TINDOC

Group 2:
1. 122442 Pasit Tiwawongrut
2. 122149 Saratoon Khantasima
3. 122410 Nopphawan Nurnuansuwan
4. 122050 Vineela Mukkamala
5. 122556 Harold Popluhar

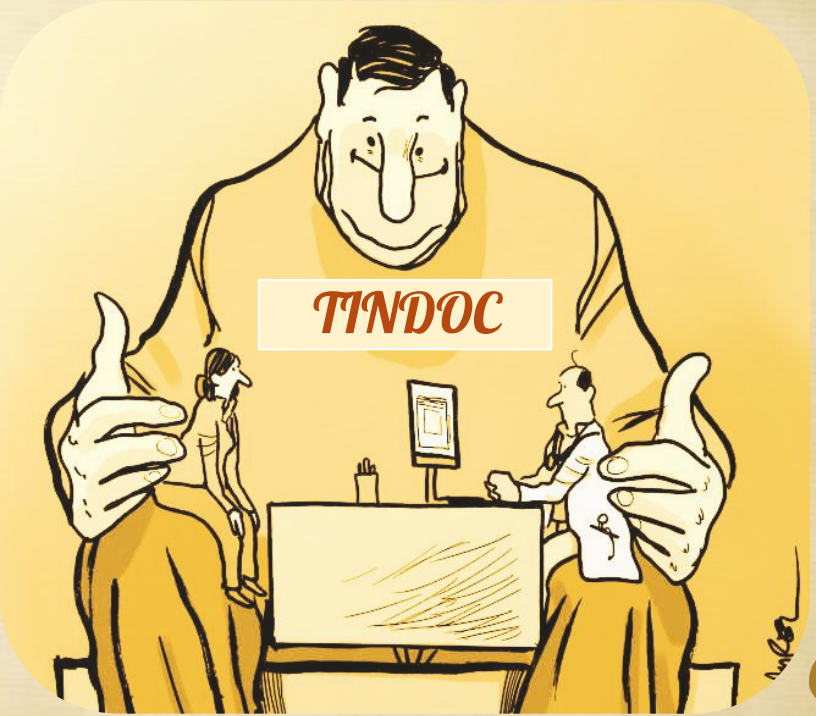
Contents

Why is it important and interesting ?

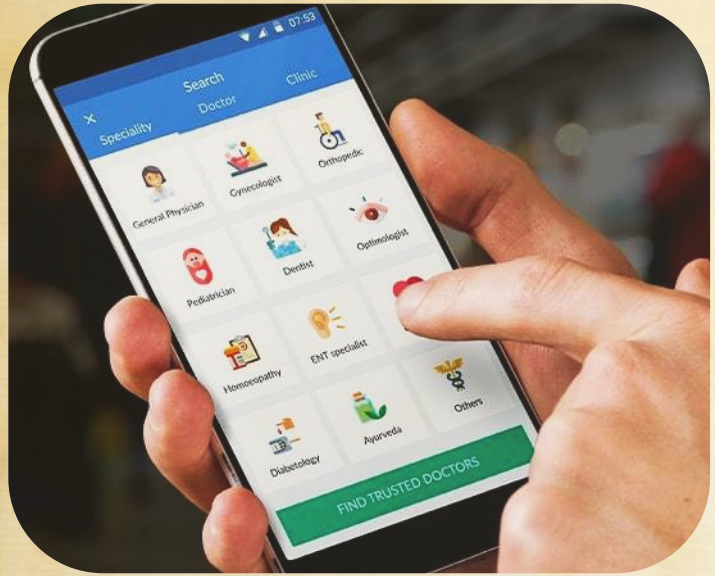
Covid has shaken up habits, especially in medical sector

more difficult to have an appointment with a doctor.

Tindoc



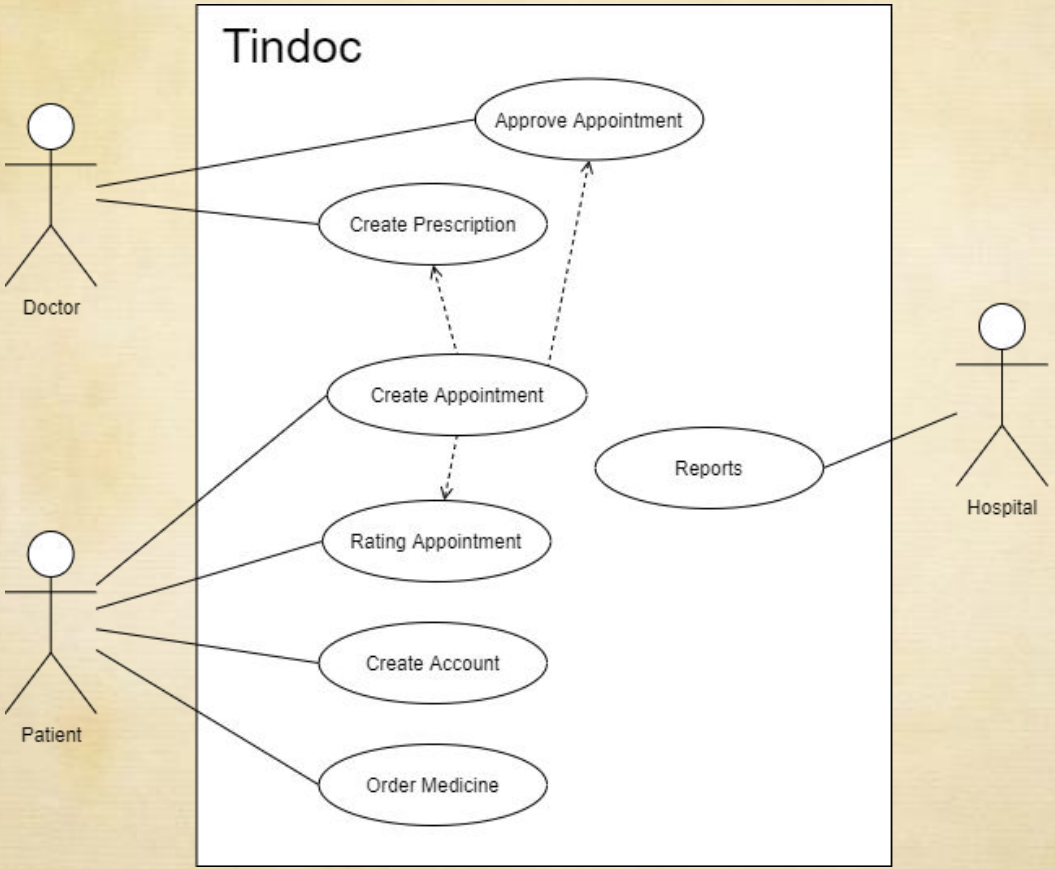
Project description



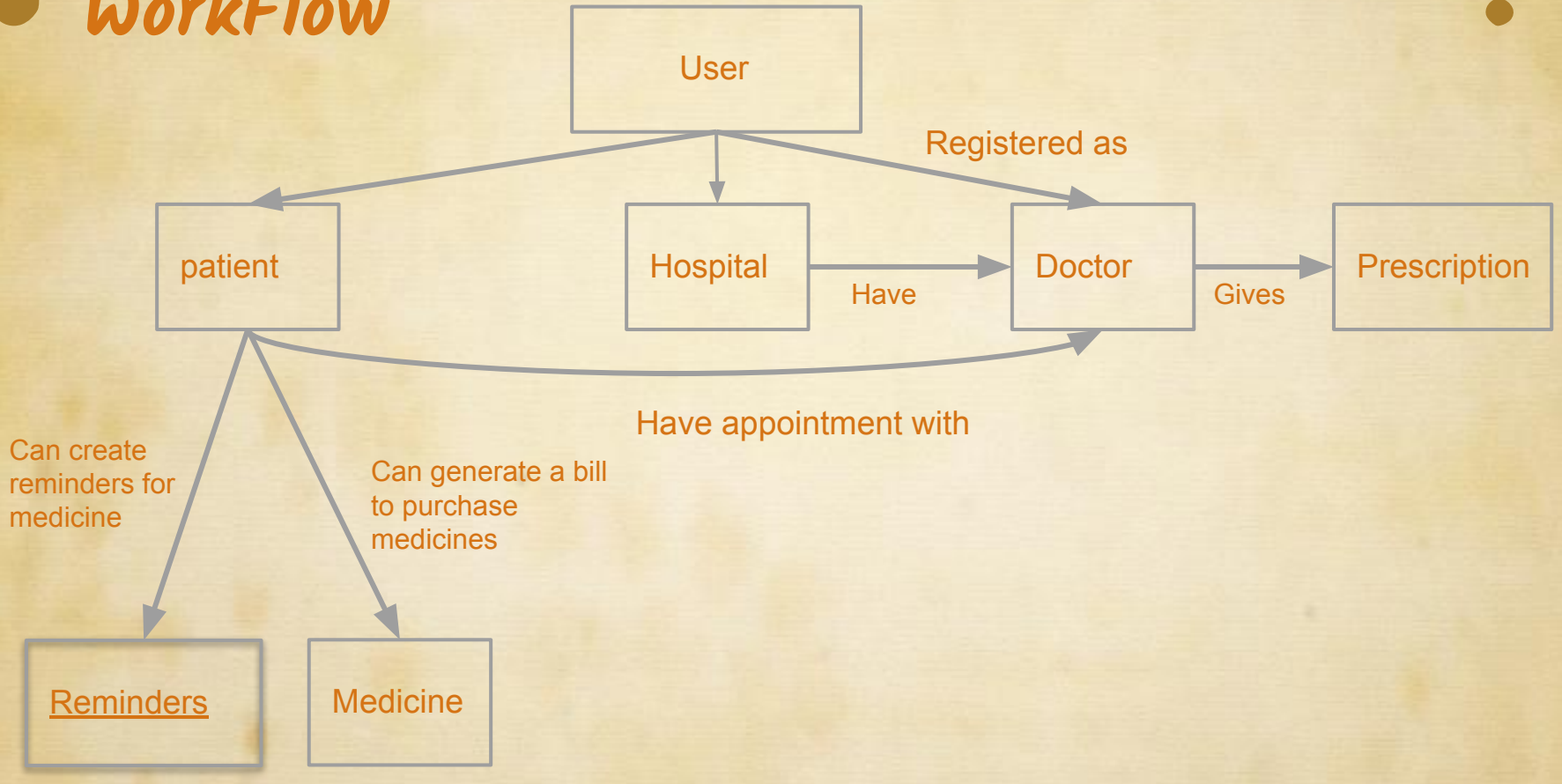
Tindoc:

- an appointment manager application in medical sector
- This application supports both online and offline appointment
- Users select their doctors according to their symptoms.
- After their appointments, patients will be able to rate it and generate a bill to purchase their medicines..

Use Case Diagram



WorkFlow


User
Registered as
patient
Hospital
Doctor
Prescription
Have
Gives
Have appointment with
Can create reminders for medicine
Can generate a bill to purchase medicines
Reminders
Medicine

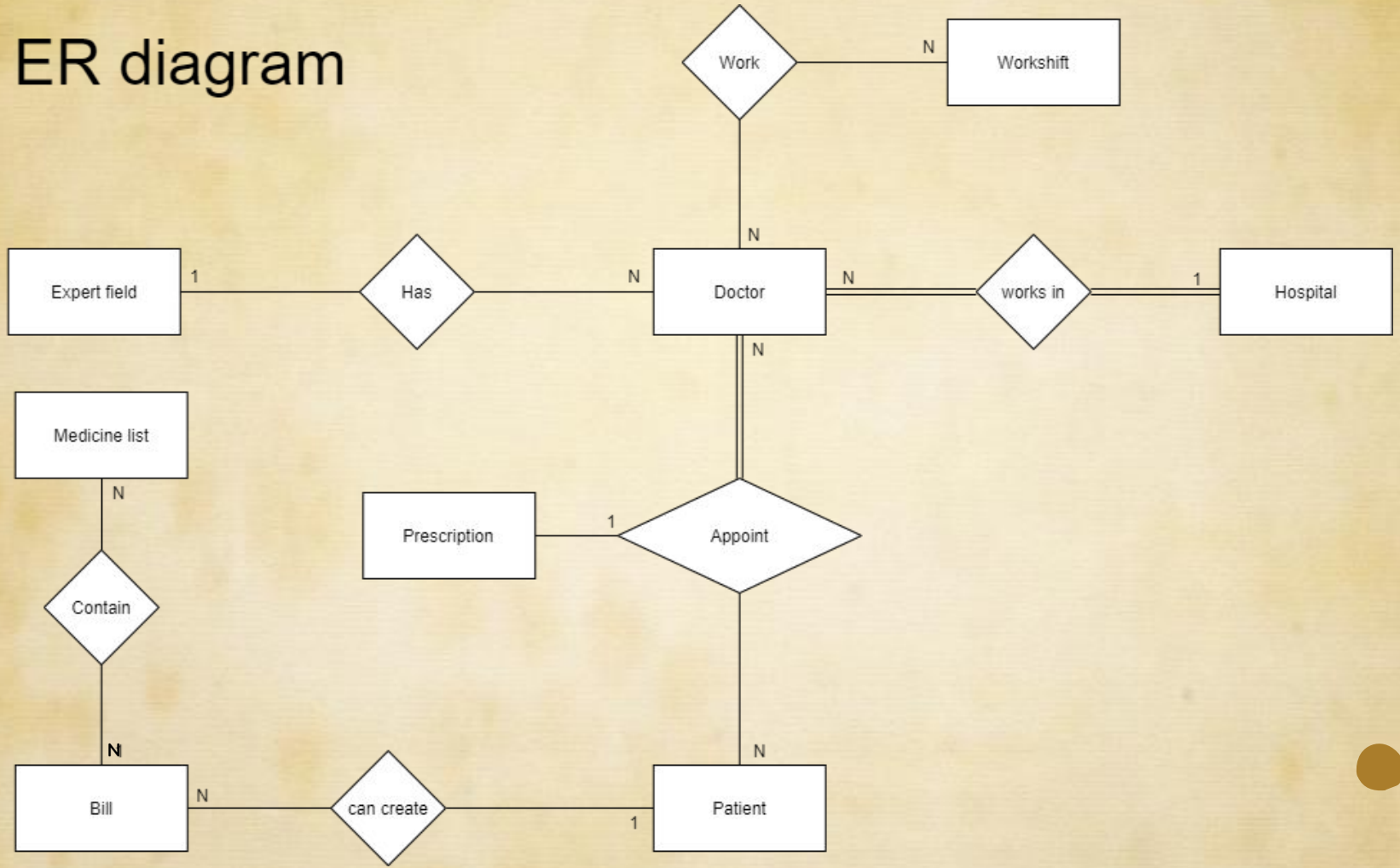
Data to be maintained

- Patients information
- Appointment detail
- Prescription detail
- Doctor information
- Hospital detail
- Expert field categories
- Doctor available hours
- Medicine list
- Bill item detail
- Bill detail

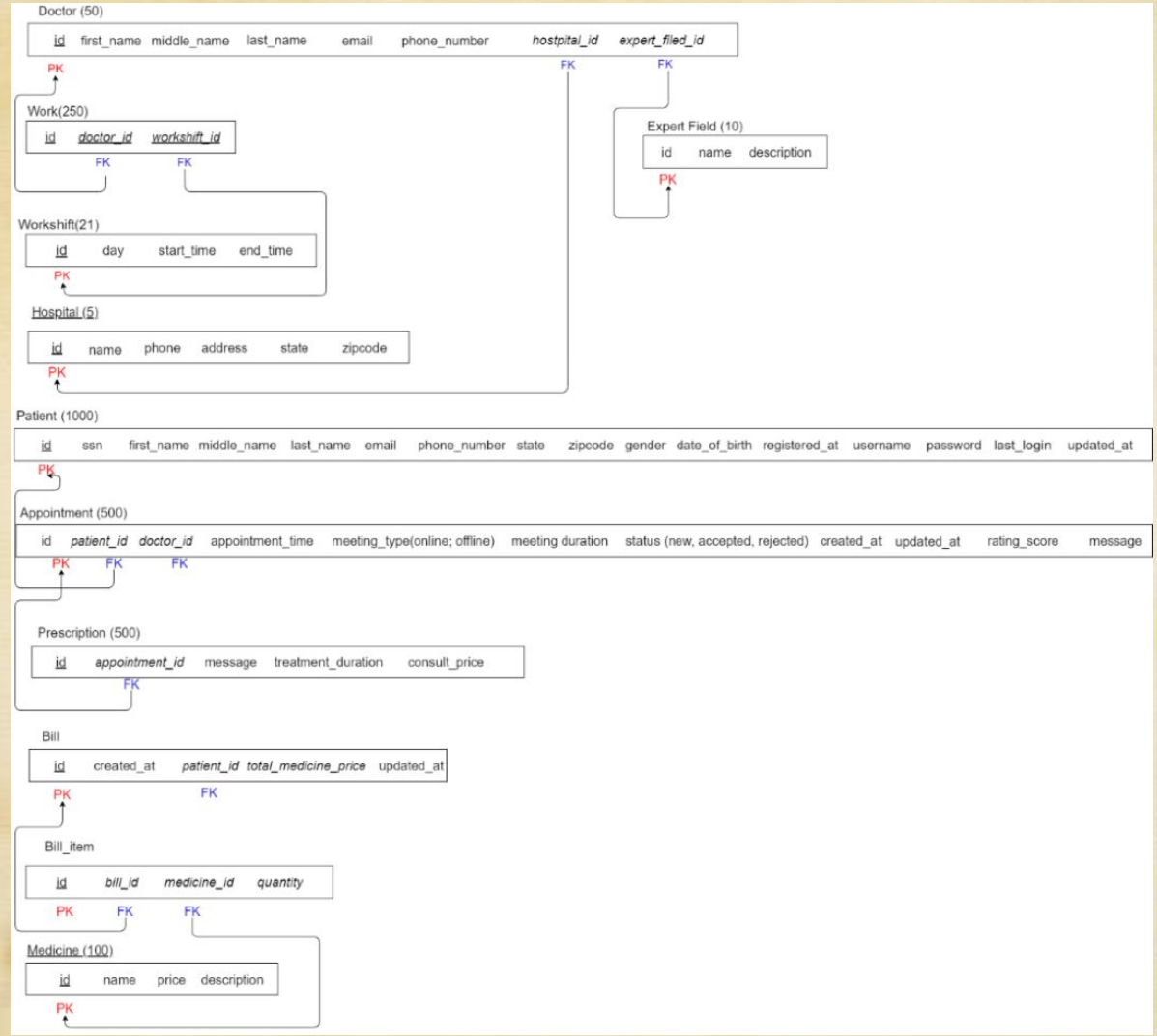
Data constraints

- One appointment requires one patient and one doctor.
- One prescription require one appointment.
- One doctor can work in one hospital.
- One doctor can be an expert in only one field.
- One doctor must work at least one workshift.
- One patient can generate many bills.
- One bill can have many bill items.
- Each bill item can have only one medicine.
- One hospital can have many doctors.

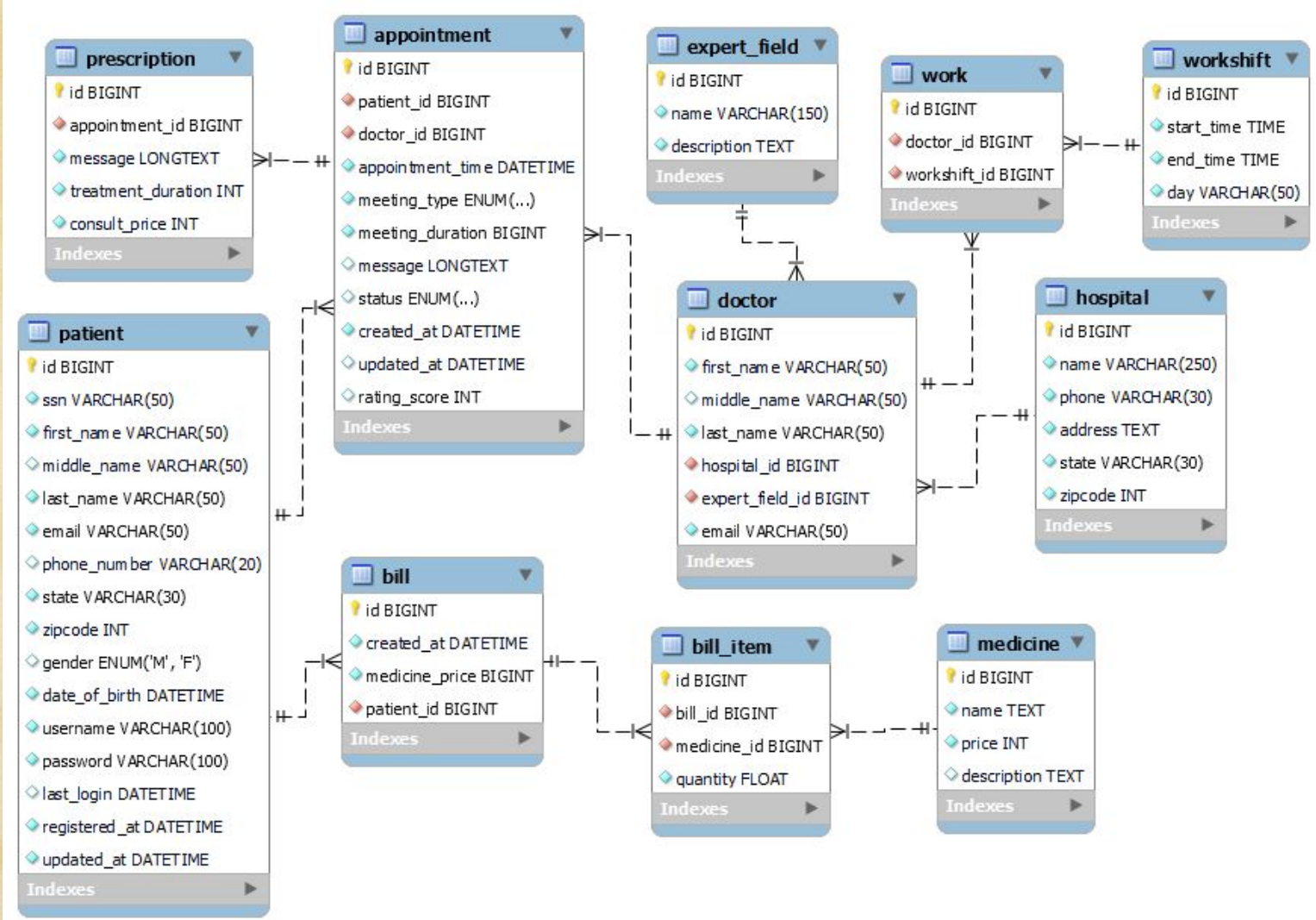
ER diagram


Work
N
Workshift
N
Expert field
1
Has
N
Doctor
N
works in
1
Hospital
N
Medicine list
N
Prescription
1
Appoint
Contain
N
N
Bill
N
can create
1
Patient
N

Logical Diagram



SCHEMA DIAGRAM crow's foot notation



Relational schema

PATIENT(*id*, *ssn*, *first_name*, *middle_name*, *last_name*, *email*, *phone_number*, *state*, *zipcode*, *gender*, *date_of_birth*, *registered_at*, *username*, *password*, *last_login*, *updated_at*)

HOSPITAL(*id*, *name*, *phone*, *address*, *state*, *zipcode*)

EXPERT_FIELD(*id*, *name*, *description*)

DOCTOR(*id*, *first_name*, *middle_name*, *last_name*, *email*, *hospital_id*, *expert_field_id*)

MEDICINE(*id*, *name*, *price*, *description*)

Relational schema

BILL(*id*, *created_at*, *medicine_price*, *patient_id*)

BILL_ITEM(*id*, *bill_id*, *medicine_id*, *quantity*)

PRESCRIPTION(*id*, *appointment_id*, *message*, *treatment_duration*, *consult_price*)

WORK(*id*, *doctor_id*, *workshift_id*)

WORKSHIFT(*id*, *day*, *start_time*, *end_time*)

APPOINTMENT(*id*, *patient_id*, *doctor_id*, *appointment_time*, *meeting_type*, *meeting_duration*, *status*, *created_at*, *updated_at*, *rating_score*, *message)*

Business Rules

- All doctor in this app must work in only one hospital.
- Patient can select only one problem for each appointment.
- Prescription can be issued only after an appointment.
- Patient must have an appointment if he wants to connect with a doctor.
- Doctors have every week the same schedule.
- Doctors can accept or reject a new appointment.
- Not all hospitals have doctor for every expert field.
- Patients can give rating to their appointments with doctors.
- The doctor must have a legal license.
- Each appointment will charge for the consultation fee.

Data dictionary

Patient

Column	Datatype	Allow Null	Key	Reference	Description
id	BIGINT	No	PK, AI		ID of patient
ssn	VARCHAR(50)	No			Identification Number of patient
first_name	VARCHAR(50)	No			First Name of patient
middle_name	VARCHAR(50)	Yes			Middle Name of patient
last_name	VARCHAR(50)	No			Last Name of patient
email	VARCHAR(50)	No			Email of patient
phone_number	VARCHAR(20)	Yes			Phone number of patient
state	VARCHAR(30)	No			State or province of patient
zipcode	INT	No			Zipcode of patient
gender	ENUM('M', 'F')	Yes			Gender ‘M’ = male ‘F’ = female
date_of_birth	DATETIME	No			Birth date of patient
username	VARCHAR(30)	No			Username for application login
password	VARCHAR(30)	No			Password for application login
last_login	DATETIME	Yes			Latest login date
registered_at	DATETIME	No			Patient created date
updated_at	DATETIME	No			Latest information update date

Hospital

Column	Data type	Allow Null	Key	Reference	Description
id	BIGINT	No	PK, AI		Hospital id
name	VARCHAR(250)	No			Hospital name
phone	VARCHAR(30)	No			Hospital phone number
address	TEXT	No			Hospital address
state	VARCHAR(30)	No			The state hospital is located at
zipcode	INT	No			The hospital zip code number

Doctor

Column	Datatype	Allow NULL	KEY	Reference	Description
id	BIGINT	NO	PK		ID of doctor
first_name	VARCHAR(50)	NO			First name of doctor
middle_name	VARCHAR(50)	YES			Middle name of doctor
last_name	VARCHAR(50)	NO			Last name of doctor
hospital_id	BIGINT	NO	FK	hospital.id	ID of hospital which doctor works in
expert_field_id	BIGINT	NO	FK	expert_field.id	ID of expert field of doctor
email	VARCHAR(50)	NO			E-mail address of doctor

Doctor

Column	Datatype	Allow NULL	KEY	Reference	Description
id	BIGINT	NO	PK		ID of doctor
first_name	VARCHAR(50)	NO			First name of doctor
middle_name	VARCHAR(50)	YES			Middle name of doctor
last_name	VARCHAR(50)	NO			Last name of doctor
hospital_id	BIGINT	NO	FK	hospital.id	ID of hospital which doctor works in
expert_field_id	BIGINT	NO	FK	expert_field.id	ID of expert field of doctor
email	VARCHAR(50)	NO			E-mail address of doctor

Medicine

Column	Datatype	Allow null	Key	Reference	Description
id	int	no	PK,AI		Id of the drug
name	varchar	no			Name of the drug
price	int	no			Price of the drug in baht
description	text	no			Composition and instructions of use of the drug.

Appointment

Column	Datatype	Allow null	Key	Reference	Description
id	int	no	primary		Id of the appointment
patient_id	int	no	foreign	patient.id	Id of the patient in the database
doctor_id	int	no	foreign	doctor.id	Id of the doctor in the database
appointment_time	datetime	no			Appointment time
meeting_type	varchar	no			Define if the appointment is online or offline
meeting_duration	int	no			Duration of the appointment in minutes
status	varchar	no			Define the appointment status between three states: new, accepted, or rejected
created_at	datetime	no			Informs about the date of creation of the appointment
updated_at	datetime	yes			Informs about the date of the last updating of the appointment
rating_score	int	yes			Rating from one to five of the appointment by the patient
message	text	yes			Space in which the patient can provide further details to the doctor

Bill

Column	Datatype	Allow Null	Key	Reference	Description
id	BIGINT	No	PK, AI		ID of medicine bill
created_at	DATETIME	Yes			Bill created date
medicine_price	BIGINT	No			Total price of medicine
patient_id	BIGINT	No	FK	patient.id	Owner of this bill

Expert field

Column	Datatype	Allow Null	Key	Reference	Description
id	BIGINT	No	PK, AI		ID of expert field categories
name	VARCHAR(150)	No			Expert field or problem categories name
description	TEXT	No			Description of that expert field

Work

Column	Datatype	Allow Null	Key	Reference	Description
id	BIGINT	No	PK, AI		ID of work
doctor_id	BIGINT	No	FK	doctor.id	ID of doctor
workshift_id	BIGINT	No	FK	workshift.id	ID of work shift.

Column	Datatype	Allow Null	Key	Reference	Description
id	BIGINT	No	PK, AI		ID of bill item
bill_id	DATETIME	No	FK	bill.id	ID of bill
medicine_id	BIGINT	No	FK	medicine.id	ID of medicine
quantity	BIGINT	No			Quantity of the medicine selected by the patient for billing.

Bill Item

Column	Data type	Allow Null	Key	Reference	Description
id	BIGINT	No	PK, AI		Workshift id
start_time	Time	No			The start time of each work shift
end_time	Time	No			The end time of each work shift
day	VARCHAR(50)	No			Day (7 days of the week)

Workshift

Column	Datatype	Allow NULL	KEY	Reference	Description
id	BIGINT	NO	PK, AI		ID of prescription
appointment_id	BIGINT	NO	FK	appointment.id	ID of appointment which the doctor create the prescription
message	LONGTEXT	NO			Message that the doctor write to the patient
treatment_duration	INT	NO			How long the petient should be treated (by medicine or any suggestion from the doctor)
consult_price	INT	NO			Doctor consultation fee

Prescription

Sample SQL data commands

- insert into patient (id, ssn, first_name, middle_name, last_name, email, phone_number, state, zipcode, gender, date_of_birth, registered_at, username, password, last_login, updated_at) values (1, '667-57-6722', 'Kellia', 'Hulda', 'Kiljan', 'hkiljan0@nps.gov', '916-644-8832', 'California', '89252', 'F', '1977-01-25 04:42:32', '2020-07-02 14:06:13', 'hkiljan0', 'cde051bdccb11b2569bee9e2bafc074a', '2021-05-25 21:30:36', '2020-10-04 18:25:30');
- insert into hospital (id, name, phone, address, state, zipcode) values (1, 'Ardea golieth', '504-938-7566', '688 3rd Way', 'Louisiana', '94481');
- insert into doctor (id, first_name, middle_name, last_name, email, hospital_id, expert_field_id) values (10, 'Kinna', null, 'Bucksey', 'kbucksey9@comcast.net', 4, 3);
- insert into appointment (id, patient_id, doctor_id, appointment_time, meeting_type, meeting_duration, status, created_at, updated_at, rating_score, message) values (5, 393, 31, '2020-09-14 21:07:39', 'online', 80, 'new', '2020-09-29 01:36:23', '2020-09-18 21:56:37', 4, 'This is message from patient to doctor');
- insert into medicine (id, name, price, description) values (56, 'quetiapine fumarate', 70, 'Nondisplaced lateral mass fracture of first cervical vertebra');
- insert into bill (id, created_at, medicine_price, patient_id) values (35, '2021-04-02 11:45:43', 384, 577);
- Update patient set middle_name=’venkata’ where id=1;
- Delete from doctor where id=1;

Important operations

- **Insert:**
  - Insert information of new patients.
  - Insert information of new appointments made by each patient.
  - Insert details of new prescriptions for each appointment.
  - Insert new workshifts of each doctor.
  - Insert new workshift details.
  - Insert information of new expert fields.
  - Insert information of new hospital.
  - Insert details of new bills.
  - Insert details of new bill items.
  - Insert information of medicine information.

Important operations

- **Update:**
  - Update appointment detail such as the message from patient.
  - Update the status of the appointment
  - Update the doctor details.
  - Update the doctor work hours.
  - Update the place of work of the doctor
  - Update the price of medicine.
- **Delete:**
  - Delete medicine
  - Delete an appointment
  - Delete doctor’s details
  - Delete patient’s information
  - Delete bill detail

Important data inquiries and reports

- **Patient**
  - Can see the appointment time.
  - Can see prescription from doctor.
  - Can see appointment and prescription history.
  - Can see bill detail.
  - List of medicine patient can order.
  - Most encountered field of problem by gender and age group.
  - Typical disease by each quarter.
  - The average duration for each medical problem.
  - Can see total money spent by each patient till date.
  - Number of hospitals which treat the medical problem near to the location of the patient.
  - Can see average duration spent by each doctor on patients.

Important data inquiries and reports

- **Patient**
  - Can see a list of doctor who is expert in the particular problem.
  - Show the max and average waiting time for appointment to be approved.
  - Can see which doctor has the highest, average review rating.
  - Can see the latest appointment given by the specific doctor.
  - Can see average days that each doctor accepts an appointment.
  - Find which hospitals have more doctors than average assume that the average doctors in a hospital might be 25 (Use having clause).
  - Show all the doctors information with number of appointments, ratings, number of patients visited a doctor.
  - Patient can choose on which day he wants to have an appointment by seeing the total number of appointments for a particular doctor on a specific day. Show the number of appointments for a doctor on a specific day and probability of patients having appointments on that specific day(patients having appointment on a particular day with a specific doctor / number of patients having appointment with that specific doctor).
  - Patients can see the maximum consultation price, minimum consultation price, average consultation price of all doctors and can choose the doctor accordingly

Important data inquiries and reports

- **Doctor**
  - Can see their work hours
  - Can see the number of patients that make appointment between particular time.
  - See if the patient is often sick.
  - Get a fast report of previous medical problems of the patient (with his previous appointment)
- **Hospital**
  - Number of patient living in each state in selected hospital
  - How many patient per month
  - Get the right distribution of doctors in each medical sector [Expert Field, Optimized number of doctor] (Know the average appointment time in this expert field, know the number of doctor)

Important data inquiries and reports

- **Hospital**
  - Get the most common disease in the state (Max number of cases in specific field in each state)
  - How many doctor for each work shift
  - See if a doctor is efficient or not check from the rating score.
  - Get average meeting time for each doctor and display medical field.
  - Check average consult price for each doctor.
  - No of cases for each doctor
  - Total cases for each expert field in specific date.(can be group with previous queries I think)
  - Show all patients who didn’t have any appointment and bills
  - Show the patient information with their current age, number of appointment that they have made, number of bill that they have created, total appointment and bill, and number of medicine that they have ordered.

Query examples

- See the patient who is often sick

```
99      -- See the patient who often sick
100 •   SELECT patient_id,count(id) as "number of appointment" FROM myfirstdb.appointment
101     group by patient_id
102     order by count(id) desc;
```

patient_id	number of appointment
136	5
353	5
52	4
329	4
342	4
3	3
5	3
13	3
35	3
39	3
41	3

Query examples

- Most needed field of expertise filter by age and gender:Query

```
(SELECT e.name FROM myfirstdb.patient p
inner join myfirstdb.appointment a on p.id = a.patient_id
inner join myfirstdb.doctor d on a.doctor_id = d.id
inner join myfirstdb.expert_field e on d.expert_field_id = e.id
where p.gender = 'F' and timestampdiff(year,p.date_of_birth,now()) > 20
and timestampdiff(year,p.date_of_birth,now()) < 40
group by e.id
order by count(e.id) desc
limit 1) as "Most needed field of expertise for Female age between 20 and 40",
```

```
(SELECT e.name FROM myfirstdb.patient p
inner join myfirstdb.appointment a on p.id = a.patient_id
inner join myfirstdb.doctor d on a.doctor_id = d.id
inner join myfirstdb.expert_field e on d.expert_field_id = e.id
where p.gender = 'F' and timestampdiff(year,p.date_of_birth,now()) > 40
group by e.id
order by count(e.id) desc
limit 1) as "Most needed field of expertise for Female age above 40";
```

Query examples

- Most needed field of expertise filter by age and gender: Result

	Most needed field of expertise for Male age below 20	Most needed field of expertise for Male age between 20 and 40	Most needed field of expertise for Male age above 40
▶	Psychiatry	Dermatology	Psychiatry

Most needed field of expertise for Female age below 20	Most needed field of expertise for Female age between 20 and 40	Most needed field of expertise for Female age above 40
Psychiatry	Psychiatry	Psychiatry

Query examples

- How many patients per month :query

```
71  (SELECT count(a.patient_id) FROM myfirstdb.appointment a inner join myfirstdb.patient p
72  on a.patient_id = p.Id
73  where a.appointment_time between '2021-03-01 00:00:00' and '2021-04-01 00:00:00') as "Number of patients in Mar 2021",
74  (SELECT count(a.patient_id) FROM myfirstdb.appointment a inner join myfirstdb.patient p
75  on a.patient_id = p.Id
76  where a.appointment_time between '2021-04-01 00:00:00' and '2021-05-01 00:00:00') as "Number of patients in Apr 2021",
77  (SELECT count(a.patient_id) FROM myfirstdb.appointment a inner join myfirstdb.patient p
78  on a.patient_id = p.Id
79  where a.appointment_time between '2021-05-01 00:00:00' and '2021-06-01 00:00:00') as "Number of patients in May 2021",
```

```
80  (SELECT count(a.patient_id) FROM myfirstdb.appointment a inner join myfirstdb.patient p
81  on a.patient_id = p.Id
82  where a.appointment_time between '2021-06-01 00:00:00' and '2021-07-01 00:00:00') as "Number of patients in Jun 2021",
83  (SELECT count(a.patient_id) FROM myfirstdb.appointment a inner join myfirstdb.patient p
84  on a.patient_id = p.Id
85  where a.appointment_time between '2021-07-01 00:00:00' and '2021-08-01 00:00:00') as "Number of patients in Jul 2021",
86  (SELECT count(a.patient_id) FROM myfirstdb.appointment a inner join myfirstdb.patient p
87  on a.patient_id = p.Id
88  where a.appointment_time between '2021-08-01 00:00:00' and '2021-09-01 00:00:00') as "Number of patients in Aug 2021";
```

Query examples

- How many patients per month : results

Number of patients in Sep 2020	Number of patients in Oct 2020	Number of patients in Nov 2020	Number of patients in Dec 2020
20	33	50	54

Number of patients in Jan 2021	Number of patients in Feb 2021	Number of patients in Mar 2021	Number of patients in Apr 2021
45	43	33	38

Number of patients in May 2021	Number of patients in Jun 2021	Number of patients in Jul 2021	Number of patients in Aug 2021
35	44	49	36

Query examples

- Number of cases for each doctor

```
104     -- No of case for each doctor
105 •   SELECT d.id,d.first_name,d.middle_name,d.last_name,count(a.id) as "number of cases" FROM myfirstdb.doctor d
106     inner join myfirstdb.appointment a on d.id = a.doctor_id
107     group by d.id;
```

	id	first_name	middle_name	last_name	number of cases
▶	1	Lucinda	Osanne	Eykel	6
	2	Edsel	MacFadzan	McLugish	11
	3	Donnie	Gooden	Guly	6
	4	Kaia	Quidenham	Lube	12
	5	Julienne	NULL	Cosins	7
	6	Alfi	Marnane	Semeradova	6
	7	Tamra	NULL	Scholar	8
	8	Lars	NULL	Latta	10
	9	Barry	Lower	St. Ledger	13
	10	Kinna	NULL	Bucksey	13
	11	Brandais	NULL	Battelle	7

Query examples

- See bill details

```
96      -- See bill details
97 •    SELECT * FROM myfirstdb.bill;
98
```

Result Grid | Filter Rows: | Edit:

id	created_at	medicine_price	patient_id
1	2021-04-05 23:20:37	641	848
2	2021-02-14 14:29:58	545	725
3	2021-04-25 11:43:32	550	116
4	2021-07-02 05:20:26	395	34
5	2021-06-09 17:51:19	541	192
6	2021-06-08 15:10:24	443	180
7	2021-02-27 06:00:15	727	518
8	2021-06-19 03:38:32	582	585

Query examples

- Check the rating for doctors

```
108    -- check the rating for doctor
109 •  SELECT d.id,d.first_name,d.middle_name,d.last_name,avg(a.rating_score) as "Rating for doctor"
110    FROM myfirstdb.appointment a inner join myfirstdb.doctor d on a.doctor_id = d.id
111    group by d.id;
```

id	first_name	middle_name	last_name	Rating for doctor
1	Lucinda	Osanne	Eykel	2.3333
2	Edsel	MacFadzan	McLugish	2.9091
3	Donnie	Gooden	Guly	3.0000
4	Kaia	Quidenham	Lube	2.7500
5	Julienne	NULL	Cosins	3.8571
6	Alfi	Marnane	Semeradova	2.3333
7	Tamra	NULL	Scholar	2.3750
8	Lars	NULL	Latta	3.4000
9	Barry	Lower	St. Ledger	2.3846
10	Kinna	NULL	Bucksey	3.2308
11	Brandais	NULL	Battelle	4.1429

Query examples

- Patients can see average days that each doctor accepts an appointment.

```
SELECT CONCAT('Dr.', dr.first_name, ' ', dr.last_name) AS 'Doctor name',
AVG((DATEDIFF(ap.created_at, ap.updated_at))) AS Average_day
FROM appointment ap LEFT JOIN doctor dr ON ap.doctor_id = dr.id
WHERE ap.status = 'accepted'
GROUP BY dr.id
ORDER BY Average_day DESC;
```

Doctor name	Average_day
Dr.Eba Pickring	95.6000
Dr.Joshua Somerfield	93.5000
Dr.Gerta Dillintone	91.6667
Dr.Alida Simeon	90.6667
Dr.Edsel McLugish	82.5000
Dr.Randy Currey	79.4000
Dr.Dore MacCombe	72.5000
Dr.Kathy Totterdell	65.0000
Dr.Nevins Wilcot	60.6667
Dr.Halimeda O'Brollachain	55.3333
Dr.Petronilla Athey	49.0000
Dr.Kellen Duval	42.5000
Dr.Tamra Scholar	40.5000
Dr.Donnie Guly	28.0000

Query examples

- Hospital can get the most common disease in each state of patients (Max number of cases in specific field in each state)

```
SELECT State, Disease, MAX(CT) AS 'Number of case'
FROM (SELECT pt.state AS State, ep.name AS Disease, COUNT(*) AS CT
    FROM (patient pt RIGHT JOIN appointment ap ON pt.id = ap.patient_id
        LEFT JOIN doctor dr ON ap.doctor_id = dr.id
        LEFT JOIN expert_field ep ON dr.expert_field_id = ep.id)
    GROUP BY ep.name, pt.state) T
GROUP BY State
ORDER BY State;
```

State	Disease	Number of case
Alabama	Cardiology	5
Alaska	Respiratory	2
Arizona	Psychiatry	4
California	Infectious disease	10
Colorado	Infectious disease	2
Connecticut	Psychiatry	1
District of Columbia	Urology	3
Florida	Urology	8
Georgia	Internal medicine	6
Hawaii	Dermatology	1
Idaho	Dermatology	2
Illinois	Dermatology	4
Indiana	Infectious disease	4
Iowa	Psychiatry	2
Kansas	Cardiology	2

Query examples

- Hospital can see the patient information with their current age, number of appointment that they have made, number of bill that they have created, total appointment and bill, and number of medicine that they have ordered.

```
SELECT CONCAT(pt.first_name, ' ', pt.last_name) AS 'Patient name',
    pt.phone_number, pt.email, pt.state,
    TIMESTAMPDIFF(YEAR, pt.date_of_birth, now()) AS 'age',
    A.Ap_no AS Appointment_no,
    B.Bill_no AS Bill_no,
    (SELECT Appointment_no + Bill_no) AS Total_Appointment_Bill,
    BT.Med_no AS Medicine_no
FROM patient pt
    LEFT JOIN (SELECT pt.id AS id, COUNT(ap.id) AS Ap_no FROM patient pt LEFT JOIN appointment ap ON pt.id = ap.patient_id GROUP BY pt.id) A ON A.id = pt.id
    LEFT JOIN (SELECT pt.id AS id, COUNT(b.id) AS Bill_no FROM patient pt LEFT JOIN bill b ON pt.id = b.patient_id GROUP BY pt.id) B ON B.id = pt.id
    LEFT JOIN (SELECT pt.id AS id, COUNT(bt.id) AS Med_no FROM patient pt
        LEFT JOIN bill b ON pt.id = b.patient_id
        LEFT JOIN bill_item bt ON b.id = bt.bill_id GROUP BY pt.id) BT ON BT.id = pt.id
ORDER BY Total_Appointment_Bill DESC;
```

Patient name	phone_number	email	state	age	Appointment_no	Bill_no	Total_Appointment_Bill	Medicine_no
Fidelity Heys	217-624-0363	hheys3r@biglobe.ne.jp	Illinois	17	5	2	7	4
Inglis Scherer	269-404-3883	bscherer1f@cmu.edu	Michigan	17	4	2	6	4
Cole Connew	678-723-6386	mconnewj@netscape.com	Georgia	36	2	3	5	1
Issiah Keir	412-438-8151	rkeir18@cpanel.net	Pennsylvania	36	2	3	5	8
Betteanne MacGuiness	325-467-0631	lmacguiness3a@taobao.com	Texas	37	3	2	5	1
Farlie Hannent	573-628-4689	bhannent6d@squidoo.com	Missouri	49	2	3	5	9

Query examples

- Prescription from each doctor

```
-- Prescription from each doctor
select d.id as doctor_id, d.first_name, d.last_name, p.id as precription_id,  p.message, p.treatment_duration,
p.consult_price
from prescription p inner join appointment a on p.appointment_id = a.id
    inner join doctor d on a.doctor_id = d.id order by d.id;
```

doctor_id	first_name	last_name	precription_id	message	treatment_duration	consult_price
1	Lucinda	Eykel	163	Duis ac nibh.	3	400
1	Lucinda	Eykel	316	Nullam molestie nibh in lectus.	8	500
1	Lucinda	Eykel	327	Vivamus metus arcu, adipiscing molestie, hendr...	32	450
1	Lucinda	Eykel	431	Etiam faucibus cursus urna.	26	400
1	Lucinda	Eykel	443	In congue.	25	450
1	Lucinda	Eykel	476	Donec posuere metus vitae ipsum.	56	200
2	Edsel	McLugish	3	Vestibulum sed magna at nunc commodo placerat.	46	50
2	Edsel	McLugish	40	Integer aliquet, massa id lobortis convallis, tort...	60	50
2	Edsel	McLugish	45	Nulla mollis molestie lorem.	29	450

Query examples

- Prescription from each doctor

```
-- Prescription from each doctor
select d.id as doctor_id, d.first_name, d.last_name, p.id as precription_id,  p.message, p.treatment_duration,
p.consult_price
from prescription p inner join appointment a on p.appointment_id = a.id
    inner join doctor d on a.doctor_id = d.id order by d.id;
```

doctor_id	first_name	last_name	precription_id	message	treatment_duration	consult_price
1	Lucinda	Eykel	163	Duis ac nibh.	3	400
1	Lucinda	Eykel	316	Nullam molestie nibh in lectus.	8	500
1	Lucinda	Eykel	327	Vivamus metus arcu, adipiscing molestie, hendr...	32	450
1	Lucinda	Eykel	431	Etiam faucibus cursus urna.	26	400
1	Lucinda	Eykel	443	In congue.	25	450
1	Lucinda	Eykel	476	Donec posuere metus vitae ipsum.	56	200
2	Edsel	McLugish	3	Vestibulum sed magna at nunc commodo placerat.	46	50
2	Edsel	McLugish	40	Integer aliquet, massa id lobortis convallis, tort...	60	50
2	Edsel	McLugish	45	Nulla mollis molestie lorem.	29	450

Query examples

List of drugs the patient can order

```
-- List of medecine the patient can order
select p.id as patient_id, p.first_name, p.last_name, m.id as medicine_id, m.name
from patient p inner join bill b on p.id = b.patient_id
inner join bill_item b_i on b_i.bill_id = b.id
inner join medicine m on m.id = b_i.medicine_id
order by p.id asc;
```

patient_id	first_name	last_name	medicine_id	name
5	Ganny	Longfield	69	JIANZE SURGICAL SCRUB BRUSH NAIL CLEANER
5	Ganny	Longfield	23	TEMOVATE
6	Blinni	Dmitrievski	74	Amlodipine Besylate and Benazepril Hydrochloride
6	Blinni	Dmitrievski	86	Pentoxifylline
8	Osbourne	Duckett	12	Trazodone Hydrochloride
8	Osbourne	Duckett	40	Nicardipine Hydrochloride
8	Osbourne	Duckett	56	quetiapine fumarate
9	Tanitansy	Giovanetti	25	Amitriptyline Hydrochloride
9	Tanitansy	Giovanetti	55	rizatriptan benzoate

Query examples

- Typical disease per quarter

id	name	Prob_first_quarter	Prob_second_quarter	Prob_third_quarter	Prob_fourth_quarter
4	Psychiatry	0.2066	0.2308	0.2000	0.1679
2	Dermatology	0.1901	0.1197	0.1760	0.1022
5	Infectious disease	0.1322	0.1624	0.1360	0.1387
1	Cardiology	0.1157	0.1624	0.1040	0.1679
3	Respiratory	0.0909	0.0684	0.1520	0.1314
10	Urology	0.0909	0.0940	0.0800	0.0949
7	Nephrology	0.0579	0.0256	0.0240	0.0219
8	Neurology	0.0496	0.0769	0.0640	0.0511
6	Internal medicine	0.0413	0.0342	0.0560	0.0803
9	Orthopedic	0.0248	0.0256	0.0080	0.0438

Query examples

- Previous medical problem of a patient

```
select p.id, p.first_Name, p.last_Name, ef.name, presc.treatment_duration, presc.message
from expert_field ef inner join doctor d on d.expert_field_id = ef.id
inner join appointment a on a.doctor_id = d.id
inner join patient p on a.patient_id = p.id
inner join prescription presc on a.id = presc.appointment_id
where p.id = 1 -- id of the patient the doctor wants to check
order by ef.id;
```

id	first_Name	last_Name	name	treatment_duration	message
1	Kellia	Kiljan	Psychiatry	59	Duis at velit eu est congue elementum.
1	Kellia	Kiljan	Infectious disease	48	Vestibulum quam sapien, varius ut, blandit non, ...

Query examples

- Average treatment duration per medical sector

id	name	average_treatment_duration
1	Cardiology	29.1884
2	Dermatology	29.3014
3	Respiratory	29.2857
4	Psychiatry	30.7200
5	Infectious disease	28.8592
6	Internal medicine	26.1852
7	Nephrology	29.3125
8	Neurology	24.1000
9	Orthopedic	37.7692
10	Urology	31.9111

Query examples

- Number of appointment hours over number of working hours by hospital and expert_field

h_id	ef_id	ef_name	average_of_appointment_hours_p	number_of_doctor	number_of_working_hours_in_the_sec	average_number_of_working_hours_per
1	1	Cardiology	0.80833333	2	112	56.0000
1	3	Respiratory	0.87037037	1	96	96.0000
1	4	Psychiatry	0.80952381	1	40	40.0000
2	1	Cardiology	1.30158730	2	120	60.0000
2	2	Dermatology	1.28240741	3	144	48.0000
2	3	Respiratory	0.73055556	1	56	56.0000
2	4	Psychiatry	0.66805556	1	56	56.0000
2	5	Infectious disease	1.00277778	2	120	60.0000
2	10	Urology	0.88235294	2	64	32.0000
3	1	Cardiology	0.86041667	1	16	16.0000
3	3	Respiratory	1.18253968	3	144	48.0000
3	4	Psychiatry	0.91388889	2	88	44.0000
3	5	Infectious disease	1.01458333	2	128	64.0000
3	6	Internal medicine	0.76875000	1	56	56.0000

Query examples

- Show the total revenue generated in a particular year.

```
SELECT Year(b.created_at)                  AS year,
       Sum(Ifnull(b.medicine_price, 0)) AS 'total revenue (USD)'
FROM   myfirstdb.bill b
GROUP  BY Year(b.created_at);
```

year	total revenue (USD)
2020	104864
2021	228332

Query examples

- Total bill of patient till date

id	patient_name	medicine price	consultation price	total bill made by the patient till date	average amount purchased by the patient till date
1	Eulalie Junina Loffhead	0	0	0	0.0000
2	Kaiser Nikita Oliff	526	100	626	621.6667
3	Reinwald Ferdinand Rowlatt	364	0	364	397.5000
4	Robbyn Claire Wankel	0	350	350	310.0000

- Show the most frequent appointment type(online or offline) with respect to age and gender.

most_frequent_type_for_male_under 20 years	most_frequent_type_for_female_under 20 years	most_frequent_type_for_male_between 20 - 40 years
online - no of patients 48	online - no of patients 40	online - no of patients 49

most_frequent_type_for_female_between 20 - 40 years	most_frequent_type_for_male_above 40 years	most_frequent_type_for_female_above 40 years
online - no of patients 56	online - no of patients 28	offline - no of patients 30

Query examples

- Patients can choose on which day he wants to have an appointment by seeing the total number of appointments for a particular doctor on a specific day. Show the number of appointments for a doctor on a specific day and probability of patients having appointments on that specific day(patients having appointment on a particular day with a specific doctor / number of patients having appointment with that specific doctor). **

doctor id	doctor name	no of appointments for a doctor on that particular day	day	probably of patients having appointment on that particular day
1	Mable Wink	1	Monday	0.2000
1	Mable Wink	1	Saturday	0.2000
1	Mable Wink	1	Sunday	0.2000
1	Mable Wink	1	Tuesday	0.2000

Query examples

- Patients can see the maximum consultation price, minimum consultation price, average consultation price of all doctors and can choose the doctor accordingly.

id	doctor_name	expert_field_name	average_consult_price	min_consult_price	max_consult_price
1	Mable La Batie Wink	Respiratory	210.0000	50	500
2	Thorny Dalgarnowch Tregien	Urology	288.8889	50	450
3	Adolpho Astill Lamble	Infectious disease	250.0000	100	500
4	Karla Wareham Horbath	Infectious disease	290.0000	100	500
5	Alisander Irnis Tench	Urology	233.3333	50	500

- Get the average duration spent by each doctor on patients and display his details along with the expert field.

id	doctor_name	email	expert_field_id	NAME	averageDuration spent on each meeting (min)
1	Mable La Batie Wink	mwink0@t-online.de	3	Respiratory	50.0000
2	Thorny Dalgarnowch Tregien	ttregien1@1688.com	10	Urology	46.7778
3	Adolpho Astill Lamble	alamble2@cyberchimps.com	5	Infectious disease	55.6154

Query examples

- Patients can see the latest appointment given by a particular doctor.

```
SELECT dr.id,
       Concat(dr.first_name, ' ', dr.middle_name, ' ', dr.last_name) AS
       'doctor name',
       ap.patient_id,
       Concat(p.first_name, ' ', p.middle_name, ' ', p.last_name)    AS
       'patient name',
       ap.appointment_time
FROM   myfirstdb.doctor dr
       INNER JOIN myfirstdb.appointment ap
               ON ap.doctor_id = dr.id
       INNER JOIN myfirstdb.patient p
               ON ap.patient_id = p.id
WHERE  dr.id = 1
       AND status = 'accepted'
ORDER  BY ap.appointment_time DESC
LIMIT  1;
```

id	doctor name	patient_id	patient name	appointment_time
1	Mable La Batie Wink	138	Grover Guillermo Ferney	2021-09-06 11:38:13

Query examples

Find which hospitals have more doctors than average

```
SELECT hospital.id       AS hospital_id,
       hospital.NAME     AS hospital_name,
       hospital.state,
       Count(doctor.id) AS doctor_count
FROM   doctor
       INNER JOIN hospital
               ON doctor.hospital_id = hospital.id
GROUP  BY hospital.id
HAVING doctor_count > (SELECT Avg(doctor_count)
                       FROM   (SELECT Count(doctor.id) AS doctor_count
                               FROM   doctor
                                      INNER JOIN hospital
                                      ON doctor.hospital_id = hospital.id
                               GROUP  BY hospital.id) count);
```

hospital_id	hospital_name	state	doctor_count
2	Dasypus novemcinctus	Georgia	11
3	Stercorarius longicausus	New Mexico	11
4	Trichosurus vulpecula	Ohio	16

Query examples

Doctors can see the number of patients that make an appointment between particular times.

```
SELECT appointment.id,
       Concat(patient.first_name, ' ', patient.last_name) AS patient_name,
       appointment_time,
       meeting_type,
       message,
       status,
       appointment.created_at,
       appointment.updated_at
FROM   appointment
       INNER JOIN patient
               ON appointment.patient_id = patient.id
WHERE  doctor_id = 1
       AND appointment_time BETWEEN '2021-08-01 23:59:59' AND
                                    '2021-09-09 23:59:59'
       AND status = 'accepted';
```

id	patient_name	appointment_time	meeting_type	message	status	created_at	updated_at
163	Dorian Dugan	2021-08-20 12:30:50	offline	This is message from patient to doctor	accepted	2021-01-11 19:52:59	2021-03-10 19:09:52
316	Conny Possek	2021-09-04 11:51:33	offline	This is message from patient to doctor	accepted	2021-02-23 19:43:10	2020-11-22 09:26:41

Query examples

Total cases for each expert field on the specific date

```
SELECT expert_field_id,
       expert_field.NAME,
       expert_field.description,
       Count(appointment.id)
FROM   appointment
       INNER JOIN doctor
               ON appointment.doctor_id = doctor.id
       INNER JOIN expert_field
               ON expert_field.id = doctor.expert_field_id
WHERE  appointment_time BETWEEN '2020-01-01 23:59:59' AND '2021-09-09 23:59:59'
GROUP  BY expert_field_id
ORDER  BY expert_field_id;
```

expert_field_id	name	description	COUNT(appointment.id)
1	Cardiology	a branch of medicine that deals with the disorde...	69
2	Dermatology	the branch of medicine dealing with the skin. It i...	73
3	Respiratory	a medical specialty that deals with diseases invo...	56
4	Psychiatry	the medical specialty devoted to the diagnosis p...	100
5	Infectious disease	a medical specialty dealing with the diagnosis an...	71
6	Internal medicine	the medical specialty dealing with the preventio...	27
7	Nephrology	a specialty of adult internal medicine and pediat...	16
8	Neurology	a branch of medicine dealing with disorders of t...	30
9	Orthopedic	the branch of surgery concerned with condition...	13
10	Urology	the branch of medicine that focuses on surgical ...	45

Conclusion

- Designed an appointment management application
- Learn generation of data using mockaroo.
- Can advice users of the application to make some choices
- We can now make interesting reports about hospital situation and even advice them in management.

Future work

- Add more functions such as appointment of **lab tests**, a **post and message system** to make it more collaborative, and the **remainders** to make patient get the medicine regularly.
- The scope can be increased to the inclusion of lab tests and health checkups.
- Offers and packages for the health checkups and lab tests can also be included.
- Add more detail to the table structure
- Implement and develop the application
- We also include the online recorded videos for common problems faced by the citizens during this covid time.

Q&A