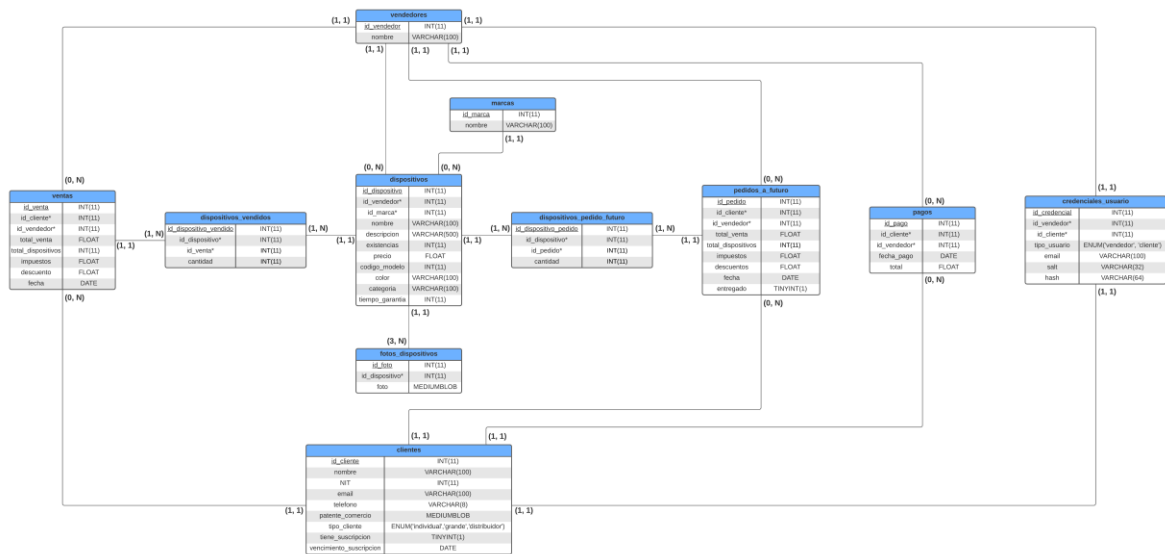


Diagrama Entidad-Relación



Tablas del sistema de ventas

Dispositivos		
Nombre	Tipo	Descripción
id_dispositivo	int	Llave foránea del dispositivo.
id_vendedor	int	Llave foránea que identifica a cuál vendedor pertenece el dispositivo.
id_marca	int	Llave foránea de la marca del dispositivo.
nombre	varchar(100)	Nombre del dispositivo.
descripcion	varchar(500)	Descripción del dispositivo.
existencias	int	Cantidad del dispositivo en existencia en bodega.
precio	float	Precio de lista del dispositivo.
codigo_modelo	int	Código del modelo del dispositivo.
color	varchar(100)	Color del dispositivo.
categoria	varchar(100)	Categoría del dispositivo.
tiempo_garantia	int	Años/meses/días de garantía.

Tabla con información de los dispositivos.

Fotografías_dispositivos		
Nombre	Tipo	Descripción
id_fotografia	int	Llave primaria de la fotografía.
id_dispositivo	int	Llave foránea que identifica al dispositivo asociado con la fotografía.
foto	medblob	Fotografía del dispositivo.

Tabla con las fotografías de los dispositivos.

<i>Marcas</i>		
<i>Nombre</i>	<i>Tipo</i>	<i>Descripción</i>
id_marca	int	Llave primaria de la marca.
nombre	varchar(100)	Nombre de la marca.

Tabla del catálogo de marcas.

<i>Vendedores</i>		
<i>Nombre</i>	<i>Tipo</i>	<i>Descripción</i>
id_vendedor	int	Llave primaria.
nombre	varchar(100)	Nombre de la tienda/vendedor.

Tabla con los datos de los vendedores.

<i>Clientes</i>		
<i>Nombre</i>	<i>Tipo</i>	<i>Descripción</i>
id_cliente	int	Llave primaria.
nombre	varchar(100)	Nombre del cliente.
nit	int	NIT del gran cliente o distribuidor mayorista. (Nullable).
Email	Varchar(100)	Email del gran cliente o distribuidor mayorista. Puede ser distinto al email de las credenciales. (Nullable).
telefono	varchar(8)	Teléfono del gran cliente o distribuidor mayorista. (Nullable).
patente_de_comercio	medblob	Patente de comercio (imagen) del gran cliente o distribuidor mayorista. (Nullable).
tipo_cliente	enum	Tipo del cliente (individual, gran cliente o distribuidor mayorista).
tiene_suscripcion	boolean	Indica si tiene o no suscripción.
vencimiento_suscripcion	date	Fecha del vencimiento de la suscripción. (Nullable).

Tabla con los datos de los clientes.

<i>Ventas</i>		
<i>Nombre</i>	<i>Tipo</i>	<i>Descripción</i>
id_venta	int	Llave primaria
id_cliente	int	Llave foránea del cliente que compró los dispositivos.
id_vendedor	int	Llave foránea del vendedor.
total_venta	float	Total facturado al cliente.
total_dispositivos	int	Total de dispositivos vendidos.
impuestos	float	Total de impuestos.
descuentos	float	Descuento aplicados a la venta.
fecha	date	Fecha en que se realizó la venta.

Tabla con la información de las ventas realizadas.

Dispositivos_vendidos		
<i>Nombre</i>	<i>Tipo</i>	<i>Descripción</i>
id_dispositivo_vendido	int	Llave primaria.
id_venta	int	Llave foránea de la venta.
id_dispositivo	int	Llave foránea del dispositivo.
cantidad	Int	Cantidad de dispositivos vendidos.

Tabla con los datos de los dispositivos adquiridos en una venta.

Pedidos_a_futuro		
<i>Nombre</i>	<i>Tipo</i>	<i>Descripción</i>
id_pedido	int	Llave primaria del pedido.
id_cliente	int	Llave foránea del distribuidor que adquirirá los dispositivos.
id_vendedor	int	Llave foránea del vendedor.
total_venta	float	Total facturado al cliente.
total_dispositivos	int	Total de dispositivos vendidos.
impuestos	float	Total de impuestos.
descuentos	float	Descuento aplicados a la venta.
fecha	date	Fecha en que se realizó el pedido.
entrega	boolean	Indica si el pedido ya ha sido entregado.

Tabla con la información de los pedidos a futuro de los distribuidores mayoristas.

Dispositivos_pedido_futuro		
<i>Nombre</i>	<i>Tipo</i>	<i>Descripción</i>
id_disp_pedido	int	Llave primaria.
id_dispositivo	int	Llave foránea del dispositivo.
id_pedido	int	Llave foránea del pedido a futuro.
cantidad	int	Cantidad del dispositivo adquirido en el pedido a futuro.

Tabla con la información de los dispositivos que serán adquiridos en un pedido futuro.

Pagos		
<i>Nombre</i>	<i>Tipo</i>	<i>Descripción</i>
id_pago	int	Llave primaria del pago
id_cliente	int	Llave foránea del cliente.
id_vendedor	int	Llave foránea del vendedor.
fecha_pago	date	Fecha en que se realizó el pago.
total	float	Total pagado.

Tabla con la información de los pagos.

<i>Credenciales_usuarios</i>		
<i>Nombre</i>	<i>Tipo</i>	<i>Descripción</i>
id_usuario	int	Llave primaria del usuario.
id_vendedor	int	Llave foránea del vendedor (null si el usuario es cliente).
id_cliente	int	Llave foránea del cliente (null si el usuario es vendedor).
tipo_usuario	boolean/enum	Tipo del usuario (vendedor o cliente). Dato para poder dirigirlo a la página correspondiente.
email	varchar(100)	Email del usuario utilizado para el login.
salt	varchar(32)	Sal criptográfica.
hash	varchar(64)	Digest resultado de hashear la concatenación de contraseña y sal.

Tabla con la información de los usuarios.

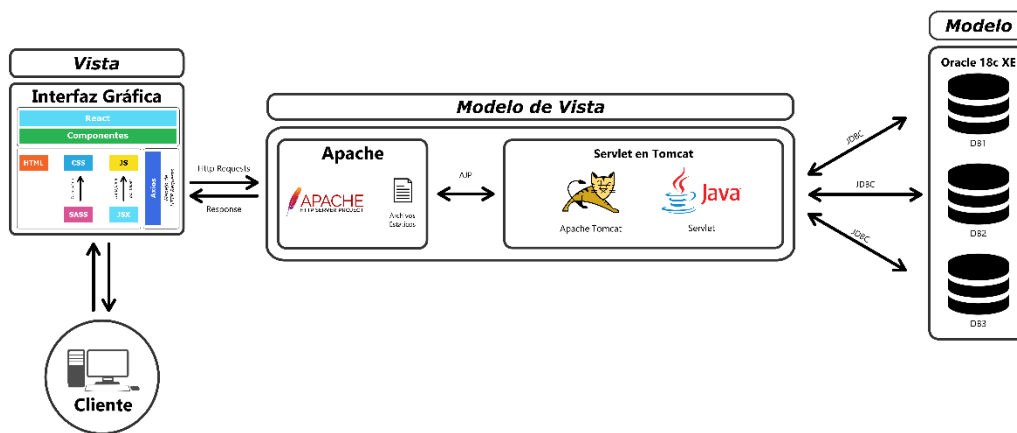
Propuesta de objetos de la base de datos:

- Vistas
 - Vista que muestre los dispositivos en existencia o que no lo están.
 - Vistas para mostrar los tipos de clientes:
 - Una vista que muestre a los clientes individuales.
 - Una vista que muestre a los grandes clientes.
 - Una vista que muestre a los distribuidores mayoristas.
 - Vista que muestre los pedidos ya entregados o aún no entregados.
- Función
 - Función que indica si un dispositivo todavía está cubierto por la garantía a partir del id del dispositivo y de la fecha en que se adquirió.
- Procedimiento almacenado
 - Procedimiento almacenado que despliegue todas las fotografías de un dispositivo usando como parámetro el id del dispositivo.
 - Como todas las fotos pertenecen a guardan en una sola tabla, de esta forma se pueden obtener todas aquellas correspondientes a un dispositivo en específico.
- Trigger
 - Trigger que cuando se agregue un producto a un pedido a futuro modifique el total de productos, así como el total por pagar.
 - De forma similar, si se elimina un producto del pedido o se modifica la cantidad, que existan otros dos triggers, uno por update y otro por delete que hagan lo correspondiente dependiendo de la acción.

Stack de tecnologías por utilizar

- Front-end: **React**
- Sistema de fábrica (MERN stack):
 - **MongoDB**: como base de datos no relacional de documentos.
 - **Express.js**: como framework web de Node.js.
 - **Node.js**: como entorno del backend.
- Sistema de ventas:
 - **Oracle Database 18c XE**: como base de datos relacional.
 - **Apache HTTP Server**: como servidor web de archivos estáticos para la entrega de archivos de JSP, actuando como proxy de las peticiones del servlet desplegado en Tomcat.
 - **Tomcat**: como servidor de aplicaciones.
 - **Servlet**: pequeño programa de Java servido por Tomcat. Estará configurado para recibir y responder los requests HTTP, a la vez que gestiona las operaciones CRUD en la base de datos.
- Protocolos/conectores de comunicación:
 - **AJP**: (Apache JServ Protocol). protocolo que permite el envío de solicitudes desde un servidor web, como Apache, a un servidor de aplicaciones, como Tomcat.
 - **JDBC**: (Java Database Connectivity) es un estándar de Java que proporciona la interfaz para conectarse desde Java a bases de datos relaciones, como Oracle 18c XE.

Diagrama de arquitectura del sistema de ventas



Arquitectura del Sistema de Ventas

Screenshots del sistema corriendo

```
[azureuser@CentOS8 ~]$ neofetch

..
.PLTJ.
<<<<<<>
KKSSV' 4KKK LJ KKKL.'VSSKK
KKV' 4KKKKK LJ KKKKAL 'VKK
V' ' 'VKKKK LJ KKKKV' ' 'V
.4MA.' 'VKK LJ KKV' '.4Mb.
. KKKKA.' 'V LJ V' '.4KKKKK .
.4D KKKKKKKK.' LJ ' '.4KKKKKKK FA.
<QDD ++++++++ GFD>
'VD KKKKKKKK'.. LJ ..'KKKKKKKK FV
'VKKKK'. .4 LJ K. 'KKKKKV '
'VK'. .4KK LJ KKA. . 'KV'
A. . .4KKK LJ KKKKA. . .4
KKA. 'KKKKK LJ KKKK' .4KK
KKSSA. VKKK LJ KKKV .4SSKK

<<<<<<>
'MKKM'
..

azureuser@CentOS8.5-Azure
-----
OS: CentOS Linux 8 x86_64
Host: Virtual Machine 7.0
Kernel: 4.18.0-348.7.1.el8_5.x86_64
Uptime: 8 mins
Packages: 683 (rpm)
Shell: bash 4.4.20
Terminal: /dev/pts/0
CPU: Intel Xeon E5-2673 v4 (2) @ 2.294GHz
GPU: 00:08.0 Microsoft Corporation Hyper-V virtual VGA
Memory: 2367MiB / 3736MiB
```

Se usará CentOS como distribución del Linux

```
SQL*Plus: Release 18.0.0.0.0 - Production on Sat Feb 19 11:41:27 2022
Version 18.4.0.0.0

Copyright (c) 1982, 2018, Oracle. All rights reserved.

Last Successful login time: Fri Feb 18 2022 12:46:20 -06:00

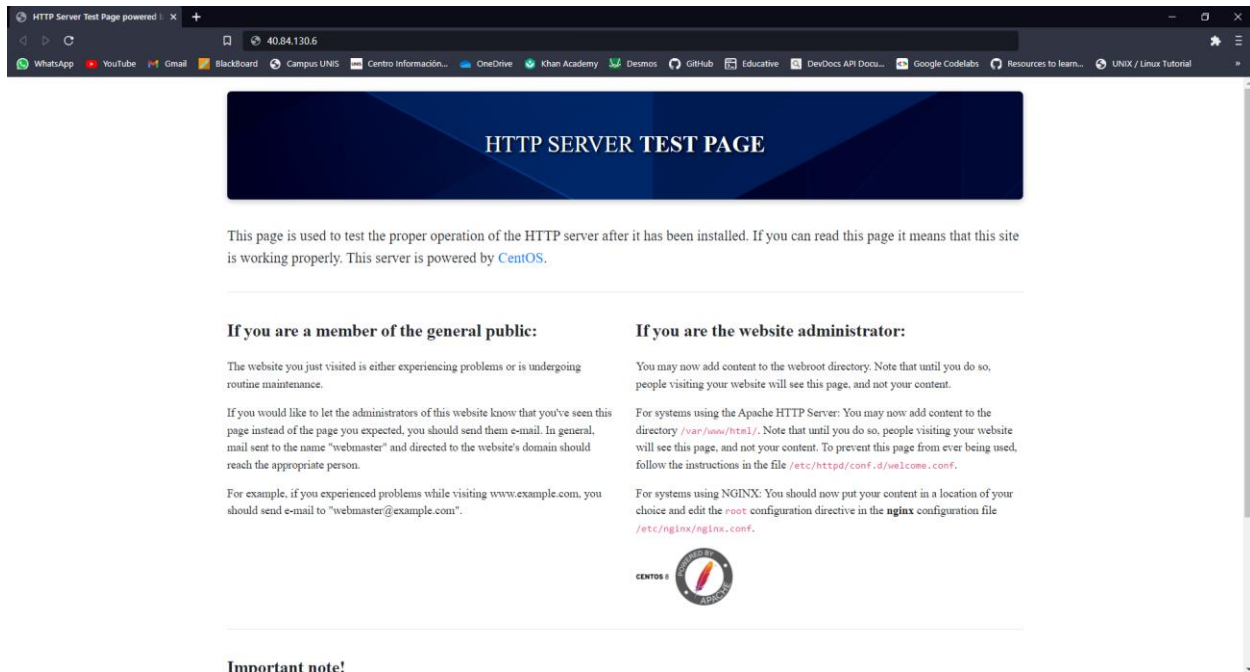
Connected to:
Oracle Database 18c Express Edition Release 18.0.0.0.0 - Production
Version 18.4.0.0.0

SQL>
```

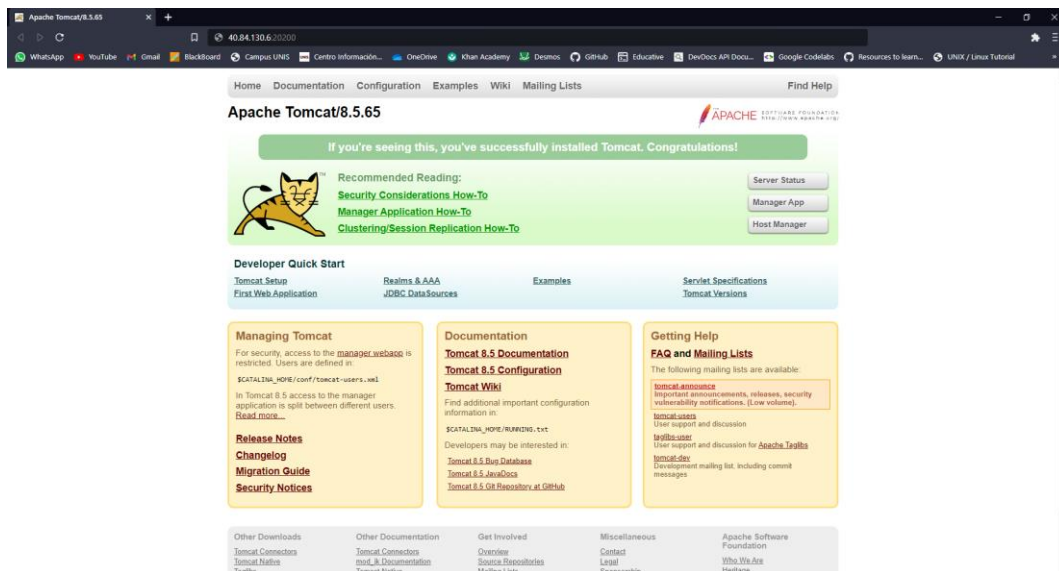
Conexión a Oracle Database 18c XE mediante SQL Plus

ID	DESCRIPCION	EXISTENCIAS	PRECIO	CODIGO	MODELO	COLOR	CATEGORIA	TIEMPO_GARANTIA
1	Smart TV	250	2700	2045	Smart	Blanco	Utilidades	1
2	Smart TV	0	3500	1245	Smart	Blanco	Utilidades	2
3	Computadora	20	6700	1324	Smart	Blanco	Utilidades	1
4	Smart TV	12	3750	6700	Smart	Blanco	Utilidades	2
5	Smart TV	12	2615	3600	Smart	Blanco	Utilidades	1
6	Smart TV	241	5700	3450	Smart	Blanco	Utilidades	1
7	Smart TV	145	5450	3600	Smart	Blanco	Utilidades	1
8	Smart TV	0	6700	3245	Smart	Blanco	Utilidades	1
9	Smart TV	22	6900	3225	Smart	Blanco	Utilidades	1
10	Smart TV	222	3750	3600	Smart	Blanco	Utilidades	1
11	Smart TV	124	3600	3245	Smart	Blanco	Utilidades	1
12	Smart TV	51	3225	4400	Smart	Blanco	Utilidades	1
13	Smart TV	210	5450	3600	Smart	Blanco	Utilidades	1
14	Smart TV	105	3600	3245	Smart	Blanco	Utilidades	1
15	Smart TV	27	6750	3225	Smart	Blanco	Utilidades	1
16	Smart TV	0	3700	3450	Smart	Blanco	Utilidades	1
17	Smart TV	25	7400	3570	Smart	Blanco	Utilidades	2
18	Smart TV	241	2200	6700	Smart	Blanco	Utilidades	1

Conexión a Oracle Database 18c XE mediante Oracle SQL Developer



Conexión a Apache HTTP



Conexión a Tomcat

```

[azureuser@CentOS8 ~]$ systemctl status mongod
● mongod.service - MongoDB Database Server
   Loaded: loaded (/usr/lib/systemd/system/mongod.service; enabled; vendor preset: disabled)
   Active: active (running) since Sat 2022-02-19 17:42:02 UTC; 17min ago
     Docs: https://docs.mongodb.org/manual
    Main PID: 948 (mongod)
      Memory: 178.4M
    CGroup: /system.slice/mongod.service
            └─948 /usr/bin/mongod -f /etc/mongod.conf

Feb 19 17:41:51 CentOS8.5-Azure systemd[1]: Starting MongoDB Database Server...
Feb 19 17:41:53 CentOS8.5-Azure mongod[887]: about to fork child process, waiting until server is ready for connections.
Feb 19 17:41:54 CentOS8.5-Azure mongod[887]: forked process: 948
Feb 19 17:42:02 CentOS8.5-Azure mongod[887]: child process started successfully, parent exiting
Feb 19 17:42:02 CentOS8.5-Azure systemd[1]: Started MongoDB Database Server.
[azureuser@CentOS8 ~]$ mongo
MongoDB shell version v4.2.18
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongod
Implicit session: session { "id" : UUID("22da89a7-367f-4e5b-b26d-c7afeed326f4") }
MongoDB server version: 4.2.18
Server has startup warnings:
2022-02-19T17:42:01.837+0000 I CONTROL [initandlisten]
2022-02-19T17:42:01.837+0000 I CONTROL [initandlisten] ** WARNING: Access control is not enabled for the database.
2022-02-19T17:42:01.837+0000 I CONTROL [initandlisten] ** Read and write access to data and configuration is unrestricted.
2022-02-19T17:42:01.837+0000 I CONTROL [initandlisten]
---
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).

The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---
> |

```

Verificación de MongoDB corriendo