

Instrucciones de instalación

Introducción

En el presente documento se definen las instrucciones y pasos a seguir para poder realizar la correcta instalación de la tienda online. Esta guía está orientada sobretodo a los futuros mantenedores de la misma, ya que a nivel de cliente no es necesario seguir estos pasos.

Especificaciones técnicas previas

Previo la instalación, se deben cumplir los siguientes requisitos:

1. Tener acceso a internet
2. Tener instalado python
3. Tener instalado git
4. Tener instalado algún IDE de desarrollo
5. Tener instalado Docker/Docker desktop

Instalación del entorno de desarrollo

En esta sección se definen los requisitos para comenzar a trabajar sobre el proyecto, por lo que se deben seguir los siguientes pasos:

1. En la carpeta donde se quiera trabajar, ejecutar el comando: `git clone https://github.com/IsaacSolisPadilla/Proyecto_PGPI.git`
2. Una vez hecho esto, podemos escoger entre tener un entorno virtual de desarrollo o no. Si no queremos, podemos saltarnos este paso, en caso contrario, deberemos ejecutar en orden los siguientes comandos:
 - `python -m venv venv`
 - `venv\Scripts\activate`
3. Ejecutar el comando `pip install -r requirements.txt` para poder instalar todas las dependencias. Si se desea mantener los datos de prueba que vienen por defecto, podemos saltar al paso 5. De lo contrario, deberemos eliminar todo lo que hay dentro de la carpeta migrations (salvo el `init.py`) y el archivo `db.sqlite3`. El administrador que viene con los datos de prueba tiene las credenciales grupo17@gmail.com y como contraseña 1234
4. Ahora, deberemos crear la base de datos e incluir los datos de prueba, por lo que ejecutaremos los siguientes comandos, y en este orden:
 - `python manage.py makemigrations`
 - `python manage.py migrate`
 - `python manage.py loaddata ./Tienda/fixtures/datos_prueba.json`
 - `python manage.py createsuperuser (username: grupo17, email: grupo17@gmail.com, password: 1234)`

NOTA: Estos comandos pueden encontrarse en el archivo "instalation.sh"

5. Para correr la aplicación ejecutamos el comando `python manage.py runserver`
6. SI NOS SALTAMOS EL PASO 4, PODEMOS SALTAR TAMBIÉN ESTE PASO. Para poder probar la aplicación con el usuario que tiene habilitado la llegada de correos cuando se realizan pedidos, se deberá registrar un nuevo usuario con correo electrónico: grupo17@gmail.com, y contraseña 1234. Cuando se realiza un pedido, se manda un correo electrónico a la cuenta grupo17pgpi@gmail.com, cuya contraseña es 123456789PGPI. Esto es por lo comentado en la incidencia 3 (el hecho de que la API MailSender necesita que pagues para poder enviar correos a usuarios distintos al que generó el token de la API)

Desarrollo de nuevas funcionalidades

Las nuevas funcionalidades deben implementarse como se solicite y se añadirán a través de commits al repositorio siguiendo las metodologías pertinentes a lo establecido por el futuro equipo de desarrollo.

Si se desea actualizar el `requirements.txt` de forma automática, se debe ejecutar el comando `pip freeze > requirements.txt`, pero siempre se deberá comentar la línea `pywin32==306`, ya que genera errores para dockerizar y desplegar la aplicación.

Ejecución de pruebas

Para ejecutar las pruebas se deberá ejecutar el comando `pytest`. Es importante que se debe tener la aplicación en ejecución y ésta deberá tener los datos de prueba en la base de datos.

- Si se desea ejecutar solo las pruebas unitarias, se deberá ejecutar el siguiente comando: `pytest Tienda/pytest`
- Si se desea ejecutar solo las pruebas de selenium, se deberá tener la aplicación corriendo y después ejecutar el comando: `pytest Tienda/selenium`.
- Si se desean ejecutar todas las pruebas y además ver el porcentaje de cobertura de tests, se deberá ejecutar el comando: `pytest --cov`