

Universidad Mariano Gálvez de Guatemala, Sede de San José Pinula.

Ingeniería en Sistemas de la información

Base de Datos 1

Ingeniero Ronald Leonel Ayala Garrido



Proyecto Final Manual Técnico

Integrantes:

José Rodrigo Monroy Vargas	3190-14-10964
Mario Adolfo Hernández Cabeza	3190-09-12203
Ingrid Alicia Marcela Tojin Mejía	3190-17-13430

Toma de Requerimientos.

Todo este proyecto comienza a la hora de requerir la capacidad para lograr resolver los problemas o necesidades del usuario con el fin de ahorrar el trabajo y recursos que al final de todo no satisface.

Los requerimientos que identificamos fueron:

Fue necesario por la gran influencia de crecimiento de uso de apps para médicos y pacientes en salud está causando una auténtica revolución en el sector. Si bien es posible que las más conocidas sean aquellas dirigidas al paciente, el desarrollo de aplicaciones para médicos y hospitales ha transformado y mejorado muchos aspectos de la práctica clínica. Desde ver los resultados de laboratorio de un paciente hasta consultar sus constantes vitales, estas herramientas ayudan al diagnóstico, monitorización y tratamiento de enfermedades.

Una de las principales motivaciones en la utilización de esta aplicación para médicos es cubrir la necesidad de acceder a la información y datos de salud de los pacientes provenientes de su Historia Clínica Electrónica. Así, estas apps para la Historia Clínica Electrónica tienen el objetivo de mejorar la atención y obtener mejores resultados de salud.

Será factible porque mejora la calidad y continuidad de la atención. La aplicación permitirá a los profesionales clínicos acceder a la Historia Clínica del paciente desde cualquier lugar, consiguiendo mejorar la calidad y continuidad de los cuidados al paciente.

Toma de Requerimientos

Fue necesario por la gran influencia de crecimiento de uso de apps para médicos y pacientes en salud está causando una auténtica revolución en el sector.

Esperamos que la aplicación Funcione de Una manera donde todos puedan Saber utilizarla

Utilizaremos SQL developer, Oracle Apex, Data modelos

Diccionario de Datos:

Tablas	▼
<input type="text"/>	
	
ALIMENTO	
BANCOS	
CATEGORIA	
CITA	
CITA_PERSONA	
CLIENTE	
CLINICAS	
COLONIA	
COMPANIA_TELEFONO	
COMPRAS	
CREDITO_CLIENTE	
CUENTA_PROVEEDORES	
CUENTAS_POR_PAGAR	
DEMO_CONSTRAINT_LOOKUP	
DEMO_CUSTOMERS	
DEMO_ORDER_ITEMS	
DEMO_ORDERS	
DEMO_PRODUCT_INFO	

Tablas



DEMO_STATES

DEMO_TAGS

DEMO_TAGS_SUM

DEMO_TAGS_TYPE_SUM

DEPARTAMENTO

DETALLE_COMPRA

DETALLE_DIETA

DETALLE_FACTURA

DETALLE_PAGO

DETALLE_PAGO_CLIENTE

DIAGNOSTICO

DIETA

DIRECCION

DIRECCION_PERSONA

DIRECCION_PROVEEDORES

DOCUMENTO_PERSONA

ENCARGADO_SEDE

ENFERMEDAD

Tablas



ESPECIALIDAD

ESPECIALIDAD_MEDICO

ESTADO

EXAMENES

EXAMEN_PERSONA

FACTURACION

FARMACIAS

HISTORIAL_CLINICO

INSUMO

INSUMO_CLINICAS

INSUMO_LABORATORIOS

INTERES

LAB_FARMACEUTICOS_INSUMOS

LOTES

MEDICO

MENU

MOVIMIENTOS

Tablas



SEDE

SEGURO

SIGNOS_VITALES

STATUS

SUSPENSION_MEDICA

SUSPENSION_PERSONA

TELEFONO

TELEFONO_PERSONA

TELEFONO_PROVEEDORES

TIEMPO_COMIDA

TIPO_DOCUMENTO

TIPO_DOCUMENTOS_PAGO

TIPO_ENFERMEDAD

TIPO_MOV

TIPO_ODONTOLOGIA

TIPO_PACIENTE_O

TIPO_PAGO

TIPO_PAGO

TIPO_PERSONAL_TECNICO

TIPO_RAZON_SOCIAL

TIPO_SANGRE

TIPO_SEGURO

TIPO_SERVICIO

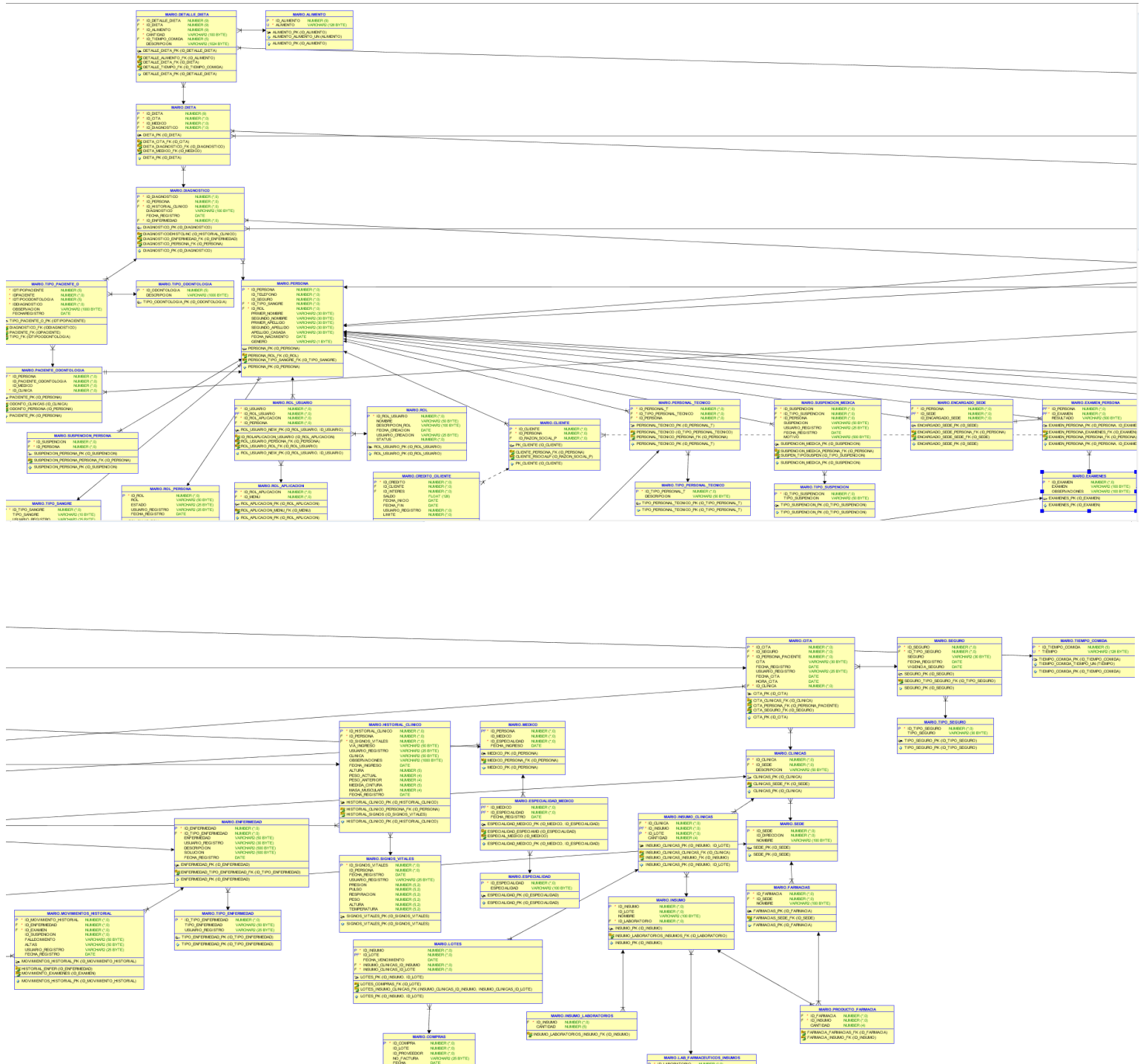
TIPO_SUSPENSION

TIPO_TELEFONO

TRANSACCIONES

ZONAS

Modelo E-R (El archivo está adjunto en la entrega con GitHub y Blackboard y puede ser apreciado con Data Modeler)



Script de la Base de Datos

```
CREATE TABLE bancos (  
    id_banco    INTEGER NOT NULL,  
    banco       VARCHAR2(100)  
);
```

```
ALTER TABLE bancos ADD CONSTRAINT bancos_pk PRIMARY KEY ( id_banco );
```

```
CREATE TABLE cita (  
    id_cita          INTEGER NOT NULL,  
    id_seguro        INTEGER NOT NULL,  
    id_persona_paciente  INTEGER NOT NULL,  
    cita             VARCHAR2(30),  
    fecha_registro   DATE,  
    usuario_registro  VARCHAR2(25),  
    fecha_cita       DATE,  
    hora_cita        DATE,  
    id_clinica       INTEGER NOT NULL  
);
```

```
ALTER TABLE cita ADD CONSTRAINT cita_pk PRIMARY KEY ( id_cita );
```

```
CREATE TABLE cita_persona (  
    id_cita    INTEGER NOT NULL,  
    id_persona INTEGER NOT NULL  
);
```

```
ALTER TABLE cita_persona ADD CONSTRAINT cita_persona_pk PRIMARY KEY (  
id_cita,  
  
id_persona );
```

```
create table CATEGORIA
```

```
(  
    ID_CATEGORIA      INTEGER          not null,  
    NOMBRE             VARCHAR2(30),  
    DESCRIPCION        VARCHAR2(100),  
    constraint PK_CATEGORIA primary key (ID_CATEGORIA)  
);
```

```
create table CLIENTE
```

```
(  
    ID_CLIENTE         INTEGER          not null,  
    ID_PERSONA         INTEGER          not null,  
    ID_RAZON_SOCIAL_P  INTEGER,  
    constraint PK_CLIENTE primary key (ID_CLIENTE)  
);
```

```
create table CREDITO_CLIENTE
```

```
(  
    ID_CREDITO         INTEGER          not null,  
    ID_CLIENTE         INTEGER,  
    ID_INTERES         INTEGER,  
    SALDO              FLOAT,  
    FECHA_INICIO       DATE,  
    FECHA_FIN          DATE,  
    USUARIO_REGISTRO   INTEGER,  
    LIMITE             INTEGER,  
    constraint PK_CREDITO_CLIENTE primary key (ID_CREDITO)  
);
```

```

create table DETALLE_FACTURA
(
    ID_DET_FAC            INTEGER            not null,
    ID_FACTURA           INTEGER            not null,
    CANTIDAD              INTEGER            not null,
    PRECIO                FLOAT              not null,
    constraint PK_DETALLE_FACTURA primary key (ID_DET_FAC)
);

```

```

create table DETALLE_PAGO_CLIENTE
(
    ID_DET_GAGO           INTEGER            not null,
    ID_TIPO_PAGO          INTEGER            not null,
    CANTIDAD              FLOAT              not null,
    constraint PK_DETALLE_PAGO_CLIENTE primary key (ID_DET_GAGO)
);

```

```

create table ESTADO
(
    ID_ESTADO             INTEGER            not null,
    ESTADO                VARCHAR2(15),
    constraint PK_ESTADO primary key (ID_ESTADO)
);

```

```

create table FACTURACION
(
    ID_FACTURA INTEGER            not null,
    ID_CLIENTE    INTEGER,
    ID_ESTADO     INTEGER,
    FECHA_TRANSACCION DATE,

```

```
USUARIO_REGISTRO    INTEGER,  
NIT                  VARCHAR2(30),  
constraint PK_FACTURACION primary key (ID_FACTURA)  
);
```

```
create table INTERES  
(  
    ID_INTERES        INTEGER            not null,  
    INTERES            FLOAT,  
    FECHA              DATE,  
    USUARIO_REGISTRO  INTEGER,  
    constraint PK_INTERES primary key (ID_INTERES)  
);
```

```
create table MOVIMIENTOS  
(  
    ID_MOV             INTEGER            not null,  
    ID_CREDITO         INTEGER,  
    ID_TIPO_MOV        INTEGER,  
    CANTIDAD           FLOAT,  
    FECHA              DATE,  
    DESCRIPCION        VARCHAR2(200),  
    USUARIO_REGISTRO  INTEGER,  
    constraint PK_MOVIMIENTOS primary key (ID_MOV)  
);
```

```
create table PRODUCTOS_SERVICIOS  
(  
    ID_PRO_SER         INTEGER            not null,  
    ID_CATEGORIA       INTEGER,
```

```
DESCRIPCION          VARCHAR2(200),
constraint PK_PRODUCTOS_SERVICIOS primary key (ID_PRO_SER)
);
```

```
create table TIPO_MOV
(
  ID_TIPO_MOV          INTEGER          not null,
  TIPO_MOV             VARCHAR2(50),
  DESCRIPCION          VARCHAR2(15),
  constraint PK_TIPO_MOV primary key (ID_TIPO_MOV)
);
```

```
create table TIPO_PAGO
(
  ID_TIPO_PAGO         INTEGER          not null,
  TIPO_PAGO            VARCHAR2(30),
  OTROS_DETALLES       VARCHAR2(40),
  constraint PK_TIPO_PAGO primary key (ID_TIPO_PAGO)
);
```

```
CREATE TABLE clinicas (
  id_clinica    INTEGER NOT NULL,
  id_sede       INTEGER NOT NULL,
  descripcion   VARCHAR2(50)
);
```

```
ALTER TABLE clinicas ADD CONSTRAINT clinicas_pk PRIMARY KEY ( id_clinica );
```

```

CREATE TABLE colonia (
    id_colonia    INTEGER NOT NULL,
    id_municipio  INTEGER NOT NULL,
    colonia       VARCHAR2(30)
);

ALTER TABLE colonia ADD CONSTRAINT colonia_pk PRIMARY KEY ( id_colonia );

CREATE TABLE compania_telefono (
    id_compania   INTEGER NOT NULL,
    compania      VARCHAR2(25)
);

ALTER TABLE compania_telefono ADD CONSTRAINT compania_telefono_pk PRIMARY KEY
( id_compania );

CREATE TABLE compras (
    id_compra     INTEGER NOT NULL,
    id_lote       INTEGER,
    id_proveedor  INTEGER,
    no_factura    VARCHAR2(25),
    fecha         DATE,
    recargos      NUMBER,
    descuentos    NUMBER,
    total         NUMBER
);

ALTER TABLE compras ADD CONSTRAINT compras_pk PRIMARY KEY ( id_compra );

```

```
CREATE TABLE cuenta_proveedores (  
    id_cuenta_prov    INTEGER NOT NULL,  
    id_proveedor      INTEGER NOT NULL,  
    id_banco_cuenta   INTEGER NOT NULL,  
    cuenta            VARCHAR2(50)  
);
```

```
ALTER TABLE cuenta_proveedores ADD CONSTRAINT cuenta_proveedores_pk PRIMARY  
KEY ( id_cuenta_prov );
```

```
CREATE TABLE cuentas_por_pagar (  
    id_cuenta          INTEGER NOT NULL,  
    id_proveedor       INTEGER NOT NULL,  
    id_compra           INTEGER,  
    id_pago_realizado  INTEGER NOT NULL,  
    id_status_pago     INTEGER NOT NULL  
);
```

```
ALTER TABLE cuentas_por_pagar ADD CONSTRAINT cuentas_por_pagar_pk PRIMARY KEY  
( id_cuenta );
```

```
CREATE TABLE departamento (  
    id_departamento   INTEGER NOT NULL,  
    id_pais            INTEGER NOT NULL,  
    departamento       VARCHAR2(50),  
    fecha_registro     DATE  
);
```

```
ALTER TABLE departamento ADD CONSTRAINT departamento_pk PRIMARY KEY (  
id_departamento );
```

```
CREATE TABLE detalle_compra (  
    id_compra        INTEGER NOT NULL,  
    id_producto      INTEGER,  
    cantidad         NUMBER,  
    precio_unidad    NUMBER,  
    total            NUMBER  
);
```

```
CREATE TABLE detalle_pago (  
    id_detalle_pago  INTEGER NOT NULL,  
    id_pago           INTEGER,  
    id_compra        INTEGER NOT NULL,  
    id_medio_pago     INTEGER,  
    id_cuenta_proveedor INTEGER,  
    id_tipo_documento_pago INTEGER,  
    no_doc_pago       NUMBER,  
    monto_pago        NUMBER  
);
```

```
ALTER TABLE detalle_pago ADD CONSTRAINT detalle_pago_pk PRIMARY KEY (  
id_detalle_pago );
```

```
CREATE TABLE diagnostico (  
    id_diagnostico    INTEGER,  
    id_persona        INTEGER NOT NULL,  
    id_historial_clinico INTEGER NOT NULL,  
    diagnostico       VARCHAR2(100),  
    fecha_registro    DATE,  
    id_enfermedad     INTEGER NOT NULL  
);
```



```
CREATE TABLE direccion (
```

```
    id_direccion          INTEGER NOT NULL,
```

```
    id_zona                INTEGER NOT NULL,
```

```
    id_municipio          INTEGER NOT NULL,
```

```
    id_colonia             INTEGER NOT NULL,
```

```
    calle_avenida         VARCHAR2(200),
```

```
    numero                 VARCHAR2(50),
```

```
    edificio              VARCHAR2(200)
```

```
);
```

```
ALTER TABLE direccion ADD CONSTRAINT direccion_pk PRIMARY KEY ( id_direccion
);
```

```
CREATE TABLE direccion_persona (
```

```
    id_direccion  INTEGER NOT NULL,
```

```
    id_persona    INTEGER NOT NULL
```

```
);
```

```
ALTER TABLE direccion_persona ADD CONSTRAINT direccion_persona_pk PRIMARY KEY
( id_direccion,id_persona );
```

```
CREATE TABLE direccion_proveedores (
```

```
    id_direccion_prov  INTEGER NOT NULL,
```

```
    id_proveedor        INTEGER
```

```
);
```

```
ALTER TABLE direccion_proveedores ADD CONSTRAINT direccion_proveedores_pk
PRIMARY KEY ( id_direccion_prov );
```

```
CREATE TABLE documento_persona (  
    id_tipo_documento    INTEGER NOT NULL,  
    id_persona            INTEGER NOT NULL  
);
```

```
ALTER TABLE documento_persona ADD CONSTRAINT documento_persona_pk PRIMARY KEY (  
id_tipo_documento );
```

```
CREATE TABLE encargado_sede (  
    id_persona            INTEGER NOT NULL,  
    id_sede               INTEGER NOT NULL,  
    id_encargado_sede     INTEGER  
);
```

```
ALTER TABLE encargado_sede ADD CONSTRAINT encargado_sede_pk PRIMARY KEY (  
id_sede );
```

```
CREATE TABLE enfermedad (  
    id_enfermedad         INTEGER NOT NULL,  
    id_tipo_enfermedad     INTEGER NOT NULL,  
    enfermedad            VARCHAR2(50),  
    usuario_registro       VARCHAR2(30),  
    descripcion            VARCHAR2(500),  
    solucion               VARCHAR2(500),  
    fecha_registro         DATE  
);
```

```
ALTER TABLE enfermedad ADD CONSTRAINT enfermedad_pk PRIMARY KEY (  
id_enfermedad );
```

```
CREATE TABLE especialidad (  
    id_especialidad    INTEGER NOT NULL,  
    especialidad       VARCHAR2(100)  
);
```

```
ALTER TABLE especialidad ADD CONSTRAINT especialidad_pk PRIMARY KEY (  
id_especialidad );
```

```
CREATE TABLE especialidad_medico (  
    id_medico          INTEGER NOT NULL,  
    id_especialidad    INTEGER NOT NULL,  
    fecha_registro     DATE  
);
```

```
ALTER TABLE especialidad_medico ADD CONSTRAINT especialidad_medico_pk PRIMARY  
KEY ( id_medico,  
  
id_especialidad );
```

```
CREATE TABLE examen_persona (  
    id_persona    INTEGER NOT NULL,  
    id_examen     INTEGER NOT NULL,  
    resultado     VARCHAR2(500)  
);
```

```
ALTER TABLE examen_persona ADD CONSTRAINT examen_persona_pk PRIMARY KEY (  
id_persona,  
  
id_examen );
```

```
CREATE TABLE examenes (  
    id_examen      INTEGER NOT NULL,  
    examen         VARCHAR2(100),  
    observaciones  VARCHAR2(100)  
);
```

```
ALTER TABLE examenes ADD CONSTRAINT examenes_pk PRIMARY KEY ( id_examen );
```

```
CREATE TABLE farmacias (  
    id_farmacia    INTEGER NOT NULL,  
    id_sede        INTEGER NOT NULL,  
    nombre         VARCHAR2(100)  
);
```

```
ALTER TABLE farmacias ADD CONSTRAINT farmacias_pk PRIMARY KEY ( id_farmacia  
);
```

```
CREATE TABLE historial_clinico (  
    id_historial_clinico  INTEGER NOT NULL,  
    id_persona            INTEGER NOT NULL,  
    id_signos_vitales     INTEGER NOT NULL,  
    via_ingreso           VARCHAR2(50),  
    usuario_registro      VARCHAR2(25),  
    clinica               VARCHAR2(50),  
    observaciones         VARCHAR2(1000),  
    fecha_ingreso         DATE,  
    altura               NUMBER(5),  
    peso_actual           NUMBER(4),  
    peso_anterior         NUMBER(4),  
    medida_cintura        NUMBER(5),
```

```
        masa_muscular          NUMBER(4),
        fecha_registro          DATE
    );
```

```
ALTER TABLE historial_clinico ADD CONSTRAINT historial_clinico_pk PRIMARY KEY
( id_historial_clinico );
```

```
CREATE TABLE insumo (
    id_insumo          INTEGER NOT NULL,
    id_lote            INTEGER,
    nombre             VARCHAR2(100),
    id_laboratorio     INTEGER NOT NULL
);
```

```
ALTER TABLE insumo ADD CONSTRAINT insumo_pk PRIMARY KEY ( id_insumo );
```

```
CREATE TABLE insumo_clinicas (
    id_clinica         INTEGER NOT NULL,
    id_insumo          INTEGER NOT NULL,
    id_lote            INTEGER NOT NULL,
    cantidad           NUMBER(4)
);
```

```
ALTER TABLE insumo_clinicas ADD CONSTRAINT insumo_clinicas_pk PRIMARY KEY (
id_insumo,
```

```
id_lote );
```

```
CREATE TABLE insumo_laboratorios (
    id_insumo          INTEGER NOT NULL,
    cantidad           NUMBER(5)
);
```

```
CREATE TABLE lab_farmaceuticos_insumos (  
    id_laboratorio    INTEGER NOT NULL,  
    nombre            VARCHAR2(100)  
);
```

```
ALTER TABLE lab_farmaceuticos_insumos ADD CONSTRAINT laboratorios_insumos_pk  
PRIMARY KEY ( id_laboratorio );
```

```
CREATE TABLE lotes (  
    id_insumo          INTEGER NOT NULL,  
    id_lote            INTEGER NOT NULL,  
    fecha_vencimiento  DATE,  
    insumo_clinicas_id_insumo  INTEGER NOT NULL,  
    insumo_clinicas_id_lote    INTEGER NOT NULL  
);
```

```
ALTER TABLE lotes ADD CONSTRAINT lotes_pk PRIMARY KEY ( id_insumo,  
                                                         id_lote );
```

```
CREATE TABLE medico (  
    id_persona         INTEGER NOT NULL,  
    id_medico          INTEGER,  
    id_especialidad    INTEGER NOT NULL,  
    fecha_ingreso       DATE  
);
```

```
ALTER TABLE medico ADD CONSTRAINT medico_pk PRIMARY KEY ( id_persona );
```

```
CREATE TABLE menu (  
    id_menu    INTEGER NOT NULL,  
    nombre     VARCHAR2(50),  
    url        VARCHAR2(500),  
    id_padre   INTEGER  
);
```

```
ALTER TABLE menu ADD CONSTRAINT menu_pk PRIMARY KEY ( id_menu );
```

```
CREATE TABLE movimientos_historial (  
    id_movimiento_historial  INTEGER NOT NULL,  
    id_enfermedad            INTEGER NOT NULL,  
    id_examen                INTEGER NOT NULL,  
    id_suspension            INTEGER,  
    fallecimiento            VARCHAR2(50),  
    altas                   VARCHAR2(50),  
    usuario_registro         VARCHAR2(25),  
    fecha_registro           DATE  
);
```

```
ALTER TABLE movimientos_historial ADD CONSTRAINT movimientos_historial_pk  
PRIMARY KEY ( id_movimiento_historial );
```

```
CREATE TABLE municipios (  
    id_municipio    INTEGER NOT NULL,  
    id_departamento  INTEGER,  
    id_pais          INTEGER,  
    municipio        VARCHAR2(50),  
    fecha_registro   DATE  
);
```

```
ALTER TABLE municipios ADD CONSTRAINT municipios_pk PRIMARY KEY (
id_municipio );
```

```
CREATE TABLE paciente_odontologia (
    id_persona            INTEGER NOT NULL,
    id_paciente_odontologia  INTEGER,
    id_medico              INTEGER,
    id_clinica              INTEGER NOT NULL
);
```

```
CREATE TABLE pago_proveedores (
    id_pago              INTEGER NOT NULL,
    fecha_pago           DATE,
    id_proveedor          INTEGER NOT NULL
);
```

```
ALTER TABLE pago_proveedores ADD CONSTRAINT pago_pk PRIMARY KEY ( id_pago );
```

```
CREATE TABLE pagos_clientes (
    id_pago              INTEGER NOT NULL,
    id_transaccion        INTEGER NOT NULL,
    id_tipo_pago          INTEGER NOT NULL,
    monto_cancelado        NUMBER(5,2),
    usuario_registro       VARCHAR2(25)
);
```

```
ALTER TABLE pagos_clientes ADD CONSTRAINT pagos_clientes_pk PRIMARY KEY (
id_pago );
```



```
CREATE TABLE pais (  
    id_pais    INTEGER NOT NULL,  
    pais       VARCHAR2(30)  
);
```

```
ALTER TABLE pais ADD CONSTRAINT pais_pk PRIMARY KEY ( id_pais );
```

```
CREATE TABLE persona (  
    id_persona      INTEGER NOT NULL,  
    id_telefono     INTEGER,  
    id_seguro       INTEGER,  
    id_tipo_sangre  INTEGER NOT NULL,  
    id_rol          INTEGER NOT NULL,  
    primer_nombre   VARCHAR2(30),  
    segundo_nombre  VARCHAR2(30),  
    primer_apellido VARCHAR2(30),  
    segundo_apellido VARCHAR2(30),  
    apellido_casada VARCHAR2(30),  
    fecha_nacimiento DATE,  
    genero          VARCHAR2(1)  
);
```

```
ALTER TABLE persona ADD CONSTRAINT persona_pk PRIMARY KEY ( id_persona );
```

```
CREATE TABLE personal_tecnico (  
    id_personal_t      INTEGER NOT NULL,  
    id_tipo_personal_tecnico INTEGER NOT NULL,  
    id_persona         INTEGER NOT NULL  
);
```

```
ALTER TABLE personal_tecnico ADD CONSTRAINT personal_tecnico_pk PRIMARY KEY (
id_personal_t );
```

```
CREATE TABLE producto_farmacia (
    id_farmacia    INTEGER NOT NULL,
    id_insumo      INTEGER NOT NULL,
    cantidad       NUMBER(4)
);
```

```
CREATE TABLE proveedores (
    id_proveedor          INTEGER NOT NULL,
    razon_social           VARCHAR2(200),
    nit                   VARCHAR2(50),
    representante_legal_id_persona  INTEGER,
    status_proveedor      INTEGER
);
```

```
ALTER TABLE proveedores ADD CONSTRAINT proveedores_pk PRIMARY KEY (
id_proveedor );
```

```
CREATE TABLE razon_social_persona (
    id_razon_social      INTEGER NOT NULL,
    id_persona           INTEGER NOT NULL,
    id_tipo_razon_social  VARCHAR2(100),
    nit                  VARCHAR2(25)
);
```

```
ALTER TABLE razon_social_persona ADD CONSTRAINT razon_social_persona_pk
PRIMARY KEY ( id_razon_social );
```

```
CREATE TABLE resultados_laboratorios (  
    id_examen            INTEGER NOT NULL,  
    id_laboratorista     INTEGER NOT NULL,  
    resultado            VARCHAR2(100)  
);
```

```
CREATE TABLE rol (  
    id_rol_usuario       INTEGER NOT NULL,  
    nombre               VARCHAR2(50),  
    descripcion_rol      VARCHAR2(100),  
    fecha_creacion       DATE,  
    usuario_creacion     VARCHAR2(25),  
    status               INTEGER  
);
```

```
ALTER TABLE rol ADD CONSTRAINT rol_usuario_pk PRIMARY KEY ( id_rol_usuario );
```

```
CREATE TABLE rol_aplicacion (  
    id_rol_aplicacion    INTEGER NOT NULL,  
    id_menu              INTEGER NOT NULL  
);
```

```
ALTER TABLE rol_aplicacion ADD CONSTRAINT rol_aplicacion_pk PRIMARY KEY (  
id_rol_aplicacion );
```

```
CREATE TABLE rol_persona (  
    id_rol               INTEGER NOT NULL,  
    rol                  VARCHAR2(50),  
    estado               VARCHAR2(25),  
    usuario_registro     VARCHAR2(25),
```

```
        fecha_registro    DATE
    );
```

```
ALTER TABLE rol_persona ADD CONSTRAINT rol_pk PRIMARY KEY ( id_rol );
```

```
CREATE TABLE rol_usuario (
    id_usuario            INTEGER NOT NULL,
    id_rol_usuario        INTEGER NOT NULL,
    id_rol_aplicacion     INTEGER NOT NULL,
    id_persona            INTEGER NOT NULL
);
```

```
ALTER TABLE rol_usuario ADD CONSTRAINT rol_usuario_new_pk PRIMARY KEY (
id_rol_usuario,id_usuario );
```

```
CREATE TABLE sede (
    id_sede              INTEGER NOT NULL,
    id_direccion         INTEGER,
    nombre               VARCHAR2(100)
);
```

```
ALTER TABLE sede ADD CONSTRAINT sede_pk PRIMARY KEY ( id_sede );
```

```
CREATE TABLE seguro (
    id_seguro            INTEGER NOT NULL,
    id_tipo_seguro       INTEGER NOT NULL,
    seguro               VARCHAR2(30),
    fecha_registro       DATE,
    vigencia_seguro      DATE
);
```

```
ALTER TABLE seguro ADD CONSTRAINT seguro_pk PRIMARY KEY ( id_seguro );
```

```
CREATE TABLE signos_vitales (  
    id_signos_vitales    INTEGER NOT NULL,  
    id_persona           INTEGER,  
    fecha_registro       DATE,  
    usuario_registro     VARCHAR2(25),  
    presion               NUMBER(5,2),  
    pulso                 NUMBER(5,2),  
    respiracion           NUMBER(5,2),  
    peso                  NUMBER(5,2),  
    altura                NUMBER(5,2),  
    temperatura           NUMBER(5,2)  
);
```

```
ALTER TABLE signos_vitales ADD CONSTRAINT signos_vitales_pk PRIMARY KEY (  
id_signos_vitales );
```

```
CREATE TABLE status (  
    id_status            INTEGER NOT NULL,  
    nombre_status        VARCHAR2(50)  
);
```

```
ALTER TABLE status ADD CONSTRAINT status_pk PRIMARY KEY ( id_status );
```

```
CREATE TABLE suspencion_medica (  
    id_suspencion        INTEGER NOT NULL,  
    id_tipo_suspencion    INTEGER NOT NULL,  
    id_persona            INTEGER NOT NULL,  
    suspencion            VARCHAR2(50),  
    usuario_registro      VARCHAR2(25),  
    fecha_registro        DATE,
```

```

        motivo                VARCHAR2(500)
    );

ALTER TABLE suspencion_medica ADD CONSTRAINT suspencion_medica_pk PRIMARY KEY
( id_suspencion );

CREATE TABLE suspencion_persona (
    id_suspencion    INTEGER NOT NULL,
    id_persona       INTEGER NOT NULL
);

ALTER TABLE suspencion_persona ADD CONSTRAINT suspencion_persona_pk PRIMARY
KEY ( id_suspencion );

CREATE TABLE telefono (
    id_telefono      INTEGER NOT NULL,
    id_tipo_telefono INTEGER,
    id_compania      INTEGER,
    telefono         NUMBER(15)
);

ALTER TABLE telefono ADD CONSTRAINT telefono_pk PRIMARY KEY ( id_telefono );

CREATE TABLE telefono_persona (
    id_telefono    INTEGER NOT NULL,
    id_persona     INTEGER NOT NULL
);

ALTER TABLE telefono_persona ADD CONSTRAINT telefono_persona_pk PRIMARY KEY (
id_telefono );

CREATE TABLE telefono_proveedores (
    id_telefono    INTEGER NOT NULL,

```

```
        id_proveedor    INTEGER
    );
```

```
ALTER TABLE telefono_proveedores ADD CONSTRAINT telefono_proveedores_pk
PRIMARY KEY ( id_telefono );
```

```
CREATE TABLE tipo_documento (
    id_tipo_documento    INTEGER NOT NULL,
    tipo_documento       VARCHAR2(30)
);
```

```
ALTER TABLE tipo_documento ADD CONSTRAINT tipo_documento_pk PRIMARY KEY (
id_tipo_documento );
```

```
CREATE TABLE tipo_documentos_pago (
    id_tipo_documento    INTEGER NOT NULL,
    documento            VARCHAR2(50)
);
```

```
ALTER TABLE tipo_documentos_pago ADD CONSTRAINT tipo_documentos_pk PRIMARY
KEY ( id_tipo_documento );
```

```
CREATE TABLE tipo_enfermedad (
    id_tipo_enfermedad    INTEGER NOT NULL,
    tipo_enfermedad       VARCHAR2(50),
    usuario_registro      VARCHAR2(25)
);
```

```
ALTER TABLE tipo_enfermedad ADD CONSTRAINT tipo_enfermedad_pk PRIMARY KEY (
id_tipo_enfermedad );
```

```
CREATE TABLE tipo_personal_tecnico (  
    id_tipo_personal_t    INTEGER NOT NULL,  
    descripcion            VARCHAR2(50)  
);
```

```
ALTER TABLE tipo_personal_tecnico ADD CONSTRAINT tipo_personal_tecnico_pk  
PRIMARY KEY ( id_tipo_personal_t );
```

```
CREATE TABLE tipo_razon_social (  
    id_tipo_razon_social    INTEGER NOT NULL,  
    tipo_razon_social       VARCHAR2(100)  
);
```

```
ALTER TABLE tipo_razon_social ADD CONSTRAINT tipo_razon_social_pk PRIMARY KEY  
( id_tipo_razon_social );
```

```
CREATE TABLE tipo_sangre (  
    id_tipo_sangre        INTEGER NOT NULL,  
    tipo_sangre           VARCHAR2(10),  
    usuario_registro      VARCHAR2(25)  
);
```

```
ALTER TABLE tipo_sangre ADD CONSTRAINT tipo_sangre_pk PRIMARY KEY (  
id_tipo_sangre );
```

```
CREATE TABLE tipo_seguro (  
    id_tipo_seguro        INTEGER NOT NULL,  
    tipo_seguro           VARCHAR2(30)  
);
```

```
ALTER TABLE tipo_seguro ADD CONSTRAINT tipo_seguro_pk PRIMARY KEY (  
id_tipo_seguro );
```



```
CREATE TABLE tipo_servicio (  
    id_tipo_servicio    INTEGER NOT NULL,  
    tipo_servicio       VARCHAR2(100),  
    precio_publico      NUMBER(5),  
    precio_costo        NUMBER(5)  
);  
  
ALTER TABLE tipo_servicio ADD CONSTRAINT tipo_servicio_pk PRIMARY KEY (  
id_tipo_servicio );
```

```
CREATE TABLE tipo_suspension (  
    id_tipo_suspension  INTEGER NOT NULL,  
    tipo_suspension     VARCHAR2(50)  
);
```

```
ALTER TABLE tipo_suspension ADD CONSTRAINT tipo_suspension_pk PRIMARY KEY (  
id_tipo_suspension );
```

```
CREATE TABLE tipo_telefono (  
    id_tipo_telefono    INTEGER NOT NULL,  
    tipo_telefono       VARCHAR2(50)  
);
```

```
ALTER TABLE tipo_telefono ADD CONSTRAINT tipo_telefono_pk PRIMARY KEY (  
id_tipo_telefono );
```

```
CREATE TABLE transacciones (  
    id_transaccion      INTEGER NOT NULL,  
    id_factura          INTEGER,  
    id_cliente          INTEGER,  
    id_caja             INTEGER,  
    id_pago             INTEGER,
```

```
        usuario_registro    VARCHAR2(25)
    );
```

```
ALTER TABLE transacciones ADD CONSTRAINT transacciones_pk PRIMARY KEY (
id_transaccion );
```

```
CREATE TABLE zonas (
    id_zona                INTEGER NOT NULL,
    id_municipio           INTEGER,
    zona                   NUMBER(2),
    fecha_registro         DATE,
    usuario_registro       VARCHAR2(25)
);
```

```
CREATE TABLE Tipo_Odontologia(
    id_Odontologia NUMBER(5) NOT NULL,
    descripcion varchar2(1000),
    PRIMARY KEY(id_Odontologia)
);
```

```
CREATE TABLE Tipo_Paciente_0(
    idTipoPaciente NUMBER(5) NOT NULL,
    idPaciente INTEGER NOT NULL,
    idTipoOdontologia NUMBER(5) NOT NULL,
    idDiagnostico INTEGER NOT NULL,
    observacion VARCHAR2(1000),
    fechaRegistro date,
    PRIMARY KEY(idTipoPaciente)
);
```

```
CREATE TABLE ALIMENTO (  
    id_alimento number(9) not null,  
    alimento varchar2(128) UNIQUE not null  
);
```

```
ALTER TABLE ALIMENTO ADD CONSTRAINT alimento_pk PRIMARY KEY ( id_alimento );
```

```
CREATE TABLE tiempo_comida (  
    id_tiempo_comida number(5) not null,  
    tiempo varchar2(128) UNIQUE not null  
);
```

```
ALTER TABLE tiempo_comida ADD CONSTRAINT tiempo_comida_pk PRIMARY KEY (  
id_tiempo_comida );
```

```
CREATE TABLE DIETA (  
    id_dieta    number(9) NOT NULL,  
    id_cita INTEGER NOT NULL,  
    id_medico INTEGER not null,  
    id_diagnostico INTEGER not null  
);
```

```
ALTER TABLE PACIENTE_ODONTOLOGIA ADD CONSTRAINT paciente_pk PRIMARY KEY (  
id_persona );
```

```
ALTER TABLE DIAGNOSTICO ADD CONSTRAINT diagnostico_pk PRIMARY KEY (  
id_diagnostico );
```

```
ALTER TABLE DIETA ADD CONSTRAINT dieta_pk PRIMARY KEY ( id_dieta );
```

```
ALTER TABLE DIETA ADD CONSTRAINT dieta_cita_fk FOREIGN KEY ( id_cita )  
REFERENCES cita ( id_cita );
```

```
ALTER TABLE DIETA ADD CONSTRAINT dieta_medico_fk FOREIGN KEY ( id_medico )  
REFERENCES medico ( id_persona );
```

```
ALTER TABLE DIETA ADD CONSTRAINT dieta_diagnostico_fk FOREIGN KEY ( id_diagnostico )  
REFERENCES DIAGNOSTICO ( id_diagnostico );
```

```
CREATE TABLE DETALLE_DIETA (  
    id_detalle_dieta number(9) not null,  
    id_dieta number(9) not null,  
    id_alimento      number(9) not null,  
    cantidad          varchar2(100) not null,  
    id_tiempo_comida number(5) not null,  
    descripcion varchar2(1024)  
);
```

```
ALTER TABLE DETALLE_DIETA ADD CONSTRAINT detalle_dieta_pk PRIMARY KEY (  
id_detalle_dieta );
```

```
ALTER TABLE DETALLE_DIETA ADD CONSTRAINT detalle_dieta_fk FOREIGN KEY (  
id_dieta ) REFERENCES DIETA ( id_dieta );
```

```
ALTER TABLE DETALLE_DIETA ADD CONSTRAINT detalle_alimento_fk FOREIGN KEY (  
id_alimento ) REFERENCES alimento ( id_alimento );
```

```
ALTER TABLE DETALLE_DIETA ADD CONSTRAINT detalle_tiempo_fk FOREIGN KEY (  
id_tiempo_comida ) REFERENCES tiempo_comida ( id_tiempo_comida );
```

```
COMMENT ON COLUMN DETALLE_DIETA.cantidad IS 'Cantidad de alimento, gramos,  
cucharadas, vasos, porciones, etc.';
```

```
COMMENT ON COLUMN DETALLE_DIETA.descripcion IS 'Descripci n de como cocinar  
o consumir el alimento.';
```

```
ALTER TABLE TIPO_PACIENTE_O ADD CONSTRAINT Paciente_fk  
    FOREIGN KEY(idPaciente)  
    REFERENCES PACIENTE_ODONTOLOGIA(id_persona);
```

```
ALTER TABLE TIPO_PACIENTE_O ADD CONSTRAINT tipo_fk  
    FOREIGN KEY(idTipoOdontologia)  
    REFERENCES Tipo_Odontologia(id_Odontologia);
```

```
ALTER TABLE TIPO_PACIENTE_O ADD CONSTRAINT diagnostico_fk  
    FOREIGN KEY(idDiagnostico)  
    REFERENCES DIAGNOSTICO(ID_DIAGNOSTICO);
```

--Alter a tablas direccion, municipio, departamento, pais, colonia, direccion
persona--

```
ALTER TABLE zonas ADD CONSTRAINT zonas_pk PRIMARY KEY ( id_zona );
```

```
ALTER TABLE colonia  
    ADD CONSTRAINT colonia_municipios_fk FOREIGN KEY ( id_municipio )  
    REFERENCES municipios ( id_municipio );
```

```
ALTER TABLE departamento  
    ADD CONSTRAINT departamento_pais_fk FOREIGN KEY ( id_pais )  
    REFERENCES pais ( id_pais );
```

```
ALTER TABLE direccion
```

```
    ADD CONSTRAINT direccion_municipios_fk FOREIGN KEY ( id_municipio )  
    REFERENCES municipios ( id_municipio );
```

```
ALTER TABLE direccion
```

```
    ADD CONSTRAINT direccion_zonas_fk FOREIGN KEY ( id_zona )  
    REFERENCES zonas ( id_zona );
```

```
ALTER TABLE direccion
```

```
    ADD CONSTRAINT direccion_colonia_fk FOREIGN KEY ( id_colonia )  
    REFERENCES colonia ( id_colonia );
```

```
ALTER TABLE direccion_persona
```

```
    ADD CONSTRAINT direccion_persona_direccion_fk FOREIGN KEY ( id_direccion  
)  
    REFERENCES direccion ( id_direccion );
```

```
ALTER TABLE direccion_persona
```

```
    ADD CONSTRAINT direccion_persona_persona_fk FOREIGN KEY ( id_persona )  
    REFERENCES persona ( id_persona );
```

```
ALTER TABLE municipios
```

```
    ADD CONSTRAINT municipio_departamento_fk FOREIGN KEY ( id_departamento )  
    REFERENCES departamento ( id_departamento );
```

```
ALTER TABLE documento_persona
```

```
    ADD CONSTRAINT documento_persona_persona_fk FOREIGN KEY ( id_persona )  
    REFERENCES persona ( id_persona );
```

```
ALTER TABLE cita
```

```
    ADD CONSTRAINT cita_clinicas_fk FOREIGN KEY ( id_clinica )  
        REFERENCES clinicas ( id_clinica );
```

```
ALTER TABLE cita
```

```
    ADD CONSTRAINT cita_persona_fk FOREIGN KEY ( id_persona_paciente )  
        REFERENCES persona ( id_persona );
```

```
ALTER TABLE cita
```

```
    ADD CONSTRAINT cita_seguro_fk FOREIGN KEY ( id_seguro )  
        REFERENCES seguro ( id_seguro );
```

```
ALTER TABLE cliente
```

```
    ADD CONSTRAINT cliente_persona_fk FOREIGN KEY ( id_persona )  
        REFERENCES persona ( id_persona );
```

```
ALTER TABLE cliente
```

```
    ADD CONSTRAINT cliente_rsoci alp FOREIGN KEY ( id_razon_social_p )  
        REFERENCES razon_social_persona ( id_razon_social );
```

```
ALTER TABLE clinicas
```

```
    ADD CONSTRAINT clinicas_sede_fk FOREIGN KEY ( id_sede )  
        REFERENCES sede ( id_sede );
```

```
ALTER TABLE compania_telefono
```

```
    ADD CONSTRAINT compania_telefono_telefono_fk FOREIGN KEY ( id_compania )  
        REFERENCES telefono ( id_telefono );
```

```
ALTER TABLE cuenta_proveedores
    ADD CONSTRAINT cuenta_proveedores_bancos_fk FOREIGN KEY ( id_banco_cuenta
)
    REFERENCES bancos ( id_banco );
```

```
ALTER TABLE cuentas_por_pagar
    ADD CONSTRAINT cuentas_por_pagar_status_fk FOREIGN KEY ( id_status_pago )
    REFERENCES status ( id_status );
```

```
ALTER TABLE cuenta_proveedores
    ADD CONSTRAINT cuentayproveedores FOREIGN KEY ( id_proveedor )
    REFERENCES proveedores ( id_proveedor );
```

```
ALTER TABLE detalle_compra
    ADD CONSTRAINT detalle_compra_compras_fk FOREIGN KEY ( id_compra )
    REFERENCES compras ( id_compra );
```

```
ALTER TABLE detalle_factura
    ADD CONSTRAINT detalle_factura_facturacion_fk FOREIGN KEY ( id_factura )
    REFERENCES facturacion ( id_factura );
```

```
ALTER TABLE detalle_pago
    ADD CONSTRAINT detalle_pago_compras_fk FOREIGN KEY ( id_compra )
    REFERENCES compras ( id_compra );
```

```
ALTER TABLE diagnostico
    ADD CONSTRAINT diagnostico_enfermedad_fk FOREIGN KEY ( id_enfermedad )
    REFERENCES enfermedad ( id_enfermedad );
```



```
ALTER TABLE diagnostico
```

```
    ADD CONSTRAINT diagnostico_persona_fk FOREIGN KEY ( id_persona )  
    REFERENCES persona ( id_persona );
```

```
ALTER TABLE diagnostico
```

```
    ADD CONSTRAINT diagnosticoehistclinc FOREIGN KEY ( id_historial_clinico )  
    REFERENCES historial_clinico ( id_historial_clinico );
```

```
ALTER TABLE direccion_proveedores
```

```
    ADD CONSTRAINT direcc_provee FOREIGN KEY ( id_direccion_prov )  
    REFERENCES direccion ( id_direccion );
```

```
ALTER TABLE encargado_sede
```

```
    ADD CONSTRAINT encargado_sede_persona_fk FOREIGN KEY ( id_persona )  
    REFERENCES persona ( id_persona );
```

```
ALTER TABLE encargado_sede
```

```
    ADD CONSTRAINT encargado_sede_sede_fk FOREIGN KEY ( id_sede )  
    REFERENCES sede ( id_sede );
```

```
ALTER TABLE enfermedad
```

```
    ADD CONSTRAINT enfermedad_tipo_enfermedad_fk FOREIGN KEY ( id_tipo_enfermedad )  
    REFERENCES tipo_enfermedad ( id_tipo_enfermedad );
```

```
ALTER TABLE especialidad_medico
```

```
    ADD CONSTRAINT especial_medico FOREIGN KEY ( id_medico )  
    REFERENCES medico ( id_persona );
```

```
ALTER TABLE especialidad_medico
    ADD CONSTRAINT especialidad_especiamd FOREIGN KEY ( id_especialidad )
        REFERENCES especialidad ( id_especialidad );
```

```
ALTER TABLE examen_persona
    ADD CONSTRAINT examen_persona_exámenes_fk FOREIGN KEY ( id_examen )
        REFERENCES exámenes ( id_examen );
```

```
ALTER TABLE examen_persona
    ADD CONSTRAINT examen_persona_persona_fk FOREIGN KEY ( id_persona )
        REFERENCES persona ( id_persona );
```

```
ALTER TABLE producto_farmacia
    ADD CONSTRAINT farmacia_farmacias_fk FOREIGN KEY ( id_farmacia )
        REFERENCES farmacias ( id_farmacia );
```

```
ALTER TABLE producto_farmacia
    ADD CONSTRAINT farmacia_insumo_fk FOREIGN KEY ( id_insumo )
        REFERENCES insumo ( id_insumo );
```

```
ALTER TABLE farmacias
    ADD CONSTRAINT farmacias_sede_fk FOREIGN KEY ( id_sede )
        REFERENCES sede ( id_sede );
```

```
ALTER TABLE historial_clinico
    ADD CONSTRAINT historial_clinico_persona_fk FOREIGN KEY ( id_persona )
        REFERENCES persona ( id_persona );
```

```
ALTER TABLE movimientos_historial
```

```
    ADD CONSTRAINT historial_enfer FOREIGN KEY ( id_enfermedad )  
        REFERENCES enfermedad ( id_enfermedad );
```

```
ALTER TABLE historial_clinico
```

```
    ADD CONSTRAINT historial_signos FOREIGN KEY ( id_signos_vitales )  
        REFERENCES signos_vitales ( id_signos_vitales );
```

```
ALTER TABLE tipo_razon_social
```

```
    ADD CONSTRAINT id_razonsocial FOREIGN KEY ( id_tipo_razon_social )  
        REFERENCES razon_social_persona ( id_razon_social );
```

```
ALTER TABLE rol_usuario
```

```
    ADD CONSTRAINT id_rolaplicacion_usuario FOREIGN KEY ( id_rol_aplicacion )  
        REFERENCES rol_aplicacion ( id_rol_aplicacion );
```

```
ALTER TABLE insumo_clinicas
```

```
    ADD CONSTRAINT insumo_clinicas_clinicas_fk FOREIGN KEY ( id_clinica )  
        REFERENCES clinicas ( id_clinica );
```

```
ALTER TABLE insumo_clinicas
```

```
    ADD CONSTRAINT insumo_clinicas_insumo_fk FOREIGN KEY ( id_insumo )  
        REFERENCES insumo ( id_insumo );
```

```
ALTER TABLE insumo_laboratorios
```

```
    ADD CONSTRAINT insumo_laboratorios_insumo_fk FOREIGN KEY ( id_insumo )  
        REFERENCES insumo ( id_insumo );
```

```
ALTER TABLE insumo
```

```
    ADD CONSTRAINT insumo_laboratorios_insumos_fk FOREIGN KEY (
id_laboratorio )
        REFERENCES lab_farmaceuticos_insumos ( id_laboratorio );
```

```
ALTER TABLE resultados_laboratorios
```

```
    ADD CONSTRAINT laboratorio_examen FOREIGN KEY ( id_examen )
        REFERENCES examenes ( id_examen );
```

```
ALTER TABLE lotes
```

```
    ADD CONSTRAINT lotes_compras_fk FOREIGN KEY ( id_lote )
        REFERENCES compras ( id_compra );
```

```
ALTER TABLE lotes
```

```
    ADD CONSTRAINT lotes_insumo_clinicas_fk FOREIGN KEY (
insumo_clinicas_id_insumo, insumo_clinicas_id_lote )
        REFERENCES insumo_clinicas ( id_insumo,
                                     id_lote );
```

```
ALTER TABLE medico
```

```
    ADD CONSTRAINT medico_persona_fk FOREIGN KEY ( id_persona )
        REFERENCES persona ( id_persona );
```

```
ALTER TABLE movimientos_historial
```

```
    ADD CONSTRAINT movimiento_examenes FOREIGN KEY ( id_examen )
        REFERENCES examenes ( id_examen );
```

```
ALTER TABLE paciente_odontologia
```

```
    ADD CONSTRAINT odonto_clinicas FOREIGN KEY ( id_clinica )
        REFERENCES clinicas ( id_clinica );
```

```
ALTER TABLE paciente_odontologia
```

```
    ADD CONSTRAINT odonto_persona FOREIGN KEY ( id_persona )  
    REFERENCES persona ( id_persona );
```

```
ALTER TABLE pago_proveedores
```

```
    ADD CONSTRAINT pago_detalle_pago_fk FOREIGN KEY ( id_pago )  
    REFERENCES detalle_pago ( id_detalle_pago );
```

```
ALTER TABLE pago_proveedores
```

```
    ADD CONSTRAINT pago_proveedores_fk FOREIGN KEY ( id_proveedor )  
    REFERENCES proveedores ( id_proveedor );
```

```
ALTER TABLE persona
```

```
    ADD CONSTRAINT persona_rol_fk FOREIGN KEY ( id_rol )  
    REFERENCES rol_persona ( id_rol );
```

```
ALTER TABLE persona
```

```
    ADD CONSTRAINT persona_tipo_sangre_fk FOREIGN KEY ( id_tipo_sangre )  
    REFERENCES tipo_sangre ( id_tipo_sangre );
```

```
ALTER TABLE personal_tecnico
```

```
    ADD CONSTRAINT personal_tecnico FOREIGN KEY ( id_tipo_personal_tecnico )  
    REFERENCES tipo_personal_tecnico ( id_tipo_personal_t );
```

```
ALTER TABLE personal_tecnico
```

```
    ADD CONSTRAINT personal_tecnico_persona_fk FOREIGN KEY ( id_persona )  
    REFERENCES persona ( id_persona );
```

```
ALTER TABLE cuentas_por_pagar
```

```
    ADD CONSTRAINT porpagaryprovee FOREIGN KEY ( id_proveedor )  
    REFERENCES proveedores ( id_proveedor );
```

```
ALTER TABLE proveedores
```

```
    ADD CONSTRAINT prov_direccion FOREIGN KEY ( id_proveedor )  
    REFERENCES direccion_proveedores ( id_direccion_prov );
```

```
ALTER TABLE proveedores
```

```
    ADD CONSTRAINT prov_telefono FOREIGN KEY ( id_proveedor )  
    REFERENCES telefono_proveedores ( id_telefono );
```

```
ALTER TABLE razon_social_persona
```

```
    ADD CONSTRAINT razonsocial_persona FOREIGN KEY ( id_persona )  
    REFERENCES persona ( id_persona );
```

```
ALTER TABLE cuenta_proveedores
```

```
    ADD CONSTRAINT relacion_cuentas FOREIGN KEY ( id_cuenta_prov )  
    REFERENCES cuentas_por_pagar ( id_cuenta );
```

```
ALTER TABLE resultados_laboratorios
```

```
    ADD CONSTRAINT resultado_personaltecnico FOREIGN KEY ( id_laboratorista )  
    REFERENCES personal_tecnico ( id_personal_t );
```

```
ALTER TABLE rol_aplicacion
```

```
    ADD CONSTRAINT rol_aplicacion_menu_fk FOREIGN KEY ( id_menu )  
    REFERENCES menu ( id_menu );
```

```
ALTER TABLE rol_usuario
```

```
    ADD CONSTRAINT rol_usuario_persona_fk FOREIGN KEY ( id_persona )  
    REFERENCES persona ( id_persona );
```

```
ALTER TABLE rol_usuario
```

```
    ADD CONSTRAINT rol_usuario_rol_fk FOREIGN KEY ( id_rol_usuario )  
    REFERENCES rol ( id_rol_usuario );
```

```
ALTER TABLE seguro
```

```
    ADD CONSTRAINT seguro_tipo_seguro_fk FOREIGN KEY ( id_tipo_seguro )  
    REFERENCES tipo_seguro ( id_tipo_seguro );
```

```
ALTER TABLE suspencion_medica
```

```
    ADD CONSTRAINT suspen_tiposuspen FOREIGN KEY ( id_tipo_suspencion )  
    REFERENCES tipo_suspencion ( id_tipo_suspencion );
```

```
ALTER TABLE suspencion_medica
```

```
    ADD CONSTRAINT suspencion_medica_persona_fk FOREIGN KEY ( id_persona )  
    REFERENCES persona ( id_persona );
```

```
ALTER TABLE suspencion_persona
```

```
    ADD CONSTRAINT suspencion_persona_persona_fk FOREIGN KEY ( id_persona )  
    REFERENCES persona ( id_persona );
```

```
ALTER TABLE telefono_persona
```

```
    ADD CONSTRAINT telefono_persona_persona_fk FOREIGN KEY ( id_persona )  
    REFERENCES persona ( id_persona );
```

```
ALTER TABLE telefono
```

```
    ADD CONSTRAINT telefono_telefono_persona_fk FOREIGN KEY ( id_telefono )  
        REFERENCES telefono_persona ( id_telefono );
```

```
ALTER TABLE telefono_proveedores
```

```
    ADD CONSTRAINT telprov_tel FOREIGN KEY ( id_telefono )  
        REFERENCES telefono ( id_telefono );
```

```
ALTER TABLE tipo_telefono
```

```
    ADD CONSTRAINT tipo_telefono_telefono_fk FOREIGN KEY ( id_tipo_telefono )  
        REFERENCES telefono ( id_telefono );
```

```
ALTER TABLE tipo_documento
```

```
    ADD CONSTRAINT tipodoc_docpersona FOREIGN KEY ( id_tipo_documento )  
        REFERENCES documento_persona ( id_tipo_documento );
```

```
ALTER TABLE tipo_documentos_pago
```

```
    ADD CONSTRAINT tipodocpag_detpago FOREIGN KEY ( id_tipo_documento )  
        REFERENCES detalle_pago ( id_detalle_pago );
```

```
alter table CREDITO_CILIENTE
```

```
    add constraint FK_CREDITO__REFERENCE_INTERES foreign key (ID_INTERES)  
        references INTERES (ID_INTERES);
```

```
alter table CREDITO_CILIENTE
```

```
    add constraint FK_CREDITO__REFERENCE_CLIENTE foreign key (ID_CLIENTE)  
        references CLIENTE (ID_CLIENTE);
```

```
alter table DETALLE_FACTURA
```

```
    add constraint FK_DETALLE__REFERENCE_PRODUCTO foreign key (ID_FACTURA)  
        references PRODUCTOS_SERVICIOS (ID_PRO_SER);
```



```
alter table DETALLE_PAGO_CLIENTE
    add constraint FK_DETALLE__REFERENCE_TIPO_PAG foreign key (ID_TIPO_PAGO)
        references TIPO_PAGO (ID_TIPO_PAGO);
```

```
alter table FACTURACION
    add constraint FK_FACTURAC_REFERENCE_ESTADO foreign key (ID_ESTADO)
        references ESTADO (ID_ESTADO);
```

```
alter table MOVIMIENTOS
    add constraint FK_MOVIMIEN_REFERENCE_CREDITO_ foreign key (ID_CREDITO)
        references CREDITO_CILIENTE (ID_CREDITO);
```

```
alter table MOVIMIENTOS
    add constraint FK_MOVIMIEN_REFERENCE_TIPO_MOV foreign key (ID_TIPO_MOV)
        references TIPO_MOV (ID_TIPO_MOV);
```

```
alter table PRODUCTOS_SERVICIOS
    add constraint FK_PRODUCTO_REFERENCE_CATEGORI foreign key (ID_CATEGORIA)
        references CATEGORIA (ID_CATEGORIA);
```

--CREACION DE PAQUETES POR MODULOS

-- PAQUETE ROLES

```
CREATE OR REPLACE PACKAGE paquete_rol AS
    PROCEDURE agregar_rol(Cnombre varchar2, Cdescripcion varchar2, Cusuario
        varchar2, Cactivo integer);
END paquete_rol;
```

```
CREATE OR REPLACE PACKAGE BODY paquete_rol AS

    PROCEDURE agregar_rol(Cnombre varchar2, Cdescripcion varchar2, Cusuario
varchar2, Cactivo integer) IS

        BEGIN

            INSERT INTO
ROL(id_rol_usuario,nombre,descripcion_rol,fecha_creacion,usuario_creacion,sta
tus)

                VALUES((SELECT max(id_rol_usuario)+1 FROM
ROL),Cnombre,Cdescripcion,SYSDATE,Cusuario,Cactivo);

        END agregar_rol;

END paquete_rol;
```