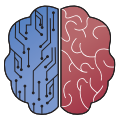
UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería de la Salud



INGENIERÍA
DE LA SALUD

**TFG del Grado en Ingeniería de la Salud**

# Automatización de extracción de datos de informes de secuenciación masiva y análisis.

Presentado por Lucía Vítores López
en Universidad de Burgos

26 de mayo de 2023

Tutor: Antonio Jesús Canepa Oneto

UNIVERSIDAD DE BURGOS

ESCUELA POLITÉCNICA SUPERIOR

Grado en Ingeniería de la Salud

D. Antonio Jesús Canepa Oneto, Departamento de Ingeniería Informática, área de Lenguajes y Sistemas Informáticos.

Expone:

Que el alumno D. Lucía Vítores López, con DNI 71970531N, ha realizado el Trabajo final de Grado en Ingeniería de la Salud titulado: Automatización de extracción de datos de informes de secuenciación masiva y análisis.

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 26 de mayo de 2023

Vº. Bº. del Tutor:

D. Antonio Jesús Canepa Oneto

## Resumen

Este Trabajo de Fin de Grado se centra en la automatización de la extracción de datos en el servicio de anatomía patológica del Hospital universitario de Burgos (HUBU).

La extracción manual de datos es un proceso muy costoso donde es muy probable que se cometan errores. Sin embargo, el uso de automatización puede mejorar tanto la eficiencia como la precisión del proceso. El estudio se basa en implementar un sistema automático para la extracción de una serie de datos de especial interés de distintos archivos, utilizando técnicas de procesamiento.

Los resultados muestran que la automatización proporciona muchas ventajas y mejoras a la hora de trabajar con los resultados, aportando información más precisa, concreta y exacta sobre el cáncer para estudiar los distintos tipos y genes y no perder tiempo escogiendo los datos.



## Descriptores

Anatomía patológica, cáncer, genes, automatizar, extracción de datos, Python, GitHub ...

## Abstract

This final degree project is about data extraction automation in the Pathological Anatomy Service at Burgos Hospital.

Manual extraction is a difficult job and often prone to mistakes. However, the use of automation can improve efficiency and accuracy of the process. The objective of this project is to implement an automatic system for retrieve information from various files using different processing techniques.

Results show that implementing automation has a lot of benefits and improvements when working with results. It provides specific information about cancer to study different types and genes and it's an easy way to not waste time.



## Keywords

Pathological anatomy, cancer, genes, automate, data extraction, Python, GitHub ...

# Índice general

# Índice de figuras

# Índice de tablas

# Objetivos

En este apartado aclararemos los objetivos generales del proyecto, así como las posibles dudas sobre su finalidad.

El objetivo general del presente proyecto es solucionar el problema de obtención manual de los datos en la máquina Oncomine Reporter del servicio de Anatomía Patológica en el Hospital Universitario de Burgos (HUBU) mediante la automatización de la extracción de los datos, partiendo de los campos más relevantes para el manejo de datos de secuenciación masiva generados directamente por el software comercial Oncomine Reporter en un informe en formato Portable Document Format (PDF) poco adecuado para su tratamiento.

Para conseguir este objetivo general, se ha dividido en distintos objetivos más pequeños que al unirlos permiten la creación final del proyecto.

## 1.1. Objetivos marcados por software, hardware o análisis.

Los principales objetivos que abarcan este apartado son:

- Desarrollo de un código capaz de facilitar la extracción de datos de ficheros PDF.
- Lograr una interfaz sencilla para que cualquier persona sea capaz de usarla correctamente y obtener los resultados esperados, con el fin de facilitar el estudio.

## 1.2. Objetivos técnicos.

El proyecto fue fragmentado en pequeños bloques de trabajo, que al unirlos han creado el proyecto final completo.

- Revisión bibliográfica sobre PDFscraping. **??**
- Creación de código sencillo que seleccione las variables de interés de un PDF usando técnicas de PDFscraping.
- Lectura de todos los PDF proporcionados para la obtención de la información.
- Separación de la información en tablas específicas.
- Obtención de varios archivos .CSV con los resultados.
- Creación de material para ayudar a su implementación en otros dispositivos.

## 1.3. Objetivos de aprendizaje.

Los principales objetivos con los que se pretende finalizar el código son:

- Adquisición de nuevos conocimientos en el ámbito anatomopatológico.
- Aumento de conocimiento en el ámbito de programación.
- Uso de distintas herramientas para un resultado final adecuado.
- Comprensión de la información biológica para su aplicación en el desarrollo del código.

# Introducción.

## 2.1. Conceptos teóricos básicos.

Vamos a proceder a explicar los conceptos teóricos básicos para el correcto entendimiento del trabajo.

### Anatomía patológica.

La anatomía patológica es una rama de la Medicina que se encarga principalmente del estudio de los efectos que produce una enfermedad en los distintos órganos del cuerpo, tanto en aspectos macroscópicos como microscópicos, con el fin de diagnosticar y tratar las distintas enfermedades. [30]

Todas las especialidades médicas necesitan la información de este servicio para llevar a cabo sus respectivos procedimientos, por lo que el informe anatomopatológico final que se genera con toda la información obtenida a través de los exámenes de las muestras del tejido o de las células mediante distintos procedimientos es de gran importancia.

En el servicio de Anatomía Patológica del Hospital Universitario de Burgos se usa el software Oncomine Reporter, que solamente entrega sus resultados en formato PDF impidiendo la manipulación de los datos de forma directa y dificultando el trabajo del investigador. Para ello, su solución actual es crear un archivo Excel en el que se van introduciendo los datos de interés de forma manual para así poder trabajar sobre ellos. Sin embargo, es un trabajo muy laborioso porque se trabajan con grandes cantidades de datos al día.

Es por ello que la creación de este proyecto ayudará al almacenamiento de los datos de una forma automática, rápida y organizada, evitando que sean los propios investigadores los que tengan que realizar este proceso manualmente, con el fin de facilitar la tarea a los profesionales y la comprensión de los resultados obtenidos.

### Herramientas

### Automatización de extracción de datos.

La extracción de datos se basa en la obtención de información de distintas fuentes para posteriormente usarla con distintos fines.

La finalidad de automatizar este proceso es ahorrar tiempo y recursos a la hora de almacenar la información, así como minimizar errores y facilitar el trabajo.

La automatización de la extracción de los datos es un procedimiento generalmente útil y que muchas empresas están empezando a usar, ya que esto permite recopilar y trabajar con grandes cantidades de datos pertenecientes a varias fuentes distintas para contrastarlas en el menor tiempo posible y filtrar aquellas de mayor interés. [39]

Aunque también es posible enfocarlo al ámbito médico [11] para trabajar con datos reales de pacientes, ya que la gran cantidad de datos con los que se trabaja habitualmente hace necesario sacarles el mayor provecho posible para mejorar la salud pública. Con este procedimiento de automatización de la extracción, se pretende anticiparse a los hechos y tener unos resultados lo más seguros y precisos posibles. Algunas de las ventajas que presentaría son la mejora de la calidad de la atención, el aumento de la productividad o la satisfacción de un mayor número de pacientes entre otros.

Por esto y muchas más ventajas es necesario automatizar los datos, es decir, ser capaz de trabajar con dichos datos usando herramientas en lugar de hacerlo de forma manual, como actualmente ocurre en el HUBU.

## 2.2. Estado del arte y trabajos relacionados.

La automatización de la extracción de datos es un tema que está adquiriendo cada vez más importancia tanto en la academia como en la industria.

Algunas de las herramientas más destacados son:

1. **Astera ReportMiner**: es un software de extracción de datos automatizado a nivel empresaria que extrae datos no estructurados de archivos PDF a una base de datos con funciones de limpieza y programación integradas. Las herramientas también pueden automatizar el proceso de extracción de los datos y cargarlos o almacenarlos en una base de datos o en un archivo de Excel sin el uso de código. Su edición Enterprise Edition también proporciona un programa integrado con funciones en tiempo real capaces de realizar procesos de programación y mantenimiento, mejorando el resultado de la extracción. La información de este apartado hace referencia a la información encontrada en [20] y [15]

2. **Apify**: es una plataforma para automatización y extracción de datos de distintos sitios web, haciendo posible que cualquier persona independiente o empresa, sea capaz de automatizar cualquier flujo de trabajo. También permite a los programadores implementar y monitorizar distintas herramientas de automatización. Proporciona plantillas de código con las que se puede comenzar a desarrollar dichas herramientas siendo posible trabajar en varios lenguajes de programación entre los que se encuentran Python y Java. [13]

3. **Nano**: se trata de una inteligencia artificial capaz de leer documentos semiestructurados o que no siguen una plantilla estándar para extraer sus datos. Aprende y mejora a medida que su uso va aumentando, acortando el tiempo de respuesta. Además, es posible extraer solo aquellos campos de interés y no todo el documento, añadiendo nuevos campos en caso de que sea necesario. Está desarrollado principalmente para empresas, donde los clientes envían los documentos por correo electrónicos y Nanonets es capaz de exportarlo/importarlo a su flujo de trabajo sin interrumpir su sistema. [28]

También existen algunas herramientas más enfocadas al tratamiento de datos médicos.

1. **Apache cTAKES**: es una herramienta de código abierto que usa procesamiento de lenguaje natural (NLP) para extraer la información de textos clínicos no estructurados, siendo capaz de identificar conceptos médicos y relaciones semánticas y así conseguir un resultado más rápido. La información se devuelve en un archivo y en forma de código. Sin embargo, no es muy útil cuando personas sin conocimiento sinformáticos necesitan usarlo y habría que procesar dichos datos antes de usarlos. [19]

2. **Clinithink**: es una empresa de tecnología creada en torno a CLiX, la primera inteligencia artificial de atención médica capaz de comprender los datos médicos no estructurados. CLiX es capaz de generar conocimiento sobre el ser humano, ahorrando tiempo y esfuerzo a las personas a la hora de realizar estos procedimientos. [18] Se puede usar para cualquier tipo de consulta médica, lo que permite un mejor tratamiento individual a cada uno y en su conjunto, unos mejores resultados. Actualmente se esta usando la información que recoge para mejorar algunas áreas.

Vemos que cada vez se requiere más el uso de herramientas que permitan extraer datos de forma automática, pero no siempre existen las mismas necesidades. Por ello es importante que existan distintas herramientas especializadas en los distintos campos.

## Cáncer.

Cáncer (o sus sinónimos tumores malignos o neoplasias malignas) es un término que designa a un gran número de enfermedades que afecta a cualquier parte del organismo.

El cáncer es la principal causa de muerte en el mundo según la Organización Mundial de la Salud. En 2020 se atribuyeron casi 10 millones de muertes. [29] Sin embargo, la OMS ha determinado que alrededor del 30 - 50 % de los casos de cáncer pueden ser evitados reduciendo los factores de riesgo y aplicando estrategias preventivas. A esto hay que añadirle la detección precoz de la enfermedad.

Es una enfermedad bastante difícil de detectar, ya que hay muchas variables y factores que deben tenerse en cuenta y son específicos en cada persona. En las primeras etapas los síntomas son poco notables o incluso pueden llegar a no detectarse, también es posible que sean confundidos con los síntomas de otras enfermedades.[29]

El cáncer puede comenzar en cualquier parte del cuerpo, ya que este está formado por billones de células. En condiciones normales, las células humanas se multiplican mediante división celular. Cuando éstas envejecen o se dañan, mueren y son reemplazadas por las nuevas. [26] Sin embargo, con el paso de los años, la capacidad del cuerpo de realizar esta función va desapareciendo, haciendo que las células dañadas se vuelvan cancerosas. Lo que explica que con la edad aumente la probabilidad de desarrollar algún tipo de cáncer. Cuando las células dañadas se multiplican en lugar de ser eliminadas, se llegan a formar tumores (benignos o malignos).

Los tumores malignos son capaces de invadir tejidos cercanos o incluso lejanos, proceso conocido como metástasis. El cáncer metastásico tiene el mismo nombre y el mismo tipo de células cancerosas que el cáncer original o primario. Un ejemplo que explica muy bien este proceso se encuentra en NIH y dice:

> *El cáncer de mama que forma un tumor metastásico en el pulmón es cáncer de mama metastásico, no cáncer de pulmón.* [26]

En algunas ocasiones los tumores benignos pueden llegar a ser muy grandes, pero una vez que se extirpan, no vuelen a aparecer. Cosa que no se garantiza con los tumores malignos o cancerígenos, ya que es posible que estos tumores remitan, es decir, que los signos y síntomas de la enfermedad hayan desaparecido por completo pero sigan existiendo células malignas capaces de volver a causar la enfermedad. [25]

Las posibles causas de sufrir esta enfermedad son muy variadas, abarcan genética (hay ciertas mutaciones genéticas que pueden aumentar el riesgo de desarrollo de distintos tipos), edad (el riesgo de cáncer aumenta con la edad, puede deberse a que se van acumulando factores de riesgo o a la incapacidad del cuerpo para eliminar las células cancerosas), estilo de vida (hábitos poco saludables como el tabaco, el alcohol o una vida sedentaria aumenta el riesgo), la exposición a sustancias tóxicas (como productos químicos o radiactivos) o infecciones (virus del papiloma humano VPH o hepatitis B y C).

Los cambios génicos que contribuyen a esta enfermedad suelen afectar a tres genes:

- **Protooncogenes**: genes encargados del crecimiento normal de las células. Sus mutaciones pueden conseguir que se conviertan en genes causantes del cáncer o también conocidos como oncogenes. [23]
- **Genes supresores de tumores**: genes encargados de la codificación de proteínas para controlar la división celular. Cuando estos genes se inactiva, la proteína deja de funcionar correctamente, provocando una división celular descontrolada. [24]
- **Genes de reparación de ADN**: genes encargados de la reparación de los errores cometidos durante la transcripción. [17]

La mayoría de los tipos de cáncer, generalmente vuelven a aparecer en los primeros 5 años tras el tratamiento, aunque también es posible que lo

hagan una vez pasado ese periodo. Debido a esto, es de gran importancia realizar revisiones una vez pasada la enfermedad para ver si los posibles signos o síntomas vuelven a aparecer.

Al tener tanta información sobre algunos genes, los científicos han sido capaces de determinar las mutaciones más comunes en los distintos tipos de cáncer. Lo que conlleva un mayor número de tratamientos dirigidos a esos genes específicos, haciendo posible la aplicación de un tratamiento determinado para cualquier tipo de cáncer siempre que contenga la mutación específica.

Para su tratamiento, es imprescindible un buen diagnóstico, ya que cada tipo de cáncer requiere un tratamiento concreto. Algunos tratamientos son intervenciones quirúrgicas, quimioterapia, radioterapia o terapia sistémica.

El principal objetivo a la hora de aplicar el tratamiento es su cura o la prolongación de la vida del paciente. Sin embargo, hay otros casos en los que el objetivo fundamental es mejorar la calidad de vida del paciente mediante cuidados paliativos.

## Herramientas aplicadas a anatomía patológica.

€Explciar las distintas herramientas y centrarse un poco mas en esta

Oncomine Reporter es una herramienta de software de análisis génico desarrollada específicamente para un examen más profundo de la secuenciación masiva, lo que permite un informe final en pasos muy sencillos. Es el software usado por el servicio de Anatomía Patológica del Hospital Universitario de Burgos y del cuál obtendremos los resultados. Permite una investigación contextual de variantes específicas de la muestra para comprender su uso con respecto a ensayos clínicos globales actuales. Incluye también una gran variedad de flujos de trabajo de análisis reconstruidos y firmas génicas para distintos tipos de cáncer, lo que permite a los investigadores usar dicha información para avanzar en investigación. También es muy útil a la hora de desarrollar nuevos fármacos.

Partiendo de una muestra del paciente, el software es capaz de cargar los datos en una plataforma y usar distintas herramientas o algoritmos para identificar los patrones de expresión génica relevantes para el cáncer. Una vez realizados los análisis, el software devuelve los resultados en una carpeta con distintos ficheros (un fichero por cada muestra de paciente analizada). Estos ficheros son los que nos interesan para llevar a cabo este proyecto.

Sin embargo, uno de los principales inconvenientes es que devuelve falsos casos. Es decir, algunos polimorfismos que son benignos vienen como patológicos, por lo que es importante revisarlo de forma manual para evitar estos fallos.

# Metodología.

## 3.1. Descripción de los datos.

Los datos necesarios para llevar a cabo este trabajo de fin de grado han sido proporcionados por el Hospital Universitario de Burgos.

En el servicio de anatomía patológica del HUBU se usa el software Oncomine Reporter para el análisis de las muestras. Este software solo es capaz de trabajar con 8 muestras a la vez, de forma que el resultado es una única carpeta con 8 archivos Portable Document Format o PDF (cada uno de ellos representa los resultados de una de las muestras analizadas). Cada vez que se ejecute, devolverá una carpeta distinta con 8 archivos en su interior.

Los informes con los que se ha trabajado están disponibles aquí.

Al tratarse de datos de pacientes reales, se ha firmado un acuerdo de confidencialidad en conjunto con el HUBU para no mostrar los datos reales. De este modo, a la hora de presentar el código frente al tribunal, se han usado unos archivos sintéticos a los que se les ha eliminado todo tipo de información, tanto personal como institucional para cumplir dicho acuerdo. Estos archivos han sido aportados por la investigadora del Hospital Universitario con el fin de tener unos modelos similares a los resultantes de Oncomine Reporter con los que poder trabajar y enseñar el resultado. Esto nos permite trabajar de una manera más segura, cumpliendo el acuerdo de confidencialidad de datos.

Sin embargo, al implementar dicho código en el hospital, los técnicos e investigadores podrán trabajar con los datos reales de los pacientes sin ningún problema.

Se realizará una descripción más intensa de los datos en su anexo extra correspondiente. **??**

## 3.2. Lenguajes de programación.

A lo largo del grado se han usado distintos lenguajes de programación, entre los cuales se encuentran Python, R, Java y SQL.

Java es un lenguaje sobre el que se adquieren conocimientos básicos en la carrera, pero no lo suficiente avanzado como para ser capaz de programar dicho algoritmo. Lo mismo ocurre con SQL.

Como Python y R son lenguajes cuya forma de trabajar es muy similar y ambos poseen gran cantidad de bibliotecas enfocadas al análisis y tratamiento de grandes cantidades de datos, inicialmente se estudió la posibilidad de realizar el código en ambos lenguajes.

Sin embargo, como Python ha sido el lenguaje más usado en la carrera y del que más conocimientos se tienen, se eligió esa opción.

## 3.3. Técnicas y herramientas.

Se han usado distintas técnicas metodológicas así como herramientas para el proyecto. Este apartado se centra en explicar las posibles alternativas que han surgido, por qué se han usado y los posibles problemas o dudas que han surgido.

**GitHub** [38] se trata de una de las principales plataformas usadas para la creación de repositorios abiertos. Comenzó a desarrollarse el 19 de octubre de 2007 pero no fue lanzado hasta abril de 2008 por Tom Preston-Werner , Chris Wanstrath , PJ Hyett y Scott Chacon, tras haber estado disponible para unos meses como una versión beta. [35]

Los repositorios pueden ser colaborativos (donde todos los usuarios pueden aportar algo de ayuda para la mejora del código) o privados (donde solo tiene acceso al repositorio el propio colaborador y los invitados que este elija).

Permite a todos los usuarios desarrollar proyectos creando repositorios de forma gratuita, por lo que se trata de proyectos de código abierto. El código abierto hace referencia a que es accesible a todo el público donde pueden ver, modificar, actualizar y distribuir el código de la forma que ellos consideren oportuna.[22] Por lo que el código depende tanto del propio

creador como de los distintos usuarios que forman la comunidad que se encargan de modificarlo o revisarlo.

El repositorio creado para este trabajo se ha denominado Automatizacion_PDF_scraping e inicialmente comenzó privado por tener un reporte previo real, pero una vez se comenzó a trabajar con los archivos sintéticos, se hizo público.

Para poder gestionar los cambios en el proyecto se ha usado Git, un control de versiones distribuido que permite a los colaboradores trabajar de una manera eficiente y tener un historial de las modificaciones realizadas. [37] Permite realizar distintas operaciones usando líneas de comando en la terminal o a través de interfaces gráficas. Principales funciones!!

A su vez, se ha trabajado con GitHub Desktop como aplicación de interfaz gráfica de usuario (GUI) que proporciona una

Se ha creado un branch para trabajar en una rama nueva paralela a la principal, es decir, permite hacer cambios en los archivos sin modificar la rama principal (también conocida como main). [12]

En este caso, la nueva rama ha sido creada para subir los PDF sintéticos, pero como inicialmente estos contenían un nombre real del personal del HUBU, se eliminaron y se volvieron a subir posteriormente los archivos correctos.

Para la organización del contenido de cada semana, se han ido creando distintos Millestones donde se almacenan los Issues, que es donde se explica que objetivos se pretende finalizar en cada periodo de tiempo entre reuniones. Esto se verá correctamente explicado aquí **??**.

Se ha intentado unir la memoria de Overleaf con el propio repositorio, pero no ha sido posible porque no existía una versión gratuita para su uso. Por ello, se han ido subiendo distintos ficheros Portable Document Format con las modificaciones que se han ido realizando para llegar a tener la memoria final.

En algunas de las asignaturas de la carrera ya se había usado dicha plataforma, por lo que ya se tenían unos conocimientos básicos sobre su funcionamiento, pero ha sido necesario la búsqueda de contenido para ser capaz de crear un repositorio totalmente claro y entendible, así como de los comandos necesarios para su uso.

**Anaconda** [6] se trata de una plataforma de distribución y gestión de paquetes tanto para Python como para R, usada principalmente en ciencia de datos, inteligencia artificial o desarrollo de aplicaciones. Fue fundada

en 2012 por Peter Wang y Travis Oliphant a partir de la necesidad de llevar Python al análisis de datos comerciales.[7] Proporciona un entorno de trabajo completo que incluye una amplia colección de paquetes, bibliotecas y herramientas con el fin de facilitar el proceso de desarrollo, análisis y visualización de datos.

Uno de sus paquetes más interesantes es Conda, ya que nos permite instalar, actualizar y administrar fácilmente paquetes y dependencias de software. Además, Anaconda incluye un gestor de entornos virtuales para crear y gestionar entornos de desarrollo aislados. Esto es especialmente útil cuando se trabaja en proyectos con diferentes versiones de paquetes o cuando se desea mantener un entorno limpio y consistente.

La plataforma Anaconda es compatible con múltiples sistemas operativos, como son Windows, macOS o Linux y ofrece una interfaz gráfica de usuario llamada Anaconda Navigator, que facilita la gestión de paquetes y entornos de desarrollo de forma visual e intuitiva.

Anaconda Navigator y Jupyter Notebook son dos herramientas que forman parte de la plataforma Anaconda. Anaconda Navigator es una interfaz gráfica de usuario (GUI) que permite trabajar con entornos virtuales, paquetes y proyectos en Anaconda, así como crear y gestionar entornos, instalar paquetes, y administrar aplicaciones y extensiones. A través de Anaconda Navigator, se accede a Jupyter Notebook.

También es posible acceder a entornos de desarrollo integrados (IDE) que permiten a los programadores consolidar los distintos aspectos de la escritura de un programa, aumentando la productividad combinando distintas actividades dentro de un software [10] como en este caso sería Python. Permiten depurar y editar el código, realizar pruebas...como por ejemplo Spyder o PyCharm. Jupyter Notebook es una aplicación web interactiva para crear y compartir documentos, también conocidos como "notebooks", pero no es considerada una IDE completa. [31] Una vez instalado y abierto Anaconda Navigator, se puede seleccionar Jupyter Notebook para lanzarlo y poder trabajar directamente en este sin tener que realizar ninguna otra instalación mediante la creación de un fichero .ipynb.

Se había creado inicialmente un notebook llamado PruebaLucia.ipynb para crear el código, aunque posteriormente se ha modificado el nombre a CodigoPython.ipynb.

**LaTeX** es un sistema de composición de documentos basado en TeX, que usa Overleaf como plataforma en línea, ya que ofrece un entorno colaborativo para editar y compilar documentos LaTeX sin requerir instalaciones locales.

LaTeX fue desarrollado Leslie Lamport en 1980, siendo esta una extensión del sistema de composición de documentos TeX, desarrollado por Donald Knuth en 1970. Aunque desde entonces ha experimentado grandes mejoras. Overleaf fue desarrollado por John Hammersley y John Lees-Miller y lanzada públicamente en 2012, con el objetivo de proporcionar una solución fácil y colaborativa para la edición de documentos LaTeX en la nube. [36]

Es un sistema para crear textos estructurados o con fórmulas matemáticas, siendo Overleaf su entorno de edición en línea. Se usa principalmente en textos donde lo importante es el texto y su estructura, no el tipo de letra o el salto de página. Esta y parte de la información encontrada posteriormente ha sido obtenida en parte de [4]

Los documentos se escriben en texto plano, por lo que existen distintos parámetros o caracteres especiales para usar distintos comandos, como barras bajas (_)para subíndices o almohadilla (%) para comentarios. El encabezado debe tener unas instrucciones claras para determinar cosas como el idioma, tipo y tamaño de letra... Además debe incluir comando para inicio y fin del documento (begin document/ end document). Entre estos dos comandos se debe desarrollar el informe, usando las secciones y subsecciones necesarias para su organización. También es posible añadir imágenes o tablas para aclarar el contenido del texto, hacer enumeraciones o introducir fórmulas matemáticas.

Un comportamiento que tiene por defecto LaTeX es que prefiere que una palabra sobrepase el margen derecho a tener que pasarlo a la siguiente línea y dejar un hueco demasiado grande. Para ello existe un comando especial denominado sloppy que permite el salto de línea de dicha palabra.

El paquete BibTeX nos permite almacenar las URLs los distintos libros o páginas web que se han utilizado para buscar la información del proyecto y hacer referencias en el texto mediante identificadores. Para ello se crea un archivo .bib nuevo (en este caso llamado bibliografia.bib para la bibliografía de la memoria y otro llamado bibliografiaAnexos.bib para la bibliografía de los anexos) donde se almacenan todos los datos para tenerlos registrados y poder crear la bibliografía final.

Cuando recompilamos, podemos observar como quedaría toda la documentación organizada en el archivo. En el caso de que haya algún error, LaTeX es capaz de indicarlo. Es posible imprimir el documento final e incluso pasarlo a otro tipo de archivo.

**Excel** [9] es una hoja de cálculo de Microsoft que nos permite trabajar tanto con datos numéricos como texto en distintas tablas formadas por líneas

y columnas. No fue desarrollado por Bill Gates, sino por un programador llamado Charles Simonyi, al que también se le atribuye el desarrollo de otros productos del software de Microsoft. [8]

Estas hojas de cálculo nos permite analizar o realizar distintas acciones mediante gráficos o tablas.

Un archivo .xls es un archivo de hoja de cálculo que puede ser creado con Excel o con otras hojas de cálculo. Sin embargo, en versiones posteriores a 2007 se usan archivos .xlsx ya que permiten un formato más abierto y estructurado, aunque también son capaces de devolver y leer su versión anterior. Los archivos habilitados para almacenar instrucciones para la automatización de los procesos de Excel usan una extensión .xlsm. [21]

En este trabajo se han usado dos hojas de cálculo distintas. Una para crear una tabla que determine los genes de interés junto con un valor numérico que se ha asignado a cada uno de ellos para tener un formato distinto con el que poder buscar estos valores (Genes.xlsx). La otra hoja ha sido creada para determinar los diagnósticos que usa el hospital junto con un valor numérico que identifica a cada uno (Diagnostico.xlsx). Ambos ficheros están disponibles dentro de la subcarpeta Datos que se encuentra en la carpeta INPUT.

Se ha hecho de esta manera por si en algún momento es necesario insertar un nuevo gen o diagnóstico. En este caso se añadiría el nuevo gen o diagnóstico en la fila siguiente al último elemento junto con su valor numérico determinado (siguiendo la numeración de la hoja). Como el código realizado es capaz de leer los Excel, al guardar el cambio y ejecutar el código, ya aparecería el nuevo gen/diagnóstico y será tenido en cuenta desde el momento que se añada.

DRAW.IO PARA DIAGRAMAS??

## Librerías.

### Pandas.

La librería Pandas [27] fue desarrollada por Wes McKinney en 2008. Se usa principalmente para el tratamiento de big data. Debido a la gran cantidad de datos con los que se trabajan en el hospital, se ha decidido que es una buena herramienta para el tratamiento de los datos. Permite leer los datos de los ficheros .xlsx en forma de DataFrame de una forma directa, lo que es de gran interés al tener dos archivos .xlsx con los que se tienen que trabajar.

**Numpy.**

El precursor de NumPy, Numeric, fue creado por Jim Hugunin con contribuciones de varios otros desarrolladores, hasta que en 2005, Travis Oliphant creó NumPy incorporando características de la competencia Numarray en Numeric, con amplias modificaciones. El uso de Numpy [27] para el tratamiento y análisis de los datos es de gran ayuda. Esta librería se encarga del tratamiento, procesamiento y cálculo de los datos que internamente realizan el programa.

**Fitz.**

Es una librería usada para la lectura de archivos PDF en Python.

**Re.**

Este módulo proporciona operaciones de coincidencia de expresiones regulares. [3]

**Os.**

Pproporciona una forma portátil de usar la funcionalidad dependiente del sistema operativo. [1]

# Conclusiones.

- Actualmente no existe una herramienta capaz de satisfacer las necesidades del HUBU, por lo que es un gran reto conseguir crear un código que cumpla con sus expectativas y sea funcional.

- Las entregas realizadas durante la carrera eran funciones que ya existían o se podían obtener mediante la conjugación de distintas funciones, en las que el tutor aportaba comentarios como ayuda. Para la creación de un código desde cero sin referencias en este sector ha sido necesario la determinación de pequeños objetivos para cumplir el objetivo general. La idea se centraba en ir abarcando objetivos fáciles de cumplir con la ayuda de lo aprendido en las asignaturas de la carrera para obtener un código que fuera de interés para el hospital.

- Ha sido complicado el desarrollo del proyecto desde cero. Este requería un gran entendimiento en el servicio de anatomía patológica para ser capaz de obtener el producto final que cumpliera las expectativas del hospital y que estuviera listo para su uso inmediato.

- Los informes proporcionados eran una réplica de los obtenidos del software. Sin embargo, había muchas excepciones que no aparecían, por lo que ha sido todo un reto poder crear el código para que abarcara todas las posibilidades que se puedan llegar a dar.

- Crear hojas de cálculo con la información necesaria, que permita realizar modificaciones por técnicos sin conocimientos informáticos y que estas sean tenidas en cuenta para el resultado final.

- El trabajo de investigación ha sido bastante complejo debido a la gran cantidad de genes que existen, sus fusiones, las excepciones... Sin

embargo, el conocimiento de la profesional de este servicio del hospital ha sido de gran ayuda, aclarando conceptos, explicando los resultados obtenidos del software y los datos de interés y respondiendo a todas las dudas que iban surgiendo en el ámbito de la salud.

## 4.1. Resumen de resultados.

Según la RAE, se define algoritmo como un conjunto ordenado de operaciones sistemáticas que permite hacer un cálculo y hallar la solución de un tipo de problemas. La creación de este código permitirá tener una información más detallada sobre los distintos datos de interés como pueden ser el tipo de cáncer, los genes implicados... en distintas personas para tener una mayor facilidad a la hora de buscar la información, con el fin de poder detectar antes el cáncer y ser capaces de crear tratamientos enfocados a pacientes individuales basándose en sus genes (farmacogenética).

Como se ha comentado anteriormente, el cáncer es una enfermedad bastante complicada de detectar y cuanto antes se detecte, más probabilidades hay de superarla usando el tratamiento más adecuado.

Se han detallado algoritmos o aplicaciones capaces de extraer datos de PDF en distintas empresas, pero ninguno enfocado en la detección de los diagnósticos y genes mutados que los producen ni en el software Oncomie Reporter, por lo que no hay posibles datos con los que contrastar los resultados obtenidos.

Ninguno de ellos capaz de satisfacer las necesidades del Hospital Universitario de Burgos, por lo que la creación desde cero de un proyecto de esta importancia ha sido un gran reto, principalmente porque va a ser una aplicación usada para la mejora de la salud de la población.

El proceso de investigación ha sido un trabajo fácil de llevar a cabo gracias a la ayuda de Patricia, investigadora principal del hospital, ya que la forma de exponer sus conocimientos era muy clara y concisa, haciendo que los datos fueran más fáciles entender y, por lo tanto, consiguiendo una mayor facilidad a la hora de trabajar con ellos.

Los resultados obtenidos en este proyecto han sido distintas tablas con diferente información para mejorar el estudio y la comprensión de los resultados de las pruebas se secuenciación múltiple de última generación en tumores sólidos, así como facilitar la búsqueda de la información, ya que en el resultado final los datos se encuentran estructurados pero en el fichero

inicial, los datos no seguían ningún orden y eran mucho más difíciles de encontrar.

Podemos distinguir dos tablas finales, una con la información de todos los genes y otra más específica solo con la información de los genes patogénicos.

## 4.2. Discusión.

Los resultados obtenidos son una nueva forma de ordenar y entender los datos, en ningún momento se han creado nuevos datos. Lo único que ha cambiado ha sido el formato de los datos, que ha pasado de un formado PDF inicial a un formato Excel final. [14]

La idea de esto es obtener un archivo Excel que almacene todos los resultados de interés para que luego sea más fácil tanto editarlos como trabajar sobre ellos. Al convertir los datos de PDF a Excel se ha conseguido un resultado en estructura tabular mucho más legible y mostrando solo los resultados de interés de una forma mucho más clara sin tener que buscar cada valor en todo el PDF, ya que estos últimos no tienen una estructura tabular definida y es mucho más difícil encontrar los datos necesarios.

Además, los archivos PDF son bastante más difíciles de manipular, editar o escribir sobre ellos, son principalmente usados para la visualización. De esta forma se garantiza que cualquier cambio que se desee realizar pueda hacerse con la mayor brevedad y facilidad posible. Los PDF suelen mantener el formato original, en Excel es posible modificar los valores de las celdas, letras, bordes, estilo de la hoja...

El principal inconveniente es que los PDF son ampliamente compatibles y pueden ser abiertos en casi cualquier dispositivo sin necesidad de tener descargado algún programa específico. Esto no ocurre con Excel, ya que es necesario tener Microsoft Excel o un programa compatible para poder abrir este tipo de ficheros. Nos hemos asegurado de que el ordenador principal con el que se va a trabajar es capaz de abrir ficheros Excel sin ningún inconveniente.

## 4.3. Aspectos relevantes.

### Elección del lenguaje.

Inicialmente se pretendía usar tanto R como Python. A la hora de la búsqueda de información sobre PDFscraping se tuvieron en cuenta los

posibles paquetes de ambos lenguajes, su funcionamiento, las instalaciones necesarias... Una vez se tenía en código en Python, se comenzó con R, pero debido a una falta de tiempo se decidió que era más preciso centrarse en modificar el acabado que seguir con el otro desde cero.

El uso de Python como lenguaje de programación principal se debe a que tiene una gran cantidad de bibliotecas y herramientas diseñadas para el análisis de datos (al igual que R). Además, al ser el lenguaje de programación usado por excelencia en la carrera, se tienen más conocimientos acerca de este.

Se ha usado el libro *Python for Data Analisis* para la ayuda del código de Python [27]

## PDFscraping.

Se comenzó con la búsqueda de información sobre PDFscraping y las distintas bibliotecas que podían usarse en Python. Entre ellas destacaban:

- **Tabula-py**: biblioteca que permite extraer información de un formato estructurado introduciendo la ubicación de los datos tabulares dentro del PDF, especificando las coordenadas siguiendo el formato (arriba, izquierda, abajo, derecha). Esto se obtendrá principalmente por prueba y error. Si el fichero solo contiene la tabla en la que se quiere buscar la información, no es necesario especificar el área porque las filas y las columnas deberían detectarse automáticamente. [40] También se puede extraer información de formatos no estructurados, pero se debería de seguir una serie determinada de pasos mucho más complejos.

  Los datos de los informes con los que trabajamos están desordenados, por lo que esta biblioteca no sería capaz de trabajar correctamente. El método de prueba-error sería buena elección en caso de que todos los ficheros siguieran una misma estructura, no cuando se quiere trabajar con grandes conjuntos de datos donde los informes ni siguen la misma estructura ni los datos se encuentran en la misma posición siempre. Además, esto conllevaría una gran cantidad de tiempo para encontrar las posicione de todos los valores que se piden.

- **PyPDF2**: esta biblioteca puede ser usada de distintas maneras. Puede usarse como herramienta de línea de comandos para crear o modificar archivos de formato PDF, también se puede ejecutar la biblioteca dentro de los scripts de Python importándolo como un módulo y

llamando a sus funciones (esto es especialmente útil si se quiere automatizar tareas), se puede leer, analizar y escribir archivos PDF y permite trabajar con cadenas Unicode para poder manejar distintos caracteres. [32] Se suele utilizar cuando se trabaja con formatos PDF no estructurados.

Fue una de las últimas en descartar, ya que nos permitía trabajar con datos no estructurados. Pero se vio que para este proyecto era mejor el uso de fitz, ya que se adaptaba mejor a las necesidades que se pretendían solventar.

- **PDFQuery**: para poder trabajar con esta biblioteca es necesario convertir los ficheros PDF en formatos XML, ya que este formato permite definir un conjunto de reglas para codificar PDF en un formato que es tanto legible por humanos como por máquinas. [41] Para obtener los valores hay que buscar las coordenadas de la palabra de interés (izquierda, abajo, derecha, arriba) dentro del cuadro de texto, siendo el eje X el ancho de la página y el eje Y el alto. Cada elemento tiene sus límites definidos por un cuadro delimitador formado por cuatro coordenadas. Otro método para obtener información es usando palabras clave vecinas, es decir, buscar las palabras clave que nos interesan para extraer los datos asociados a esa palabra. Una vez tenemos detectada la palabra clave, podemos usar el método keywords.get para extraer las coordenadas de la palabra clave y desplazarlas sumando o restando valores para obtener las verdaderas coordenadas de la palabra de interés.

  Esta biblioteca tampoco se ha usado debido a que los informes exportados por el software están en formato PDF y no nos interesa pasarlo a XML habiendo bibliotecas que nos permitan no modificar su extensión. Se tardaría más tiempo en encontrar las coordenadas que en realizar el código, ya que el texto de nuestro PDF no contiene tablas donde se encuentre toda la información que necesitamos, sino que los datos de interés se encuentran como texto a lo largo del PDF.

- **PDFMiner**: es una herramienta de extracción de información de ficheros PDF. Permite obtener una ubicación real y exacta de texto de una página. Viene con dos herramientas de interés como son pdf2txt.py que permite extraer el contenido del PDF representando el texto en ASCII o Unicode y dumppdf.py que vuelva el contenido del PDF en un formato xml. [5]

  Viendo algunos ejemplos se concluyó que es una biblioteca bastante compleja de usar y requiere un conocimiento profundo de la estructura

interna de Portable Document Format. No tiene actualizaciones recientes, por lo que puede haberse quedado algo obsoleto frente a otras bibliotecas y cuenta con una menor documentación.

- **PyMuPDF o Fitz**: permite trabajar con datos no estructurados y contiene una gran cantidad de funciones avanzadas para el procesamiento de PDF. Puede combinarse con otras bibliotecas y tiene una gran documentación de interés. Debido a su alto rendimiento y sólido soporte, se ha determinado como la mejor opción para este proyecto.

## Tablas

Una vez buscada esta información, nos centramos en el entendimiento tanto de los datos que tenemos como del resultado final ideal. Como el objetivo es obtener una tabla que contenga toda la información de una manera ordenada y estructurada, se plantean las posibles tablas específicas. Hay que poder responder a preguntas básicas como ¿cuántas tablas necesito?, ¿cómo las voy a unir?, ¿tengo toda la información para rellenar las tablas?, ¿necesito información adicional?, ¿para qué es cada tabla?

### Formación de tablas.

Se ha optado por crear cinco tablas distintas y unirlas para obtener las finales. La primera tabla es la más importante y contiene la información específica de los pacientes. Permite identificar al paciente y relacionarlo con la muestra que le pertenece. La segunda tabla se basa en los diagnósticos específicos de cada persona. La tercera tabla hace referencia a las mutaciones que presenta cada paciente y los datos que permiten complementar esta información. La cuarta se centra en las mutaciones patogénicas. Se ha determinado una tabla específica para estas mutaciones porque son las que predisponen al paciente a tener ciertas enfermedades o trastornos. La quinta tabla es información acerca de los ensayos clínicos que existen y los tratamientos disponibles para cada diagnóstico.

Estas tablas se unen mediante una única clave primaria compuesta, formada por dos atributos. Para elegir estos atributos, nos aseguramos de que sean valores específicos y que no se repitan para evitar la formación de filas/columnas espurias. Como el NHC es un número único e intransferible para cada persona, se ha determinado como primer atributo. En el caso de que una persona pudiera realizarse más pruebas, se ha usado el número de chip como segundo atributo. Esto nos asegura que aunque un paciente se realice dos pruebas en dos días seguidos, no se crearán filas/espurias, ya que

como en cada chip hay tan solo ocho pacientes, es imposible que sus dos muestras se estudien en el mismo chip.

**Unión de tablas.**

Se compararon varios métodos para ver con cuál se obtenía el mejor resultado.

- **Concat** concatena tablas o DataFrames. Con este método pueden aparecer columnas espurias si estas tienen nombres duplicados o si no se usa correctamente este método. La unión se realizaba sin tener en cuenta las coincidencias de los valores, por lo que este método no unía realmente los DataFrames, sino que posponía uno tras otro.
- **Merge** es una biblioteca de Pandas bastante rápida que realiza la unión basándose en los valores de las columnas. Cuando aparecen valores duplicados o vacíos o si los nombres de las columnas no coinciden en ambas tablas, aparecen resultados espurios.
- **Join** también pertenece a Pandas. Proporciona opciones flexibles a la hora de realizar las uniones (join, left, rigth o outer). No se crearán columnas espurias al unir los DataFrames porque une usando índices en lugar de columnas. Existen métodos como **lsuffix** o **rsuffix** que permiten agregar sufijos a los nombres de las columnas para evitar resultados irrelevantes.

La mejor opción en este caso es join por no crear resultados espurios tras las uniones, facilitando el proceso de la creación de DataFrames finales.

**Tablas resultantes.**

La combinación de las cinco tablas iniciales nos ha permitido obtener dos hojas de cálculo donde se almacena la información completa y otras donde se almacenarán solo 80 líneas de resultado.

Lo más lógico en este caso sería usar una base de datos porque permitiría almacenar los datos de una manera ordenada manteniendo la integridad y consistencia de los datos. Pero como el usuario necesitaba una forma sencilla de almacenar y modificar posteriormente los resultados, se decidió usar Excel.

Antes de la selección final, se estudióla diferencia entre dos ficheros distintos, CSV y Excel. [34] Se ha determinado que la mejor forma de

exportación era Excel, ya que nos devolvía los resultados de una forma mucho más ordenada que CSV por filas y columnas sin necesidad de usar separadores. Además, Excel proporciona una gran cantidad de fórmulas y funciones para trabajar con los datos y facilita la formación de gráficos, ofreciendo una gran ventaja al hospital, por si en un futuro se quiere modificar o trabajar sobre los datos que aparecen en este fichero resultante.

## Código.

Se han creado distintas funciones propias:

La función **LeerDocumento** recibe el nombre de un fichero y devuelve una lista de lineas que componen cada uno de los PDF.

**BuscarValor**, donde al buscar una palabra, devolvía el resultado que había tras encontrarla. Como no funcionaba en todos los casos (por ejemplo en tratamiento disponible o ensayo clínico, el número estaba delante) se han tenido que buscar distintos métodos para ser capaces de obtener todos los valores. En algunos casos se han usado expresiones regulares que servían de patrones para buscar dicho patrón en el text, uso de posiciones...

Las expresiones regulares son un pequeño apartado de gran interés dentro del lenguaje de programación que especifica un conjunto de caracteres posibles que se desean encontrar en el texto sobre el que se está trabajando. No todas las búsquedas se pueden realizar siguiendo este método, en algunos casos es mejor realizar un código de búsqueda que usar dichas expresiones, aunque sea un proceso más laborioso, puede llegar a ser más comprensible.

Hay patrones simples muy básicos como una sola letra, donde el código encontrará todas las letras determinadas. Sin embargo, hay un conjunto de caracteres conocidos como metacarcteres que tienen un significado especial y no pueden ser buscados a lo largo del texto. Un ejemplo de ellos son los corchetes [], que permiten identificar una clase de carácter o un guión (-) que nos permitirá indicar el rango de valores entre los dos caracteres. Por ejemplo, [abc] nos indica que quiere buscar cualquiera de las tres letras que aparece ente los corchetes. El mismo resultado pero usando un guión sería [a-c], que busca tanto las letras indicadas como los posibles valores que podría tomar -. Al igual que estos ejemplos, hay muchas más excepciones.

También hay expresiones mucho más complejas donde se repiten valores de búsqueda o se usan valores con referencias propias. Esto quiere decir que existen valores como D que coincide con cualquier valor que no sea numérico o w que coincide con cualquier carácter alfanumérico. [16]

El resultado final ha sido la obtención de una variable por cada campo de interés, para poder crear posteriormente las tablas.

Una vez obtenidas todas las variables, se debían unir las tablas. Pero el principal problema era cómo realizar la unión. Para ello se investigó un poco a cerca de todas las posibles opciones [33] [2]

# Aspectos relevantes del desarrollo del proyecto

# Lineas de trabajo futuras

Este proyecto tiene un gran potencial para ser usado en el campo de farmacogenética. Se espera que en un futuro, pueda ser mejorada para ser capaz de adaptarse a las necesidades de los distintos hospitales y así, permitir el análisis e interpretación de un mayor número de datos. También se espera que este proyecto, en un futuro, sea capaz de formar parte de empresas dedicadas a la farmacogenética, para ser capaces de analizar como reaccionarían distintos genes a fármacos determinados para conseguir que en un futuro, el cáncer sea una enfermedad más fácil de tratar y que no se encuentre ente las primeras causas de muerte mundiales

Partiendo del punto en el que se ha dejado y enfocado a su uso como herramienta futura en el área de salud, se propone configurarlo para que sea capaz de ejecutarse de forma automática en intervalos regulares de tiempo determinados por el investigador principal y una vez finalizada la ejecución, enviar una notificación, tanto en el ordenador desde el que se ejecuta como en un dispositivo electrónico de interés, para indicar que los resultados están listos.

También se deja pendiente el crear una base de datos en SQL para que almacenar toda la información obtenida y realizar posteriores estudios partiendo de ella.

Sería interesante si se lograra implementar dicho código (con mejoras en el caso de que se viera necesario) en otros lenguajes de programación como R, por ser uno de los lenguajes de programación enfocado especialmente en análisis de datos. Una vez obtenido este código, se pueda implementar una aplicación alojada en la web como servicio usando por ejemplo el framework Shiny. Shiny es una herramienta de desarrollo web de código abierto creada por RStudio que permite construir aplicaciones interactivas basadas en R y

desplegarlas en la web, por lo que para llegar a este punto se debería tener el código en R, aunque también da la opción de usar Python.

# Bibliografía


[1] os — miscellaneous operating system interfaces.

[2] pandas.

[3] re — regular expression operations.

[4] Latex general help, 2016.

[5] Pdfminer, 2016.

[6] Anaconda, 2023.

[7] Anaconda, 2023.

[8] Britannica, 2023.

[9] ¿qué es excel y para qué sirve?, 2023. blog.

[10] Code Academy. What is an ide?, 2023.

[11] ADEA. La automatización de procesos en el sector salud, 2022.

[12] ALTASSIAN. Git branch.

[13] Apify. Build reliable web scrapers. fast. Uso de la herramienta.

[14] APPSTATE. Pros and cons of pdfs., 2021.

[15] ASTERA. Astera: Reporminer, 2023.

[16] La biblioteca estándar de Python. re - operaciones con expresiones regulares, 2023.

[17] Cancer.net. Genes and cancer. Definición.

[18] Clinithink. The key to saving human lives is understanding human words. Uso de la herramienta.

[19] CTakes. Apache ctakes. Uso de la herramienta.

[20] G2. Reporminer, 2023. Página de reseñas.

[21] Alexander S. Gillis.

[22] Red Hat. ¿qué es el open source?, 2023.

[23] National Cancer Institute. Proto-oncogenes. Definición.

[24] National Cancer Institute. Tumor suppressor gene. Definición.

[25] National Cancer Institute. Understanding cancer prognosis.

[26] National Cancer Institute. What is cancer?

[27] Wes McKinney. *Python for Data Analisis*. O'RIELLY, 2022.

[28] Nanonets. Automate manual data entry using ai. Uso de la herramienta.

[29] OMS. Cáncer, 2022.

[30] Sonia Ordoñez. ¿qué es anatomía patológica? Página usada en la memoria de Anatomía Patológica 2021-2022 durante prácticas.

[31] Dimitris Poulopoulos. Jupyter is now a full-fledged ide: Annual review, 2022.

[32] Dhana Shree. Pypdf2 library: How can you work with pdf files in python?, 2022.

[33] Kyle Stratis. Combining data in pandas with merge(), .join(), and concat().

[34] Toggl track. Difference between csv and xls, 2023.

[35] Wikipedia. Github, 2023.

[36] Wikipedia. Latex, 2023.

[37] XATAKA. Git –distributed-is-the-new-centralized, 2019.

[38] XATAKA. Qué es github y qué es lo que le ofrece a los desarrolladores, 2019.

[39] ZAKHAR YUNG. 10 best data extraction tools in 2023 for your business, 2023.

[40] Aaron Zhu. How to scrape and extract data from pdfs using python and tabula-py, 2021.

[41] Aaron Zhu. How to scrape and extract data from pdfs using python and pdfquery, 2022.