DOCPLUS

(A report on table structure of this website)

Mohd. Suhail

Table of Content

Abstract:

Consumers want to do things quickly and get the services (shopping, ticketing, etc.) done as soon as possible in today's digital age, which is why immediacy is becoming the norm. The internet has recently emerged as another way of making appointment for various purpose and it has reached medical industry too for making doctor appointment online and consultation.

INTRODUCTION :

In today's digital age, customers want everything to be done quickly through online like online shopping, online ticketing, and making an online appointment soon. The growth of the internet has recently made an online appointment easier for various purposes and to be specific online appointment for medical industry is making the rounds now.

Most of the people expect the medical services to be as quick as possible without much delay. In today's fast paced work schedule most of the people are probably juggling with multiple schedules from home to work and finding very difficult to make an appointment with a doctor by going to the clinic/hospital or making a phone call to get much needed medical services for them.

This traditional way of booking appointment with a doctor is posing an inconvenience for people who needs a free medical service as quick as possible. One of the best ways to eliminate traditional way of booking an appointment is by booking an appointment online in just a few easy steps.

Benefits:

- User can search for doctor's help at any point of time.
- User can talk by **TELE-CONSULT** about their illness and get the required medicine's prescription.
- Doctors can handle emergency situation by providing primary help, till the patient can be taken to the hospital.

Identification:

- Doctor
- User (Patient)

Managing Doctors' Data:

Doctors can register with the app by filling in certain mandatory details, but the appointment booking feature is enabled only after they complete their full profile. This includes their qualifications (professional degrees, certifications/specializations, and internships), and their past and current affiliations with hospitals and healthcare service providers.

The tables shown below manage this information.

doctor		
ATTRIBUTES	**DATA TYPE**	**KEYS & EXTRA**
id	Int(6)	PK
first_name	Varchar(50)	
last_name	Varchar(50)	
Professional_statement	Varchar(4000)	
practicing_from	Date	

The **doctor** table stores elementary details about doctors, which they enter during registration. The columns in this table are:

id – A unique number that the app assigns to doctors during registration.

first_name – Doctor's first name.

last_name – Doctor's last name.

professional_statement – A detailed overview of the doctor's qualifications, experience, their professional motto, etc. This information is entered by the doctor and is displayed on each doctor's profile page.

practicing_from – The date the doctor started practicing medicine. This has deep significance, as the app will derive its experience rating for each doctor based on the information in this column.

Specialization		
ATTRIBUTES	**DATA TYPE**	**KEYS & EXTRA**
id	int(6)	PK
specialization_name	Varchar(100)	

doctor_specialization		
ATTRIBUTES	**DATA TYPE**	**KEYS & EXTRA**
Id	Int(6)	PK
doctor_id	Int(6)	FK
specialization_id	Int(6)	FK

The **specialization** table holds all existing medical specializations like orthopedic, neurologist, dentist, etc. A doctor can have more than one specialization; in fact, it's pretty common for a doctor to specialize in related fields. For example, a neurologist can also be a psychiatrist; a gynecologist can be an endocrinologist, and so on.

Therefore, the **doctor_specialization** table allows a many-to-many relationship between
the doctor and specialization tables. The attributes on these two tables are self-explanatory.

Qualification		
ATTRIBUTES	**DATA TYPE**	**KEYS & EXTRA**
id	int(6)	PK
doctor_id	int(6)	FK
qualification_name	varchar(100)	
institute_name	varchar(100)	N

procurement_year	Date	

The **qualification** table stores details about doctors' education and professional qualifications, including degrees, certifications, research papers, seminars, ongoing training, etc. To facilitate the various types of qualification details, I have given these fields quite generic names:

id – The primary key of the table.

doctor_id – References the doctor table and relates the doctor with the qualification.

qualification_name – Signifies the name of the degree, certification, research paper, etc.

institute_name – The institution that issued the qualification to the doctor. This can be a university, a medical institution, an international association of medical practitioners, etc.

procurement_year – The date when the qualification was obtained or awarded.

hospital_affiliation		
Attributes	**Data_type**	**Keys & extra**
id	int(6)	PK
hospital_id	int(6)	FK
hospital_name	varchar(100)	
city	varchar(50)	
country	varchar(50)	
start_date	date	
end_date	date	N

The **hospital_affiliation** table keeps information about doctors' affiliations with hospitals and healthcare service providers. This data is only for display on a doctor's profile and has no significance in the appointment booking feature. OPD (Outpatient Department) details are entered separately and will be handled later in this article.

This table's columns/attr. are:

id – The primary key of the table.

doctor_id – References the doctor table and links the doctor to the affiliated hospital.

hospital_name – The affiliated hospital's name.

city and country – The city and country where the hospital is located. These address columns do not

play any role in the website search function; they are only for display on the doctor's profile.

start_date – When the doctor's affiliation with the hospital commenced.

end_date – When the affiliation ended. It is nullable because current affiliations will not have an end date.

Managing Doctors' Clinic Details (IN PERSON) :

The information in this section is entered by doctors (or their staff) and plays a significant role in the app's search and booking functionalities.

In_person_doctor_availability		
ATTRIBUTES	**DATA TYPE**	**KEYS & EXTRA**
id	int(6)	PK
office _id	int(6)	FK
day_of_week	varchar(10)	
start_time	Timestamp	
end_time	Timestamp	
is_available	varchar(1)	
reason_of_unavaibility	varchar(500)	N

The **IN_PERSON_doctor_availability** table stores doctors' OPD/ clinic availability in terms of time slots (say 2 hours in the morning and 4 hours in the evening,for example). Splitting up the day this way is pretty common, so having an additional table to store availability slots makes sense. Plus, doctors can work more than one OPD shift. The columns for this table are:

id – The primary key of the table.

office_id – References the “office” table.

day_of_week – The day of the week, i.e. Monday, Tuesday, etc. This allows doctors to have different availabilities for different days (weekends vs. weekdays, for example).

start_time – When the doctor is ready for the first patient.

end_time – When the final appointment or shift is scheduled to end.

is_available – Allows doctors to mark their availability for particular days or time slots. This column is initialized with a ‘Y’ as default and is updated to an ‘N’ when doctors mark their unavailability.

reason_of_unavailablity – Many doctors prefer to disclose why they are unavailable or must cancel an appointment. This helps to build a transparent relationship between doctors and their patients. Since it is optional, I have kept this as nullable column.

Office		
Attributes	**Data_type**	**Keys & extra**
id	int(6)	PK
doctor_id	int(6)	FK
hospital_affiliation_id	int(6)	FK N
time_slot_per_client_in_min	int(2)	
Address	varchar(200)	

The office table holds information about the Outpatient Department of the hospitals doctors are affiliated with as well as their own clinics. The columns in this table are:

id – The primary key of this table.

doctor_id – References the doctor table and indicates the relevant doctor.

hospital_affiliation_id –Signifies the hospital where the doctor is available for OPD. Since the column is applicable to OPDs but not clinics, it is nullable.

time_slot_per_client_in_min – Stores an amount of time (in minutes) allotted for consultations. The number of minutes is entered by doctors based on their experience. This column helps the app determine the next available slot. Note that this number is not a

guarantee of appointment length, but it helps to minimize patient wait times if they use the app to book an appointment.

address – The address of the hospital OPD or clinic.

Client_account		
Attributes	**DATA_TYPE**	**KEYS & EXTRA**
id	int(6)	PK
first_name	varchar(50)	
last_name	varchar(50)	
mobile	int(10)	
email	varchar(50)	

The client_account table stores basic details for clients. These details are captured at the time of registration. The columns in this table are:

id – A unique number assigned to each client.

first_name – The client's first name.

last_name – The client's last name.

contact_number – The client's phone number, preferably a mobile number, to which appointment information can be sent. This is also the number where the client can be contacted by the doctor's office staff.

email – The client's email address. The website may send appointment reminders to clients.

Client_review		
Attributes	**Data_type**	**Keys & extra**
id	int(6)	PK
user_account_id	int(6)	FK
doctor_id	int(6)	FK
is_review_anonymous	varchar(1)	
rating	Float(1,1)	
review	varchar(5000)	N
review_date	Date	

The **client_review** table not only offers feedback (i.e. client reviews) for doctors, but it also helps potential clients to choose doctors. It is an integral component of this website. Columns for this table are:

id – The primary key of this table.

user_account_id – Signifies which user is submitting the review.

doctor_id – The doctor being reviewed.

is_review_anonymous – If the client's name will be published with the review or not. This is a security feature for clients.

rating – Client's rating of their general experience with the doctor.

review – Clients can give their detailed feedback here.

review_date – When the review was submitted.

Managing Appointments:

This website allows clients to check the availability of various doctors and book an appointment.

appointment		
atrributes	**Data_type**	**Keys & extra**
id	int(6)	PK
user_account_id	int(6)	FK
office_id	int(6)	FK N
probable_start_time	timestamp	
actual_end_time	timestamp	N
appointment_status_id	int(6)	FK
appointment_status_date	date	

The **appointment** table holds appointment details for clients. Its columns include:

id – A unique number is assigned to each appointment. This number is referenced elsewhere.

user_account_id – Which client is booking the appointment.

office_id – Signifies which doctor and which hospital OPD or clinic is involved in the appointment. (for in-person appointment). May be null.

probable_start_time – This is a timestamp column that holds the probable start time of the appointment. Medical appointments' start times are usually probable rather than absolute.

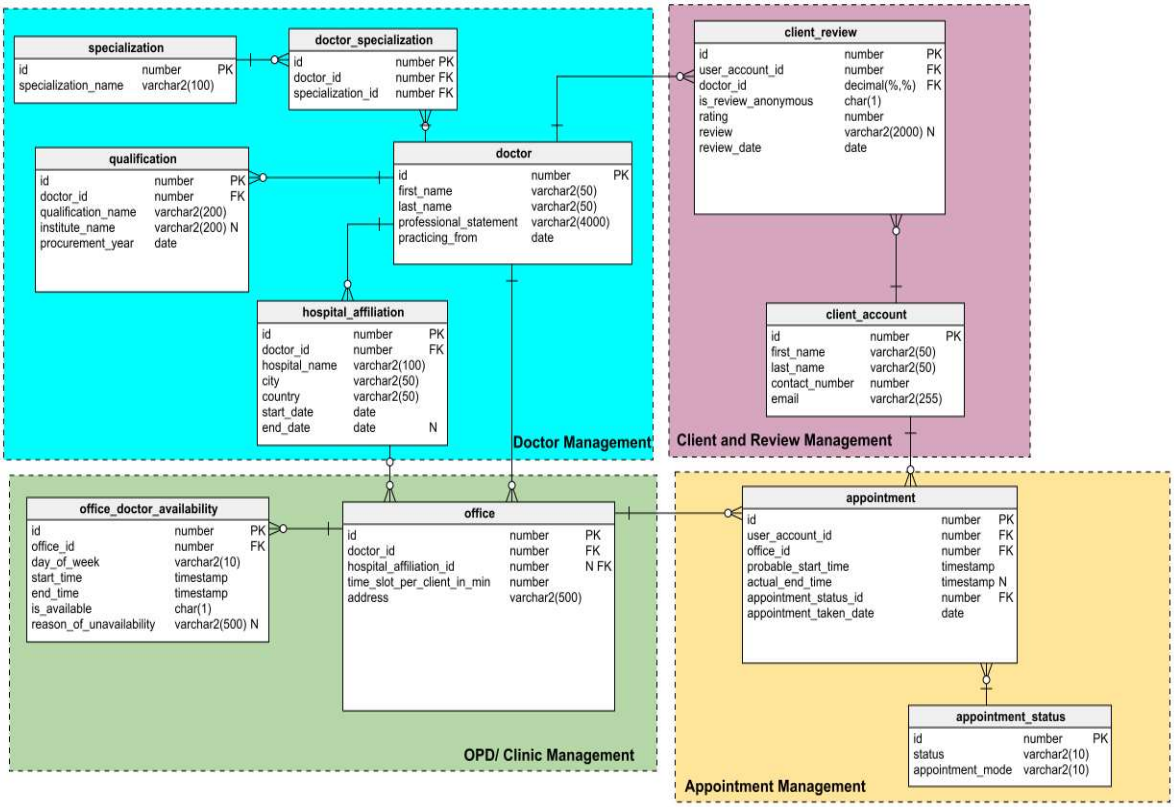
actual_end_time – The actual end time of the consultation. Initially this column is blank, as many factors can influence when an appointments ends. Therefore, this is a nullable column.

appointment_status_id – This is referenced from the **appointment_status** table, and it signifies the current status of the appointment. Possible values for status are "active", "canceled", and "complete". Initially the status would be "active". It would become "complete" once the appointment is done. It will become "canceled" if the client cancels their appointment.

appointment_taken_date – The date when appointment was made.

Appointment_status		
attributes	**Data type**	**Keys & extra**
id	int(6)	PK
status	varchar(10)	

appointment_mode	varchar(10)	



ENTITY RELATIONSHIP DIAGRAM

CONCLUSION:

Booking doctor consultation can be a troublesome issue for most of the people. It is very inconvenient when people need to physically be there just to book a consultation and waiting for their turn. Therefore, the online booking consultation is needed to solve this issue.

REFRENCES:

http://webpage.pace.edu/pd89983n/portfolio/Team4%20-%20EasyMed%20Report.pdf

https://www.freeprojectz.com/entity-relationship/doctor-appointment-system-er-diagram

https://www.researchgate.net/publication/40617163_A_Patient-centric_Attribute-based_Source-verifiable_Framework_for_Health_Record_Sharing#pf5