

Escuela Superior de Ingeniería y Tecnología

Grado en Ingeniería Informática

Trabajo de Fin de Grado

Extracción y procesamiento de datos a partir de historiales clínicos de pacientes.

Data extraction and processing from patient records.

Omar Mendo Mesa

La Laguna, 27 de julio de 2019

D. **Carlos J. Pérez González**, con N.I.F. 45.452.719-G profesor Titular de Universidad adscrito al Departamento de Matemáticas, Estadística e investigación Operativa de la Universidad de La Laguna, como tutor,

C E R T I F I C A

Que la presente memoria titulada:

"Extracción y procesamiento de datos a partir de historiales clínicos de pacientes"

ha sido realizada bajo su dirección por D. **Omar Mendo Mesa**, con N.I.F. 42.238.212-T.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 27 de julio de 2019.

Agradecimientos

A mi tutor Carlos, por brindarme la oportunidad de realizar este trabajo, siendo un referente en el tema. Agradecerle la gran implicación y el gran apoyo que me ha dado durante la realización de este proyecto.

A mi madre, por ser un apoyo incondicional y darme la fuerza que siempre he necesitado para no tirar la toalla en ningún momento.

A Sara Machado, por todo el apoyo que me ha dado durante el Grado de Ingeniería Informática y el desarrollo de este trabajo.

A mi familia y mis compañeros del grado, en concreto, a los que integrábamos la Delegación de Alumnos hace unos años, por pasar grandes momentos juntos.

Licencia



© Esta obra está bajo una licencia de Creative Commons Reconocimiento-CompartirIgual 4.0 Internacional.

Resumen

Con el paso de los años, el aumento exponencial de los datos no estructurados ha hecho que su administración y la extracción de información sea un gran desafío, ya que los datos deben convertirse en formatos legibles para una máquina, con el fin de que esta pueda procesar los datos recogidos, analizarlos, y dar una solución esperada.

En colaboración con el Centro Neurológico Antonio Alayón, el TFG se centrará en el diseño y desarrollo de una herramienta de extracción, tratamiento y visualización de datos a partir de historiales clínicos de los pacientes del centro (en formato ODT/DOC), así como estructurar los datos en una base de datos para su posterior visualización.

Principalmente, el trabajo se centra en la extracción y procesamiento de los datos recogidos de los historiales clínicos e introducirlos de forma estructurada en una base de datos, pero, además del cometido principal, se ha desarrollado una aplicación web de gestión de datos para añadir, editar o borrar nuevos historiales clínicos, además, de poder visualizar de forma gráfica los datos recogidos del proceso de extracción, sirviendo así como una herramienta útil para el personal del centro.

Las principales tecnologías utilizadas en el proyecto van desde el IDE (PyCharm 2019.2), lenguajes de programación, principalmente, Python y JavaScript, así como de frameworks de programación, tanto de backend (Flask) como frontend (BootStrap, DataTables, Plotly)

Palabras clave

Python, PyCharm, Flask, Extracción, Datos clínicos, JavaScript, Framework

Abstract

Over the years, the exponential increase in unstructured data has supposed that its administration and information extraction represent a great challenge, since the data must be converted into readable formats for a machine, so that it can process the collected data, analyze it, and give an expected solution.

In collaboration with Centro Neurológico Antonio Alayón, TFG will focus on the design and development of a data extraction, treatment and visualization tool from the center's patients records (ODT / DOC format), as well as structuring the data in a database for later visualization.

The project was focused on the extraction and processing of the data collected from the clinical records and stored in a structured way into a database, but, in addition to the main proposal, a web application for data management has been developed to add, edit or delete new clinical records. In addition, this application allows the users to visualize the data obtained from the extraction process, serving as a useful tool for the center's staff.

The main technologies used in the project are the IDE (PyCharm 2019.2), programming languages, mainly Python and JavaScript, as well as programming frameworks, both backend (Flask) and frontend (BootStrap, DataTables, Plotly)

Keywords

Python, PyCharm, Flask, Extraction, Clinical data, JavaScript, Framework

Índice general

Capítulo 1. Introducción	1
1.1 ¿Qué es la extracción de datos?	1
1.2 Técnicas de extracción de datos	2
1.3 Fuentes de datos: historiales clínicos	2
1.4 Objetivos	3
Capítulo 2. Antecedentes y estado actual del tema	4
2.1 Evolución de la extracción y tratamiento de datos	4
2.1.1 Problemas actuales	5
2.1.2 Soluciones actuales	5
2.2 Documento con historial clínico: examen CAMDEX	6
Capítulo 3. Desarrollo de la herramienta de extracción y procesamiento	8
3.1 Elección de tecnologías	8
3.1.1 Python	8
3.1.2 Poetry	9
3.1.3 Flask	10
3.2 Estructura del proyecto	10
3.3 Extracción de datos de historiales clínicos	13
3.3.1 Metodología y procesos	14
3.4 Aplicación web: Extract-py	18
3.4.1 Tecnologías utilizadas	18
3.4.2 Principales funciones	19
Capítulo 4. Extract-py y algunos ejemplos	20
4.1 Historial clínico del Paciente 1 (extracción de datos)	20
4.1.1 Proceso de extracción de datos clínicos	20
4.1.2 Visualización de la información	22
4.1.3 Gráficas	23
4.1.4 Manejo del historial del paciente	24
4.2 Historial clínico del Paciente 2 (aplicación)	25
4.2.1 Creación del historial del paciente	25
4.2.2 Visualización de la información	26

Capítulo 5. Conclusiones y líneas futuras	27
Capítulo 6. Summary and Conclusions	28
Capítulo 7. Presupuesto	29
Capítulo 8. Bibliografía	30

Índice de figuras

2.1.	Ejemplo de pruebas del examen CAMDEX-R	6
3.1.	Estructura de directorios del proyecto (I)	10
3.2.	Estructura de directorios del proyecto (II)	11
3.3.	Esquema de la base de datos - tablas y columnas	12
3.4.	Disposición datos personales del paciente	13
3.5.	Disposición datos examen CAMDEX del paciente	13
3.6.	Bash script simple para la conversión de formato	14
3.7.	Método de conversión y eliminación líneas vacías	15
3.8.	Método de extracción de datos personales del paciente	16
3.9.	Método de extracción examen CAMDEX paciente	17
4.1.	Terminal con la ayuda del comando <code>'extract'</code>	21
4.2.	Terminal con ejecución del comando <code>'extract convert'</code>	21
4.3.	Terminal con la extracción de datos personales con el comando <code>'extract patient'</code>	21
4.4.	Terminal con la extracción de los datos del examen CAMDEX con el comando <code>'extract camdex'</code>	22
4.5.	Página principal con los datos del Paciente 1	22
4.6.	Página principal del apartado <code>'Gráficas'</code>	23
4.7.	Visualización de gráficas de pruebas MMSE y MEC	24
4.8.	Manipulación de historiales clínicos	25
4.9.	Página principal con los datos del Paciente 2	26

Índice de tablas

7.1 Resumen de horas/precio por trabajo

29

Capítulo 1. Introducción

El objetivo del presente Trabajo de Fin de Grado es el estudio de la extracción y el procesamiento de datos a partir de historiales clínicos de pacientes, en colaboración con el Centro Neurológico Antonio Alayón. Conjuntamente, se diseñará una aplicación web con el fin de visualizar los datos de forma estructurada en tablas y gráficas, además de funcionar como gestor de historiales clínicos, pudiendo el cuerpo médico poder crear nuevos historiales, modificarlos o borrarlos.

Cabe decir, que todos los datos sensibles, como nombres, direcciones, números de teléfono, fechas de nacimiento, resultados de pruebas médicas entre otros, son datos ficticios, con el fin de mantener en el anonimato a los pacientes del centro, respetando la Ley de Protección de Datos Médicos, también conocida como Ley de Autonomía del Paciente[1].

1.1 ¿Qué es la extracción de datos?

Entendemos por *extracción*, hablando en el ámbito de la informática [2], concretamente, de datos, a un mecanismo de recuperación de información, con el objetivo de extraer de forma automatizada datos estructurados a partir de ficheros legibles por un ordenador.

En la actualidad, el aumento exponencial de la generación masiva de datos en nuestra sociedad a través de las nuevas tecnologías, ha abierto una puerta a las organizaciones a conocer más en detalle a los individuos a través de la información (compras, hábitos, pruebas médicas, búsquedas en Internet, etc), pudiendo así tomar decisiones de forma correcta en función de la interpretación de los datos recogidos.

Por ejemplo, un centro médico realiza unas determinadas pruebas de carácter neurológico a un grupo de pacientes, los cuales presentan un declive en su salud al suministrarle un determinado fármaco a una parte de los pacientes. Extrayendo los datos recogidos en las pruebas de estos pacientes y analizando los resultados, los especialistas pueden visualizar en qué momento empieza el declive en función del suministro del fármaco, pudiendo tomar una decisión correcta en el diagnóstico y cambiar el tratamiento a uno que realmente funcione en el paciente.

1.2 Técnicas de extracción de datos

Existen múltiples soluciones a la hora de realizar una extracción de datos, ya sea de un fichero, una página web, una base de datos, etc. Una de las técnicas más extendidas es el *Web Scraping* (raspado web) [3], usada para extraer información de sitios web de forma automatizada. Algunas de las técnicas más significativas de Web Scraping son las siguientes:

- Uso de *expresiones regulares*: En algunas situaciones, su uso es de gran ayuda para encontrar patrones de datos que se quieren extraer.
- Algoritmos de *minería de datos*
- Aplicaciones para web scraping

En nuestro caso, haremos uso de métodos de extracción propios para la extracción de información de ficheros locales, en este caso, *historiales clínicos* de pacientes.

1.3 Fuentes de datos: historiales clínicos

Un *historial clínico* [4] se basa principalmente en el archivo que incluye los datos clínicos que tengan relación con la situación actual de un paciente, además de datos de antecedentes personales y familiares que pueden estar relacionados en enfermedades hereditarias.

En este caso, contaremos como fuente de datos para nuestro proyecto con los historiales clínicos del Centro Neurológico Antonio Alayón pero, para preservar la intimidad de los pacientes, se han eliminado sus datos personales, creando otros totalmente ficticios para su uso en el desarrollo del proceso de extracción de datos en nuestro proyecto.

1.4 Objetivos

A partir de diversos modelos de historial clínico se llevará a cabo el estudio de diferentes propuestas para la extracción de la información y la configuración de una base de datos. Para ello, se analizarán los diferentes desarrollos que puedan existir y se encuentren publicados en artículos y repositorios de libre acceso. Se realizará un análisis comparativo de las diferentes soluciones y se implementará un desarrollo adaptado a las características específicas del modelo de historial clínico. Asimismo, una vez configurada la base de datos se podrán realizar varios análisis de carácter estadístico (tablas, gráficos, etc..) utilizando la mismas herramientas para el desarrollo del proyecto. También se ha propuesto desarrollar una aplicación web con el objetivo de visualizar los datos extraídos previamente de los historiales clínicos de los pacientes del centro, así como alojar las distintas gráficas resultado del análisis estadístico de los datos recogidos.

Desde la propia aplicación se podrán gestionar los historiales clínicos de los paciente, así como tener la posibilidad de crear nuevos historiales, modificarlo o borrarlos a demanda, teniendo así un mayor control de los historiales.

Capítulo 2. Antecedentes y estado actual del tema

A menudo las organizaciones y empresas disponen de una gran cantidad de información, textual y no estructurada, la cual se va generando y almacenando de forma constante. Por esta razón, existe una demanda de soluciones que permitan un mayor aprovechamiento de la información contenida para la generación de conocimiento y la toma de decisiones. La minería de datos y, en particular, la de textos constituye un área de conocimiento dentro de la cual se estudia este tipo de problemas y donde se generan estas soluciones para la extracción de conocimiento de una colección de textos, a partir de la identificación y exploración de patrones interesantes. Para ello, resulta importante establecer una metodología para la estructuración y almacenamiento de los contenidos extraídos en una etapa de pre-procesamiento. De este modo, será posible aplicar a los datos extraídos las técnicas de análisis estadístico o de visualización de la información.

2.1 Evolución de la información

Muchos años han pasado desde que la información de las organizaciones, ya sean pequeñas o grandes, se almacenaba en simples hojas de cálculo, completadas de forma manual, para pasar a los inmensos volúmenes de datos con los que nos encontramos en la sociedad, aquellos que nos desvelan sus costumbres, hábitos, relaciones o maneras de actuar.

En plena era digital, existen organizaciones con cantidades ingentes de datos que son manejados cada segundo por lo que, con el exponencial crecimiento de este volumen de datos, también se han desarrollado problemas para gestionar, almacenar o mantener seguros dichos datos.

2.1.1 Problemas actuales

Los principales problemas a los que se enfrentan hoy en día las organizaciones respecto a la gestión, se basan, principalmente en su tratamiento y almacenamiento, ya que es imposible tenerlos gestionados en simples hojas de cálculo, por lo que con el tiempo, se han desarrollado dificultades tales como:

- Insuficiencia de capacidad para almacenar grandes volúmenes de datos.
- Incapacidad para gestionar de forma estructurada dichos datos.
- Falta de claridad para la toma de decisiones en función de datos.

Por lo que, con el tiempo, ha surgido la necesidad de que expertos se especialicen en el desarrollo de soluciones para dichos problemas, naciendo así la figura del científico de datos (Data Scientist) [5]

2.1.2 Soluciones actuales

En la actualidad los problemas anteriormente descritos han sido subsanados en parte gracias a los procesos de extracción y análisis de datos, concretamente, con la minería de datos (Data Mining) [6]. Hace referencia al campo de la estadística y la informática que tiene como principal cometido el descubrimiento de patrones en grande volúmenes de conjuntos de datos, haciendo uso de métodos de inteligencia artificial, aprendizaje automático, estadística y sistemas de bases de datos.

Como principal objetivo, la minería de datos tiene la extracción de información de conjuntos voluminosos de datos y transformarlos en estructuras definidas, también denominados datos estructurados.

Gracias a la minería de datos y, en concreto, la de textos, se logra dar una solución a los problemas que evolucionaron paralelamente al desarrollo de grandes volúmenes de datos. Con esto, pueden ser alojados en bases de datos, estructurando dichos datos, así como realizar estudios estadísticos y análisis para su mayor comprensión.

2.2 Documento con historial clínico: examen CAMDEX-R

El examen CAMDEX-R [7] (Prueba de Exploración Cambridge Revisada para la Valoración de los Trastornos Mentales en la Vejez), tiene como objetivo el diagnóstico diferencial de la demencia y de otros trastornos mentales y físicos de las personas de avanzada edad, que se divide en diversas pruebas:

- Una entrevista clínica estructurada: Tiene como objetivo recopilar información sobre el estado actual del paciente y su historia médica. Un familiar o cuidador del paciente debe estar en la entrevista.
- Otra entrevista clínica estructurada con el paciente: Recoge información sobre el estado de salud mental actual del paciente y síntomas de demencia que presenta.
- Una breve batería neuropsicológica (CAMCOG-R): Incluye un amplio número de tests objetivos para la valoración de la función cognitiva.

Este examen será de gran importancia en el desarrollo del proyecto, ya que es una de las fuentes de datos que se tienen para su extracción y análisis posterior de los documentos que contienen los historiales clínicos.

Exploración Neuropsicológica

CAMDEX (Examen Cambridge para los trastornos mentales en la vejez)	PD
MMSE	25/30
MEC	29/35
Test de Atención-Memoria-Concentración de Roth y Hopkins	5/10
Camcog Total	78/107
Orientación	10/10
Lenguaje total	26/30
• Lenguaje comprensivo	9/9
• Lenguaje productivo	17/21
Memoria total	13/27
• Memoria reciente	4/4
• Memoria remota	6/6
• Memoria de aprendizaje	3/17
Atención y concentración	4/7
Praxis (ideomotora, ideacional, constructiva)	11/12
Cálculo	2/2
Pensamiento abstracto	4/8
Percepción táctil y visual	7/11

Figura 2.1. Ejemplo de pruebas del examen CAMDEX-R

El conjunto de pruebas más significativo se encuentra en el CAMCOG-R (Evaluación Cognitiva de Cambridge Revisada) [8], una batería de tests sensible a la detección de grados leves de demencia y especialmente útil para evaluar a los pacientes con disfunciones del lenguaje o de las capacidades visuoespaciales.

Las pruebas más significativas del CAMCOG-R son las siguientes:

- **MMSE** (Mini Mental State de Folstein) [9]: Se trata de un test de cribado de demencias, basado en preguntas simples para comprobar la orientación témporo-espacial del paciente, entre otras.
- **Orientación:** Prueba diseñada para conocer la capacidad del paciente para orientarse en el tiempo y espacio, con preguntas como "*¿Qué hora es?*" o "*¿Dónde estamos ahora?*".
- **Lenguaje:** Conjunto de pruebas para conocer la capacidad de comunicarse del paciente.
- **Memoria:** Conjunto de pruebas para conocer la capacidad del paciente para retener información y poder transmitirla.
- **Praxis:** Conjunto de pruebas relacionadas con las capacidades del paciente para copiar y dibujar figuras, así como escribir dictados.

Capítulo 3. Desarrollo de la herramienta de extracción y procesamiento

Debido a las necesidades del centro neurológico por tener un sistema de consulta de historiales mucho más eficiente que el que actualmente tienen, en este proyecto se ha desarrollado una herramienta que permite la extracción de los datos personales del paciente en cuestión y sus registros del examen CAMDEX-R a lo largo del tiempo de forma acumulativa. Posteriormente, parsear dichos datos adecuadamente para ser introducidos en una base de datos de forma estructurada y visualizar dichos datos en una aplicación web. Finalmente, se podrá consultar el conjunto de historiales clínicos de pacientes del centro.

3.1 Elección de tecnologías

La elección de las tecnologías adecuadas para el correcto desarrollo del proyecto ha sido una tarea sencilla, ya que se ha visto facilitada gracias a la experiencia adquirida en las Prácticas Externas durante el segundo cuatrimestre en el Centro Meteorológico de Tenerife - AEMET [10]. Muchas de las tecnologías que usé para realizar procesos de extracción y visualización de datos de estaciones meteorológicas, van de la mano con las tareas que había que desarrollar en el Trabajo de Fin de Grado.

Todas las tecnologías utilizadas en el proyecto son actuales, de software libre y disponibles en su última versión, permitiéndonos así manejar todas las características posibles para su uso.

3.1.1 Python

Python [11] es un lenguaje de programación interpretado de software libre, diseñado por Guido van Rossum y que vio la luz por primera vez en 1991.

Si hablamos de lenguajes de programación para el desarrollo de proyectos referentes al tratamiento de datos, como principales exponentes tenemos a R [12] y a Python.

Para este proyecto en concreto, se ha decidido utilizar Python, principalmente por las siguientes ventajas:

- Mayor robustez a la hora de automatizar procesos.
- En cuanto al desarrollo de la aplicación web, se ha desarrollado con *Flask*, framework de desarrollo web para Python del que hablaremos más adelante.

Además de estas ventajas, el conocimiento previo de Python iba a facilitar aún más los objetivos a cumplir para el correcto desarrollo del proyecto.

3.1.2 Poetry

Poetry [13] es un gestor de dependencias para Python, que permite instalar o actualizar, según proceda, las diferentes librerías de las que depende el proyecto, manteniendo un control sobre qué librerías se están usando, en qué versión y cómo.

Para crear un nuevo proyecto de Python con Poetry, se ejecuta el siguiente comando:

- `$ poetry new project-name`

Dando como resultado, la siguiente estructura de proyecto:

```
project-name
|--- pyproject.toml
|--- README.md
|--- project-name
    |--- __init__.py
```

En el fichero `'pyproject.toml'` es donde Poetry guardará las dependencias del proyecto, así como el nombre del mismo, su versión, descripción y el nombre del autor.

3.1.1 Flask

Flask [14] es un framework orientado para el desarrollo web para Python, que permite diseñar aplicaciones web con el menor número de líneas de código posible de la manera más eficiente. Una de las ventajas que tiene Flask, es que incluye un servidor de desarrollo, no necesitando así una infraestructura con un servidor web para probar o visualizar nuestras aplicaciones.

Otra gran ventaja de Flask es su sencillez para el manejo de rutas, sabiendo en cada momento las peticiones que realiza el cliente y hacia dónde se dirige en la aplicación.

Además, para la conexión de la base de datos con la extracción de los datos mediante Python, existe un puente de enlace que funciona a la perfección con Flask, llamado *SQLAlchemy* [15], un ORM (Object-Relational Mapping), con soporte con SQLite para nuestra base de datos.

3.2 Estructura del proyecto

Siguiendo la estructura que propone Poetry al crear el proyecto, la estructura de directorios del proyecto es la siguiente, la cual desglosamos a continuación, describiendo los ficheros más significativos del proyecto.

Para visualizar el código y contenido de cada fichero, el proyecto está alojado en un repositorio público en **GitHub** [16].

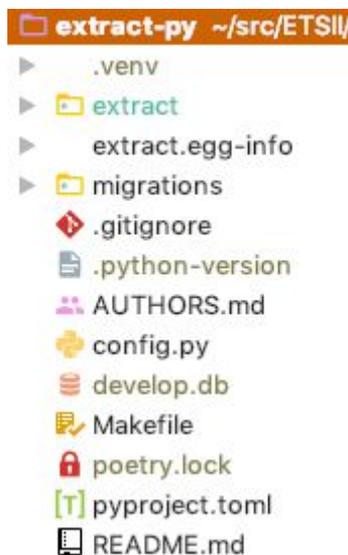


Figura 3.1. Estructura de directorios del proyecto (I)

Además, dentro de la carpeta `'extract'`, se alojan todos los código Python del proyecto, tanto los que alojan los métodos de extracción de datos de los historiales clínicos, como archivos de configuración de la base de datos y creación de la aplicación web con Flask, así como las plantillas HTML, estilos y ficheros JavaScript correspondientes a la aplicación web.



Figura 3.2. Estructura de directorios del proyecto (II)

Los principales ficheros a tener en cuenta del proyecto son los siguientes:

- `Makefile`: Archivo correspondiente a la herramienta **make**, ideal para la ejecución de determinadas tareas.
- `pyproject.toml`: Fichero que aloja las dependencias de librerías del proyecto, así como su descripción.
- `app.py`: Fichero de configuración y creación de la aplicación web de Flask.
- `chart.py`: Fichero que aloja los métodos necesarios para la visualización de las gráficas en la aplicación
- `cli.py`: Fichero que aloja los métodos de conversión de formato de ficheros, métodos de extracción de datos y desarrollo de una interfaz de línea de comandos para estas tareas.
- `model.py`: Fichero que aloja el esquema y relaciones de las tablas de la base de datos.
- `views.py`: Fichero principal de Flask donde crear vistas para las rutas de la aplicación, así como métodos y consultas a la base de datos.

- **develop.db**: Base de datos, la cual tiene un esquema sencillo. Solo necesitamos tres tablas diferenciadas, una para los usuarios de la aplicación, otra para los datos de los pacientes y otra para las mediciones del examen CAMDEX asociadas a cada paciente.
 - **Patient**: Tabla para datos de pacientes.
 - **CamdexData**: Tabla para datos examen CAMDEX. Cada una de las columnas corresponde con el acrónimo de una prueba (Ej: mmse = 'Mini Mental State de Folstein')
 - **User**: Tabla para usuarios de la aplicación.

La única relación que tenemos entre las tablas Patient y CamdexData es el nombre del paciente, con el que asociaremos los datos personales del pacientes con su examen CAMDEX.

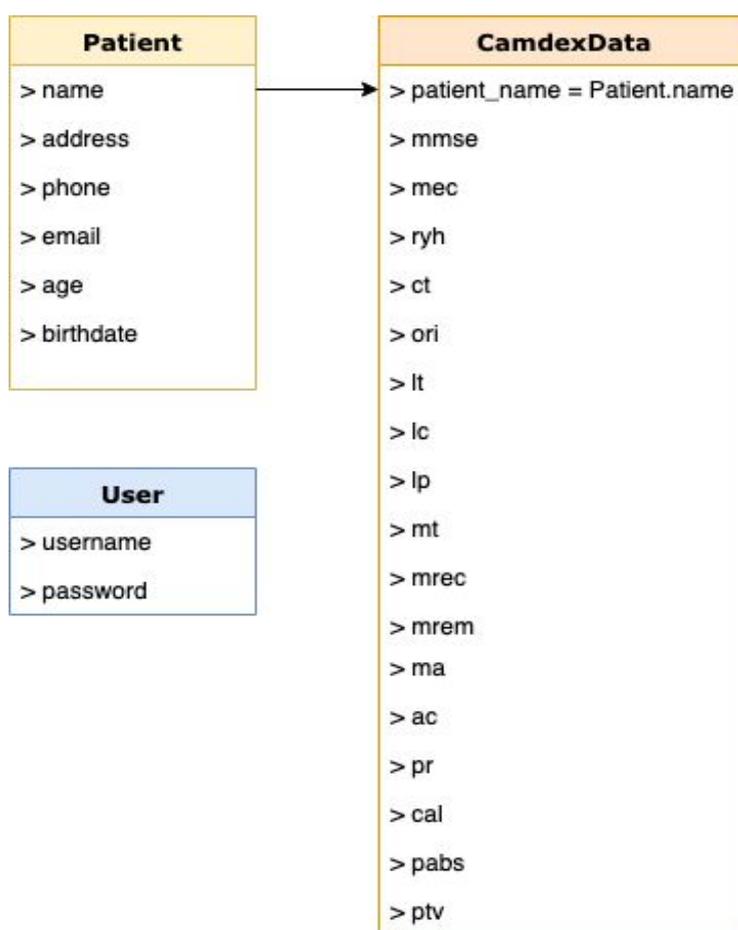


Figura 3.3. Esquema de la base de datos - tablas y columnas

Con esta puesta en contexto de las tecnologías usadas en el proyecto, estructura de directorios, ficheros más importantes y esquema de la base de datos, podemos comenzar a describir el proceso de extracción de datos, así como el el diseño y funcionamiento de la aplicación web.

3.3 Extracción de datos de historiales clínicos

El proceso de extracción de los datos alojados en los historiales clínicos de los pacientes del centro comienza con una previa visualización de cómo se encuentran los datos y de qué manera se nos presentan: si están dispuestos en columnas, un dato seguido de otro en la misma línea, uno debajo de otro, etc.

Al abrir uno de los historiales clínicos, podemos observar que los datos personales del paciente están dispuestos uno seguido del otro en distintas líneas, es decir, el título del dato y su valor (Ej: D/Dña: Juan Pérez González)

DR. D. ANTONIO ALAYÓN

Especialista en Neurología
Postgrado en Electromiografía Clínica
(Universidad Autónoma de Barcelona)

Doctor en Medicina y Cirugía. Col. 2474

American Board Certified en Neurosonología
International Lecturer en Neurosonología
(Sociedad Americana Neurología, Universidad Wake Forest, EEUU)

D/Dña: Juan Pérez González

Dirección: c/El Calvario – La Orotava
Tfño: 645345345
E-Mail: jperez@gmail.com
Edad: 65 años. DNI: 42234456T
F.N.: 18 de septiembre de 1949
F. Ad: 10 de septiembre de 2015

Figura 3.4. Disposición datos personales del paciente

A su vez, podemos ver, unos párrafos más abajo, los datos del examen CAMDEX del paciente, que están dispuestos en una tabla.

Exploración Neuropsicológica

CAMDEX (Examen Cambridge para los trastornos mentales en la vejez)	PD
MMSE	25/30
MEC	29/35
Test de Atención-Memoria-Concentración de Roth y Hopkins	5/10
Camcog Total	78/107
Orientación	10/10
Lenguaje total	26/30
• Lenguaje comprensivo	9/9
• Lenguaje productivo	17/21
Memoria total	13/27
• Memoria reciente	4/4
• Memoria remota	6/6
• Memoria de aprendizaje	3/17
Atención y concentración	4/7
Praxis (ideomotora, ideacional, constructiva)	11/12
Cálculo	2/2
Pensamiento abstracto	4/8
Percepción táctil y visual	7/11

Figura 3.5. Disposición datos examen CAMDEX del paciente

Estos datos, que son los que nos interesa recoger, se repiten a lo largo del historial en función del número de consultas a la que haya acudido el paciente, por ejemplo, si el paciente ha acudido a cinco sesiones, los datos personales y el examen CAMDEX se repetirán a lo largo del historial cinco veces, actualizándose los datos en la tabla de forma acumulativa.

Por ello, en función de la distribución de los datos, se han desarrollado métodos de extracción en función de esta distribución, ya que en todos los historiales clínicos, se repite el mismo patrón.

3.3.1 Metodología y procesos

Una vez descrita la disposición y distribución de los datos del paciente y de su examen CAMDEX en el historial clínico, pasamos a desarrollar los métodos de extracción de la información adecuados para dicha recogida

El proceso de extracción lo hemos dividido en tres partes diferenciadas, principalmente, para poder ver el correcto funcionamiento de cada etapa. A efectos prácticos, estas tres etapas se pueden condensar en una única etapa, que más adelante trataremos en el apartado **Líneas futuras**.

Las etapas del proceso de extracción de datos son las siguientes:

- Conversión del formato del fichero de .odt a .txt

Se realiza una conversión de formato mediante un bash script, ya que la extracción de datos es mucho más fácil y potente mediante ficheros de texto plano, ya que de forma nativa, Python no puede leer ficheros con extensión .odt. Una vez realizada la conversión, se eliminan las líneas en blanco para procesar mejor el texto.

```
#!/bin/bash
# -*- ENCODING: UTF-8 -*-
odt2txt $1.odt > $1.txt
exit
```

Figura 3.6. Bash script simple para la conversión de formato

Una vez se realiza la llamada al bash script desde el método `convert()`, se llama al método `remove_empty_lines()`, que abre el fichero en modo lectura, lee las líneas una a una, luego lo abre en modo edición y selecciona dichas líneas vacías para eliminarlas.

```
@cli.command(help="Realiza una conversión de formato (.odt a .txt) y elimina líneas vacías")
@click.argument("in_filename")
def convert(in_filename):

    os.system('./odt2txt.sh ' + in_filename)

    print('\nFilename: ' + in_filename)
    time.sleep(0.5)
    print("[...]")
    time.sleep(0.5)
    print("[...]")
    time.sleep(0.5)
    print("[...]")
    time.sleep(1)
    print(f'>>> ODT file converted to .txt ✅\n\n ')

def remove_empty_lines(in_filename):
    if not os.path.isfile(in_filename):
        print("{} does not exist ".format())
        return
    with open(in_filename) as filehandle:
        lines = filehandle.readlines()

    with open(in_filename, 'w') as filehandle:
        lines = filter(lambda x: x.strip(), lines)
        filehandle.writelines(lines)
```

Figura 3.7. Método de conversión y eliminación líneas vacías

Con esta etapa, logramos convertir el fichero en formato `.txt` y eliminamos los espacios entre líneas válidas.

- Extracción de los datos personales del paciente

Entrando de lleno en el proceso de extracción de datos, empezaremos por la segunda etapa, recogiendo la información personal del paciente.

Para ello, tenemos el método `patient()`, donde encontramos el principal mecanismo de extracción de datos del historial.

```
with open(in_filename, 'r') as file:
    rows = file.readlines()
    for row in reversed(rows):
        row.strip()
        if re.match('Dirección', row):
            row_parse = row.strip()
            add_address = row_parse[10:]
            break
        elif re.match('Tfno', row):
            row_parse = row.strip()
            add_phone = row_parse[5:]
        elif re.match('Edad', row):
            row_parse = row.strip()
            add_age = row_parse[6:-20]
            add_dni = row_parse[19:]
        elif re.match('F.N.', row):
            row_parse = row.strip()
            add_birth = row_parse[6:]
    extract_patient = Patient(name=add_name,
                              address=add_address,
                              phone=add_phone,
                              age=add_age,
                              dni=add_dni,
                              birthdate=add_birth
                              )
    db.session.add(extract_patient)
    db.session.commit()
```

Figura 3.8. Método de extracción de datos personales del paciente

Primero, abrimos el historial clínico ya convertido en formato `.txt`, que posteriormente, recorreremos línea por línea de forma inversa, ya que los datos que nos interesa recoger son las últimas ocurrencias de cada dato personal y, recorriendo el fichero de esta manera, conseguimos que la última ocurrencia de cada dato sea la primera, facilitando la tarea de encontrar los datos. Luego, comprobamos que si aparece el encabezado del dato que buscamos, mediante `'slicing'`, desechamos los caracteres de la línea encontrada y nos quedamos con los datos que queremos. Una vez tenemos los datos parseados, los introducimos en la base de datos como un paciente nuevo.

Teniendo los datos personales del paciente dentro de la base de datos (en especial, el nombre), podremos concluir el proceso de extracción con la tercera y última etapa.

- Extracción de los datos del examen CAMDEX-R del paciente

Finalizando con el método de extracción de datos del historial clínico, solo nos queda extraer los datos del examen CAMDEX-R del paciente, que realizaremos con el método `camdex()`:

```
with open(in_filename, encoding="utf-8") as file:
    for line in file:
        if re.match('D/Dña', line):
            add_data = []
            x = line[7:]
            add_data.append(x.strip('\n'))
        elif re.match('MMSE', line):
            x = file.readline()
            parse = x[:-4]
            add_data.append(parse.strip('\n'))
        elif re.match('MEC', line):
            x = file.readline()
            parse = x[:-4]
            add_data.append(parse.strip('\n'))
        [...]
        [...]
        [...]
    extract_camdex = CamdexData(patient_name=add_data[0],
                                mmse=add_data[1],
                                mec=add_data[2],
                                )
    db.session.add(extract_camdex)
    db.session.commit()
```

Figura 3.9. Método de extracción examen CAMDEX-R paciente

Este método difiere del método de extracción de la información personal del paciente, ya que al convertir el historial a `.txt`, la tabla del examen CAMDEX-R pierde su formato y los datos no se disponen en la misma línea como la información personal, si no un dato debajo de otro en líneas separadas (Ej: el campo MMSE y en la siguiente línea, los datos).

Para ello, hacemos uso de una lista, la cual, al encontrar la última ocurrencia del título del dato, la incluye en la lista de forma ordenada y haciendo de nuevo slicing para parsear nuestros datos de forma correcta. Una vez se tienen los datos, se introducen en la base de datos accediendo a la lista con el índice correspondiente del dato.

Con esto, habremos finalizado el proceso de extracción del historial clínico y tendremos todos los datos requeridos del paciente en la base de datos.

3.4 Aplicación web: Extract-py

El principal objetivo del Trabajo de Fin de Grado era lograr extraer los datos requeridos por el centro de forma satisfactoria, así como alojarlos en una base de datos de forma estructurada, objetivos que se han logrado con acierto, por lo que se decidió dar un plus al trabajo y darle más valor a los datos extraídos, por lo que se procedió a desarrollar una aplicación web con el fin de poder hacer las tareas de gestor de historiales clínicos para el centro: poder crear nuevos historiales, poder modificarlos o borrarlos a demanda, además de consultarlos de forma automática mediante un buscador, evitando así la principal problemática del centro, que era buscar el historial deseado a mano, además de poder visualizar los datos de forma estructurada en tablas y contrastar la evolución del paciente en su examen CAMDEX-R a lo largo del tiempo entre consulta y consulta.

A continuación trataremos de forma breve las tecnologías usadas para la aplicación y las principales funciones de la misma.

Más adelante, se describirán dos situaciones en la aplicación: visualizar los datos de un paciente después de extraer los datos de su historial de forma automática y otra distinta, que será crear el historial de un paciente directamente desde la aplicación.

3.4.1 Tecnologías utilizadas

Como describimos con anterioridad, Flask es el framework que se ha seleccionado para el desarrollo de la aplicación web, pero hemos hecho uso de otras herramientas, sobre todo, para la visualización de los datos en la aplicación. Además de las tecnologías esenciales para el desarrollo de una aplicación web (HTML/CSS/JS), se han utilizado las siguientes:

- **BootStrap**

BootStrap [17] es un framework CSS desarrollado por Twitter, que permite dar forma a una aplicación web mediante el uso de librerías CSS que incluyen todo tipo de componentes para su uso en la aplicación.

Concretamente, en el proyecto estamos usando su versión 4, la más completa y estable hasta el momento, permitiendo que baje el tiempo de respuesta al cargar menos código en el navegador.

- **DataTables**

DataTables [18] es un plugin para jQuery, con la función de permitirnos crear tablas dinámicas con el fin de olvidarnos de dar formato a las tablas y trabajar con código CSS, ya que la base de estilos que utiliza es la de Bootstrap

- **Plotly.js**

Plotly.js [19] es una librería JavaScript que nos permite crear gráficas interactivas con una gran cantidad de funcionalidades, pudiendo elegir entre una gran variedad de tipos de gráficas para visualizar nuestros datos

3.4.2 Principales funciones

Extract-py permite realizar diversas acciones, como crear nuevos historiales desde la página principal, modificar los propios datos de los historiales o borrarlos del sistema.

Además, se desarrolló una sección de gráficas, en las que el cuerpo médico puede visualizar la evolución del paciente, ya sea favorable o no, de cada prueba correspondiente al examen CAMDEX realizado en cada una de las sesiones a las que ha acudido el paciente.

A continuación, se describirán dos situaciones que se pueden dar en el proyecto:

- Se extraen los datos clínicos del paciente desde su historial y se muestran y se podrán modificar en la aplicación.
- Se crea el historial clínico del paciente directamente en la propia aplicación.

Para ello, se describirán los pasos a seguir para obtener los datos que se desean y poder manipularlos de forma independiente, ya sean extraídos desde los historiales clínicos o directamente creados en la aplicación.

Capítulo 4. Extract-py y algunos ejemplos

Comenzando con la aplicación real del proyecto para el cuerpo médico del Centro Neurológico Antonio Alayón, se nos presentan dos posibles situaciones para la visualización y tratamiento de los datos de los historiales clínicos de los pacientes del centro: extraer los datos automáticamente desde el historial clínico del paciente, recoger los mismos en la base de datos y muestrear en la aplicación o que el especialista cree el historial clínico de un nuevo paciente directamente en la aplicación web.

Por orden de prioridad e importancia en el desarrollo del Trabajo de Fin de Grado, comenzaremos por el caso del Paciente 1, del que extraemos sus datos personales y su examen CAMDEX-R final de su historial clínico.

De nuevo, se recuerda que la información cedida por el Centro Neurológico Antonio Alayón la componen datos ficticios, con el fin de que no se vulnere la identidad de los pacientes del centro, respetando de esta forma la Ley de Protección de Datos Médicos ([1]).

4.1 Historial clínico del Paciente 1 (extracción de datos)

Para el primer ejemplo, usaremos el caso de Juan Pérez González (Paciente 1), paciente del centro, por lo que ya se cuenta con un historial clínico para poder extraer sus datos a continuación.

Para entender mejor el procedimiento a seguir, se ha realizado un vídeo de apoyo para este ejemplo del Paciente 1 [20].

4.1.1 Proceso de extracción de datos clínicos

Comenzando con el proceso de obtención de los datos del Paciente 1, aplicaremos las tres etapas de extracción de datos desarrolladas que se han explicado anteriormente en el apartado **Metodología y procesos**.

Para hacer uso de dichos procesos, se ha desarrollado una interfaz de línea de comandos (Command Line Interface), desde la que se llamará a los métodos concretos del proceso.

```
[(.venv) {11:12}~/src/ETSII/TFG/extract-py/extract:develop x ↻ extract --help
Usage: extract [OPTIONS] COMMAND [ARGS]...

Options:
  --help  Show this message and exit.

Commands:
  camdex  Realiza una extracción a medida de los datos del examen CAMDEX...
  convert Realiza una conversión de formato (.odt a .txt) y elimina líneas...
  patient Realiza una extracción a medida de los datos personales del...
```

Figura 4.1. Terminal con la ayuda del comando 'extract'

Primero, realizaremos la conversión del formato del historial clínico de extensión .odt a .txt, con el fin de poder manejar a posteriori el fichero y posicionarnos dentro de él.

```
[(.venv) {12:55}~/src/ETSII/TFG/extract-py/extract:develop x ↻ extract convert jpg49

Filename: jpg49
[...]
[...]
[...]
>>> ODT file converted to .txt ✓
```

Figura 4.2. Terminal con ejecución del comando 'extract convert'

Una vez realizada la conversión satisfactoriamente, comenzaremos con la extracción de la información personal del Paciente 1.

```
[(.venv) {12:58}~/src/ETSII/TFG/extract-py/extract:develop x ↻ extract patient jpg49.txt
<Patient Juan Pérez González>
Extracting and parsing patient personal data[...]
[...]
[...]
>>> Personal info for patient "Juan Pérez González" was added ✓
```

Figura 4.3. Terminal con la extracción de datos personales con el comando 'extract patient'

Para finalizar con la extracción de los datos del historial clínico del paciente, procedemos a extraer los datos del examen CAMDEX-R más actualizado del historial del paciente.

```
(.venv) {12:58}~/src/ETSII/TFG/extract-py/extract:develop x ↻ extract camdex jpg49.txt
<CamdexData Juan Pérez González>
Extracting and parsing patient CAMDEX data[...]
[...]
[...]
[...]
>>> CAMDEX data for patient "Juan Pérez González" was added ✓
```

Figura 4.4. Terminal con la extracción de los datos del examen CAMDEX-R con el comando 'extract camdex'

Como se puede observar, los datos han sido extraídos correctamente del historial clínico del Paciente 1, por lo que accederemos a la aplicación web para poder visualizar los datos del paciente.

4.1.2 Visualización de la información

Una vez el especialista se haya registrado con sus credenciales en la plataforma, se visualizará en la pantalla principal dos tablas diferenciadas: la primera, con los datos personales del paciente y la segunda, con los datos del examen CAMDEX-R del paciente.

Información del paciente

Nombre	Dirección	Teléfono	E-mail	Edad	DNI	F. Nacimiento
Juan Pérez González	c/El Calvario – La Orotava	645345345		65	42234456T	18 de septiembre de 1949

Mostrando registros del 0 al 0 de un total de 0 registros (filtrado de un total de NaN registros)

Datos de examen CAMDEX

MMSE	MEC	Test Roth y Hopkins	Camcog total	Orientación	Lenguaje total	Lenguaje comprensivo	Lenguaje productivo	Memoria to
22,26,28,27,25,24,25	24,30,35,31,27,29,29	7,9,9,9,7,6,5	82,86,93,96,86,79,78	9,9,10,10,10,10,10	27,26,27,28,27,27,26	9,9,9,9,9,9	18,17,18,19,18,18,17	15,18,20,20,

Mostrando registros del 0 al 0 de un total de 0 registros (filtrado de un total de NaN registros)

Figura 4.5. Página principal con los datos del Paciente 1

4.1.3 Gráficas

Una vez visualizados los datos del Paciente 1 en la página principal de la aplicación, en el menú superior podemos ver un apartado llamado Gráficas. Accederemos para visualizar las diferentes gráficas desarrolladas para cada prueba del examen CAMDEX. El especialista podrá visualizar la evolución que ha experimentado el paciente a lo largo del tiempo, con las mediciones de cada prueba tomadas en cada consulta, por lo que podrá dar un diagnóstico más concreto en función de los resultados visualizados en las gráficas.

Nombre del paciente: AÑADIR MODIFICAR

Datos de examen CAMDEX

MMSE	MEC	Test Roth y Hopkins	Camcog total	Orientación	Lenguaje total	Lenguaje comprensivo	Lenguaje productivo	Memoria to
22,26,28,27,25,24,25	24,30,35,31,27,29,29	7,9,9,9,7,6,5	82,86,93,96,86,79,78	9,9,10,10,10,10,10	27,26,27,28,27,27,26	9,9,9,9,9,9,9	18,17,18,19,18,18,17	15,18,20,20,

Mostrando registros del 0 al 0 de un total de 0 registros (filtrado de un total de NaN registros)

OTRAS GRÁFICAS GRÁFICAS - LENGUAJE GRÁFICAS - MEMORIA

Figura 4.6. Página principal del apartado 'Gráficas'

Tenemos tres grupos diferenciados de gráficas:

- Otras gráficas: Engloban pruebas como MMSE, MEC, etc.
- Lenguaje: Incluye pruebas referentes al lenguaje.
- Memoria: Incluye pruebas referentes a la memoria.

Se ha realizado esta disposición para mejorar la eficiencia de búsqueda de los especialistas, sabiendo a qué sección acudir para visualizar la gráfica de una determinada prueba.

Por ejemplo, si el especialista desea buscar las gráficas de las pruebas MMSE (Mini Mental State de Folstein) y MEC (Mini Examen Cognoscitivo de Lobo y colbs), entonces deberá pulsar en el botón **Otras gráficas**, donde podrá visualizar la evolución del paciente en estas dos pruebas.



Figura 4.7. Visualización de gráficas de pruebas MMSE y MEC

4.1.4 Manejo del historial del paciente

La última funcionalidad que queda por abarcar sería la posibilidad de modificar los datos de los historiales clínicos o borrarlos del sistema. Para ello, accedemos en la página principal de la aplicación y, dentro de **Modificar**, encontramos un buscador de historiales con la posibilidad de **Editar** o **Borrar**.

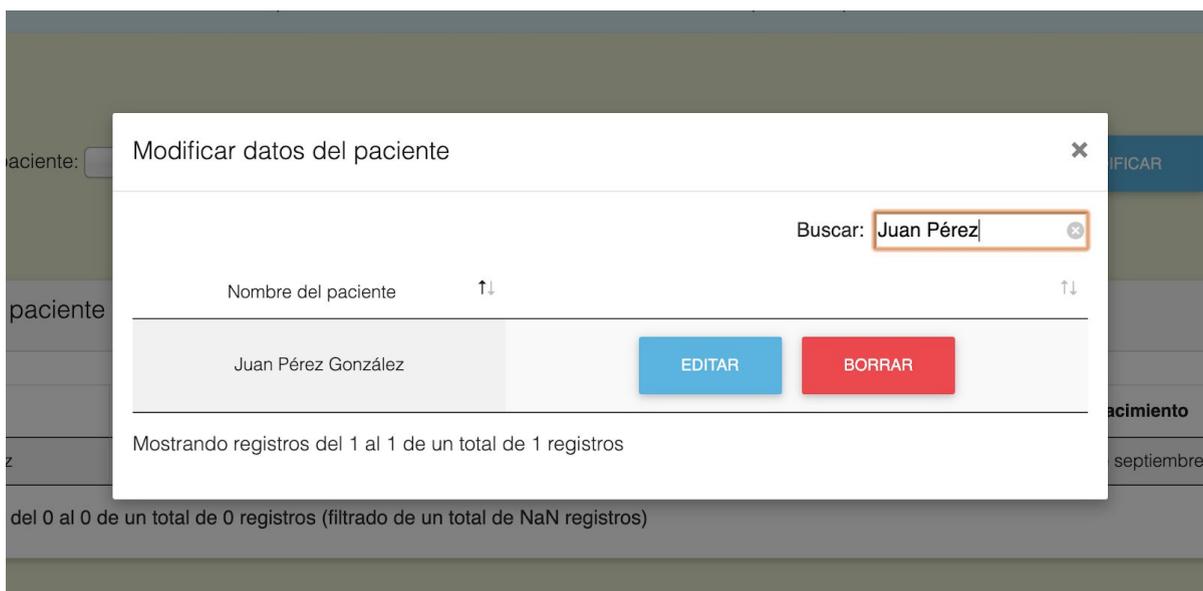


Figura 4.8. Manipulación de historiales clínicos

4.2 Historial clínico del Paciente 2 (aplicación)

Una vez finalizado el análisis del proceso de extracción de datos, muestreo de la información extraída en la aplicación y visualización de gráficas, podemos concluir con una función de la aplicación, la de poder crear historiales clínicos desde cero.

Para entender mejor el procedimiento a seguir, se ha realizado un vídeo de apoyo para este ejemplo del Paciente 2 [21].

4.1.1 Creación del historial clínico

En el caso de que llegue un paciente nuevo a la clínica, tan solo hay que pulsar en el botón **Añadir** de la página principal de la aplicación, apareciendo así un formulario para ingresar los datos personales del paciente y el conjunto de mediciones de su examen CAMDEX.

4.2.2 Visualización de la información

Al igual que con el caso del Paciente 1, una vez los datos estén ingresados en la base de datos, estos estarán disponibles para su manejo, principalmente, agrupados en las tablas de la página principal de la aplicación, así como en el apartado de gráficas, con el fin de poder visualizar y estudiar la evolución que ha sufrido el paciente a lo largo del tiempo.

Como los pasos de visualización de las gráficas y las funciones de edición/eliminación de datos se ejecutan de la misma manera que para el Paciente 2, obviamos estos pasos, que podrán observarse en los vídeos de ayuda que se incluyen en la cabecera del capítulo.

Información del paciente						
Nombre	Dirección	Teléfono	E-mail	Edad	DNI	F. Nacimiento
Ana Méndez Ruiz	c/Morgallana, 12 - La Orotava	667890925	amruiz@gmail.com	76	42367778T	18/04/1943

Mostrando registros del 0 al 0 de un total de 0 registros (filtrado de un total de NaN registros)

Datos de examen CAMDEX												
MMSE	MEC	Test Roth y Hopkins	Camcog total	Orientación	Lenguaje total	Lenguaje comprensivo	Lenguaje productivo	Memoria total	Memoria reciente	Memoria remota	Memoria aprendizaje	Aten conc
26,24	28,26	6,9	73,75	10,8	22,23	8,8	14,15	22,20	4,4	5,6	7,10	3,5

Mostrando registros del 0 al 0 de un total de 0 registros (filtrado de un total de NaN registros)

Figura 4.9. Página principal con los datos del Paciente 2

Capítulo 5. Conclusiones y líneas futuras

En conclusión, este proyecto abarca diversos aspectos relacionados con la extracción y análisis de datos que, con trabajo y constancia, ha sido desarrollado con un excelente grado de calidad, utilidad y que, en colaboración con el Centro Neurológico Antonio Alayón, ha sido posible dar una solución interesante a los diferentes problemas que nos presentaban.

La posibilidad de acceder a fuentes de datos, como son los historiales clínicos, ha supuesto una gran oportunidad para poder desarrollar métodos propios de extracción de datos eficaces para dar una solución óptima a los problemas del centro. Por ello, el desarrollo de una aplicación amigable para cualquier persona, ha sido un incentivo para implementar un análisis de datos por medio de gráficas, con el fin de que los especialistas del centro puedan visualizar de una manera más clara la evolución de sus pacientes y acceder rápidamente a sus historiales clínicos.

Además, la aplicación permitirá a los especialistas del centro llevar un control de los historiales clínicos, ya que todos estarán alojados en una base de datos, donde estarán a su disposición listo para ser consultados al instante.

Como líneas de trabajo futuro, cabe destacar que aún se podrían realizar algunas mejoras tales como:

- Desarrollar una mejora automatizada de los procesos de extracción (automatización de tareas)
- Incluir nuevos campos para los historiales, con el fin de mejorar el análisis de datos (Sexo, municipio, etc.)
- Desarrollar un formulario con gráficas a medida, por medio de consultas a la base de datos.

Capítulo 6. Summary and Conclusions

In conclusion, this project covers several aspects related to the extraction and analysis of data that, with effort and perseverance, has been developed with an excellent degree of quality and usefulness. In this way, in collaboration with the Antonio Alayón Neurological Center, it has been possible to give an interesting solution to the different problems that they presented us.

The possibility of accessing data sources, such as medical records, has supposed a great opportunity to develop effective methods of extracting effective data to provide an optimal solution to the problems of the medical center. Therefore, the development of a friendly application for any kind of user has been an incentive to implement a data analysis through graphs, so that the center's specialists can visualize in a clearer way the evolution of their patients and quickly access to medical records.

In addition, the application will allow the center's specialists to keep track of the medical records, since all will be stored (saved) in a database, where they will be at their disposal ready to be queried instantly.

As future lines of work, it should be noted that some improvements could still be made such as:

- Develop an automated improvement of extraction processes (task automation)
- Include new fields for records, in order to improve data analysis (Sex, municipality, etc.)
- Develop a form with custom graphics, through queries to the database.

Capítulo 7. Presupuesto

El presupuesto que se realiza para el desarrollo de este Trabajo de Fin de Grado se refleja en la siguiente tabla.

Tareas realizadas	Horas	Precio por horas
Revisión bibliográfica de tecnologías	20h	480€
Selección de tecnologías a usar	10h	240€
Selección metodología extracción de datos	10h	240€
Desarrollo métodos de extracción	40h	960€
Estudio diseño aplicación web	30h	720€
Desarrollo de aplicación web	90h	2160€
Documentación y difusión de resultados	10h	240€
TOTAL	210h	5040€

Tabla 7.1: Resumen de horas/precio por trabajo

En la tabla anterior se recoge una relación entre las horas invertidas en cada una de las tareas realizadas en este trabajo y el precio estipulado de cada hora, en relación a la actividad realizada, que serían 24€/hora. Así pues, por un total de 210 horas invertidas, la cantidad necesaria asciende a un total de 5040.00 euros.

Capítulo 8. Bibliografía

[1] BOE. Ley de Autonomía del Paciente 41/2002. URL:
<https://www.boe.es/buscar/act.php?id=BOE-A-2002-22188>.

Acceso: 2019-04-19

[2] Extracción de la información. URL:
https://es.wikipedia.org/wiki/Extracci%C3%B3n_de_la_informaci%C3%B3n.

Acceso: 2019-04-19

[3] Definición de Web Scraping. URL:
https://en.wikipedia.org/wiki/Web_scraping#Techniques

Acceso: 2019-04-19

[4] Definición de historial clínico. URL:
https://es.wikipedia.org/wiki/Historia_cl%C3%ADnica

Acceso: 2019-04-19

[5] Data Science: qué es (y que no es). URL:
<https://aukera.es/blog/data-science-que-es-y-que-no-es/>

Acceso: 2019-05-02

[6] Minería de datos (Data Mining). URL:
https://es.wikipedia.org/wiki/Miner%C3%ADa_de_datos

Acceso: 2019-05-04

[7] Lozano Gallego M., Llinàs Reglà J., López-Pousa S., Vilalta Franch. CAMDEX-R in the clinical evaluation of dementias. URL:
<https://www.ncbi.nlm.nih.gov/pubmed/10937393>, 2000.

Acceso: 2019-05-08

[8] Mas Esquerdo J.J., Robles Sánchez J.I., Barrachina Plaza N. Puntuaciones normativas en la evaluación Cognitiva de Cambridge Revisada (CAMCOG-R). URL:
http://scielo.isciii.es/scielo.php?script=sci_arttext&pid=S1887-85712017000100003, 2017.

Acceso: 2019-05-08

[9] M.F. Folstein. Mini Mental State Examination (MMSE) de Folstein. URL: <https://www.hipocampo.org/folstein.asp>.

Acceso: 2019-05-09

[10] Agencia Estatal de Meteorología. URL: <http://www.aemet.es/es/portada>

Acceso: 2019-05-11

[11] Guido Van Rossum. Python. URL: <https://www.python.org/>, 1991.

Acceso: 2019-05-14

[12] Ross Ihaka, Robert Gentleman. R. URL: <https://www.r-project.org/>, 1993.

Acceso: 2019-05-14

[13] Sébastien Eustace. Poetry. URL: <https://poetry.eustace.io/>, 2018.

Acceso: 2019-05-15

[14] Armin Ronacher. Flask. URL: <https://palletsprojects.com/p/flask/>, 2010.

Acceso: 2019-05-16

[15] Michael Bayer. SQLAlchemy. URL: <https://www.sqlalchemy.org/>, 2007.

Acceso: 2019-05-17

[16] GitHub. Repositorio del TFG. URL: <https://github.com/beejeke/extract-py>, 2019.

[17] Mark Otto, Jacob Thornton. Bootstrap. URL: <https://getbootstrap.com/>, 2011.

Acceso: 2019-06-03

[18] SpryMedia Ltd. DataTables. URL: <https://datatables.net/>, 2007.

Acceso: 2019-06-08

[19] Plot.ly. Plotly.js. URL: <https://plot.ly/javascript/>, 2012.

Acceso: 2019-06-19

[20] Enlace a YouTube con vídeo explicativo del proceso de extracción de datos del Paciente 1. URL:
https://youtu.be/14x-fft__qI

[21] Enlace a YouTube con vídeo explicativo del proceso de creación del historial del Paciente 2. URL:
<https://youtu.be/71QT6fAyFjQ>