



# UNIVERSIDAD DE CÓRDOBA.

Cadena de Restaurantes Italianos

## Descripción breve

Solución al ejercicio práctico propuesto por Miguel Ángel Gómez Nieto para los alumnos de la asignatura Bases de Datos del segundo curso del grado de Ingeniería Informática.

Francisco José Rodríguez Ramírez, Víctor Monserrat Villatoro

i32rodrf@uco.es, i32moviv@uco.es





# Índice

<b>1. Comprensión del problema y resolución de dudas.</b>	<b>Página 4</b>
<b>2. Documentación formal del problema.</b>	<b>Página 4</b>
2.1. Descripción del problema.	Página 4
2.2. Requisitos.	Página 4
2.2.1. Requisitos respecto a la aplicación móvil y página Web.	Página 4
2.2.2. Requisitos respecto a las ofertas.	Página 4
2.2.2.1. Tipos de ofertas.	Página 5
2.3. Objetivos.	Página 5
2.3.1. Objetivo general.	Página 5
2.3.2. Objetivos específicos.	Página 5
2.3.3. Otros objetivos.	Página 5
2.4. Solución propuesta.	Página 6
<b>3. Análisis de la Base de Datos.</b>	<b>Página 9</b>
3.1. Descripción completa de las entidades, atributos y relaciones.	Página 9
3.2. Diagrama EE/R de la solución conceptual.	Página 14
3.3. Descripción sintáctico-semántica de las restricciones.	Página 19
<b>4. Diseño de la Base de Datos.</b>	<b>Página 20</b>
4.1. Traducción del modelo conceptual al modelo relacional.	Página 20
4.2. Refinamiento del modelo relacional.	Página 20
4.3. Descripción sintáctico-semántica de las restricciones.	Página 21
<b>5. Implementación de la Base de Datos.</b>	<b>Página 22</b>
5.1. Construcción del script de la Base de Datos relacional.	Página 22
5.2. Construcción de scripts de carga de los datos.	Página 25
<b>6. Manipulación de la Base de Datos.</b>	<b>Página 27</b>



## 1. Comprensión del problema y resolución de dudas.

El cliente, propietario de la cadena de restaurantes italianos “La Pizzeta”, nos propone la realización del servicio y la gestión de datos para la empresa. Una vez comprendido el problema planteado, nos surgen una serie de dudas que debemos resolver junto al posible adquirente de nuestro producto. Hemos solicitado la siguiente información para la resolución de nuestras dudas:

- Carta de productos. Necesitamos información de los productos que oferta nuestro cliente, como los valores nutricionales e ingredientes, para incluirlos en la base de datos, web y aplicación móvil.
- Establecimientos. Conocer el número y la situación geográfica de los establecimientos para enviar las ofertas personalizadas a la ubicación de los clientes.
- Precio/coste. Los precios de elaboración de sus productos finales para poder comparar y estudiar el beneficio económico con respecto a las ofertas y si son viables o no al cabo de un plazo determinado de tiempo.

## 2. Documentación formal del problema.

### 2.1. Descripción del problema.

El problema planteado por el cliente es el diseño y la elaboración de una base de datos, cuyo contenido será información del negocio para poder realizar ofertas personalizadas al usuario. A esta se podrá acceder tanto a través de una aplicación móvil como desde una página Web.

### 2.2. Requisitos.

#### 2.2.1. Requisitos respecto a la aplicación móvil y página Web.

La aplicación móvil deberá ser gratuita y operativa para los principales sistemas operativos móviles. Almacenará una mínima cantidad de información requerida, que será: correo electrónico, fecha de nacimiento y distrito postal. El usuario podrá ampliar su información, introduciendo datos como su nombre, apellidos, dirección, teléfonos, sexo, cuentas en redes sociales, etc. Toda la información de los usuarios será almacenada de forma segura en la Web. El registro deberá ser lo más rápido y simple posible. El usuario tendrá acceso a su información histórica.

#### 2.2.2. Requisitos respecto a las ofertas.

Cada oferta irá asociada a un producto y se realizarán sobre alguno o varios productos del cliente. Tendrán vigencia horaria y podrán ser válidas para uno o varios de los establecimientos del cliente. Las ofertas serán notificadas a través de notificaciones push, siendo personalizadas para cada usuario. No todos los clientes recibirán las mismas ofertas,

permitiendo también el envío a usuarios o grupos VIPs, que alcanzan porcentajes altos de uso de las ofertas. Las ofertas serán ilimitadas para los consumidores y se harán en base a uno o varios atributos de la información de los usuarios de forma que la campaña de marketing se pueda segmentar. Al cliente le interesa conocer el porcentaje de éxito de esta campaña de marketing, es decir, los usuarios que han recibido la oferta y los que la utilizaron.

#### **2.2.2.1. Tipos de ofertas.**

- Descuento sobre el precio.
- Descuento en un producto por la compra de otro.
- Productos gratis por la compra de otro.
- Productos gratis por comprar en el establecimiento.
- Productos gratis por interaccionar en las redes sociales.

### **2.3. Objetivos.**

#### **2.3.1. Objetivo general.**

El objetivo final es construir un producto capaz de llevar la gestión de las ofertas, teniendo acceso desde el sistema Web a cualquier información relacionada con el negocio de la base de datos (usuarios, ofertas, historial, estadísticas). La aplicación Web deberá poder ser ejecutada en cualquier dispositivo y bajo cualquier sistema operativo.

#### **2.3.2. Objetivos específicos.**

Para alcanzar el objetivo general, el cliente solicita que el producto permita mantener información de todos los usuarios del sistema, de las ofertas, de las notificaciones, del uso de las ofertas por los consumidores, información contable de las ventas gracias a las ofertas y producir estadísticas de rendimiento y efectividad, en tablas y gráficos.

#### **2.3.3. Otros objetivos requeridos.**

Las condiciones añadidas para la contratación del servicio son:

- La solución debe realizarse con independencia de plataforma o sistema operativo.
- Debe ser simple de usar, intuitiva y gráfica, permitiendo una gestión en el menor número de pantallas y clicks posibles.
- Tener un control alto sobre errores de introducción o manipulación de datos por parte del usuario.
- Debe permitir almacenar la información histórica.
- No debe requerir de instalación de ningún software de pago o alquiler, es decir, totalmente gratuita.
- Su uso podrá ser desde cualquier dispositivo móvil.

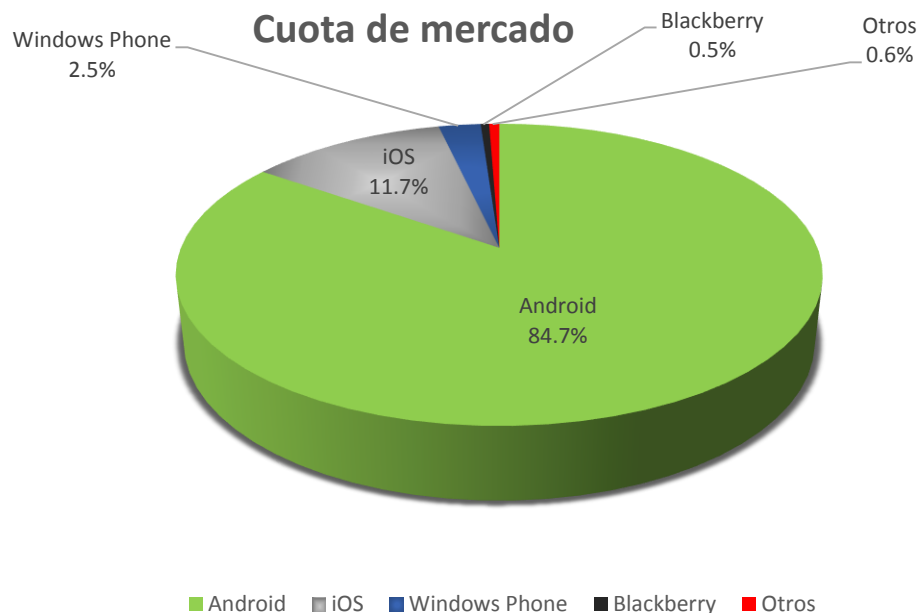
- La solución deberá estar definida con anterioridad al 31/12/2014.

#### 2.4. Solución propuesta.

Tras el análisis y comprensión del problema planteado por el cliente nos disponemos a encontrar una solución eficiente y eficaz.

Respecto a la aplicación móvil, será totalmente gratuita para el usuario. No tendría sentido hablar de una aplicación de pago puesto que el interés del cliente es el envío de ofertas sobre sus productos a los usuarios, incentivando así el consumo en sus establecimientos, y no conseguir el beneficio económico a través de la venta de esta aplicación.

Respecto a los sistemas operativos para el mercado de la aplicación se ha realizado un estudio sobre la cuota de mercado actual en los distintos sistemas operativos móviles.



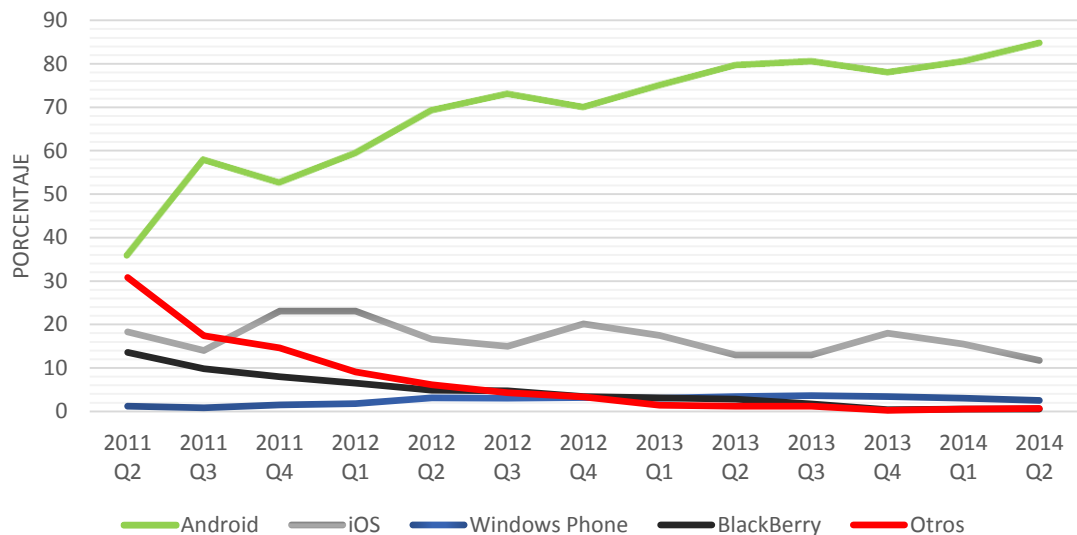
**\*Figura 1.** Cuota de mercado actual de los sistemas operativos en dispositivos móviles.

Como se observa en la Figura 1, la cuota de mercado es muy favorable para los sistemas Android, seguido de lejos por iOS. El resto quedan en un porcentaje tan reducido que no es viable ofertar el soporte de esta aplicación en el resto de plataformas.

**\*Fuente:** <http://www.idc.com/prodserv/smartphone-os-market-share.jsp>

Además, según el estudio realizado en un tiempo prolongado (2011-2014), se puede apreciar una subida muy considerable por parte de Android, que ha conseguido hacerse prácticamente con la hegemonía del mercado, dejando muy mermada a su competencia.

## Tendencia en las cuotas de mercado



**\*Figura 2.** Tendencia en las cuotas de mercado de sistemas operativos en dispositivos móviles.

En la Figura 2, se aprecia una subida de las cuotas de mercado en dispositivos Android y Windows Phone, aunque no es tan grande en este último.

Por el contrario, iOS y BlackBerry han sufrido una gran bajada al igual que el resto de plataformas. Se pueden observar claramente que las tendencias de mercado son muy favorables para unos pero no para otros. Por tanto, actualmente lo más factible sería soportar la aplicación en Android y iOS debido a su gran cuota de mercado actual.

En un futuro, se podría estudiar el desarrollo para Windows Phone si su cuota sigue subiendo durante un gran tiempo. Para la exportación de nuestra aplicación tanto en Android como en iOS se deberán incluir en las tiendas oficiales de cada sistema operativo (Google Play y App Store respectivamente).



**\*Fuente:** <http://www.idc.com/prodserv/smartphone-os-market-share.jsp>



La información mínima será recogida en el proceso del registro de usuario para hacerlo lo más rápido y simple posible. Los datos requeridos al usuario serán correo electrónico, fecha de nacimiento, distrito postal y una contraseña para acceder a una cuenta personalizada para cada usuario.

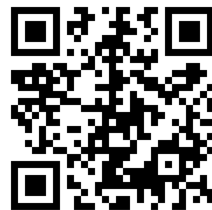
Además, esta información facilitada por el usuario podrá ser ampliada en beneficio de ofertas personalizadas, aplicadas sobre el mismo. Se podrán introducir datos como el nombre, los apellidos, el tratamiento, provincia, localidad, dirección (calle y número), teléfonos, sexo, cuentas en las redes (Twitter y Facebook), etc.



A más información proporcionada, mayor será la personalización de estas ofertas. Otros datos que creemos útiles para la aplicación será poder introducir alergias o intolerancias a determinados tipos de alimentos, para así poder avisar al usuario de inclusión de esos ingredientes en el producto. Asimismo, se podrán incluir los valores nutricionales asociados a cada producto final.

El usuario tendrá acceso a su información histórica, la cual se almacenará en la base de datos, teniendo como campos principales las ofertas recibidas, las ofertas utilizadas y cantidad de ocasiones usadas, y por último el dinero que se ha ahorrado al usarlas. Todos estos datos estarán representados en función del tiempo.

Respecto a las ofertas, concebimos la idea de su canjeo a través de códigos QR generados, donde cada uno es individual y lleva asociado una oferta a un cliente.



Este código será enviado a los clientes seleccionados para el uso de la oferta. Los clientes podrán canjearlo tanto a través de su impresión en papel, como a través del mismo dispositivo móvil.

Las ofertas podrán ser personalizadas en base a una serie de características propias de los clientes, como a una franja de edad, distrito postal, al tiempo que se lleva registrado o al número de ofertas usadas por ejemplo.

Las ofertas estarán destinadas a distintos establecimientos, donde podrán ser canjeadas unas determinadas ofertas, llevarán asociada una fecha de lanzamiento y una fecha de caducidad, el tipo de oferta, una breve descripción de ella y un código.

La finalidad de nuestra solución planteada es el superávit de ventas a través de la difusión de ofertas en forma de notificaciones push a los clientes en sus respectivos

dispositivos móviles y así, incentivar el consumo de los clientes en los establecimientos de nuestro cliente.

### 3. Análisis de la Base de Datos.

#### 3.1. Descripción completa de las entidades, atributos y relaciones.

USUARIO: cliente registrado en La Pizzeta.

ID-usuario está definido por el email. Tiene dominio string. Es una entidad con cardinalidad (1, not null). Es el identificador único que representa a cada uno de los usuarios.

Nombre será el nombre del usuario. Tiene dominio string. Es una entidad con cardinalidad (1, not null)

Apellido-1 será el primer apellido del usuario. Tiene dominio string. Es una entidad con cardinalidad (1, not null).

Apellido-2 será el segundo apellido del usuario. Tiene dominio string. Es una entidad con cardinalidad (1).

Fecha-nacimiento será la fecha de nacimiento del usuario. Tiene dominio date. Es una entidad con cardinalidad (1, not null).

Sexo será el sexo del usuario. Tiene de dominio string. Es una entidad con cardinalidad (1, not null). Puede tomar los valores:

I: No contesta

H: Hombre.

M: Mujer.

Dirección será la dirección del usuario. Tiene dominio string. Es una entidad con cardinalidad (1).

Localidad será la localidad del usuario. Tiene dominio string. Es una entidad con cardinalidad (1).

Postal será el código postal del usuario. Tiene dominio number. Es una entidad con cardinalidad (1).

Telefono-movil será el número del teléfono móvil del usuario. Tiene dominio number. Es una entidad con cardinalidad (1).

Facebook será el nombre de usuario de Facebook del usuario. Tiene dominio string. Es una entidad con cardinalidad (1).

Twitter será el nombre de usuario de Twitter del usuario. Tiene dominio string. Es una entidad con cardinalidad (1).

Fecha-alta será la fecha en la que el usuario se da de alta. Tiene dominio date. Es una entidad con cardinalidad (1, not null).

Fecha-baja será la fecha en la que el usuario se da de baja. Tiene dominio date. Es una entidad con cardinalidad (1).

Estado será el estado que tendrá el usuario (activo o inactivo). Tiene dominio string. Es una entidad con cardinalidad (1, not null).

ESTABLECIMIENTO: lugares preparados para la venta de la cadena de restaurantes.

ID-shop es el identificador único que representa a cada uno de los establecimientos. Tiene dominio serial. Es una entidad con cardinalidad (1, not null).

Nombre será el nombre del establecimiento. Tiene dominio string. Es una entidad con cardinalidad (1, not null)

Dirección será la dirección del establecimiento. Tiene dominio string. Es una entidad con cardinalidad (1)

Localidad será la localidad del establecimiento. Tiene dominio string. Es una entidad con cardinalidad (1)

Provincia será la provincia en la que se encuentra el establecimiento. Tiene dominio string. Es una entidad con cardinalidad (1)

Postal será el código postal el cual se le asigna al establecimiento. Tiene dominio number. Es una entidad con cardinalidad (1)

Telefono será el número de teléfono del establecimiento. Tiene dominio number. Es una entidad con cardinalidad (1)

Latitud será la latitud asociada al establecimiento. Tiene dominio string. Es una entidad con cardinalidad (1)

Longitud será la longitud asociada al establecimiento. Tiene dominio string. Es una entidad con cardinalidad (1)

Fecha-alta será la fecha en la que inauguró el establecimiento. Tiene dominio date. Es una entidad con cardinalidad (1, not null)

Fecha-baja será la fecha en la que se da de baja el establecimiento. Tiene dominio date. Es una entidad con cardinalidad (1)

Estado será el estado que tendrá el establecimiento (activo o inactivo). Tiene dominio string. Es una entidad con cardinalidad (1, not null). Puede tomar los valores:

0: Activo

1: Inactivo

PRODUCTO: productos preparados para su venta asociados a la Pizzeta.

ID-producto está definido por el serial del producto. Tiene dominio serial. Es una entidad con cardinalidad (1, not null).

Nombre será el nombre del producto. Tiene dominio string. Es una entidad con cardinalidad (1, not null).

Descripción será una breve descripción del producto. Tiene dominio string. Es una entidad con cardinalidad (1, not null).

Detalle será una descripción más detallada del producto. Tiene dominio string. Es una entidad con cardinalidad (1, not null).

Foto será una dirección URL de una foto. Tiene dominio URL. Es una entidad con cardinalidad (1, not null).

Coste será el coste que le cuesta al cliente obtener el producto. Tiene dominio real. Es una entidad con cardinalidad (1, not null).

Venta será el precio por el que se vende el producto. Tiene dominio real. Es una entidad con cardinalidad (1, not null).

Fecha-alta será la fecha en la que se da de alta un producto. Tiene dominio date. Es una entidad con cardinalidad (1, not null).

Fecha-baja será la fecha en la que se da de baja un producto. Tiene dominio date. Es una entidad con cardinalidad (1).

Estado será el estado que tendrá el producto (activo o inactivo). Tiene dominio string. Es una entidad con cardinalidad (1, not null). Puede tomar los valores:

0: Activo.

1: Inactivo.

OFERTA: permite mantener información de las ofertas de la empresa.

ID-oferta es el identificador único que representa a cada una de las ofertas.

Tiene dominio serial. Es una entidad con cardinalidad (1, not null).

Nombre será el nombre de la oferta. Tiene dominio string. Es una entidad con cardinalidad (1, not null).

Descripción será una breve descripción de la oferta. Tiene dominio string. Es una entidad con cardinalidad (1, not null).

Detalle será una descripción más detallada de la oferta. Tiene dominio string. Es una entidad con cardinalidad (1, not null).

Foto será una dirección URL de una foto. Tiene dominio URL. Es una entidad con cardinalidad (1, not null).

Venta será el precio por el que se vende la oferta. Tiene dominio real. Es una entidad con cardinalidad (1, not null).

Fecha-inicio será la fecha en la que se inicia una oferta. Tiene dominio date. Es una entidad con cardinalidad (1, not null).

Fecha-termina será la fecha en la que se termina una oferta. Tiene dominio date. Es una entidad con cardinalidad (1, not null).

Fecha-alta será la fecha en la que se da de alta una oferta. Tiene dominio date. Es una entidad con cardinalidad (1, not null).

Fecha-baja será la fecha en la que se da de baja una oferta. Tiene dominio date. Es una entidad con cardinalidad (1).

Estado será el estado que tendrá el producto (activo o inactivo). Tiene dominio string. Es una entidad con cardinalidad (1, not null). Puede tomar los valores:

0: Activo.

1: Inactivo.

OFERTA-ESTABLECIMIENTO: permite mantener información de los establecimientos asociados a las oferta.

ID-Of-Es 1 será el identificador único que representa a una oferta en un establecimiento. Es una entidad con cardinalidad (1, not null).

ID-oferta será el identificador único que representa a una oferta. Tiene dominio serial.

ID-shop será el identificador único que representa a un establecimiento. Tiene dominio serial.

Estado será el estado que tendrá una oferta en un establecimiento (activo o inactivo). Tiene dominio string. Es una entidad con cardinalidad (1, not null).

Puede tomar los valores:

0: Activo.

1: Inactivo.

OFERTA-USUARIO: permite mantener información de las ofertas enviadas a los usuarios.

ID-Of-Us será el identificador único que representa a un usuario que consuma una oferta. Es una entidad con cardinalidad (1, not null).

ID-oferta será el identificador único que representa a una oferta. Tiene dominio serial.

ID-usuario será el identificador único que representa a un usuario. Tiene dominio serial.

Fecha-envío será la fecha en la que se envía una oferta a un usuario. Tiene dominio date. Es una entidad con cardinalidad (1, not null).

Estado-envío será el estado que puede tener una oferta sobre un usuario. Tiene dominio string. Es una entidad con cardinalidad (1, not null).

Los estados que puede tener una oferta son:

0: No recibido.

1: Recibido y descartado.

2: Recibido y leído.

Fecha-lectura será la fecha en la que el usuario lee la oferta. Tiene dominio date. Es una entidad con cardinalidad (1).

Fecha-uso será la fecha en la que el usuario consume la oferta. Tiene dominio date. Es una entidad con cardinalidad (1).

Venta será el precio por el que se vende la oferta. Tiene dominio real. Es una entidad con cardinalidad (1).

OFERTA-PRODUCTO: Esta Tabla permite mantener información de los productos asociados a las ofertas.

ID-OF-Pr es el identificador único que representa a un usuario que consume un

producto. Es una entidad con cardinalidad (1, not null).

ID-oferta será el identificador único que representa a una oferta. Tiene dominio serial.

ID-producto es el identificador único que representa a cada producto. Tiene dominio serial.

Venta será el precio por el que se vende la oferta. Tiene dominio real. Es una entidad con cardinalidad (1, not null).

Información sobre los estados posibles.

Estado será el estado que tendrá una oferta en un establecimiento (activo o inactivo). Tiene dominio string. Es una entidad con cardinalidad (1, not null).

Puede tomar los valores:

0: Activo.

1: Inactivo.

USUARIO-PRODUCTO: permite mantener información los productos que consume el usuario.

Atributo Descripción Dominio Identificador Cardinalidad.

ID-Us-Pr será el identificador único que representa a un usuario que consuma un producto. Es una entidad con cardinalidad (1, not null).

ID-Of-Us será el identificador único que representa a un usuario que consuma una oferta. Tiene dominio ID-Of-Us. Es una entidad con cardinalidad (1, not null).

ID-producto es el identificador único que representa a cada producto. Tiene dominio serial. Es una entidad con cardinalidad (1, not null).

Unidades será la cantidad de unidades que se venden de un producto. Tiene dominio integer. Es una entidad con cardinalidad (1, not null).

Venta será el precio por el que se vende la oferta. Tiene dominio real. Es una entidad con cardinalidad (1, not null).

USUARIO-REDES: permite mantener información del comportamiento de los usuarios en las redes.

ID-redes es el identificador único que representa las redes de un usuario. Tiene dominio serial. Es una entidad con cardinalidad (1, not null).

ID-usuario está definido por el email. Es el identificador único que representa a cada uno de los usuarios. Tiene dominio serial. Es una entidad con cardinalidad (1, not null).

ID-oferta es el identificador único que representa a cada una de las ofertas. Tiene dominio serial.

Fecha-acción será la fecha en la que un usuario realiza una acción. Tiene dominio date. Es una entidad con cardinalidad (1, not null).

Tipo-red será el tipo de red que el usuario tendrá. Tiene dominio date. Es una entidad con cardinalidad (1, not null).

Los distintos tipos de red que puede tener un socio son:

0: Facebook.

1: Twitter.

Tipo-acción será el tipo de acción que realiza un usuario en su Tipo-red. Tiene dominio string. Es una entidad con cardinalidad (1, not null).

Los distintos tipos de acciones que puede realizar un usuario son:

0: Me gusta.

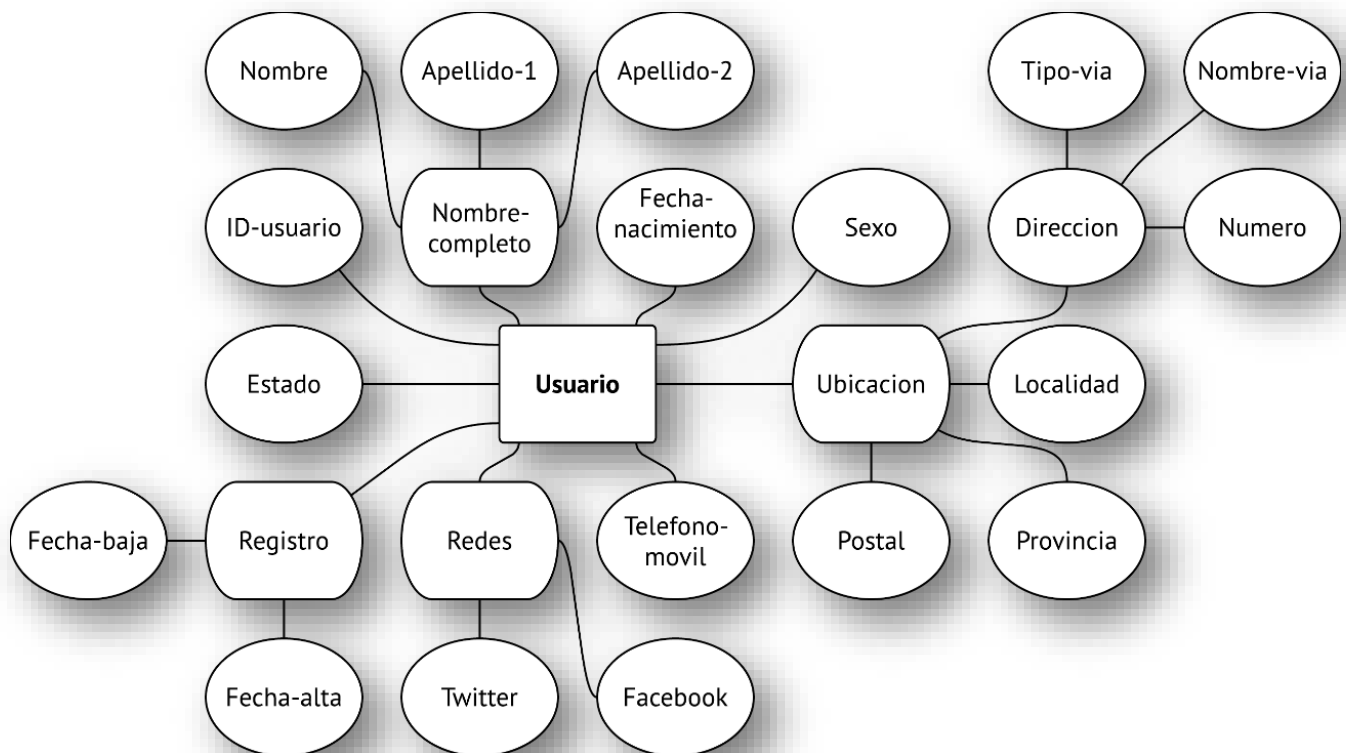
1: Compartir.

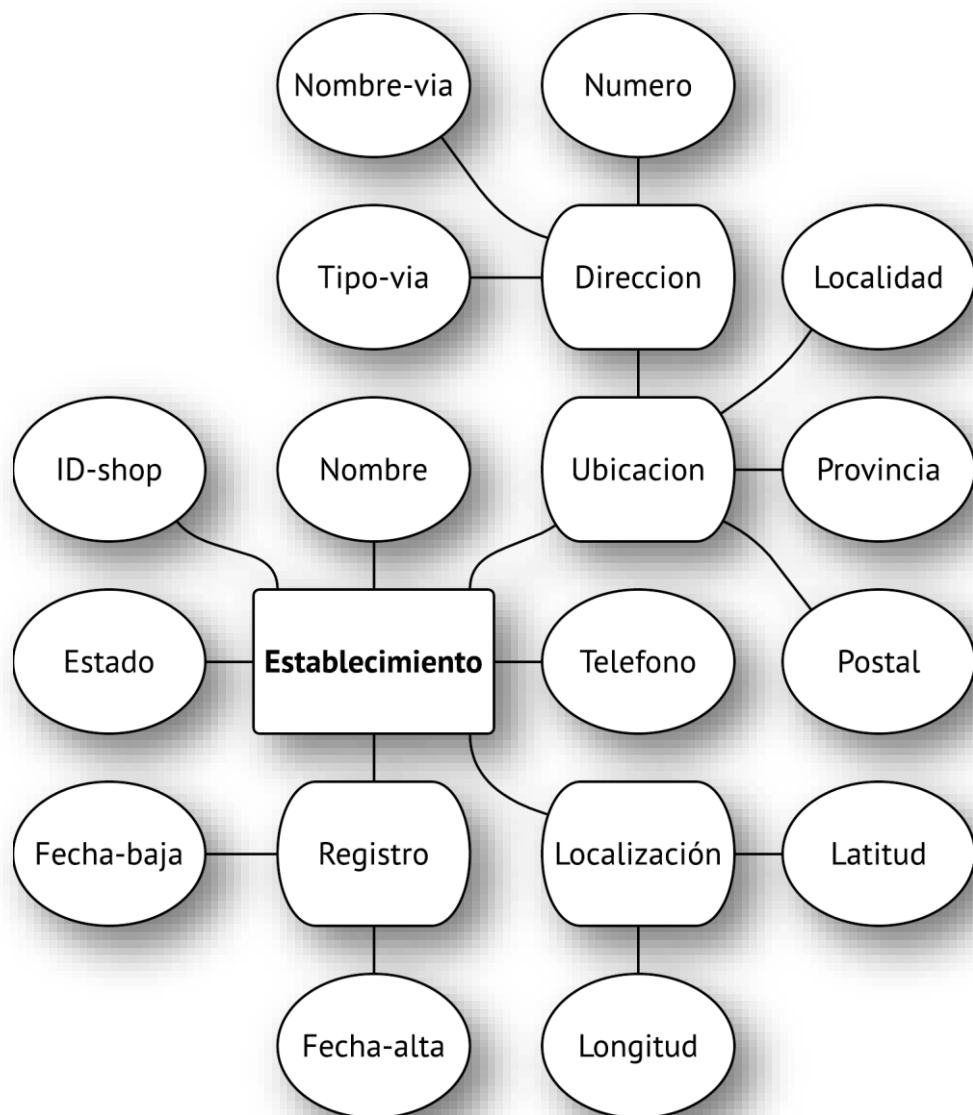
2: Tuitear.

3: Favorito.

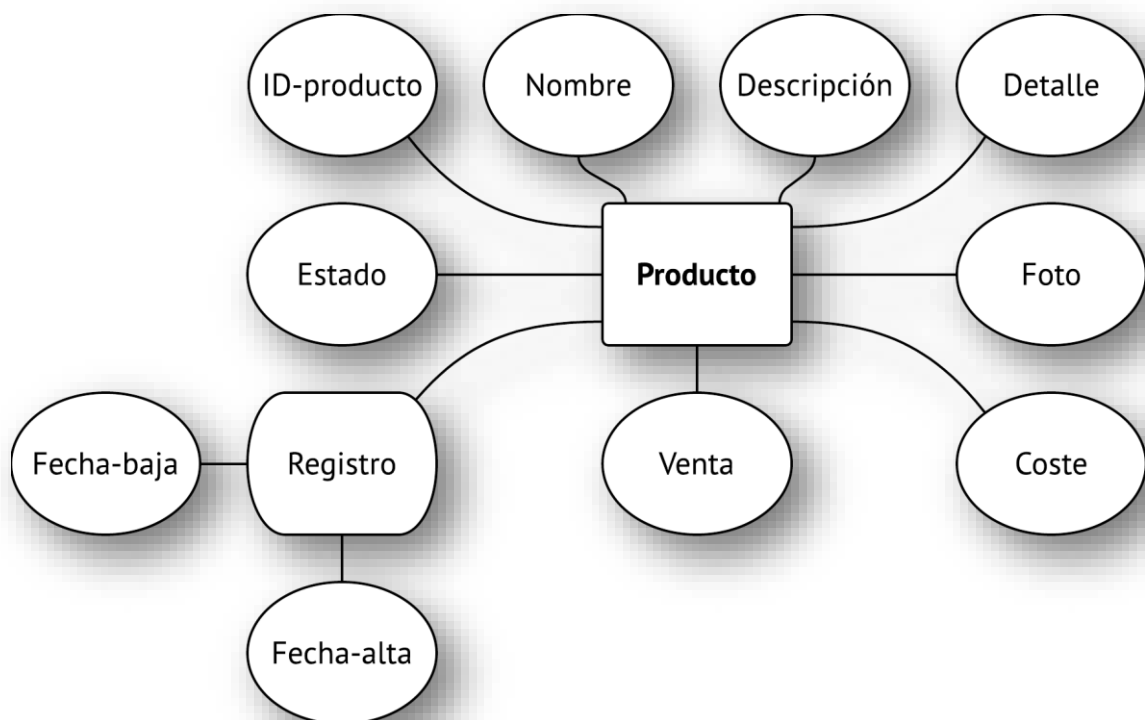
Seguimiento será el seguimiento que tendrá el cliente sobre un usuario. Tiene dominio integer. Es una entidad con cardinalidad (1).

### 3.2. Diagrama EE/R de la solución conceptual.

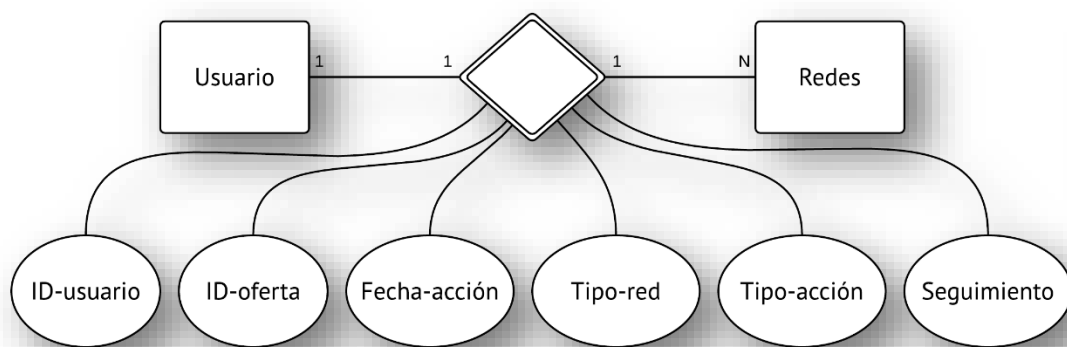
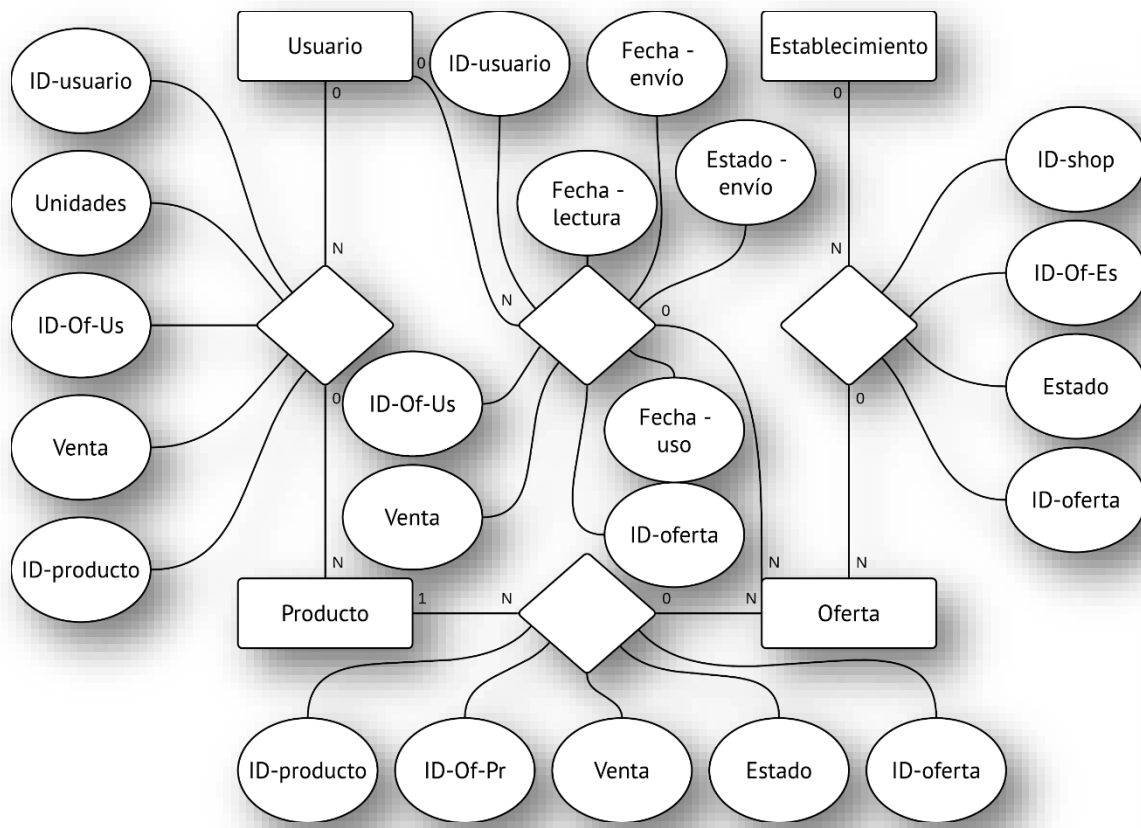












### 3.3. Descripción sintáctico-semántica de las restricciones.

En la tabla usuario:

fecha de alta no puede ser nulo.

estado no puede ser nulo.

sexo tiene que ser (I, H o M).

estado tiene que ser 0 o 1.

En la tabla establecimientos:

nombre no puede ser nulo.

fecha de alta no puede ser nulo.

estado no puede ser nulo.

estado tiene que ser 0 o 1.

En la tabla producto:

nombre no puede ser nulo.

descripcion no puede ser nulo.

detalle no puede ser nulo.

foto no puede ser nulo.

coste no puede ser nulo.

venta no puede ser nulo.

fecha\_alta no puede ser nulo.

estado no puede ser nulo.

estado tiene que ser 0 o 1.

En la tabla oferta:

nombre no puede ser nulo.

descripcion no puede ser nulo.

detalle no puede ser nulo.

foto no puede ser nulo.

venta no puede ser nulo.

fecha\_inicio no puede ser nulo.

fecha\_alta no puede ser nulo.

estado no puede ser nulo.

estado tiene que ser 0 o 1.

En la tabla oferta\_producto:

venta no puede ser nulo.

En la tabla oferta\_usuario:

fecha\_envio no puede ser nulo.

envio no puede ser nulo.

En la tabla usuario\_producto:

unidades no puede ser nulo.

venta no puede ser nulo.

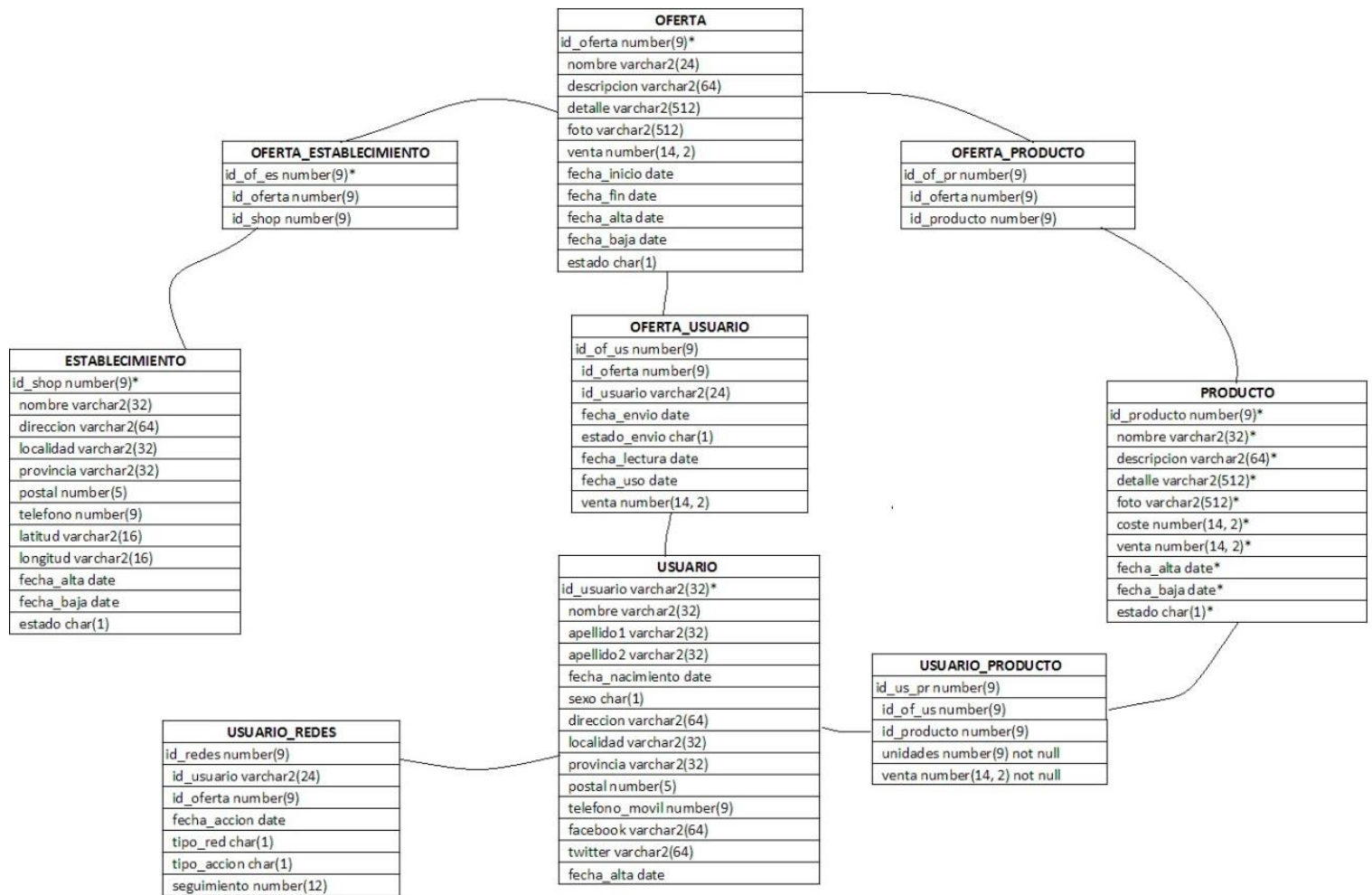
En la tabla usuario\_redes:

fecha\_accion no puede ser nulo.

tipo\_red no puede ser nulo.  
 tipo\_accion no puede ser nulo.  
 tipo\_red tiene que ser 0 o 1.  
 tipo\_accion tiene que ser 0, 1, 2 o 3.

#### 4. Diseño de la Base de Datos.

##### 4.1. Traducción del modelo conceptual al modelo relacional.



##### 4.2. Refinamiento del modelo relacional.

Usuario (id\_usuario, nombre, apellido1, apellido2, fecha\_nacimiento, sexo, dirección, localidad, provincia, postal, teléfono\_movil, Facebook, twitter, fecha\_alta)

Establecimiento (id\_shop, nombre, dirección, localidad, provincia, postal, teléfono, latitud, longitud, fecha\_alta, fecha\_baja, estado)

Producto (id\_producto, nombre, descripción, detalle, foto, coste, venta, fecha\_alta, fecha\_baja, estado)

Oferta (id\_oferta, nombre, descripción, detalle, foto, venta, fecha\_inicio, fecha\_fin, fecha\_alta, fecha\_baja, estado)

Oferta\_establecimiento (id\_of\_es, id\_oferta, id\_shop)

Oferta\_producto (id\_of\_pr, id\_oferta, id\_producto)

Oferta\_usuario (id\_of\_us, id\_oferta, id\_usuario, fecha\_envío, estado\_envío, fecha\_lectura, fecha\_uso, venta)

Usuario\_producto (id\_us\_pr, id\_of\_us, id\_producto, unidades, venta)

Usuario\_redes (id\_redes, id\_usuario, id\_oferta, fecha\_accion, tipo\_red, tipo\_accion, seguimiento)

#### 4.3. Descripción sintáctico-semántica de las restricciones.

En la tabla usuario:

fecha de alta no puede ser nulo.

estado no puede ser nulo.

sexo tiene que ser (I, H o M).

estado tiene que ser 0 o 1.

En la tabla establecimientos:

nombre no puede ser nulo.

fecha de alta no puede ser nulo.

estado no puede ser nulo.

estado tiene que ser 0 o 1.

En la tabla producto:

nombre no puede ser nulo.

descripcion no puede ser nulo.

detalle no puede ser nulo.

foto no puede ser nulo.

coste no puede ser nulo.

venta no puede ser nulo.

fecha\_alta no puede ser nulo.

estado no puede ser nulo.

estado tiene que ser 0 o 1.

En la tabla oferta:

nombre no puede ser nulo.

descripcion no puede ser nulo.

detalle no puede ser nulo.  
foto no puede ser nulo.  
venta no puede ser nulo.  
fecha\_inicio no puede ser nulo.  
fecha\_alta no puede ser nulo.  
estado no puede ser nulo.  
estado tiene que ser 0 o 1.  
En la tabla oferta\_producto:  
venta no puede ser nulo.  
En la tabla oferta\_usuario:  
fecha\_envio no puede ser nulo.  
envio no puede ser nulo.  
En la tabla usuario\_producto:  
unidades no puede ser nulo.  
venta no puede ser nulo.  
En la tabla usuario\_redes:  
fecha\_accion no puede ser nulo.  
tipo\_red no puede ser nulo.  
tipo\_accion no puede ser nulo.  
tipo\_red tiene que ser 0 o 1.  
tipo\_accion tiene que ser 0, 1, 2 o 3.

## 5. Implementación de la Base de Datos.

### 5.1. Construcción del script de la Base de Datos relacional.

```
1. /* Creación de las entidades y atributos así como
   relaciones de nuestro problema a partir de sql:*/
2.
3. drop table if exists usuario_redes;
4. drop table if exists usuario_producto;
5. drop table if exists oferta_usuario;
6. drop table if exists oferta_producto;
7. drop table if exists oferta_establecimiento;
8. drop table if exists oferta;
9. drop table if exists producto;
10. drop table if exists establecimiento;
11. drop table if exists usuario;
12.
13. /*Se crea la tabla USUARIOS */
14. create table usuario
15. (id_usuario varchar2(24) primary key,
16.  nombre varchar2(24),
17.  apellido1 varchar2(24),
18.  apellido2 varchar2(24),
19.  fecha_nacimiento date,
20.  sexo char(1),
```

```
21.      direccion varchar2(64),
22.      localidad varchar2(32),
23.      provincia varchar2(32),
24.      postal number(5),
25.      telefono_movil number(9),
26.      facebook varchar2(64),
27.      twitter varchar2(64),
28.      fecha_alta date not null,
29.      fecha_baja date,
30.      estado char(1) not null,
31.
32.      /*El sexo del usuario puede tener uno de los tres
valores diferentes (I, H, M)*/
33.      constraint usuario_sexo_ck check (sexo in ('I',
'H', 'M')),
34.
35.      /*El estado del usuario puede tomar el valor 0 o 1
*/
36.      constraint usuario_estado_ck check (estado in ('0',
'1')));
37.
38.      /*Se crea la tabla ESTABLECIMIENTO */
39.      create table establecimiento
40.      (id_shop number(9) primary key,
41.      nombre varchar2(24) not null,
42.      direccion varchar2(64),
43.      localidad varchar2(32),
44.      provincia varchar2(32),
45.      postal number(5),
46.      telefono number(9),
47.      latitud varchar2(16),
48.      longitud varchar2(16),
49.      fecha_alta date not null,
50.      fecha_baja date,
51.      estado char(1) not null,
52.
53.      /*El estado del establecimiento puede tomar el
valor 0 o 1 */
54.      constraint establecimiento_estado_ck check (estado
in ('0', '1')));
55.
56.      /*Se crea la tabla PRODUCTOS*/
57.      create table producto
58.      (id_producto number(9) primary key,
59.      nombre varchar2(24) not null,
60.      descripcion varchar2(64) not null,
61.      detalle varchar2(512) not null,
62.      foto varchar2(512) not null,
63.      coste number(14, 2) not null,
64.      venta number(14, 2) not null,
65.      fecha_alta date not null,
66.      fecha_baja date,
67.      estado char(1) not null,
68.
```



```
69.      /*El estado del producto puede tomar el valor 0 o 1
70.      */
71.      constraint producto_estado_ck check (estado in
72.      ('0', '1')));
73.
74.      /*Se crea la tabla OFERTAS */
75.      create table oferta
76.      (id_oferta number(9) primary key,
77.      nombre varchar2(24) not null,
78.      descripcion varchar2(64) not null,
79.      detalle varchar2(512) not null,
80.      foto varchar2(512) not null,
81.      venta number(14, 2) not null,
82.      fecha_inicio date not null,
83.      fecha_fin date,
84.      fecha_alta date not null,
85.      fecha_baja date,
86.      estado char(1) not null,
87.
88.      /*El estado de la oferta puede tomar el valor 0 o 1
89.      */
90.      constraint oferta_estado_ck check (estado in ('0',
91.      '1')));
92.
93.      /*Se crea la tabla OFERTA_ESTABLECIMIENTOS */
94.      create table oferta_establecimiento
95.      (id_of_es number(9) primary key,
96.      id_oferta number(9),
97.      id_shop number(9),
98.      constraint of_es_id_of_fk foreign key (id_oferta)
99.      references oferta(id_oferta),
100.      constraint of_es_id_es_fk foreign key (id_shop)
101.      references establecimiento(id_shop));
102.
103.      /*Se crea la tabla OFERTA_PRODUCTOS */
104.      create table oferta_producto
105.      (id_of_pr number(9) primary key,
106.      id_oferta number(9),
107.      id_producto number(9),
108.      venta number(14, 2) not null,
109.      constraint of_pr_id_of_fk foreign key (id_oferta)
110.      references oferta(id_oferta),
111.      constraint of_pr_id_pr_fk foreign key (id_producto)
112.      references producto(id_producto));
113.
114.      /*Se crea la tabla OFERTA_USUARIOS */
115.      create table oferta_usuario
116.      (id_of_us number(9) primary key,
117.      id_oferta number(9),
118.      id_usuario varchar2(24),
119.      fecha_envio date not null,
120.      estado_envio char(1) not null,
121.      fecha_lectura date,
```

```
114.     fecha_uso date,
115.     venta number(14, 2),
116.     constraint of_us_id_of_fk foreign key (id_oferta)
       references oferta(id_oferta),
117.     constraint of_us_id_us_fk foreign key (id_usuario)
       references usuario(id_usuario));
118.
119.     /*Se crea la tabla USUARIO_PRODUCTO */
120.     create table usuario_producto
121.     (id_us_pr number(9) primary key,
122.     id_of_us number(9),
123.     id_producto number(9),
124.     unidades number(9) not null,
125.     venta number(14, 2) not null,
126.     constraint us_pr_id_of_us_fk foreign key (id_of_us)
       references oferta_usuario(id_of_us),
127.     constraint us_pr_id_pr_fk foreign key (id_producto)
       references producto(id_producto));
128.
129.     /*Se crea la tabla USUARIO_REDES */
130.     create table usuario_redes
131.     (id_redes number(9) primary key,
132.     id_usuario varchar2(24),
133.     id_oferta number(9),
134.     fecha_accion date not null,
135.     tipo_red char(1) not null,
136.     tipo_accion char(1) not null,
137.     seguimiento number(12),
138.     constraint us_re_id_us_fk foreign key (id_usuario)
       references usuario(id_usuario),
139.     constraint us_re_id_of_fk foreign key (id_oferta)
       references oferta(id_oferta),
140.     constraint us_re_ti_re_ck check (tipo_red in ('0',
       '1')),
141.     constraint us_re_ti_ac_ck check (tipo_accion in
       ('0', '1', '2', '3')));
```

## 5.2. Construcción de scripts de carga de los datos.

```
1. /*Inserta establecimientos en las tablas */
2. insert into establecimiento
3. values ('0000', 'BRILLANTE', 'BRILLANTE', 'CORDOBA',
       'CORDOBA', '14001', '957465935', '15', '20', '10-11-2014',
       NULL, '1');
4. insert into establecimiento
5. values ('0001', 'MEZQUITA', 'MEZQUITA', 'CORDOBA',
       'CORDOBA', '14014', '957474545', '15', '21', '08-07-2014',
       NULL, '1');
6. insert into establecimiento
7. values ('0002', 'PATIOS', 'SERRANO', 'CORDOBA', 'CORDOBA',
       '142213', '957433935', '17', '19', '19-12-2014', NULL,
       '1');
```

```
8.
9. /*Inserta productos en las tablas */
10.     insert into producto
11.     values ('0000', 'CARBONARA', 'RICA', 'NATA',
12.     'http://www.pizzeta.es/producto0000.png', '4', '9', '10-
13.     04-2014', NULL, '1');
14.     insert into producto
15.     values ('0001', 'BARBACOA', 'CRUJIENTE', 'SALSA',
16.     'http://www.pizzeta.es/producto0001.png', '4', '9', '10-
17.     04-2014', NULL, '1');
18.     insert into producto
19.     values ('0002', 'MARGARITA', 'TÍPICA', 'TOMATE',
20.     'http://www.pizzeta.es/producto0002.png', '4', '9', '10-
21.     04-2014', NULL, '1');
22.
23. /*Inserta ofertas en las tablas */
24.     insert into oferta
25.     values ('0000', '2x1', 'REBAJA', 'LUNES',
26.     'http://www.pizzeta.es/oferta0000.png', '9', '10-04-2014',
27.     '10-05-2014', '08-04-2014', NULL, '1');
28.     insert into oferta
29.     values ('0001', '3x1', 'REBAJA', 'MARTES',
30.     'http://www.pizzeta.es/oferta0001.png', '9', '10-08-2014',
31.     '10-10-2014', '08-04-2014', NULL, '1');
32.     insert into oferta
33.     values ('0002', 'REGALO', 'PIZZA', 'ANIVERSARIO',
34.     'http://www.pizzeta.es/oferta0002.png', '0', '10-02-2014',
35.     '11-02-2014', '08-01-2014', NULL, '1');
36.
37. /*Inserta usuarios en las tablas */
38.     insert into usuario
39.     values ('U0000', 'PEDRO', 'CASTRO', 'ROMERO',
40.     SYSDATE-365*25, 'M', 'CRUCES', 4,
41.     'MALAGA', 'MALAGA', 29007, 697123456, 'ACC',
42.     '@ACC', SYSDATE-410, NULL, '1');
43.     insert into usuario
44.     values ('U0001', 'ANTONIO', 'MERINO', 'LUQUE',
45.     SYSDATE-365*22, 'H', 'CARRETERO', 4,
46.     'MALAGA', 'MALAGA', 29003, 677123456, 'JCC',
47.     '@JCC', SYSDATE-400, NULL, '1');
48.     insert into usuario
49.     values ('U0002', 'FRANCISCA', 'LEAL', 'VILLATORO',
50.     SYSDATE-365*19-100, 'H', 'GONDOMAR', 11,
51.     'CORDOBA', 'CORDOBA', 14003, 617123456, 'ASG',
52.     '@ASG', SYSDATE-390, NULL, '1');
53.     insert into usuario
54.     values ('U0003', 'VANESA', 'CRUZ', 'PORTILLO',
55.     SYSDATE-365*39-33, 'M', 'REAL', 21,
56.     'CASTRO DEL RIO', 'CORDOBA', 14321, 677112233,
57.     NULL, NULL, SYSDATE-380, NULL, '1');
58.     insert into usuario
59.     values ('U0004', 'KLEBER', 'VALDIVIESO', 'FAJARDO',
60.     SYSDATE-365*27-45, 'H', 'CLAVELES', 2,
```

```
40.      'RONDA', 'MALAGA', 29207, 677223344, 'ECT',
      '@ECT', SYSDATE-370, SYSDATE-20, '0');
41.      insert into usuario
42.      values ('U0005', 'ANTONIO', 'CAZALLA', 'REYES',
      SYSDATE-365*27, 'M', 'ALDERETES, 15',
43.      'CORDOBA', 'CORDOBA', 14004, 677334455, 'SHV',
      '@SHV', SYSDATE-360, NULL, '1');
44.      insert into usuario
45.      values ('U0006', 'ROBERTO', 'GALLARDO', 'MORENO',
      SYSDATE-365*19-27, 'H', 'QUESO, 2',
46.      'CORDOBA', 'CORDOBA', 14008, 677445566, 'JGL',
      '@JGL', SYSDATE-350, SYSDATE-121, '0');
47.      insert into usuario
48.      values ('U0007', 'JOAQUINA', 'SERRANO', 'MONEDERO',
      SYSDATE-365*28-12, 'M', 'QUINTANAR, 11',
49.      'POSADAS', 'CORDOBA', 14302, 677556677, 'JMBM',
      '@JMBM', SYSDATE-340, SYSDATE-25, '0');
50.      insert into usuario
51.      values ('U0008', 'GASPAR', 'REYES', 'MELERO',
      SYSDATE-365*19-65, 'H', 'PALOMAS, 22',
52.      'ANTEQUERA', 'MALAGA', 29220, 677667788, NULL,
      NULL, SYSDATE-330, NULL, '1');
53.      insert into usuario
54.      values ('U0009', 'DAVID', 'BISBAL', 'CONTRERAS',
      SYSDATE-365*44, 'M', 'MADRID, 33',
55.      'BAEZA', 'JAEN', 23403, 722112233, 'SMB',
      '@SMB', SYSDATE-320, SYSDATE-79, '0');
56.      insert into usuario
57.      values ('U1010', 'DAVID', 'BUSTAMANTE', 'MORA',
      SYSDATE-365*24-100, 'H', 'BARRILES, 4',
58.      'BAEZA', 'JAEN', 23403, 677778899, 'MCC',
      '@MCC', SYSDATE-310, NULL, '1');
59.      insert into usuario
60.      values ('U1011', 'RISTO', 'MEJIDE', 'HURTADO',
      SYSDATE-365*21, 'M', NULL,
61.      'BAEZA', 'JAEN', 23403, 611987654, 'MCH',
      '@MCH', SYSDATE-300, SYSDATE-2, '0');
62.      insert into usuario
63.      values ('U1012', 'JUANA', 'RIVAS', 'NAVAS', SYSDATE-
      365*33-33, 'H', 'QUINTANAR, 7',
64.      'JAEN', 'JAEN', 23006, 611111111, 'VLO', '@VLO',
      SYSDATE-290, NULL, '1');
65.      insert into usuario
66.      values ('U1013', 'CARLA', 'BRUNI', 'ARBELOA',
      SYSDATE-365*25-25, 'M', NULL,
67.      NULL, NULL, NULL, 677123456, NULL, NULL,
      SYSDATE-280, NULL, '1');
68.      insert into usuario
69.      values ('U1014', 'ALBERTO', 'HENS', 'MAGDALENO',
      SYSDATE-365*34-14, 'H', 'BASTIDORES, 6',
70.      'JAEN', 'JAEN', 23006, 6112222, 'ILI', '@ILI',
      SYSDATE-270, NULL, '1');
```

## 6. Manipulación de la Base de Datos.

1. SELECT \* FROM establecimiento WHERE provincia = "CORDOBA"  
ORDER BY nombre ASC
2. SELECT \* FROM usuario WHERE sexo = "H" ORDER BY nomber ASC
3. SELECT nombre, apellido1, apellido2, telefono\_movil WHERE  
provincia = "CORDOBA" ORDER BY nombre ASC
4. DELETE FROM establecimiento WHERE postal = "14011"
5. DELETE FROM producto WHERE coste > "15"
6. DELETE FROM usuario WHERE provincia = "MALAGA"
7. UPDATE producto SET coste = coste - 2\*coste/100 WHERE venta <  
10
8. UPDATE usuario SET localidad = "PRIEGO" WHERE id\_usuario =  
"U0005"
9. UPDATE establecimiento SET estado = "1" WHERE provincia =  
"CORDOBA"