

# Implementación de un cliente multiplataforma en Python para el procesado remoto de imagen médica

Autores:

González Carrizo, Rafael ([ragonca2@inf.upv.es](mailto:ragonca2@inf.upv.es))

Ribera Vázquez, Víctor ([vicrivaz@inf.upv.es](mailto:vicrivaz@inf.upv.es))

## Resumen:

El objetivo es crear una herramienta multiplataforma para ser utilizada por los médicos y que les facilite el trabajo de subir los ficheros a un servidor, esperar a que estos se procesen y tener que descargarlo, automatizando este proceso y haciéndolo transparente para el usuario, y por supuesto, configurable. Las características principales de este cliente son:

- Poder subir varios archivos a la vez al servidor con un click
- Poder seleccionar el pipeline de trabajo, así como dar la edad y genero del paciente
- Poder elegir si queremos que los archivos subidos al servidor se descarguen automáticamente.
- Poder elegir en que carpeta queremos que se descarguen los archivos
- Poder elegir si queremos descargar todo el resultado o solo una parte de él

En cuanto a la interfaz, esta se adapta al entorno en el que este, así pues, aunque se parecerá en todos los dispositivos, tendrá colores o menús diferentes dependiendo del sistema operativo o el entorno de escritorio instalado.

## Introducción: justificación y aspectos generales

La necesidad de esta herramienta nace por los usuarios del portal volBrain, que normalmente son médicos que tienen poca experiencia en el campo de la ofimática, y aunque son capaces de utilizar dicho portal, es un trabajo pesado y repetitivo, sobre todo cuando la cantidad de pacientes a tratar es masiva. Esta herramienta tiene como objetivo automatizar todo este proceso para que sea lo más simple y transparente para el usuario, sin renunciar a ninguna característica del portal. Como se ha mencionado en el resumen, seremos capaces de subir muchos ficheros a la vez y automatizar la descarga de ficheros cuando estos terminen de procesarse en el servidor sin necesidad de la interacción con la herramienta.

Cuando hablamos de que la herramienta es multiplataforma, nos referimos siempre en el contexto de ordenadores de sobremesa y portátiles, no se desarrollará una herramienta para sistemas móviles, pues no es la herramienta común de uso de los usuarios objetivos.

## **Materiales:**

Los materiales de los que se disponían eran una cuenta en el portal de volBrain para poder hacer pruebas, un script en Python que era capaz de subir un archivo al servidor de volBrain, el programa QtDesigner junto a su librería de Python, **pyqt5**, y, por supuesto, Python.

## **Datos:**

Los datos de los que se disponían era un único caso de un paciente anónimo. Esto era más que suficiente, pues el objetivo de este dato era únicamente hacer pruebas con el servidor, sin importar cual fuera el resultado, mientras fuera correcto.

## **Preproceso anterior a experimentos:**

Antes y durante el desarrollo de la herramienta, se hicieron numerosos experimentos para ver como funcionaba realmente el portal de volBrain, pues no solo había que subir el fichero, sino que había que elegir las pipelines, la edad y el género, y en cuanto a descargar los archivos, ocurrieron cosas que era imposible saberlas sin probarlo antes como por ejemplo que los resultados se borran 30 días después de su subida, que si hay muchos resultados hay diferentes páginas de resultados, etc.

## **Resultados:**

Los resultados obtenidos fueron muy favorables, pues se consiguió la herramienta deseada con todos los requisitos funcionales y no funcionales que se habían propuesto. Esta herramienta fue probada en diferentes distribuciones de Linux y una de Windows con resultados muy positivos y sin ningún fallo. El resultado es un

cliente listo para entregar al usuario final y este no requiere enseñanza alguna, pues su uso es tan sencillo que no lo requiere.

## Discusión:

Aunque se han cumplido todos los objetivos, esta herramienta puede conllevar varias conclusiones negativas que se podrían solucionar en futuras correcciones. La primera viene como consecuencia de un requisito no funcional, la facilidad de uso, al ser tan sencillo y rápido subir archivos, existe el peligro de que el usuario este subiendo constantemente el mismo archivo una y otra vez sin siquiera darse cuenta. Esto tiene dos soluciones, prohibir la subida de archivos aparentemente iguales desde el cliente, o cambiar el servidor de volBrain para que guarde un registro de los archivos que se han subido y rechace esas peticiones repetitivas. La segunda es la posible confusión por la descarga automática de los archivos, es posible que a la hora de subir varios archivos, al descargarse el usuario confunda unos con otros si tienen nombres parecidos o incluso no se haya dado cuenta de cual es cual, pues los archivos subidos no se procesan en el orden que se suben, lo que puede producir confusión.

## Conclusión:

Para concluir, decir que, aunque se han encontrado vulnerabilidades como las anteriormente comentadas, no es algo por lo que preocuparse a la hora de poner este producto en manos de los usuarios finales, pues si es verdad que en algún momento puede ocurrir algo no deseado, esto no debería generalizarse y no se verán grandes efectos ni el usuario ni el servidor.

## Referencias:

[1]: Python → <https://www.python.org/>

[2]: PyQt → <https://riverbankcomputing.com/software/pyqt/intro>

[3]: volBrain → <http://volbrain.upv.es/>

[4]: BeautifulSoup → <https://www.crummy.com/software/BeautifulSoup/>