

# **Proyecto Sistema de Facturación**

**Presentado por:**

**Héctor Josué Ardón Morga**

**Dirigido a:**

**Drogueria Farinter**

## Introducción

Un sistema de facturación es algo que es bastante útil para las empresas porque ayuda a administrar información de forma ordenada sobre las cuentas y sobre las entradas y salidas de dinero, de esa forma aprendemos a administrar mejor las finanzas de nuestra empresa

## Herramientas utilizadas

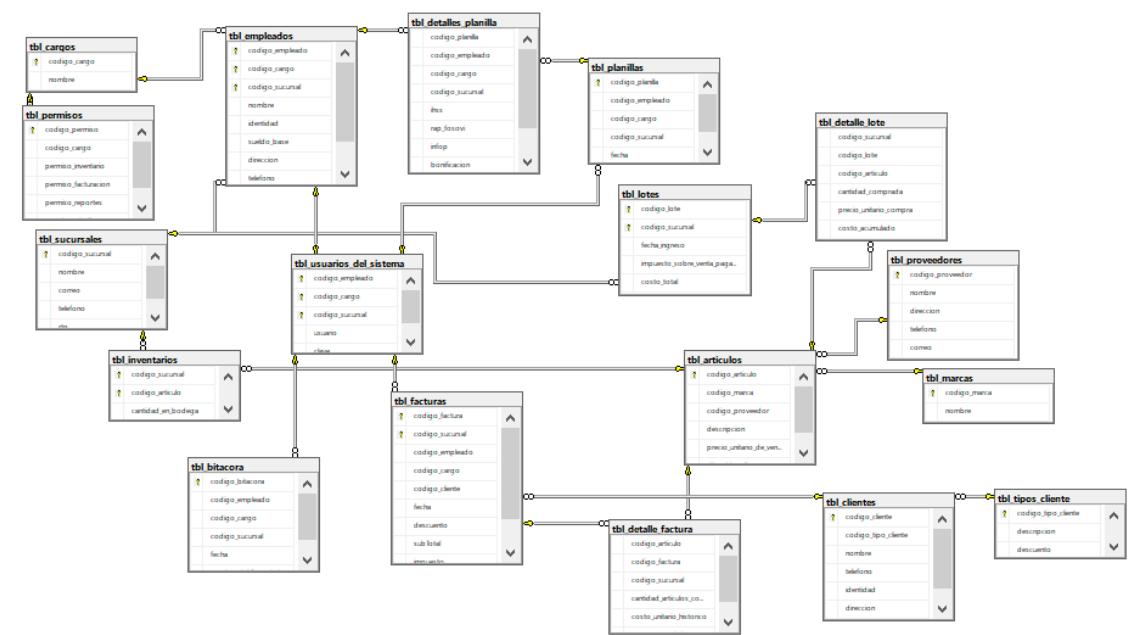
Base de datos: SQL Server

Front End: React JS

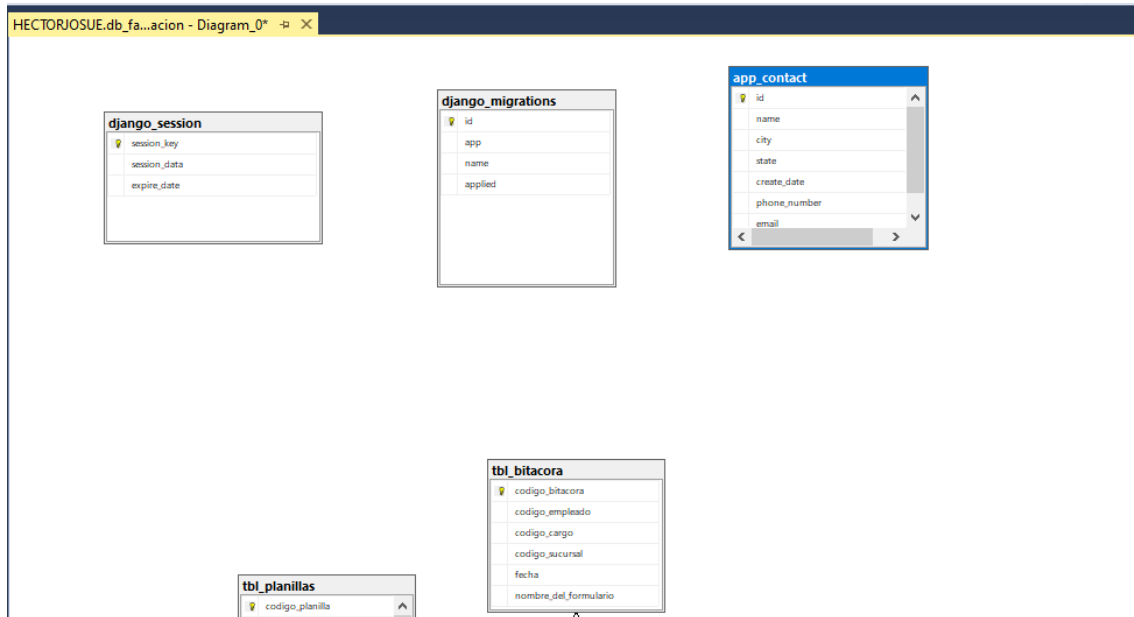
Back End: Django (Python)

## Modelo de base de datos

Este es el modelo de base de datos que se utilizó para este proyecto, está hecho en SQL Server



Cuando se intento hacer la conexión con la base de datos, creo estas tablas de forma automatica



Aquí en el archivo settings.py cambiar el nombre del Host por el de la empresa, el usuario y el password

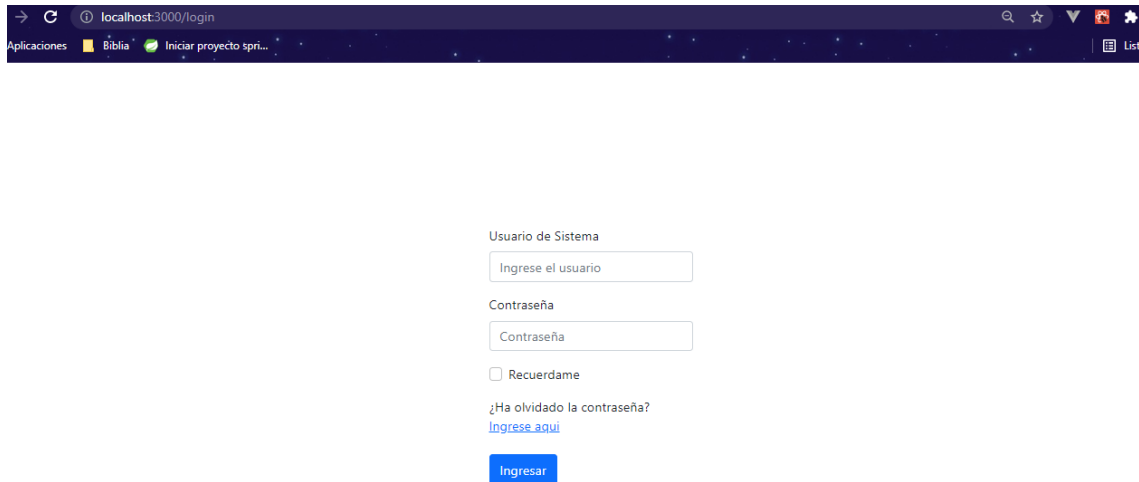
```
66     },
67 },
68 },
69 ]
70
71 WSGI_APPLICATION = 'proyecto.wsgi.application'
72
73
74 # Database
75 # https://docs.djangoproject.com/en/3.0/ref/settings/#databases
76
77 DATABASES = {
78     'default': {
79         'ENGINE': 'sql_server.pyodbc',
80         'NAME': 'db_facturacion',
81         'USER': '',
82         'PASSWORD': '',
83         'HOST': 'HECTORJOSUE',
84         'PORT': '',
85         'OPTIONS': {
86             'driver': 'ODBC Driver 13 for SQL Server',
87         },
88     },
89 }
90
```

Descargar el driver ODBC Driver, este driver es necesario para conectar SQL Server con la aplicación Django, de este Link

<https://www.microsoft.com/en-us/download/details.aspx?id=50420>

## Pantallas diseñadas aplicación 1 con React JS

### Login



Usuario de Sistema

Contraseña

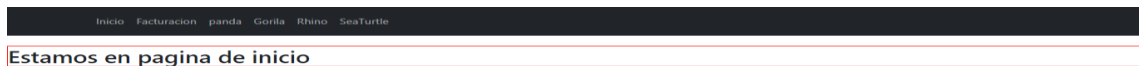
☐ Recuerdame

¿Ha olvidado la contraseña?  
[Ingrese aqui](#)

---

### Bienvenida

El objetivo aquí es darle la bienvenida al usuario, podría mostrarse el perfil del usuario



---

Aquí mostramos una pantalla para lo que es la facturación, el buscador, busca productos en la base de datos, al seleccionar un producto debe mostrar su información en el recuadro datos del producto que está encerrado en rojo y un campo para indicar cuantas unidades de ese producto desea llevar, en la tabla

de abajo que dice facturación se van a ir agregando todos los productos que se añadan con el botón agregar producto

InicioFacturaciónpandaGorilaRhinoSeaTurtle

Nos encontramos en Facturación de la compañía X

Datos del producto

Código del artículo:XXXXXX

Descripción:XXXXXX

Precio unitario de venta:XXXXXX

Días de vida útil:XXXXXX

Marca:XXXXXX

Cantidad en Bodega:XXXXXX

Cantidad de productos al carrito:

Eliminar producto

Agregar producto

Buscar producto en

Q

Productos disponibles:

Código del pro...	Descripción	Precio	Días de vida útil	Marca	Cantidad en bo...
	producto ingresa...			marca ingresada 1	
	producto ingresa...			marca ingresada 2	

Filas por página: 101-2 de 2<>>

Facturación

Código de...	Descripción	Marca	Precio Uni...	Cantidad a
	producto numero 1	marca#1		
	producto numero 2	marca#2		

Datos del cliente

Nombre:

Identidad:

SUB-TOTAL:XXXXXX

La idea es lograr diseñar algo similar a esta pantalla, pero en react js

Sistema de Información Ferretería X --- [SUCURSAL A] --- [JUAN CARDONA]

SesiónMódulosVentanaAyuda

Facturación

FERRETERÍA X

Facturación

Datos del Producto

CÓDIGO ARTÍCULO:

DESCRIPCIÓN:

PRECIO UNITARIO DE VENTA:

DÍAS DE VIDA ÚTIL:

MARCA:

CANTIDAD:

Buscar:

Buscar

CÓDIGO	DESCRIPCIÓN	PRECIO	DÍAS DE VIDA	MARCA	CANTIDAD
64	Bata de uña	325		Truper	105
34	Batería 12V	68		Duracell	52
7	Batería 1.5V	8		Rayovac	118
8	Batería 9V	42		Eveready	142

Facturación:

CÓDIGO ARTÍCULO	DESCRIPCIÓN	MARCA	PRECIO UNITARIO	CANTIDAD	COSTO ACUMULADO
-----------------	-------------	-------	-----------------	----------	-----------------

DATOS DEL CLIENTE

IDENTIDAD:

NOMBRE:

SUB-TOTAL

15% ISV

DESCUENTO:

TOTAL:

Agregar Producto

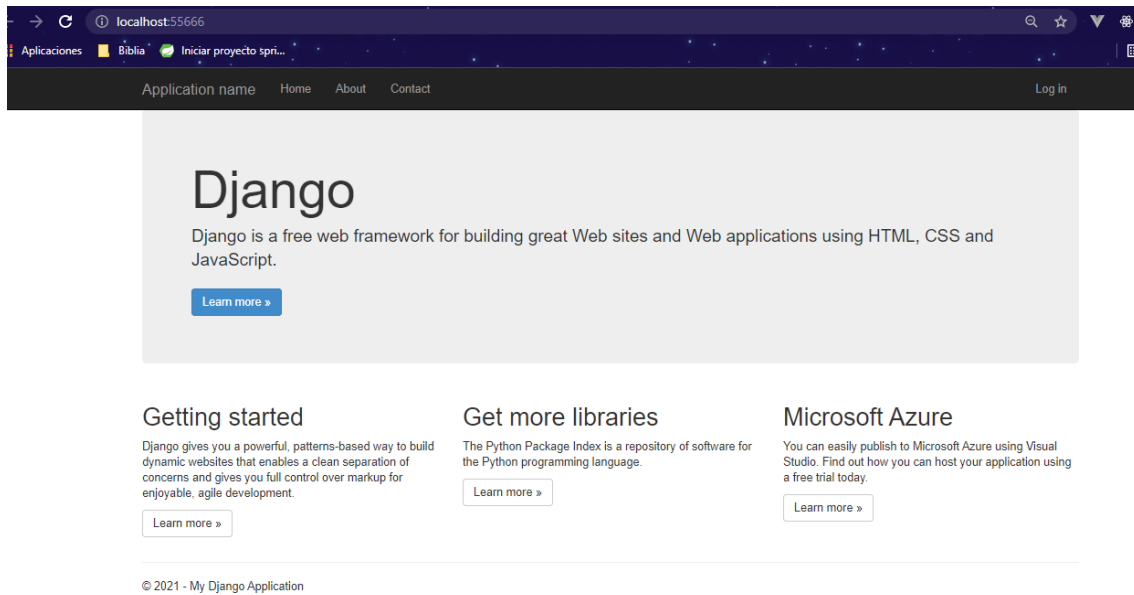
Eliminar Producto

Facturar

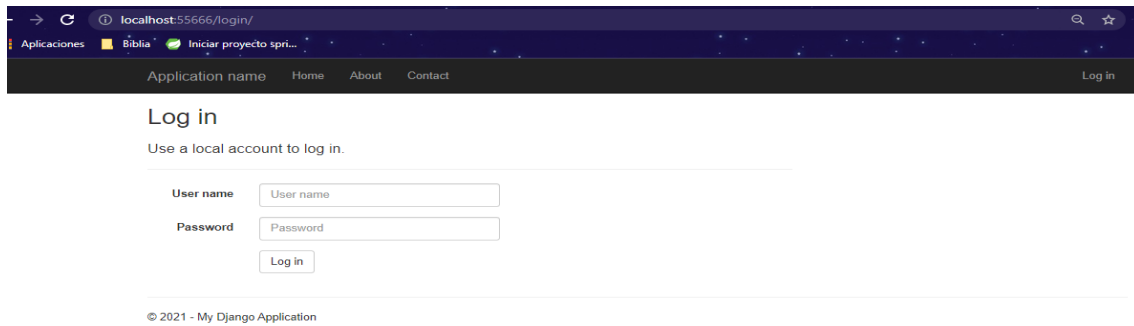
## Pantalla aplicación 2

Se hizo de otra forma la conexión a la base de datos utilizando el IDE de visual studio, como resultado se obtuvieron algunas pantallas que ya crea por defecto el IDE

## Landing page:



## Login:



## About:

## About.

### Your application description page.

Use this area to provide additional information.

---

© 2021 - My Django Application

#### Contact:

## Contact.

### Your contact page.

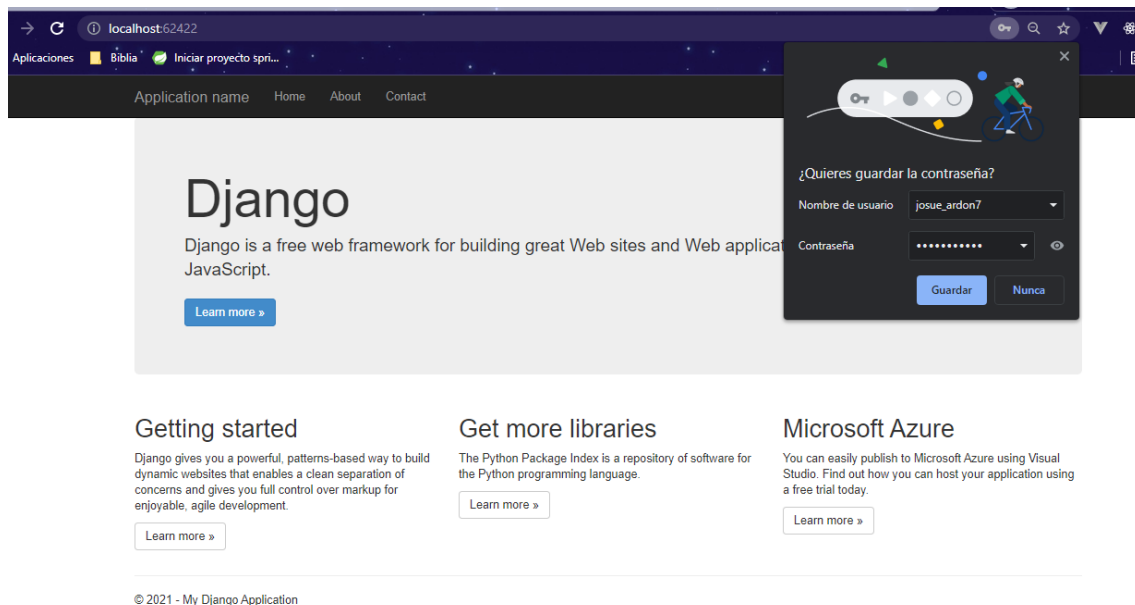
One Microsoft Way  
Redmond, WA 98052-6399  
P: 425.555.0100

**Support:** [Support@example.com](mailto:Support@example.com)  
**Marketing:** [Marketing@example.com](mailto:Marketing@example.com)

---

© 2021 - My Django Application

Algo que puedo notar es que crea una especie de login por defecto, al momento de crear un super usuario, así se ve la vista si se logueo bien



## Comandos utilizados

`virtualenv -p python3 env`: Este comando es para crear un entorno virtual en python 3

luego hacer esto `.\env\scripts\activate`

`pip install Django==3.0.6`: Instalar esta versión de Django porque es la que es compatible con SQL Server

`pip install django-mssql-backend`: Utilizar este comando para instalar el conector de SQL Server en la carpeta de Back End de django

`django-admin startproject proyecto`:

`django-admin startapp Aplicacion1`: Con este comando crea una aplicación dentro de la carpeta del proyecto

`python manage.py migrate` con este comando se encarga de hacer la conexión de la aplicación django con SQL Server, migra tablas

`python manage.py makemigrations`: Sirve para migrar un modelo que este dentro de la carpeta Aplicacion1



`python manage.py createsuperuser`: Sirve para crear un super usuario, pide opciones para darle nombre al usuario, un correo y una contraseña

`python manage.py runserver`: Este comando es para iniciar el servidor de django

`pip freeze > requirements.txt`

`pip install -r .\requirements.txt`: Instalar la lista de dependencias en otro proyecto, se instala su entorno virtual

## **Entornos virtuales**

Un entorno es como si tuviéramos un sistema emulado independiente y totalmente abstraído dentro de nuestra computadora

Para crear un entorno virtual se ejecuta desde el CMD en la carpeta del proyecto el siguiente comando `virtualenv -p python3 venv`

<https://www.youtube.com/watch?v=NkvvWMS220g&t=567s>

`python manage.py makemigrations profiles_api`

## **Bibliografia**

[https://www.youtube.com/watch?v=hftvg3-E\\_dc](https://www.youtube.com/watch?v=hftvg3-E_dc)

<https://www.youtube.com/watch?v=d1r0aK5awWk>

<https://www.youtube.com/watch?v=65gPLE6n9QE>

<https://www.mssqltips.com/sqlservertip/6693/python-django-tutorial-website-sql-server-database/>

## **Glosario de términos**

**Entorno Virtual:** Se usa este comando `virtualenv -p python3 env` el entorno virtual es una especie de maquina virtual, eso significa que en un proyecto de python

se instalan dependencias con esas versiones y esos modulos que se indiquen por lo general en un archivo requirements.txt