Luís Pedro Maia Seco (2020211927)

Hospital Management System

Report Project carried out within the scope of the Database Curricular Unit of the Degree in Informatics Engineering

Coimbra, 28 May 2024

Índice

Introduction

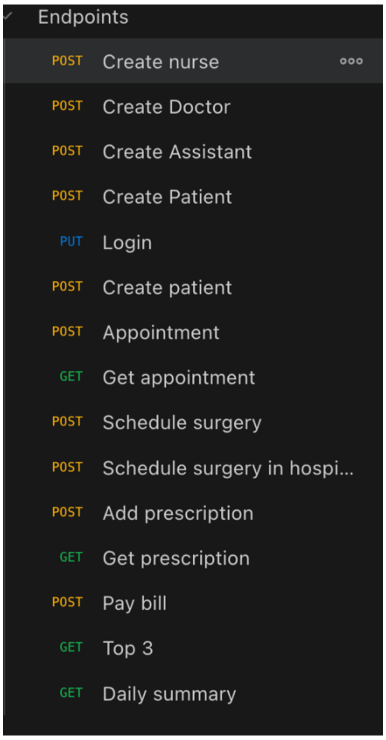
In this report will be made a summary of the actions realized for a Hospital Management System where the actors were, the patients, and employes (doctors, nurses and assistants). This system is capable of scheduling and searching appointments, surgeries and hospitalizations, etc.

Installation Manual

To run this application, you will need to have the most recent version of **python** and **pipenv** installed of your machine. To migrate the database run them directly in psql with the *\i <path>* command. To run the application, use the command *python -m flask run -p 5050* in your shell.

User Manual

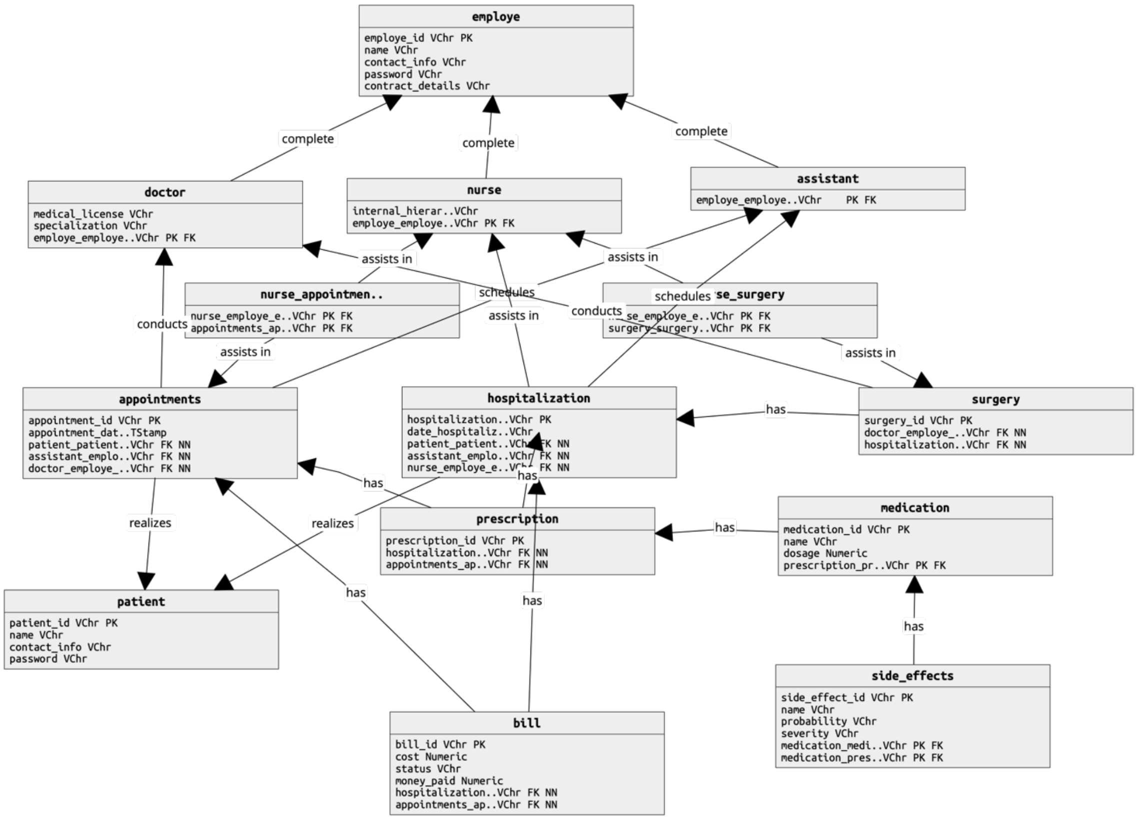
The postman collection includes all the endpoints requested in the assignment except the last one, we have a different request for the creation of each user, so it would be easier to add different users to the database since each one has different data models. Then we have an endpoint for the user authentication, if successful the user receives a jws token. After that, depending on the token of the user, he can perform some actions while others will be restricted. For example, a patient can only use the endpoint of requesting appointments if the appointments are from the user, while an assistant can see the appointments of all users.



Endpoints

POST	Create nurse
POST	Create Doctor
POST	Create Assistant
POST	Create Patient
PUT	Login
POST	Create patient
POST	Appointment
GET	Get appointment
POST	Schedule surgery
POST	Schedule surgery in hospi...
POST	Add prescription
GET	Get prescription
POST	Pay bill
GET	Top 3
GET	Daily summary

ER diagram


employe
employe_id VChr PK
name VChr
contact_info VChr
password VChr
contract_details VChr
complete
complete
complete
doctor
medical_license VChr
specialization VChr
employe_employe..VChr PK FK
nurse
internal_hierar..VChr
employe_employe..VChr PK FK
assistant
employe_employe..VChr PK FK
assists in
assists in
schedules
schedules
conducts
conducts
assists in
assists in
assists in
nurse_appointmen..
nurse_employe_e..VChr PK FK
appointments_ap..VChr PK FK
se_surgery
se_employe_e..VChr PK FK
surgery_surgery..VChr PK FK
appointments
appointment_id VChr PK
appointment_dat..TStamp
patient_patient..VChr FK NN
assistant_emplo..VChr FK NN
doctor_employe_..VChr FK NN
hospitalization
hospitalization..VChr PK
date_hospitaliz..VChr
patient_patient..VChr FK NN
assistant_emplo..VChr FK NN
nurse_employe_e..VChr FK NN
surgery
surgery_id VChr PK
doctor_employe_..VChr FK NN
hospitalization..VChr FK NN
has
has
has
has
has
has
has
realizes
realizes
prescription
prescription_id VChr PK
hospitalization..VChr FK NN
appointments_ap..VChr FK NN
medication
medication_id VChr PK
name VChr
dosage Numeric
prescription_pr..VChr PK FK
patient
patient_id VChr PK
name VChr
contact_info VChr
password VChr
side_effects
side_effect_id VChr PK
name VChr
probability VChr
severity VChr
medication_medi..VChr PK FK
medication_pres..VChr PK FK
bill
bill_id VChr PK
cost Numeric
status VChr
money_paid Numeric
hospitalization..VChr FK NN
appointments_ap..VChr FK NN

Additional information

Some responses from the endpoints may not be the same has was asked in the worksheet, this happens in the scheduling of a surgery where multiple nurses should be involved but only one is possible to have, since the application was not built to have different nurses in a surgery.

In the bill table to make it easier for the control of money already paid from the user, I created a column that keeps track of the user spendings in the hospital so it would be easier to obtain the right information in the endpoint of the top 3 patient spenders.

To make sure only the correct users were using the endpoints, I created a query that checks the user_id when a user it's loggin in, so in the query it's being compared the user_id from the login to the doctor_id, assistant_id and nurse id in the database, if it matches with one of those id's then when the jwt it's being created, the role of the user becomes one of the three employes depending on the id match, also in the identity of the jws token it keeps track of the user_id.