

Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ciencias y Sistemas
Laboratorio Sistemas de Bases de Datos 1
Ing. Luis Espino
Aux. Gerardo Chay
Primer Semestre 2022

Documentación

Proyecto 1

Datos:
Luis Dannel Ernesto Castellanos Galindo
201902238

Modelo relacional propuesto para la solución:

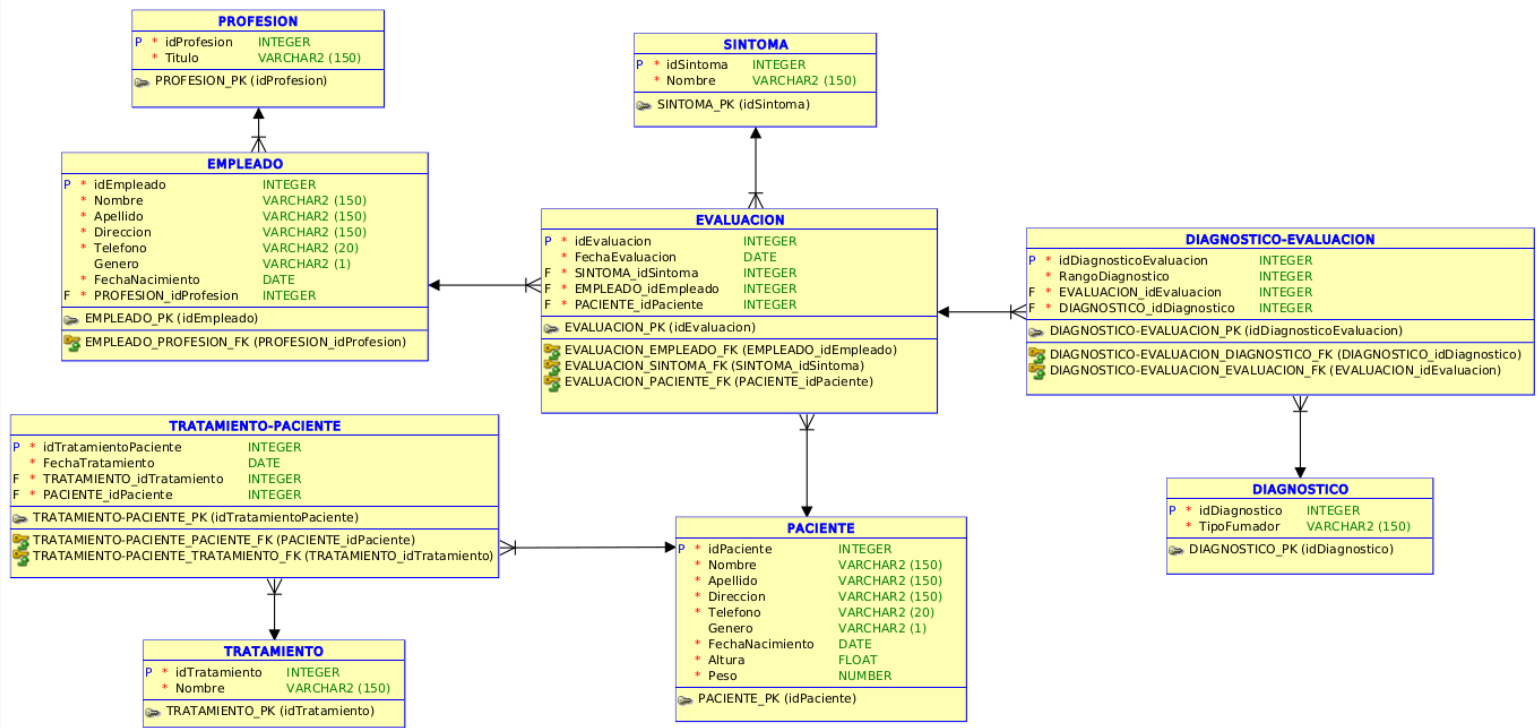


Tabla Temporal:

```
CREATE TABLE TEMPORAL (
  -- EMPLEADO
  NOMBRE_EMPLEADO VARCHAR2(150),
  APELLIDO_EMPLEADO VARCHAR2(150),
  DIRECCION_EMPLEADO VARCHAR2(150),
  TELEFONO_EMPLEADO VARCHAR2(20),
  GENERO_EMPLEADO VARCHAR2(1),
  FECHA_NACIMIENTO_EMPLEADO DATE,
  TITULO_DEL_EMPLEADO VARCHAR2(150),
  -- PACIENTE
  NOMBRE_PACIENTE VARCHAR2(150),
  APELLIDO_PACIENTE VARCHAR2(150),
  DIRECCION_PACIENTE VARCHAR2(150),
  TELEFONO_PACIENTE VARCHAR2(20),
  GENERO_PACIENTE VARCHAR2(1),
  FECHA_NACIMIENTO_PACIENTE DATE,
  ALTURA FLOAT,
  PESO NUMBER,
  FECHA_EVALUACION DATE,
  SINTOMA_DEL_PACIENTE VARCHAR2(150),
  DIAGNOSTICO_DEL_SINTOMA VARCHAR2(150),
  RANGO_DEL_DIAGNOSTICO INTEGER,
  FECHA_TRATAMIENTO DATE,
  TRATAMIENTO_APLICADO VARCHAR2(150)
);
```

Tabla Diagnóstico:

Se encarga de guardar solamente los distintos tipos de fumador y genera su identificador numérico.

```
CREATE TABLE DIAGNOSTICO (  
    idDiagnostico NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,  
    tipoFumador VARCHAR2(150) NOT NULL  
);
```

Tabla Diagnóstico-Evaluación:

Sirve para relacionar cada una de las evaluaciones y sus diagnósticos, guardando su rango y un identificador.

```
CREATE TABLE "DIAGNOSTICO-EVALUACION" (  
    idDiagnosticoEvaluacion NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,  
    rangoDiagnostico INTEGER NOT NULL,  
    evaluacion_idEvaluacion INTEGER NOT NULL,  
    diagnostico_idDiagnostico INTEGER NOT NULL  
);
```

Tabla Empleado:

Almacena cada uno de los empleados existentes en la base de datos, con sus datos personales e identificador.

```
CREATE TABLE EMPLEADO (  
    idEmpleado NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,  
    nombre VARCHAR2(150) NOT NULL,  
    apellido VARCHAR2(150) NOT NULL,  
    direccion VARCHAR2(150) NOT NULL,  
    telefono VARCHAR2(20) NOT NULL,  
    genero VARCHAR2(1),  
    fechaNacimiento DATE NOT NULL,  
    profesion_idProfesion INTEGER NOT NULL  
);
```

Tabla Evaluación:

Registra cada una de las fechas en que se realizó una evaluación, guardando el identificador del paciente y del empleado que lo atendió.

```
CREATE TABLE EVALUACION (  
    idEvaluacion    NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,  
    fechaEvaluacion DATE NOT NULL,  
    sintoma_idSintoma INTEGER,  
    empleado_idEmpleado INTEGER NOT NULL,  
    paciente_idPaciente INTEGER NOT NULL  
);
```

Tabla Paciente:

Almacena cada uno de los pacientes existentes en la base de datos, con sus datos personales e identificador.

```
CREATE TABLE PACIENTE (  
    idPaciente    NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,  
    nombre        VARCHAR2(150) NOT NULL,  
    apellido      VARCHAR2(150) NOT NULL,  
    direccion     VARCHAR2(150) NOT NULL,  
    telefono      VARCHAR2(20) NOT NULL,  
    genero        VARCHAR2(1),  
    fechaNacimiento DATE NOT NULL,  
    altura        FLOAT NOT NULL,  
    peso          NUMBER NOT NULL  
);
```

Tabla Profesión:

Guarda cada uno de los distintos títulos universitarios existentes que los empleados han obtenido.

```
CREATE TABLE PROFESION (  
    idProfesion    NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,  
    titulo         VARCHAR2(150) NOT NULL  
);
```

Tabla Síntoma:

Guarda cada uno de los diferentes síntomas posibles a los que se les puede detectar a los pacientes, con su identificador único.

```
CREATE TABLE SINTOMA (  
    idSintoma NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,  
    nombre VARCHAR2(150) NOT NULL  
);
```

Tabla tratamiento:

Guarda cada una de las fechas en que se le administró un tratamiento a un paciente.

```
CREATE TABLE "TRATAMIENTO-PACIENTE" (  
    idTratamientoPaciente NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,  
    fechaTratamiento DATE NOT NULL,  
    tratamiento_idTratamiento INTEGER NOT NULL,  
    paciente_idPaciente INTEGER NOT NULL  
);
```

Restricciones del modelo:

Para todas y cada una de las tablas se siguieron ciertas normas, tales como:

- ➔ Cada tabla genera un identificador numérico ascendente, no se puede repetir e inicia desde 1, funcionan como las llaves primarias de cada tabla.
- ➔ La mayoría de atributos son de carácter obligatorio por lo que no se aceptan nulos para asegurar la integridad y consistencia de los datos.
- ➔ Para los campos largos de texto, se consideró un máximo de 150 caracteres y para los teléfonos un máximo de 20 caracteres.
- ➔ Las relaciones entre las tablas se colocan mediante sus identificadores (primary key).
- ➔ Las vistas creadas fueron utilizadas para ahorrar espacio de consultas y evitar instrucciones repetitivas.