**Assignment 5**

As a database designer, you have created a database for two domains according to the requirements. Now, you have to write the queries from different stakeholder’s perspectives.

For example:

For health care, you have created entities like doctors, department, appointment and inserted data into those tables. Now as a user, I might have following queries:

* List all the doctors
* Book an appointment
* Schedule an existing appointment for a later date
* View all my appointments

Similar to these examples, create all the possible queries that exist in your system.

**Health Care**

* **Requirement Gathering :**

**Request for Proposal (RPF):**

An Online Health Care system should be made. It should consists the details of registered Clinics, Hospitals and Pharmacies with their provided services. Users can register as a Doctor, Patient , Clinic, Hospital or Pharmacy. The registered Doctors should be listed in the site along with registered clinics,hospitals and pharmacies. User can have appointment to clinic,hospital or doctor. There can also be consultation with doctor directly. The billing can be done through payment like esewa to registered clinic/hospital/doctor/pharmacies. There can also be Discussion section to write about the problems related to users.

**Collecting Information:**

When collecting information from key stakeholders, it is essential to focus on two main questions that should serve as guidelines for the overall requirements gathering process.

1. How can an this system contribute to your practice’s strategic goals?
2. What features are the best for achieving your practice’s strategic goals?

So, this may include meeting sample like:

Meeting Notes

Date: Sept 7, 2021

Agenda:

* Discussion about the proposed site.
* Cost finalization.
* Gathering info for design.

Meeting Notes:

1. User friendly site to be made.

Decision Made:

1. Project Accepted.
2. Design for site finalized.

Action Items:

Discovery Questions

|  |  |
| --- | --- |
| **Questions/ Thoughts** | **Answers** |
| Who will be using the new system? | It should be focused to give public a online health care options. So, the system is frequently used by patients , doctors, pharmacies and admin to update regarding new registered details or for removing existing details. |
| So the users can register using any of the options. But how will be the authenticity be done if the doctor/pharmacy/clinic registered directly ? So I recommend admin verify before these registration.. do you consider this ? | Yeah definitely, it would be better if the users are real and valid. So, that should be done. |
| What patient population are we documenting? (i.e., pediatrics, geriatrics, patients, etc.) | There can be different categories of patients obviously i.e pediatrics, geriatrics but all can be kept under patients. |
| Should the payment system be other than esewa? | Since esewa is popular, it should be used as a payment method. Other payments would be plus if they could be add`ed like khalti. |
| What database to use? | MySQL or Posgres, any! |

**Analyzing and drafting requirements**

After Analyzing , the entities may include:

* Hospitals , Clinics, Pharmacies,Doctor, Patients can be registered
* Services are provided by Hospitals, Clinics, Doctors
* Patient can appoint to Hospitals, Clinics, Doctors
* Patient can consult to specific Doctor
* Patient can give feedback to doctor
* Patient can open conversation in Discussion section
* Payments be done by patients to hospital, pharmacies, doctors
* History is maintained for users
* Admin verifies Hospital, Clinic, Doctor and Pharmacy registration
* Admin can modify registered user details, payments and other features
* **Conceptual Modeling**

1. Identifying Entities:

clinic - clinic\_id ,name,location

hospital - hospital\_id, name, location

doctor - doctor\_id, doctor\_name, phone\_no,qualification, specialization(service)

services - service\_id, name ,cost

pharmacy - pharmacy\_id,name, location

medicine - med\_id, name, brand,cost

patient - patient\_id , name, age, sex, address ,phone, blood\_group

specialization - specialization\_id, specialization\_name

hospital\_service - hospital\_service\_id , patient\_id ,hospital\_id ,service\_id , amount, service\_date

clinic\_service - clinic\_service\_id, patient\_id,clinic\_id,service\_id,amount,service\_date

appointment - appointment\_id , patient\_id , date

consultation - c\_id, doctor\_id, date,patient\_id

feedback - feedback\_id, description,doctor\_id, patient\_id

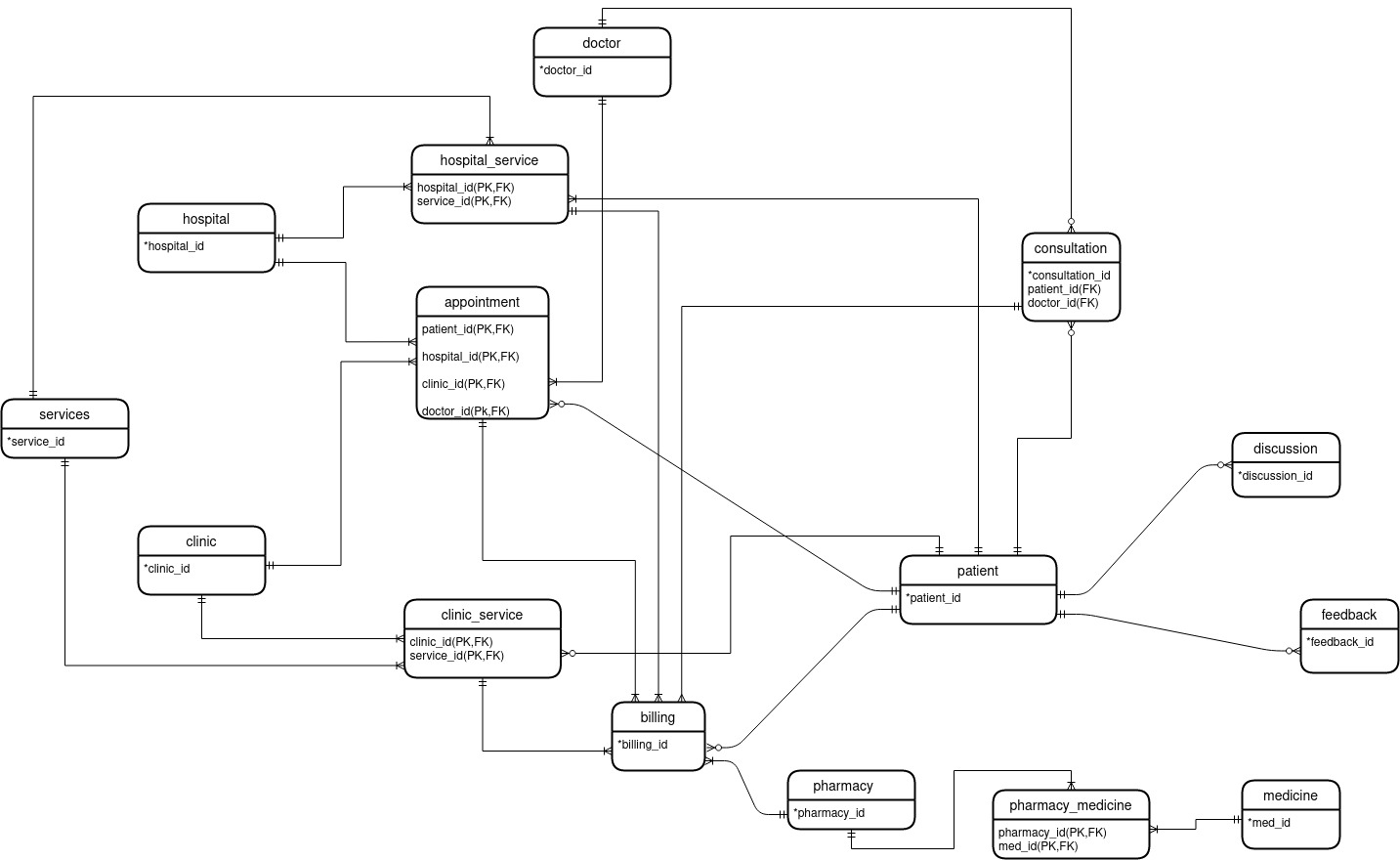
discussion - discussion\_id,patient\_id, question,answer

billing - billing\_id, amount , type

2. Defining business rules

* Hospitals , Clinics, Pharmacies,Doctor, Patients can be registered
* Services are provided by Hospitals, Clinics, Doctors
* Patient can appoint to Hospitals, Clinics, Doctors
* Patient can consult to specific Doctor
* Patient can give feedback to doctor
* Patient can open conversation in Discussion section
* payments be done by patients to hospital, pharmacies, doctors
* History is maintained for users
* Admin verifies Hospital, Clinic, Doctor and Pharmacy registration
* Admin can modify registered user details, payments and other features

3. Conceptual Model :



* **Logical Modeling**

1. Identifying attributes and domains for entities and relationships:

|  |  |  |
| --- | --- | --- |
| **Entity** | **Description** | **Domain** |
| **hospital** | Hospital information |  |
| Attributes:  hospital\_id  name  location | Identifier for user entity, SK, PK  Name of hospital  Location of hospital | Auto Generated  Text  Text |
| **clinic** | Clinic information |  |
| Attributes:  clinic\_id  name  location | Identifier for user entity, SK, PK  Name of clinic  Location of clinic | Auto Generated  Text  Text |
| **pharmacy** | Pharmacy information |  |
| Attributes:  pharmacy\_id  name  location | Identifier for user entity, SK, PK  Name of pharmacy  Location of pharmacy | Auto Generated  Text  Text |
| **doctor** | Doctor information |  |
| Attributes:  doctor\_id  name  phone\_no  qualification  specialization  service\_id | Identifier for user entity, SK, PK  Name of doctor  Valid phone no. of doctor  Qualification of doctor  Specialization info  Service Code (FK) | Auto Generated  Text  Number  Text  Text  Valid id from service table |
| **services** | Services provided for health care |  |
| Attributes:  service\_id  name  cost  patient\_id | Identifier for user entity, SK, PK  Name of service  Price of service  Patient Code | Auto Generated  Text  Valid Number  Valid Code from patient table |
| **appointment** | Relation for patient appointments |  |
| Attributes:  patient\_id  hospital\_id  clinic\_id  doctor\_id  date | Patient Code (PK, FK)  Hospital Code (PK, FK)  Clinic Code(PK,FK)  Doctor Code(PK,FK)  Date of appointment | Valid patient id from patient table  Valid id from hospital table  Valid id from clinic table  Valid id from doctor table  Date |
| **patient** | Patient Details |  |
| Attributes:  patient\_id  name  dob  phone\_no  location  blood\_group | Identifier for user entity, SK, PK  Name of patient  Date of Birth  Valid Phone number  Address  Blood Group | Auto Generated  Text  Date  Number  Text  Text |
| **consultation** | Relation for consulting doctor |  |
| Attributes:  patient\_id  doctor\_id | Patient Code (PK,FK)  Doctor Code (PK,FK) | Valid id from patient table  Valid id from doctor table |
| feedback | Feedback give by patient to doctor |  |
| Attributes:  patient\_id  doctor \_id  description | Patient Code(PK,FK)  Doctor Code (PK,FK)  Description given by patient | Valid id from patient table  Valid id from doctor table  Text |
| **discussion** | Discussion about patient problems |  |
| Attributes:  patient\_id  description | Patient Code(PK,FK)  Discussion opened by patient | Valid id from patient table  Text |
| **billing** | Payment Details |  |
| Attributes:  billing\_id  type  amount | Identifier for user entity, SK, PK  Type of pay  Valid amount | Auto Generated  Text  Number |
| **medicine** | Medicine Details |  |
| Attributes:  medicine\_id  name  brand  cost | Identifier for user entity, SK, PK  Name of medicine  Brand of medicine  Price of medicine | Auto Generated  Text  Text  Number |

2. Logical Mode:

<https://drive.google.com/drive/folders/1BQbZpGRJ17VY9wNK3QT1x-xRBJT3DLnp?usp=sharing>

* **Physical Model**

DROP DATABASE if exists healthcare;

CREATE DATABASE healthcare;

**-- Creating necessary TABLE FROM logical model**

**--city**

CREATE TABLE city(

id serial primary key,

city\_code INT not null,

city\_name VARCHAR(50) not null

);

SELECT \* FROM city;

INSERT INTO city values

(1,44600,'Kathmandu'),

(2,44800,'Bhaktapur'),

(3,44700,'Lalitpur');

**--hospital**

create table hospital (

hospital\_id serial primary key,

name varchar(50) not null,

location int references city(id)

);

SELECT \* FROM hospital;

INSERT INTO hospital values

(1,'Kantipur Hospital',1),

(2,'KMC Hospital',1),

(3,'Lalitpur Hospital',3),

(4,'Ivamura Hospital',2),

(5,'Patan Hospital',3),

(6,'Civil Hospital',1),

(7,'Bhaktapur Hospital',2);

**--clinic**

CREATE TABLE clinic(

clinic\_id serial PRIMARY KEY,

name VARCHAR(50) not null,

location INT REFERENCES city(id) not null

);

SELECT \* FROM clinic;

INSERT INTO clinic values

(1,'Kantipur Clinic',1),

(2,'Hamro Clinic',2),

(3,'Lalitpur Clinic',3),

(4,'Light Clinic',2),

(5,'Patan Clinic',3),

(6,'Rajdhani Clinic',1);

**--pharmacy**

DROP table pharmacy ;

CREATE TABLE pharmacy(

pharmacy\_id serial PRIMARY KEY,

pharmacy\_name VARCHAR(50) not null,

location INT REFERENCES city(id) not null

);

SELECT \* FROM pharmacy;

INSERT INTO pharmacy values

(1,'Ramro Pharmacy',2),

(2,'Ramro Pharmacy',1),

(3,'Lalit Pharmacy',2);

**--medicine**

CREATE TABLE medicine(

med\_id serial PRIMARY KEY,

name VARCHAR(50) not null ,

brand VARCHAR(50) not null,

cost INT not null

);

SELECT \* FROM medicine ;

INSERT INTO medicine values

(1,'Verocil','VDD',1200),

(2,'Tablet','Nickto',500),

(3,'Cliveng','Sancho',2000),

(4,'Pilera','DDD',5000),

(5,'Saee','DDD',6666);

**--services**

CREATE TABLE services(

service\_id serial PRIMARY KEY,

service\_name VARCHAR(50) not null,

cost INT not null

);

SELECT \* FROM services;

INSERT INTO services values

(1,'Covid Check',10000),

(2,'Psychatricst Help',5000),

(3,'Dental Help',5000),

(4,'Skin Care',6000),

(5,'Skeleton Help',10000);

**--specialization**

CREATE TABLE specialization(

specialization\_id serial PRIMARY KEY,

specialization\_name VARCHAR(50) not null

);

SELECT \* FROM specialization;

INSERT INTO specialization values

(1,'Skeleton'),

(2,'Skin'),

(3,'Dental'),

(4,'Heart'),

(5,'Hair'),

(6,'Kidney');

**--doctor**

CREATE TABLE doctor(

doctor\_id serial PRIMARY KEY,

name VARCHAR(50) not null,

phone\_no INT unique,

qualification VARCHAR(50) not null,

specialization INT REFERENCES specialization(specialization\_id),

service\_id INT REFERENCES services(service\_id)

);

SELECT \* FROM doctor;

INSERT INTO doctor values

(1,'Ram Giri',985100000,'MBBS',3,3),

(2,'Prem Prajapati',984300000,'MBBS',2,4),

(3,'Mohan Lal',984333333,'MBBS',4,2),

(4,'Sita Siwakoti',984156656,'MD',1,5),

(5,'Sa Shah',984332404,'MD',6,null);

**--patient**

CREATE TABLE patient(

patient\_id serial PRIMARY KEY,

name VARCHAR(50) not null,

dob DATE not null,

phone INT unique,

location INT REFERENCES city(id),

blood\_group VARCHAR(10) not null

);

SELECT \* FROM patient ;

INSERT INTO patient values

(1,'Ram shah','1999-03-01',984454,1,'A+'),

(2,'Raju lama','1970-08-01',90050,1,'AB+'),

(3,'Sam Ale','1989-03-11',5064,2,'B-'),

(4,'Rita Tle','2000-03-01',152054,2,'A-'),

(5,'Sajana Twane','1994-07-01',15154,3,'O+');

**--------------------------------hospital\_service**

CREATE TABLE hospital\_service(

hospital\_service\_id serial primary key,

patient\_id int references patient(patient\_id),

hospital\_id INT REFERENCES hospital(hospital\_id),

service\_id INT REFERENCES services(service\_id),

unique(patient\_id,hospital\_id, service\_id),

amount int not null,

service\_date date not null

);

SELECT \* FROM hospital\_service ;

INSERT INTO hospital\_service values

(1,1,1,1,5000,'2021-09-12'),

(2,2,3,1,10000,'2021-09-10'),

(3,4,1,3,3000,'2021-09-01'),

(4,2,2,4,4000,'2021-09-13');

**------------------------------clinic\_service**

CREATE TABLE clinic\_service(

clinic\_service\_id serial primary key,

patient\_id int references patient(patient\_id),

clinic\_id INT REFERENCES clinic(clinic\_id),

service\_id INT REFERENCES services(service\_id),

unique(patient\_id,clinic\_id, service\_id),

amount int not null,

service\_date date not null

);

SELECT \* FROM clinic\_service;

INSERT INTO clinic\_service values

(1,2,1,3,2000,'2021-09-03'),

(2,5,1,4,3000,'2021-09-14'),

(3,1,2,4,5000,'2021-08-12');

**--consultaion**

CREATE TABLE consultation(

consultation\_id serial PRIMARY KEY,

patient\_id INT REFERENCES patient(patient\_id),

doctor\_id INT REFERENCES doctor(doctor\_id),

consultation\_date DATE not null,

amount int not null

);

SELECT \* FROM consultation ;

INSERT INTO consultation values

(1,1,3,'2021-09-12',3000),

(2,3,4,'2021-09-10',2000),

(3,4,5,'2021-09-13',5000)

**-- discussion**

CREATE TABLE discussion(

discussion\_id serial PRIMARY KEY,

patient\_id INT REFERENCES patient(patient\_id),

question VARCHAR(200),

solution VARCHAR(200)

);

SELECT \* FROM discussion;

INSERT INTO discussion values

(1, 2, 'How to prevent common cold?','Stay Away FROM cold!'),

(2, 4,'Where is Bir hospital?','Kathmandu.'),

(3,5,'How to stop chicken-pox?','Maintain hygiene.');

**--feedback**

CREATE TABLE feedback(

feedback\_id serial PRIMARY KEY,

patinent\_id INT REFERENCES patient(patient\_id),

doctor\_id INT REFERENCES doctor(doctor\_id),

description VARCHAR(300)

);

SELECT \* FROM feedback;

INSERT INTO feedback values

(1, 1, 1 ,'Great consultant!'),

(2,5, 3,'Happy with your servies!');

**-- appointment**

CREATE TABLE appointment(

appointment\_id serial PRIMARY KEY,

patient\_id INT REFERENCES patient(patient\_id),

hospital\_id INT REFERENCES hospital(hospital\_id),

clinic\_id INT REFERENCES clinic(clinic\_id),

doctor\_id INT REFERENCES doctor(doctor\_id),

appointment\_date DATE not null,

cost INT not null

);

SELECT \* FROM appointment;

INSERT INTO appointment values

(1,2,3,null,null,'2021-02-10',4000),

(2,3,null,null,3,'2021-04-05',5000),

(3,1,null,3,null,'2021-03-11',2000);

**--pharmacy\_medicine**

CREATE TABLE pharmacy\_medicine(

id serial PRIMARY KEY,

pharmacy\_id INT REFERENCES pharmacy(pharmacy\_id),

med\_id INT REFERENCES medicine(med\_id)

);

SELECT \* FROM pharmacy\_medicine;

INSERT INTO pharmacy\_medicine values

(1,2,3),

(2,3,5),

(3,2,1),

(4,1,4),

(5,1,1),

(6,3,2);

**--billing**

CREATE TABLE billing(

billing\_id serial PRIMARY KEY,

patient\_id INT REFERENCES patient(patient\_id),

appointment\_id INT REFERENCES appointment(appointment\_id),

hospital\_service\_id INT REFERENCES hospital\_service(hospital\_service\_id),

clinic\_service\_id int references clinic\_service(clinic\_service\_id),

consultation\_id int references consultation(consultation\_id),

pharmacy\_id INT REFERENCES pharmacy(pharmacy\_id),

amount INT

);

SELECT \* FROM billing;

INSERT INTO billing values

(1,2,1,null,null,null,1,6000),

(2,1,null,null,null,2,null,10000);

**--------------------------- QUERIES**

**-- all patients**

SELECT \* FROM patient;

**-- all hospitals**

SELECT \* FROM hospital;

**-- all clinic**

SELECT \* FROM clinic;

**-- all pharmacies**

SELECT \* FROM pharmacy ;

**-- patient feedback**

SELECT \* FROM feedback f ;

SELECT feedback\_id ,

p."name" as Patient\_Name,

d."name" as Doctor,

description

FROM feedback f

JOIN patient p on

f.patinent\_id = p.patient\_id

JOIN doctor d on

d.doctor\_id = f.doctor\_id ;

**-- consultaion**

SELECT \* FROM consultation c ;

SELECT consultation\_id ,

p."name" as Patient,

d."name" as Doctor,

consultation\_date,

amount

FROM consultation c

JOIN patient p on

c.patient\_id = p.patient\_id

JOIN doctor d on

d.doctor\_id = c.doctor\_id

-- appointment to hospital

SELECT \* FROM appointment a ;

SELECT appointment\_id ,

p."name" as patient,

h."name" as Hospital,

appointment\_date ,

"cost"

FROM appointment a

JOIN patient p ON

a.patient\_id = p.patient\_id

JOIN hospital h on

h.hospital\_id = a.hospital\_id ;

**-- appointment to clinic**

SELECT \* FROM appointment a ;

SELECT appointment\_id ,

p."name" as patient,

c."name" as clinic ,

appointment\_date ,

"cost"

FROM appointment a

JOIN patient p ON

a.patient\_id = p.patient\_id

JOIN clinic c on

c.clinic\_id = a.clinic\_id ;

-- appointment to doctor

SELECT \* FROM appointment a ;

SELECT appointment\_id ,

p."name" as patient,

d.name as Doctor,

appointment\_date ,

"cost"

FROM appointment a

JOIN patient p ON

a.patient\_id = p.patient\_id

JOIN doctor d on

d.doctor\_id = a.doctor\_id ;

**-- patients using hospital services:**

SELECT \* FROM hospital\_service hs2 ;

SELECT hospital\_service\_id ,

p."name",

h."name" ,

s.service\_name ,

service\_date ,

amount

FROM hospital\_service hs

JOIN patient p on

hs.patient\_id = p.patient\_id

JOIN hospital h on

hs.hospital\_id = h.hospital\_id

JOIN services s ON

hs.service\_id = s.service\_id ;

**-- patients using clinic services**

SELECT \* FROM clinic\_service cs ;

SELECT clinic\_service\_id ,

p."name" ,

c."name" ,

s.service\_name ,

service\_date ,

amount

FROM clinic\_service cs

JOIN patient p on

cs.patient\_id = p.patient\_id

JOIN clinic c on

cs.clinic\_id = c.clinic\_id

JOIN services s on

s.service\_id = cs.service\_id;

**-- patients consulting doctors**

SELECT \* FROM consultation c ;

SELECT consultation\_id ,

p."name" as Patient ,

d."name" as Doctor ,

consultation\_date ,

amount

FROM consultation c

JOIN patient p on

c.patient\_id = p.patient\_id

JOIN doctor d on

d.doctor\_id = c.doctor\_id ;

**-- billing for pharmacy info**

SELECT \* FROM billing b ;

SELECT billing\_id ,

p."name" as Patient\_Name,

c.city\_name as patient\_address,

p2.pharmacy\_name ,

amount

FROM billing b

JOIN patient p on

p.patient\_id = b.patient\_id

JOIN city c on

p."location" = c.id

JOIN pharmacy p2 ON

b.pharmacy\_id = p2.pharmacy\_id ;