



Introducción al Desarrollo de Software I - Lanzillotta

Capra - Chaves - Di Matteo - Sosa - Villegas - Palavecino

Agenda

- Flask
- Ejemplo práctico

Flask

Flask

- Flask es un framework minimalista web muy popular que deja casi todas las decisiones de diseño y arquitectura en manos del desarrollador.
- Para trabajar con distintos proyectos es necesario crear un ambiente virtual (virtual environment), esto permitirá que no hayan incompatibilidades entre diferentes versiones de dependencias instaladas entre un proyecto y otro.
- pipenv

<https://docs.python-guide.org/dev/virtualenvs/>

Creando un proyecto Flask

1- Crear un directorio donde usaremos el proyecto

`mkdir mi_proyecto`

2- Acceder al directorio

`cd mi_proyecto`

3- Ingresar a Visual Studio Code

`code .` (crear el archivo `app.py` y un directorio `.venv`)

4- Crear el ambiente virtual

`pipenv install flask`

Se instalarán las dependencias en los archivos `pipfile` y `pipfile.lock`

Creando un proyecto Flask

5- Activamos el proyecto

pipenv shell

6- Continuamos configurando el proyecto dentro del archivo app.py

from Flask import Flask

Estructura básica de un proyecto (Sin Blueprints)

/mi_proyecto

app.py

/.venv

/static

style.css

/templates

layout.html

index.html

login.html

...

Jinja nos permite agregar funcionalidad a estos templates, haciendo que puedan integrar data desde el template dinámicamente.

Arrancando el proyecto

```
app.py > ...  
1  from flask import Flask, render_template, request, redirect, url_for  
2  
3  app = Flask(__name__)  
4  
5  @app.route("/")  
6  def hello_msg():  
7      return render_template('home.html', nombre="Bruno")  
8  
9  
10 if __name__ == "__main__":  
11     app.run("127.0.0.1", port="8080", debug=True)  
12
```

Inicialización de Flask en nuestro proyecto

Decorator @, indicando la ruta de acceso a la función

Setup del servidor

Render de Jinja para abrir la vista del template

Templates :: Variables y estructuras

Dentro de un template, además de tener código HTML, también podremos incorporar en forma dinámica **variables y estructuras** tipo if o for, entre muchas cosas más.

- **Variables**

Se incorporarán en forma dinámica su valor cuando las llamemos

{{ nombre_variable }}

Variables dentro de los Templates

- Los templates pueden soportar muchos tipos de variables, por ejemplo:

Diccionarios, listas, cadenas, objetos, enteros

`<p>A value from a dictionary: {{ mydict['key'] }}.</p>`

`<p>A value from a list: {{ mylist[3] }}.</p>`

`<p>A value from a list, with a variable index: {{ mylist[myintvar] }}.</p>`

`<p>A value from an object's method: {{ myobj.somemethod() }}.</p>`

Estructuras If dentro de los Templates

- Como dijimos también pueden existir estructuras del tipo condicionales If

Las mismas de deberán estar entre `{% %}`

```
{% if user %}  
    Hello, {{ user }}!  
{% else %}  
    Hello, Stranger!  
{% endif %}
```

Estructuras for dentro de los Templates

- También se podrían utilizar for para hacer iterar una lista o diccionario de datos.

Las mismas de deberán estar entre {% %}

```
<ul>
    {% for comment in comments %}
        <li>{{ comment }}</li>
    {% endfor %}
</ul>
```

Links dentro de los templates

- Los links dentro de los templates se recomienda que se use el método **url_for()**, de esta forma evitaremos posibles problemas o roturas de nuestro código.
- Los links pueden ser estáticos o dinámicos.

Ejemplos:

url_for('index', _external=True) ⇒ <http://localhost:5000/>

url_for('user', name='john', _external=True) ⇒ <http://localhost:5000/user/john>

(la opción `_external=True`, hace que se escriba la dirección absoluta)

Pongamos en marcha el proyecto

Crearemos un pequeño sitio web y veremos:

- **carpetas/directorios necesarios**
- **exports para arrancar el proyecto**
- **funciones principales para iniciar el proyecto**
- **cómo enviar variables desde el código Python al frontend y viceversa (GET y POST)**
- **variable session (stored in the server)**

Ejercitación

Ejercicio 1

- 1- Crear un primer proyecto en Flask. Escribir en el navegador “Hola Mundo”
- 2- Continuar con este mismo proyecto, y en la vista donde escribieron hola mundo, incorporar debajo un link que lleve a una nueva vista llamada howiam.html que tenga un texto con una breve reseña del alumno.