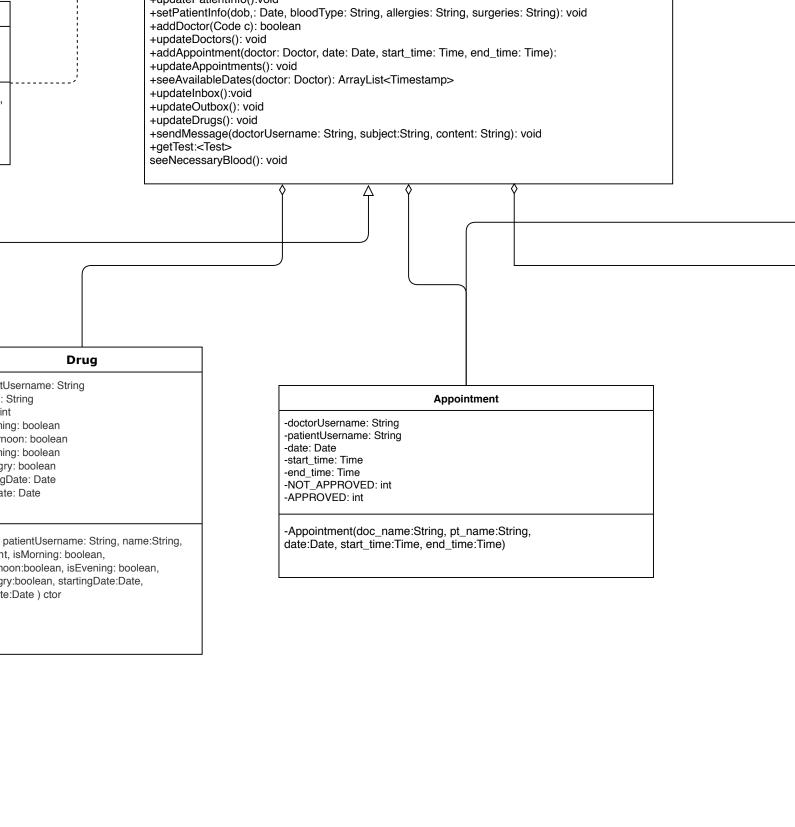
DoctorInfoCard(doctorUsername: String, doctorEmail: String, doctorEmail, doctorName: String, doctorSurname: String, doctorSex: String, doctorSpeciality: String):ctor

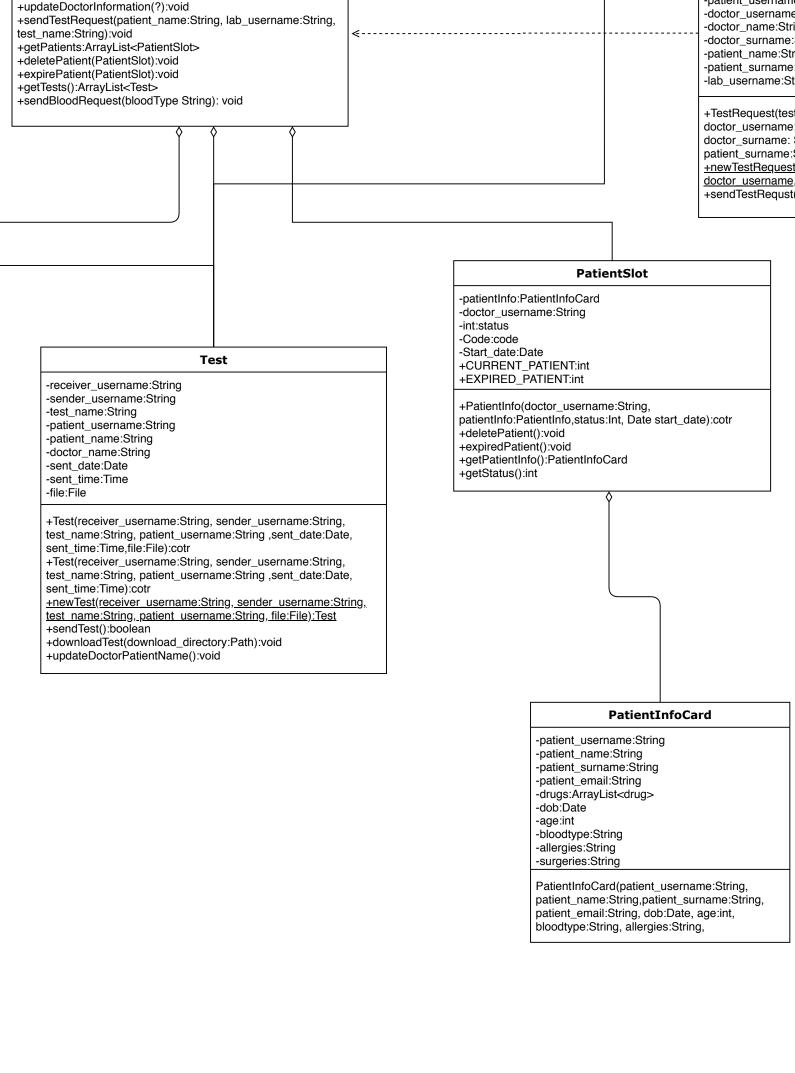
-patien -name -dose:

-isMorr -isAfter -isEver

-isHung -startin

+Drug( dose: i isAfteri isHung finalDa





e:String ng String ing String ring

t\_name:String, patient\_username:String, :String, doctor\_name:String, String, patient\_name:String, String, lab\_username:String):cotr (test\_name, patient\_username, \_lab\_username):TestRequest ():void