Utilicé el lenguaje de programación **Python**, junto con el framework de **Flask** quien hace que a través de pocas líneas de código, podamos desplegar un SW de manera fácil y rápida. Como lenguaje de marcado usé **HTML**.

Los protocolos que se utilizan en la implementación de un Servidor web es HTTP con requests/solicitudes de tipo Get.

* Para comenzar debemos crear tres archivos, uno donde se ubicará el HTML de la página (index.html), otro que será el texto donde se almacenarán los parámetros de la URL, y otro .py donde estará el código para desplegar el SW.

El archivo .py:

Básicamente si analizamos la solución, lo que el programa hace, antes que nada importa librerías específicas de flask que son fundamentales, luego se puede apreciar que hace que el directorio raíz cuando se ejecuta, una función lo redirecciona indicándonos “\*Ingrese su mensaje\*” (cosa que no saldrá impresa en pantalla).

Luego se establece que lo que se escriba al lado del igual, será impreso en la pantalla. La función msg recibe como parámetro, el parámetro de la URL. Generamos un if en el que si el mensaje es distinto a “\*Ingrese su mensaje\*”, abre el archivo txt, y appendea (agrega) el parámetro del mensaje como la variable file, asi se usa la función write() que escribe el mensaje en el txt, y /n que hace el salto de línea. Después con la función .close() hace que se cierre el archivo. Fuera del if se abre el archivo en modo lectura (r de read). Y retorna con la función de Flask render\_template() el index.html, con el parámetro determinado adentro del archivo txt en modo lectura.

El archivo de texto:

Se encuentra vacío, y a medida que se le agregan parámetros a <mensaje>, se guarda en este archivo, para luego ser ejecutado, leído y mostrado en la página.

El archivo index.html:

Se ubica adentro de la carpeta templates, porque allí es donde la función de Flask lo busca.

Dentro de este archivo nos encontramos con código default de HTML que viene predeterminado, con un tittle de “WALL PIRATA” y luego en el dentro del body tenemos la etiqueta **pre** que hace que se respeten los espacios originales de los archivos, y los saltos de lineas. Por último dentro de esta etiqueta tenemos {{ mensaje }} que hace que funcione el parámetro del URL en el archivo .py.

\*Instalar Visual Studio Code

\*Python viene instalado en Debian

\*Instalar Flask- adentro de la

consola ponemos

pip install flask (suponiendo

que instalamos previamente

pip)

Con esto estamos listo para comenzar el proceso de desplegar nuestro primer WS