Universidad Tecnológica de Panamá

Facultad de Ingeniería de Sistemas Computacionales

Licenciatura en Ingeniería de Sistemas de Información

Departamento de Sistemas de Información, Control y Evaluación de Recursos Informáticos

Sistemas de Bases de Datos II

Laboratorio No.5 Implementación de un modelo lógico relacional transaccional.

Facilitador:

Ing. Henry Lezcano

Integrantes:

David Cedeño 8-951-990

Simon Billeke E-8-154322

Gabriel Grimaldo 8-949-56

Daniel Rodriguez 3-740-2242

Grupo:

1IF131

II Semestre, 2020

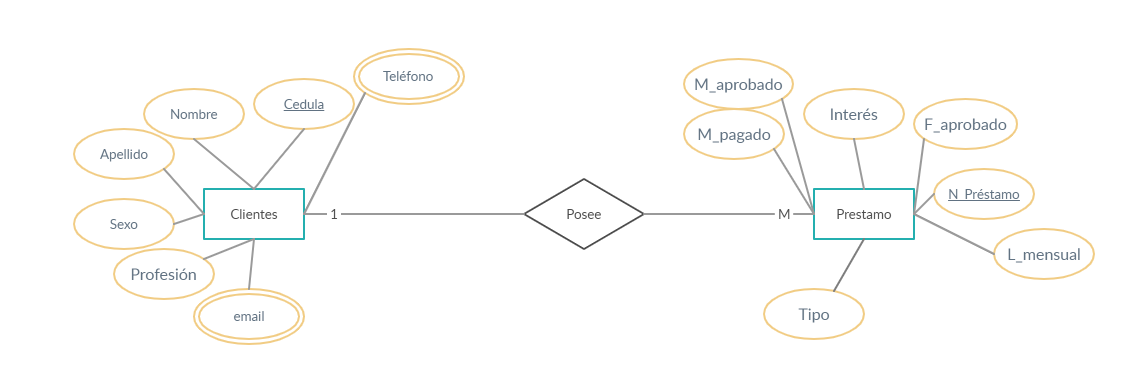
1. La Compañía Financiera Márquez-Cedeño, S.A., necesita levantar un modelo físico de datos a partir de un modelo lógico relacional para una sección de su proceso de negocio (otorgamiento de préstamos), en lo corresponde a la entrada de la información de los clientes y los préstamos e información relacionada a esta sección del proceso a dicha base de datos. Cuenta con unas relaciones o tablas de bases de datos que han sido identificadas y no están normalizadas que pueden ser utilizada por los Ingenieros de Sistemas de Información para futuras implementación. La relación Cliente está compuesta por identificación del cliente, cedula, nombre, apellido, sexo, email (personal, laboral, académico), teléfono (celular, residencia, celular del familiar más cercano, celular del conyugue), profesión (cualquier profesión del cliente), fecha de nacimiento. La relación Préstamo está compuesta por identificación del cliente, tipo de préstamo (personal, auto, hipoteca, garantizado con ahorros), número de préstamos, fecha de aprobado, monto aprobado, tasa de interés, letra mensual, monto pagado, fecha de pago. Los clientes pueden tener varios tipos de préstamos en la institución financiera.

• Muestre el modelo Lógico relacional normalizado

• Muestre el Script (DDL) que corresponde al levantamiento del modelo físico.

• Levante el Modelo Físico a partir del modelo Lógico relacional normalizado usando su script creado cumpliendo con o Las restricciones inherentes al modelo o Las restricciones semánticas o del usuario Se pide una vez el esquema de Base de Datos este levantado en el sistema de gestión, la información de la Base de Datos también quede levantada usando un script (DML) para su evaluación.

**Modelo Entidad relación**



**Modelo lógico normalizado**

**Modelo Físico**

Create Table Profesion (

IDProfesion number not null,

Descrpcion varchar(100) not null,

constraint pk\_Profesion\_ID primary key (IDProfesion));

Create Table Clientes(

IDCliente number not null,

Profesion number not null,

cedula varchar(20) not null,

Nombre varchar(30) not null,

Apellido varchar(30) not null,

Sexo varchar(10) not null,

constraint pk\_Clientes\_profesion foreign key (Profesion) References Profesion(IDProfesion),

constraint pk\_Clientes\_ primary key(IDCliente));

Create table Tipo\_Email(

IDTipo number not null,

Descripcion varchar(60) not null,

constraint pk\_Tipo\_email\_ID primary key (IDTipo));

Create table Clientes\_Email(

IDCliente number not null,

IDTipo number not null,

constraint pk\_Clientes\_email\_ID primary key (IDCliente,IDTipo),

constraint fk\_Clientes\_email\_ID\_C foreign key (IDCliente) References Clientes(IDCliente),

constraint fk\_Clientes\_email\_ID\_T foreign key (IDTipo) References Tipo\_Email(IDTipo));

Create table Tipo\_Telefono(

IDTipo number not null,

Descripcion varchar(60) not null,

constraint pk\_Tipo\_Telefono\_ID primary key (IDTipo));

Create table Clientes\_Telefono(

IDCliente number not null,

IDTipo number not null,

constraint pk\_Clientes\_Telefono\_ID primary key (IDCliente,IDTipo),

constraint fk\_Clientes\_Telefono\_ID\_C foreign key (IDCliente) References Clientes(IDCliente),

constraint fk\_Clientes\_Telefono\_ID\_T foreign key (IDTipo) References Tipo\_Telefono(IDTipo));

Create Table Tipo\_Prestamos(

ID\_Tipo varchar(5) not null,

Nombre varchar (20) not null,

Interes number not null,

constraint pk\_Tipo\_Prestamos\_ID\_Tipo primary key (ID\_Tipo));

Create Table Prestamos(

ID\_Cliente number not null,

ID\_Tipo varchar(5) not null,

N\_prestamo number not null,

F\_aprovado date not null,

M\_aprobado number not null,

L\_mensual number not null,

M\_pagado number not null,

F\_pago date not null,

constraint pk\_Prestamos primary key(ID\_Cliente,ID\_Tipo),

constraint fk\_Prestamos\_ID\_cli foreign key (ID\_Cliente) references Clientes (IDCliente),

constraint fk\_Prestamos\_ID\_tipo foreign key (ID\_Tipo) references Tipo\_Prestamos(ID\_Tipo));

Create unique index num\_pres on Prestamos (N\_prestamo);

