



Comp204 Final

Mehmet Anıl İrfanoğlu

Instructor

Ahmet Soran
Samet Tonyalı

Video Link:

<https://www.youtube.com/watch?v=Os9QcayOirM>

Abstract

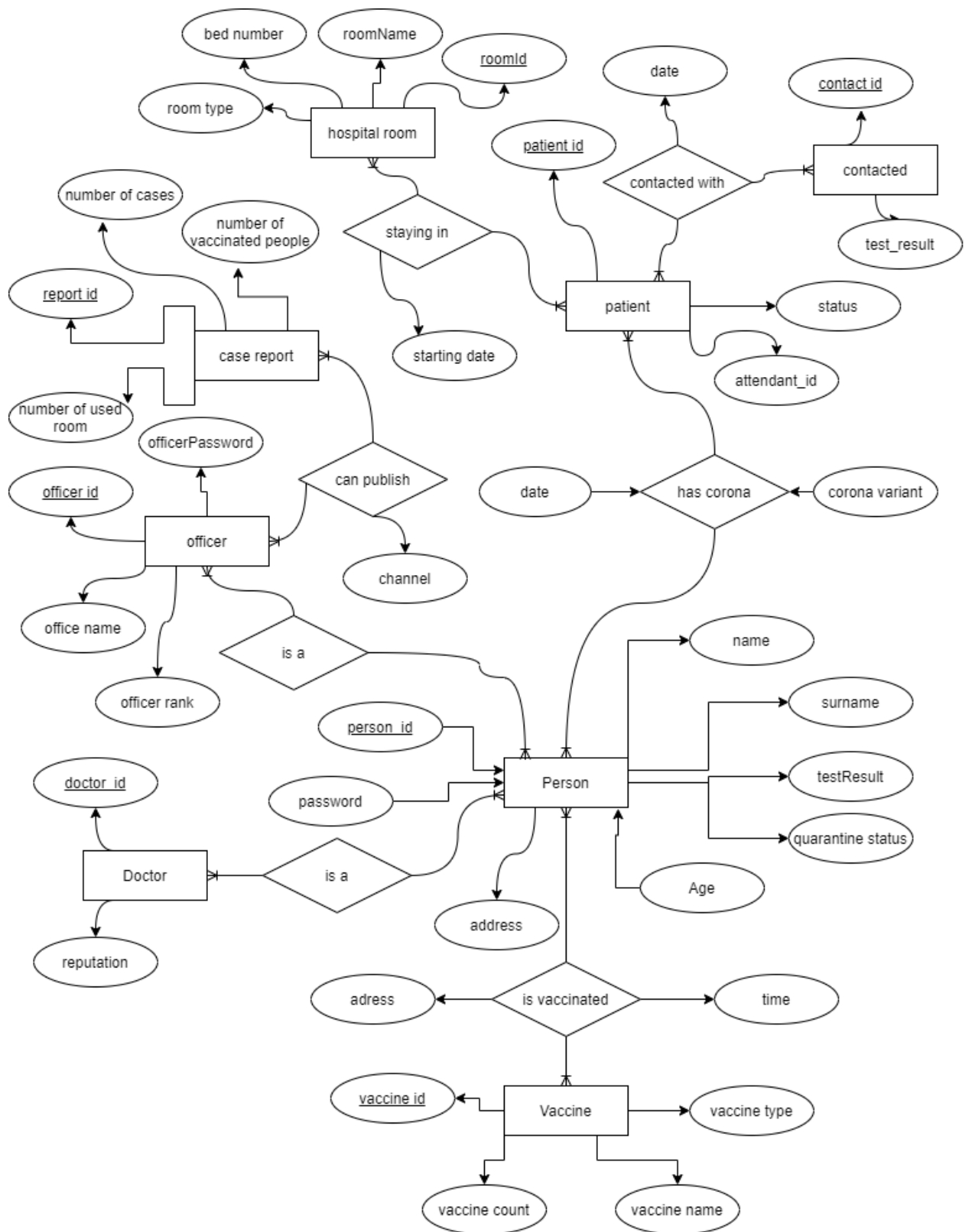
I made a database study on the first subject given in the visa in this project. In this context, I aimed to predict what might be required and establish a system accordingly. I have done more than one work for this purpose. I examined the corona epidemic, which is today's problem in general terms, and presented an application design in which the weight to contribute to the measures to be taken against this disease is given to the database.

REQUIREMENT ANALYSIS

I will write what might be needed in creating this project in this section. In my opinion, priority should be given to people in a database to be established about the disease. In this context, I have determined the person as the main starting point. but I do not think that the person alone is enough, and in this context, I have created a few more fields that depend on the person's structure and receive support from him. Chief among them is the patient class. I aim to keep a person's disease information here. In addition to this, I think that doctors who take care of patients should also be in this system. I also added government officials, who are above all else, who have access to everything, to the system. These people will be able to share the necessary parts of the collected data with the public at certain time intervals.

I built structures related to the disease as well as people. For example, I am thinking of recording the information about who the sick people are in contact with. I also aim to keep information about the rooms where the patients stay. Based on this, I think to include another important issue of vaccination into the system. In this context, I aim to keep information about which vaccine is available, which vaccine is used and vaccine. Finally, I think of designing places where there are general status statistics and where information that can be shared by government offices will be kept.

E-R DIAGRAM



DESIGN PHILOSOPHY

ER Diagram:

- **Address**
- **Person**
- **Doctor**
- **Patient**
- **Contacted**
- **HospitalRoom**
- **Officer**
- **CaseReport**
- **Vaccine**

In this section, I will talk about the elements in the design I created and what they do. First of all, I want to talk about the **person** being. person as the main starting point; It contains general information such as **person_id**, **name**, **surname**, **age**. This information is important in identifying individuals. Based on these, the person's **password**, **test result** and whether he is in **quarantine** are also kept in person.

Another structure that depends on the person is the **patient**. With the **hasCorona** relation linked to the patient, I keep track of **date** the person is sick and the **type of corona**. As I am currently keeping general information in the person, I did not specify them again in the patient section.

This patient entity is linked to other tables with 2 different relations. For example, **contactedWith** relation and **contacted** person was kept. In **contactedWith**, I also record when the contact occurred. On the **contacted** table, I keep the **ID** and **test result** of the person in contact.

The other relation to which the patient is connected is **hospitalRoom**. Here, I connect which patient is staying in which **room** with **stayingIn** relation and I keep the information on which **date** the patient started to stay in that room.

On the **hospitalRoom** table, I keep important information about the hospital room. For example, I keep the **numberOfBeds** in the hospital room and what **type of room** this room is. By room type I mean, for example, a room can be **intensive care**, and since not all equipment in intensive care is in normal rooms, keeping this information will be very important in the future.

It is also important to know how many beds are in the room because if the room is full, no more patients can be accommodated in that room. I wanted to determine which **doctor** was looking at which patient by adding the doctor table. This information is connected to the

patient with **attendant_id**.

Speaking of the **officer**, these people can be thought of as high-level users. To give an example, these people can also be considered as officials working in the infrastructure of the ministry of health. The purpose of these officers is to share situation **reports** at regular intervals, as stated in **caseReport**, by looking at the data collected in general.

In these reports they share, the information about how many people are sick with **NumberOfCases**, how many rooms are used now with **numberOfUsedRoom**, and lastly, how many people have been vaccinated with **numberOfVaccinatedPeople** are kept.

These shared reports are shared by the officer by adding the information on **canPublish** relation and the **channel** that will be published on which platform. these reports can only be shared by the officer.

CARDINALITIES

MANY TO MANY

Patient,Contacted

Patient,HospitalRoom

Person,Patient

Person,Vaccine

Person,Doctor

Person,Officer

Officer,CaseReport

FUNCTIONAL DEPENDENCIES

Address(Address_Id,City,District)

Address_Id->City, Address_Id->District so,

Address_Id⁺ = Address_Id City District ;

so **Address_Id** is **candidate key**.

Person(Person_id , Name ,Surname ,Password, Age ,TestResult, Address_Id ,Quarantine_Status)

Person_id ->Name, Person_id ->Surname, Person_id ->Password, Person_id ->Age, Person_id ->TestResult, Person_id ->Address_Id, Person_id ->Quarantine_Status

Person_id⁺ = Person_id Name Surname Password Age TestResult Address_Id Quarantine_Status ;

so **Person_id** is candidate key.

Doctor(Doctor_Id , Reputation)

Doctor_Id -> Reputation so,

Doctor_Id ⁺ = Doctor_Id Reputation ;

so **Doctor_Id** is candidate key.

Patient(Patient_Id,Status,Attendant_Id)

Patient_Id -> Status, Patient_Id -> Attendant_Id so,

Patient_Id ⁺ = Patient_Id Status Attendant_Id;

so **Patient_Id** is candidate key.

Contacted(Contact_Id,TestResult)

Contact_Id -> TestResult so,

Contact_Id ⁺ = Contact_Id TestResult ;

so **Contact_Id** is candidate key.

HospitalRoom(Room_Id,BedNumber,RoomType)

Room_Id -> roomName ,Room_Id -> BedNumber, Room_Id -> RoomType so,

Room_Id ⁺ = Room_Id roomName BedNumber RoomType;

so **Room_Id** is candidate key.

Officer(Officer_Id,OfficeName,OfficerPassword,OfficerRank)

Officer_Id -> OfficeName, Officer_Id -> OfficerPassword, Officer_Id -> OfficerRank so,

Officer_Id ⁺ = Officer_Id OfficeName OfficerPassword OfficerRank;

so **Officer_Id** is candidate key.

CaseReport(Report_Id,NumberOfCases,NumberOfUsedRooms,NumberOfVaccinatedPeople)

Report_Id -> NumberOfCases, Report_Id -> NumberOfUsedRooms, Report_Id ->

NumberOfVaccinatedPeople so,

Report_Id ⁺ = Report_Id NumberOfCases NumberOfUsedRooms NumberOfVaccinatedPeople;

so **Report_Id** is candidate key.

Vaccine(Vaccine_Id,VaccineCount,VaccineName,VaccineType)

Vaccine_Id -> VaccineCount, Vaccine_Id -> VaccineName, Vaccine_Id -> VaccineType so,

Vaccine_Id + = Vaccine_Id VaccineCount VaccineName VaccineType;

so **Vaccine_Id** is **candidate key**.

ContactedWith(Carrier_Id,Patient_Id,Contact_Id,date)

Carrier_Id -> Patient_Id, Carrier_Id -> Contact_Id, Carrier_Id -> date so,

Carrier_Id + = Carrier_Id Patient_Id Contact_Id date;

so **Carrier_Id** is **candidate key**.

HasCorona(Case_Id,Person_Id,Patient_Id,Date,CoronaVariant)

Case_Id -> Person_Id, Case_Id -> Patient_Id, Case_Id -> Date , Case_Id -> CoronaVariant so,

Case_Id + = Case_Id Person_Id Patient_Id Date CoronaVariant;

so **Case_Id** is **candidate key**.

IsVaccinated(Vaccination_Id,Person_Id,Vaccine_Id,Address_Id,Date)

Vaccination_Id -> Person_Id, Vaccination_Id -> Vaccine_Id, Vaccination_Id -> Address_Id ,
Vaccination_Id -> Date so,

Vaccination_Id + = Vaccination_Id Person_Id Vaccine_Id Address_Id Date;

so **Vaccination_Id** is **candidate key**.

CanPublish(Publish_Id,Officer_Id,Report_Id,Channel)

Publish_Id -> Officer_Id, Publish_Id -> Report_Id, Publish_Id -> Channel so,

Publish_Id + = Publish_Id Officer_Id Report_Id Channel;

so **Publish_Id** is **candidate key**.

StayingIn(Staying_Id,Room_Id,Patient_Id,StartingDate)

Staying_Id -> Room_Id, Staying_Id -> Patient_Id, Staying_Id -> StartingDate so,

Staying_Id + = Staying_Id Patient_Id Report_Id StartingDate;

so **Staying_Id** is **candidate key**.

NORMALIZATION

1NF,2NF,3NF EXPLANATIONS

In order for my system to be in 1nf, it had to be an element in each cell. To achieve this, contact list, published reports, vaccine event can be considered. Because one person may have contact with more than one person and we cannot keep all contacts in one cell, and therefore, the information can be kept in a table for contact and the information can be stored in this way, and at the same time, multiple reports can be put into 1nf form with the same logic and set up a published table. Another case is that a person may have been vaccinated more than once, and to store it, the information that should be kept in a cell can be stored via a table by using the isVaccinated table. I will give an example of the things I have done to make my design in 2nf. The most important example is the person, doctor, patient structure. With this structure, I linked the tables that are independent from each other, but also the tables that store the data of the same person, with foreign keys. For example, a person can also be an officer, as well as a doctor. But no matter what, they are the same people all connected to each other through the foreign keys that I have defined. With this logic, a person who is not person cannot be a doctor, and these features are the requirements for 2nf. Finally, I will talk about what I have done for the 3nf form. Address information and vaccine information can be given as examples of what I do to catch this form. Because the vaccine information is linked to vaccine_id and if I open a vaccine table and keep it there rather than keeping this information in the vaccine's table, I can access this information from the person person through vaccine_id, but there is another important point here if I want to keep vaccine information about a person, also isvaccinated table. You should also check it out because someone who does not have vaccine does not have to have a vaccine_id. Another situation is related to the address. As for the address, there is the following situation, the city and district information depends on addree_id, which we can think of as the postal code because postal code is a set of numbers used to define a region and if I keep the address_id in person rather than keeping each city district information in person The address_id is actually the postal code specifying city and district, in this case, I set up the address table and keep the city district information for each person through the postal code address id, and I have provided 3nf because in the definition of the sought condition in 3nf, "A relation that is in First and Second Normal Form and in which no non-primary-key attribute is transitively dependent on the primary key, then it is in Third Normal Form (3NF) "and this definition is provided through these operations.

ADDITIONAL NOTES ABOUT NORMALIZATION

Normalization is normally done after the er diagram is designed, but while I was designing the er diagram, I made some changes that should be made in the future, namely the normalization process while designing ER.

First of all, the attribute of the address information in the er diagram causes an data redundancy use so I transferred the address information to a table. And in this way, if many people are streaming at the same address, there will be no data redundancy in the database. Another issue is the vaccine status.

Normally, whether or not to be vaccinated could be stated directly in person, but in this case it would take up unnecessary space because more than one user can get the same vaccine. The same is true for hospital rooms, because it makes more sense to create a

separate table rather than giving the hospital room information on the patient table. In this way, if more than one person stays in the same room, the database will not keep extra information for nothing.

There is another situation related to the contact list. As we know, a patient may have been in contact with many people, so I think it would be much more logical to open a table called contact and link them to the patient rather than adding an attribute to the patient table. In fact, it is the state person structure that reduces data redundancy the most. Because the most basic information of the person is stored in the person table, and whether a person is a doctor, an officer or a patient, they can connect to the person table and retrieve their information from there. In this way, instead of storing extra name and surname type information for each type of person, these are stored once and when necessary, the requested information is used by accessing them.

Another situation is related to the case report shared by the officers. As it is known, a report can be shared by more than one person on more than one platform, so it would be very logical to keep them as a separate table and in this way, data redundancy will be reduced.

Another situation is that if a person has been vaccinated more than once, we are saved from keeping the same information for the second time by keeping it in the relation table. Since the system minimizes this type of data redundancy, a person can be a doctor, a patient and a person at the same time, and while doing this, the information of the same person is not recorded repeatedly because different information is stored in each table.

ER TO RELATIONAL MAPPING

Normal Entities:

Address(Address_Id, City, District)

Person(Person_id, Name, Surname, Password, Age, TestResult, Address_Id, Quarantine_Status)

Doctor(Doctor_Id, Reputation)

Patient(Patient_Id, Status, Attendant_Id)

Contacted(Contact_Id, TestResult)

HospitalRoom(Room_Id, RoomName, BedNumber, RoomType)

Officer(Officer_Id,OfficeName,OfficerPassword,OfficerRank)

CaseReport(Report_Id,NumberOfCases,NumberOfUsedRooms,NumberOfVaccinatedPeople)

Vaccine(Vaccine_Id,VaccineCount,VaccineName,VaccineType)

Relations:

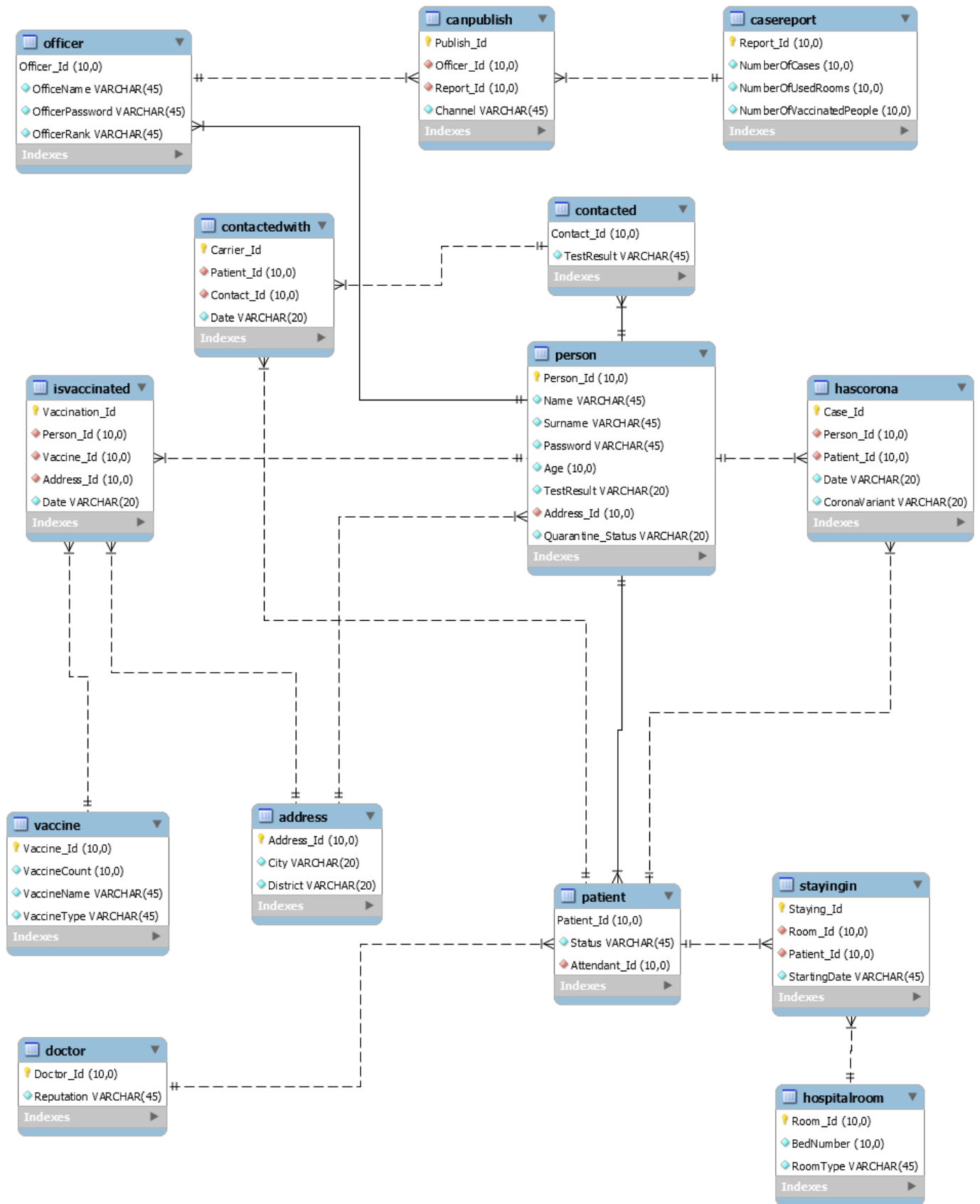
ContactedWith(Carrier_Id,Patient_Id,Contact_Id,date)

HasCorona(Case_Id,Person_Id,Patient_Id,Date,CoronaVariant)

IsVaccinated(Vaccination_Id,Person_Id,Vaccine_Id,Address_Id,Date)

CanPublish(Publish_Id,Officer_Id,Report_Id,Channel)

StayingIn(Staying_Id,Room_Id,Patient_Id,StartingDate)



SQL QUESTIONS

1. Who are persons over the age of 20 who have not been vaccinated?

Ans: if we look at if a person in isvaccinated table and his age over than 20, we have the result

Age için indexing

```
select Name,Surname,age
from person
where age>20 and (Person_Id) not in
      (select Person_Id
       from isvaccinated
      );
```

$\Pi\{\text{Name,Surname,Age} \mid \sigma\{\text{age}>20 \wedge \text{Person_Id} \notin (\Pi\{\text{Person_Id}\}(\sigma(\text{isvaccinated})))\}(\text{Person})\}$

2. How many empty beds are in which rooms in the hospital?

Ans: if we look at all beds which are accomadated and if we took difference, we arrive answer

```
select Room_Id,BedNumber-1 as availableRoomNumber ,RoomType
from hospitalroom
where (Room_Id) in
      (select Room_Id
       from stayingin
      ) union
select Room_Id,BedNumber as availableRoomNumber,RoomType
from hospitalroom
where (Room_Id) not in
      (select Room_Id
       from stayingin
      );
```

$\Pi\{\text{Room_Id,BedNumber-1} \mid \sigma\{\text{Room_Id} \in (\Pi\{\text{Room_Id}\}(\sigma(\text{stayingin})))\}(\text{hospitalroom})\} \cup \Pi\{\text{Room_Id,BedNumber} \mid \sigma\{\text{Room_Id} \notin (\Pi\{\text{Room_Id}\}(\sigma(\text{stayingin})))\}(\text{hospitalroom})\}$

3. Who are the people with the mutated virus?

Ans: mutated virus is not the original one so in that case if we check for that, we arrive result

```
select Name,Surname,CoronaVariant from hascorona inner join person on
person.Person_Id=hascorona.Person_Id where CoronaVariant !="original";
```

```
 $\Pi_{\{Name,Surname,CoronaVariant\}}(\sigma_{\{CoronaVariant \neq "original"\}}(\text{hascorona} \bowtie_{\text{person.Person\_Id=hascorona.Person\_Id}} \text{person}))$ 
```

4. What are the names and conditions of people who are not in the hospital although they have corona?

Ans:we will choose patients which are not in hospital.

```
select Name,Surname,age,Status
from patient inner join person on patient.Patient_Id=person.Person_Id
where (Patient_Id) not in
      (select Patient_Id
       from stayingin
       );
;
```

```
 $\Pi_{\{Name,Surname,Age,Status\}}(\sigma_{\{Patient\_Id \neq (\Pi_{\{Patient\_Id\}}(\sigma_{\{stayingin\}}))\}}(\text{Patient} \bowtie \text{person}))$ 
```

5. What is the ratio of people who have been quarantined so far to general persons?

Ans:we will divide the number of people who in quarantine to all people

```
SELECT
  (CAST((SELECT COUNT(*) FROM person WHERE Quarantine_Status='yes') AS FLOAT)/
   CAST((SELECT COUNT(*) FROM person) AS FLOAT)*100)
  AS QuarantineRatio;
```

```
 $\sigma_{\{CAST(\Pi_{\{count(*)\}}(\sigma_{\{Quarantine\_Status='yes'\}}(\text{person}))) / \sigma_{\{CAST(\Pi_{\{count(*)\}}(\sigma_{\{\text{person}\}}))\}}\}}$ 
```

6. Who are the people who have corona but are still not vaccinated and what are their conditions?

Ans:we will look the people who have corona but couldn't have vaccine.

```

select Name,Surname,age,Status
from patient inner join person on patient.Patient_Id=person.Person_Id
where (Person_Id) not in
      (select Person_Id
       from isvaccinated
       );
;

```

$\Pi\{Name, Surname, Age, Status\}(\sigma\{Person_Id \neq (\Pi\{Person_Id\}(\sigma(isvaccinated)))\}(Patient \bowtie person))$

7. How many people are in the system in total, how many of them are corona, how many of them are vaccinated and finally, how many hospital rooms are used?

Ans: we will gather total information

```

SELECT count(*) as TotalPerson,
      (SELECT Count(*) FROM isvaccinated) as TotalVaccinated, (SELECT Count(*) FROM hascorona)
as TotalCorona, (SELECT Count(*) FROM stayingin) as TotalUsedHospitalRoom
FROM person
;

```

$\Pi\{\Pi\{count(*)\}(\sigma(person)), \Pi\{count(*)\}(\sigma(isvaccinated)), \Pi\{count(*)\}(\sigma(hacorona)), \Pi\{count(*)\}(\sigma(stayingin))\}(Person)$

8. how many people ill in rize city?

Ans: we will gather data to specific area

```

select City,count(*) as NumberOfIllPeople from person inner join address on person.Address_Id
=address.Address_Id where city="Rize" and (Person_Id) in (select Person_Id from hascorona);

```

$\Pi\{City, count(*)\}(\sigma\{city="Rize" \wedge Person_Id \in (\Pi\{Person_Id\}(\sigma(hacorona)))\}(Address \bowtie Person))$

9. how many reports did the chosen person published until now and what is the name, surname and department of this person?

Ans: we will count by looking to canpublis table

```

select name,surname,OfficeName,OfficerRank,(SELECT Count(*) FROM canpublish inner join

```

officer inner join person on person.Person_Id=officer.Officer_Id and
officer.Officer_Id=canpublish.Officer_Id where Name="anil") as SharedReports from officer inner
join person on person.Person_Id = officer.Officer_Id where Name="anil";

$\Pi\{\text{name,surname,OfficeName,OfficerRank}\}(\Pi\{\text{count(*)}\}(\sigma\{\text{Name="anil"}\}(\text{canpublish} \bowtie \text{officer} \bowtie \text{person})))$

10. how many people didn't get corona even they contacted with ill people?

Ans: we will count negative results from contacts

select count(*) as DidntGet from contacted where TestResult="negative";

$\Pi\{\text{count(*)}\}(\sigma\{\text{TestResult="negative"}\}(\text{contacted}))$

Raid Level Choice:

If I wanted to use my project on multiple drives, I would use raid 6 to achieve this. Data created by increasing the disk usage of raid 5, parity data is written to 4 drives by using raid 6. Thanks to this system, a single drive crash will not put my system in trouble, my system will be able to withstand up to 2 drive crashes. By using 4 drives, the possibility of losing data is reduced, while the speed of writing to the disc is reduced.

I think Raid 6 is suitable for my system because data read speeds are very high in raid 6. The data writing speed is lower than normal, but considering my system, I will have entered the data of every citizen, doctor's data, hospital rooms in the first place, like the e-devlet application. I will actively use data writing only when enrolling patients. I thought, if I put 80 million data in the system like e-devlet first, and if I increase the security and keep the reading process fast, creating a patient record will be much easier. To put it simply, I will enter 80 million of them into the system because this database will be used by the state and every citizen's information must be in the system because as we know, a state is responsible for protecting the health of its citizens. If we look at the event from this perspective, five thousand and ten thousand new data to be entered into the daily system is very small. The reason for this is that if the raid level was lower, yes, the insertion speed will be higher, but the process of entering person and patient information will still be 30 thousand 40 thousand lines. and in this case the risk of losing data is very high and if patient data is lost it will be much more difficult to recover from paper documents and serious problems will arise in vaccination and hospital stay situations.

In summary, the raid 6 level is to protect the efficiency and to return the desired data as soon as possible by performing fast reading operations on the database. If we consider that the data to be entered into the system is the patient record information, I would prefer to

give up the fast data writing and build my system on security. In the event of possible driver failure, I will have a much higher chance of recovering efficiency and while providing this, I will not sacrifice the read speed, which is the most important thing for me.

Indexing:

Question 1: Who are the persons over the age of 20 who have not been vaccinated?

Ans: if we look at if a person in isvaccinated table and his/her age over than 20, total is the answer

```
select Name,Surname,age
from person
where age>20 and (Person_Id) not in
      (select Person_Id
       from isvaccinated
      );
```

This is the query that I will work on.

I will create index for age attribute which is in person table.

Index Definition:

```
CREATE INDEX age_index ON person (age asc);
```

Index Explanation:

Above, you can see index creation.

In this part, I created an index for age because the age information is not related to the search made from the primary key. To explain better, the primary key order goes from smallest to largest, but the ages in the rows do not go sequentially. When I want to reach people in a certain age range, the time spent to find the specified age range increases exponentially, as the primary key is searched row by row.

In order to overcome this problem, I set up an index table for the age attribute, which is not a primary key, in person and put them in ascending order in this table, making it easier to find people in a certain age range.

Question 2: How many people ill in Çayeli district?

Ans: we will gather data to specific district such as number of case

```
select City,count(*) as NumberOfIllPeople from person inner join address on person.Address_Id
=address.Address_Id where district="Çayeli" and (Person_Id) in (select Person_Id from
hascorona);
```

Index Definition:


```
CREATE INDEX district_index ON address (district asc);
```

Index Explanation:

I created an index for district because sometimes we have to focus on certain regions in order to make a corona case report. For example, in order to find out how many patients there are in Çayeli district, we must first find the people whose district is Çayeli, and when we do this, the first thing we notice is that the number of districts is very, very large and it takes a lot of time to find the person sought from among so many districts.

For this reason, I am trying to minimize the search time by creating an index for the district in the address table by sorting the districts in alphabetical order.

Question 3: Who are the people with the England variation of Coronavirus?

Ans: We will look for person who has England variant.

Index Definition:

```
CREATE INDEX variant_index ON hascorona (CoronaVariant asc);
```

Above you can see creation of index.

```
select Name,Surname,CoronaVariant from hascorona inner join person on  
person.Person_Id=hascorona.Person_Id where CoronaVariant ="England";
```

Index Explanation:

I add coronavariant as an index because when reporting corona cases, I want to know who the people with a certain variant are, just like the ministry of health does. This is because there are different variants such as England, Brazil, Egypt, Argentina, India, and the process takes a long time when a particular variant is searched.

In order to achieve this goal, I will index the coronavariant and keep it in alphabetical order. Thanks to this process, in case of my next variant search, I will receive results much faster.

Question 4: How many people ill in Rize City?

Ans: we will gather data to specific city such as number of ill people

```
select City,count(*) as NumberOfIllPeople from person inner join address on person.Address_Id  
=address.Address_Id where city="Rize" and (Person_Id) in (select Person_Id from hascorona);
```

Index Definition:

```
CREATE INDEX city_index ON address (city asc);
```

Index Explanation:

I created an index for city same reason which is applied for district case. sometimes we have to focus on certain City in order to make a corona case report. For example, in order to find out how many patients there are in Rize City, we must first find the people whose City is Rize, and when we do this, the first thing we notice is that the number of Cities are not too much but we are using them too much and if we add index for city we can increase the speed of City search.

For this reason, I am trying to minimize the search time by creating an index for the City attribute in the address table by sorting the Cities in alphabetical order.

Question 5: What is the health condition of specified people who are in the hospital?

Ans: we will choose specified patient which is in hospital and show his/her status.

```
select Name,Surname,age,Status
from patient inner join person on patient.Patient_Id=person.Person_Id
where Name="Ferhat" and Surname="Ak" and (Patient_Id) in
    (select Patient_Id
    from stayingin
    );
;
```

Index Definitions:

```
CREATE INDEX name_surname_index ON person (Name,Surname asc);
```

Index Explanations:

Let's say the relative of the sick person came to the reception to learn the condition of the sick person. In this case, in order to inform the incoming person, this person has to be called by name and surname in the system. Since there are too many people in the hospital, it takes a long time to find the desired person in the database. In order to make this process faster, name and surname indexes can be created and these attributes can be sorted alphabetically, and the information about the submitted person reaches the desired person before long.

This process is of great importance because hundreds of people ask the reception about the status of their relatives, and if an index of the fields with this name and surname is created in alphabetical order, information about the patients can be obtained in a very short time.

INTERFACE PART

The screenshot shows a web interface for 'Corona Must Go'. At the top is a blue header bar with the text 'Corona Must Go' in white. Below the header, the word 'Login' is centered in a large, bold, black font. Underneath 'Login', there are two labels: 'Name' and 'Password'. To the right of 'Name' is a text input field containing the word 'user'. To the right of 'Password' is a password input field represented by a series of black dots. Below these input fields, there are three buttons. The first two are 'Login' and 'Officer Login', positioned side-by-side. Below them is a single, wider button labeled 'Learn Your Password'.

This is the login screen. In our system person can create its own person account but in order to get an officer account, person should contact with the medicine ministry to get officer account to access all data, except that one, a person can enter to system and can see overall status.

Person Informations

Name

Surname

Age

City

District

Test Result

Quarantine Status

Create Person

In this part person can create his account with filling these informations which are listed in photo.

Overall Status

Overall Illness Status

	TotalPerson	TotalVaccinated	TotalCorona	TotalUsedHospitalRoom
▶	43	9	10	8

Number Of Quarantine People

	QuarantinePeople
▶	10

Number OF Contacted People

	ContactedPeople
▶	10

This screen is provided to show everyone the overall status about coronavirus disease. This page is accessible by all persons' accounts and they can see the overall status about the disease.

Person's Status

Have Corona/Not Staying Hospital

	Name	Surname	age	Status
▶	Numan	Uzun	50	alive
	Beyza	Demirci	27	alive

Have Corona/Not Vaccinated

	Name	Surname	age	Status
▶	Asim	Ayten	75	alive
	Ferhat	Ak	55	alive
	Aybala	Kara	65	alive
	Numan	Uzun	50	alive
	Beyza	Demirci	27	alive

Have Corona/Corona Variants

	Name	Surname	CoronaVariant
▶	Kasim	Yildiz	england
	Asim	Ayten	england
	Ferhat	Ak	brazil
	Aybala	Kara	england
	Akif	Azman	brazil
	Numan	Uzun	england
	Beyza	Demirci	brazil

This page is for the officers. Only officers can access that page and they will monitor about person's status in that page and they will take actions according to these datas. Actually there are very important tables which states some emergency problems and they will take action immediately.

About You

General Info

	Name	surname	age	TestResult	IsInQuarantine
▶	Anil	irfanoğlu	20	negative	no

Vaccination

	Date	VaccineName
▶	08/05/2021	Sinovac

Case Number in Your City

	City	NumberOfIllPeople
▶	Rize	2

Chracteristic Informations

	name	surname	OfficeName	OfficerRank	SharedReports
▶	Anil	irfanoğlu	Healthcare HQ	Head of Department	2

In this page, a person can see information about itself and about his status. Whoever you are, you can see you general informations and also you can look at your characteristic informations for example if you are an officer, you can check your officer informations and you can look at how many magazine did you publish , if you doctor, you could see the number of patients that you look and also you can access the informations about them.

References:

<https://dataschool.com/sql-optimization/how-indexing-works/>