Covid-Hospital Database

A picture containing plant

Description automatically generated

Katerina Tsilingiri, 2806

Chrysa Noli, 2780

**Contents:**

* Files included – explanation
* Description of database – General Information
* Requirements (Domain)
* ER-Diagram
* Implementation of tables in detail
* Relation Schema
* Functional Dependencies
* Normalization in 3NF
* SQL Queries
* Instructions to use our application
* Bibliography – Helpful links

**Files included** in project Covid-Hospital Database:

* **covidhospital-database.erx** 🡪 ER-diagram made in ER2SQL
* **covidhospital-database.sql**🡪SQL-code which upload to pgAdmin4 in postgreSQL12
* **presentation.docx**
* **covidhospital-database.accdb** 🡪 Access File, our user interface
* **relationalschema.png** 🡪 relational schema for our database

**Description of database – General Information**

The main aim of this project is to give information about the Fuctioning of Databases in a Covid-Hospital System. This Covid-Hospital database will allow creating and maintaining database including information of all patients (covid or not), all the hospital employee’s including the doctors, the nurses, the reception, the technical staff and the relations that the doctors and nurses have with their patients.

Patients entering the hospital will need to register first by filling a form at the reception . Each form has a unique number whose domain will be like all-natural numbers which will be assigned to that patient in the form of p\_id (aka patient’s id) with the integer datatype and will be getting stored into the patient information database. This database will also include the result of covid test the patient did, the medicine the doctor gave him and of course the doctor who supervises him/her. There will be a database containing useful information about all employees. Each one will be provided a unique identification number (aka s\_id) whose domain will be like all-natural numbers. Also, there will be database containing an information about the doctors (part of employees) within a covid department. Each doctor will be assigned a unique identification number (aka doc\_id) whose domain will be like all-natural numbers too. There will be another database containing information about nurses (nur\_id), technical staff (tec\_id) and reception (rec\_id) too.

The user groups involved with respect to Covid-Hospital database are:

1. Receptionist
2. Doctors
3. Nurses
4. Technical Staff

Covid-Hospital database is a vast application and in real time, it involves many users. But in this project we will focus only on 4 users and their interaction with covid or non-covid patients.

Overall Objective:

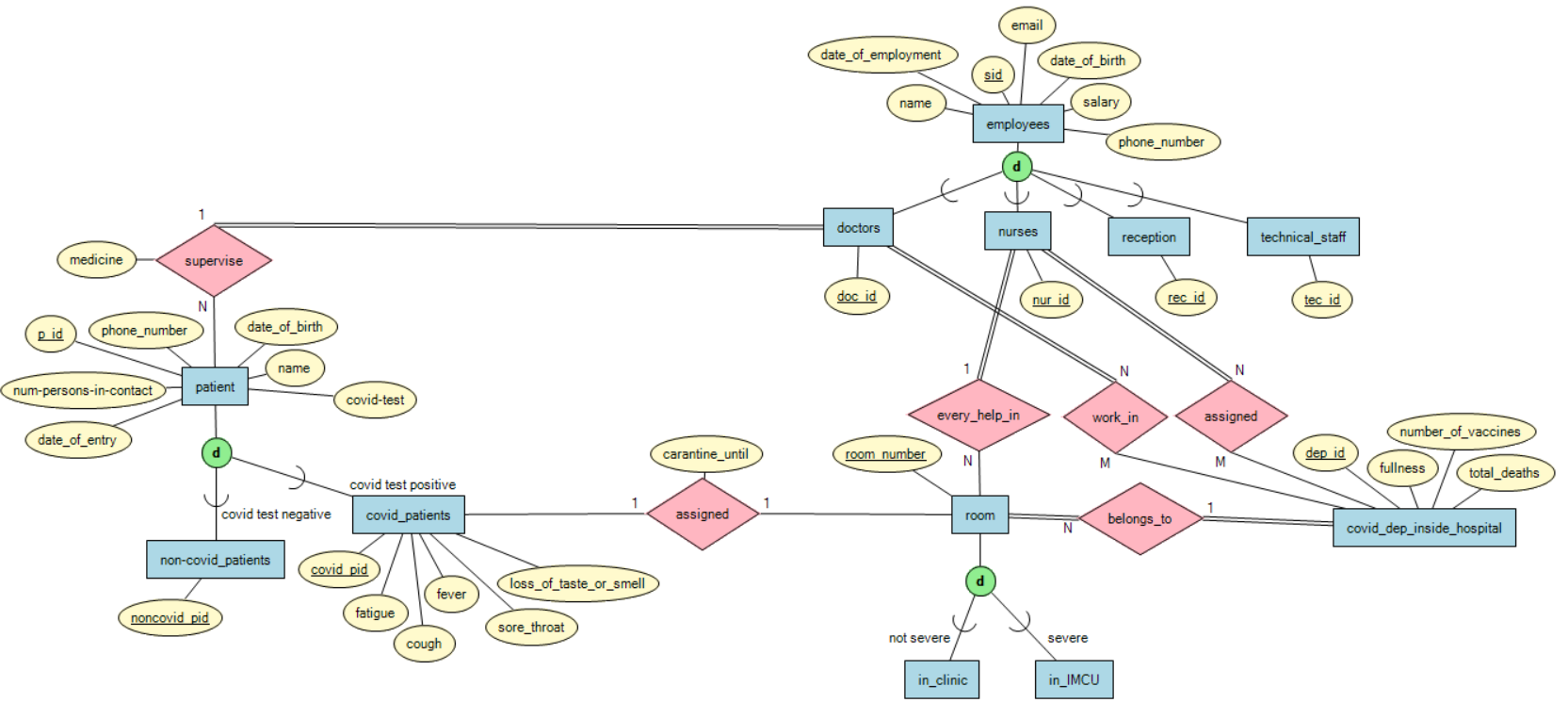
* Registration: Patient register by filling visiting form.
* Patients Details: Patient Bio data, also covid test applied
* Supervision: A patient is supervised by a doctor who also gave to him/her medication.
* Doctors: Doctor information and in which department work in
* Nurses: Nurses information and in which department assigned in
* Covid-Patient: If covid test positive, patient’s information about his covid symptoms and how much days carantine.
* Non-Covid-Patient: If covid test negative, patient’s information.
* Room: Room information. Assigned to those who have covid. Either in clinic or in imcu. Always help in room from a nurse.

**Domain Requirements**

The domain requirements are as following:

1. All the patients, doctors as well as the covid-hospital employees are assigned a unique identification number.
2. All the unique number assigned are not nuttable, i.e. they can’t be assigned a null value.
3. The minimum salary of employees should be 1000$.
4. Covid Symptoms except fever, assigned as ‘Y’ or ‘N’, covid test too.
5. A room must belongs to a covid department and sure a nurse must help in room.
6. Doctors and nurses (minimum 1 of each) must work in a covid department .

**ER-Diagram**:



(for better quality open the .erx file)

**Implementation of tables in detail**

To implement the above requirements, we would require a following relation.

1. The relation “Patients” contains all the information about the patient visiting hospital such as patient name, birthday etc. The patient id provided is unique to all patients. Each fk1\_id references to the sid mentioned in Doctors relation.

Format for date\_of\_entry and date\_of\_birth is MM/DD/YYYY.

Domain for Covid Test in ‘Y’ or ‘N’.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Patient | | | | |
| Attribute | Data type | Size | Domains | Cascading |
| p\_id | integer |  | not null,unique |  |
| name | varchar | 30 | not null |  |
| Date\_of\_entry | date |  | not null |  |
| Num\_of\_persons\_in\_contact | integer |  | not null |  |
| Phone number | bigint |  | not null |  |
| Date\_of\_birth | date |  | not null |  |
| Covid test | char | 1 | not null,(Y,N) |  |
| Medicine | varchar | 15 | not null |  |
| Fk1\_sid | integer |  | not null,unique | Delete,  update |

1. The relation “Non-Covid Patient” is same as patient relationship above, except the add of a field called non-covid\_pid which is unique for each non-covid patient (aka primary key).
2. The relation “Covid-Patient” contains all the information about the patient whose covid test is positive such as patient name, birthday etc. The covid patient id provided is unique to all covid patients. Each fk1\_id references to the sid mentioned in Doctors relation.

Format for date\_of\_entry and date\_of\_birth is MM/DD/YYYY.

Domain for Covid Test, cough, sore\_throat, fatigue, loss\_of\_taste\_or\_smell is ‘Y’ or ‘N’.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Covid Patient | | | | |
| Attribute | Data type | Size | Domains | Cascading |
| p\_id | integer |  | not null,unique |  |
| name | varchar | 30 | not null |  |
| Date\_of\_entry | date |  | not null |  |
| Num\_of\_persons\_in\_contact | integer |  | not null |  |
| Phone number | bigint |  | not null |  |
| Date\_of\_birth | date |  | not null |  |
| Covid test | char | 1 | not null,(Y) |  |
| Medicine | varchar | 15 | not null |  |
| Fk1\_sid | integer |  | not null,unique | Delete,  update |
| cough | char | 1 | not null,(Y,N) |  |
| fever | integer |  |  |  |
| sore\_throat | char | 1 | not null,(Y,N) |  |
| fatigue | char | 1 | not null,(Y,N) |  |
| covid\_pid | integer |  | not null,unique |  |
| loss\_of\_taste\_or\_smell | char | 1 | not null,(Y,N) |  |

4.The relation “Employees” contains all the information about the hospital staff such as name, birthday etc. The employee id provided is unique to all employees. Format for date\_of\_employment and date\_of\_birth is MM/DD/YYYY.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Employees | | | | |
| Attribute | Data type | Size | Domains | Cascading |
| sid | integer |  | not null,unique |  |
| name | varchar | 30 | not null |  |
| Date\_of\_employment | date |  | not null |  |
| email | varchar | 25 | not null |  |
| Phone number | bigint |  | not null |  |
| Date\_of\_birth | date |  | not null |  |
| salary | int |  | not null,>1000 |  |

5.The relation “Doctors” contains all the information about the hospital doctors such as name, birthday etc. The doctor id provided is unique to all doctors. Format for date\_of\_employment and date\_of\_birth is MM/DD/YYYY.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Doctors | | | | |
| Attribute | Data type | Size | Domains | Cascading |
| sid | integer |  | not null,unique |  |
| name | varchar | 30 | not null |  |
| Date\_of\_employment | date |  | not null |  |
| email | varchar | 25 | not null |  |
| Phone number | bigint |  | not null |  |
| Date\_of\_birth | date |  | not null |  |
| salary | int |  | not null,>1000 |  |
| doc\_id | int |  | not null,unique | Delete,  update |

6.The relation “Nurses” contains all the information about the hospital nurses such as name, birthday etc. The nurse id provided is unique to all nurses. Format for date\_of\_employment and date\_of\_birth is MM/DD/YYYY.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nurses | | | | |
| Attribute | Data type | Size | Domains | Cascading |
| sid | integer |  | not null,unique |  |
| name | varchar | 30 | not null |  |
| Date\_of\_employment | date |  | not null |  |
| email | varchar | 25 | not null |  |
| Phone number | bigint |  | not null |  |
| Date\_of\_birth | date |  | not null |  |
| salary | int |  | not null,>1000 |  |
| nur\_id | int |  | not null,unique | Delete,  update |

7. The relation “Reception” and “Technical Staff” is same as the above ones except from rec\_id which is unique for the employees who work in reception and tec\_id which is unique for those who are in technical staff.

8.The relation “Covid Department Inside Hospital” contains all the information about the covid departments that exist, such as fullness, total deaths etc. The dep\_id is unique for each covid department.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Covid Department Inside Hospital | | | | |
| Attribute | Data type | Size | Domains | Cascading |
| dep\_id | integer |  | not null,unique |  |
| Number\_of\_vaccines | integer |  | not null |  |
| fullness | integer |  | not null |  |
| Total\_deaths | integer |  | not null |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| assigned | | | | |
| Attribute | Data type | Size | Domains | Cascading |
| Fk1\_sid | integer |  | not null,unique | Delete,  update |
| Fk2\_dep\_id | integer |  | not null,unique | Delete,  update |

9.The relation “assigned” contains the information of which nurse works in which department. The field fk1\_sid is sid from “nurses” relation and fk2\_dep\_id is the department’s id from the above relation. Both unique.

10.The relation “work\_in” contains the information of which doctor works in which department. The field fk1\_sid is sid from “doctor” relation and fk2\_dep\_id is the department’s id from the department’s relation. Both unique.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Work\_in | | | | |
| Attribute | Data type | Size | Domains | Cascading |
| Fk1\_sid | integer |  | not null,unique | Delete,  update |
| Fk2\_dep\_id | integer |  | not null,unique | Delete,  update |

11.The relation “Room” contains the information about room such as room number etc. The room number is unique for each room.The field fk1\_dep\_id is dep\_id from department’s relationship, fk2\_sid is the sid from nurses’ relationship because as we said a room has a nurse to help and the field fk3\_pid is the pid from Covid\_Patient’s relationship for the patient who is now in the room. Format carantine\_until is MM/DD/YYYY.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Room | | | | |
| Attribute | Data type | Size | Domains | Cascading |
| Room number | integer |  | not null,unique |  |
| Fk1\_dep\_id | integer |  | not null,unique | Delete,update |
| Fk2\_sid | integer |  | not null,unique | Delete,update |
| Fk3\_pid | integer |  | not null,unique | Delete,update |
| Carantine\_until | date |  | not null |  |

12. The relation “in\_clinic” and “in\_imcu” is same as the above ones.

**Relation Schema**: (open the .png for better quality)

Diagram, table, engineering drawing

Description automatically generated

**Functional Dependencies**

In “Employees” relation:

sid --> name, date\_of\_birth, date\_of\_employment, email, salary, phone\_number

\*if we consider that 2 persons can’t have the same phone number

phone\_number --> sid, name, date\_of\_birth, date\_of\_employment, email, salary

\*if we consider the same for email

email --> sid, name, date\_of\_birth, date\_of\_employment, salary, phone\_number

More specific in “Doctors”, “Nurses”, “Reception”, “Technical Staff” relations:

doc\_id --> sid, name, date\_of\_birth, date\_of\_employment, email, salary, phone\_number

nurs\_id --> name, date\_of\_birth, date\_of\_employment, email, salary, phone\_number

rec\_id --> name, date\_of\_birth, date\_of\_employment, email, salary, phone\_number

tec\_id --> name, date\_of\_birth, date\_of\_employment, email, salary, phone\_number

In “Patients” relation:

p\_id --> name, date\_of\_birth, phone\_number, date\_of\_entry, covid\_test, num\_persons\_in\_contact

p\_id --> doc\_id, medicine

\* if we consider that 2 persons can’t have the same phone number

phone\_number --> p\_id, name, date\_of\_birth, date\_of\_entry, covid\_test, num\_persons\_in\_contact

More specific in “Covid-Patient” and “Non-Covid-Patient” relations:

noncovid\_pid --> name, date\_of\_birth, phone\_number, date\_of\_entry, covid\_test, num\_persons\_in\_contact

noncovid\_pid --> doc\_id, medicine

covid\_pid --> name, date\_of\_birth, phone\_number, date\_of\_entry, covid\_test, num\_persons\_in\_contact

covid\_pid -> fatigue, cough, fever, sore\_throat, loss\_of\_taste\_or\_smell, room\_number, carantile\_until, doc\_id

In “Room” relation:

room\_number -> nurs\_id, p\_id

In “Covid-Dep-Inside-Hospital” relation:

dep\_id -> fullness, number\_of\_vaccines, total\_deaths

**Normalization in 3NF**

First Normal Form (1NF)  
A table is in 1NF if :

* + it has only single (atomic) valued attributes/columns
  + the values stored in a column are of the same domain
  + there is a unique name for a every attribute/column
  + the order in which data is stored, does not matter

Second Normal Form (2NF)

A table is in 2NF if :

* + it is already in 1NF
  + all non-key fields depend on all primary keys (no partial dependency)

Third Normal Form (3NF)

A table is in 3NF if :

* + it is already in 2NF
  + no non-key value depends upon one another (no transitive dependency)

Our database satisfies all the rules of all the three normal forms, that are mentioned above.

**SQL-Queries**

For space economy, you can see the SQL-Queries and generally the code we wrote to make these tables and their relationships in covidhospital-database.sql.

**Instructions to use our application**

When you open the covidhospital-database.accdb file a switchboard will appear. You can navigate here like:

* You can choose either to see the forms patients, employees, room, covid-department and add someone in tha database, or simply see those who are registered in database so far.
* Also, the application offers you the ability to study some interesting and useful (for us) reports we create for specific examples.
* You can see the SQL-Queries too and edit them in order to take different results.
* You can Search in Employees or you can Search in Patients by name and take the info the database has for the specific person.
* In the end, to exit Access you can push the “Quit Access” button we made for closing our application.

**Bibliography – Helpful links**

<https://www.guru99.com/database-normalization.html>

<https://www.studytonight.com/dbms/second-normal-form.php?fbclid=IwAR0xDf0RswQ01O7nylrMq1wIdSzxf67M72JLVtNol0ZpGj_BmUbTRUeN46k>

<https://www.youtube.com/watch?v=HSnHGQeX9so&ab_channel=ComputerLearningZone>

<http://www.techonthenet.com/access/switchboard/index.php>

<https://www.w3resource.com/sql-exercises/hospital-database-exercise/sql-exercise-hospital-database-22.php>