Medical billing and insurance management

23.04.2023

Group No -5

Group Id - G5_1

Bhavya Shah (202101426) (Contact Info : 8200090380)

Natvar Prajapati (202101402)

Vandan Bhuva (202101404)

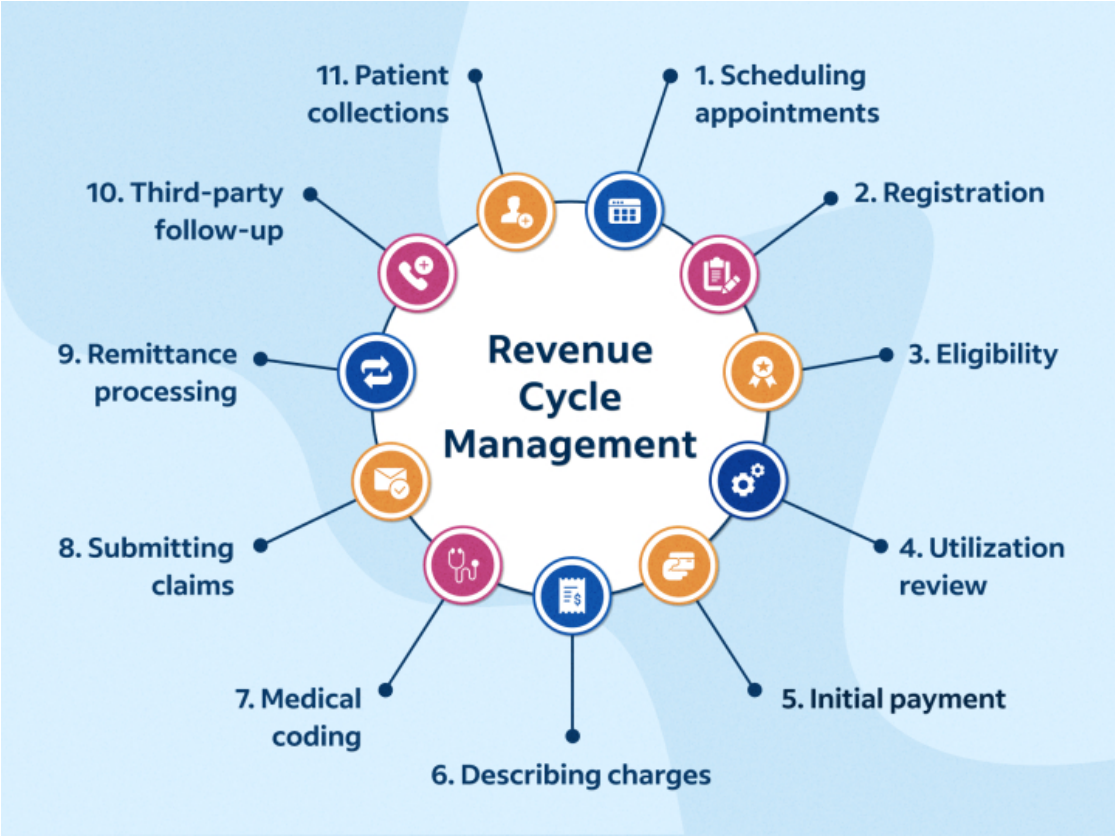
Shashank Upadhyay (202101411)

Objective

Revenue cycle management in healthcare aims to design and develop a comprehensive system that can effectively manage the financial processes involved in the healthcare revenue cycle. The system should streamline the billing and payment processes, ensure accurate and timely reimbursement, and improve overall revenue performance. This project will serve as a medium for a healthcare organization and medical professionals to check the claim status of patients and the payments collections.

Reason behind picking this project

This system is currently working in the USA and a solution to this is available but all of them are commercialized and are having higher rates. So, we can build a system which is efficient with privacy as data is quite private. So, we thought that this would be a great project to implement. -> Example website -> eClinicalWorks Error(will only open with VPN)

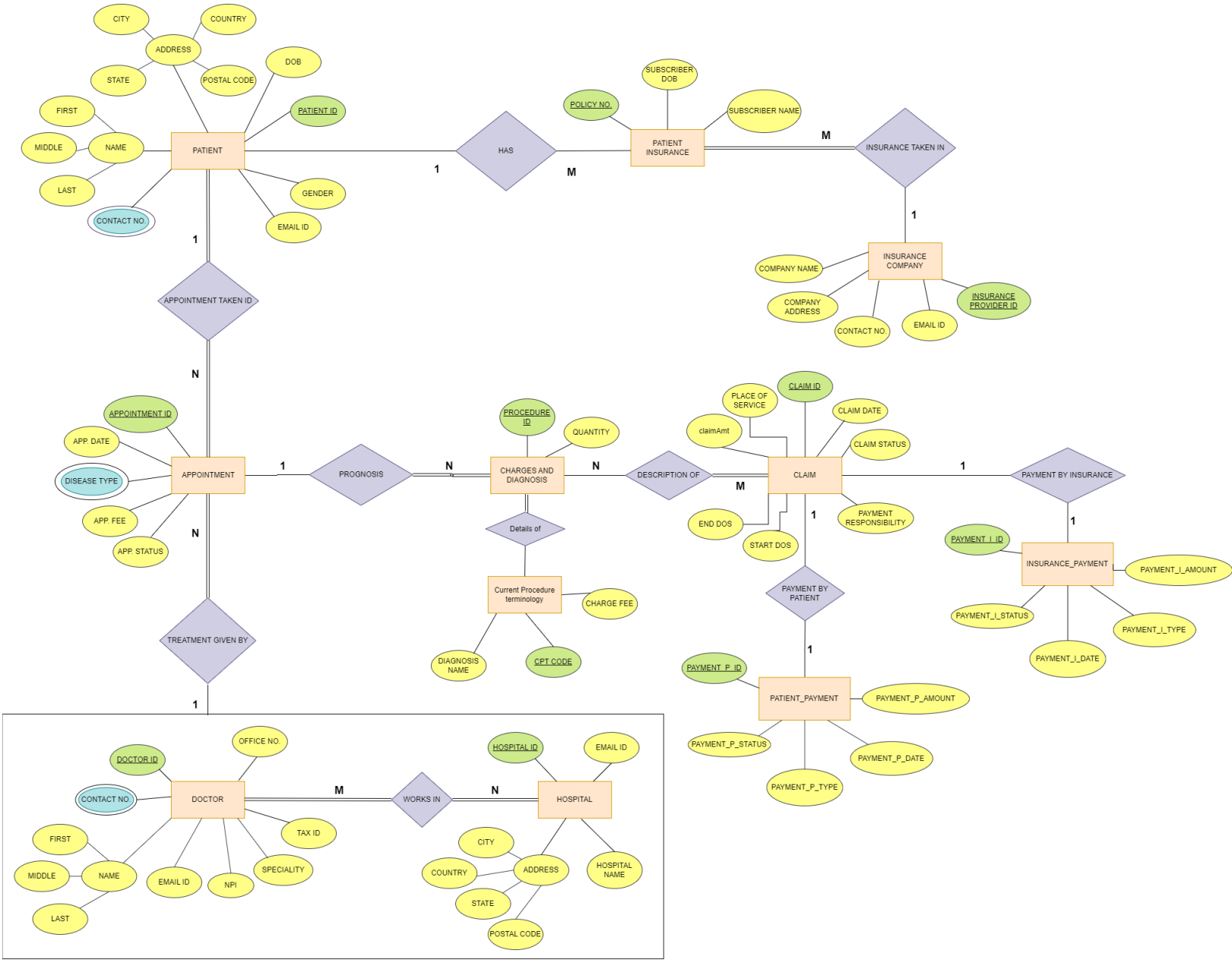


Description

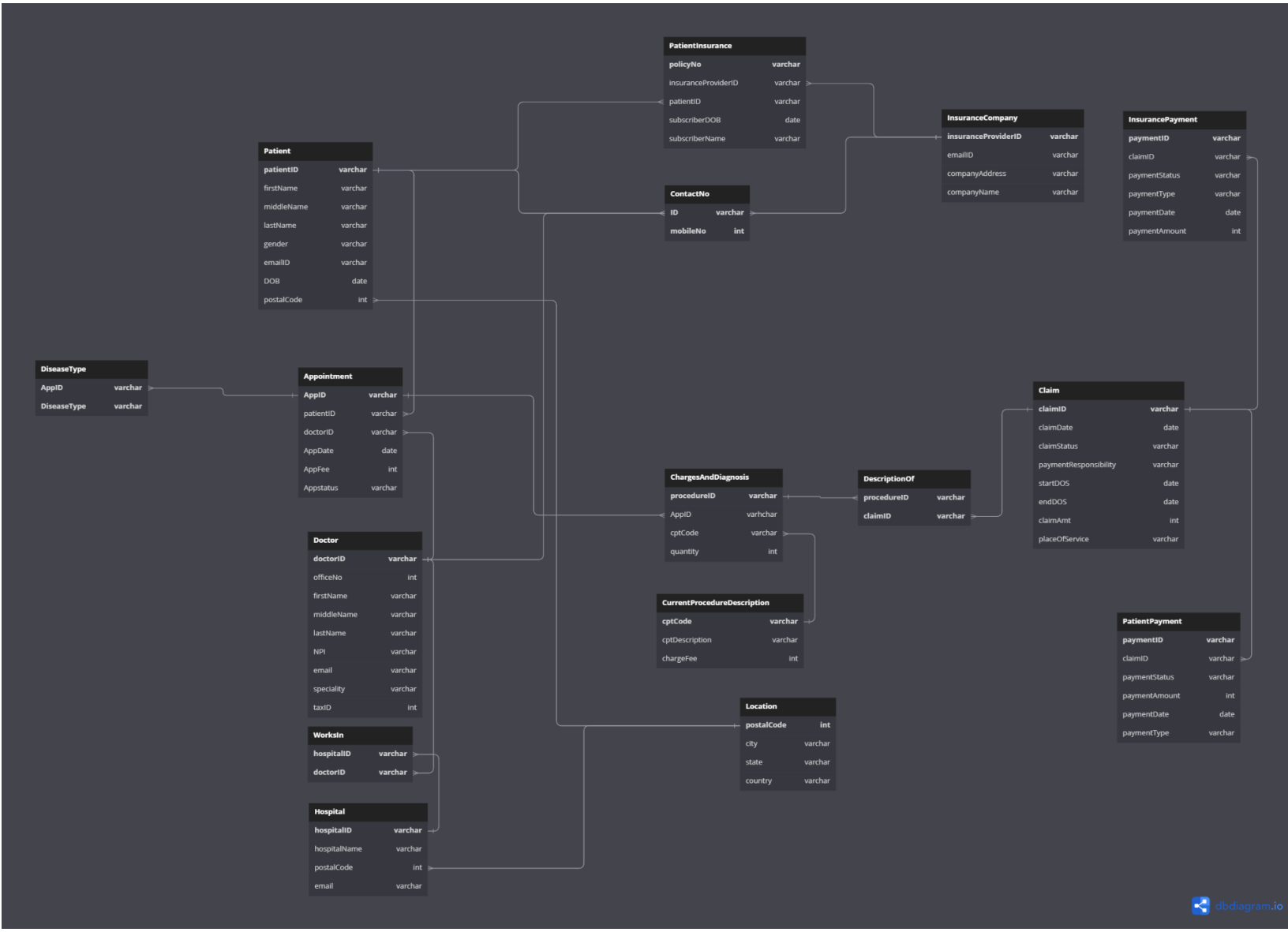
System

- The system will allow patients to submit claims electronically, and healthcare providers to track the status of claims, submit additional information as needed, and receive payment.
- The database will maintain a patient record that includes their personal information, medical history, and insurance coverage details. Each healthcare provider will have a unique identifier, and their contact information and services provided will be recorded in the database. Insurance providers will be identified by their name, contact information.
- Medical services will be classified based on standard medical coding systems, Current Procedural Terminology (CPT). The database will store details of each medical service, including its code, description, and cost. For each medical service provided to a patient, the healthcare provider will enter the details of the service in the system, including the date of service, code, and cost.
- The system will also handle the billing process, where the healthcare provider will submit the insurance claim to the patient's insurance provider. The system will automatically calculate the amount of payment due based on the medical service provided and the patient's insurance coverage details.
- The database will also store information about the status of each insurance claim, including whether it has been submitted, processed, or denied. In case of a denied claim, the reason for denial will be recorded in the system.
- Overall, the medical insurance claim management database project will streamline the process of managing medical insurance claims, improve efficiency, and ensure accurate and timely payments to doctors for medical services provided to patients.

ER Diagram:


CITY
COUNTRY
ADDRESS
STATE
POSTAL CODE
DOB
PATIENT ID
FIRST
MIDDLE
NAME
LAST
PATIENT
CONTACT NO.
GENDER
EMAIL ID
1
HAS
M
1
SUBSCRIBER DOB
POLICY NO.
SUBSCRIBER NAME
PATIENT INSURANCE
M
INSURANCE TAKEN IN
1
INSURANCE COMPANY
COMPANY NAME
COMPANY ADDRESS
CONTACT NO.
EMAIL ID
INSURANCE PROVIDER ID
1
APPOINTMENT TAKEN ID
N
APPOINTMENT ID
APP. DATE
DISEASE TYPE
APP. FEE
APP. STATUS
APPOINTMENT
1
PROGNOSIS
N
PROCEDURE ID
QUANTITY
CHARGES AND DIAGNOSIS
N
DESCRIPTION OF
M
PLACE OF SERVICE
CLAIM ID
claimAmt
CLAIM DATE
CLAIM STATUS
CLAIM
END DOS
START DOS
PAYMENT RESPONSIBILITY
1
PAYMENT BY INSURANCE
1
PAYMENT_I_ID
INSURANCE_PAYMENT
PAYMENT_I_AMOUNT
PAYMENT_I_STATUS
PAYMENT_I_TYPE
PAYMENT_I_DATE
Details of
Current Procedure terminology
CHARGE FEE
DIAGNOSIS NAME
CPT CODE
N
TREATMENT GIVEN BY
1
PAYMENT BY PATIENT
1
PAYMENT_P_ID
PATIENT_PAYMENT
PAYMENT_P_AMOUNT
PAYMENT_P_STATUS
PAYMENT_P_DATE
PAYMENT_P_TYPE
DOCTOR ID
OFFICE NO.
CONTACT NO.
DOCTOR
M
WORKS IN
N
HOSPITAL ID
EMAIL ID
HOSPITAL
TAX ID
FIRST
MIDDLE
NAME
LAST
EMAIL ID
NPI
SPECIALITY
CITY
COUNTRY
ADDRESS
STATE
POSTAL CODE
HOSPITAL NAME

Relational Diagram:



Tables:

Patient Information

Service provider first of all needs to create a new patient entry which will consist of information about patient such as their name, gender, dob, contact number , email, address (Postal code, city, state, country) and creating the patient will generate a unique PID (Patient ID) .

Appointment

Further, the provider typically carries out the task of entering the appointment details when a patient visits the doctor. So, this will contain information about each patient encounter including appt_id,patient_id, provider_id(referencing Provider table -> Contains information about the doctor), date of service, disease_type, payment_ID, appt_date, appt_fee.

Doctor

Next step will be to enter the information of the medical professional from whom patient has taken treatment from and this will include the information such as provider_id, provider_name, contact_no, office_no, provider_type, NPI(Non profit identifier), TAX_ID, address(Postal code, city, state, country),email_address.

Patient_Insurance

Following table stores the information about the patients insurance id which will be unique and the other required detais like pid, insurance_provider_id, policy_number, subscriber_name,subscriber_dob.

Insurance_company

This table contains details of insurance provider which will include unique insurance_provider_id, company name, address(city, state, zipcode), contactno, email_address.

Hospitals

This table contains details about hospitals which will include hospitalid which is a primary key, hospital name, city/state/email address.

Diagnosis and Treatment (Diagnosis Code)

This will have set of procedure and cpt codes with their description and the fee regarding each cpt code -> procedure_id , cptcode,qty

Current Procedure Description

This will link the appointment information table to procedure information and stores the data of quantity and amount charged for each procedure. -> CPT code,CPT description, charge fee

Claims

This table will have the information when the claim of patient is generated. Claim_id, appointment_id ,insurance_provider_id,totalCharges,claim_date,POS(place of service),claim_status, startDOS, endDOS,claimAmount

Insurance_Payments

This table will contain information about the payment done by the insurance company and its type. claim_id,payment_i_id ,payment_type , payment_status, payment_date

Patient_Payments

Next step, is to calculate the responsibilty of the patient to pay in which first step is to have payment amount ->billing amount-return amount(Insurance company),claim_id,payment_p_id ,payment_type and payment_status, payment_date. (This table will be containing the information related to the payment done by the patient)

Minimal FD and BCNF Forms:

Minimal FD:

A minimal cover of a set of functional dependencies (FD) α is a minimal set of dependencies F that is equivalent to β.

The formal definition is: A set of FD α to be minimal if it satisfies the following conditions −

- Every dependency in α has a single attribute for its right-hand side.
- We cannot replace any dependency X->A in α with a dependency Y->A, where Y is a proper subset of X, and still have a set of dependencies that is equivalent to α.
- We cannot remove any dependency from α and still have a set of dependencies that are equivalent to α.

By calculating the functional dependencies for various attributes of our database, by observing we found out that normally the minimal FD are those which are having FD : α→β, where α is the key of the set.

BCNF (Boyce-Codd normal form):

Boyce-Codd normal form (BCNF) is a normal form used in database normalization. It is a slightly stronger version of the third normal form (3NF). BCNF was developed in 1974 by Raymond F. Boyce and Edgar F. Codd to address certain types of anomalies not dealt with by 3NF as originally defined.
FD : α → β where α is a determinant and β is dependent.
A relation is in BCNF if every determinant α is a candidate key. A determinant is any attribute whose value determines other values within a row. A candidate

key is a minimal set of attributes that can uniquely identify each tuple in a relation.

1) Patient

R(PatientID, firstName,middleName,lastName, gender,email ID,DOB,postalCode)

Keys: PatientID

Minimal FD: PatientID → firstName PatientID → middleName PatientID → lastName PatientID → gender PatientID → emailID PatientID → DOB PatientID → postalCode
$(PatientID)^+$ = R(PatientID, firstName,middleName,lastName, gender,email ID,DOB,postalCode) Hence, PatientID is the key
BCNF Proof: For every minimal FD dependencies listed above α is the candidate key, hence the relation is in BCNF.

2) Location

R(postalCode,city,state,country)

Keys: postalCode

Minimal FD: postalCode → city postalCode → state postalCode → country
$(postalCode)^+$ = R(postalCode,city,state,country) Hence postalCode is the key
BCNF Proof: For every minimal FD dependencies listed above α is the candidate key, hence the relation is in BCNF.

3) ContactNo

R(ID, mobileNo)

Keys: ID,mobileNo

Minimal FD: ID → Mobile No.
$\{ID, mobileNo\}^+$ = R(ID, mobileNo) Hence {ID, mobileNo} is the key
BCNF Proof: For every minimal FD dependencies listed above α is the candidate key, hence the relation is in BCNF.

4) DescriptionOf

R(procedureID, claimID)

Keys: procedureID, claimID

Minimal FD: claimID → procedureID procedureID →claimID
{procedureID, claimID}$^{+}$ = R(procedureID, claimID) Hence {procedureID, claimID} is the key
BCNF Proof: For every minimal FD dependencies listed above α is the candidate key, hence the relation is in BCNF.

5) DiseaseType

R(AppId, DiseaseType)

Keys: AppID, DiseaseType

Minimal FD: AppId → DiseaseType
{AppId}$^{+}$ = R(AppId, DiseaseType) Hence {AppId} is the key
BCNF Proof: For every minimal FD dependencies listed above α is the candidate key, hence the relation is in BCNF.

6) Insurance Payment

R(paymentID, claimID,paymentStatus,paymentType, paymentDate, paymentAmount)

Keys: paymentID

Minimal FD: paymentID → claimID paymentID → paymentStatus paymentID → paymentType paymentID → paymentDate paymentID → paymentAmount
$(paymentID)^+$ = R(paymentID, claimID,paymentStatus,paymentType, paymentDate, paymentAmount) Hence paymentID is the key
BCNF Proof: For every minimal FD dependency listed above α is the candidate key, hence the relation is in BCNF.

7) Hospital

R(hospitalID, hospitalName, postalCode, email)

Keys: hospitalID

Minimal FD: hospitalID → hospitalName hospitalID → postalCode hospitalID → email
$(hospitalID)^+$ = R(hospitalID, hospitalName,postalCode, email) Hence hospitalID is the key
BCNF Proof: For every minimal FD dependency listed above α is the candidate key,

hence the relation is in BCNF.

8) Works_In

R(HospitalID,DoctorID)

Keys: hosptialID, doctorID

Minimal FD:

{hospitalID,doctorID}$^{+}$ = R(hospitalID,doctorID)
Hence {hospitalID,doctorID} is the key

BCNF Proof:

For every minimal FD dependencies listed above α is the candidate key, hence the relation is in BCNF.

9) Doctor

R(doctorID,officeNo,firstName,middleName,lastName,NPI,email ,specialty, taxID)

Keys: doctorID

Minimal FD:

{doctorID} → firstName
{doctorID} → middleName
{doctorID} → lastName
{doctorID} → specialty
{doctorID} → email

{doctorID} → NPI {doctorID} → taxID
{doctorID}$^{+}$ = R(doctorID,officeNo,firstName,middleName,lastName,NPI,email ,specialty, taxID) Hence {doctorID} is the key
BCNF Proof: For every minimal FD dependencies listed above α is the candidate key, hence the relation is in BCNF.

10) InsuranceCompany

R(insuranceProviderID, emailID, companyAddress, companyName)

Keys: insuranceProviderID

Minimal FD: insuranceProviderID → emailID insuranceProviderID → companyAddress insuranceProviderID → companyName
(insuranceProviderID)$^{+}$ = R(insuranceProviderID, emailID, companyAddress, companyName) Hence insuranceProviderID is the key
BCNF Proof: For every minimal FD dependency listed above α is the candidate key, hence the relation is in BCNF.

11) Claim

R(claimID, claimDate, claimStatus, paymentResponsibility, startDOS, endDos, placeOfService)

Keys: claimID

Minimal FD: claimID → claimDate claimID → claimStatus claimID → patientResponsibility claimID → startDOS claimID → endDOS claimID →claimAmt claimID → placeOfService
$(claimID)^+$ = R(claimID, claimDate, claimStatus, paymentResponsibility, startDOS, endDos, placeOfService) Hence claim ID is the key
BCNF Proof: For every minimal FD dependency listed above α is the candidate key, hence the relation is in BCNF.

12) PatientPayment

R(paymentID, claimID, paymentStatus, paymentAmount, paymentDate, paymentType)

Keys: paymentID

Minimal FD: paymentID → claimID paymentID → paymentStatus paymentID → paymentAmount

paymentID → paymentDate paymentID → paymentType
(paymentID)$^{+}$ = R(paymentID, claimID, paymentStatus, paymentAmount, paymentDate, paymentType) Hence paymentID is the key
BCNF Proof: For every minimal FD dependency listed above α is the candidate key, hence the relation is in BCNF.

13) ChargesAndDiagnosis

R(procedureID, AppID, cptCode, quantity)

Keys: procedureID

Minimal FD: {procedureID} → AppID {procedureID} → cptCode {procedureID} → quantity
(procedureID)$^{+}$ = R(diagnosisName, procedureID, AppID, chargeFee, cptCode, quantity) Hence {procedureID} is the key
BCNF Proof: For every minimal FD dependency listed above α is the candidate key, hence the relation is in BCNF.

14) PatientInsurance

R(policyNo, insuranceProviderID, patientID, subscriberDOB, subscriberName)

Keys: policyNo

Minimal FD: policyNo → insuranceProviderId policyNo → patientID policyNo → subscriberDOB policyNo → subscriberName
$(policyNo)^+$ = R(policyNo, insuranceProviderID, patientID, subscriberDOB, subscriberName) Hence policyNo is the key
BCNF Proof: For every minimal FD dependency listed above α is the candidate key, hence the relation is in BCNF.

15) Appointment

R(AppID, patientID, doctorID, AppDate, AppFee, AppStatus)

Keys: AppID

Minimal FD: AppID → patientID AppID → doctorID AppID → AppDate AppID → AppFee AppID → AppStatus
$(AppID)^+$ = R(AppID, patientID, doctorID, AppDate, AppFee, AppStatus) Hence AppID is the key
BCNF Proof: For every minimal FD dependency listed above α is the candidate key, hence the relation is in BCNF.

16) CurrentProcedureDescription

R(cptCode,cptDescription,chargeFee)

Keys: cptCode

Minimal FD: cptCode → cptDescription cptCode → chargeFee
(cptCode)$^{+}$ = R(cptDescription,chargeFee) Hence cptCode is the key
BCNF Proof: For every minimal FD dependency listed above α is the candidate key, hence the relation is in BCNF.

- **Some BCNF Decomposition:**

  - Initial table→Patientappointemenet(PatientID,firstName,middleName, lastName,gender,emailID,DOB,city,state,country,postalcode,AppID, doctorID, AppDate, AppFee, AppStatus)

    ### The minimal FD:-

    Patient→{PatientID,firstName,middleName,lastName,gender,emailID, DOB,city,saate,country,postalcode,AppID, doctorID, AppDate, AppFee, AppStatus}
    AppID→{AppID, doctorID, AppDate, AppFee, AppStatus}
    Postalcode→{city,state,country}

    ### Final tables:-

    If we apply BCNF decomposition to above table we can divide it in three tables as given below:-
    Patient(PatientID, firstName,middleName,lastName, gender,email ID,DOB,postalCode)

Appointment(AppID, patientID, doctorID, AppDate, AppFee, AppStatus)
Location(postalcode,city,state,country)

- **Here the table connection is mainted using postalcode{Location-Patient} and patientID{patient-Appointment}.**

- Initial table→Patientinsurance(policyNo, patientID, subscriberDOB, subscriberName,insuranceProviderID, insurance_emailID, companyAddress, companyName)

  ## The minimal FD:-

  policyno→{policyNo, insuranceProviderID, patientID, subscriberDOB, subscriberName,insuranceProviderID, emailID, companyAddress, companyName}
  insuranceProviderID→ {emailID, companyAddress, companyName}

  ## Final tables:-

  If we apply BCNF decomposition to above table we can divide it in two tables as given below:-
  Patientinsurance(policyNo, insuranceProviderID, patientID, subscriberDOB, subscriberName)
  Insurancecompany(insuranceProviderID, insurance_emailID, companyAddress, companyName)

- **Here the table connection is mainted using insuranceproviderID{patientinsurance-Insurancecompany}.**

DDL Scripts:

Attached in zip

Data Insertion scripts:

Attached in zip

SQL Queries:

Attached in zip