ITIS 6120 - APPLIED DATABASES PROJECT

GROUP 6

AN EMR FOR AN OUT PATIENT CLINIC

VENKATESH UMAMAHESWARAN
VINOTH SIVAKUMAR

CONTENTS

DESCRIPTION ABOUT THE PROJECT

An Electronic Medical Record (EMR) is a digital version of the traditional paper-based medical record for an individual. The EMR represents a medical record within a single facility, such as a doctor's office or a clinic. The Electronic Medical Record (EMR) system is a database management system that uses database technology to build, keep up and control different sorts of information about a man's therapeutic history and care crosswise over time. The DBMS can track and refresh all the data of enlisted patients in the therapeutic Center amid a specific period. Medical records are created when a patient receives treatment from a health professional. Records may incorporate the patients:

- Personal information
- Clinical care information
- Laboratory test results
- Medications prescribed
- Disabilities, allergies and;
- Life or accidental insurance with private insurers or government programs.

SCOPE

The medical record serves a variety of purposes and is essential to the proper functioning of the medical practice—especially in today’s complicated health care environment. The medical record is a key instrument used in planning, evaluating, and coordinating patient care in both the inpatient and the outpatient settings. The content of the medical record is essential for patient care, accreditation (if applicable to the practitioner), and reimbursement purposes. The medical record administrator oversees documenting patient data in the medical record. He or she is likewise in charge of entering the doctor's record into the database. He or she is likewise in charge of knowing medical insurance contract requirements, legitimate necessities relating to security and privacy of the patient.

BUSINESS RULE

The business rule for a medical record database:

- A **Patient** medical record is registered on the first visit to the **Doctor**. Patient's address is being stored in **Patient_Address** database.
- A patient can make many appointments with one or more doctors.
- A doctor can accept appointments with many patients.
- However, each **Appointment** is made with only one doctor, and each appointment references a single patient. The appointment is at a particular **Hospital_Location**
- If an appointment yields a **Visit** with a doctor, the visit yields a **Prescription** and prescription is written for the patient that has appropriate **Medicines**.
- With each visit, the patient's history is updated.
- Each visit creates a **Bill**. Each patient visit is billed by one doctor, and each doctor can bill many patients.
- Billing covers **Payment**. Each bill must be paid. However, a bill may be paid in many installments, and a payment may cover more than one bill.
- A patient may pay the bill directly and patient might be having an **Insurance_Policy**, or the bill may be the basis for a claim submitted to an **Insurance** company.
- If a bill is paid by an insurance company, the deductible is submitted to the patient for payment.

OBJECTIVES OF THE SYSTEM

- The database should support editing of existing records to correct data entry mistakes or legitimate changes of information (e.g. Change of address or insurance).

- The Database should support searching of patient records based on name, ID, and possibly other information such as visit dates.

- The Database should support reporting functions such as listing of all patients who satisfy certain selection criteria, such as those who have been given certain diagnosis, or who visited on certain days, or who have been seen by certain doctors, or combinations of these such as, the diagnoses of patients who visited the clinic twice within the shortest time interval.

- To have an authentication and role based access to all tables

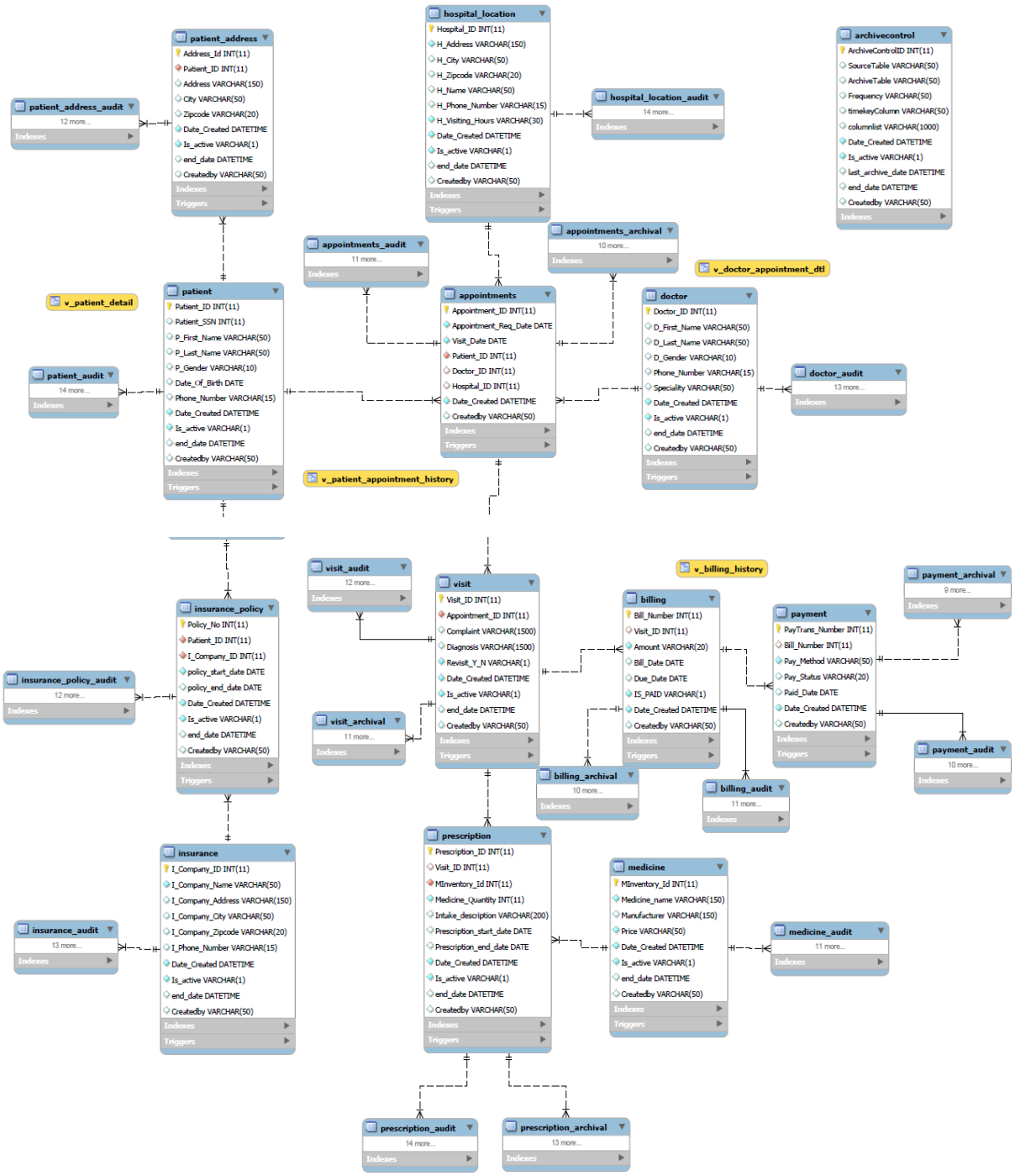
- Create a audit trail for error correction and maintenance.

- To create an archival system for data.

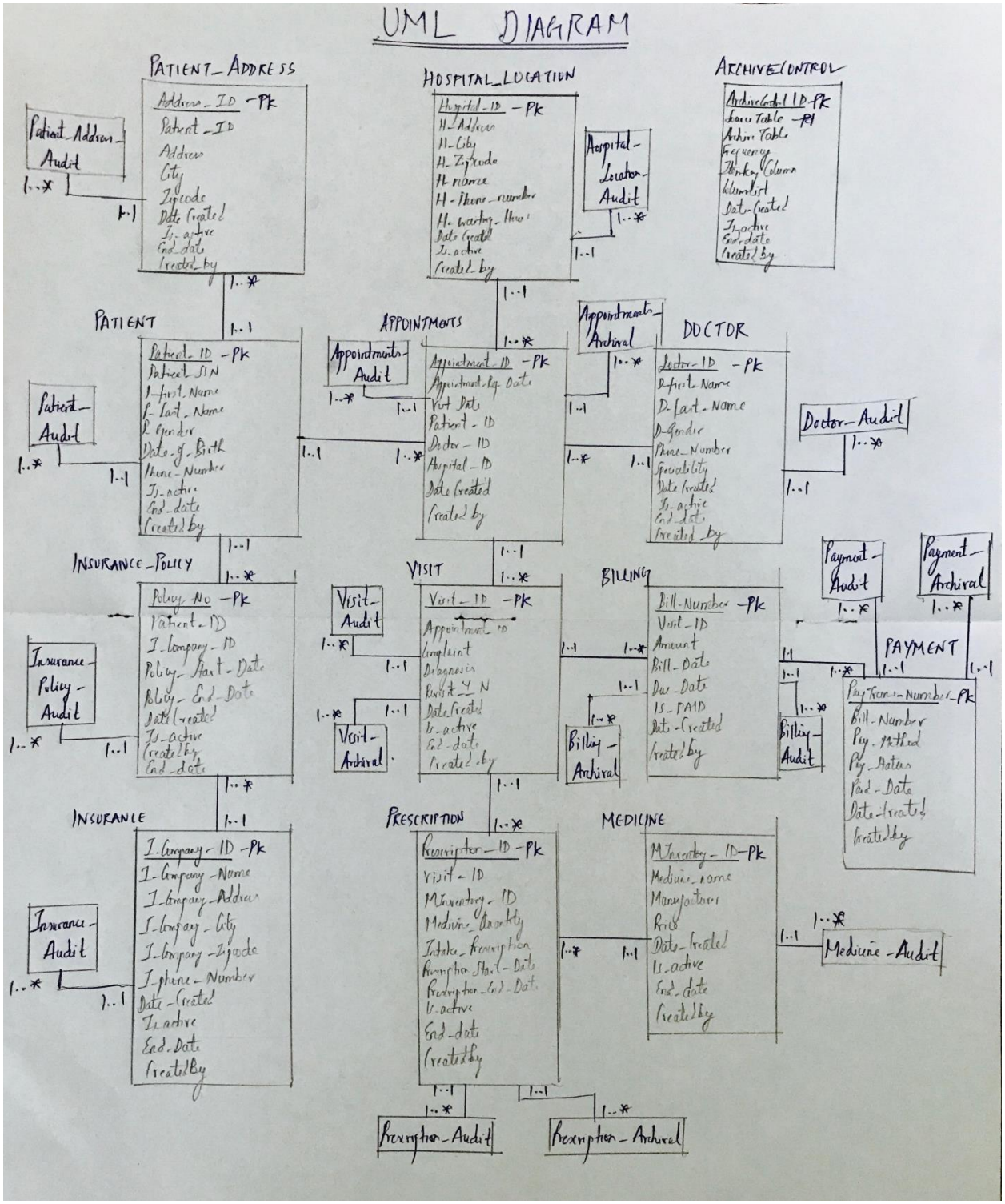
- Use triggers and stored procedures for maintaining data integrity, and ease of usage.

- Use views and indexes for easy access to frequently used data effeciently.

ER DIAGRAM


hospital_location
Hospital_ID INT(11)
H_Address VARCHAR(150)
H_City VARCHAR(50)
H_Zipcode VARCHAR(20)
H_Name VARCHAR(50)
H_Phone_Number VARCHAR(15)
H_Visiting_Hours VARCHAR(30)
Date_Created DATETIME
Is_active VARCHAR(1)
end_date DATETIME
Createdby VARCHAR(50)
patient_address
Address_Id INT(11)
Patient_ID INT(11)
Address VARCHAR(150)
City VARCHAR(50)
Zipcode VARCHAR(20)
Date_Created DATETIME
Is_active VARCHAR(1)
end_date DATETIME
Createdby VARCHAR(50)
archivecontrol
ArchiveControlID INT(11)
SourceTable VARCHAR(50)
ArchiveTable VARCHAR(50)
Frequency VARCHAR(50)
timekeyColumn VARCHAR(50)
columnlist VARCHAR(1000)
Date_Created DATETIME
Is_active VARCHAR(1)
last_archive_date DATETIME
end_date DATETIME
Createdby VARCHAR(50)
patient_address_audit
hospital_location_audit
appointments_audit
appointments_archival
v_doctor_appointment_dtl
v_patient_detail
patient
Patient_ID INT(11)
Patient_SSN INT(11)
P_First_Name VARCHAR(50)
P_Last_Name VARCHAR(50)
P_Gender VARCHAR(10)
Date_Of_Birth DATE
Phone_Number VARCHAR(15)
Date_Created DATETIME
Is_active VARCHAR(1)
end_date DATETIME
Createdby VARCHAR(50)
appointments
Appointment_ID INT(11)
Appointment_Req_Date DATE
Visit_Date DATE
Patient_ID INT(11)
Doctor_ID INT(11)
Hospital_ID INT(11)
Date_Created DATETIME
Createdby VARCHAR(50)
doctor
Doctor_ID INT(11)
D_First_Name VARCHAR(50)
D_Last_Name VARCHAR(50)
D_Gender VARCHAR(10)
Phone_Number VARCHAR(15)
Speciality VARCHAR(50)
Date_Created DATETIME
Is_active VARCHAR(1)
end_date DATETIME
Createdby VARCHAR(50)
patient_audit
doctor_audit
v_patient_appointment_history
visit_audit
visit
Visit_ID INT(11)
Appointment_ID INT(11)
Complaint VARCHAR(1500)
Diagnosis VARCHAR(1500)
Revisit_Y_N VARCHAR(1)
Date_Created DATETIME
Is_active VARCHAR(1)
end_date DATETIME
Createdby VARCHAR(50)
v_billing_history
billing
Bill_Number INT(11)
Visit_ID INT(11)
Amount VARCHAR(20)
Bill_Date DATE
Due_Date DATE
IS_PAID VARCHAR(1)
Date_Created DATETIME
Createdby VARCHAR(50)
payment
PayTrans_Number INT(11)
Bill_Number INT(11)
Pay_Method VARCHAR(50)
Pay_Status VARCHAR(20)
Paid_Date DATE
Date_Created DATETIME
Createdby VARCHAR(50)
payment_archival
payment_audit
insurance_policy
Policy_No INT(11)
Patient_ID INT(11)
I_Company_ID INT(11)
policy_start_date DATE
policy_end_date DATE
Date_Created DATETIME
Is_active VARCHAR(1)
end_date DATETIME
Createdby VARCHAR(50)
insurance_policy_audit
visit_archival
billing_archival
billing_audit
insurance
I_Company_ID INT(11)
I_Company_Name VARCHAR(50)
I_Company_Address VARCHAR(150)
I_Company_City VARCHAR(50)
I_Company_Zipcode VARCHAR(20)
I_Phone_Number VARCHAR(15)
Date_Created DATETIME
Is_active VARCHAR(1)
end_date DATETIME
Createdby VARCHAR(50)
prescription
Prescription_ID INT(11)
Visit_ID INT(11)
MInventory_Id INT(11)
Medicine_Quantity INT(11)
Intake_description VARCHAR(200)
Prescription_start_date DATE
Prescription_end_date DATE
Date_Created DATETIME
Is_active VARCHAR(1)
end_date DATETIME
Createdby VARCHAR(50)
medicine
MInventory_Id INT(11)
Medicine_name VARCHAR(150)
Manufacturer VARCHAR(150)
Price VARCHAR(50)
Date_Created DATETIME
Is_active VARCHAR(1)
end_date DATETIME
Createdby VARCHAR(50)
insurance_audit
medicine_audit
prescription_audit
prescription_archival

UML DIAGRAM:


UML DIAGRAM
PATIENT_ADDRESS
Address_ID -PK
Patient_ID
Address
City
Zipcode
Date Created
Is_active
End_date
Created_by
Patient_Addres_Audit
HOSPITAL_LOCATION
Hospital_ID -Pk
H_Address
H_City
H_Zipcode
H_name
H_Phone_number
H_working_Hours
Date Created
Is_active
Created by
Hospital_Location_Audit
ARCHIVECONTROL
ArchiveControl ID-Pk
Source Table -PK
Archive Table
Frequency
Monkey Column
Number of
Date_Created
Is_active
End_date
Created by
PATIENT
Patient_ID -Pk
Patient_SIN
P_first_Name
P_last_Name
P_Gender
Date_of_Birth
Phone_Number
Is_active
End_date
Created by
Patient_Audit
APPOINTMENTS
Appointment_ID -Pk
Appointment_by Date
Visit Date
Patient_ID
Doctor_ID
Hospital_ID
Date Created
Created by
Appointments_Audit
Appointments_Archival
DOCTOR
Doctor_ID -Pk
D_first_Name
D_last_Name
D_Gender
Phone_Number
Speciality
Date Created
Is_active
End_date
Created_by
Doctor_Audit
INSURANCE_POLICY
Policy_No -Pk
Patient_ID
I_Company_ID
Policy_Start_Date
Policy_End_Date
Date Created
Is_active
Created by
End_date
Insurance_Policy_Audit
VISIT
Visit_ID -Pk
Appointment_ID
Complaint
Diagnosis
Visit_Y_N
Date Created
Is_active
End_date
Created_by
Visit_Audit
Visit_Archival
BILLING
Bill_Number -Pk
Visit_ID
Amount
Bill_Date
Due_Date
IS_PAID
Date_Created
Created by
Billing_Archival
Billing_Audit
Payment_Audit
Payment_Archival
PAYMENT
PayTrans_Number -Pk
Bill_Number
Pay_method
Pay_dates
Paid_Date
Date_Created
Created by
INSURANCE
I_Company_ID -Pk
I_Company_Name
I_Company_Address
I_Company_City
I_Company_Zipcode
I_phone_Number
Date_Created
Is_active
End_Date
CreatedBy
Insurance_Audit
PRESCRIPTION
Prescription_ID -Pk
Visit_ID
MInventory_ID
Medicine_Quantity
Intake_Prescription
Prescription_Start_Date
Prescription_End_Date
Is_active
End_date
Created by
Prescription_Audit
Prescription_Archival
MEDICINE
MInventory_ID-Pk
Medicine_name
Manufacturers
Price
Date_Created
Is_active
End_date
Created by
Medicine_Audit

ENTITY AND ATTRIBUTES:

ENTITY	ATTRIBUTES
Patient	Patient_ID – PK, Patient_SSN, P_First_Name, P_Last_Name, P_Gender, Date_Of_Birth, Phone_Number, Date_Created, Is_active, End_date,Createdby
Doctor	Doctor_ID – PK, D_First_Name, D_Last_Name, D_Gender, Phone_Number, Speciality, Date_Created, Is_active, End_date,Createdby
Patient_Address	Address_ID – PK, Patient_ID, Address , City, Zipcode, Date_Created , Is_active , End_date ,Createdby
Visit	Visit_ID – PK, Appointment_ID, Complaint, Diagnosis, Revisit_Y_N, Date_Created, Is_active, End_date ,Createdby
Insurance_Policy	Policy_No, Patient_ID, I_Company_ID, policy_start_date, policy_end_date, Date_Created , Is_active , End_date ,Createdby
Insurance	I_Company_ID – PK, I_Company_Name, I_Company_Address, I_Company_City, I_Company_Zipcode, I_Phone_Number, Date_Created , Is_active , End_date ,Createdby
Prescription	Prescription_ID - PK, Visit_ID, MInventory_Id , Medicine_Quantity , Intake_description , Prescription_start_date , Prescription_end_date , Date_Created , Is_active , End_date ,Createdby
Medicine	MInventory_Id - PK , Medicine_name , Manufacturer , Price , Date_Created , Is_active , End_date ,Createdby
Hospital_Location	Hospital_ID - PK, H_Name , H_Address , H_City , H_Zipcode , H_Phone_Number , H_Visiting_Hours , Date_Created , Is_active , End_date ,Createdby
Appointments	Appointment_ID- PK, Appointment_Req_Date , Visit_Date , Patient_ID , Doctor_ID , Hospital_ID , Date_Created ,Createdby
Billing	Bill_Number - PK , Visit_ID, Amount , Bill_Date, Due_Date, Date_Created,Createdby
Payment	PayTrans_Number – PK, Bill_Number – PK, Pay_Method, Pay_Status, Paid_Date, Date_Created,Createdby

- **12 data tables,**
- **1 Archival control table,**
- **12 Audit tables,5 Archival tables.**

ROLE BASED ACTION CONTROL:

	hospitaladmin	dataoperator	physician	cashier	frontdesk
appointments					
archivecontrol					
billing					
doctor					
hospital_location					
insurance					
insurance_policy					
medicine					
patient					
patient_address					
payment					
prescription					
visit					
archival tables					
audit tables					

	all permissions
	SELECT, INSERT, UPDATE, DELETE
	SELECT,INSERT,UPDATE
	SELECT
	No permission

Role-Based Access Control has been implemented to limit the functions a user can perform,on tables and views .

The various permissions that various users have is been represented in the table above with color codes. The users are provided specific permissions to stored procedures as well.

The hospital admin is the most powerful user with all permissions including grant permission. The dataoperator has controls on all tables except the archival and audit tables. The users are provided specific control based on their roles. For eg. A Physcian will not be allowed to view the patients billing or payment details. Similarly the cashier will not have any access to patients prescription record.

SCREENSHOTS OF USER CREATION SCRIPTS:

```
create user 'dataoperator' identified by 'dataoperator';

GRANT SELECT,INSERT,UPDATE on appointments to dataoperator;
GRANT SELECT,INSERT,UPDATE on archivecontrol to dataoperator;
GRANT SELECT,INSERT,UPDATE on billing to dataoperator;
GRANT SELECT,INSERT,UPDATE on doctor to dataoperator;
GRANT SELECT,INSERT,UPDATE on hospital_location to dataoperator;
GRANT SELECT,INSERT,UPDATE on insurance to dataoperator;
GRANT SELECT,INSERT,UPDATE on insurance_policy to dataoperator;
GRANT SELECT,INSERT,UPDATE on medicine to dataoperator;
GRANT SELECT,INSERT,UPDATE on patient to dataoperator;
GRANT SELECT,INSERT,UPDATE on patient_address to dataoperator;
GRANT SELECT,INSERT,UPDATE on payment to dataoperator;
GRANT SELECT,INSERT,UPDATE on prescription to dataoperator;
GRANT SELECT,INSERT,UPDATE on visit to dataoperator;


GRANT SELECT on appointments_archival to dataoperator;
GRANT SELECT on billing_archival to dataoperator;
GRANT SELECT on payment_archival to dataoperator;
GRANT SELECT on prescription_archival to dataoperator;
GRANT SELECT on visit_archival to dataoperator;

GRANT SELECT on appointments_audit to dataoperator;
GRANT SELECT on billing_audit to dataoperator;
GRANT SELECT on doctor_audit to dataoperator;
GRANT SELECT on insurance_audit to dataoperator;
GRANT SELECT on insurance_policy_audit to dataoperator;
GRANT SELECT on medicine_audit to dataoperator;
GRANT SELECT on patient_audit to dataoperator;
GRANT SELECT on patient_address_audit to dataoperator;
GRANT SELECT on payment_audit to dataoperator;
GRANT SELECT on prescription_audit to dataoperator;
GRANT SELECT on visit_audit to dataoperator;

create user 'physician' identified by 'physician';
GRANT SELECT,INSERT,UPDATE on appointments to physician;
GRANT SELECT,INSERT,UPDATE on doctor to physician;
GRANT SELECT on hospital_location to physician;
```

```
create user 'frontdesk' identified by 'frontdesk';
GRANT SELECT,INSERT,UPDATE on appointments to frontdesk;
GRANT SELECT on billing to frontdesk;
GRANT SELECT,INSERT,UPDATE on doctor to frontdesk;
GRANT SELECT,INSERT,UPDATE on hospital_location to frontdesk;
GRANT SELECT,INSERT,UPDATE on insurance to frontdesk;
GRANT SELECT,INSERT,UPDATE on insurance_policy to frontdesk;
GRANT SELECT,INSERT,UPDATE on patient to frontdesk;
GRANT SELECT,INSERT,UPDATE on patient_address to frontdesk;
GRANT SELECT on payment to frontdesk;
GRANT SELECT,INSERT,UPDATE on visit to frontdesk;



GRANT EXECUTE ON procedure ins_appointments to hospitaladmin;
GRANT EXECUTE ON procedure ins_billing to hospitaladmin;
GRANT EXECUTE ON procedure ins_doctor to hospitaladmin;
GRANT EXECUTE ON procedure ins_hospital_location to hospitaladmin;
GRANT EXECUTE ON procedure ins_insurance to hospitaladmin;
GRANT EXECUTE ON procedure ins_insurance_policy to hospitaladmin;
GRANT EXECUTE ON procedure ins_medicine to hospitaladmin;
GRANT EXECUTE ON procedure ins_patient to hospitaladmin;
GRANT EXECUTE ON procedure ins_patient_address to hospitaladmin;
GRANT EXECUTE ON procedure ins_payment to hospitaladmin;
GRANT EXECUTE ON procedure ins_prescription to hospitaladmin;
GRANT EXECUTE ON procedure ins_visit to hospitaladmin;

GRANT EXECUTE ON procedure ins_appointments to dataoperator;
GRANT EXECUTE ON procedure ins_billing to dataoperator;
GRANT EXECUTE ON procedure ins_doctor to dataoperator;
GRANT EXECUTE ON procedure ins_hospital_location to dataoperator;
GRANT EXECUTE ON procedure ins_insurance to dataoperator;
GRANT EXECUTE ON procedure ins_insurance_policy to dataoperator;
GRANT EXECUTE ON procedure ins_medicine to dataoperator;
GRANT EXECUTE ON procedure ins_patient to dataoperator;
GRANT EXECUTE ON procedure ins_patient_address to dataoperator;
GRANT EXECUTE ON procedure ins_payment to dataoperator;
```

TRIGGERS:

- **We have implemented 3 triggers for each table**
- **Tablename_BU**
  - copies old data from data table to audit table with entry of U flag for update, also updates the user name doing update
- **Tablename_BD**
  - copies old data from data table to audit table with entry of D flag for Delete, also updates the user name doing delete
- **Tablename_BI**
  - To capture the user Name doing the insert in main table for audit
  - BI to change payment flag in billing table when entry made into payment table

```
delimiter |
CREATE    TRIGGER Patient_BU BEFORE UPDATE  ON Patient
    FOR EACH ROW
    BEGIN
        INSERT INTO Patient_audit set Patient_ID = old.Patient_ID,
        Patient_SSN=old.Patient_SSN,
        P_First_Name=old.P_First_Name,
        P_Last_Name=old.P_Last_Name,
        P_Gender =old.P_Gender,
        Date_Of_Birth=old.Date_Of_Birth,
        Phone_Number=old.Phone_Number,
        Date_Created=old.Date_Created,
        Is_active =old.Is_active,
        end_date =old.end_date,
        Createdby =old.Createdby,
        Changeby =SESSION_USER(),

        Changetype='U';
        if new.is_Active='N' then
         set new.end_Date=now();
        end if;
END;
        |
delimiter ;

delimiter |
CREATE    TRIGGER Patient_BD BEFORE delete  ON Patient
    FOR EACH ROW
    BEGIN
        INSERT INTO Patient_audit set Patient_ID = old.Patient_ID,
        Patient_SSN=old.Patient_SSN,
        P_First_Name=old.P_First_Name,
        P_Last_Name=old.P_Last_Name,
        P_Gender =old.P_Gender,
        Date_Of_Birth=old.Date_Of_Birth,
        Phone_Number=old.Phone_Number,
        Date_Created=old.Date_Created,
        Is_active =old.Is_active,
        end_date =old.end_date,
        Createdby =old.Createdby,
        Changeby =SESSION_USER(),

        Changetype='D';
        END;
        |
delimiter ;

delimiter |
CREATE    TRIGGER Patient_BI BEFORE insert  ON Patient
    FOR EACH ROW
    BEGIN
    set new.createdby=SESSION_USER();
        END;
        |
delimiter ;
```

```
delimiter |
CREATE    TRIGGER Payment_BI BEFORE insert  ON Payment
    FOR EACH ROW
    BEGIN
    set new.createdby=SESSION_USER();

        update billing t1 set t1.is_paid='Y' where t1.Bill_Number=new.Bill_Number;

        END;
        |

delimiter ;

=================

delimiter |
CREATE    TRIGGER Archivecontrol_BU BEFORE UPDATE  ON Archivecontrol
    FOR EACH ROW
    BEGIN

        if new.is_Active='N' then
         set new.end_Date=now();
        end if;
END;
        |
delimiter |
CREATE    TRIGGER Archivecontrol_BI BEFORE insert  ON Archivecontrol
    FOR EACH ROW
    BEGIN
    set new.createdby=SESSION_USER();
        END;
        |

delimiter ;
```

In the above trigger Payment_BI we have implemented a system for updating the billing record when any payment record is entered.

The Billing table contains a is paid field which has a default value of N. when any payment entry is made the the bill number in both the tables are matched and the value in flag field of N is changed to Y when the entry is made to the payment table for the corresponding bill number .

INDEXES:

Table Name	Indexes Created
Patient	(P_First_Name,P_Last_Name)
Patient_Address	Patient_ID
Doctor	(D_First_Name ,D_Last_Name)
Insurance	I_Company_Name
Hospital_Location	H_Name
Insurance_Policy	No Index
Medicine	(Medicine_name ,Manufacturer)
Appointments	Visit_Date ,Hospital_ID Patient_ID ,Doctor_ID
Visit	Appointment_ID
Prescription	Visit_ID
Billing	Bill_Date ,Due_Date ,Visit_ID
Payment	Pay_Status , Bill_Number

We have implemented indexes for easy access to frequently used data efficiently.

The fields represented within the braces can only be searched with that combination. The indexes are made keeping in mind the important fields which can be used as search criteria's for querying the table or creation of views

INDEX CREATION FOR ONE OF THE TABLES

SCREENSHOT:

```
create table IF NOT EXISTS Patient (
Patient_ID Int primary key AUTO_INCREMENT,
Patient_SSN INT,
P_First_Name VARCHAR(50),
P_Last_Name VARCHAR (50),
P_Gender VARCHAR (10),
Date_Of_Birth DATE,
Phone_Number VARCHAR (15),
Date_Created DATETIME NOT NULL DEFAULT NOW(),
Is_active VARCHAR(1) NOT NULL DEFAULT 'Y',
end_date DATETIME,
Createdby VARCHAR (50),
INDEX (P_First_Name,p_Last_Name)
);
```

VIEWS:

Created Views for various purpose using data and archival tables.

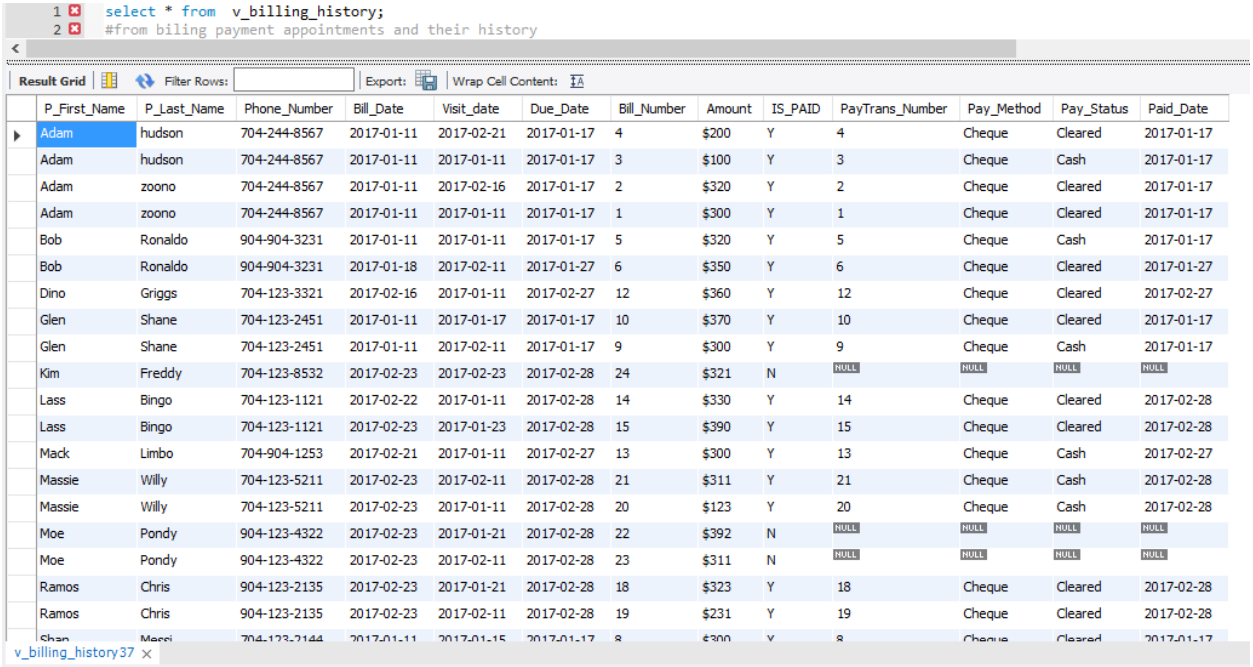
V_billing_history

V_doctor_appointment_dtl

V_patient_appointment_history

v_patient_detail

SCREENSHOTS:



```
1  select * from  v_billing_history;
2  #from biling payment appointments and their history
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

P_First_Name	P_Last_Name	Phone_Number	Bill_Date	Visit_date	Due_Date	Bill_Number	Amount	IS_PAID	PayTrans_Number	Pay_Method	Pay_Status	Paid_Date
Adam	hudson	704-244-8567	2017-01-11	2017-02-21	2017-01-17	4	$200	Y	4	Cheque	Cleared	2017-01-17
Adam	hudson	704-244-8567	2017-01-11	2017-01-11	2017-01-17	3	$100	Y	3	Cheque	Cash	2017-01-17
Adam	zoono	704-244-8567	2017-01-11	2017-02-16	2017-01-17	2	$320	Y	2	Cheque	Cleared	2017-01-17
Adam	zoono	704-244-8567	2017-01-11	2017-01-11	2017-01-17	1	$300	Y	1	Cheque	Cleared	2017-01-17
Bob	Ronaldo	904-904-3231	2017-01-11	2017-01-11	2017-01-17	5	$320	Y	5	Cheque	Cash	2017-01-17
Bob	Ronaldo	904-904-3231	2017-01-18	2017-02-11	2017-01-27	6	$350	Y	6	Cheque	Cleared	2017-01-27
Dino	Griggs	704-123-3321	2017-02-16	2017-01-11	2017-02-27	12	$360	Y	12	Cheque	Cleared	2017-02-27
Glen	Shane	704-123-2451	2017-01-11	2017-01-17	2017-01-17	10	$370	Y	10	Cheque	Cleared	2017-01-17
Glen	Shane	704-123-2451	2017-01-11	2017-02-11	2017-01-17	9	$300	Y	9	Cheque	Cash	2017-01-17
Kim	Freddy	704-123-8532	2017-02-23	2017-02-23	2017-02-28	24	$321	N	NULL	NULL	NULL	NULL
Lass	Bingo	704-123-1121	2017-02-22	2017-01-11	2017-02-28	14	$330	Y	14	Cheque	Cleared	2017-02-28
Lass	Bingo	704-123-1121	2017-02-23	2017-01-23	2017-02-28	15	$390	Y	15	Cheque	Cleared	2017-02-28
Mack	Limbo	704-904-1253	2017-02-21	2017-01-11	2017-02-27	13	$300	Y	13	Cheque	Cash	2017-02-27
Massie	Willy	704-123-5211	2017-02-23	2017-02-11	2017-02-28	21	$311	Y	21	Cheque	Cash	2017-02-28
Massie	Willy	704-123-5211	2017-02-23	2017-01-11	2017-02-28	20	$123	Y	20	Cheque	Cash	2017-02-28
Moe	Pondy	904-123-4322	2017-02-23	2017-01-21	2017-02-28	22	$392	N	NULL	NULL	NULL	NULL
Moe	Pondy	904-123-4322	2017-02-23	2017-02-11	2017-02-28	23	$311	N	NULL	NULL	NULL	NULL
Ramos	Chris	904-123-2135	2017-02-23	2017-01-21	2017-02-28	18	$323	Y	18	Cheque	Cleared	2017-02-28
Ramos	Chris	904-123-2135	2017-02-23	2017-02-11	2017-02-28	19	$231	Y	19	Cheque	Cleared	2017-02-28

v_billing_history37

This view v_billing_history is created using the patient, visit, billing, payment tables along with the archival tables. It provides a comprehensive view of all bill payments and un paid and details of each payment along with the patient detail and visit detail.

```
select * from v_billing_history where IS_PAID = 'N';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

P_First_Name	P_Last_Name	Phone_Number	Bill_Date	Visit_date	Due_Date	Bill_Number	Amount	IS_PAID	PayTrans_Number	Pay_Method	Pay_Status	Paid_Date
Kim	Freddy	704-123-8532	2017-02-23	2017-02-23	2017-02-28	24	$321	N	NULL	NULL	NULL	NULL
Moe	Pondy	904-123-4322	2017-02-23	2017-02-11	2017-02-28	23	$311	N	NULL	NULL	NULL	NULL
Moe	Pondy	904-123-4322	2017-02-23	2017-01-21	2017-02-28	22	$392	N	NULL	NULL	NULL	NULL

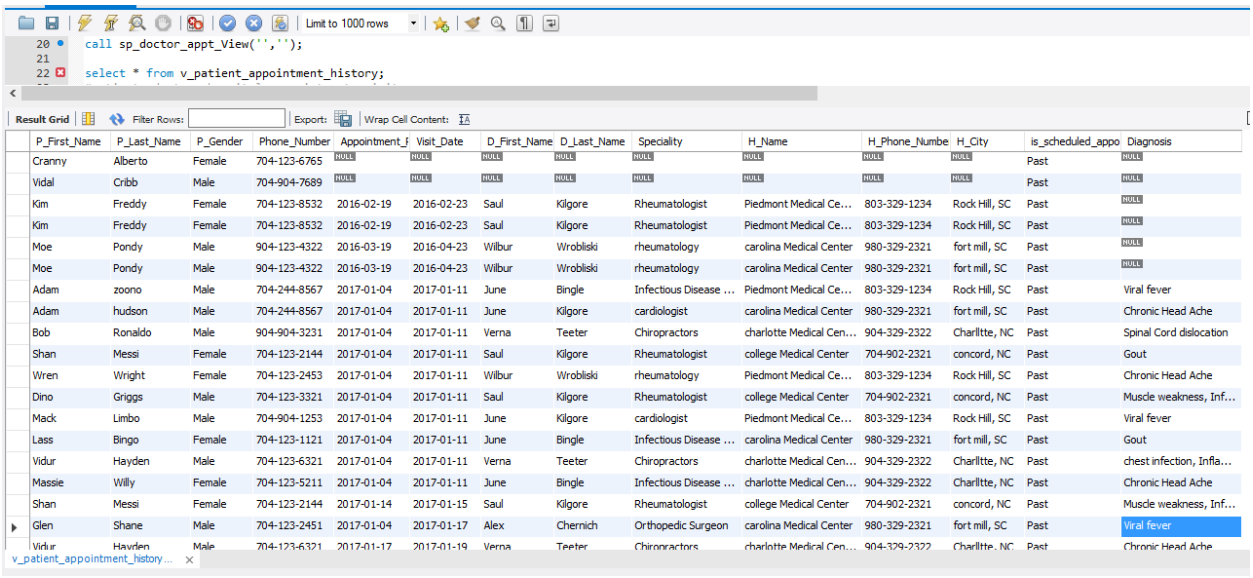
Using the view v_billing_history we can check for paid and unpaid details we can also filter based on due date, payment amount or paid date .

```
select * from v_doctor_appointment_dtl;
#doctor patient visit hospital
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Doctor_ID	D_First_Name	D_Last_Name	Phone_Number	Speciality	visit_Date	p_first_name	p_last_name	p_Gender	p_ph_number	H_Name	H_Address	H_City	H_Zipcode	H
1	June	Bingle	704-904-1546	Infectious Disease Specialist	2017-01-11	Adam	zoono	Male	704-244-8567	Piedmont Medical Center	222 S Herlong Ave	Rock Hill, SC	29732	80
1	June	Bingle	704-904-1546	Infectious Disease Specialist	2017-02-16	Adam	zoono	Male	704-244-8567	Piedmont Medical Center	222 S Herlong Ave	Rock Hill, SC	29732	80
1	June	Bingle	704-904-1546	Infectious Disease Specialist	2017-06-23	Adam	zoono	Male	704-244-8567	Piedmont Medical Center	222 S Herlong Ave	Rock Hill, SC	29732	80
2	Saul	Kilgore	704-918-1546	Rheumatologist	2017-02-23	Kim	Freddy	Female	704-123-8532	Piedmont Medical Center	222 S Herlong Ave	Rock Hill, SC	29732	80
2	Saul	Kilgore	704-918-1546	Rheumatologist	2017-06-23	Kim	Freddy	Female	704-123-8532	Piedmont Medical Center	222 S Herlong Ave	Rock Hill, SC	29732	80
3	June	Kilgore	706-904-1546	cardiologist	2017-01-11	Mack	Limbo	Male	704-904-1253	Piedmont Medical Center	222 S Herlong Ave	Rock Hill, SC	29732	80
4	Shelby	Lulewicz	706-904-1546	Acupuncturists	2017-02-11	Glen	Shane	Male	704-123-2451	Piedmont Medical Center	222 S Herlong Ave	Rock Hill, SC	29732	80
6	Wilbur	Wrobliski	980-904-1546	rheumatology	2017-01-11	Wren	Wright	Female	704-123-2453	Piedmont Medical Center	222 S Herlong Ave	Rock Hill, SC	29732	80
1	June	Bingle	704-904-1546	Infectious Disease Specialist	2017-01-11	Lass	Bingo	Female	704-123-1121	carolina Medical Center	12 north Ave	fort mill, SC	29782	98
1	June	Bingle	704-904-1546	Infectious Disease Specialist	2017-01-23	Lass	Bingo	Female	704-123-1121	carolina Medical Center	12 north Ave	fort mill, SC	29782	98
2	Saul	Kilgore	704-918-1546	Rheumatologist	2017-01-21	Ramos	Chris	Male	904-123-2135	carolina Medical Center	12 north Ave	fort mill, SC	29782	98
2	Saul	Kilgore	704-918-1546	Rheumatologist	2017-02-11	Ramos	Chris	Male	904-123-2135	carolina Medical Center	12 north Ave	fort mill, SC	29782	98
3	June	Kilgore	706-904-1546	cardiologist	2017-01-11	Adam	hudson	Male	704-244-8567	carolina Medical Center	12 north Ave	fort mill, SC	29782	98
3	June	Kilgore	706-904-1546	cardiologist	2017-02-21	Adam	hudson	Male	704-244-8567	carolina Medical Center	12 north Ave	fort mill, SC	29782	98
6	Wilbur	Wrobliski	980-904-1546	rheumatology	2017-01-21	Moe	Pondy	Male	904-123-4322	carolina Medical Center	12 north Ave	fort mill, SC	29782	98
6	Wilbur	Wrobliski	980-904-1546	rheumatology	2017-02-11	Moe	Pondy	Male	904-123-4322	carolina Medical Center	12 north Ave	fort mill, SC	29782	98
7	Alex	Chernich	704-980-1546	Orthopedic Surgeon	2017-01-17	Glen	Shane	Male	704-123-2451	carolina Medical Center	12 north Ave	fort mill, SC	29782	98
1	June	Bingle	704-904-1546	Infectious Disease Specialist	2017-01-11	Massie	Willy	Female	704-123-5211	charlotte Medical Center	211 south Ave	Charlltte, NC	28232	90

This view v_doctor_appoinemtnt_dtl is created by left joining patient, visit, and hospital details along with their archival data. It provides a comprehensive view of all details which can be filtered with additional filters for doctors, patients or hospital specific details which are required.



```
call sp_doctor_appt_View('','');

select * from v_patient_appointment_history;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

P_First_Name	P_Last_Name	P_Gender	Phone_Number	Appointment_I	Visit_Date	D_First_Name	D_Last_Name	Speciality	H_Name	H_Phone_Numbe	H_City	is_scheduled_appo	Diagnosis
Cranny	Alberto	Female	704-123-6765	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	Past	NULL
Vidal	Cribb	Male	704-904-7689	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	Past	NULL
Kim	Freddy	Female	704-123-8532	2016-02-19	2016-02-23	Saul	Kilgore	Rheumatologist	Piedmont Medical Ce...	803-329-1234	Rock Hill, SC	Past	NULL
Kim	Freddy	Female	704-123-8532	2016-02-19	2016-02-23	Saul	Kilgore	Rheumatologist	Piedmont Medical Ce...	803-329-1234	Rock Hill, SC	Past	NULL
Moe	Pondy	Male	904-123-4322	2016-03-19	2016-04-23	Wilbur	Wrobliski	rheumatology	carolina Medical Center	980-329-2321	fort mill, SC	Past	NULL
Moe	Pondy	Male	904-123-4322	2016-03-19	2016-04-23	Wilbur	Wrobliski	rheumatology	carolina Medical Center	980-329-2321	fort mill, SC	Past	NULL
Adam	zoono	Male	704-244-8567	2017-01-04	2017-01-11	June	Bingle	Infectious Disease ...	Piedmont Medical Ce...	803-329-1234	Rock Hill, SC	Past	Viral fever
Adam	hudson	Male	704-244-8567	2017-01-04	2017-01-11	June	Kilgore	cardiologist	carolina Medical Center	980-329-2321	fort mill, SC	Past	Chronic Head Ache
Bob	Ronaldo	Male	904-904-3231	2017-01-04	2017-01-11	Verna	Teeter	Chiropractors	charlotte Medical Cen...	904-329-2322	Charlltte, NC	Past	Spinal Cord dislocation
Shan	Messi	Female	704-123-2144	2017-01-04	2017-01-11	Saul	Kilgore	Rheumatologist	college Medical Center	704-902-2321	concord, NC	Past	Gout
Wren	Wright	Female	704-123-2453	2017-01-04	2017-01-11	Wilbur	Wrobliski	rheumatology	Piedmont Medical Ce...	803-329-1234	Rock Hill, SC	Past	Chronic Head Ache
Dino	Griggs	Male	704-123-3321	2017-01-04	2017-01-11	Saul	Kilgore	Rheumatologist	college Medical Center	704-902-2321	concord, NC	Past	Muscle weakness, Inf...
Mack	Limbo	Male	704-904-1253	2017-01-04	2017-01-11	June	Kilgore	cardiologist	Piedmont Medical Ce...	803-329-1234	Rock Hill, SC	Past	Viral fever
Lass	Bingo	Female	704-123-1121	2017-01-04	2017-01-11	June	Bingle	Infectious Disease ...	carolina Medical Center	980-329-2321	fort mill, SC	Past	Gout
Vidur	Hayden	Male	704-123-6321	2017-01-04	2017-01-11	Verna	Teeter	Chiropractors	charlotte Medical Cen...	904-329-2322	Charlltte, NC	Past	chest infection, Infla...
Massie	Willy	Female	704-123-5211	2017-01-04	2017-01-11	June	Bingle	Infectious Disease ...	charlotte Medical Cen...	904-329-2322	Charlltte, NC	Past	Chronic Head Ache
Shan	Messi	Female	704-123-2144	2017-01-14	2017-01-15	Saul	Kilgore	Rheumatologist	college Medical Center	704-902-2321	concord, NC	Past	Muscle weakness, Inf...
Glen	Shane	Male	704-123-2451	2017-01-04	2017-01-17	Alex	Chernich	Orthopedic Surgeon	carolina Medical Center	980-329-2321	fort mill, SC	Past	Viral fever
Vidur	Hayden	Male	704-123-6321	2017-01-17	2017-01-19	Verna	Teeter	Chiropractors	charlotte Medical Cen...	904-329-2322	Charlltte, NC	Past	Chronic Head Ache

v_patient_appointment_history...

This comprehensive view is created with patient details, appointment, visit and doctor details, along with the archival table for the fields. The tables are left joined with union between the data table and the archival table. Here the data operator or hospital admin can get a comprehensive record of all the visits to this day for various patients. This can be further filtered according to the needs to find specific information of patient, date of visit, doctor name or hospital location.

STORED PROCEDURES:

- **Stored procedure for data insert.**

```
DELIMITER //
CREATE PROCEDURE ins_patient_address
(IN Patient_ID  INT, IN Address VARCHAR(150), IN City VARCHAR (50), IN Zipcode VARCHAR (20))
BEGIN
  insert into Patient_Address (Patient_ID,Address,City,Zipcode) values (Patient_ID,Address,City,Zipcode);
END; //
```

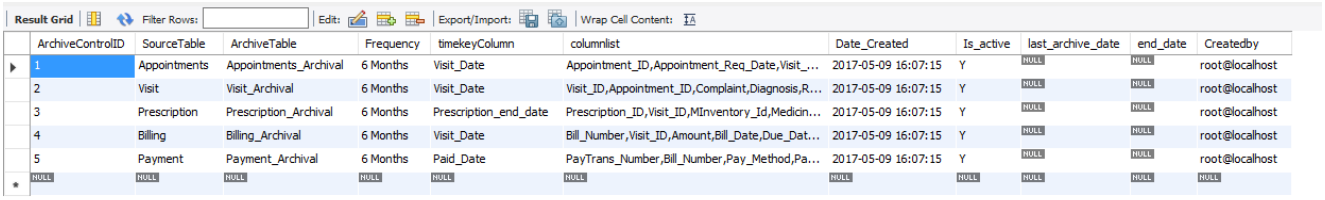
This stored procedure is used for inserting data. Permission to the stored procedure is provided only for specific user who have insert permission to the table. Such stored procedures are created for entry to all the tables in the Database.

- **Stored procedure for archive control**

```
DELIMITER //
CREATE PROCEDURE sp_archive
(IN in_SourceTable VARCHAR(50))
BEGIN
DECLARE done INT DEFAULT FALSE;
DECLARE v_ArchiveControlID Int;
DECLARE v_SourceTable VARCHAR(50);
DECLARE v_ArchiveTable VARCHAR(50);
DECLARE v_Frequency VARCHAR(50);
DECLARE v_timekeyColumn VARCHAR(50);
DECLARE v_columnlist varchar(1000);
DECLARE v_time_filter VARCHAR(50);
DECLARE table_list CURSOR FOR select ArchiveControlID,SourceTable,ArchiveTable,Frequency,timekeyColumn,columnlist from ArchiveControl where
DECLARE table_sub_list CURSOR FOR select ArchiveControlID,SourceTable,ArchiveTable,Frequency,timekeyColumn,columnlist from ArchiveControl wh
DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;


if in_SourceTable='' then
open table_list;
read_loop:LOOP
fetch table_list into v_ArchiveControlID,v_SourceTable,v_ArchiveTable,v_Frequency,v_timekeyColumn,v_columnlist;
IF done THEN
      LEAVE read_loop;
    END IF;
    if v_Frequency='6 Months' then
set @v_time_filter='<= NOW() - INTERVAL 6 Month';
elseif v_Frequency='12 Months' then
set @v_time_filter='<= NOW() - INTERVAL 12 Month';
elseif v_Frequency='1 Month' then
set @v_time_filter='<= NOW() - INTERVAL 1 Month';
end if;
```

This stored procedure for archival control is used to call the function for archival control. In this example we have configured archival to happen based on the detail in the table for 6 months 12 months or 1 month interval.



Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

ArchiveControlID	SourceTable	ArchiveTable	Frequency	timekeyColumn	columnlist	Date_Created	Is_active	last_archive_date	end_date	Createdby
1	Appointments	Appointments_Archival	6 Months	Visit_Date	Appointment_ID,Appointment_Req_Date,Visit_...	2017-05-09 16:07:15	Y	NULL	NULL	root@localhost
2	Visit	Visit_Archival	6 Months	Visit_Date	Visit_ID,Appointment_ID,Complaint,Diagnosis,R...	2017-05-09 16:07:15	Y	NULL	NULL	root@localhost
3	Prescription	Prescription_Archival	6 Months	Prescription_end_date	Prescription_ID,Visit_ID,MInventory_Id,Medicin...	2017-05-09 16:07:15	Y	NULL	NULL	root@localhost
4	Billing	Billing_Archival	6 Months	Visit_Date	Bill_Number,Visit_ID,Amount,Bill_Date,Due_Dat...	2017-05-09 16:07:15	Y	NULL	NULL	root@localhost
5	Payment	Payment_Archival	6 Months	Paid_Date	PayTrans_Number,Bill_Number,Pay_Method,Pa...	2017-05-09 16:07:15	Y	NULL	NULL	root@localhost
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

This file acts as control file to do the archival. Our setup has archival setup for 5 files.

Apointments, Visit, Prescription, Billing, Payment. The Column Sourcetable contains the table names, the destination table where the archival has to be done is in the archiveTable, the frequency can be manually changed according to our needs. We have setup 3 intervals 6 months, 12 months and 1 month. The timekey column contains the attribute on which the time interval wil be applied for the archival.

The columnlist contains all the coumns to be backed up, last archival date provides the last day the archival was done.

A user can archive all tables by running sp_archive(''), or do archival of a specific table by providing sp_archive('appointments').

- **Stored procedure for viewing specific data based on inputs.**

```
DELIMITER //
CREATE PROCEDURE sp_doctor_appt_dt_dtl_View
(IN in_D_First_Name VARCHAR(50), IN in_D_Last_Name VARCHAR(50),IN in_visit_Date date)
BEGIN
if  in_D_First_Name<>'' and in_D_Last_Name <>''  then

SELECT * from v_doctor_appointment_dtl where visit_Date=in_visit_Date and D_First_Name=in_D_First_Name and D_Last_Name=in_D_Last_Name;
elseif  in_D_First_Name <>'' and in_D_Last_Name ='' then

SELECT * from v_doctor_appointment_dtl where visit_Date=in_visit_Date and D_First_Name=in_D_First_Name ;
elseif  in_D_First_Name ='' and in_D_Last_Name <>'' then

SELECT * from v_doctor_appointment_dtl where visit_Date=in_visit_Date and D_Last_Name=in_D_Last_Name ;
elseif  in_D_First_Name ='' and in_D_Last_Name ='' then

SELECT * from v_doctor_appointment_dtl where visit_Date=in_visit_Date ;

elseif   in_visit_Date is null then

SELECT "Please input VisitDate to view results" from dual ;

end if;
  END //
DELIMITER ;
```

Stored procedures for calls on views

```
call sp_doctor_appt_dt_dtl_View('','','2017-01-11');
# ordered based on doctor id
```

Doctor_ID	D_First_Name	D_Last_Name	Phone_Number	Speciality	visit_Date	p_first_na	p_last_na	p_Gender	p_ph_number	H_Name	H_Address	H_City	H_Zipcode	H_Phone_Number	H_Visiting_Hours
1	June	Bingle	704-904-1546	Infectious Disease Speci...	2017-01-11	Adam	zoono	Male	704-244-8567	Piedmont Me...	222 S Herlong ...	Rock Hill, SC	29732	803-329-1234	M-F , 8 AM to 5 PM
1	June	Bingle	704-904-1546	Infectious Disease Speci...	2017-01-11	Lass	Bingo	Female	704-123-1121	carolina Medi...	12 north Ave	fort mill, SC	29782	980-329-2321	M-S , 8 AM to 5 PM
1	June	Bingle	704-904-1546	Infectious Disease Speci...	2017-01-11	Massie	Willy	Female	704-123-5211	charlotte Me...	211 south Ave	Charlltte, NC	28232	904-329-2322	M-SU , 8 AM to 11 PM
2	Saul	Kilgore	704-918-1546	Rheumatologist	2017-01-11	Shan	Messi	Female	704-123-2144	college Medic...	112 east Ave	concord, NC	22787	704-902-2321	M-F , 8 AM to 11 PM
2	Saul	Kilgore	704-918-1546	Rheumatologist	2017-01-11	Dino	Griggs	Male	704-123-3321	college Medic...	112 east Ave	concord, NC	22787	704-902-2321	M-F , 8 AM to 11 PM
3	June	Kilgore	706-904-1546	cardiologist	2017-01-11	Adam	hudson	Male	704-244-8567	carolina Medi...	12 north Ave	fort mill, SC	29782	980-329-2321	M-S , 8 AM to 5 PM
3	June	Kilgore	706-904-1546	cardiologist	2017-01-11	Mack	Limbo	Male	704-904-1253	Piedmont Me...	222 S Herlong ...	Rock Hill, SC	29732	803-329-1234	M-F , 8 AM to 5 PM
5	Verna	Teeter	705-904-1546	Chiropractors	2017-01-11	Bob	Ronaldo	Male	904-904-3231	charlotte Me...	211 south Ave	Charlltte, NC	28232	904-329-2322	M-SU , 8 AM to 11 PM
5	Verna	Teeter	705-904-1546	Chiropractors	2017-01-11	Vidur	Hayden	Male	704-123-6321	charlotte Me...	211 south Ave	Charlltte, NC	28232	904-329-2322	M-SU , 8 AM to 11 PM
6	Wilbur	Wrobliski	980-904-1546	rheumatology	2017-01-11	Wren	Wright	Female	704-123-2453	Piedmont Me...	222 S Herlong ...	Rock Hill, SC	29732	803-329-1234	M-F , 8 AM to 5 PM

This stored procedure sp_doctor_appt_dt_dtl_view can be used to see all the appointments on a single day. This stored procedure populates all the appointments for all doctors on that particular day.

```
call sp_doctor_appt_dt_dtl_View('saul','kilgore','2017-02-23');
#search based on doctor name and date fo visit
```

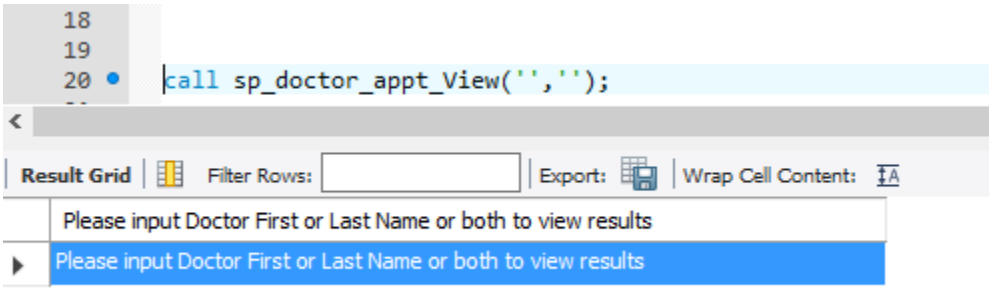
Doctor_ID	D_First_Na	D_Last_N	Phone_Numbe	Speciality	visit_Date	p_first_na	p_last_na	p_Gender	p_ph_number	H_Name	H_Address	H_City	H_Zipcode	H_Phone_Number	H_Visiting_Hours
2	Saul	Kilgore	704-918-1546	Rheumatologist	2017-02-23	Kim	Freddy	Female	704-123-8532	Piedmont Me...	222 S Herlong ...	Rock Hill, SC	29732	803-329-1234	M-F , 8 AM to 5 PM

In this view sp_doctor_appt_dt_dtl_view the doctor can provide his first or last name and the Date for which he would like to see all his appointments.

```
call sp_doctor_appt_View('June','Bingle');
# doctor view for patient details and hospital location
```

Doctor_ID	D_First_Na	D_Last_N	Phone_Numbe	Speciality	visit_Date	p_first_na	p_last_na	p_Gender	p_ph_number	H_Name	H_Address	H_City	H_Zipcode	H_Phone_Number	H_Visiting_Hours
1	June	Bingle	704-904-1546	Infectious Dis...	2017-01-11	Adam	zoono	Male	704-244-8567	Piedmont Me...	222 S Herlong ...	Rock Hill, SC	29732	803-329-1234	M-F , 8 AM to 5 PM
1	June	Bingle	704-904-1546	Infectious Dis...	2017-02-16	Adam	zoono	Male	704-244-8567	Piedmont Me...	222 S Herlong ...	Rock Hill, SC	29732	803-329-1234	M-F , 8 AM to 5 PM
1	June	Bingle	704-904-1546	Infectious Dis...	2017-01-11	Lass	Bingo	Female	704-123-1121	carolina Medi...	12 north Ave	fort mill, SC	29782	980-329-2321	M-S , 8 AM to 5 PM
1	June	Bingle	704-904-1546	Infectious Dis...	2017-01-23	Lass	Bingo	Female	704-123-1121	carolina Medi...	12 north Ave	fort mill, SC	29782	980-329-2321	M-S , 8 AM to 5 PM
1	June	Bingle	704-904-1546	Infectious Dis...	2017-01-11	Massie	Willy	Female	704-123-5211	charlotte Me...	211 south Ave	Charlltte, NC	28232	904-329-2322	M-SU , 8 AM to 11 PM
1	June	Bingle	704-904-1546	Infectious Dis...	2017-02-11	Massie	Willy	Female	704-123-5211	charlotte Me...	211 south Ave	Charlltte, NC	28232	904-329-2322	M-SU , 8 AM to 11 PM
1	June	Bingle	704-904-1546	Infectious Dis...	2017-06-23	Adam	zoono	Male	704-244-8567	Piedmont Me...	222 S Herlong ...	Rock Hill, SC	29732	803-329-1234	M-F , 8 AM to 5 PM

In the above example the sp_doctor_appt_view stored procedure is used to see all the appointments of the doctor along with the patient details and hospital details . this view is filtered based on the doctors first and last name.



This call which is to view all doctor appointmet will throw an error if a specific doctors first or last name is not provided.

CONCLUSION:

The learning curve that was obtained by doing this project was a great experience. The more challenges we faced, the more chances were there for us to gain knowledge. The requirements for the projects were spot on. By means of trying to implement all of the requirements mentioned for the project we got to acquire a decent perception of the big picture of the overall database operations. The feedbacks that we got from professor right through the project and at the end of the project will help us be prepared and more challenging for the REAL WORLD PROBLEMS that we might face in the database WORLD.

INDIVIDUAL REFLECTION:

Having chosen the aspect of Out Patient Clinic , we faced challenges initially to actually understand the medical system incorporated here. Because the medical system that is in existence in India(Native Country) is completely different from the medical system here in USA. This was our first step which helped us in choosing the necessary tables that was needed in our Database design. We understood the complexity of the EMR outpatient clinic and we came up with our ER diagram that establishes the relationship between the tables in the Database design.

Having finished our ER diagram, the next step was to create UML. We had to learn the concepts of functional dependancies and the relationships that the tables in our database had. It was a good learning curve that we obtained while doing our ER and UML diagrams.

We faced a lot of challenges while creating scripts for Triggers, Stored Procedures, User Creation. Everytime, we had to refer to the Canvas Videos and Internet to get a solution for the problem that we faced. But it was all worth it. We ended up learning a lot of things. Also the valuable feedbacks that we got from Professor helped us improve each and every time.

Venkatesh Umamaheswaran

Tasks completed: Creation of ER and UML diagrams, Insertion of data into the Database, Documentation, Triggers Creation, Test cases to verify the working of Datbase.

Vinoth Sivakumar

Tasks completed: Creation and Design of Database, Stored Procedures, Creating Test Scenarios, User Creation Scripts, Views Creation, Index Creation while creating database tables .