Documentación Proyecto web portafolio CAHL

1. Creación del proyecto

Terminal:

Microsoft Windows [Versión 10.0.22621.2715]

(c) Microsoft Corporation. Todos los derechos reservados.

#Dirigirse a la carpeta que va a contener el proyecto

#Crear la carpeta que va a contener el proyecto

C:\Users\herre>cd.\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign

C:\Users\herre\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign>mkdir ProjectPortfolioWeb

#Crear el ambiente de Python que vamos a utilizar para el proyecto

C:\Users\herre\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign\ProjectPortfolioWeb>python -m venv env

#Activar el ambiente de trabajo

C:\Users\herre\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign\ProjectPortfolioWeb>.\env\Scripts\activate

#instalar Django

(env) C:\Users\herre\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign\ProjectPortfolioWeb>pip install django==3.2.16

Collecting django==3.2.16

Using cached Django-3.2.16-py3-none-any.whl (7.9 MB)

Collecting asgiref<4,>=3.3.2 (from django==3.2.16)

Obtaining dependency information for asgiref<4,>=3.3.2 from https://files.pythonhosted.org/packages/9b/80/b9051a4a07ad231558fcd8ffc89232711b4e618c15cb7a392a17384bbeef/asgiref-3.7.2-py3-none-any.whl.metadata

Using cached asgiref-3.7.2-py3-none-any.whl.metadata (9.2 kB)

Collecting pytz (from django==3.2.16)

Obtaining dependency information for pytz from https://files.pythonhosted.org/packages/32/4d/aaf7eff5deb402fd9a24a1449a8119f00d74ae9c2efa79f8ef9994261fc2/pytz-2023.3.post1-py2.py3-none-any.whl.metadata

Using cached pytz-2023.3.post1-py2.py3-none-any.whl.metadata (22 kB)

Collecting sqlparse>=0.2.2 (from django==3.2.16)

Using cached sqlparse-0.4.4-py3-none-any.whl (41 kB)

Using cached asgiref-3.7.2-py3-none-any.whl (24 kB)

Using cached pytz-2023.3.post1-py2.py3-none-any.whl (502 kB)

Installing collected packages: pytz, sqlparse, asgiref, django

Successfully installed asgiref-3.7.2 django-3.2.16 pytz-2023.3.post1 sqlparse-0.4.4

[notice] A new release of pip is available: 23.2.1 -> 23.3.1

[notice] To update, run: python.exe -m pip install --upgrade pip

#Crear el core de django

(env) C:\Users\herre\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign\ProjectPortfolioWeb>django-admin s

tartproject core .

#Verificar los archivos creados

(env) C:\Users\herre\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign\ProjectPortfolioWeb>dir

El volumen de la unidad C es OS

El número de serie del volumen es: EED1-0ED8

Directorio de C:\Users\herre\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign\ProjectPortfolioWeb

03/12/2023 02:28 p. m. <DIR> .

03/12/2023 02:16 p. m. <DIR> ..

03/12/2023 02:15 p. m. <DIR> core

03/12/2023 02:13 p. m. <DIR> env

03/12/2023 02:15 p. m. 682 manage.py

03/12/2023 02:12 p. m. 169 requirements.txt

2 archivos 851 bytes

4 dirs 144.880.443.392 bytes libres

#Crear un archive texto dentro de la carpeta del proyecto que contenga las librerías a instalar para desarrollar el proyecto “requirements.txt”

(env) C:\Users\herre\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign\ProjectPortfolioWeb>pip install -r

requirements.txt

Requirement already satisfied: django==3.2.16 in c:\users\herre\jupyternotebooks\notebooks\portfoliocahl\portfoliocahl\webdesign\projectportfolioweb\env\lib\site-packages (from -r requirements.txt (line 1)) (3.2.16)

Collecting django-environ==0.9.0 (from -r requirements.txt (line 2))

Using cached django\_environ-0.9.0-py2.py3-none-any.whl (17 kB)

Collecting django-cors-headers==3.13.0 (from -r requirements.txt (line 3))

Using cached django\_cors\_headers-3.13.0-py3-none-any.whl (13 kB)

Collecting djangorestframework==3.14.0 (from -r requirements.txt (line 4))

Using cached djangorestframework-3.14.0-py3-none-any.whl (1.1 MB)

Collecting Pillow==9.3.0 (from -r requirements.txt (line 5))

Using cached Pillow-9.3.0.tar.gz (50.4 MB)

Installing build dependencies ... done

Getting requirements to build wheel ... done

Preparing metadata (pyproject.toml) ... done

Collecting django-storages==1.13.1 (from -r requirements.txt (line 6))

Using cached django\_storages-1.13.1-py3-none-any.whl (46 kB)

Collecting django-ckeditor==6.3.2 (from -r requirements.txt (line 7))

Using cached django\_ckeditor-6.3.2-py2.py3-none-any.whl (2.4 MB)

Collecting psycopg2==2.9.5 (from -r requirements.txt (line 8))

Using cached psycopg2-2.9.5.tar.gz (384 kB)

Installing build dependencies ... done

Getting requirements to build wheel ... done

Preparing metadata (pyproject.toml) ... done

Requirement already satisfied: asgiref<4,>=3.3.2 in c:\users\herre\jupyternotebooks\notebooks\portfoliocahl\portfoliocahl\webdesign\projectportfolioweb\env\lib\site-packages (from django==3.2.16->-r requirements.txt (line 1)) (3.7.2)

Requirement already satisfied: pytz in c:\users\herre\jupyternotebooks\notebooks\portfoliocahl\portfoliocahl\webdesign\projectportfolioweb\env\lib\site-packages (from django==3.2.16->-r requirements.txt (line 1)) (2023.3.post1)

Requirement already satisfied: sqlparse>=0.2.2 in c:\users\herre\jupyternotebooks\notebooks\portfoliocahl\portfoliocahl\webdesign\projectportfolioweb\env\lib\site-packages (from django==3.2.16->-r requirements.txt (line 1)) (0.4.4)

Collecting django-js-asset>=1.2.2 (from django-ckeditor==6.3.2->-r requirements.txt (line 7))

Obtaining dependency information for django-js-asset>=1.2.2 from https://files.pythonhosted.org/packages/6b/97/dc8601f87f22498a02c1183cf3d691079440b3c97704b327be778c55391e/django\_js\_asset-2.1.0-py3-none-any.whl.metadata

Downloading django\_js\_asset-2.1.0-py3-none-any.whl.metadata (3.4 kB)

Downloading django\_js\_asset-2.1.0-py3-none-any.whl (4.7 kB)

Building wheels for collected packages: Pillow, psycopg2

Building wheel for Pillow (pyproject.toml) ... error

error: subprocess-exited-with-error

× Building wheel for Pillow (pyproject.toml) did not run successfully.

│ exit code: 1

╰─> [203 lines of output]

running bdist\_wheel

running build

running build\_py

creating build

creating build\lib.win-amd64-cpython-312

creating build\lib.win-amd64-cpython-312\PIL

copying src\PIL\BdfFontFile.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\BlpImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\BmpImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\BufrStubImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ContainerIO.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\CurImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\DcxImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\DdsImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\EpsImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ExifTags.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\features.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\FitsImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\FitsStubImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\FliImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\FontFile.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\FpxImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\FtexImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\GbrImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\GdImageFile.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\GifImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\GimpGradientFile.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\GimpPaletteFile.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\GribStubImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\Hdf5StubImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\IcnsImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\IcoImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\Image.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageChops.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageCms.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageColor.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageDraw.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageDraw2.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageEnhance.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageFile.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageFilter.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageFont.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageGrab.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageMath.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageMode.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageMorph.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageOps.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImagePalette.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImagePath.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageQt.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageSequence.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageShow.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageStat.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageTk.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageTransform.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImageWin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\ImtImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\IptcImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\Jpeg2KImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\JpegImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\JpegPresets.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\McIdasImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\MicImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\MpegImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\MpoImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\MspImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\PaletteFile.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\PalmImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\PcdImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\PcfFontFile.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\PcxImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\PdfImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\PdfParser.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\PixarImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\PngImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\PpmImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\PsdImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\PSDraw.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\PyAccess.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\SgiImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\SpiderImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\SunImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\TarIO.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\TgaImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\TiffImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\TiffTags.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\WalImageFile.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\WebPImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\WmfImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\XbmImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\XpmImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\XVThumbImagePlugin.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\\_binary.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\\_deprecate.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\\_tkinter\_finder.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\\_util.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\\_version.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\\_\_init\_\_.py -> build\lib.win-amd64-cpython-312\PIL

copying src\PIL\\_\_main\_\_.py -> build\lib.win-amd64-cpython-312\PIL

running egg\_info

writing src\Pillow.egg-info\PKG-INFO

writing dependency\_links to src\Pillow.egg-info\dependency\_links.txt

writing requirements to src\Pillow.egg-info\requires.txt

writing top-level names to src\Pillow.egg-info\top\_level.txt

reading manifest file 'src\Pillow.egg-info\SOURCES.txt'

reading manifest template 'MANIFEST.in'

warning: no files found matching '\*.c'

warning: no files found matching '\*.h'

warning: no files found matching '\*.sh'

warning: no files found matching '\*.txt'

warning: no previously-included files found matching '.appveyor.yml'

warning: no previously-included files found matching '.clang-format'

warning: no previously-included files found matching '.coveragerc'

warning: no previously-included files found matching '.editorconfig'

warning: no previously-included files found matching '.readthedocs.yml'

warning: no previously-included files found matching 'codecov.yml'

warning: no previously-included files found matching 'renovate.json'

warning: no previously-included files matching '.git\*' found anywhere in distribution

warning: no previously-included files matching '\*.pyc' found anywhere in distribution

warning: no previously-included files matching '\*.so' found anywhere in distribution

no previously-included directories found matching '.ci'

adding license file 'LICENSE'

writing manifest file 'src\Pillow.egg-info\SOURCES.txt'

running build\_ext

The headers or library files could not be found for zlib,

a required dependency when compiling Pillow from source.

Please see the install instructions at:

https://pillow.readthedocs.io/en/latest/installation.html

Traceback (most recent call last):

File "<string>", line 995, in <module>

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\\_\_init\_\_.py", line 103, in setup

return distutils.core.setup(\*\*attrs)

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\\_distutils\core.py", line 185, in setup

return run\_commands(dist)

^^^^^^^^^^^^^^^^^^

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\\_distutils\core.py", line 201, in run\_commands

dist.run\_commands()

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\\_distutils\dist.py", line 969, in run\_commands

self.run\_command(cmd)

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\dist.py", line 963, in run\_command

super().run\_command(command)

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\\_distutils\dist.py", line 988, in run\_command

cmd\_obj.run()

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\wheel\bdist\_wheel.py", line 368, in run

self.run\_command("build")

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\\_distutils\cmd.py", line 318, in run\_command

self.distribution.run\_command(command)

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\dist.py", line 963, in run\_command

super().run\_command(command)

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\\_distutils\dist.py", line 988, in run\_command

cmd\_obj.run()

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\\_distutils\command\build.py", line 131, in run

self.run\_command(cmd\_name)

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\\_distutils\cmd.py", line 318, in run\_command

self.distribution.run\_command(command)

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\dist.py", line 963, in run\_command

super().run\_command(command)

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\\_distutils\dist.py", line 988, in run\_command

cmd\_obj.run()

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\command\build\_ext.py", line 88, in run

\_build\_ext.run(self)

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\\_distutils\command\build\_ext.py", line 345, in run

self.build\_extensions()

File "<string>", line 810, in build\_extensions

RequiredDependencyException: zlib

During handling of the above exception, another exception occurred:

Traceback (most recent call last):

File "C:\Users\herre\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign\ProjectPortfolioWeb\env\Lib\site-packages\pip\\_vendor\pyproject\_hooks\\_in\_process\\_in\_process.py", line 353, in <module>

main()

File "C:\Users\herre\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign\ProjectPortfolioWeb\env\Lib\site-packages\pip\\_vendor\pyproject\_hooks\\_in\_process\\_in\_process.py", line 335, in main

json\_out['return\_val'] = hook(\*\*hook\_input['kwargs'])

^^^^^^^^^^^^^^^^^^^^^^^^^^^^

File "C:\Users\herre\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign\ProjectPortfolioWeb\env\Lib\site-packages\pip\\_vendor\pyproject\_hooks\\_in\_process\\_in\_process.py", line 251, in build\_wheel

return \_build\_backend().build\_wheel(wheel\_directory, config\_settings,

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\build\_meta.py", line 404, in build\_wheel

return self.\_build\_with\_temp\_dir(

^^^^^^^^^^^^^^^^^^^^^^^^^^

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\build\_meta.py", line 389, in \_build\_with\_temp\_dir

self.run\_setup()

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\build\_meta.py", line 480, in run\_setup

super(\_BuildMetaLegacyBackend, self).run\_setup(setup\_script=setup\_script)

File "C:\Users\herre\AppData\Local\Temp\pip-build-env-nt7\_ynww\overlay\Lib\site-packages\setuptools\build\_meta.py", line 311, in run\_setup

exec(code, locals())

File "<string>", line 1012, in <module>

RequiredDependencyException:

The headers or library files could not be found for zlib,

a required dependency when compiling Pillow from source.

Please see the install instructions at:

https://pillow.readthedocs.io/en/latest/installation.html

<string>:45: RuntimeWarning: Pillow 9.3.0 does not support Python 3.12 and does not provide prebuilt Windows binaries. We do not recommend building from source on Windows.

[end of output]

note: This error originates from a subprocess, and is likely not a problem with pip.

ERROR: Failed building wheel for Pillow

Building wheel for psycopg2 (pyproject.toml) ... error

error: subprocess-exited-with-error

× Building wheel for psycopg2 (pyproject.toml) did not run successfully.

│ exit code: 1

╰─> [20 lines of output]

running bdist\_wheel

running build

running build\_py

creating build

creating build\lib.win-amd64-cpython-312

creating build\lib.win-amd64-cpython-312\psycopg2

copying lib\errorcodes.py -> build\lib.win-amd64-cpython-312\psycopg2

copying lib\errors.py -> build\lib.win-amd64-cpython-312\psycopg2

copying lib\extensions.py -> build\lib.win-amd64-cpython-312\psycopg2

copying lib\extras.py -> build\lib.win-amd64-cpython-312\psycopg2

copying lib\pool.py -> build\lib.win-amd64-cpython-312\psycopg2

copying lib\sql.py -> build\lib.win-amd64-cpython-312\psycopg2

copying lib\tz.py -> build\lib.win-amd64-cpython-312\psycopg2

copying lib\\_ipaddress.py -> build\lib.win-amd64-cpython-312\psycopg2

copying lib\\_json.py -> build\lib.win-amd64-cpython-312\psycopg2

copying lib\\_range.py -> build\lib.win-amd64-cpython-312\psycopg2

copying lib\\_\_init\_\_.py -> build\lib.win-amd64-cpython-312\psycopg2

running build\_ext

building 'psycopg2.\_psycopg' extension

error: Microsoft Visual C++ 14.0 or greater is required. Get it with "Microsoft C++ Build Tools": https://visualstudio.microsoft.com/visual-cpp-build-tools/

[end of output]

note: This error originates from a subprocess, and is likely not a problem with pip.

ERROR: Failed building wheel for psycopg2

Failed to build Pillow psycopg2

ERROR: Could not build wheels for Pillow, psycopg2, which is required to install pyproject.toml-based projects

[notice] A new release of pip is available: 23.2.1 -> 23.3.1

[notice] To update, run: python.exe -m pip install --upgrade pip

(env) C:\Users\herre\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign\ProjectPortfolioWeb>pip install -r

#Si se presentan errores verificar que versiones de las librerías se pueden utilizar para el proyecto e identificar si se pueden usar las más actualizadas depende de que librerías soporte las paginas donde se van a publicar los desarrollos web

#Actualizar pip

(env) C:\Users\herre\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign\ProjectPortfolioWeb>python.exe -m

pip install --upgrade pip

Requirement already satisfied: pip in c:\users\herre\jupyternotebooks\notebooks\portfoliocahl\portfoliocahl\webdesign\projectportfolioweb\env\lib\site-packages (23.2.1)

Collecting pip

Obtaining dependency information for pip from https://files.pythonhosted.org/packages/47/6a/453160888fab7c6a432a6e25f8afe6256d0d9f2cbd25971021da6491d899/pip-23.3.1-py3-none-any.whl.metadata

Using cached pip-23.3.1-py3-none-any.whl.metadata (3.5 kB)

Using cached pip-23.3.1-py3-none-any.whl (2.1 MB)

Installing collected packages: pip

Attempting uninstall: pip

Found existing installation: pip 23.2.1

Uninstalling pip-23.2.1:

Successfully uninstalled pip-23.2.1

Successfully installed pip-23.3.1

#Instalamos las versiones actuales de las librerías que presentaron problemas Pillow y psicopg2

(env) C:\Users\herre\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign\ProjectPortfolioWeb>pip install Pillow

Collecting Pillow

Using cached Pillow-10.1.0-cp312-cp312-win\_amd64.whl.metadata (9.6 kB)

Using cached Pillow-10.1.0-cp312-cp312-win\_amd64.whl (2.6 MB)

Installing collected packages: Pillow

Successfully installed Pillow-10.1.0

(env) C:\Users\herre\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign\ProjectPortfolioWeb>pip install ps

ycopg2

Collecting psycopg2

Downloading psycopg2-2.9.9-cp312-cp312-win\_amd64.whl.metadata (4.5 kB)

Downloading psycopg2-2.9.9-cp312-cp312-win\_amd64.whl (1.2 MB)

━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.2/1.2 MB 1.8 MB/s eta 0:00:00

Installing collected packages: psycopg2

Successfully installed psycopg2-2.9.9

(env) C:\Users\herre\JupyterNotebooks\Notebooks\PortfolioCAHL\PortfolioCAHL\WebDesign\ProjectPortfolioWeb>

1. Abrir el proyecto en VSCode
2. Conectar el proyecto a Github

#Se debe crear una ramificación nueva del proyecto en github desde VSCode para poder gestionar el proyecto “staging” se llama en el ejemplo

1. Ir a toptal para crear el archivo gitignore

#Dentro de VSCode crear un archivo .gitignore

#Ir a la página web de toptal <https://www.toptal.com/developers/gitignore> y colocar Django + Pythom + React y crear el archivo que vamos a copiar dentro de VSCode en el.gitignore

#Luego vamos a verificar que .env se encuentre sin comentar y que .Build se encuentre comentado

#Lo que esta comentado es lo que se va a subir a la web y lo que no esta comentado es lo que se va a ignorar

1. Trabajar en settings.py

#Se importan librerías os y environ

import os #para manejar variables

import environ #para manejar variables más complejas como listas

#Se crea la función para variables environ

Env = environ.Env()

Environ.Env.read\_env()

#Se protege la variable de SECRET\_KEY

Para esto vamos a cortar la clave y la vamos a llevar al archivo .env

#Va a buscar con os dentro de .env la SECRET\_KEY y la va a traer

Dentro de settings, definirmos SECRET\_KEY = os.environ.get(‘SECRET\_KEY’)

#Hacemos lo mismo para DEBUG

Dentro de settings editamos DEBUG = os.environ.get(‘DEBUG’)

Dentro de .env colocamos DEBUG=True

#Modificar ALLOWED\_HOSTS

Para este apartado creamos en .env

ALLOWED\_HOSTS.DEV

ALLOWED\_HOSTS.DEPLOY

#Dentro de settings modificamos ALLOWED\_HOSTS

ALLOWED\_HOSTS = env.list(‘ALLOWED\_HOSTS\_DEV’)

#Al final del archive settings colocamos

If not DEBUG:

ALLOWED\_HOSTS = env.list(‘ALLOWED\_HOSTS\_DEPLOY’)

#Ahora vamos a modificar applications definition para esto creamos una carpeta fuera de core llamada ‘apps’

#Ahora vamos a copiar \_\_init\_\_.py de ‘core’ dentro de ‘apps’

#Modificamos INSTALLED\_APPS por DJANGO\_APPS y creamos PROJECTS\_APPS Y THIRD\_PARTY\_APPS

DJANGO\_APPS = [

'django.contrib.admin',

'django.contrib.auth',

'django.contrib.contenttypes',

'django.contrib.sessions',

'django.contrib.messages',

'django.contrib.staticfiles',

]

PROJECT\_APPS = [

'corsheaders',

'rest\_framework',

'ckeditor',

'ckeditor\_uploader'

]

THIRD\_PARTY\_APPS=[

]

INSTALLED\_APPS = DJANGO\_APPS + PROJECT\_APPS + THIRD\_PARTY\_APPS

1. Ahora vamos a activar ckeditor dentro de settings

#Dentro de la documentación esta la línea de código para activar ckeditor

<https://pypi.org/project/django-ckeditor/3.6.2.1/>

#ckeditor config

CKEDITOR\_CONFIGS = {

'default': {

'toolbar':'full',

'autoParagraph': False

}

}

#creamos una carpeta llamada ‘media’ al nivel de core

CKEDITOR\_UPLOAD\_PATH = "/media/"

1. Ahora vamos a configurar el MIDDLEWARE dentro de setting

#Adicinamos al principio

'corsheaders.middleware.CorsMiddleware'

1. Crear la base de datos sqlite

Se corre el comando Python .\manage.py migrate

Esto permite crear la base de datos

1. Definir database cuando no estamos en DEBUG

#Se coloca al final del archivo dentro del if no DEBUG:

DATABASES = {

'default':env.db('DATABASE\_URL'),

}

DATABASES['deafult']['ATOMIC\_REQUESTS'] = True

#En el archivo .env se define DATABASE\_URL = ‘ ‘ vacío por ahora

1. Configurar lenguaje de la página

Dentro de LANGUAGE\_CODE, especificar el lenguaje y la zona horaria

1. Static

#dentro de settings en STATIC Podemos adicionar

STATIC\_ROOT = os.path.join(BASE\_DIR, 'static')

1. Definir la carpeta para imágenes dentro de setting

#debajo de STATIC definimos MEDIA

STATIC\_ROOT = os.path.join(BASE\_DIR, 'static')

STATIC\_URL = ‘/media/’

1. Definer build/static

#Debajo de STATIC\_URL

STATICFILES\_\_DIRS = [

os.path.join(BASE\_DIR, 'build/static')

1. Configurar Djano RestFramework

#debajo de DEFAUL\_AUTO\_FIELD

Vamos a definir los permisos basados en esta pagina de documentación

<https://www.django-rest-framework.org/api-guide/permissions/>

REST\_FRAMEWORK = {

'DEFAULT\_PERMISSION\_CLASSES': [

'rest\_framework.permissions.IsAuthenticatedOrReadOnly',

]

}

1. Definir quien puede usar nuestro sitio / hacemos lo mismo para CORS\_ORIGIN\_WHITELIST y para CSRF\_TRUSTED\_ORIGINS

#En settings debajo de rest\_framework

#localhost:react

CORS\_ORIGIN\_WHITELIST = env.list('CORS\_ORIGIN\_WHITELIST\_DEV')

#En .env

CORS\_ORIGIN\_WHITELIST\_DEV = 'http://localhost:3000'

CORS\_ORIGIN\_WHITELIST\_DEPLOY = ' '

#En if not DEBUG:

CORS\_ORIGIN\_WHITELIST = env.list('CORS\_ORIGIN\_WHITELIST\_DEPLOY')

#En settings debajo de rest\_framework

#localhost:react

CSRF\_TRUSTED\_ORIGINS = env.list(CSRF\_TRUSTED\_ORIGINS\_DEV')

#En .env

CSRF\_TRUSTED\_ORIGINS \_DEV = 'http://localhost:3000'

CSRF\_TRUSTED\_ORIGINS \_DEPLOY = ' '

#En if not DEBUG:

CSRF\_TRUSTED\_ORIGINS = env.list(‘CSRF\_TRUSTED\_ORIGINS \_DEPLOY')

1. Definir correo electrónico en modo desarrollo

#Debajo de CSRF\_TRUSTED\_ORIGINS

EMAIL\_BACKEND = 'django.core.mail.backends.console.EmailBackend'

1. Vamos a definir las url

Ingresamos a urls.py

from django.conf.ulrs.static import static

from django.conf import settings

urlpatterns = [

path('admin/', admin.site.urls),

] + static(settings.MEDIA\_URL, document\_root = settings.MEDIA\_ROOT)

INSTALLAR REACT

1. Actualizar Node.JS y verificar la versión de npm, actualizar de ser necesario

#Actulizar npm

npm install -g npm@latest

1. Installar react

npx create-react-app ‘portfolio’ #se puede definir el nombre que uno desee

una vez instalado se toman las carpetas de src, public y el archivo package y los llevamos a la carpeta del proyecto de Django.

CONECTAR DJANGO CON REACT

1. Corremos en la terminal del proyecto

npm i #lo que va a correr um ambiente virtual de node\_modules con base em package.json se crea package-lock.json

1. Corremos el build de react

En la consola npm run build #se crea la carpeta de build

1. Corremos static para Python donde obtiene lo necesario para la aplicación

En la terminar

Python .\manage.py collectstatic

1. Modificamos en settings TEMPLATES

TEMPLATES = [

{

'BACKEND': 'django.template.backends.django.DjangoTemplates',

'DIRS': [os.path.join(BASE\_DIR,'build')], #este es el apartado donde apuntamos

Modificamos en urls

from django.contrib import admin

from django.urls import path, re\_path, include #adicionamos

from django.views.generic import TemplateView #adicionamos

from django.conf.urls.static import static

from django.conf import settings

urlpatterns = [

path('admin/', admin.site.urls),

] + static(settings.MEDIA\_URL, document\_root = settings.MEDIA\_ROOT)

urlpatterns += [re\_path(r'^.\*', TemplateView.as\_view(template\_name='index.html'))] #adicionamos

para ver que react corre en django en la terminal colocamos:

Python .\manage.py runserver

Se despliega en el buscador.

CONFIGURAR PROYECTO DE REACT

1. Eliminados app.css y app.test.js
2. De index.js quitamos

// If you want to start measuring performance in your app, pass a function

// to log results (for example: reportWebVitals(console.log))

// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals

reportWebVitals();

1. Creamos la carpeta assets dentro de src
2. Creamos la carpeta styles dentro de src
3. Creamos videos e images dentro de assets para contener los recursos estáticos
4. Creamos hocs, creamos containers (dentro creamos errors y pages) , creamos components, creamos redux (dentro creamos actions y reducers)
5. Vamos a instalar tailwind para mejorar las visuales

<https://tailwindcss.com/docs/guides/create-react-app>

#comando en la terminal

npm install -D tailwindcss

Seguimos la guia