**Instrucciones para la ejecución de la aplicación.**

Para la correcta ejecución de la aplicación se necesita tener instalado:

* Python versión 2.7 o superior. (Personalmente tenía instalado en mi computadora la versión 3.6.).
* Herramienta “pip” para poder instalar las librerías correspondientes.
* Librerías “Psutil”, “Flask” y “Platform”.

**Descripción de la aplicación realizada**

La aplicación realizada consiste en 3 módulos desarrollados en Python.

El primero (llamado ‘*fuente.py*’) cuenta con variables en las cuales almaceno información, obtenida del sistema operativo, para luego guardarlas en un diccionario. La forma de obtenerla es mediante el uso de las librerías “psutil” y “platform”. Dicha información trata sobre el nombre del procesador, el nombre del sistema operativo, la versión de este, la lista de los usuarios actualmente conectados y la lista de todos los procesos corriendo en el sistema.

El segundo módulo (llamado *‘información.py’*) es donde importo todas las variables del módulo ‘*fuente.py’* y las almaceno en un diccionario.

Por último, el tercer módulo (llamado ‘*app.py’*) es donde, gracias a la librería “Flask”, defino las rutas en donde posteriormente voy a ingresar para obtener como respuesta los datos solicitados en el challenge. En el mismo establezco 2 funciones: La primera es para testear la ruta para ver si esta funcionando correctamente. La segunda es para que, al ingresar la ruta “*localhost:5000/información*”, me devuelva la información obtenida en el primer módulo, y además me guarde la misma en un archivo “.json”.

Los principales problemas y soluciones que se me presentaron al realizar el challenge son:

* Fue mi primera vez usando “Unix” y, por lo tanto, tuve que aprender a cómo se utilizaba. Al principio del challenge pensé en crear un script de bash y luego, mediante un comando *‘curl’,* enviarlo a la aplicación, pero, después de comentarle esta idea a Juan, él me recomendó que checkeara la librería “psutil” y me fue de gran ayuda.
* También fue mi primera vez creando un servidor en AWS, y tuve que capacitarme viendo videos, o leyendo bibliografía de Amazon, para poder crear uno y que funcionara correctamente. Una vez dentro del servidor, tuve problemas para instalar la herramienta “pip”, pero se soluciono modificando la variable “PATH”.
* Al momento de correr mi aplicación en el servidor de AWS, la misma corría correctamente (como se ve en el print que adjunto en el mail), pero cuando quería entrar a la ruta desde internet (o realizar un -XGET en la consola de Unix) no me devolvía la lista, si no que me arrojaba un error. Este problema intenté solucionarlo cambiándole el puerto, pero no pude encontrarle solución. Me gustaría saber cual fue mi error y como es la manera correcta de hacerlo.
* Al momento de guardar la información en el archivo con extensión “.json”, no pude lograr que el archivo sea nombrado con la IP del servidor y la fecha. Me gustaría saber como se realiza esto, dado que intente con variables, o importando funciones de librerías de fecha/hora, y no pude hacerlo.
* Por último, intenté dockerizar la aplicación, pero solo escribí las líneas de código en su módulo correspondiente, ya que nunca había dockerizado anteriormente, e hice esto para saber si es la manera correcta de hacerlo. Creo que es un concepto que es de gran ayuda cuando se está trabajando con mas de un proyecto a la vez, y cuando se comparten proyectos entre desarrolladores.

La principal documentación utilizada fue la siguiente:

<https://www.softzone.es/programas/sistema/crear-configurar-maquina-virtual-virtualbox/>

https://docs.aws.amazon.com/es\_es/AmazonRDS/latest/UserGuide/CHAP\_Tutorials.WebServerDB.CreateWebServer.html

<https://docs.aws.amazon.com/elasticbeanstalk/latest/dg/eb-cli3-install-linux.html>

<https://aws.amazon.com/es/docker/>

https://hub.docker.com/\_/debian/

https://j2logo.com/flask/tutorial-como-crear-api-rest-python-con-flask/