# Manual de Usuario Reto 2

Juan Felipe Restrepo Buitrago

Talento B Bancolombia

Postulación

Mayo 18, 2023

# Requerimientos y Especificaciones

Este proyecto fue realizado en PyCharm, un IDE para el desarrollo en Python.

## Librerías

Algunas librerías necesarias para el correcto funcionamiento de este proyecto son:

* django.
* mysql-client
* django-bootstrap-v5

Otras librerías que son opcionales son:

* virtualenv

## Versiones

### Python

Para este proyecto se utilizó la versión de Python 3.11. Instalar desde la página oficial de Python.

### Virtualenv

También se hizo uso de un entorno virtual y se usó la versión 20.23.0 de virtualenv. Para su instalación usar el comando “pip install virtualenv==20.30.0”.

Tiene 2 opciones: instalar las librerías siguientes antes de crear el entorno virtual o crear el entorno virtual y a ese entorno virtual instalarle las librerías siguientes. Si no desea crear un entorno virtual simplemente instale las librerías y siga usando su versión de python original.

Para crear un entorno virtual debemos encontrarnos en la consola y en el directorio en el que queremos que se cree este entorno virtual, ejecutamos el siguiente comando “python -m venv (nombre del entorno virtual)” seguido del comando para activarlo “(nombre del entorno virtual)\Scripts\activate” y con eso su entorno virtual queda activado.

### Django

Para este proyecto se usó la versión 4.2.1 de Django. Para su instalación usar el comando “pip install django==4.2.1”.

### Django-Bootstrap-v5

Se hizo uso de la librería django-bootstrap-v5 para crear plantillas con Bootstrap, la versión utilizada fue la 1.0.11. Para su instalación usar el comando “pip install django-bootstrap-v5==1.0.11”.

### MySQL-Client

Por último, se usó mysql-client para conectar Django a la base de datos de MySQL y la versión utilizada fue la 2.1.1. Para su instalación usar el comando “pip install mysqlclient==2.1.1”.

# Ejecución

Luego de haber hecho las instalaciones y configuraciones necesarias, podemos ejecutar el código.

## Base de Datos

La base de datos que se usó para este proyecto fue MySQL, específicamente, la versión 8.0.32.

Lo primero que debemos hacer antes que nada es crear la base de datos y crear unos cuantos datos para cada tabla. Para la creación de la base de datos, debe ejecutar el script SQL “\MySQL Scripts\Crear DB.sql”.

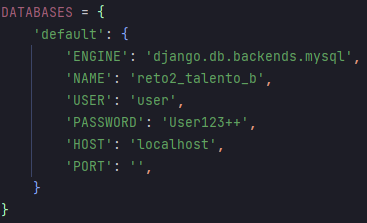
### Datos

Luego de haber creado la base de datos, la cual tiene por nombre “reto2\_talento\_b”, debe insertar los datos y para eso ejecute el script SQL “\MySQL Scripts\Insertar Datos.sql”.

## Conexión a la Base de Datos

Una vez creada la base de datos, debemos especificar unas configuraciones del proyecto para poder ejecutarlos.

Entramos al archivo de Python “Reto2/Reto2/settings.py” y buscamos la sección “DATABASES”.



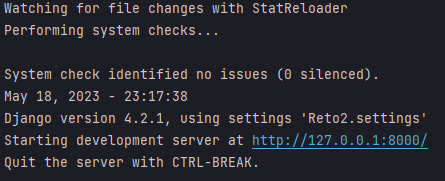
Acá, solo debe cambiar el valor de “USER”, “PASSWORD” y “PORT” por los valores que tiene en su base de datos, es decir, su usuario, la contraseña de ese usuario y el puerto que utiliza para la conexión.

Otra alternativa que puede tener es crear el usuario “user” con la contraseña “User123++” en MySQL.

## Ejecución de los Formulario

Una vez se haya creado la base de datos, insertado los datos y configurado la base de datos en el proyecto, podemos ejecutar el proyecto para ver e interactuar con los formularios.

Para ejecutarlo debemos abrir una terminal y activar el entorno virtual, si decidimos crearlo. Luego debemos movernos al directorio “\Reto2” y posterior a eso ejecutar el siguiente comando “python manage.py runserver”.



Para interactuar con los formularios debe presionar en ese link que se muestra de azul o copiarlo y pegarlo en su navegador.

## Admin

Si quiere tener acceso a la ruta “<http://127.0.0.1:8000/admin/>” debe ejecutar el siguiente comando para crear un super usuario administrados: “python manage.py createsuperuser”

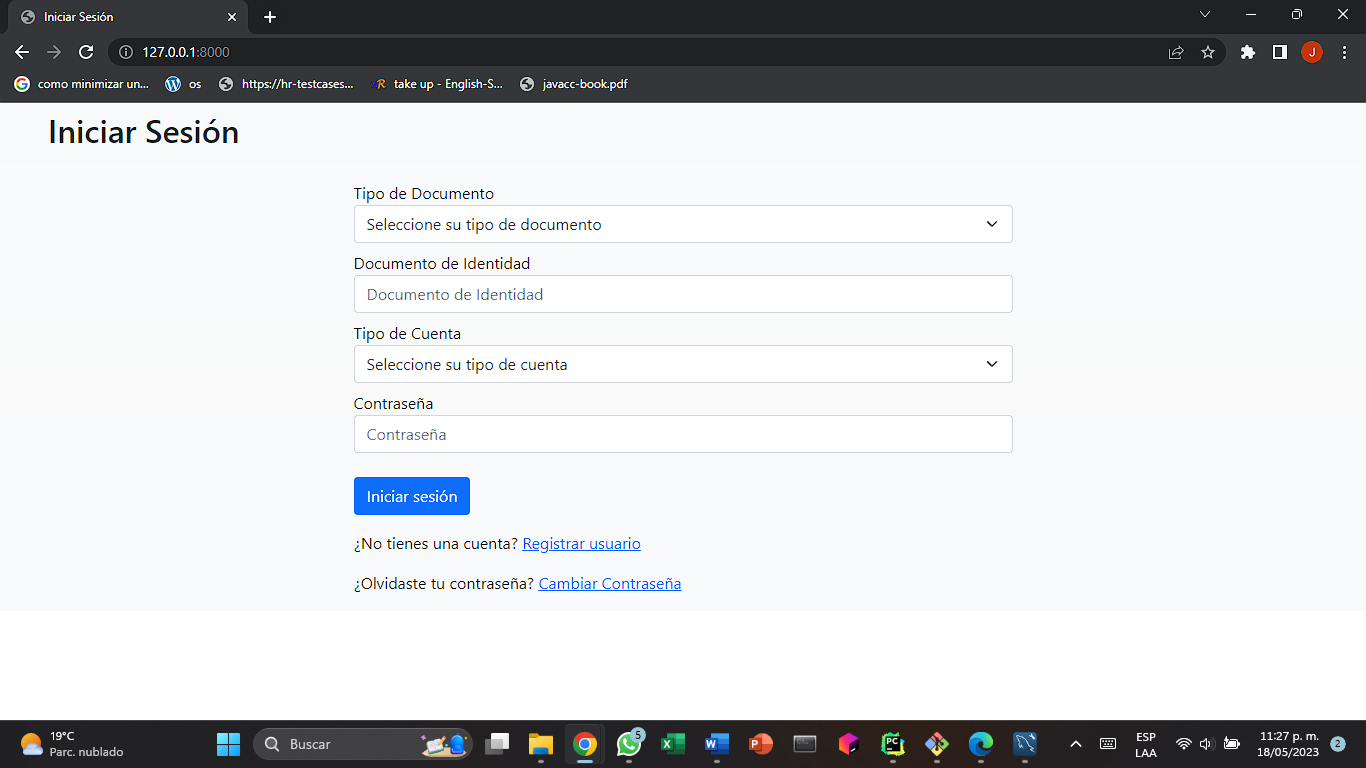
Captura de pantalla de computadora

Descripción generada automáticamente

En este lugar podrás ver, eliminar, crear y modificar cualquier valor de cualquier tabla de la base de datos,

## Importante

Para ingresar al formulario que se pedía en el ejercicio, al momento de entrar debe iniciar sesión con un usuario con cuenta ya existente o crear un usuario e ingresar.

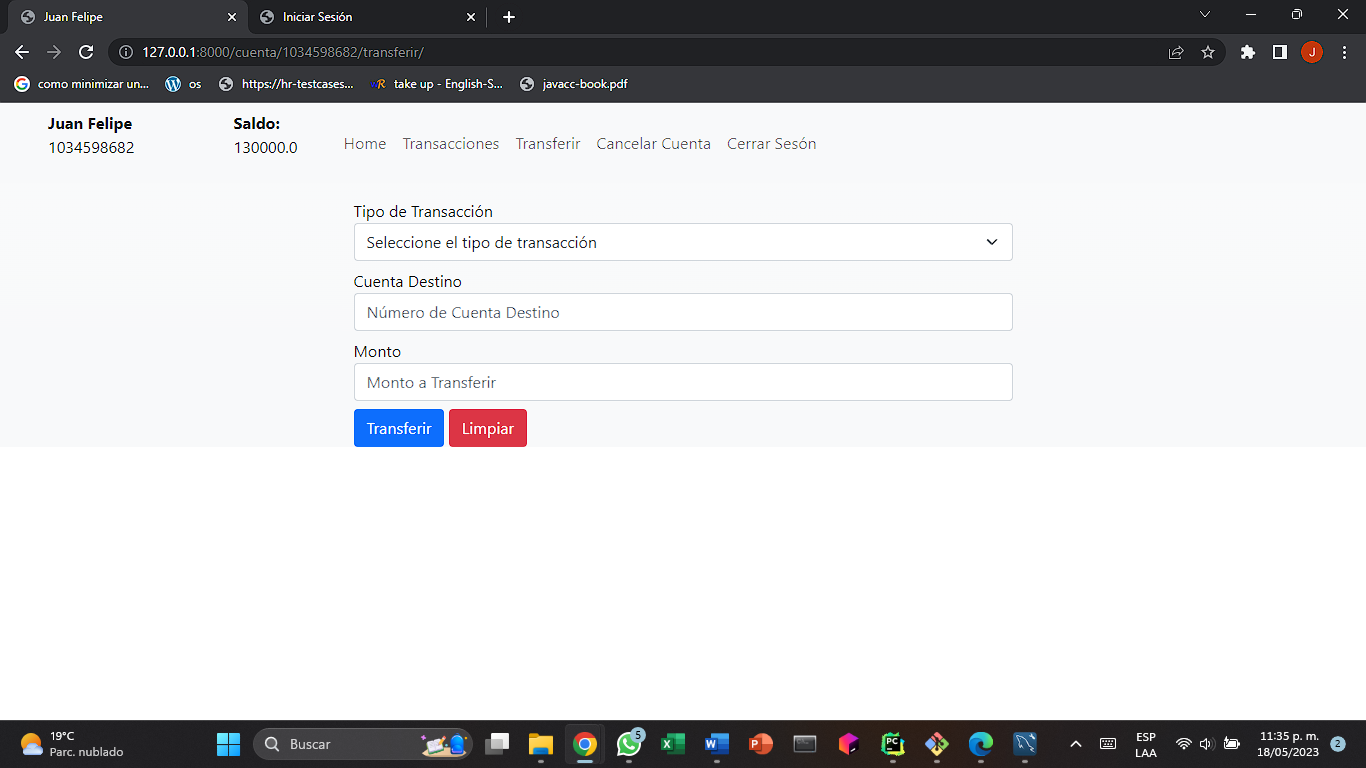


Captura de pantalla de computadora

Descripción generada automáticamente

### Hacer una transacción

Para llegar a la solución de este ejercicio, decidí implementar una función en “Reto2/transacciones/views.py” llamada transfer, donde esta me renderizara el html con el formulario necesario y mostrarme lo siguiente.



Lo que hace dicha función es primero verificar que la cuenta que se le fue pasada como argumento si exista y renderizar la página cuando el método es GET y cuando es POST, trae todos los valores del formulario y hace validación de si la cuenta de destino existe, que la cuenta de destino no sea la misma a la de salida y que el monto a transferir no sea mayor que el saldo de la cuenta de salida. También en esta parte se hace uso de un try catch, por si la función get(), no encuentra ninguna cuenta.

### Cancelar Cuenta

Luego de haber ingresado con algún usuario, desde la barra de navegación el usuario puede cancelar su cuenta. Lo hace desde la función “cancel” en “Reto2/transacciones/views.py”. Dicha función busca la cuenta a cancelar con el método get y crea un movimiento con el valor del saldo de esa cuenta y sin ninguna cuenta de entrada y renderizamos la página principal de la cuenta, donde se verán los cambios hechos, el saldo de la cuenta está en 0.

