

```

CREATE TABLE IF NOT EXISTS "AspNetRoles" ( "Id" BIGINT PRIMARY KEY, "Name" VARCHAR(256) NOT NULL ); CREATE TABLE IF NOT
EXISTS "AspNetUsers" ( "Id" BIGINT PRIMARY KEY, "UserName" VARCHAR(256) NOT NULL, "PasswordHash" VARCHAR, "Email"
VARCHAR(256), "PhoneNumber" VARCHAR(20), "IP" VARCHAR(20), "CreatedDate" TIMESTAMP NOT NULL, "ModifiedDate" TIMESTAMP );
CREATE TABLE IF NOT EXISTS "Admin" ( "AdminId" SERIAL PRIMARY KEY, "AspNetUserId" BIGINT NOT NULL REFERENCES
"AspNetUsers"("Id"), "FirstName" VARCHAR(100) NOT NULL, "LastName" VARCHAR(100), "Email" VARCHAR(50) NOT NULL, "Mobile"
VARCHAR(20), "Address1" VARCHAR(500), "Address2" VARCHAR(500), "City" VARCHAR(100), "RegionId" INT, "Zip" VARCHAR(10), "AltPhone"
VARCHAR(20), "CreatedBy" VARCHAR(128) NOT NULL, "CreatedDate" TIMESTAMP NOT NULL, "ModifiedBy" BIGINT REFERENCES
"AspNetUsers"("Id"), "ModifiedDate" TIMESTAMP, "Status" SMALLINT, "IsDeleted" BIT, "RoleId" INT ); -- Create AspNetUserRoles table(4) CREATE
TABLE IF NOT EXISTS "AspNetUserRoles" ( "UserId" BIGINT NOT NULL REFERENCES "AspNetUsers"("Id"), "RoleId" BIGINT NOT NULL,
PRIMARY KEY ("UserId", "RoleId") ); -- Create BlockRequests table(5) CREATE TABLE IF NOT EXISTS "BlockRequests" ( "BlockRequestId" BIGINT
PRIMARY KEY, "PhoneNumber" VARCHAR(50), "Email" VARCHAR(50), "IsActive" BIT, "Reason" VARCHAR, "RequestId" BIGINT NOT NULL, "IP"
VARCHAR(20), "CreatedDate" TIMESTAMP, "ModifiedDate" TIMESTAMP ); -- Create CaseTag table(6) CREATE TABLE IF NOT EXISTS "CaseTag" (
"CaseTagId" SERIAL, "Name" VARCHAR(50) NOT NULL ); -- Create EmailLog table(7) CREATE TABLE IF NOT EXISTS "EmailLog" ( "EmailLogId"
DECIMAL(9, 0) PRIMARY KEY, "EmailTemplate" VARCHAR NOT NULL, "SubjectName" VARCHAR(200) NOT NULL, "EmailID" VARCHAR(200)
NOT NULL, "ConfirmationNumber" VARCHAR(200), "FilePath" VARCHAR, "RoleId" INT, "RequestId" INT, "AdminId" INT, "PhysicianId" INT,
"CreateDate" TIMESTAMP NOT NULL, "SentDate" TIMESTAMP, "IsEmailSent" BIT, "SentTries" INT, "Action" INT ); -- Create HealthProfessionalType
table(8) CREATE TABLE IF NOT EXISTS "HealthProfessionalType" ( "HealthProfessionalId" SERIAL PRIMARY KEY, "ProfessionName" VARCHAR(50)
NOT NULL, "CreatedDate" TIMESTAMP NOT NULL, "IsActive" BIT, "IsDeleted" BIT ); -- Create HealthProfessionals table(9) CREATE TABLE IF NOT
EXISTS "HealthProfessionals" ( "VendorId" SERIAL PRIMARY KEY, "VendorName" VARCHAR(100) NOT NULL, "Profession" INT REFERENCES
"HealthProfessionalType"("HealthProfessionalId"), "FaxNumber" VARCHAR(50) NOT NULL, "Address" VARCHAR(150), "City" VARCHAR(100), "State"
VARCHAR(50), "Zip" VARCHAR(50), "RegionId" INT, "CreatedDate" TIMESTAMP NOT NULL, "ModifiedDate" TIMESTAMP, "PhoneNumber"
VARCHAR(100), "IsDeleted" BIT, "IP" VARCHAR(20), "Email" VARCHAR(50), "BusinessContact" VARCHAR(100) ); -- Create Menu table(10) CREATE
TABLE IF NOT EXISTS "Menu" ( "MenuId" BIGINT PRIMARY KEY, "Name" VARCHAR(50) NOT NULL, "AccountType" SMALLINT NOT NULL
CHECK ("AccountType" IN (1, 2)), "SortOrder" INT ); -- Create OrderDetails table(11) CREATE TABLE IF NOT EXISTS "OrderDetails" ( "Id" SERIAL
PRIMARY KEY, "VendorId" INT, "RequestId" INT, "FaxNumber" VARCHAR(50), "Email" VARCHAR(50), "BusinessContact" VARCHAR(100),
"Prescription" VARCHAR, "NoOfRefill" INT, "CreatedDate" TIMESTAMP, "CreatedBy" VARCHAR(100) ); -- Create Physician table(12) CREATE TABLE
IF NOT EXISTS "Physician" ( "PhysicianId" SERIAL PRIMARY KEY, "AspNetUserId" BIGINT REFERENCES "AspNetUsers"("Id"), "FirstName"
VARCHAR(100) NOT NULL, "LastName" VARCHAR(100), "Email" VARCHAR(50) NOT NULL, "Mobile" VARCHAR(20), "MedicalLicense"
VARCHAR(500), "Photo" VARCHAR(100), "AdminNotes" VARCHAR(500), "IsAgreementDoc" BIT, "IsBackgroundDoc" BIT, "IsTrainingDoc" BIT,
"IsNonDisclosureDoc" BIT, "Address1" VARCHAR(500), "Address2" VARCHAR(500), "City" VARCHAR(100), "RegionId" INT, "Zip" VARCHAR(10),
"AltPhone" VARCHAR(20), "CreatedBy" BIGINT NOT NULL REFERENCES "AspNetUsers"("Id"), "CreatedDate" TIMESTAMP NOT NULL,
"ModifiedBy" BIGINT REFERENCES "AspNetUsers"("Id"), "ModifiedDate" TIMESTAMP, "Status" SMALLINT, "BusinessName" VARCHAR(100) NOT
NULL, "BusinessWebsite" VARCHAR(200) NOT NULL, "IsDeleted" BIT, "RoleId" INT, "NPINumber" VARCHAR(500), "IsLicenseDoc" BIT, "Signature"
VARCHAR(100), "IsCredentialDoc" BIT, "IsTokenGenerate" BIT, "SyncEmailAddress" VARCHAR(50) ); -- Create PhysicianLocation table(13) CREATE
TABLE IF NOT EXISTS "PhysicianLocation" ( "LocationId" SERIAL, "PhysicianId" INT NOT NULL REFERENCES "Physician"("PhysicianId"), "Latitude"

```

```

DECIMAL(11, 8), "Longitude" DECIMAL(11, 8), "CreateDate" TIMESTAMP, "PhysicianName" VARCHAR(50), "Address" VARCHAR(500) ); -- Create
PhysicianNotification table(14) CREATE TABLE IF NOT EXISTS "PhysicianNotification" ( "id" SERIAL PRIMARY KEY, "PhysicianId" INT NOT NULL
REFERENCES "Physician"("PhysicianId"), "IsNotificationStopped" BIT NOT NULL ); -- Create Region table(15) CREATE TABLE IF NOT EXISTS
"Region" ( "RegionId" SERIAL PRIMARY KEY, "Name" VARCHAR(50) NOT NULL, "Abbreviation" VARCHAR(50) ); -- Create PhysicianRegion
table(16) CREATE TABLE IF NOT EXISTS "PhysicianRegion" ( "PhysicianRegionId" SERIAL PRIMARY KEY, "PhysicianId" INT NOT NULL
REFERENCES "Physician"("PhysicianId"), "RegionId" INT NOT NULL REFERENCES "Region"("RegionId") ); -- Create RequestType table(17) CREATE
TABLE IF NOT EXISTS "RequestType" ( "RequestTypeId" SERIAL PRIMARY KEY, "Name" VARCHAR(50) NOT NULL ); -- Create Role table(18)
CREATE TABLE IF NOT EXISTS "Role" ( "RoleId" SERIAL PRIMARY KEY, "Name" VARCHAR(50) NOT NULL, "AccountType" SMALLINT NOT
NULL CHECK ( "AccountType" IN (1, 2)), "CreatedBy" VARCHAR(128) NOT NULL, "CreateDate" TIMESTAMP NOT NULL, "ModifiedBy"
VARCHAR(128), "ModifiedDate" TIMESTAMP, "IsDeleted" BIT NOT NULL, "IP" VARCHAR(20) ); -- Create RoleMenu table(19) CREATE TABLE IF
NOT EXISTS "RoleMenu" ( "RoleMenuId" SERIAL PRIMARY KEY, "RoleId" INT NOT NULL REFERENCES "Role"("RoleId"), "MenuId" INT NOT
NULL REFERENCES "Menu"("MenuId") ); -- Create Shift table(20) CREATE TABLE IF NOT EXISTS "Shift" ( "ShiftId" SERIAL PRIMARY KEY,
"PhysicianId" INT NOT NULL REFERENCES "Physician"("PhysicianId"), "StartDate" DATE NOT NULL, "IsRepeat" BIT NOT NULL, "WeekDays"
CHAR(7), "RepeatUpto" INT, "CreatedBy" BIGINT NOT NULL REFERENCES "AspNetUsers"("Id"), "CreateDate" TIMESTAMP NOT NULL, "IP"
VARCHAR(20) ); -- Create ShiftDetail table(21) CREATE TABLE IF NOT EXISTS "ShiftDetail" ( "ShiftDetailId" SERIAL PRIMARY KEY, "ShiftId" INT
NOT NULL REFERENCES "Shift"("ShiftId"), "ShiftDate" TIMESTAMP NOT NULL, "RegionId" INT, "StartTime" TIME NOT NULL, "EndTime" TIME
NOT NULL, "Status" SMALLINT NOT NULL, "IsDeleted" BIT NOT NULL, "ModifiedBy" BIGINT REFERENCES "AspNetUsers"("Id"), "ModifiedDate"
TIMESTAMP, "LastRunningDate" TIMESTAMP, "EventId" VARCHAR(100), "IsSync" BIT ); -- Create ShiftDetailRegion table(22) CREATE TABLE IF
NOT EXISTS "ShiftDetailRegion" ( "ShiftDetailRegionId" SERIAL PRIMARY KEY, "ShiftDetailId" INT NOT NULL REFERENCES
"ShiftDetail"("ShiftDetailId"), "RegionId" INT NOT NULL REFERENCES "Region"("RegionId"), "IsDeleted" BIT ); -- Create SMSLog table(23) CREATE
TABLE IF NOT EXISTS "SMSLog" ( "SMSLogID" DECIMAL(9) PRIMARY KEY, "SMSTemplate" VARCHAR NOT NULL, "MobileNumber"
VARCHAR(50) NOT NULL, "ConfirmationNumber" VARCHAR(200), "RoleId" INT, "AdminId" INT, "RequestId" INT, "PhysicianId" INT, "CreateDate"
TIMESTAMP NOT NULL, "SentDate" TIMESTAMP, "IsSMSSent" BIT, "SentTries" INT NOT NULL, "Action" INT ); -- Create User table(24) CREATE
TABLE IF NOT EXISTS "User" ( "UserId" INT PRIMARY KEY, "AspNetUserId" BIGINT REFERENCES "AspNetUsers"("Id"), "FirstName"
VARCHAR(100) NOT NULL, "LastName" VARCHAR(100), "Email" VARCHAR(50) NOT NULL, "Mobile" VARCHAR(20), "IsMobile" BIT, "Street"
VARCHAR(100), "City" VARCHAR(100), "State" VARCHAR(100), "RegionId" INT, "ZipCode" VARCHAR(10), "strMonth" VARCHAR(20), "intYear"
INT, "intDate" INT, "CreatedBy" VARCHAR(128) NOT NULL, "CreateDate" TIMESTAMP NOT NULL, "ModifiedBy" VARCHAR(128), "ModifiedDate"
TIMESTAMP, "Status" SMALLINT, "IsDeleted" BIT, "IP" VARCHAR(20), "IsRequestWithEmail" BIT ); -- Create AdminRegion table(25) CREATE TABLE
IF NOT EXISTS "AdminRegion" ( "AdminRegionId" SERIAL PRIMARY KEY, "AdminId" INT NOT NULL, "RegionId" INT NOT NULL, CONSTRAINT
"FK_AdminRegion_AdminId" FOREIGN KEY ("AdminId") REFERENCES "Admin" ("AdminId"), CONSTRAINT "FK_AdminRegion_RegionId" FOREIGN
KEY ("RegionId") REFERENCES "Region" ("RegionId") ); -- Create Business table(26) CREATE TABLE IF NOT EXISTS "Business" ( "BusinessId"
SERIAL PRIMARY KEY, "Name" VARCHAR(100) NOT NULL, "Address1" VARCHAR(500), "Address2" VARCHAR(500), "City" VARCHAR(50),
"RegionId" INT REFERENCES "Region"("RegionId"), "ZipCode" VARCHAR(10), "PhoneNumber" VARCHAR(20), "FaxNumber" VARCHAR(20),
"IsRegistered" BIT, "CreatedBy" BIGINT REFERENCES "AspNetUsers"("Id"), "CreateDate" TIMESTAMP NOT NULL, "ModifiedBy" BIGINT
REFERENCES "AspNetUsers"("Id"), "ModifiedDate" TIMESTAMP, "Status" SMALLINT, "IsDeleted" BIT, "IP" VARCHAR(20) ); -- Create Concierge
table(27) CREATE TABLE IF NOT EXISTS "Concierge" ( "ConciergeId" SERIAL PRIMARY KEY, "ConciergeName" VARCHAR(100) NOT NULL,

```

```

"Address" VARCHAR(150), "Street" VARCHAR(50) NOT NULL, "City" VARCHAR(50) NOT NULL, "State" VARCHAR(50) NOT NULL, "ZipCode"
VARCHAR(50) NOT NULL, "CreatedDate" TIMESTAMP NOT NULL, "RegionId" INT NOT NULL REFERENCES "Region"("RegionId"), "RoleId"
VARCHAR(20) ); -- Create Request table(28) CREATE TABLE IF NOT EXISTS "Request" ( "RequestId" SERIAL PRIMARY KEY, "RequestTypeId" INT
NOT NULL CHECK ("RequestTypeId" IN (1, 2, 3, 4)), "UserId" INT REFERENCES "User"("UserId"), "FirstName" VARCHAR(100), "LastName"
VARCHAR(100), "PhoneNumber" VARCHAR(23), "Email" VARCHAR(50), "Status" SMALLINT NOT NULL CHECK ("Status" IN (1, 2, 3, 4, 5, 6, 7, 8, 9,
10, 11, 12, 13, 14, 15)), "PhysicianId" INT REFERENCES "Physician"("PhysicianId"), "ConfirmationNumber" VARCHAR(20), "CreatedDate" TIMESTAMP
NOT NULL, "IsDeleted" BIT, "ModifiedDate" TIMESTAMP, "DeclinedBy" VARCHAR(250), "IsUrgentEmailSent" BIT NOT NULL, "LastWellnessDate"
TIMESTAMP, "IsMobile" BIT, "CallType" SMALLINT, "CompletedByPhysician" BIT, "LastReservationDate" TIMESTAMP, "AcceptedDate"
TIMESTAMP, "RelationName" VARCHAR(100), "CaseNumber" VARCHAR(50), "IP" VARCHAR(20), "CaseTag" VARCHAR(50), "CaseTagPhysician"
VARCHAR(50), "PatientAccountId" VARCHAR(128), "CreatedUserId" INT, "ContryCode" VARCHAR(10), ); -- Create RequestBusiness table(29) CREATE
TABLE IF NOT EXISTS "RequestBusiness" ( "RequestBusinessId" SERIAL PRIMARY KEY, "RequestId" INT NOT NULL REFERENCES
"Request"("RequestId"), "BusinessId" INT NOT NULL REFERENCES "Business"("BusinessId"), "IP" VARCHAR(20) ); -- Create RequestClient table(30)
CREATE TABLE IF NOT EXISTS "RequestClient" ( "RequestClientId" SERIAL PRIMARY KEY, "RequestId" INT NOT NULL REFERENCES
"Request"("RequestId"), "FirstName" VARCHAR(100) NOT NULL, "LastName" VARCHAR(100), "PhoneNumber" VARCHAR(23), "Location"
VARCHAR(100), "Address" VARCHAR(500), "RegionId" INT REFERENCES "Region"("RegionId"), "NotiMobile" VARCHAR(20), "NotiEmail"
VARCHAR(50), "Notes" VARCHAR(500), "Email" VARCHAR(50), "DateOfBirth" DATE, "strMonth" VARCHAR(20), "intYear" INT, "intDate" INT,
"IsMobile" BIT, "Street" VARCHAR(100), "City" VARCHAR(100), "State" VARCHAR(100), "ZipCode" VARCHAR(10), "CommunicationType"
SMALLINT, "RemindReservationCount" SMALLINT, "RemindHouseCallCount" SMALLINT, "IsSetFollowupSent" SMALLINT, "IP" VARCHAR(20),
"IsReservationReminderSent" SMALLINT, "Latitude" DECIMAL(11, 8), "Longitude" DECIMAL(11, 8), "ContryCode" VARCHAR(10) ); -- Create
RequestStatusLog table(31) CREATE TABLE IF NOT EXISTS "RequestStatusLog" ( "RequestStatusLogId" SERIAL PRIMARY KEY, "RequestId" INT NOT
NULL REFERENCES "Request"("RequestId"), "Status" SMALLINT NOT NULL, "PhysicianId" INT REFERENCES "Physician"("PhysicianId"), "AdminId"
INT REFERENCES "Admin"("AdminId"), "TransToPhysicianId" INT REFERENCES "Physician"("PhysicianId"), "Notes" VARCHAR(500), "CreatedDate"
TIMESTAMP NOT NULL, "IP" VARCHAR(20), "TransToAdmin" BIT ); -- Create RequestClosed table(32) CREATE TABLE IF NOT EXISTS
"RequestClosed" ( "RequestClosedId" SERIAL PRIMARY KEY, "RequestId" INT NOT NULL REFERENCES "Request"("RequestId"), "RequestStatusLogId"
INT NOT NULL REFERENCES "RequestStatusLog"("RequestStatusLogId"), "PhyNotes" VARCHAR(500), "ClientNotes" VARCHAR(500), "IP"
VARCHAR(20) ); -- Create RequestConcierge table(33) CREATE TABLE IF NOT EXISTS "RequestConcierge" ( "Id" SERIAL PRIMARY KEY, "RequestId"
INT NOT NULL REFERENCES "Request"("RequestId"), "ConciergeId" INT NOT NULL REFERENCES "Concierge"("ConciergeId"), "IP" VARCHAR(20)
); -- Create RequestNotes table(34) CREATE TABLE IF NOT EXISTS "RequestNotes" ( "RequestNotesId" SERIAL PRIMARY KEY, "RequestId" INT NOT
NULL REFERENCES "Request"("RequestId"), "strMonth" VARCHAR(20), "intYear" INT, "intDate" INT, "PhysicianNotes" VARCHAR(500), "AdminNotes"
VARCHAR(500), "CreatedBy" VARCHAR(128) NOT NULL, "CreatedDate" TIMESTAMP NOT NULL, "ModifiedBy" VARCHAR(128), "ModifiedDate"
TIMESTAMP, "IP" VARCHAR(20), "AdministrativeNotes" VARCHAR(500) ); -- Create RequestWiseFile table(35) CREATE TABLE IF NOT EXISTS
"RequestWiseFile" ( "RequestWiseFileID" SERIAL PRIMARY KEY, "RequestId" INT NOT NULL REFERENCES "Request"("RequestId"), "FileName"
VARCHAR(500) NOT NULL, "CreatedDate" TIMESTAMP NOT NULL, "PhysicianId" INT REFERENCES "Physician"("PhysicianId"), "AdminId" INT
REFERENCES "Admin"("AdminId"), "DocType" SMALLINT CHECK ("DocType" IN (1, 2, 3)), "IsFrontSide" BIT, "IsCompensation" BIT, "IP"
VARCHAR(20), "IsFinalize" BIT, "IsDeleted" BIT, "IsPatientRecords" BIT );

```