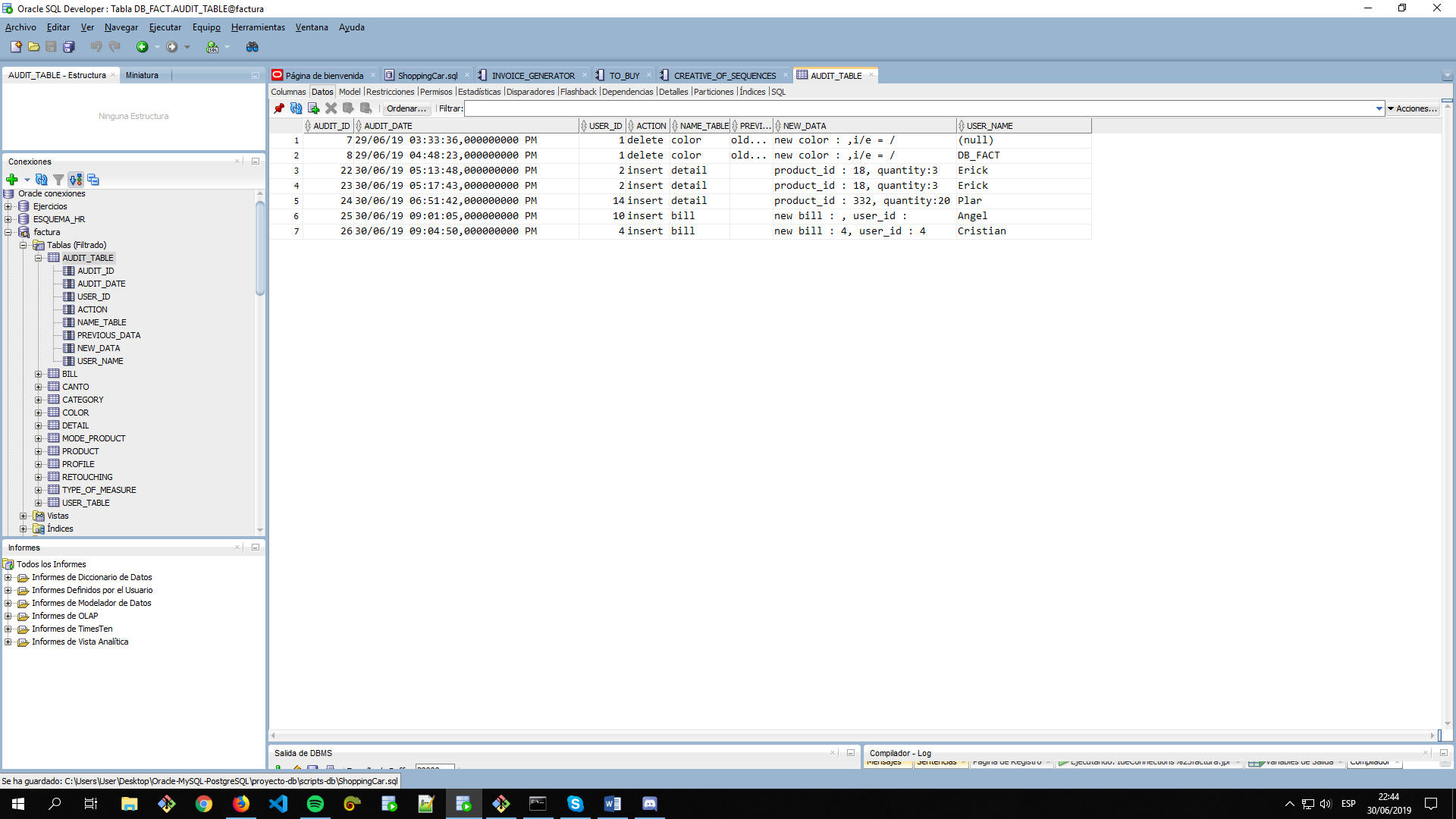


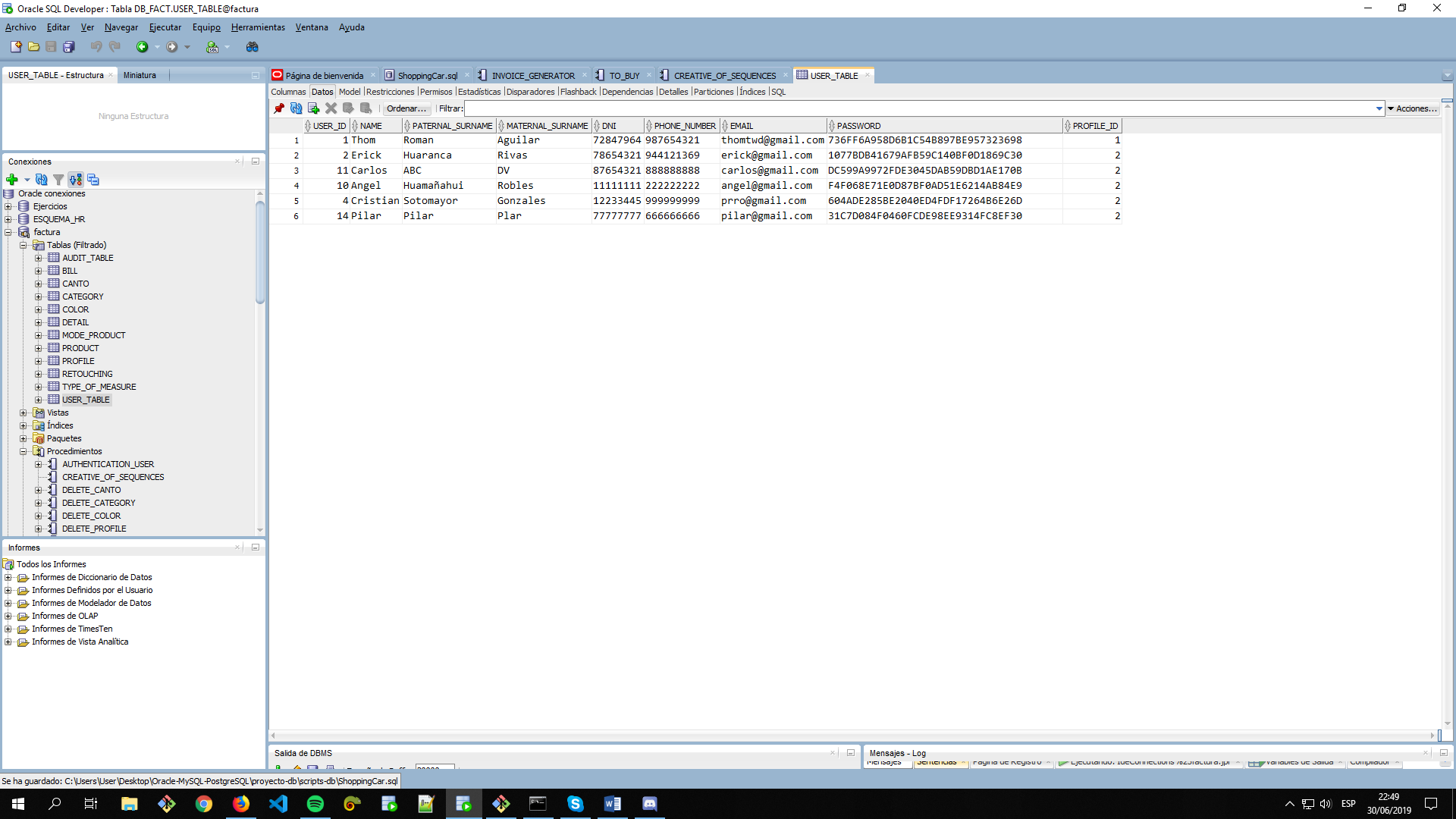
**Curso: Diseño de Base de Datos**

**Profesor: Lévano Rodríguez Daniel**

**Proyecto: “Simulador” de productos ONLINE**

TABLA DE CONTENIDO

Tabla de Auditoria

Tabla de usuarios

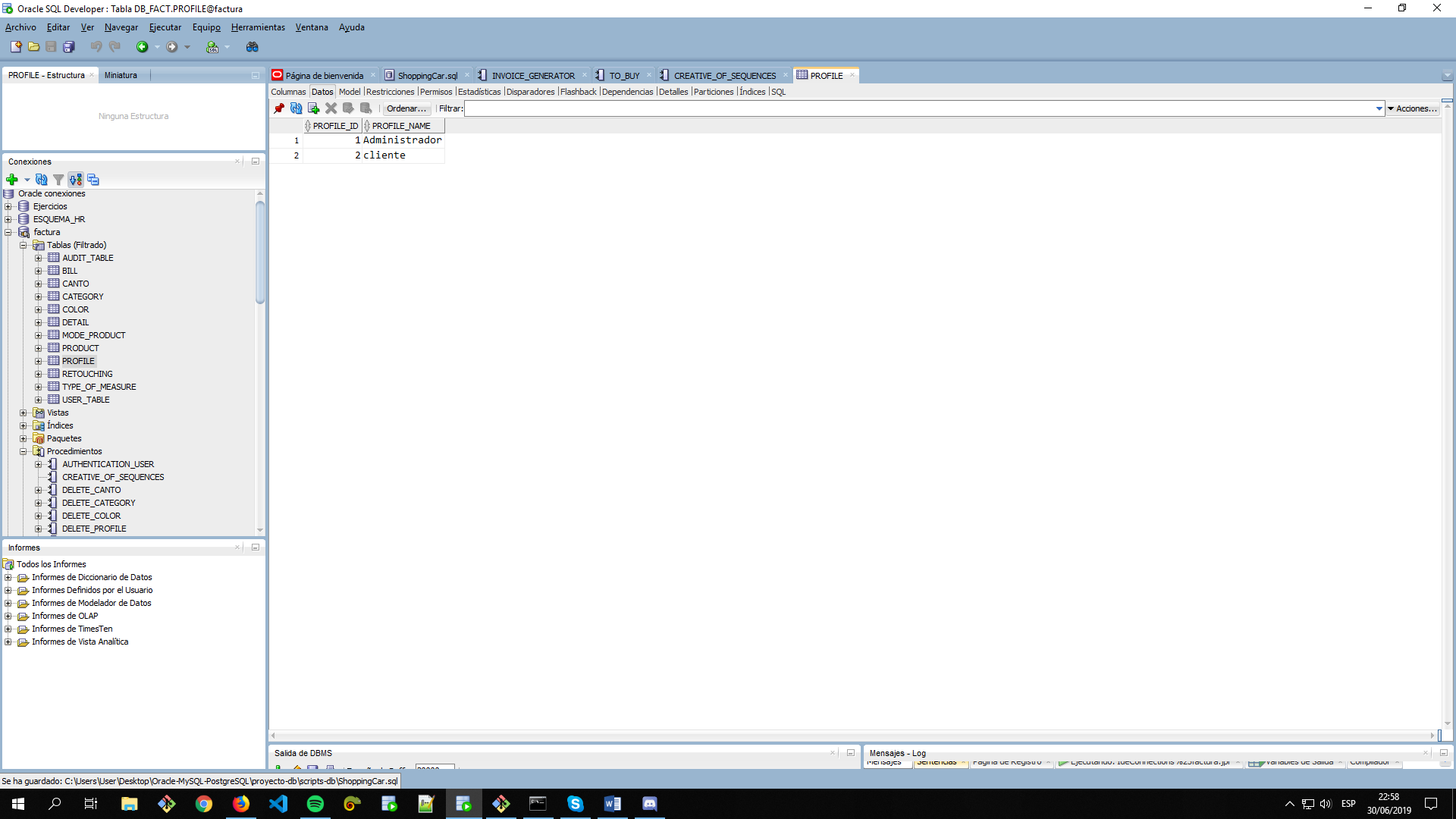
Tabla de perfiles

Tabla de Categoría

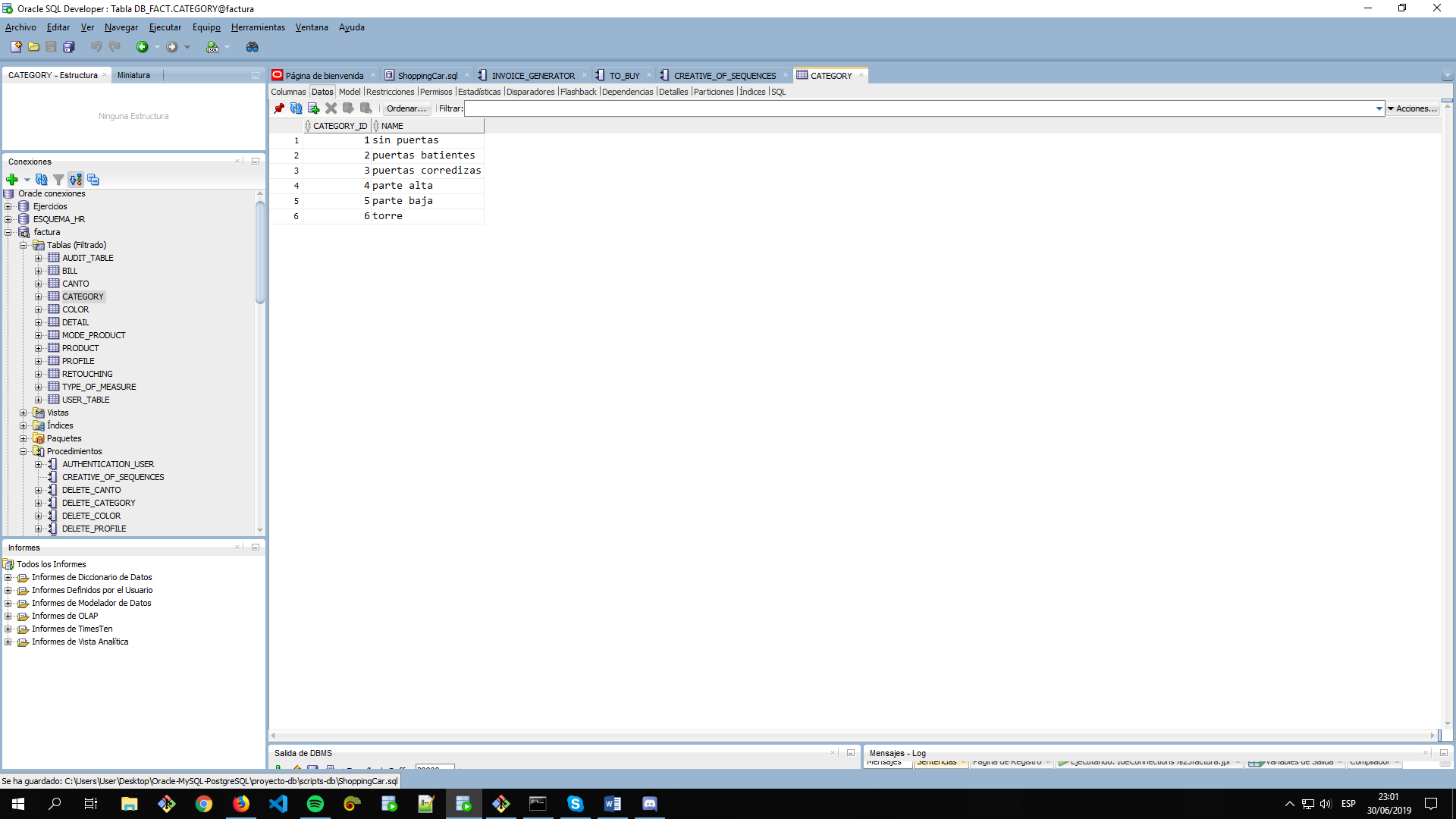


Tabla de productos

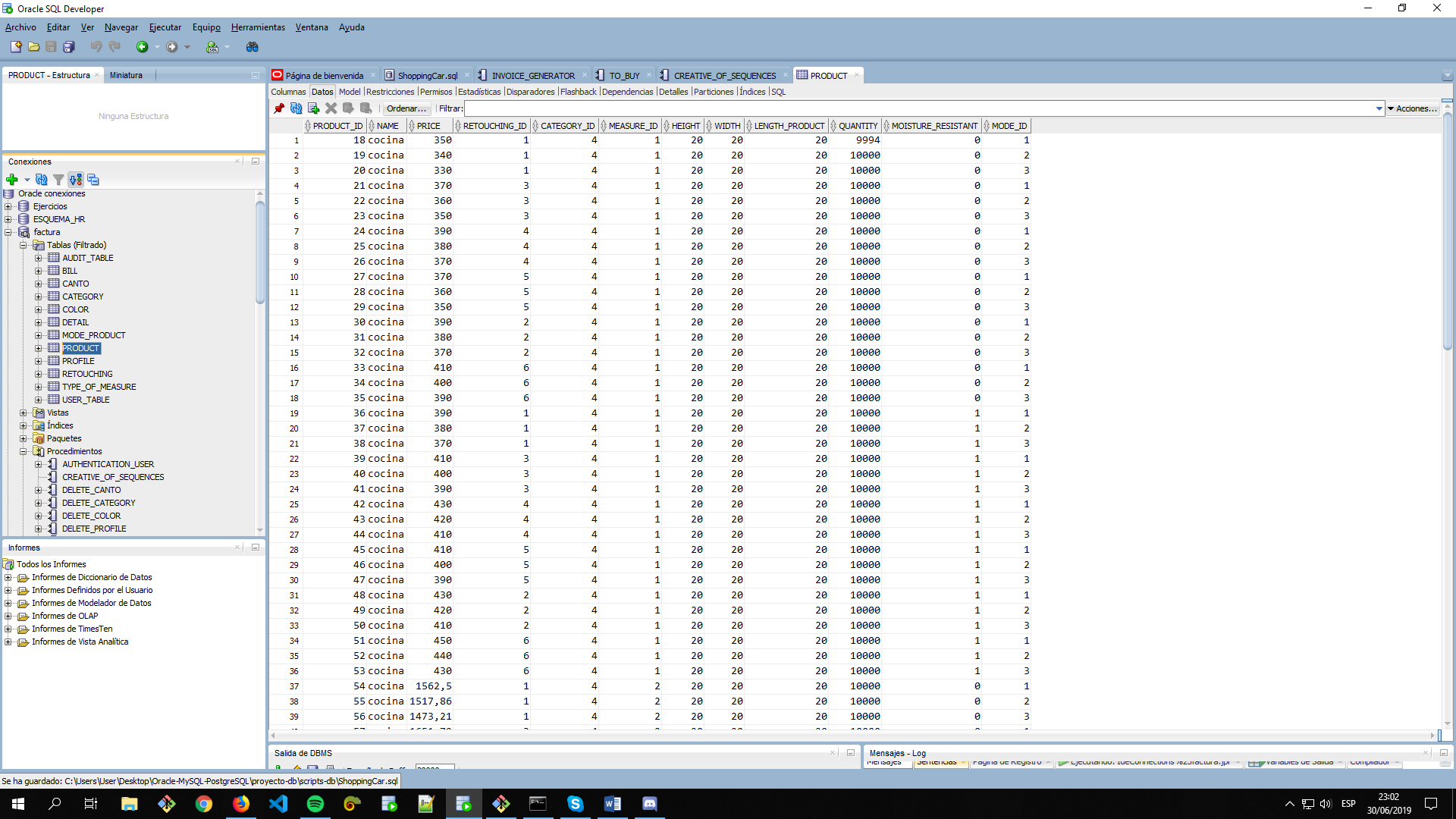
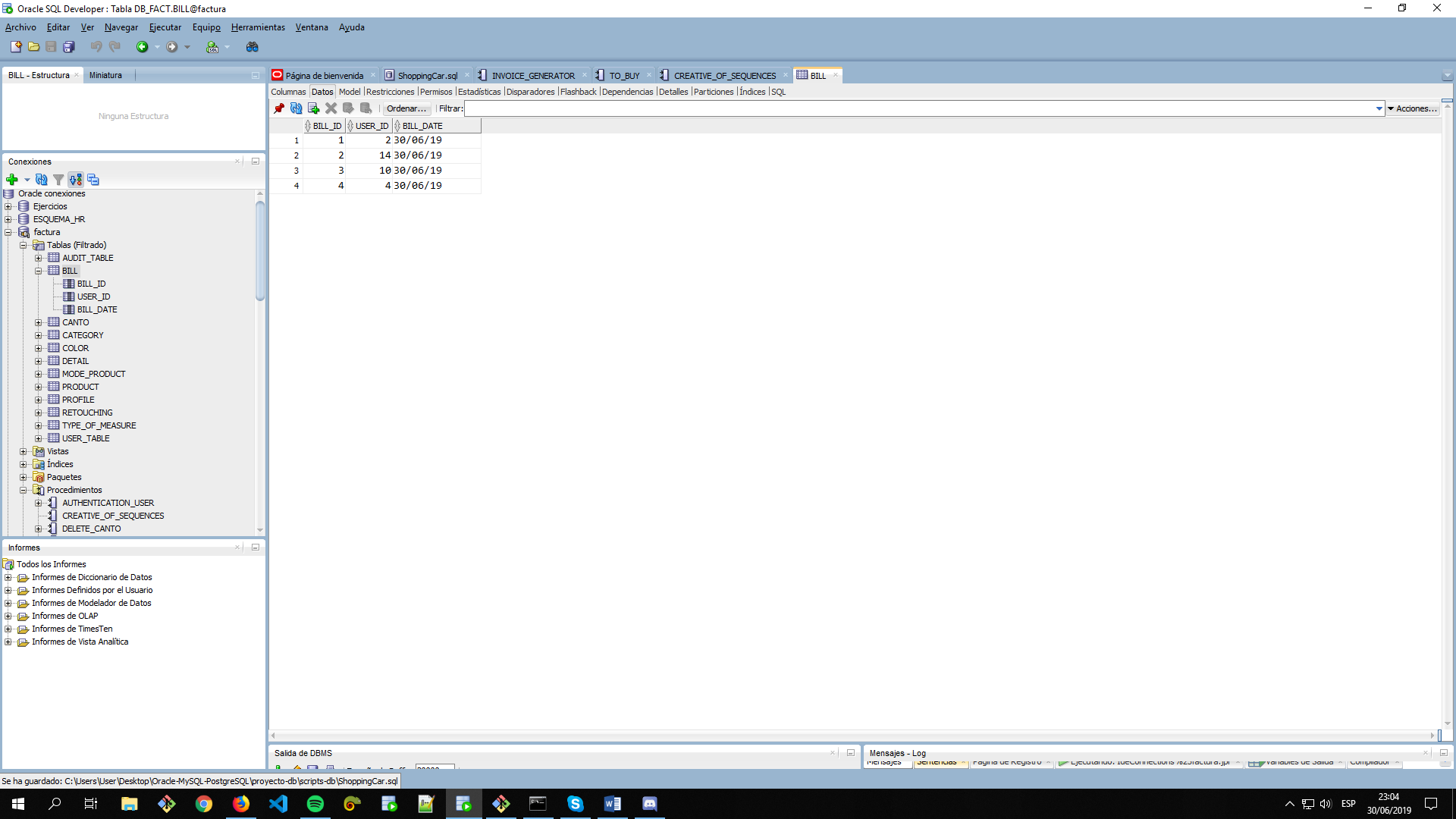


Tabla de factura



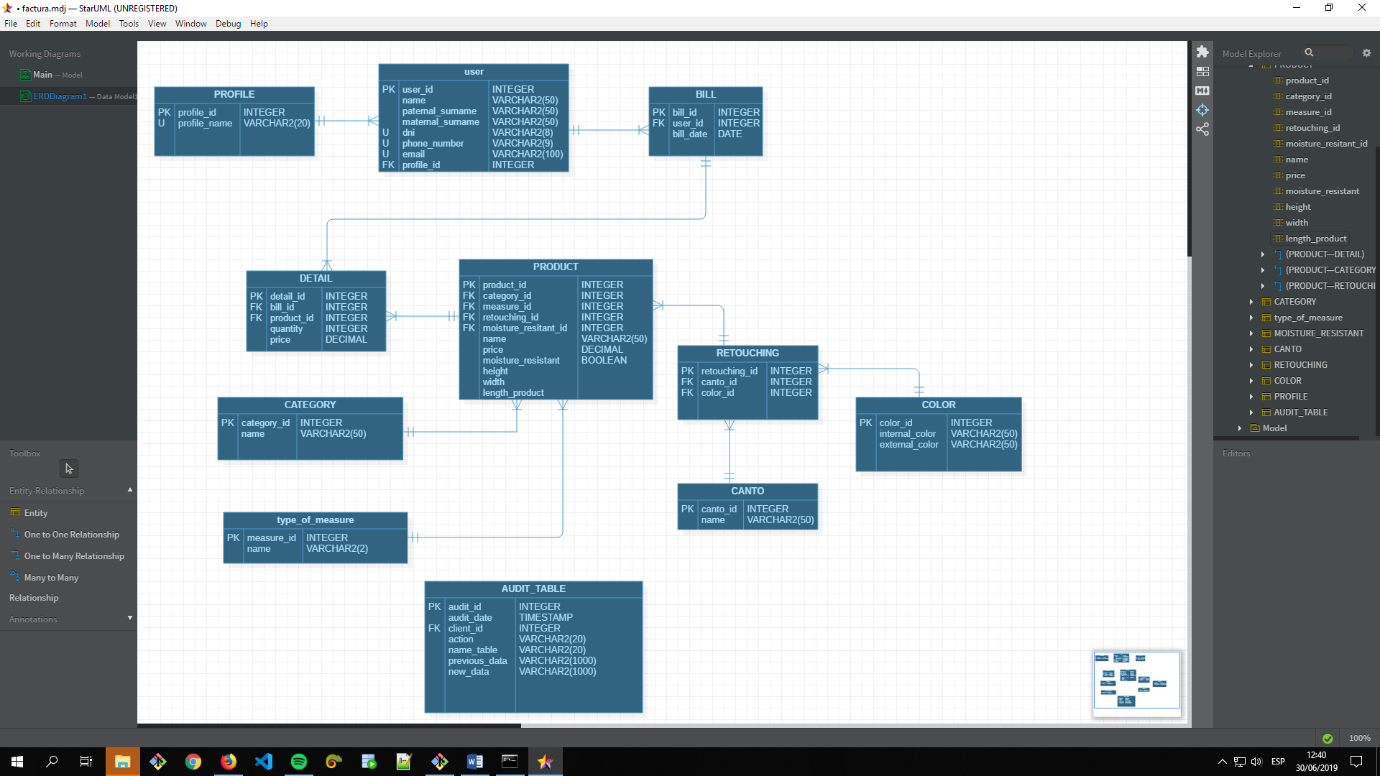
1. INTRODUCCIÓN

El presente proyecto fue realizado en base a los datos proporcionados por la empresa iCREA una empresa ubicada en Av. Angamos Este 1551 - Block I - Piso 1 - Tienda 96, 15048 Lima, dedicada a la fabricación y venta de muebles para el hogar, oficina, negocio y más, en los cuales se pueden destacar productos como: Roperos, cómodas, reposteros, closets, escritorios, etc.. Empresa para la cuál pudimos desarrollar una base de datos de la cual pudiera establecerse un E-Commerce, registrar las ventas del mismo y también se logró desarrollar en la base de datos una tabla para realizar la auditoria correspondiente. Puesto que la problemática de esta empresa es que no contaba con la digitalización de las ventas que realizaba, es decir no tenía un orden o estadísticas de los productos que vendía, con la implementación que se desarrolló esto ya será posible como también emitir facturas y demás implementaciones que se puedan realizar.

1. SERVIDOR DE BASE DE DATOS

|  |  |
| --- | --- |
| **Servidor de Base de Datos** | |
| Equipo: | * HUARANCCA RIVAS ISAAC ERIK * VASQUEZ MOLINA JEFERSON YUNIOR * NAVARRO ESPINOZA PILAR MILAGROS * ROMAN AGUILAR THOM MAURICK |
| IP: |  |
| Sistema Operativo: | Windows , Ubuntu |
| SGBD | Oracle 11g Express Edition |
| Usuario: | db\_fact |
| Clave: | db\_fact% |
| Archivo de restauración: | scriptdb.dmp |
| Otros |  |

1. DISEÑO
   1. DISEÑO DE BASE DE DATO



* 1. TABLA Profile
     1. Creación de la tabla Profile

|  |
| --- |
| create table profile  (  profile\_id number primary key not null,  profile\_name varchar2(20) not null,  constraint uk\_profile\_name UNIQUE(profile\_name)  ); |

* + 1. Secuencia: sq\_profile

|  |
| --- |
| Se creo un procedimiento creador de secuencias |

* + 1. Procedimientos Almacenados
       1. insert\_profile

|  |
| --- |
| create or replace procedure insert\_profile(profile\_n in varchar2)  as  begin  insert into profile values(SQ\_PROFILE.nextval,profile\_n);  end;  / |

* + - 1. update\_profile

|  |
| --- |
| create or replace procedure update\_profile(profile\_i in number,profile\_n in varchar2)  as  begin  update profile set profile\_name = profile\_n where profile\_id = profile\_i;  end;  / |

* + - 1. delete\_profile

|  |
| --- |
| create or replace procedure delete\_profile(profile\_i in number)  as  begin  delete from profile where profile\_id = profile\_i;  end;  / |

* + 1. Disparador trg\_profile\_BIUD

|  |
| --- |
| create or replace trigger trg\_profile\_BIUD before  insert or update or delete on  profile  for  each  row  declare  action\_ varchar2(100);  string\_action\_next varchar2(1000);  string\_action\_pre varchar2(1000):=' ';  begin  IF INSERTING THEN  action\_:='insert';  elsif updating then  action\_:='update';  string\_action\_pre:='old profile : '||:old.profile\_id||', name ='||:old.profile\_name;  elsif deleting then  action\_ :='delete';  string\_action\_pre:='old profile : '||:old.profile\_id||', name : '||:old.profile\_name;  end if;  string\_action\_next:='new profile : '||:new.profile\_id||', name :'||:old.profile\_name;  insert into audit\_table  (audit\_id,audit\_date,user\_id,action,name\_table,previous\_data,new\_data,user\_name)  values(SQ\_AUDIT.nextval, Sysdate,1,action\_, 'profile', string\_action\_pre,string\_action\_next,user);  end; |

* 1. TABLA CATEGORY
     1. Creación de la tabla CATEGORY

|  |
| --- |
| CREATE TABLE CATEGORY  (  category\_id INTEGER,  NAME VARCHAR2(50) NOT NULL,  PRIMARY KEY(category\_id)  ); |

* + 1. Secuencia: NOMBRE\_DE\_LA\_SECUENCIA

|  |
| --- |
| Se creo un procedimiento creador de secuencias |

* + 1. Procedimientos Almacenados
       1. insert\_category

|  |
| --- |
| create or replace procedure insert\_category (category\_name in varchar2)  as  begin  insert into category values (sq\_category.nextval,category\_name);  end;  / |

* + - 1. delete\_category

|  |
| --- |
| create or replace procedure delete\_category(category\_i in number)  as  begin  delete from category where category\_id = category\_i;  end;  / |

* + - 1. update\_category

|  |
| --- |
| create or replace procedure update\_category(category\_i in number,category\_name in varchar2)  as  begin  update category set name = category\_name where category\_id = category\_i;  end;  / |

* + 1. Disparador trg\_category\_BIUD

|  |
| --- |
| create or replace trigger trg\_category\_BIUD before  insert or update or delete on  category  for  each  row  declare  action\_ varchar2(100);  string\_action\_next varchar2(1000);  string\_action\_pre varchar2(1000):=' ';  begin  IF INSERTING THEN  action\_:='insert';  elsif updating then  action\_:='update';  string\_action\_pre:='old category : '||:old.category\_id||', name ='||:old.name;  elsif deleting then  action\_ :='delete';  string\_action\_pre:='old category : '||:old.category\_id||', name : '||:old.name;  end if;  string\_action\_next:='new category : '||:new.category\_id||', name :'||:old.name;  insert into audit\_table  (audit\_id,audit\_date,user\_id,action,name\_table,previous\_data,new\_data,user\_name)  values(SQ\_AUDIT.nextval, Sysdate,1,action\_, 'category', string\_action\_pre,string\_action\_next,user);  end;  / |

* + 1. OTROS

Cualquier otro objeto que hayan creado

|  |
| --- |
| /\*procediemiento creador de secuencias\*/  set SERVEROUTPUT on  create or replace procedure creative\_of\_sequences  as  type type\_sq\_array is varray(11) of varchar2(11);  sq\_array type\_sq\_array;  length\_ integer;  begin  sq\_array:=type\_sq\_array('product','detail','user','category','bill','retouching','MEASURE','color','canto','audit','profile');  length\_:=sq\_array.count;  for sq\_name in 1..length\_ loop  execute immediate 'CREATE SEQUENCE sq\_'  ||to\_char(sq\_array(sq\_name))||' start with 1 increment by 1 NOCYCLE';  end loop;  end;  /  exec creative\_of\_sequences;  --funcion para encriptar la contraseña  create or replace function md5Hash  (input\_string in varchar2)  return varchar2  is  begin  return DBMS\_OBFUSCATION\_TOOLKIT.md5 (input=> UTL\_RAW.cast\_to\_raw(input\_string));  end;  /  -- funcion para autenticar al usuario  create or replace type user\_type  as OBJECT(  user\_id number,  name\_n varchar2(50),  profile\_id number(6,2),  profile\_name varchar2(50));  create or replace type user\_table\_type  as table of user\_type;  create or replace function authenticationUser  (email\_input in varchar2,pass in varchar2)  return user\_table\_type  pipelined  as  cursor cur\_user is select us.user\_id, us.name, pro.profile\_id, pro.profile\_name from user\_table us , profile pro  where us.email=email\_input and us.password=md5Hash(pass) and pro.profile\_id=us.profile\_id;  count\_users number:=-1;  many\_users exception;  no\_user exception;  begin  for data\_use in cur\_user loop  count\_users:=count\_users+1;  if count\_users>1 then  raise many\_users;  end if;  pipe row(user\_type(data\_use.user\_id,data\_use.name,data\_use.profile\_id,data\_use.profile\_name));  end loop;  if count\_users=-1 then  raise no\_user;  else  return;  end if;  exception  when no\_user then  Raise\_application\_error(-20010,'Usuario Incorrecto o contraseÃ±a incorrecta');  when many\_users then  Raise\_application\_error(-20010,'base de datos fallando');  end;  /  --comprar  create or replace procedure to\_buy(product\_ in number,quantity\_ in number,price in number,user\_ in number,bill\_id number)  is  cursor cur\_val\_prod is select quantity from product where product\_id=product\_;  no\_data exception ;  quant number;  begin  --  for data\_ in cur\_val\_prod loop  quant:=data\_.quantity;  end loop;  if quant < quantity\_ then  raise no\_data;  else  insert into detail values (sq\_detail.nextval,bill\_id,product\_,quantity\_,price,user\_);  update product set quantity=quantity-quantity\_ where product\_id=product\_;  return;  end if;  exception  when no\_data then  raise\_application\_error(-20010,'No hay sufuciente producto para comprar');  end;  --invocar\_factura  create or replace procedure invoice\_generator (user\_id in number)  as  begin  insert into bill values (sq\_bill.nextval,user\_id,default);  end; |

