# Instalación Web Service Python en Ubuntu (Parte 1)

Para la instalación del Web Service Python será usando Python y un micro-framework de nombre Flask y, para empezar:

1. Crearemos un directorio desde la terminal de Ubuntu, con el nombre que desee con el comando

***$ mkdir nombre-carpeta***

Donde mkdir es la instrucción para crear directorios en Ubuntu y nombre-carpeta será el nombre que desee ponerle a su carpeta donde se trabajará todo el proyecto.

1. Después de crear la carpeta se entrará a ella (Ilustración 1) y por consiguiente se instalará el micro-framework antes mencionado con los siguientes comandos:

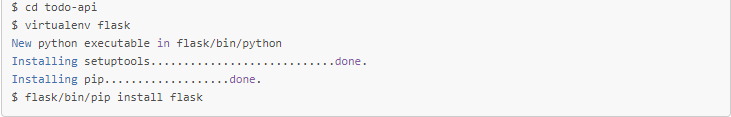


Ilustración 1 Entrada a la carpeta e instalación de micro-framework

Nota: El comando “*virtualenv flask*” no existe en la lista de comando de Ubuntu por lo que en ese caso se escribirá “*sudo apt-get install virtualenv flask*” y este se instalará correctamente.

1. Lo siguiente será ejecutar comando que permitirá el entorno de trabajo de Flask:



Ilustración 2 Comando de instalación del entorno Flask

1. Una vez instalado el entorno Flask se creará un nuevo archivo python en el cual permitirá montar el Web Service con el siguiente comando:

***$ nano app.py***

Donde nano es el editor de texto y app.py el nombre del archivo con su extensión de guardado, después ese abrirá su interfaz para poder empezar a escribir el código de python correspondiente.

1. Una vez dentro del nuevo archivo python se escribe el siguiente código para la demostración del funcionamiento del Web Service:



Ilustración 3 Código de ejemplo para el ver el funcionamiento correcto del Web Service

1. Para salir del editor nano presionamos Ctrl + X, pedirá si se desea guardar los cambios, se coloca la letra Y, se nombra al archivo de la siguiente manera app.py, y listo el archivo del proyecto quedo guardado en la ubicación en donde se ejecutó nano.

Ahora para poder ejecutar el archivo de nombre app.py, en la terminal de Ubuntu estando en la carpeta donde se encuentre guardado el archivo .py, se escribe los siguientes comandos:

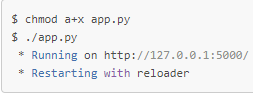


Ilustración 4 Ejecutando el archivo app.py

Como se ve la Ilustración 4 al momento de ejecutar el comando “*./app.py*” este empezara a ejecutarse de forma automática, ahora solo queda ir al navegador y escribir la dirección de URL: <http://localhost:5000> para poder ver el archivo .py en acción.

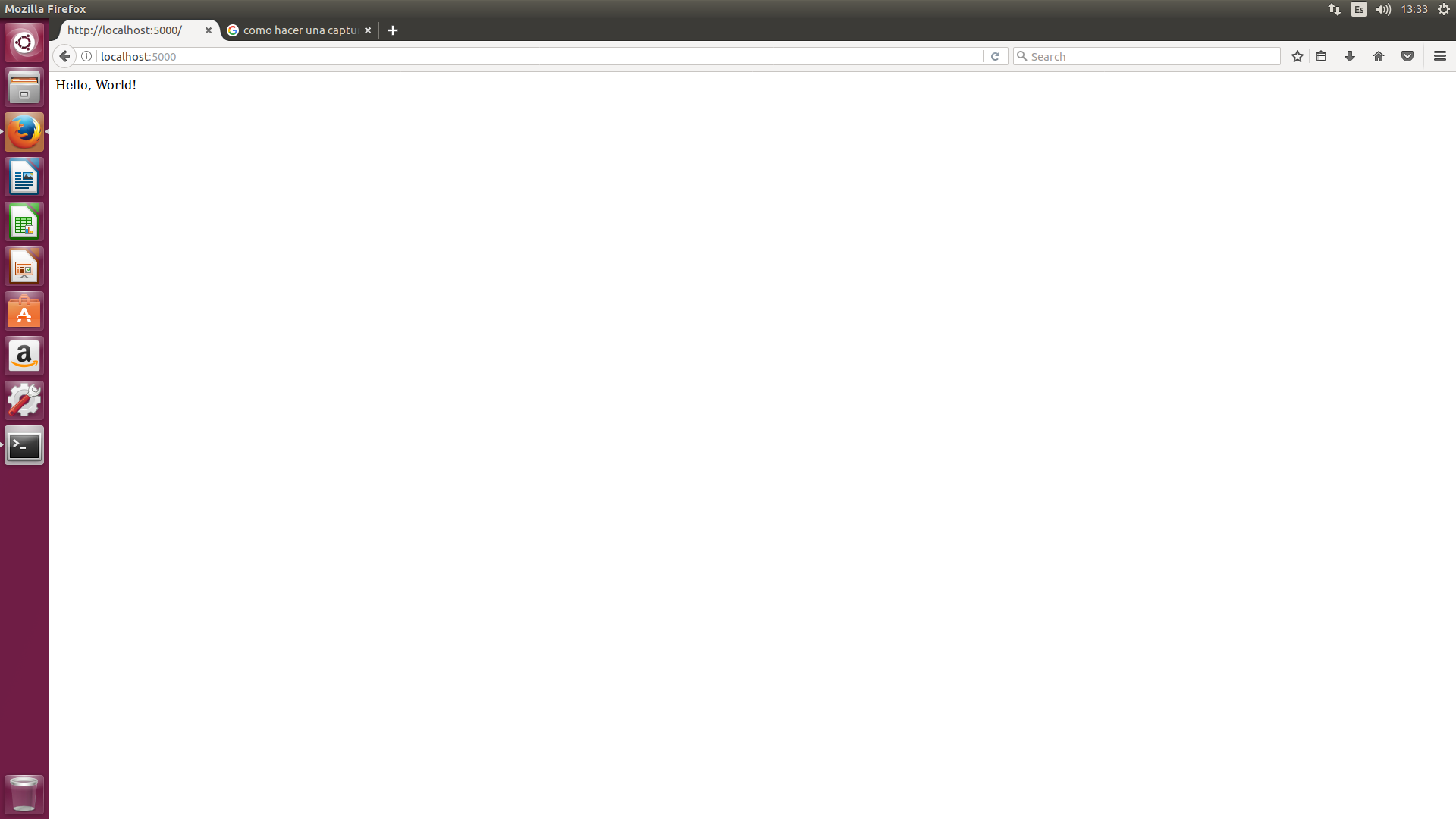


Ilustración 5 App.py ejecutándose

# Implementación del servicio

Como prueba del servicio implementaremos una lista de tareas en una estructura de memoria envés de usar una Base de datos que es el propósito del proyecto.

Esto sólo funcionará cuando el Web Server que ejecute la API sea un solo proceso y un único subproceso, es por ello que prepararemos una bases de datos adecuada.

Utilizando la aplicación de Flask base ya está listo para la implementación del primer punto de entrada al sitio web.

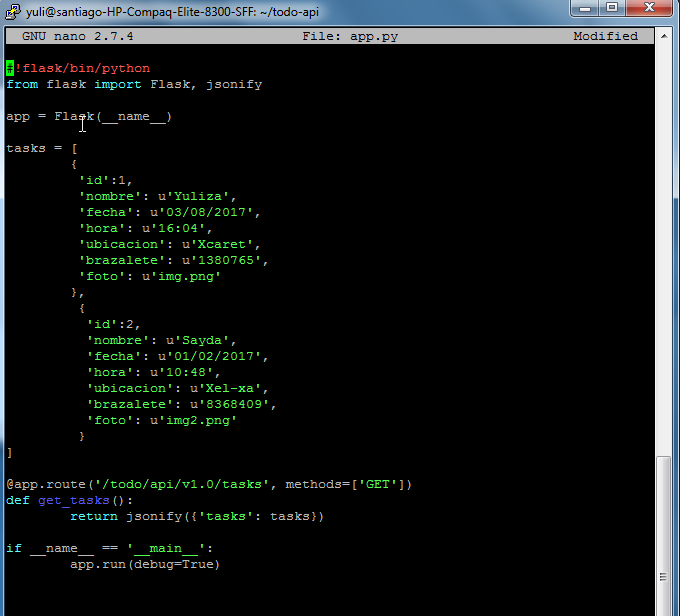


Ilustración 6 Primer punto de entrada al sitio web.

Como se puede observar la parte que se modifico es donde se crea una base de datos de memoria de tareas, que es conocido como una serie de diccionario.

En lugar de un index ahora tendremos un get\_tasks, función que es sólo para el GET método de HTTP.