

Escuela Internacional de Postgrado



Buenas Prácticas de programación con Python

**Trabajo sobre la lección 3. Programación
dirigida por la documentación**

Alumno:

Dante Alexander Espinoza Roble

Profesor:

D. Rubén Sánchez

España 2022

Para la actividad de esta lección se proponen dos ejercicios:

1. Busque una herramienta alternativa a Sphinx para generar la documentación de código en Python.
2. Realice la documentación de alguno de los códigos que ha desarrollado a lo largo de este curso. La documentación puede generarla con Sphinx o con la herramienta que ha buscado en el primer ejercicio.

Condiciones de entrega:

- Entregue un documento en formato PDF en el que describa la herramienta de documentación que ha encontrado. En dicho documento debe mostrarse un ejemplo básico del funcionamiento de esta herramienta.
- Entregue el código fuente.

1. Doxygen: Herramienta alternativa a Sphinx

Doxygen es una herramienta informática que facilita generar la documentación de un proyecto software a partir de texto marcado dentro del código fuente, estas marcas identifican las zonas que serán tomadas como entrada por Doxygen y generar una salida que puede ser un fichero de texto plano en un formato a medida, generalmente html, pdf, doc, xml, rtf u otro.

El contenido de la salida que genera Doxygen, incluye documentación de las estructuras de datos sean estas clases, atributos, métodos, también listado de ficheros que contiene el proyecto, diagramas que modelan la jerarquía de clases, un índice que describe como está distribuido dicho documento de salida, etc.

Doxygen permite hacer la documentación no solo de proyectos escritos en lenguaje Python, también es usado con los lenguajes C/C++, Java, PHP entre otros.

A continuación se muestra una lista de imágenes que detallan la instalación, configuración y uso de la herramienta, para que al final entreguen la documentación generada a partir de texto a modo de comentario e identificado con palabras reservadas de la herramienta. La salida es un documento html navegable inline.

Usando la herramienta Doxygen

```
s/puntoUno$ sudo apt update && sudo apt install doxygen doxygen-gui
[sudo] contraseña para dent:
Obj:1 http://es.archive.ubuntu.com/ubuntu focal InRelease
Obj:2 https://apt.releases.hashicorp.com focal InRelease

Des:3 http://es.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]

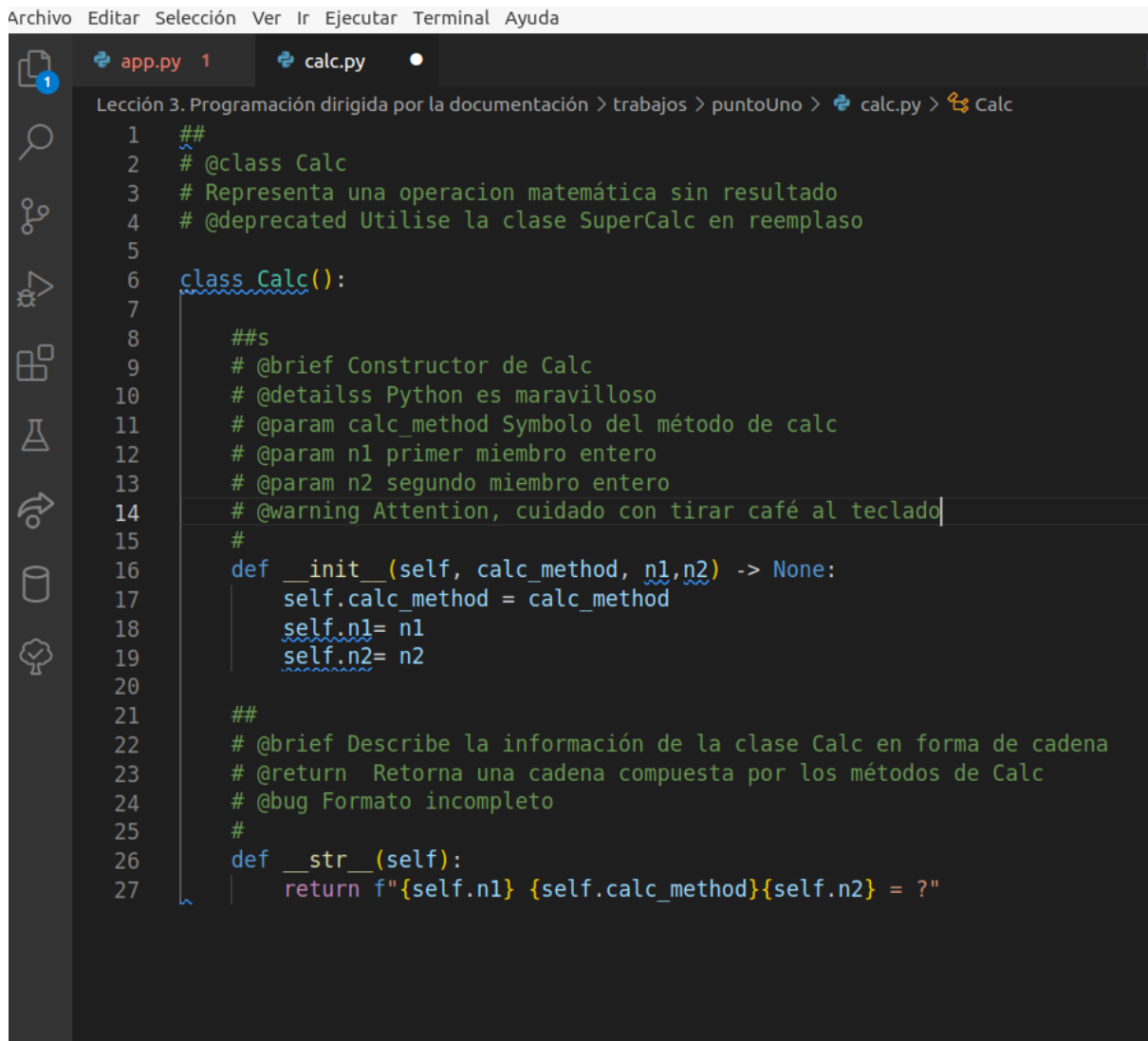
Obj:4 https://packages.microsoft.com/repos/ms-teams stable InRelease

Obj:5 https://deb.nodesource.com/node_16.x focal InRelease

Obj:6 https://dl.google.com/linux/chrome/deb stable InRelease

Obj:7 https://deb.opera.com/opera-stable stable InRelease
Des:8 http://es.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Des:9 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Des:10 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 DEP-11 Metadata [278 kB]
Des:11 http://es.archive.ubuntu.com/ubuntu focal-updates/universe amd64 DEP-11 Metadata [391 kB]
Des:12 http://security.ubuntu.com/ubuntu focal-security/main amd64 DEP-11 Metadata [40,7 kB]
```

Fig A. Instalación de Doxygen



```
Archivo Editar Selección Ver Ir Ejecutar Terminal Ayuda
app.py 1 calc.py
Lección 3. Programación dirigida por la documentación > trabajos > puntoUno > calc.py > Calc
1  ##
2  # @class Calc
3  # Representa una operacion matemática sin resultado
4  # @deprecated Utilise la clase SuperCalc en reemplazo
5
6  class Calc():
7
8      ##s
9      # @brief Constructor de Calc
10     # @detailss Python es maravilloso
11     # @param calc_method Symbolo del método de calc
12     # @param n1 primer miembro entero
13     # @param n2 segundo miembro entero
14     # @warning Attention, cuidado con tirar café al teclado
15     #
16     def __init__(self, calc_method, n1,n2) -> None:
17         self.calc_method = calc_method
18         self.n1= n1
19         self.n2= n2
20
21     ##
22     # @brief Describe la información de la clase Calc en forma de cadena
23     # @return Retorna una cadena compuesta por los métodos de Calc
24     # @bug Formato incompleto
25     #
26     def __str__(self):
27         return f"{self.n1} {self.calc_method}{self.n2} = ?"
```

Fig B. Clase Calc y texto marcado con el caracter # seguido de palabra reservadas de la herramienta

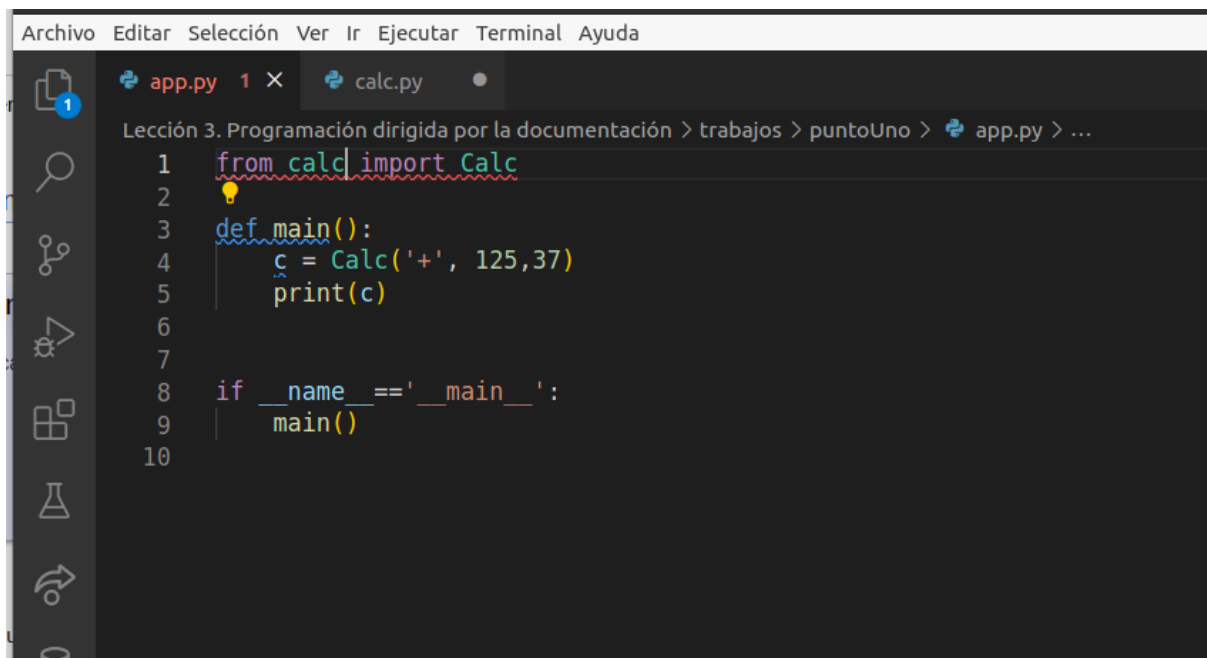


Fig C. Clase main que interactúa con clase Calc

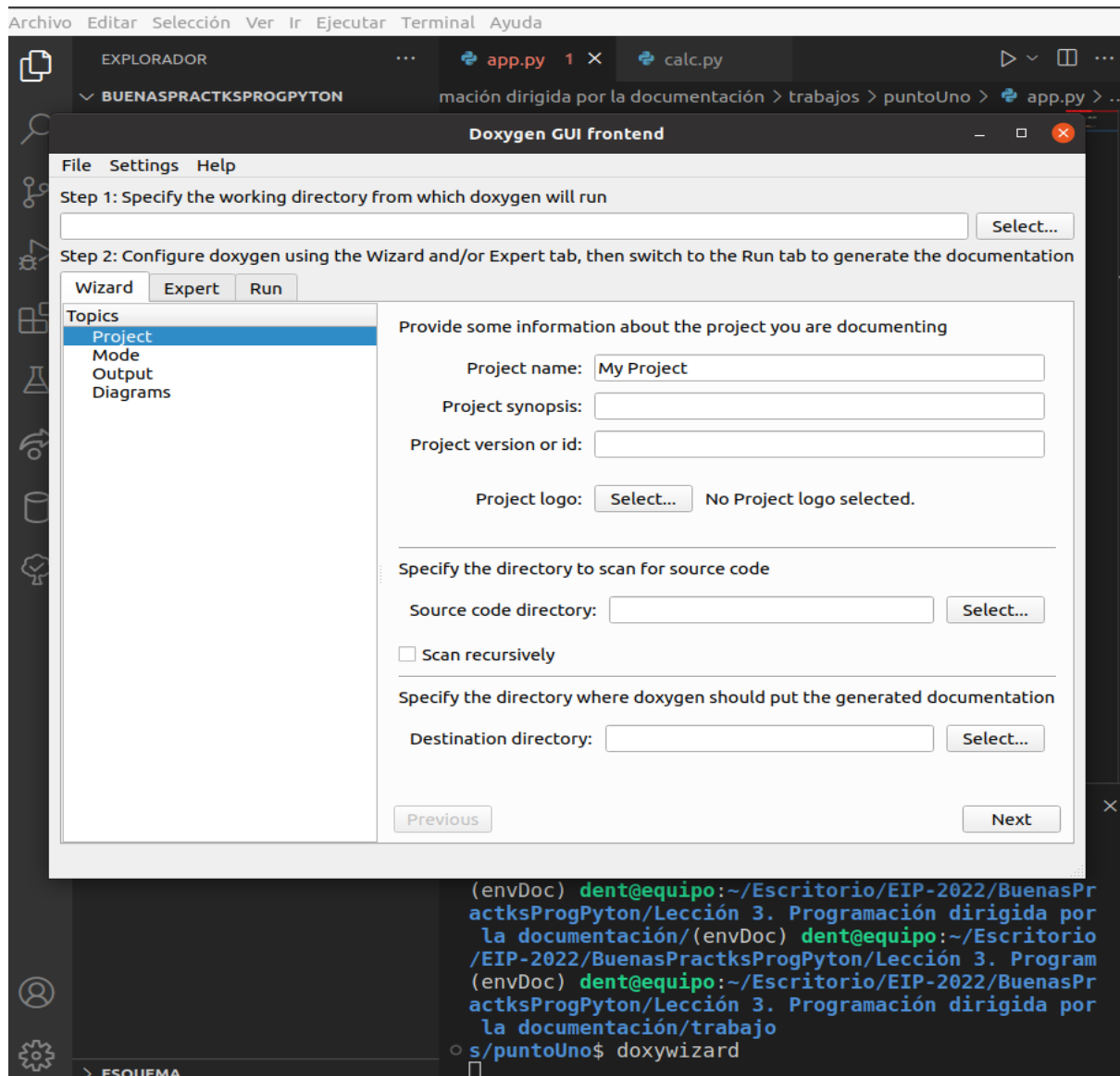


Fig D. Configuración de herramienta

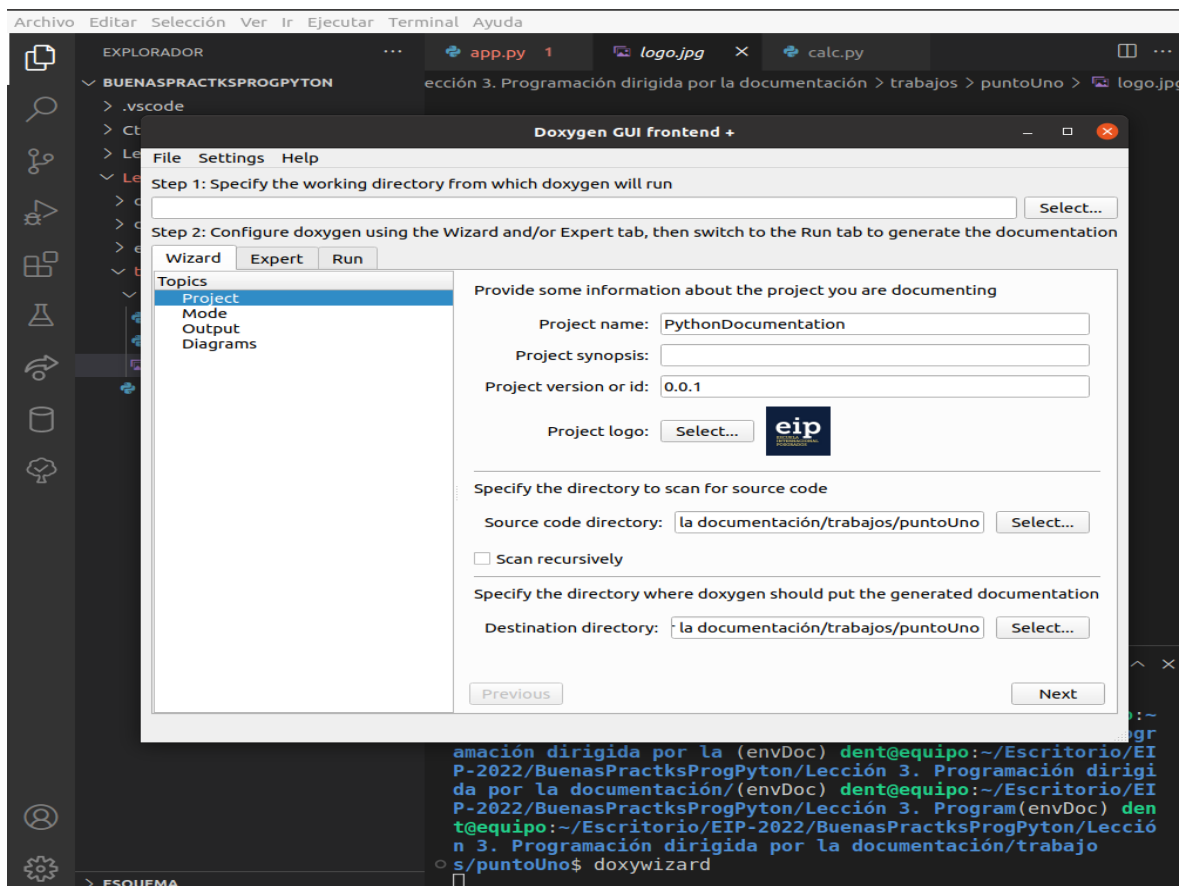


Fig E. Configuración de herramienta

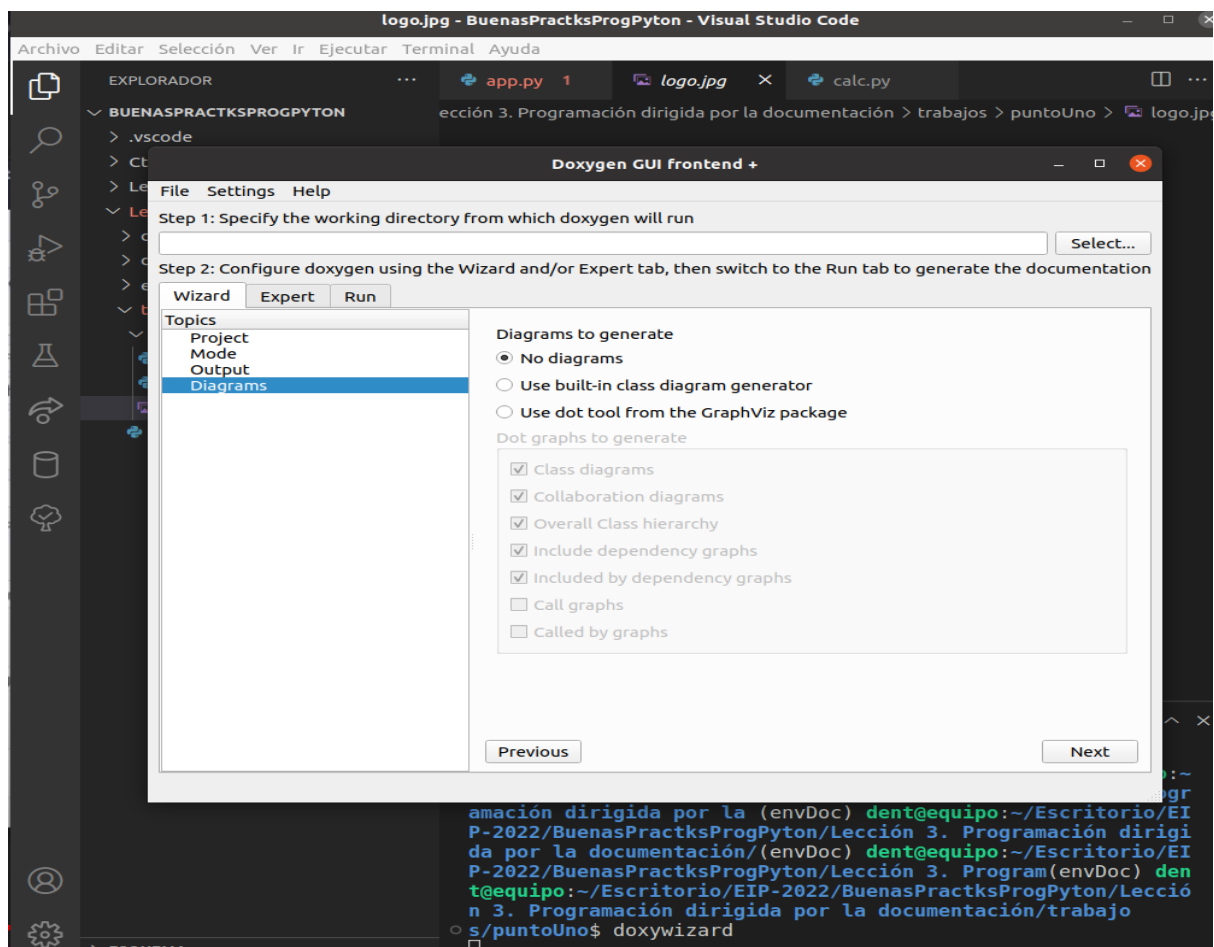


Fig F. Configuración de herramienta

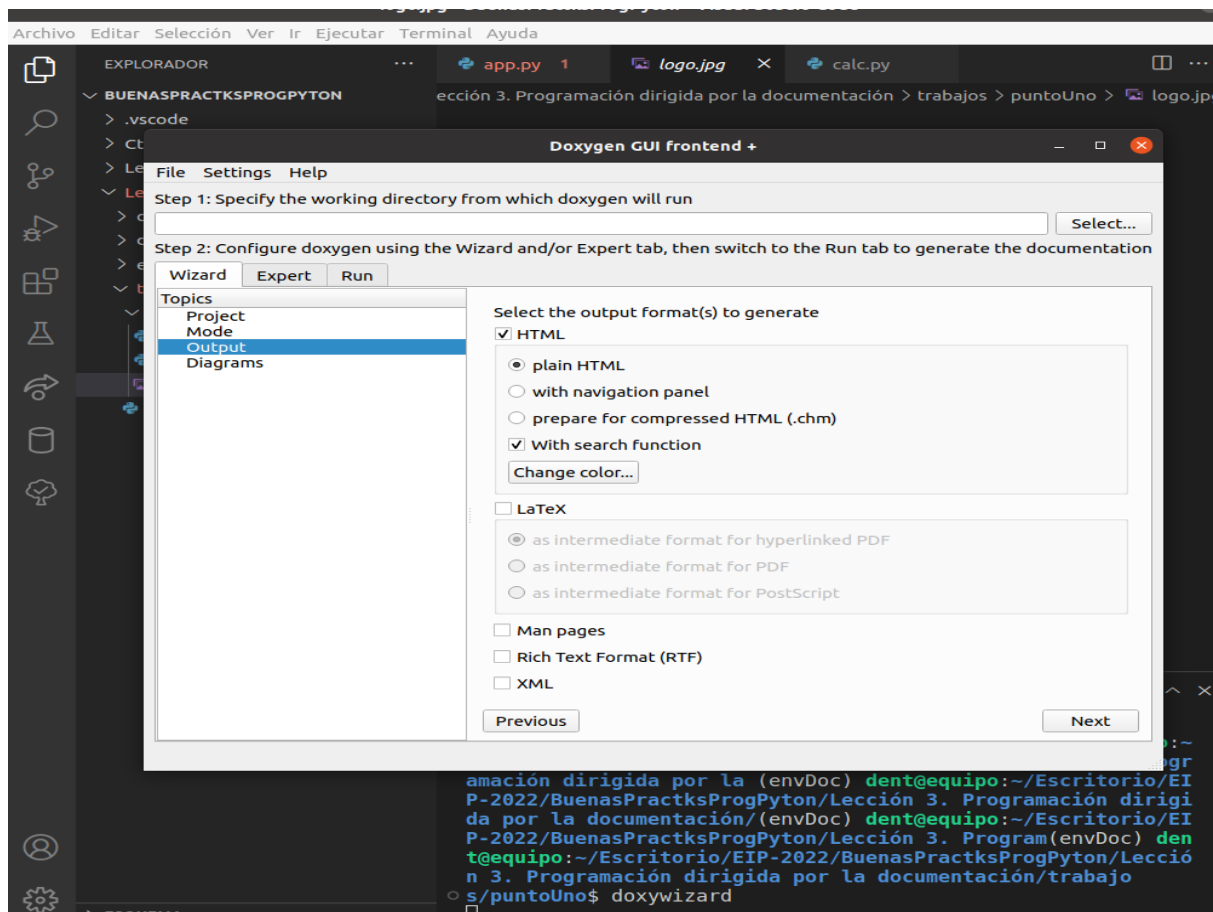


Fig G. Configuración de herramienta

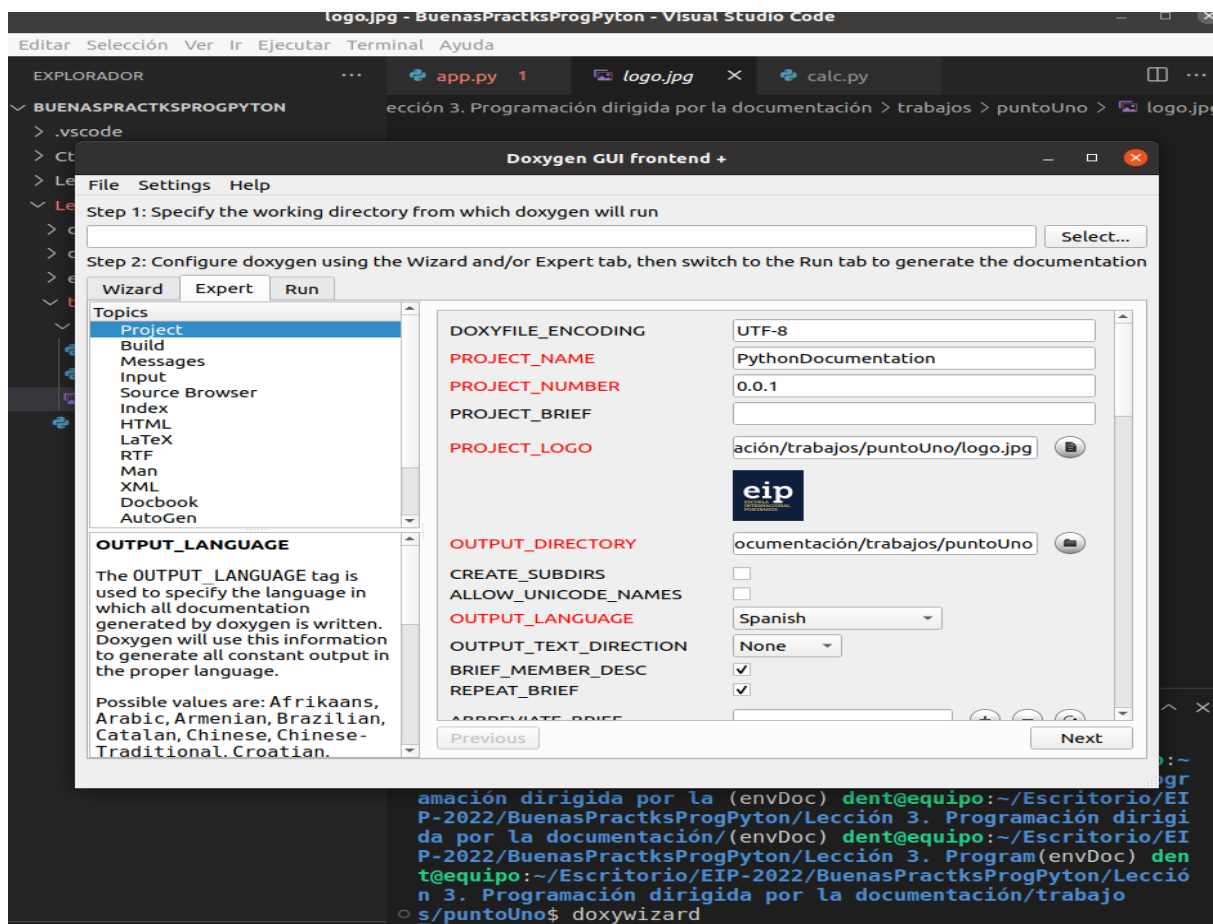


Fig H. Configuración de herramienta

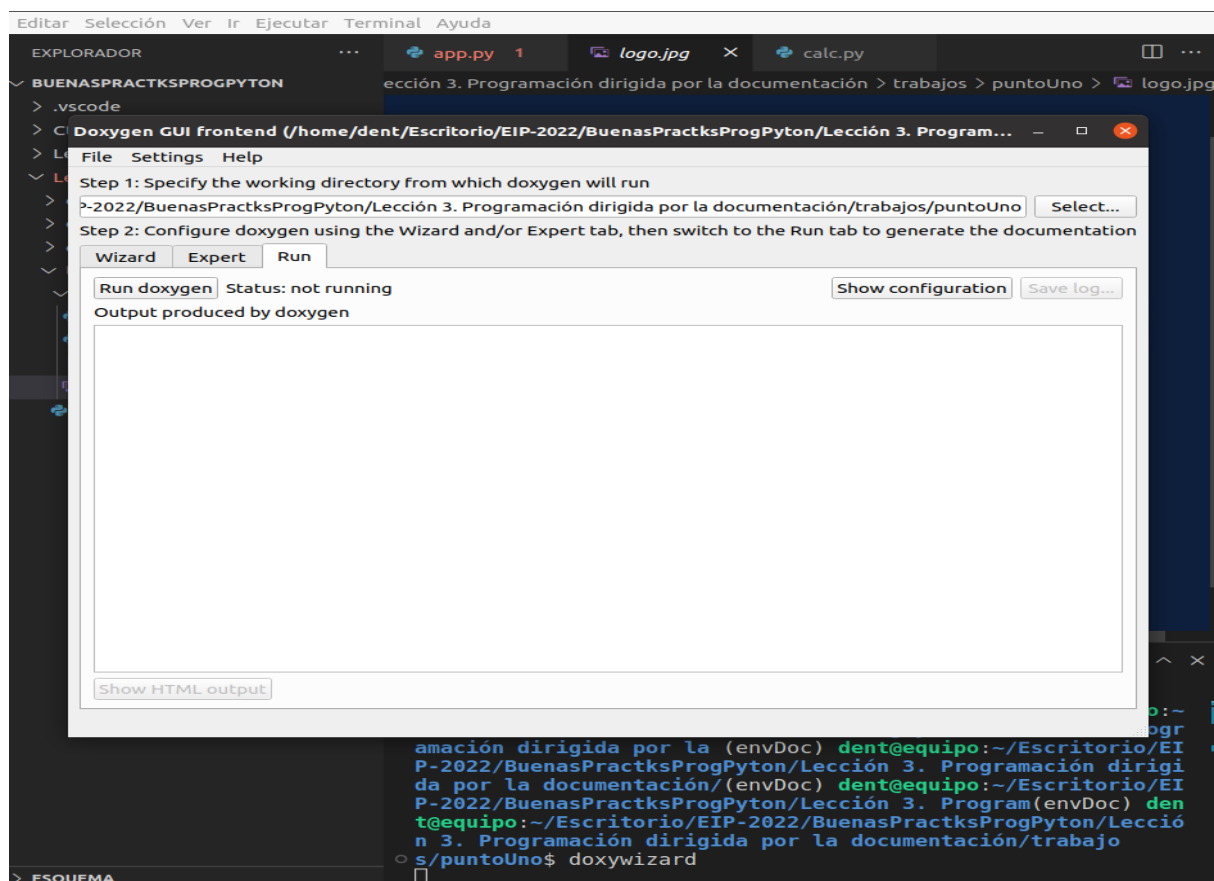


Fig I. Configuración de herramienta

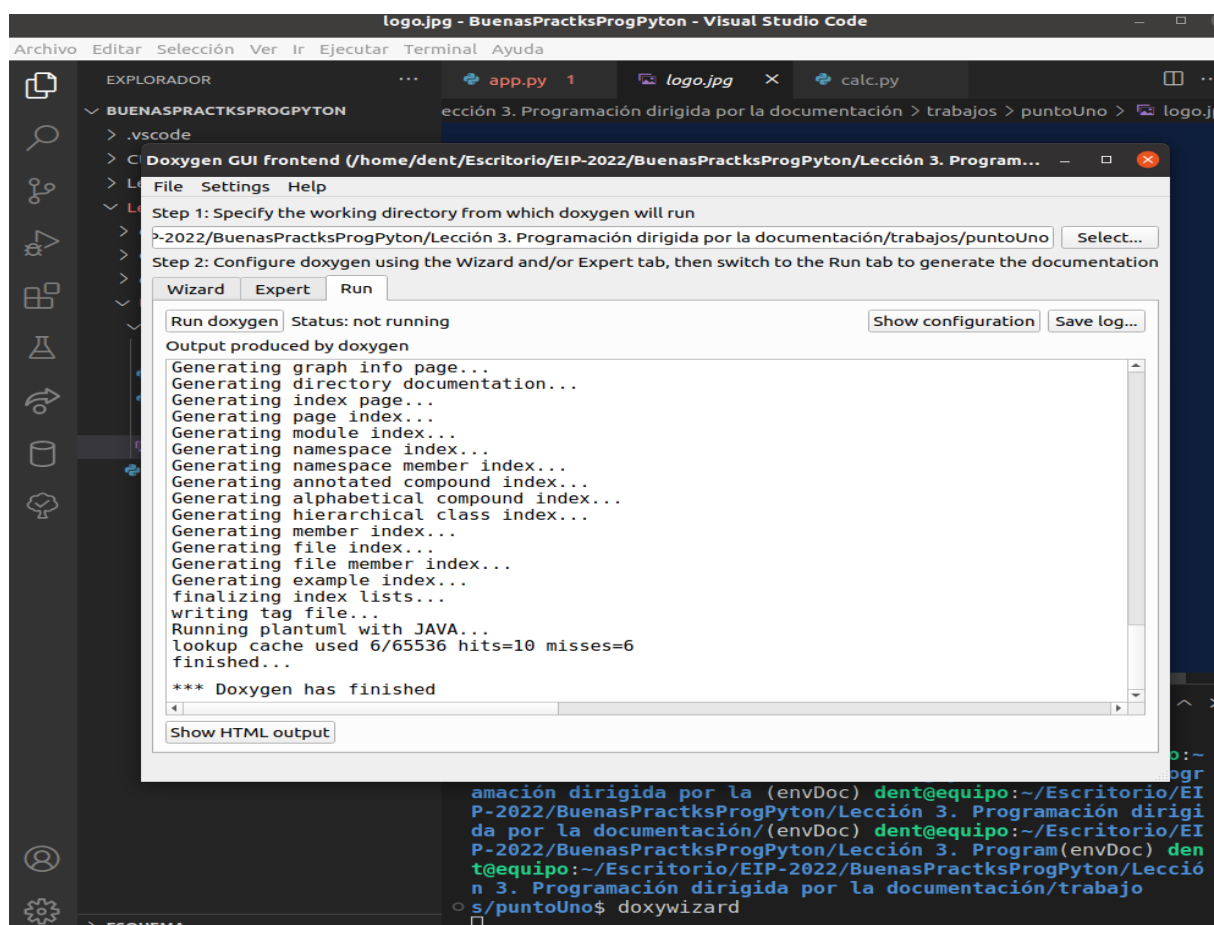


Fig J. Configuración de herramienta

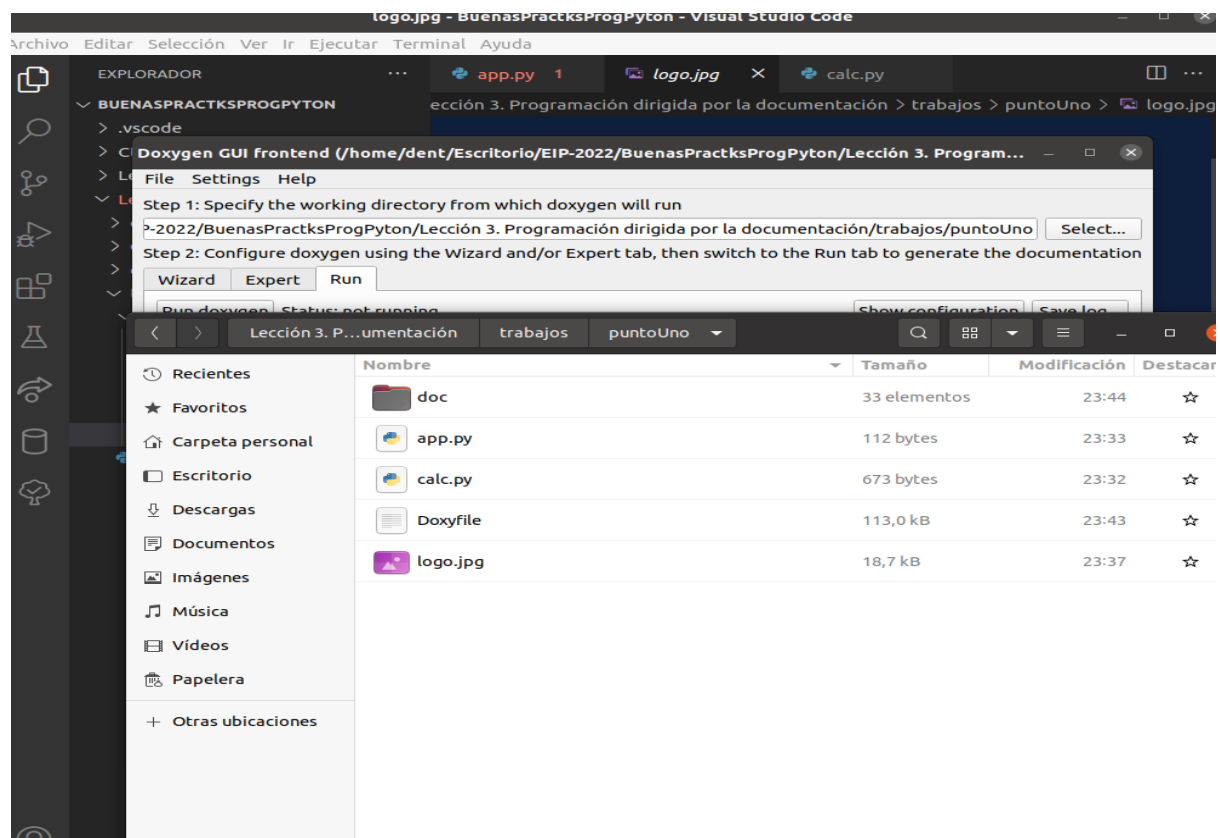


Fig K. Directorios y ficheros creados al ejecutar Doxygen

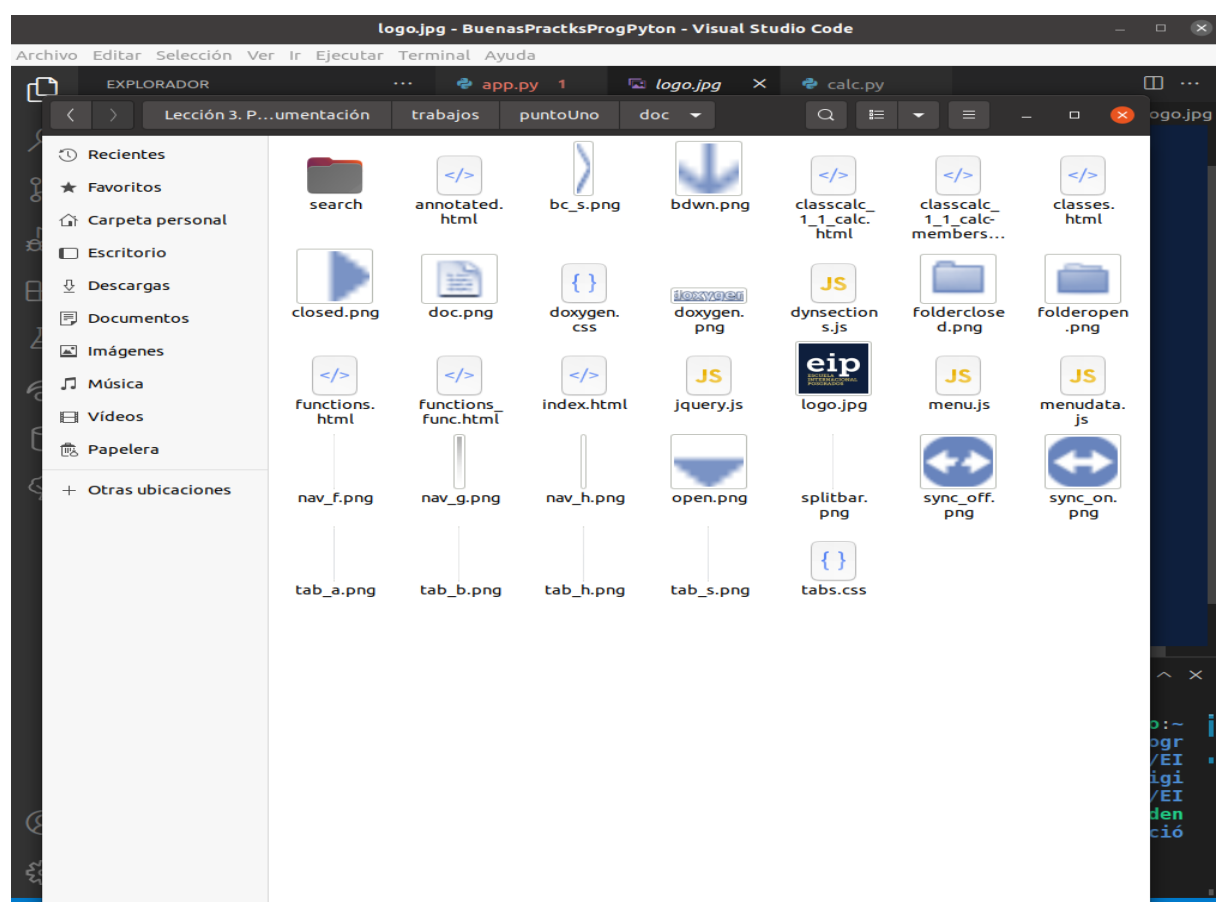


Fig L. Directorios y ficheros creados al ejecutar Doxygen

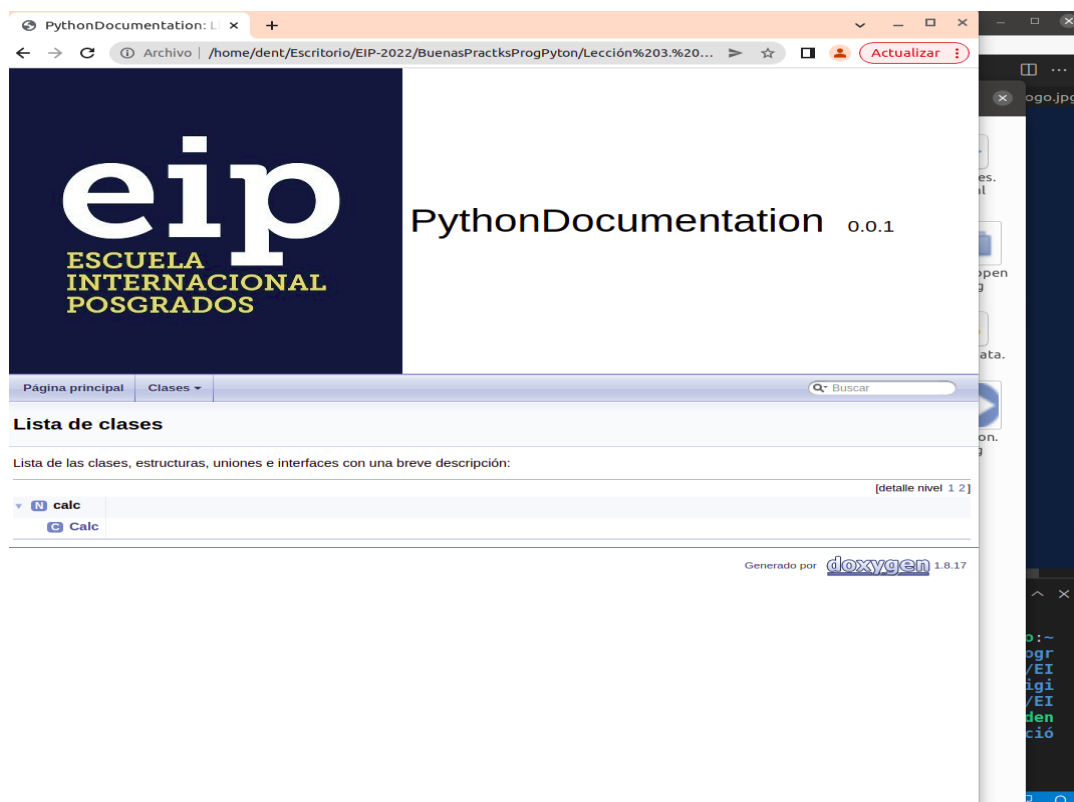


Fig M. Documento HTML generado desde las líneas comentadas en los ficheros python. DashBoard

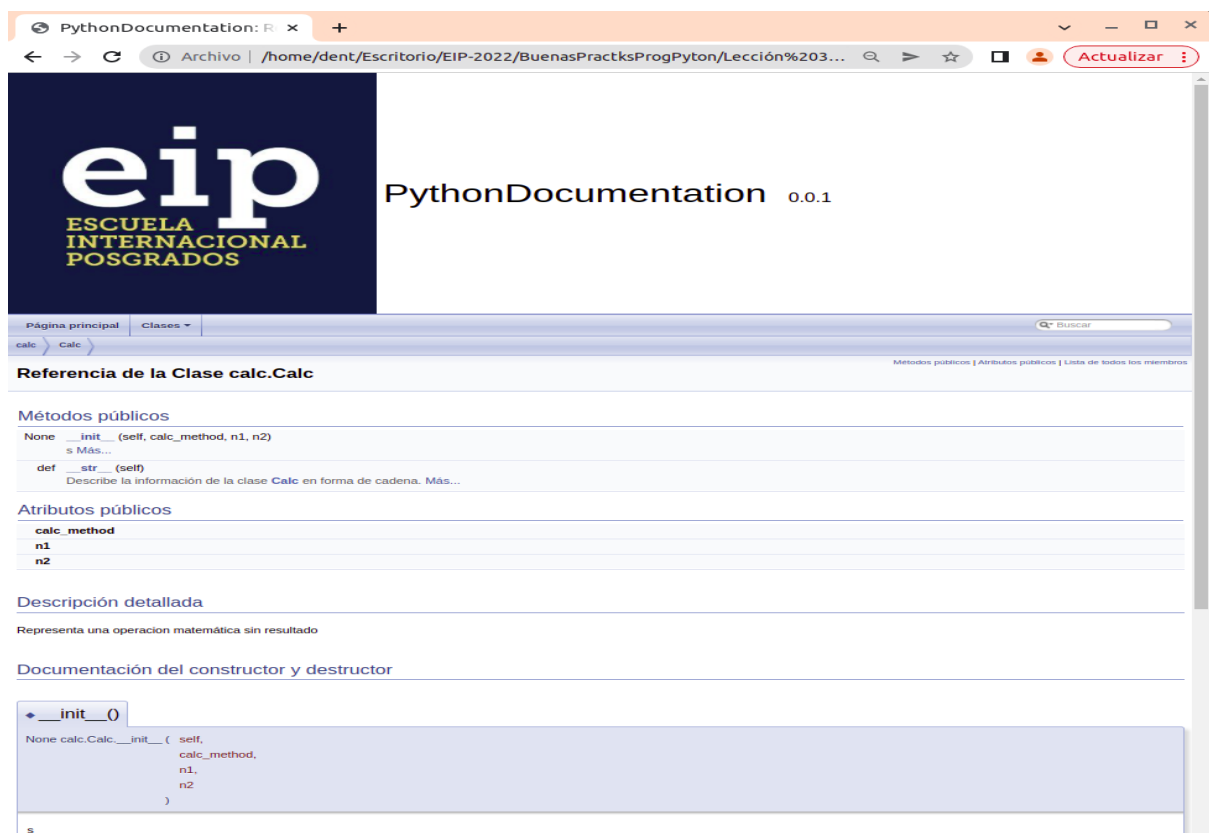


Fig N. Documento HTML generado desde las líneas comentadas en los ficheros python. Vista de Clase Calc

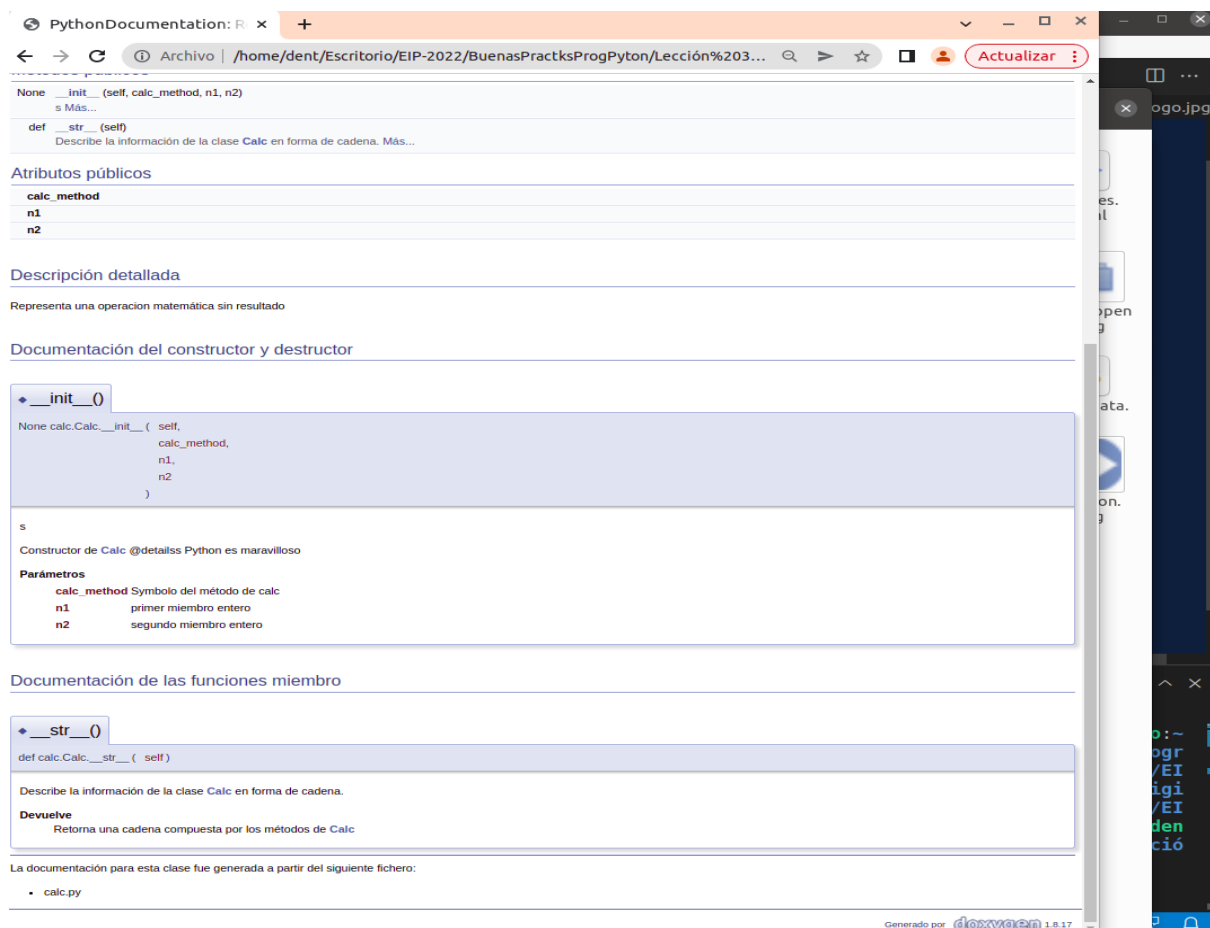


Fig O. Documento HTML generado desde las líneas comentadas en los ficheros python. Vista de métodos

2. Sphinx: Herramienta para documentar mi proyecto python.

```
(envDoc) dent@equipo:~/Escritorio/EIP-2022/BuenasPractksProgPython/Lección 3. Programación dirigida por la documentación/trabajos/puntoDos/docs$ sphinx-quickstart
Bienvenido a la utilidad de inicio rápido de Sphinx 5.2.2.

Ingrese los valores para las siguientes configuraciones (solo presione Entrar para
aceptar un valor predeterminado, si se da uno entre paréntesis).

Ruta raíz seleccionada: .

Tiene dos opciones para colocar el directorio de compilación para la salida de Sphinx.
0 usas un directorio " build" dentro de la ruta raíz, o separas
directorios "fuente" y "compilación" dentro de la ruta raíz.
> Separar directorios fuente y compilado (y/n) [n]: n

El nombre del proyecto aparecerá en varios lugares en la documentación construida.
> Nombre de proyecto: Lección 3. Programación dirigida por la documentación
> Autor(es): Dante Espinoza
> Liberación del proyecto []: 0.0.1

Si los documentos deben escribirse en un idioma que no sea inglés,
puede seleccionar un idioma aquí por su código de idioma. Sphinx entonces
traducir el texto que genera a ese idioma.

Para obtener una lista de códigos compatibles, vea
https://www.sphinx-doc.org/en/master/usage/configuration.html#confval-language.
> Lenguaje del proyecto [en]:

Creando archivo /home/dent/Escritorio/EIP-2022/BuenasPractksProgPython/Lección 3. Programación dirigida por la documentación/trabajos/puntoDos/docs/conf.py.
Creando archivo /home/dent/Escritorio/EIP-2022/BuenasPractksProgPython/Lección 3. Programación dirigida por la documentación/trabajos/puntoDos/docs/index.rst.
Creando archivo /home/dent/Escritorio/EIP-2022/BuenasPractksProgPython/Lección 3. Programación dirigida por la documentación/trabajos/puntoDos/docs/Makefile.
Creando archivo /home/dent/Escritorio/EIP-2022/BuenasPractksProgPython/Lección 3. Programación dirigida por la documentación/trabajos/puntoDos/docs/make.bat.

Terminado: se ha creado una estructura de directorio inicial.

Ahora debe completar su archivo maestro /home/dent/Escritorio/EIP-2022/BuenasPractksProgPython/Lección 3. Programación dirigida por la documentación/trabajos/puntoDos/docs/index.rst y crea
r otros archivos fuente
de documentación. Use el archivo Makefile para compilar los documentos, así ejecute el comando:
make builder
donde "builder" es uno de los constructores compatibles, por ejemplo, html, latex o linkcheck.
```

Fig A. Creando la estructura del proyecto con el comando sphinx-quickstart

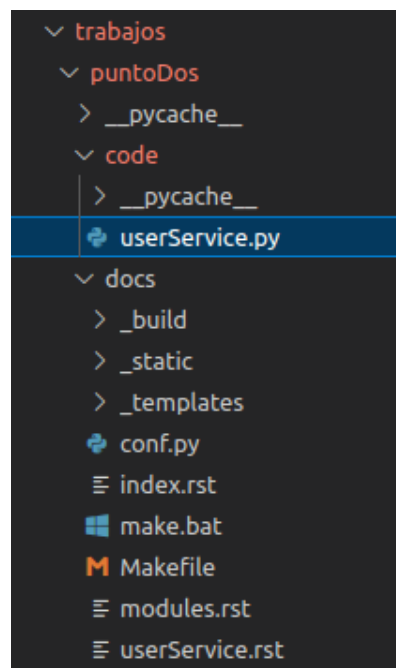


Fig B. Estructura del proyecto creada luego de ejecutar el comando sphinx-quickstart

```

(envDoc) dent@equipo:~/Escritorio/EIP-2022/BuenasPractksProgPyton/Lección 3. Programación dirigida por la documentación/trabajos/puntoDos/docs$ make clean html
Removing everything under 'build'...
Ejecutando Sphinx v5.2.2
creando directorio de salida... hecho
compilando [mo]: los objetivos para 0 los archivos po que estan desactualizados
compilando [html]: los objetivos para 1 los archivos fuentes que estan desactualizados
actualizando ambiente: [nueva configuración] lañado, 0 cambiado, 0 removido
leyendo fuentes... [100%] index
buscando por archivos no actualizados... no encontrado
preparando ambiente... hecho
verificando consistencia... hecho
preparando documentos... hecho
escribiendo salida... [100%] index
generando índices... genindex hecho
escribiendo páginas adicionales... search hecho
copiar archivos estáticos... hecho
copiando archivos extras... hecho
volcar el índice de búsqueda en English (code: en)... hecho
volcar inventario de objetos... hecho
construir exitoso.

Las páginas HTML están en build/html.

```

Fig C. Comando make clean html

```

(envDoc) dent@equipo:~/Escritorio/EIP-2022/BuenasPractksProgPyton/Lección 3. Programación dirigida por la documentación/trabajos/puntoDos$ sphinx-apidoc -o docs/ code
Creando archivo docs/userService.rst.
Creando archivo docs/modules.rst.

```

Fig D. Comando sphinx-apidoc -o docs/ code

```

(envDoc) dent@equipo:~/Escritorio/EIP-2022/BuenasPractksProgPyton/Lección 3. Programación dirigida por la documentación/trabajos/puntoDos/docs$ make html
Ejecutando Sphinx v5.2.2
cargando el ambiente pickled... hecho
compilando [mo]: los objetivos para 0 los archivos po que estan desactualizados
compilando [html]: los objetivos para 0 los archivos fuentes que estan desactualizados
actualizando ambiente: 0añadido, 2 cambiado, 0 removido
leyendo fuentes... [100%] userService

```

Fig E. Comando make html

Lección 3. Programación dirigida por la documentación

CONTENTS:

- code

userService module

- Clase de servicio UsuarioServicio
 - UsuarioServicio

» code » userService module [View page source](#)

userService module

Clase de servicio UsuarioServicio

Esta clase pertenece a la capa de servicio en el modelo de capas, conecta las instancias de objetos tipo UsuarioServicio con la capa de persistencia de datos para resolver las distintas consultas

```
class userService.UsuarioServicio(id, nombre, clave, edad, casado) \[source\]
```

Bases: `object`

Clase UsuarioServicio resuelve las reglas del negocio, consultando y actualizando a la base de datos sobre la información de usuario

```
>>> from userService import UsuarioServicio
>>> import json
>>> user = UsuarioServicio(1,"willyUser","WillyPass",23,False)
>>> print(user.__repr__())
UsuarioServicio(id = 1, nombre = willyUser, clave = WillyPass, edad = 23, sasado = False)
>>> print(user.__str__())
UsuarioServicio(id = 1 nombre = willyUser clave = WillyPass edad = 23 sasado = False)
>>> user = json.dumps(user.__dict__)
>>> print(user)
{"id": 1, "nombre": "willyUser", "clave": "WillyPass", "edad": 23, "casado": false}
```

[getListUsers\(\)](#) [\[source\]](#)

```
>>> print(getListUsers())
[
  {
    "id": 1,
    "nombre": "willyUser1",
    "clave": "WillyPass1",
    "edad": 23,
    "casado": false
  },
  {
    "id": 2,
    "nombre": "willyUser2",
    "clave": "WillyPass2",
    "edad": 19,
    "casado": true
  }
]
```

Fig E. Documento HTML a partir del fichero userService.py

🏠 Lección 3. Programación dirigida por la documentación

Search docs

CONTENTS:
code

userService module

Clase de servicio UsuarioServicio

⊕ UsuarioServicio

getListUsers() [\[source\]](#)

```
>>> print(getListUsers())
[
  {
    "id": 1,
    "nombre": "willyUser1",
    "clave": "WillyPass1",
    "edad": 23,
    "casado": false
  },
  {
    "id": 2,
    "nombre": "willyUser2",
    "clave": "WillyPass2",
    "edad": 19,
    "casado": true
  },
  {
    "id": 3,
    "nombre": "willyUser3",
    "clave": "WillyPass3",
    "edad": 34,
    "casado": false
  }
]
```

Este método se encarga de entregar al objeto que la invoca, todos los usuarios almacenados en la base de datos

Parameters: `self (UsuarioServicio)` – Variable usada para instanciar los atributos de la clase UsuarioServicio

Return lista usuario: Objeto que contiene un mensaje de advertencia por no encontrar el id solicitado

getUsuarioById(id) [\[source\]](#)

```
>>> print(getUsuarioById(1))
{
  "id": 1,
  "nombre": "willyUser",
  "clave": "WillyPass",
  "edad": 23,
  "casado": false
}
```

Fig F. Documento HTML a partir del fichero userService.py