



UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería Informática



TFG del Grado en Ingeniería  
Informática

Aplicación web para pacientes  
con párkinson



Presentado por Sandra Díaz Aguilar  
en Universidad de Burgos — 16 de enero  
de 2024

Tutor: Álgvar Anaiz González  
Cotutora: Alicia Olivares Gil







UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería Informática



D. Álgvar Arnaiz González, profesor del departamento de Ingeniería Informática, área de Lenguajes y Sistemas Informáticos.

Expone:

Que el alumno D. Sandra Díaz Aguilar, con DNI 71309532X, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado Aplicación web para pacientes con Parkinson.

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, 16 de enero de 2024

Vº. Bº. del Tutor:

Vº. Bº. del co-tutor:

D. nombre tutor

D. nombre co-tutor





## **Resumen**

Este proyecto pretende servir de ayuda a pacientes con párkinson, un trastorno neurodegenerativo crónico que lamentablemente afecta a más de 150000 personas en España. Se ha desarrollado una aplicación web en la que pacientes y médicos puedan acceder a gráficos con los datos que mide un sensor que llevan los pacientes incorporado, de forma sencilla y fácil de entender. Además utiliza el aprendizaje automático para predecir, según estos datos, cómo avanzará la enfermedad. La aplicación también permite a los médicos gestionar otra información y vídeos de sus pacientes.

## **Descriptores**

Enfermedad de Parkinson, sensor, desarrollo web, aplicación web, datos médicos, análisis de datos, aprendizaje automático, . . .

### **Abstract**

The aim of this project is to help patients with Parkinson's disease, a chronic neurodegenerative disorder that unfortunately affects more than 150,000 people in Spain. A web application has been developed, where patients and doctors can access graphics with data that was measured by a sensor that patients have incorporated. It is accessible in a simple and easy way. The app also uses machine learning to predict, based on this data, how the disease will progress. Furthermore, the app also allows doctors to manage other information and videos of their patients.

### **Keywords**

Parkinson's disease, sensor, web development, web application, medical data, data análisis, machine learning,...



---

# Índice general

---

Índice general	iii
Índice de figuras	iv
Índice de tablas	v
1. Introducción	1
2. Objetivos del proyecto	3
3. Conceptos teóricos	5
3.1. Secciones . . . . .	5
3.2. Referencias . . . . .	5
3.3. Imágenes . . . . .	6
3.4. Listas de ítems . . . . .	6
3.5. Tablas . . . . .	7
4. Técnicas y herramientas	9
5. Aspectos relevantes del desarrollo del proyecto	17
6. Trabajos relacionados	19
7. Conclusiones y Líneas de trabajo futuras	23
Bibliografía	25

---

# Índice de figuras

---

3.1. Autómata para una expresión vacía . . . . .	6
--	---

---

# Índice de tablas

---

3.1. Herramientas y tecnologías utilizadas en cada parte del proyecto	7
---	---



---

# 1. Introducción

---

Descripción del contenido del trabajo y del estructura de la memoria y del resto de materiales entregados.



---

## 2. Objetivos del proyecto

---

El objetivo principal del proyecto es desarrollar una aplicación web que ofrezca a las personas con párkinson una manera intuitiva de llevar un seguimiento de su enfermedad, permitiéndoles ver y predecir su evolución, además de permitir a los médicos gestionar a sus pacientes.

Para ello se han definido una serie de objetivos, entre los que se pueden distinguir objetivos marcados por los requisitos del software y objetivos de carácter técnico:

Objetivos marcados por los requisitos del software:

- Diseñar una aplicación intuitiva y fácil de entender tanto para los pacientes como para los médicos utilizando el framework Flask.
- Crear funcionalidades en la aplicación web que permitan a los pacientes acceder a gráficas que muestren los datos medidos por el sensor y predecir datos futuros sobre su evolución.
- Crear funcionalidades en la aplicación web que permitan a los médicos gestionar la información de sus pacientes, añadir vídeos y datos a sus historiales y ver sus gráficas y predicciones.
- Diseñar una base de datos mediante MariaDB en la que almacenar la información que utiliza la aplicación

Objetivos de carácter técnico:

- Utilizar una metodología ágil de tipo Scrum mediante la herramienta Jira para realizar un seguimiento de las actividades realizadas, así como para organizar temporalmente las tareas que vaya marcando el tutor.

- Realizar la documentación del proyecto completa utilizando LaTeX, de forma progresiva a la evolución del mismo, que muestre a personas ajenas todos los aspectos relacionados con el proyecto realizado.
- Lectura y aprendizaje sobre la enfermedad de Parkinson, sus síntomas y tratamientos, así como el alcance de la enfermedad para obtener el contexto médico del proyecto.
- Experimentación con diferentes técnicas de machine learning implementadas por la biblioteca Scikit-learn para realizar las predicciones.
- Experimentación con diferentes técnicas de realización de gráficos de la biblioteca Matplotlib para mostrar los datos del sensor y predicciones.
- Mejorar los conocimientos sobre Python, machine learning, bases de datos, análisis software y dirección de proyectos que se estudian en el grado.
- Adquirir conocimientos sobre desarrollo web.



---

## 3. Conceptos teóricos

---

En aquellos proyectos que necesiten para su comprensión y desarrollo de unos conceptos teóricos de una determinada materia o de un determinado dominio de conocimiento, debe existir un apartado que sintetice dichos conceptos.

Algunos conceptos teóricos de L<sup>A</sup>T<sub>E</sub>X <sup>1</sup>.

### 3.1. Secciones

Las secciones se incluyen con el comando `section`.

#### Subsecciones

Además de secciones tenemos subsecciones.

#### Subsubsecciones

Y subsecciones.

### 3.2. Referencias

Las referencias se incluyen en el texto usando `cite [?]`. Para citar webs, artículos o libros `[?]`, si se desean citar más de uno en el mismo lugar `[?, ?]`.

---

<sup>1</sup>Créditos a los proyectos de Álvaro López Cantero: Configurador de Presupuestos y Roberto Izquierdo Amo: PLQuiz

### 3.3. Imágenes

Se pueden incluir imágenes con los comandos standard de  $\text{\LaTeX}$ , pero esta plantilla dispone de comandos propios como por ejemplo el siguiente:



Figura 3.1: Autómata para una expresión vacía

### 3.4. Listas de items

Existen tres posibilidades:

- primer item.
- segundo item.

1. primer item.
2. segundo item.

**Primer item** más información sobre el primer item.

**Segundo item** más información sobre el segundo item.

■

Herramientas	App	AngularJS	API REST	BD	Memoria
HTML5		X			
CSS3		X			
BOOTSTRAP		X			
JavaScript		X			
AngularJS		X			
Bower		X			
PHP			X		
Karma + Jasmine		X			
Slim framework			X		
Idiorm			X		
Composer			X		
JSON		X	X		
PhpStorm		X	X		
MySQL				X	
PhpMyAdmin				X	
Git + BitBucket		X	X	X	X
MikTeX					X
TeXMaker					X
Astah					X
Balsamiq Mockups		X			
VersionOne		X	X	X	X

Tabla 3.1: Herramientas y tecnologías utilizadas en cada parte del proyecto

### 3.5. Tablas

Igualmente se pueden usar los comandos específicos de  $\text{\LaTeX}$  o bien usar alguno de los comandos de la plantilla.



---

## 4. Técnicas y herramientas

---

En esta parte de la memoria se presentan las técnicas metodológicas y las herramientas de desarrollo que se han utilizado en los diferentes ámbitos del proyecto.

### Gestión de proyectos

#### Técnica

Durante la gestión del proyecto se ha optado por usar una metodología ágil<sup>2</sup>. Estas metodologías ofrecen un enfoque interactivo e incremental, y se centran en entregar productos de calidad de forma rápida y flexible mediante colaboración activa entre miembros del equipo y clientes.

Se ha elegido esta técnica frente a un enfoque más tradicional (con un plan rígido desde el inicio del proyecto) debido a su flexibilidad, pudiendo adaptarse con el tiempo, respondiendo a los cambios, algo muy a tener en cuenta en un TFG.

Dentro de las metodologías ágiles existen muchas donde elegir. En el proyecto se ha escogido entre dos, Scrum y Kanban.

- Scrum: define roles específicos como el product owner o el equipo de desarrollo, que participan en eventos predeterminados como el daily scrum o el sprint review. Trabaja con iteraciones de 2 a 4 semanas, perfectamente detalladas, llamadas sprints, tras las que se debe entregar un incremento del producto.

---

<sup>2</sup><https://www.iebschool.com/blog/que-son-metodologias-agiles-agile-scrum/>

- Kanban: no tiene roles ni eventos específicos, ofreciendo más flexibilidad. En vez de trabajar con sprints, se opera en un flujo continuo, en el que se agregan y retiran tareas en cualquier momento.

Finalmente se escogió utilizar Scrum, ya que la idea de los sprints era la que más conveniente le parecía a la alumna. Dividir las tareas en tiempos de 2 a 4 semanas puede hacer que no te asustes con el trabajo que queda por venir, centrándote en conseguir acabar las tareas para la fecha final del sprint. Además las reuniones de fin de cada sprint sirven para tener contacto con el tutor, comentando las dificultades encontradas y ajustando la cantidad de trabajo para el siguiente sprint.

## Herramienta

Para llevar a cabo la planificación temporal del proyecto, utilizando una metodología Scrum, se optó por la herramienta Jira. Otra herramienta usada frecuentemente en los TFGs es ZenHub. A continuación se incluye un resumen de cada alternativa, incluyendo comparativas<sup>3 4</sup> y justificando la decisión:

- Jira<sup>5</sup>: herramienta de gestión de proyectos desarrollada por Atlassian. Soporta diferentes metodologías de desarrollo como Scrum o Kanban. Ofrece una versión gratuita con funcionalidades básicas y versiones premium de pago.
- ZenHub<sup>6</sup>: herramienta que se integra directamente con GitHub, ofreciendo funciones de gestión ágil directamente en el entorno de desarrollo. Actualmente es una herramienta de pago, no ofrece una versión gratuita.

Ambas opciones proporcionan herramientas para la planificación de sprints, seguimiento de tareas y gestión del flujo de trabajo mediante gráficos e informes de rendimiento, que es lo que se buscaba en el proyecto. Además ambas ofrecen características que mejoran la colaboración del equipo, asignando tareas a cada miembro, comentando problemas, ... (esta característica no es relevante ya que el equipo de trabajo está formado únicamente por la alumna).

---

<sup>3</sup><https://www.getapp.es/compare/9953/110953/jira/vs/zenhub>

<sup>4</sup><https://www.techrepublic.com/article/zenhub-vs-jira/>

<sup>5</sup><https://www.atlassian.com/software/jira>

<sup>6</sup><https://www.zenhub.com/>

Dado que se está utilizando GitHub para mantener un seguimiento del progreso del proyecto, ZenHub sería una opción interesante y fácil de integrar pero se ha elegido utilizar Jira mayoritariamente por ser una opción gratuita. Además es la herramienta que se utiliza en la asignatura de Gestión de Proyectos, por lo que se tenía experiencia previa.

## Lenguaje de programación

Como lenguaje de programación se ha usado Python ya que, además de tener una sintaxis simple y fácil de entender, dispone de herramientas para trabajar con desarrollo web, análisis de datos e inteligencia artificial, que son los campos que se han trabajado durante el proyecto.

Python<sup>7</sup> es un lenguaje de código abierto orientado a objetos. Destaca entre los principiantes gracias a su sintaxis fácil de entender. Se utiliza para prácticamente todo: automatización de industrias, creación de videojuegos, desarrollo software, análisis y representación de datos, machine learning o blockchain.

Al ser un lenguaje de alto nivel, su código fuente (fácil de entender por los programadores) con extensión .py debe ser interpretado por el intérprete de Python para convertirlo en bytecode, ejecutado por la máquina virtual de Python para generar las salidas por consola o modificaciones de archivos.

A continuación se presentan las bibliotecas de Python utilizadas, así como el framework utilizado para la aplicación web y los entornos de desarrollo sobre los que se programó:

## Bibliotecas

- Scikit-learn<sup>8 9</sup>: librería gratuita (código abierto) para Python que cuenta con gran variedad de algoritmos de aprendizaje automático como clustering, clasificación o regresión. Se utilizará durante el proyecto para la predicción de la evolución de los pacientes. Además permite realizar el preprocesamiento de los datos de forma sencilla y una evaluación de la calidad del modelo tras su realización. Es compatible con otras librerías de Python como NumPy o Matplotlib (de la que se hablará a continuación). Es conocida por su facilidad de uso, con herramientas simples y eficientes, y por la abundante documentación y comunidad activa.

---

<sup>7</sup><https://blog.hubspot.es/website/que-es-python>

<sup>8</sup><https://scikit-learn.org/stable/>

<sup>9</sup><https://www.master-data-scientist.com/scikit-learn-data-science/>

- Matplotlib<sup>10,11</sup>: biblioteca gratuita utilizada para crear visualizaciones estáticas, animadas o interactivas en Python. Se utilizará para mostrar gráficamente los datos recogidos por el sensor de los pacientes. Puede crear histogramas, diagramas de barras, circulares, etc con pocas líneas de código. Permite personalizar el estilo visual y exportar a múltiples formatos de archivo.
- Pandas<sup>12</sup>: herramienta de manipulación y análisis de datos de código abierto, potente, flexible y fácil de usar. Permite leer/escribir datos de archivos CSV, Excel o de bases de datos SQL, así como manipularlos mediante filtrados, agrupaciones, combinaciones, etc
- Numpy<sup>13</sup>: biblioteca creada para computación científica y numérica. Permite trabajar con grandes volúmenes de datos en forma de arrays multidimensionales y realizar operaciones matemáticas con ellos. Combina la flexibilidad de Python y la eficiencia de C. Cabe destacar que es la base sobre la que se construyen otras bibliotecas comentadas anteriormente, como Pandas o Scikit-learn.

## Frameworks

- Flask<sup>14,15</sup>: herramienta utilizada para desarrollar la aplicación web del proyecto con el lenguaje Python. Promete facilitar la creación de aplicaciones web con el patrón Modelo Vista Controlador. Incluye un servidor web de desarrollo para poder observar los avances en la aplicación sin necesidad de disponer de un servidor web. Soporta el uso de cookies y sesiones. Existen multitud de extensiones como Bootstrap o login. Es open source y existe mucha documentación, tanto en su página oficial como por parte de otros usuarios. Dentro de Flask se estudió el diseño web adaptable:

## Diseño web adaptable

Se presentó la idea de que la aplicación web debía ajustarse automáticamente según el dispositivo en el que se estuviera visualizando (ordenador, móvil, tableta, ...). Para ello hay que utilizar un framework receptivo, que

---

<sup>10</sup><https://matplotlib.org/>

<sup>11</sup><https://datascientest.com/es/todo-sobre-matplotlib>

<sup>12</sup><https://pandas.pydata.org/>

<sup>13</sup><https://numpy.org/>

<sup>14</sup><https://flask.palletsprojects.com/en/3.0.x/>

<sup>15</sup><https://www.epitech-it.es/flask-python/>



facilite la creación de interfaces ajustables.

Se han estudiado tres alternativas<sup>16,17</sup> de herramientas:

- Bootstrap<sup>18</sup>: herramienta popular, ampliamente utilizada y de la que existe mucha documentación y ayuda por parte de otros usuarios en la web. Dispone de una rejilla para dividir el contenido en columnas y filas, y 4 clases (teléfono, tableta, portátiles pequeños y normales), útiles para crear el diseño adaptable que se busca. Dispone de componentes predefinidos como botones, formularios o desplegables, además de gran variedad de estilos prediseñados que se pueden integrar fácilmente en la aplicación. Es conocido por ser fácil de usar e implementar, incluso para principiantes.
- Materialize CSS<sup>19</sup>: está basada en el sistema de Material Design creado por Google, por lo que es fácil de integrar con otros productos de Google. Dispone de estilos modernos y una composición minimalista. Ofrece temas de HTML estático pero son de pago.
- ZURB Foundation<sup>20</sup>: esta herramienta es también bastante popular. Dispone de un enfoque más modular, lo que ofrece una mayor flexibilidad, con componentes personalizables, para tener control total sobre la apariencia de la aplicación. La curva de aprendizaje de esta herramienta es mayor, por lo que no es adecuada para principiantes.

Finalmente se decidió utilizar Bootstrap porque, dada mi falta de experiencia en el mundo del Front-end y las aplicaciones web, era un factor importante que existiera abundante documentación en Internet sobre la herramienta. Que existan estilos y componentes predefinidos en vez de tener más flexibilidad como con ZURB Foundation es otro punto a favor teniendo en cuenta mi inexperiencia. Se ha optado por una herramienta gratuita y que sea fácil de usar y de implementar en Flask.

---

<sup>16</sup><https://blog.infranetworking.com/alternativas-a-bootstrap/>

<sup>17</sup><https://www.altexsoft.com/blog/most-popular-responsive-css-frameworks-bootstrap-foundation-materialize-pure-and-more/>

<sup>18</sup><https://getbootstrap.com/>

<sup>19</sup><https://materializecss.com/>

<sup>20</sup><https://get.foundation/>

## Entornos de desarrollo

- Jupyter Notebook<sup>21</sup> es una aplicación web de código abierto que se ha utilizado durante la experimentación y pruebas de las librerías de Python. Permite ejecutar código de forma interactiva en celdas, visualizando el resultado paso a paso. Permite crear y compartir documentos que combinan texto, gráficos interactivos y código ejecutable. Resultó muy útil para practicar con la librería de Matplotlib, teniendo el código y los gráficos resultantes en el mismo documento.
- Visual Studio Code (vscode) es un entorno de desarrollo integrado, desarrollado por Microsoft, de código abierto y gratuito. Admite muchos lenguajes de programación: Java, Python, C++, JavaScript, ... Durante el proyecto se utiliza para programar la aplicación web con Flask (Python). Tiene una interfaz de usuario simple y técnicas como el autocompletado que ayudan a los desarrolladores. Existen muchas extensiones en el mercado que proporcionan nuevos lenguajes o temas. Ofrece integración con GitHub, pudiendo subir directamente los códigos desde vscode. Se había utilizado con anterioridad, por lo que se escogió como entorno de desarrollo sobre el que trabajar.

## Editor de texto

Para llevar a cabo la documentación del proyecto (la memoria y anexos) se ha debido escoger entre los siguientes editores de texto:

- Microsoft Word: dispone de una interfaz de usuario intuitiva y fácil de usar. Es ampliamente utilizado en entornos académicos y profesionales. Dispone de funcionalidades avanzadas como tablas, gráficos o revisión de documentos. Tiene menor control sobre el formato.
- LaTeX: utilizado para documentos científicos, técnicos o académicos. Dispone de gran capacidad de gestión de fórmulas matemáticas y referencias bibliográficas. Ofrece un control preciso sobre el formato del documento. Es menos intuitivo y más complicado de aprender, ya que utiliza comandos y código.

Finalmente se optó por utilizar LaTeX porque, aunque se tuvo que invertir tiempo en aprender a manejar el editor, se obtuvo un mejor resultado visual gracias a las plantillas disponibles para realizar la documentación.

---

<sup>21</sup><https://jupyter.org/>

Se utilizó Overleaf como herramienta para editar los documentos de LaTeX de forma colaborativa en la nube.

## Prototipado

Para realizar el diseño de interfaces de la aplicación web se ha utilizado la herramienta de Pencil<sup>22</sup>.

Se barajó la idea de utilizar Adobe XD<sup>23</sup>, otra herramienta de diseño y prototipado que permite diseñar experiencias de usuario interactivas tanto para web como para móvil. Integra funciones de diseño y prototipado. Puede importar recursos de otras aplicaciones como Adobe Photoshop y se puede exportar el proyecto una vez finalizado. Se debe iniciar sesión para utilizarla. Se optó por Pencil Project ya que esta herramienta se había utilizado previamente en la asignatura de Interacción Hombre Máquina, por lo que la alumna estaba familiarizada con su interfaz. Es una herramienta muy sencilla, con varios paquetes de diseño predeterminados pero ampliable con colecciones encontradas en la web. Es open source y no se debe iniciar sesión para usarla.

Para realizar los diagramas del proyecto se ha utilizado Draw.io<sup>24</sup>, una herramienta online gratuita que permite crear diagramas y mapas mentales de forma intuitiva. Existe también una versión de escritorio.

Los usuarios deben arrastrar y soltar elementos que se encuentran organizados según el tipo de diagrama. Permite crear diversos diagramas (diagrama de flujo, UML, de red, organigramas, mapas conceptuales, ...) mediante gráficos predefinidos como bloques, clases, atributos, actores, conectores, etc. Posteriormente se pueden exportar en diversos formatos (PNG, PDF, SVG, ...).

Un dato interesante, aunque no se ha utilizado en el proyecto, es que permite la colaboración en tiempo real con otros usuarios.

Se ha utilizado para crear todos los diagramas del proyecto debido a su simplicidad y a que la alumna lo había usado en otras ocasiones.

## Base de datos

Se necesitó escoger una herramienta para organizar los datos de los pacientes recolectados por el sensor, así como los datos que serán utilizados por la aplicación web. Se estudiaron dos alternativas de sistemas gestores

---

<sup>22</sup><https://pencil.evolus.vn/Features.html>

<sup>23</sup><https://helpx.adobe.com/es/xd/user-guide.html>

<sup>24</sup><https://app.diagrams.net/>

de bases de datos relacionales de código abierto<sup>25</sup>: MariaDB y PostgreSQL. MariaDB es una bifurcación de MySQL, creada por sus desarrolladores originales, después de que Oracle adquiriera MySQL. En cambio, PostgreSQL ha evolucionado de forma más independiente. Ambas utilizan SQL estándar como lenguaje de consulta, por lo que resultarán conocidas para la alumna tras haber estudiado Bases de Datos durante el grado.

Ambas garantizan la integridad de los datos gracias a sus propiedades ACID y admiten extensiones para aumentar su funcionalidad.

A rasgos generales son muy parecidas, pero las implementaciones específicas pueden variar. Se decidió usar MariaDB por tener una comunidad activa que ofrece mucha ayuda en la web, además de por recomendación del tutor.

Como herramienta para crear la base de datos se ha utilizado HeidiSQL<sup>26</sup>, una herramienta gratuita de código abierto con la que se pueden administrar bases de datos. Permite crear, modificar o eliminar bases de datos, así como sus tablas y campos, con una interfaz intuitiva.

## Repositorio

La plataforma elegida para llevar el control de versiones y el hosting del repositorio ha sido GitHub<sup>27</sup>.

Es una plataforma de desarrollo colaborativo basada en la nube que permite gestionar proyectos software, actualizar versiones de códigos fuentes y facilitar la colaboración entre desarrolladores.

Se ha creado un repositorio en GitHub para almacenar el código fuente y otros documentos del proyecto. Con su sistema de control de versiones se puede ir viendo la evolución del proyecto a lo largo del tiempo. Si hubiera varios desarrolladores podrían trabajar de manera colaborativa fácilmente creando ramas y posteriormente fusionándolas, comparando los cambios realizados.

Se ha optado por hacer el repositorio privado durante la realización del proyecto, invitando únicamente al tutor y a la cotutora, para posteriormente hacerlo público, dejando el proyecto al alcance de otros desarrolladores que podrían estar interesados en él.

---

<sup>25</sup><https://aws.amazon.com/es/compare/the-difference-between-mariadb-vs-mysql/>

<sup>26</sup><https://www.mauriciodeveloper.com/post/heidisql-la-herramienta-imprescindible-para-administrar-bases-de-datos-mysql-529>

<sup>27</sup><https://github.com/about>

---

## 5. Aspectos relevantes del desarrollo del proyecto

---

Este apartado pretende recoger los aspectos más interesantes del desarrollo del proyecto, comentados por los autores del mismo. Debe incluir desde la exposición del ciclo de vida utilizado, hasta los detalles de mayor relevancia de las fases de análisis, diseño e implementación. Se busca que no sea una mera operación de copiar y pegar diagramas y extractos del código fuente, sino que realmente se justifiquen los caminos de solución que se han tomado, especialmente aquellos que no sean triviales. Puede ser el lugar más adecuado para documentar los aspectos más interesantes del diseño y de la implementación, con un mayor hincapié en aspectos tales como el tipo de arquitectura elegido, los índices de las tablas de la base de datos, normalización y desnormalización, distribución en ficheros<sup>3</sup>, reglas de negocio dentro de las bases de datos (EDVHV GH GDWRV DFWLYDV), aspectos de desarrollo relacionados con el WWW... Este apartado, debe convertirse en el resumen de la experiencia práctica del proyecto, y por sí mismo justifica que la memoria se convierta en un documento útil, fuente de referencia para los autores, los tutores y futuros alumnos.



---

## 6. Trabajos relacionados

---

En este apartado se van a comentar trabajos y proyectos realizados en el campo del proyecto en curso, lo que nos va a ofrecer contexto de lo que hay actualmente en el mercado, pudiendo explicar qué aporta nuestro proyecto. Nuestro proyecto fusiona el campo de la informática y la medicina, relacionando la creación de aplicaciones web y el uso de machine learning con necesidades médicas, en concreto de la enfermedad de Parkinson.

El campo de la medicina es un campo que está bastante privatizado pero a su vez se publican gran variedad de artículos. Se puede buscar información sobre el párkinson en todo tipo de fuentes fiables como bibliotecas universitarias, centros de investigación o revistas científicas.

Algunas de estas fuentes podrían ser la Organización Mundial de la Salud<sup>28</sup>, la PMC<sup>29</sup> que ofrece revistas biomédicas y artículos de la Biblioteca Nacional de EE.UU., revistas digitales como Journal of Parkinson's Disease<sup>30</sup> que publica investigaciones relacionadas con la enfermedad de Parkinson o la propia biblioteca de la UBU<sup>31</sup>.

A continuación, se podrían exponer varios de los artículos que se han utilizado como fuente de información para la realización del proyecto pero no procede.

El campo de la informática y la programación web está en constante evolución, diariamente utilizamos aplicaciones web, ya sea en el trabajo, con herramientas para colaboración en tiempo real, o en nuestro tiempo libre con juegos, redes sociales o comercio electrónico. Existen miles de aplicaciones

---

<sup>28</sup><https://www.who.int/es/news-room/fact-sheets/detail/parkinson-disease>

<sup>29</sup><https://www.ncbi.nlm.nih.gov/pmc/?term=parkinson+disease>

<sup>30</sup><https://www.journalofparkinsonsdisease.com/>

<sup>31</sup>[https://ubucat.ubu.es/discovery/search?vid=34BUC\\_UBU : VU1](https://ubucat.ubu.es/discovery/search?vid=34BUC_UBU : VU1)

web en el mercado, por lo que vamos a centrarnos en aquellas que permiten visualizar datos médicos a pacientes y personal sanitario. Tras la época de restricciones debidas a la pandemia, se fomentó el uso de aplicaciones médicas para recibir atención sanitaria, así como para consultar registros médicos, gestionar citas y muchas otras funcionalidades.

Se presentan a continuación algunas de estas aplicaciones:

- 75health<sup>32</sup>: permite a los médicos gestionar las historias clínicas, así como emitir recetas a sus pacientes. Por su parte, los pacientes tienen acceso a sus datos médicos y pueden contactar con su especialista. El problema de esta aplicación es que solo está disponible en Estados Unidos e India, además de que, si quieres todas sus funcionalidades, debes obtener una versión de pago, lo que la hace poco accesible.
- EpicCare EMR<sup>33</sup>: aplicación destinada a sanitarios que trabajan en grandes centros médicos, que les permite gestionar la documentación y organizarse el trabajo. A los pacientes les permite hablar y programar citas con sus médicos (incluso a través de videoconferencias), así como ver su información e historial médico. El problema es que no existe versión gratuita, aunque se puede solicitar una demo. Además, solo se encuentra disponible en inglés.
- MyChart<sup>34</sup>: otra aplicación parecida a las anteriores, que permite visualizar resultados de laboratorio, historial médico, programar citas y hablar con tu médico, y mucho más. Aunque nuevamente no está disponible en España.
- PatientsLikeMe<sup>35</sup>: en este caso se trata de una aplicación web destinada a pacientes con enfermedades crónicas como es el párkinson. Se utiliza mayoritariamente para compartir experiencias y obtener apoyo del resto de usuarios. También pueden realizar un seguimiento de su enfermedad y sus tratamientos. Por desgracia, está igualmente solo disponible en inglés.

Por su parte, el campo del machine learning o aprendizaje automático aplicado a la medicina es muy innovador. Existen proyectos que analizan grandes cantidades de datos médicos para realizar predicciones que pueden ayudar con diagnósticos y tratamientos:

---

<sup>32</sup><https://www.75health.com/>

<sup>33</sup><https://www.emrsystems.net/epic-emr-software/>

<sup>34</sup><https://www.mychart.org/Features>

<sup>35</sup><https://www.patientslikeme.com/>



Un ejemplo es un proyecto publicado en la base de datos científica de la Universidad de la Rioja, en el que analizaron 13 métodos de machine learning, para comprobar cuál era más útil en la predicción de la diabetes mellitus tipo 2 en pacientes mayores de edad<sup>36</sup>. Llegaron a la conclusión de que el modelo LightGBM es el que mejores resultados obtuvo de precisión, sensibilidad, tasa de clasificación errónea, etc. Teniendo en cuenta que la diabetes es una de las diez primeras causas de mortalidad entre la población adulta, realizar diagnósticos tempranos de forma rápida gracias a la inteligencia artificial ayudará a prevenir futuras complicaciones.

Otro buen proyecto relacionado con el uso del aprendizaje automático para predecir enfermedades es el TFG entregado por Javier Pérez Córdova en la Universidad de Cataluña<sup>37</sup>. En este trabajo se tratan de aplicar modelos de machine learning basados en árboles de decisión para detectar el cáncer de mama de forma menos invasiva que una mamografía. Se prueban varios métodos diferentes, obteniendo resultados más y menos óptimos, por lo que se llega a la conclusión de que serían necesarios más datos para conseguir mejores resultados de los obtenidos en dicho proyecto.

Personalmente creo que sería interesante invertir en proyectos de este tipo, que ayuden a diagnosticar enfermedades como el cáncer de mama, cuya incidencia está aumentando en España, de forma indolora para los pacientes.

Otro ejemplo de uso de machine learning para ayudar, en este caso, a pacientes con párkinson, es el TFG realizado por un compañero de la carrera el curso pasado. Debido al acuerdo de colaboración entre la UBU y la Asociación Párkinson Burgos, se han llevado a cabo varios trabajos de fin de grado relacionados con esta enfermedad. En este caso se comentará el proyecto de Catalin Andrei Cacuci, estudiante de Ingeniería Informática que presentó el curso pasado un TFG titulado “Identificación de Parkinson por visión artificial”<sup>38</sup>.

El proyecto consistió en crear un sistema que fuera capaz de detectar, mediante visión artificial, la presencia de bradicinesia (un síntoma presente en personas con párkinson que provoca ralentización del movimiento). Los individuos debían grabar un vídeo haciendo un movimiento de pinza con los dedos índice y pulgar y el sistema detectaba esa alteración de movimiento. En mi opinión, es un proyecto que podría llegar a ser muy útil para realizar un primer diagnóstico, especialmente tras vivir una experiencia como la pandemia que fomentó el uso de telemedicina.

---

<sup>36</sup><https://dialnet.unirioja.es/servlet/articulo?codigo=8510612>

<sup>37</sup><https://openaccess.uoc.edu/bitstream/10609/127711/6/javipercorTFM0121memoria.pdf>

<sup>38</sup><https://github.com/cataand/tfg-paddel>

Existen más TFG realizados en la UBU en colaboración con esta asociación, pero me gustaría comentar en especial el realizado por Sara González Bárcena, estudiante de Ingeniería de la Salud, que presentó el curso pasado un TFG titulado “Detección de la actividad muscular de las personas con enfermedad de Parkinson” <sup>39</sup>.

Se incluye este proyecto en el apartado de trabajos relacionados ya que describe la realización de un sensor que analiza la marcha de pacientes con párkinson, parecido al sensor desarrollado por Sense4Care en 2020, que genera los CSV de datos que se utilizan en nuestro proyecto. En especial, el sensor desarrollado por Sara es capaz de analizar la duración del ejercicio, el número de bloqueos durante el período de actividad y, si hay un desequilibrio entre ambos lados del cuerpo del paciente. Incorpora sensores inerciales (acelerómetro y giroscopio) para detectar la información.

Este tipo de proyectos son muy importantes ya que diseñan dispositivos muy específicos, que suelen ser poco comunes y por lo tanto costosos, pero muy útiles para ayudar al personal sanitario a diseñar terapias más precisas y personalizadas, mejorando la calidad de vida de los pacientes.

Tras comentar el contexto actual alrededor de los temas que se tratan en el proyecto, se procede a justificar resumidamente la necesidad de este. Como se ha podido observar, las aplicaciones y aplicaciones web están a la orden del día. Son utilizadas diariamente por personas de todas las edades y en ámbitos de la vida muy variados. Específicamente en el ámbito médico se empezaron a popularizar tras la cuarentena que nos tocó vivir durante la pandemia hace unos años. Lamentablemente las personas no podían acudir al médico y por ello se detectaron enfermedades de forma tardía o no se pudieron monitorizar las enfermedades de los pacientes de forma correcta, afectando gravemente a su salud. Aplicaciones como la que se ha desarrollado en este proyecto, que permiten a pacientes y médicos llevar un seguimiento de la enfermedad de forma remota, e incluso predecir la tendencia de la enfermedad a lo largo del tiempo podrían haber sido de mucha utilidad entonces, pero también ahora.

Se han mostrado aplicaciones existentes actualmente en el mercado, que permiten a pacientes consultar sus datos médicos y otras muchas funcionalidades, pero la mayoría son en inglés y de pago. Por ello hacen falta aplicaciones web como la desarrollada, para pacientes de nuestro país, y gratuitas, siendo accesibles por todo el mundo sin importar sus recursos.

---

<sup>39</sup>[https://github.com/saragonzalezbarcelona/TFG\\_Deteccion\\_Activ\\_Muscular](https://github.com/saragonzalezbarcelona/TFG_Deteccion_Activ_Muscular)

---

## **7. Conclusiones y Líneas de trabajo futuras**

---

Todo proyecto debe incluir las conclusiones que se derivan de su desarrollo. Éstas pueden ser de diferente índole, dependiendo de la tipología del proyecto, pero normalmente van a estar presentes un conjunto de conclusiones relacionadas con los resultados del proyecto y un conjunto de conclusiones técnicas. Además, resulta muy útil realizar un informe crítico indicando cómo se puede mejorar el proyecto, o cómo se puede continuar trabajando en la línea del proyecto realizado.



---

## **Bibliografía**

---