



UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería  
Informática**

**Clasificación del grado de  
Parkinson mediante  
aprendizaje supervisado  
Documentación Técnica**



Presentado por Jorge Martínez Martín  
en Universidad de Burgos — 6 de julio de 2024  
Tutor: Álvar Arnaiz González



---

# Índice general

---

<b>Índice general</b>	<b>i</b>
<b>Índice de figuras</b>	<b>iii</b>
<b>Índice de tablas</b>	<b>iv</b>
<b>Apéndice A Plan de Proyecto Software</b>	<b>1</b>
A.1. Introducción . . . . .	1
A.2. Planificación temporal . . . . .	2
A.3. Estudio de viabilidad . . . . .	8
<b>Apéndice B Especificación de Requisitos</b>	<b>15</b>
B.1. Introducción . . . . .	15
B.2. Objetivos generales . . . . .	15
B.3. Catálogo de requisitos . . . . .	15
B.4. Especificación de requisitos . . . . .	17
<b>Apéndice C Especificación de diseño</b>	<b>35</b>
C.1. Introducción . . . . .	35
C.2. Diseño de datos . . . . .	35
C.3. Diseño procedimental . . . . .	40
C.4. Diseño arquitectónico . . . . .	42
<b>Apéndice D Documentación técnica de programación</b>	<b>45</b>
D.1. Introducción . . . . .	45
D.2. Estructura de directorios . . . . .	45
D.3. Manual del programador . . . . .	48

D.4. Compilación, instalación y ejecución del proyecto . . . . .	49
D.5. Pruebas del sistema . . . . .	51
<b>Apéndice E Documentación de usuario</b>	<b>63</b>
E.1. Introducción . . . . .	63
E.2. Requisitos de usuarios . . . . .	63
E.3. Instalación . . . . .	63
E.4. Manual del usuario . . . . .	64
<b>Apéndice F Anexo de sostenibilización curricular</b>	<b>73</b>
F.1. Introducción . . . . .	73
F.2. Impacto del software a nivel de consumo . . . . .	73
F.3. Igualdad y accesibilidad . . . . .	74
F.4. Conclusiones . . . . .	75
<b>Bibliografía</b>	<b>77</b>

---

# Índice de figuras

---

B.1. Diagrama de casos de uso . . . . .	19
C.1. Diagrama entidad-relación . . . . .	36
C.2. Diagrama relacional de la aplicación . . . . .	37
C.3. Diagrama de secuencia de clasificación de los vídeos . . . . .	41
C.4. Diagrama de despliegue . . . . .	43
E.1. Página principal . . . . .	64
E.2. Página de inicio de sesión . . . . .	65
E.3. Página de información . . . . .	65
E.4. Página del administrador . . . . .	66
E.5. Página de añadir usuarios . . . . .	66
E.6. Página de modificar usuarios . . . . .	67
E.7. Página del médico . . . . .	68
E.8. Página de evolución del paciente . . . . .	68
E.9. Página de medicinas del paciente . . . . .	69
E.10. Página de añadir medicinas . . . . .	69
E.11. Página de vídeos del paciente . . . . .	70
E.12. Página de añadir vídeos . . . . .	71
E.13. Página del médico . . . . .	71

---

# Índice de tablas

---

A.1. Tabla de costes totales . . . . .	10
A.2. Licencias: dependencias o utilidades . . . . .	13
B.1. CU-1 Iniciar sesión. . . . .	20
B.2. CU-2 Cerrar sesión. . . . .	21
B.3. CU-3 Visualizar evolución. . . . .	22
B.4. CU-4.1 Dar de alta a usuarios. . . . .	23
B.5. CU-4.2 Dar de baja a usuarios. . . . .	24
B.6. CU-4.3 Modificar usuarios. . . . .	25
B.7. CU-4.4 Visualizar listado de usuarios. . . . .	26
B.8. CU-5.1 Añadir vídeos. . . . .	27
B.9. CU-5.2 Eliminar vídeos. . . . .	28
B.10.CU-5.3 Clasificar vídeos. . . . .	29
B.11.CU-6.1 Asignar paciente. . . . .	30
B.12.CU-6.2 Desasignar paciente. . . . .	31
B.13.CU-7.1 Añadir medicina. . . . .	32
B.14.CU-7.2 Eliminar medicina. . . . .	33
B.15.CU-7.3 Visualizar listado de medicinas de un paciente. . . . .	34
C.1. Diccionario de datos: Admin . . . . .	38
C.2. Diccionario de datos: Doctor . . . . .	38
C.3. Diccionario de datos: Patient . . . . .	39
C.4. Diccionario de datos: Video . . . . .	39
C.5. Diccionario de datos: Medicine . . . . .	39
C.6. Diccionario de datos: PatientMedicine . . . . .	40
D.1. CP1: Inicio de Sesión . . . . .	52
D.2. CP2: Cierre de Sesión . . . . .	52
D.3. CP3: Acceder a una página sin permisos . . . . .	53

D.4. CP4: Añadir usuario . . . . .	54
D.5. CP5: Eliminar médico con pacientes . . . . .	55
D.6. CP6: Eliminar administrador . . . . .	55
D.7. CP7: Eliminar paciente con vídeos y medicinas . . . . .	56
D.8. CP8: Modificar usuario . . . . .	57
D.9. CP9: Añadir vídeo . . . . .	58
D.10.CP10: Eliminar vídeo . . . . .	59
D.11.CP11: Añadir medicina . . . . .	60
D.12.CP12: Eliminar medicina . . . . .	61
D.13.CP13: Error en la gráfica del paciente . . . . .	61
D.14.CP14: Visualizar gráfica del paciente . . . . .	62



## *Apéndice A*

---

# **Plan de Proyecto Software**

---

## **A.1. Introducción**

El propósito de este documento es establecer una guía de forma clara y organizada para la ejecución de un proyecto. En él se definen los objetivos y la forma en la que se van a alcanzar.

### **Scrum**

Para la planificación del proyecto ha sido empleado el marco de gestión de proyectos de metodología ágil [3]. Scrum es una metodología ágil para la gestión de proyectos complejos en entornos cambiantes. Se basa en tres pilares: eventos, roles y artefactos, y se trabaja en sprints de una duración determinada, generalmente entre una semana y un mes. Los equipos de Scrum se autoorganizan y se comprometen a entregar resultados de alta calidad de manera eficiente y creativa.

### **Roles**

En este marco de gestión de proyectos, se destacan tres roles principales:

- *Scrum Master*: Líder del equipo encargado de eliminar obstáculos y facilitar la auto-organización y coordinación del equipo.
- *Product Owner*: Representa la voz del cliente, lidera el desarrollo del producto y busca el valor para los usuarios.

- Equipo de desarrollo: Responsable de ejecutar las acciones previstas para el éxito del proceso

## Eventos

Los eventos clave en Scrum son:

- *Sprint*: Reunión para planificar el trabajo del próximo sprint.
- Scrum diario: Breve reunión diaria para sincronizar actividades.
- Revisión del *sprint*: Revisión del producto al final del *sprint*.
- Retrospectiva del *sprint* : Análisis de los éxitos y fallos del *sprint*

## Artefactos

Scrum utiliza tres artefactos principales:

- Pila de producto o *product backlog*: Inventario que contiene el trabajo pendiente en el producto.
- Pila del *sprint* o *sprint backlog*: Lista de tareas a realizar durante el *sprint*.
- Incremento: Versión mejorada y funcional del producto al final de cada *sprint*.

## A.2. Planificación temporal

### Planificación mediante *sprints*

Se ha decidido que la planificación del proyecto se haga mediante *sprints* debido al tamaño del equipo (un desarrollador).

#### Sprint 1

##### ■ Objetivos

1. Configuración inicial: creación del repositorio y configuración de cuenta de ZenHub.
2. Memoria: redacción de la introducción y familiarización con las aplicaciones para edición de archivos TeX.

3. Lectura de papers: «Supervised classification of bradykinesia in Parkinson's disease from smartphone videos» [5], «The discerning eye of computer vision: Can it measure Parkinson's finger tap bradykinesia?» [4] y «A computer vision framework for finger-tapping evaluation in Parkinson's disease» [2].
- **Periodo** Este *sprint* se desarrolló entre el 2 de octubre del 2023 y el 16 de octubre del 2023.
  - **Review** En la revisión se resolvieron dudas sobre la documentación y se decidió cambiar de herramienta para la realización de la metodología scrum del proyecto a zube.

## Sprint 2

- **Objetivos**
  1. Documentación de trabajos previos: lectura del trabajo realizado por Catalin para ver la extracción de datos.
  2. Memoria: redacción del apartado de trabajos relacionados.
  3. Curso de flask: realización de los tutoriales del framework de flask para la realización del apartado web.
- **Periodo** Este *sprint* se desarrolló entre el 16 de octubre del 2023 y el 29 de octubre del 2023.
- **Review** Se resolvieron dudas sobre dónde encontrar la información dentro del proyecto de Catalin así como problemas a la hora de compilar los documentos de L<sup>A</sup>T<sub>E</sub>X.

## Sprint 3

- **Objetivos**
  1. Documentación de trabajos previos: lectura de los documentos para la extracción de datos usando el paquete de python TSFresh.
  2. Memoria: corrección de errores en la documentación y añadida biografía.
  3. Curso de flask: continuada la realización de los tutoriales del framework de flask para la realización del apartado web.

4. Repositorio: añadido el archivo gitignore para evitar la subida de archivos temporales.
- **Periodo** Este *sprint* se desarrolló entre el 30 de octubre del 2023 y el 12 de noviembre del 2023.
  - **Review** Problemas a la hora de la realización de los objetivos propuestos agregados en el siguiente *sprint*.

## Sprint 4

- **Objetivos**
  1. Creación del *mockup* de la aplicación.
  2. Documentación de los anexos.
- **Periodo** Este *sprint* se desarrolló entre el 13 de noviembre del 2023 y el 30 de diciembre del 2023.
- **Review** Se revisó el *mockup* y se propuso una modificación del mismo.

## Sprint 5

- **Objetivos**
  1. Finalización del curso de flask.
  2. Modificación del *mockup*.
  3. Creación de diagramas entidad relación y diagramas de casos de uso.
- **Periodo** Este *sprint* se desarrolló entre el 30 de noviembre del 2023 y el 12 de diciembre del 2023.
- **Review** Se revisaron tanto los diagramas como el *mockup* y se propusieron unos cambios en los diagramas.

## Sprint 6

- **Objetivos**
  1. Documentación general añadiendo técnicas y herramientas.
  2. Búsqueda de trabajos parecidos o aplicaciones médicas similares.

3. Modificación de los diagramas de casos de uso y entidad-relación.

- **Periodo** Este sprint se desarrolló entre el 13 de diciembre del 2023 y el 19 de enero de 2024.
- **Review** Se cumplieron todos los objetivos del sprint.

## Sprint 7

- **Objetivos**

1. Documentación general añadiendo técnicas y herramientas.
2. Búsqueda de trabajos parecidos o aplicaciones médicas similares.
3. Modificación del diagramas de casos de uso y entidad-relación.

- **Periodo** Este sprint se desarrolló entre el 20 de enero de 2024 y el 15 de febrero de 2024.
- **Review** Se cumplieron todos los objetivos del sprint. Se detectaron ciertos fallos en la documentación que se añadieron como tarea para el siguiente sprint.

## Sprint 8

- **Objetivos**

1. Comienzo de la página web.
2. Conceptos teóricos de la aplicación.
3. Solución de errores detectados en la anterior revisión.
4. Conexión con la base de datos.

- **Periodo** Este sprint se desarrolló entre el 16 de febrero del 2024 y el 29 de febrero de 2024.
- **Review** Se cumplieron todos los objetivos del sprint.

## Sprint 9

- **Objetivos**

1. Continuación de la redacción de los conceptos teóricos.

2. Creación de la página de registro.

- **Periodo** Este sprint se desarrolló entre el 1 de marzo del 2024 y el 14 de marzo de 2024.
- **Review** No se cumplió el objetivo de la documentación que se añadió al siguiente sprint.

## Sprint 10

- **Objetivos**
  1. Continuación de la redacción de los conceptos teóricos.
  2. Creación de la página de inicio de sesión.
- **Periodo** Este sprint se desarrolló entre el 15 de marzo del 2024 y el 8 de abril de 2024.
- **Review** Se cumplieron todos los objetivos del sprint.

## Sprint 11

- **Objetivos**
  1. Documentación de técnicas y herramientas.
  2. Creación de la página principal de administrador.
  3. Solución de errores relacionados con la página de registro.
  4. Creación de la página de modificación de usuarios.
- **Periodo** Este sprint se desarrolló entre el 9 de abril de 2024 y el 25 de abril de 2024.
- **Review** Se cumplieron todos los objetivos del sprint.

## Sprint 12

- **Objetivos**
  1. Creación de la página principal del paciente.
  2. Añadir la funcionalidad de actualizar el usuario en la base de datos una vez modificado.

3. Iniciar entrenamiento de los modelos para la predicción de características.
- **Periodo** Este *sprint* se desarrolló entre el 26 de abril de 2024 y el 8 de mayo de 2024.
  - **Review** No se pudieron realizar los experimentos debido a problemas de compatibilidad de versión.

### Sprint 13

- **Objetivos**
  1. Creación de la página principal del doctor.
  2. Continuar con el entrenamiento de los modelos.
- **Periodo** Este *sprint* se desarrolló entre el 9 de mayo de 2024 y el 15 de mayo de 2024.
- **Review** Debido a problemas durante el entrenamiento de los modelos, no se pudo completar este objetivo correctamente.

### Sprint 14

- **Objetivos**
  1. Continuar con el entrenamiento de los modelos.
- **Periodo** Este *sprint* se desarrolló entre el 16 de mayo de 2024 y el 22 de mayo de 2024.
- **Review** Se cumplieron todos los objetivos del sprint y se consiguió resolver el problema con el entrenamiento de modelos.

### Sprint 15

- **Objetivos**
  1. Agregar funcionalidad de gestionar medicinas de los pacientes.
  2. Agregar funcionalidad de ver medicinas del paciente.
  3. Modificar la base de datos para añadir medicinas a la aplicación.
  4. Agregar funcionalidad de gestionar vídeos de los pacientes.

5. Agregar funcionalidad de ver vídeos del paciente.
- **Periodo** Este *sprint* se desarrolló entre el 23 de mayo de 2024 y el 31 de mayo de 2024.
  - **Review** Se cumplieron todos los objetivos del sprint.

### Sprint 16

- **Objetivos**
  1. Refactorizar la aplicación para aplicar un Modelo-vista-controlador.
  2. Solucionar fallos de funcionalidades del doctor a la hora de gestionar vídeos y medicinas.
  3. Implementar extracción de características de los vídeos.
  4. Realizar comparación de modelos.
- **Periodo** Este *sprint* se desarrolló entre el 1 de junio de 2024 y el 13 de junio de 2024.
- **Review** Se cumplieron todos los objetivos del sprint.

## A.3. Estudio de viabilidad

En este apartado se estudian los criterios a tener en cuenta a nivel económico y legal para determinar si es factible y si cumple los objetivos propuestos.

### Viabilidad económica

En el siguiente subapartado se expone la viabilidad del proyecto a nivel económico.

#### Costes

Se entiende por coste todo aquel gasto necesario para poder llevar acabo un proyecto desde su inicio hasta su finalización. Para el análisis de los costes del proyecto y el cálculo de la amortización se ha tenido en cuenta la duración total del proyecto (40 semanas o entorno a 9 meses).

**1. Costes de empleados:** en este caso se cuenta con dos empleados que han sido el desarrollador (el alumno) y el *product owner* (tutor académico). Se estima que el tiempo real invertido por parte del desarrollador es de entorno a las 550 horas a lo largo de 9 meses. Teniendo en cuenta que el sueldo promedio de un programador junior en España es de entorno a 23.700€<sup>1</sup> y la jornada laboral completa son aproximadamente 1820 horas, el salario medio bruto por hora es de 13€. Por tanto el salario del desarrollador estará entorno a 550 horas  $\times 13 \frac{\text{€}}{\text{hora}} = 7150\text{€}$  o aproximadamente  $795 \frac{\text{€}}{\text{mes}}$ . A esta cantidad se le ha de añadir los impuestos que se han de pagar como empresa por el empleado<sup>2</sup>

- **Contingencias Comunes:** correspondiente al 23,60 %.
- **Desempleo:** correspondiente al 6,70 %.
- **FOGASA:** fondo de garantía salarial. Correspondiente al 0,20 %.
- **Formación profesional:** correspondiente al 0,60 %.
- **Accidentes de Trabajo y Enfermedades Profesionales:** correspondiente al 1,50 %.

Por tanto, tras realizar los cálculos añadiendo los impuestos correspondientes, la empresa debe pagar  $1.179,52 \frac{\text{€}}{\text{mes}}$  por el desarrollador.

$$\frac{795 \frac{\text{€}}{\text{mes}}}{(1 - (0,236 + 0,067 + 0,002 + 0,006 + 0,015))} = 1.179,52 \frac{\text{€}}{\text{mes}} \quad (\text{A.1})$$

Por otro lado el *product owner*, suponiendo un sueldo bruto anual de 39000€<sup>3</sup>, se calcula que el salario bruto hora es de 21,43€. Teniendo en cuenta que el tutor ha dedicado entorno a 3 horas por semana al proyecto, cobraría 257,16€ brutos al mes. Aplicando la formula al igual que para el desarrollador:

$$\frac{257,16 \frac{\text{€}}{\text{mes}}}{(1 - (0,236 + 0,067 + 0,002 + 0,006 + 0,015))} = 381,54 \frac{\text{€}}{\text{mes}} \quad (\text{A.2})$$

Por lo tanto, y teniendo en cuenta que la duración del proyecto ha sido de 9 meses, el total que ha de pagar la empresa es de 14.049,54€ por los empleados.

<sup>1</sup>Fuente: <https://es.indeed.com/career/programador-junior/salaries>

<sup>2</sup>Fuente: <https://www.seg-social.es/wps/portal/wss/internet/Trabajadores/CotizacionRecaudacionTrabajadores/10721/10957/9932/4315>

<sup>3</sup>Fuente: [https://www.glassdoor.es/Sueldos/product-owner-sueldo-SRCH\\_K00\\_13.htm](https://www.glassdoor.es/Sueldos/product-owner-sueldo-SRCH_K00_13.htm)

$$381,54 \frac{\text{€}}{\text{mes}} \times 9 \text{ meses} + 1179,52 \frac{\text{€}}{\text{mes}} \times 9 \text{ meses} = 14049,54 \text{ €}$$

2. **Hardware:** para el hardware solo se ha hecho uso del equipo del alumno. Este equipo se trata de un PcCom Gold Élite con un Intel Core i5-11400F, 16 GB de RAM y NVIDIA Geforce RTX 3070 con un P.V.P <sup>4</sup> de 1199,18€. Se calcula que el ordenador será amortizado en 4 años por lo que se debe calcular el precio amortizado para la duración del proyecto. En total son 224,84€.

$$\frac{1199,18\text{€} \times 9\text{meses}}{4\text{años} \times 12\text{meses}} = 224,85\text{€}$$

3. **Software:** la mayoría de programas y *software* utilizado durante el desarrollo son gratuitos a excepción de GitHub Copilot cuyo coste de la versión *entreprise* es de 36,35€ al mes. Teniendo en cuenta la duración del proyecto, esto nos da un coste total de 327,15€.

### Costes totales

A continuación se muestra la tabla con los costes totales.

Concepto	Coste(€)
Empleados	14.049,54
Hardware	224,84
Software	327,15
<b>Total</b>	<b>14.601,53</b>

Tabla A.1: Tabla de costes totales

### Ingresos

Dado que este proyecto pretende ser accesible, su uso va a ser completamente gratuito. Debido a esto una de las principales formas en las que se podría obtener rédito económico de esta aplicación sería mediante la implementación de anuncios. Dado que es proyecto pretende ser una herramienta para facilitar el trabajo a pacientes y médicos no se ha considerado oportuno y por tanto no se ha estudiado esta opción. Otra forma en la que se podría

<sup>4</sup>Fuente: <https://www.pccomponentes.com/pccom-gold-elite-intel-core-i5-11400f-16gb-1tb-ssd-rtx-3070>

sacar ingresos sería mediante donaciones o ayudas que permitiesen cubrir los gastos de mantenimiento y desarrollo del proyecto.

## Viabilidad legal

A continuación se muestra una tabla recopilando las licencias de las dependencias utilizadas durante el desarrollo del proyecto.

Dependencia	Versión	Licencia
Flask	3.0.2	BSD License
Flask-Login	0.6.3	MIT License
Flask-SQLAlchemy	3.1.1	BSD License
Flask-WTF	1.2.1	BSD License
Font Awesome Flask	0.1.4	GNU General Public License v3 or later (GPLv3+)
Jinja2	3.1.3	BSD License
MarkupSafe	2.1.5	BSD License
PyYAML	6.0.1	MIT License
Pygments	2.18.0	BSD License
SQLAlchemy	2.0.27	MIT License
WTForms	3.1.2	BSD License
Werkzeug	3.0.1	BSD License
absl-py	2.1.0	Apache Software License
annotated-types	0.6.0	MIT License
attrs	23.2.0	MIT License
blinker	1.7.0	MIT License
certifi	2024.2.2	Mozilla Public License 2.0 (MPL 2.0)
cffi	1.16.0	MIT License
charset-normalizer	3.3.2	MIT License
click	8.1.7	BSD License
cloudpickle	3.0.0	BSD License
colorama	0.4.6	BSD License
contourpy	1.2.1	BSD License
cycler	0.12.1	BSD License
dask	2024.5.0	BSD License
dask-expr	1.1.0	BSD License
distributed	2024.5.0	BSD License
flatbuffers	24.3.25	Apache Software License
fonttools	4.51.0	MIT License

*Continúa en la siguiente página*

Tabla A.2 – *Continuación*

Dependencia	Versión	Licencia
fsspec	2024.3.1	BSD License
greenlet	3.0.3	MIT License
gunicorn	22.0.0	MIT License
idna	3.7	BSD License
importlib_metadata	7.1.0	Apache Software License
itsdangerous	2.1.2	BSD License
jax	0.4.27	Apache-2.0
jaxlib	0.4.27	Apache-2.0
joblib	1.4.2	BSD License
kiwisolver	1.4.5	BSD License
llvmlite	0.42.0	BSD
locket	1.0.0	BSD License
matplotlib	3.8.4	Python Software Foundation License
mediapipe	0.10.13	Apache Software License
ml-dtypes	0.4.0	Apache Software License
msgpack	1.0.8	Apache Software License
numba	0.59.1	BSD License
numpy	1.26.4	BSD License
opencv-contrib-python	4.9.0.80	Apache Software License
opencv-python	4.9.0.80	Apache Software License
opt-einsum	3.3.0	MIT
packaging	24.0	Apache Software License; BSD License
pandas	2.2.2	BSD License
partd	1.4.2	BSD
patsy	0.5.6	BSD License
pillow	10.3.0	Historical Permission Notice and Disclaimer (HPND)
pip-licenses	4.4.0	MIT License
protobuf	4.25.3	3-Clause BSD License
psutil	5.9.8	BSD License
pure-eval	0.2.2	MIT License
pyarrow	16.0.0	Apache Software License
pycparser	2.22	BSD License
pydantic	2.7.1	MIT License
pydantic-settings	2.2.1	MIT License
pydantic_core	2.18.2	MIT License
pyparsing	3.1.2	MIT License

Continúa en la siguiente página

Tabla A.2 – *Continuación*

Dependencia	Versión	Licencia
python-dateutil	2.9.0.post0	Apache Software License; BSD License
python-dotenv	1.0.1	BSD License
pytz	2024.1	MIT License
pywin32	306	Python Software Foundation License
pyzmq	26.0.3	BSD License
requests	2.31.0	Apache Software License
scikit-learn	1.4.2	BSD License
scipy	1.13.0	BSD License
six	1.16.0	MIT License
sortedcontainers	2.4.0	Apache Software License
sounddevice	0.4.6	MIT License
statsmodels	0.14.2	BSD License
stumpy	1.12.0	BSD License
tblib	3.0.0	BSD License
threadpoolctl	3.5.0	BSD License
toolz	0.12.1	BSD License
tornado	6.4	Apache Software License
tqdm	4.66.4	MIT License; Mozilla Public License 2.0 (MPL 2.0)
traitlets	5.14.3	BSD License
tsfresh	0.20.2	MIT
typing_extensions	4.9.0	Python Software Foundation License
tzdata	2024.1	Apache Software License
urllib3	2.2.1	MIT License
zict	3.0.0	BSD License
zipp	3.18.1	MIT License

Tabla A.2: Dependencias del proyecto y licencias bajo las cuales están registradas. Extraídas mediante la librería pip-licenses<sup>5</sup>.

Todas las librerías, dependencias y herramientas utilizadas licencias encontradas en programas de código abierto. Además, el Trabajo de Fin de Grado PADDEL [1] que ha sido utilizado para el desarrollo de este proyecto tiene una licencia MIT la cual permite a cualquier persona que obtenga una copia del proyecto modificarlo, copiarlo y redistribuirlo libremente. Es por ello que se ha optado por utilizar la licencia MIT para este proyecto también.

---

<sup>5</sup><https://pypi.org/project/pip-licenses/>



## *Apéndice B*

---

# **Especificación de Requisitos**

---

## **B.1. Introducción**

Previo a la creación de un programa se debe especificar las características concretas que ha de tener. En este caso ha definido estas características utilizando UML como estándar.

## **B.2. Objetivos generales**

El objetivo principal de esta aplicación es mostrar, tanto a los pacientes como a sus respectivos doctores, su evolución mediante el uso de inteligencia artificial.

Además se busca que el usuario doctor sea capaz de realizar comparaciones de múltiples pacientes de manera simultánea.

## **B.3. Catálogo de requisitos**

A continuación se presentan los requisitos funcionales y no funcionales de la aplicación.

### **Requisitos funcionales**

**RF1:** El sistema debe permitir diferenciar entre tres tipos de roles: administrador, paciente y doctor.

**RF2:** El doctor ha de ser capaz de agregar vídeos a sus pacientes.

- RF3:** El doctor ha de ser capaz de eliminar vídeos a sus pacientes.
- RF4:** El doctor ha de ser capaz de visualizar el listado de vídeos de sus pacientes.
- RF5:** El doctor ha de ser capaces de clasificar los vídeos de sus pacientes.
- RF6:** Los usuarios han de ser capaces de iniciar sesión con sus correspondientes credenciales.
- RF7:** Los usuarios deben ser capaces de poder cerrar sesión.
- RF8:** El administrador ha de ser capaz de añadir usuarios.
- RF9:** El administrador ha de ser capaz de eliminar usuarios.
- RF10:** El administrador ha de ser capaz de modificar usuarios.
- RF11:** El administrador ha de ser capaz de visualizar a los usuarios.
- RF12:** El administrador ha de ser capaz de asignar doctor a los pacientes.
- RF13:** El administrador ha de ser capaz de desasignar doctor a los pacientes.
- RF14:** El paciente y el doctor han de ser capaces de ver la evolución del paciente.
- RF15:** El doctor ha de ser capaz de añadir medicinas a sus pacientes.
- RF16:** El doctor ha de ser capaz de eliminar medicinas a sus pacientes.
- RF17:** El doctor ha de ser capaz de visualizar el listado de medicinas de sus pacientes.

## **Requisitos no funcionales**

- RNF1:** Todos los datos sensibles, incluyendo credenciales de usuario deben ser encriptados utilizando algoritmos seguros como SHA-256.
- RNF2:** Las operaciones básicas, como el inicio de sesión, la carga y visualización de vídeos, y la gestión de usuarios, deben completarse en menos de 2 segundos bajo condiciones normales de uso.
- RNF3:** La interfaz de usuario ha de ser diseñada siguiendo principios de diseño centrado en el usuario para facilitar la usabilidad de esta.

**RNF4:** La aplicación debe ser totalmente funcional en los navegadores más utilizados (Chrome, Firefox, Safari y Edge) en sus versiones más recientes. Debe tener un diseño responsivo que asegure una experiencia de usuario óptima en dispositivos de escritorio, tablets y móviles.

## B.4. Especificación de requisitos

En este apartado se especifican los casos de uso correspondientes a los requisitos funcionales previamente expuestos así como los actores de la aplicación.

### Actores

En esta aplicación se pueden distinguir tres actores:

- Administrador: Usuario con sesión de administrador iniciada que tiene acceso a funcionalidades de añadir, modificar y eliminar usuarios.
- Doctor: Usuario con sesión de doctor que tiene acceso a funcionalidades de gestión de pacientes y medicinas así como de visualización de evolución del paciente.
- Paciente: Usuario con sesión de paciente acceso a la funcionalidad de visualizar su evolución.

Además en este apartado se muestra el diagrama de casos de uso (ver figura B.1) y las siguientes tablas:

- La tabla B.1 contiene el caso de uso Iniciar sesión.
- La tabla B.2 contiene el caso de uso Cerrar sesión.
- La tabla B.3 contiene el caso de uso Visualizar evolución.
- La tabla B.4 contiene el caso de uso Dar de alta a usuarios.
- La tabla B.5 contiene el caso de uso Dar de baja a usuarios.
- La tabla B.6 contiene el caso de uso Modificar usuarios.
- La tabla B.7 contiene el caso de uso Visualizar listado de usuarios.
- La tabla B.8 contiene el caso de uso Añadir vídeos.

- La tabla **B.9** contiene el caso de uso Eliminar vídeos.
- La tabla **B.10** contiene el caso de uso Clasificar videos.
- La tabla **B.11** contiene el caso de uso Asignar paciente.
- La tabla **B.12** contiene el caso de uso Desasignar paciente.
- La tabla **B.13** contiene el caso de uso Añadir medicina.
- La tabla **B.14** contiene el caso de uso Eliminar medicina.
- La tabla **B.15** contiene el caso de uso Visualizar listado de medicinas de un paciente.

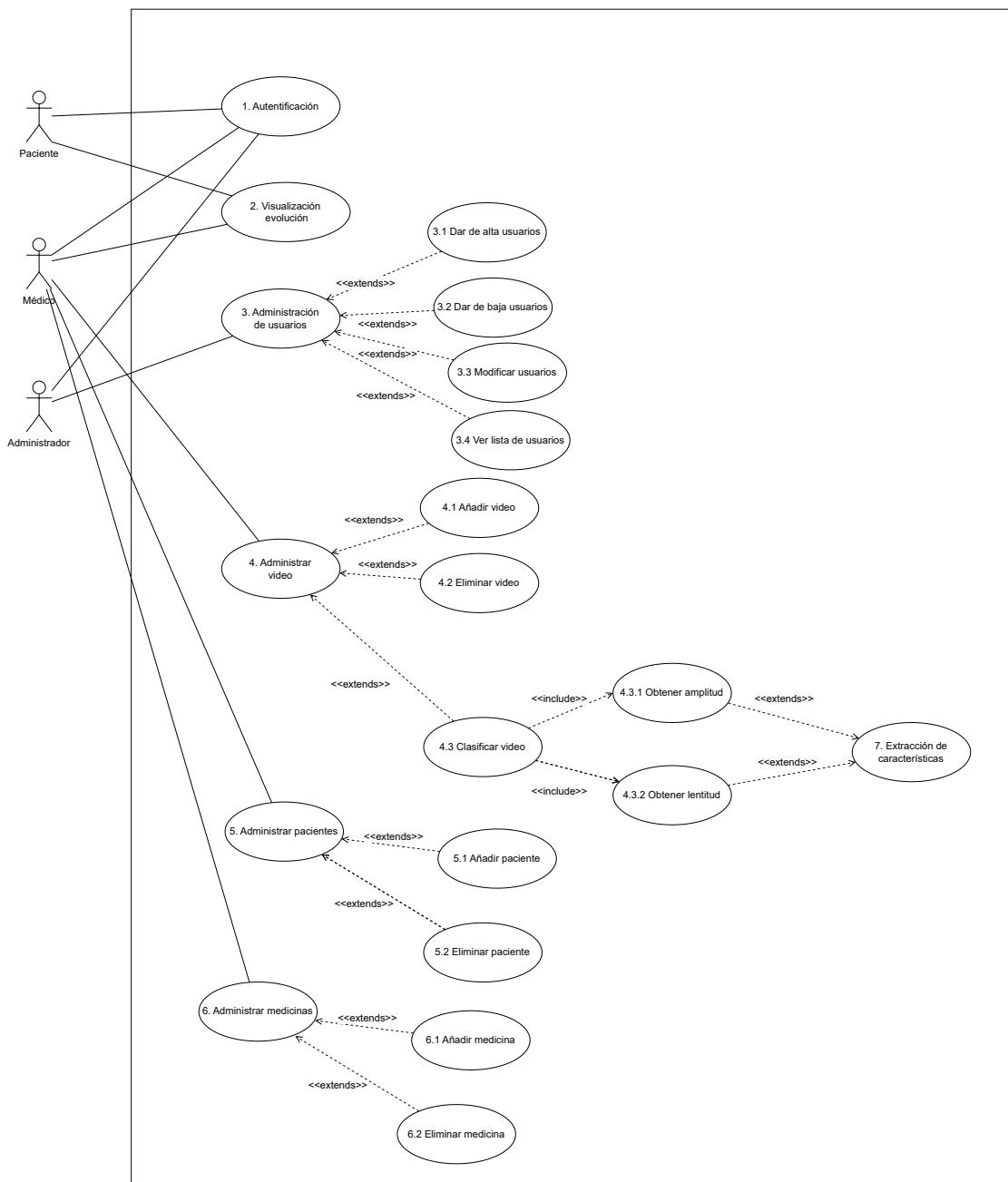


Figura B.1: Diagrama de casos de uso

---

<b>CU-1</b>	<b>Iniciar sesión</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jorge Martínez Martín
<b>Requisitos asociados</b>	RF-1,RF-6
<b>Descripción</b>	Los usuarios deben ser capaces de iniciar sesión
<b>Precondición</b>	<ul style="list-style-type: none"> <li>■ El usuario debe estar dado de alta en la base de datos.</li> <li>■ El usuario debe estar en la página de inicio de sesión.</li> <li>■ El usuario no debe tener una sesión iniciada.</li> </ul>
<b>Acciones</b>	<ul style="list-style-type: none"> <li>■ El usuario introduce su nombre de usuario.</li> <li>■ El usuario introduce su contraseña.</li> <li>■ El usuario selecciona su rol.</li> <li>■ El usuario hace clic en el botón de iniciar sesión.</li> </ul>
<b>Postcondición</b>	El usuario será redirijo a su página de perfil correspondiente
<b>Excepciones</b>	<ul style="list-style-type: none"> <li>■ Las credenciales no son correctas.</li> <li>■ El usuario no está dado de alta.</li> </ul>
<b>Importancia</b>	Alta

---

Tabla B.1: CU-1 Iniciar sesión.

CU-2	Cerrar sesión
<b>Versión</b>	1.0
<b>Autor</b>	Jorge Martínez Martín
<b>Requisitos asociados</b>	RF-1,RF-7
<b>Descripción</b>	Los usuarios deben ser capaces de cerrar sesión
<b>Precondición</b>	<ul style="list-style-type: none"><li>■ El usuario debe tener la sesión iniciada.</li></ul>
<b>Acciones</b>	<ul style="list-style-type: none"><li>■ El usuario hace clic en el botón de finalizar sesión.</li></ul>
<b>Postcondición</b>	El usuario será redirijo a la página de inicio
<b>Excepciones</b>	
<b>Importancia</b>	Alta

Tabla B.2: CU-2 Cerrar sesión.

<b>CU-3</b>	<b>Visualizar evolución</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jorge Martínez Martín
<b>Requisitos asociados</b>	RF-14
<b>Descripción</b>	El paciente y el doctor han de ser capaces de visualizar la evolución del paciente
<b>Precondición</b>	<ul style="list-style-type: none"> <li>■ El usuario debe de tener una sesión de doctor o paciente iniciada.</li> <li>■ El paciente debe estar dado de alta en la base de datos.</li> <li>■ El doctor debe ser el que esta asignado al paciente.</li> </ul>
<b>Acciones</b>	<ul style="list-style-type: none"> <li>■ El usuario accede a su perfil.</li> <li>■ Se visualiza la gráfica de la evolución del paciente.</li> </ul>
<b>Postcondición</b>	Se muestra al usuario su gráfico correspondiente
<b>Excepciones</b>	<ul style="list-style-type: none"> <li>■ El doctor no corresponde con el del paciente.</li> </ul>
<b>Importancia</b>	Alta

Tabla B.3: CU-3 Visualizar evolución.

<b>CU-4.1</b>	<b>Dar de alta a usuarios</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jorge Martínez Martín
<b>Requisitos asociados</b>	RF-8
<b>Descripción</b>	El administrador debe poder dar de alta a nuevos usuarios.
<b>Precondición</b>	<ul style="list-style-type: none"> <li>■ El usuario debe de tener una sesión de administrador iniciada.</li> <li>■ El usuario que se desea añadir no debe estar dado de alta.</li> </ul>
<b>Acciones</b>	<ul style="list-style-type: none"> <li>■ El administrador pulsa el botón de «añadir usuario».</li> <li>■ El administrador introduce los datos del nuevo usuario.</li> <li>■ El administrador confirma los datos y pulsa el botón «añadir».</li> </ul>
<b>Postcondición</b>	Se muestra el nuevo usuario en la lista de usuarios
<b>Excepciones</b>	<ul style="list-style-type: none"> <li>■ El usuario ya existe.</li> <li>■ Los datos introducidos son erróneos o incompletos.</li> </ul>
<b>Importancia</b>	Alta

Tabla B.4: CU-4.1 Dar de alta a usuarios.

CU-4.2	Dar de baja a usuarios
<b>Versión</b>	1.0
<b>Autor</b>	Jorge Martínez Martín
<b>Requisitos asociados</b>	RF-9
<b>Descripción</b>	El administrador debe poder dar de baja a usuarios.
<b>Precondición</b>	<ul style="list-style-type: none"> <li>■ El usuario debe de tener una sesión de administrador iniciada.</li> <li>■ El usuario que se desea eliminar debe estar dado de alta.</li> </ul>
<b>Acciones</b>	<ul style="list-style-type: none"> <li>■ El administrador pulsa el botón de «eliminar usuario».</li> <li>■ El administrador confirma el usuario y pulsa el botón «eliminar».</li> </ul>
<b>Postcondición</b>	Ya no se muestra al usuario en la lista de usuarios
<b>Excepciones</b>	<ul style="list-style-type: none"> <li>■ El usuario no existe.</li> <li>■ El usuario es un doctor y tiene pacientes asignados.</li> <li>■ El usuario que se desea eliminar es el mismo que el usuario de la sesión.</li> </ul>
<b>Importancia</b>	Alta

Tabla B.5: CU-4.2 Dar de baja a usuarios.

CU-4.3	Modificar usuarios
<b>Versión</b>	1.0
<b>Autor</b>	Jorge Martínez Martín
<b>Requisitos asociados</b>	RF-10
<b>Descripción</b>	El administrador debe poder modificar usuarios.
<b>Precondición</b>	<ul style="list-style-type: none"><li>■ El usuario debe de tener una sesión de administrador iniciada.</li><li>■ El usuario debe estar dado de alta.</li></ul>
<b>Acciones</b>	<ul style="list-style-type: none"><li>■ El administrador pulsa el botón de «modificar usuario».</li><li>■ El administrador realiza los cambios del usuario y pulsa el botón «confirmar».</li></ul>
<b>Postcondición</b>	Se muestra la modificación del usuario.
<b>Excepciones</b>	<ul style="list-style-type: none"><li>■ El usuario modificado coincide con otro que ya existe.</li></ul>
<b>Importancia</b>	Alta

Tabla B.6: CU-4.3 Modificar usuarios.

<b>CU-4.4</b>	<b>Visualizar listado de usuarios</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jorge Martínez Martín
<b>Requisitos asociados</b>	RF-11
<b>Descripción</b>	El administrador debe poder visualizar la lista de usuarios.
<b>Precondición</b>	<ul style="list-style-type: none"> <li>■ El usuario debe de tener una sesión de administrador iniciada.</li> <li>■ Debe existir al menos 1 usuario.</li> </ul>
<b>Acciones</b>	<ul style="list-style-type: none"> <li>■ El administrador visualiza la lista de usuarios.</li> </ul>
<b>Postcondición</b>	Se muestra la lista de usuarios.
<b>Excepciones</b>	
<b>Importancia</b>	Media

Tabla B.7: CU-4.4 Visualizar listado de usuarios.

<b>CU-5.1</b>	<b>Añadir vídeos</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jorge Martínez Martín
<b>Requisitos asociados</b>	RF-2
<b>Descripción</b>	Los doctores han de ser capaces de añadir nuevos vídeos a sus pacientes.
<b>Precondición</b>	<ul style="list-style-type: none"> <li>■ El usuario debe de tener una sesión de doctor iniciada.</li> <li>■ El doctor del paciente ha de ser el mismo que el doctor de la sesión.</li> <li>■ Debe existir el paciente al que se le quiere añadir el vídeo.</li> </ul>
<b>Acciones</b>	<ul style="list-style-type: none"> <li>■ El usuario pulsa el botón «añadir vídeo».</li> <li>■ Se rellenan los datos correspondientes del vídeo como la mano.</li> <li>■ Se añade el fichero con el vídeo que se quiere subir.</li> <li>■ Se pulsa el botón «añadir».</li> </ul>
<b>Postcondición</b>	El nuevo vídeo debe estar en la lista de vídeos del paciente correspondiente.
<b>Excepciones</b>	<ul style="list-style-type: none"> <li>■ No se han llenado los datos correctamente.</li> <li>■ El fichero subido no corresponde con formato solicitado.</li> </ul>
<b>Importancia</b>	Alta

Tabla B.8: CU-5.1 Añadir vídeos.

<b>CU-5.2</b>	<b>Eliminar vídeos</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jorge Martínez Martín
<b>Requisitos asociados</b>	RF-3
<b>Descripción</b>	Los doctores han de ser capaces de eliminar vídeos de sus pacientes.
<b>Precondición</b>	<ul style="list-style-type: none"> <li>■ El usuario debe de tener una sesión de doctor iniciada.</li> <li>■ El doctor del paciente ha de ser el mismo que el doctor de la sesión.</li> <li>■ Debe existir el paciente al que se le quiere eliminar el vídeo.</li> <li>■ El paciente debe tener al menos un vídeo.</li> </ul>
<b>Acciones</b>	<ul style="list-style-type: none"> <li>■ El usuario pulsa el botón «eliminar vídeo» al lado del vídeo que desea eliminar.</li> <li>■ El usuario confirma que se desea eliminar el vídeo.</li> </ul>
<b>Postcondición</b>	El vídeo ya no esta en la lista de vídeos del paciente correspondiente.
<b>Excepciones</b>	
<b>Importancia</b>	Media

Tabla B.9: CU-5.2 Eliminar vídeos.

<b>CU-5.3</b>	<b>Clasificar vídeos</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jorge Martínez Martín
<b>Requisitos asociados</b>	RF-5
<b>Descripción</b>	Los doctores han de ser capaces de clasificar los vídeos del paciente.
<b>Precondición</b>	<ul style="list-style-type: none"> <li>■ El usuario debe de tener una sesión de doctor iniciada.</li> <li>■ El doctor del paciente ha de ser el mismo que el doctor de la sesión.</li> <li>■ Debe existir el paciente al que se le quiere clasificar el vídeo.</li> </ul>
<b>Acciones</b>	<ul style="list-style-type: none"> <li>■ El doctor pulsa el botón «añadir vídeo».</li> <li>■ Se clasificará el vídeo correspondiente.</li> </ul>
<b>Postcondición</b>	Aparecerá la clasificación del vídeo.
<b>Excepciones</b>	
<b>Importancia</b>	Alta

Tabla B.10: CU-5.3 Clasificar vídeos.

<b>CU-6.1</b>	<b>Asignar paciente</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jorge Martínez Martín
<b>Requisitos asociados</b>	RF-12
<b>Descripción</b>	El administrador ha de ser capaz de asignar pacientes a la lista de pacientes de un doctor.
<b>Precondición</b>	<ul style="list-style-type: none"> <li>■ El usuario debe de tener una sesión de administrador iniciada.</li> <li>■ Debe existir al menos un doctor.</li> </ul>
<b>Acciones</b>	<ul style="list-style-type: none"> <li>■ El administrador registra un nuevo paciente.</li> <li>■ El administrador selecciona al doctor correspondiente.</li> <li>■ El administrador registra al paciente.</li> </ul>
<b>Postcondición</b>	Aparecerá el paciente en la lista del doctor.
<b>Excepciones</b>	<ul style="list-style-type: none"> <li>■ El paciente no existe.</li> </ul>
<b>Importancia</b>	Alta

Tabla B.11: CU-6.1 Asignar paciente.

<b>CU-6.2</b>	<b>Desasignar paciente</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jorge Martínez Martín
<b>Requisitos asociados</b>	RF-13
<b>Descripción</b>	El administrador ha de ser capaz de desasignar pacientes de la lista de un doctor.
<b>Precondición</b>	<ul style="list-style-type: none"> <li>■ Debe existir el paciente que se desea eliminar.</li> <li>■ El paciente debe estar incluido en la lista de pacientes del doctor.</li> </ul>
<b>Acciones</b>	<ul style="list-style-type: none"> <li>■ El administrador pulsa el botón «eliminar paciente».</li> <li>■ Se selecciona el paciente que se desea eliminar.</li> <li>■ Se confirma que se desea eliminar a ese paciente.</li> </ul>
<b>Postcondición</b>	Ya no aparecerá el paciente en la lista del doctor.
<b>Excepciones</b>	<ul style="list-style-type: none"> <li>■ El paciente no se encuentra en la lista del doctor.</li> </ul>
<b>Importancia</b>	Media

Tabla B.12: CU-6.2 Desasignar paciente.

<b>CU-7.1</b>	<b>Añadir medicina</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jorge Martínez Martín
<b>Requisitos asociados</b>	RF-15
<b>Descripción</b>	El administrador debe poder dar de alta a nuevos usuarios.
<b>Precondición</b>	<ul style="list-style-type: none"> <li>■ El usuario debe de tener una sesión de doctor iniciada.</li> <li>■ El doctor del paciente ha de ser el mismo que el doctor de la sesión.</li> <li>■ Debe existir el paciente al que se le quiere añadir la medicina.</li> </ul>
<b>Acciones</b>	<ul style="list-style-type: none"> <li>■ El doctor pulsa el botón de «añadir medicina».</li> <li>■ El doctor introduce los datos de la nueva medicina.</li> <li>■ El doctor confirma los datos y pulsa el botón «añadir».</li> </ul>
<b>Postcondición</b>	Se muestra la nueva medicina en la lista de medicinas del paciente.
<b>Excepciones</b>	<ul style="list-style-type: none"> <li>■ Los datos introducidos son erróneos o incompletos.</li> </ul>
<b>Importancia</b>	Media

Tabla B.13: CU-7.1 Añadir medicina.

<b>CU-7.2</b>	<b>Eliminar medicina</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jorge Martínez Martín
<b>Requisitos asociados</b>	RF-9
<b>Descripción</b>	El administrador debe poder dar de baja a usuarios.
<b>Precondición</b>	<ul style="list-style-type: none"> <li>■ El usuario debe de tener una sesión de doctor iniciada.</li> <li>■ El doctor del paciente ha de ser el mismo que el doctor de la sesión.</li> <li>■ Debe existir el paciente al que se le quiere añadir la medicina.</li> <li>■ El paciente ha de tener al menos una medicina.</li> </ul>
<b>Acciones</b>	<ul style="list-style-type: none"> <li>■ El doctor pulsa el botón de «eliminar usuario».</li> <li>■ El doctor confirma el usuario y pulsa el botón «eliminar».</li> </ul>
<b>Postcondición</b>	Ya no se muestra la medicina en la lista de medicinas
<b>Excepciones</b>	
<b>Importancia</b>	Media

Tabla B.14: CU-7.2 Eliminar medicina.

<b>CU-7.3</b>	<b>Visualizar listado de medicinas de un paciente</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jorge Martínez Martín
<b>Requisitos asociados</b>	RF-11
<b>Descripción</b>	El administrador debe poder visualizar la lista de usuarios.
<b>Precondición</b>	<ul style="list-style-type: none"> <li>■ El usuario debe de tener una sesión de doctor iniciada.</li> <li>■ El doctor del paciente ha de ser el mismo que el doctor de la sesión.</li> <li>■ Debe existir el paciente al que se le quiere añadir la medicina.</li> </ul>
<b>Acciones</b>	<ul style="list-style-type: none"> <li>■ El doctor visualiza la lista de medicinas del paciente seleccionado.</li> </ul>
<b>Postcondición</b>	Se muestra la lista de medicinas del paciente.
<b>Excepciones</b>	
<b>Importancia</b>	Media

Tabla B.15: CU-7.3 Visualizar listado de medicinas de un paciente.

## *Apéndice C*

---

# Especificación de diseño

---

## C.1. Introducción

En el siguiente apartado se describen los detalles de la estructura del *software*: el comportamiento y la interacción entre los distintos componentes.

## C.2. Diseño de datos

Para implementar la persistencia de los datos se ha hecho uso de una base de datos SQLite debido a que no requiere instalación separada y es fácil de integrar con el framework de Flask. Este diseño se ha dividido en varias fases: diseño del diagrama entidad-relación, diseño del diagrama relacional y diseño del diccionario de datos.

### Modelo entidad-relación

Para representar gráficamente las relaciones entre los datos, se ha creado un diagrama E-R (ver figura C.1). En el se puede observar una ISA exclusiva total. Esto se debe a que, tras consultarla con el profesor Jesús Maudes, se llegó a la conclusión de que crear una ISA exclusiva total era una solución válida debido a que se quiere evitar que un mismo usuario tenga acceso a varias funcionalidades. En caso de que esto se requiriese (si un doctor necesita permisos de administración) se crearía otra cuenta de forma más controlada.

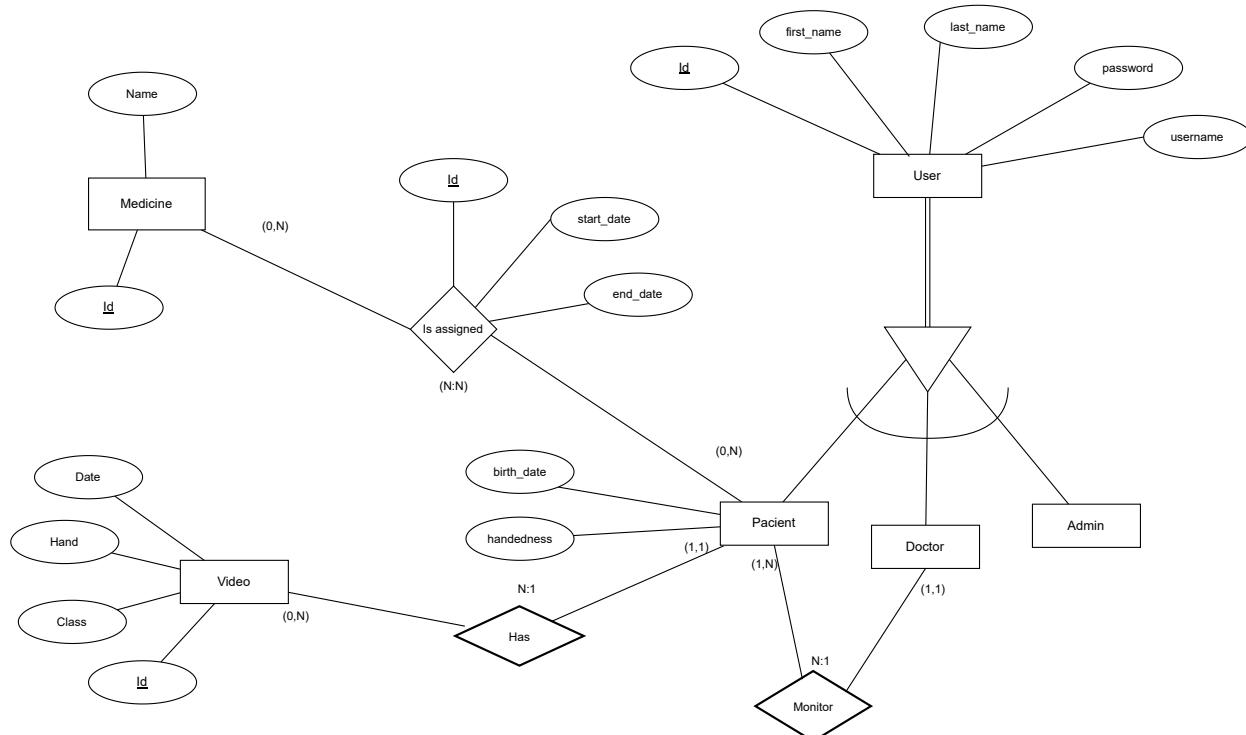


Figura C.1: Diagrama entidad-relación

## Modelo relacional

Partiendo del modelo entidad-relación (ver figura C.1) se ha creado el diagrama relacional (ver figura C.2). A continuación se mencionan algunas de las decisiones más importantes que se han decidido tomar a la hora de crear este diagrama. Con respecto a la ISA se ha decidido dividir en tres tablas distintas ya que en un principio un usuario solo va a poder tener uno de los tres roles asignados y en caso de que se necesitase que un doctor tuviera el rol de administrador, que es el un caso excepcional, se le otorgaría otra cuenta distinta con los permisos de administración. De esta manera podremos controlar los accesos de cada usuario de manera más limpia y exclusiva. El rol de administrador es un rol que pese a que no se relacione con ninguna otra entidad, se ha visto que es necesario para poder permitir el acceso a los controles de administrador.

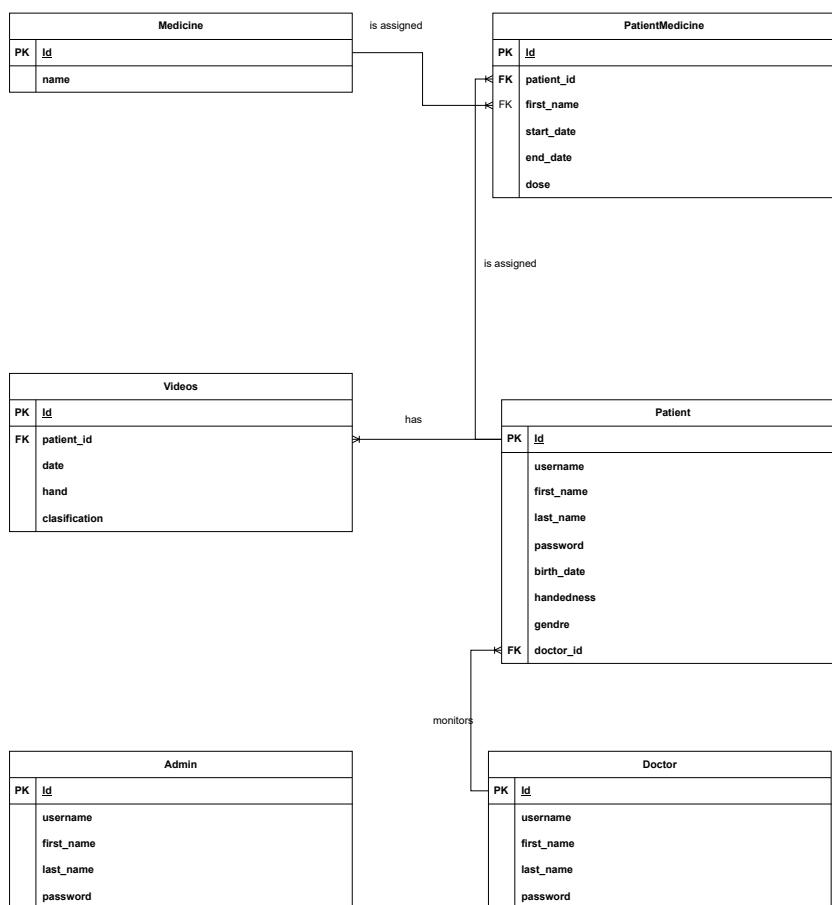


Figura C.2: Diagrama relacional de la aplicación

Nombre	Tipo	Columna	Descripción
id	INTEGER	PK	Identificador único del administrador.
username	VARCHAR(100)	UNIQUE NOT NULL	Nombre de usuario del administrador. Debe ser único.
password	VARCHAR(100)	NOT NULL	Contraseña del administrador, cifrada usando sha-256.
first_name	VARCHAR(100)	NOT NULL	Primer nombre del administrador.
last_name	VARCHAR(100)	NOT NULL	Apellidos del administrador.

Tabla C.1: Diccionario de datos. Tabla correspondiente a la clase **Admin**.

Nombre	Tipo	Columna	Descripción
id	INTEGER	PK	Identificador único del doctor.
username	VARCHAR(100)	UNIQUE NOT NULL	Nombre de usuario del doctor. Debe ser único.
password	VARCHAR(100)	NOT NULL	Contraseña del doctor, cifrada usando sha-256.
first_name	VARCHAR(100)	NOT NULL	Primer nombre del doctor.
last_name	VARCHAR(100)	NOT NULL	Apellidos del doctor.

Tabla C.2: Diccionario de datos. Tabla correspondiente a la clase **Doctor**.

## Diccionario de datos

Se adjunta a continuación, para cada tabla, el diccionario de datos correspondiente.

- **Administradores:** la tabla C.1 contiene la información correspondiente a los administradores de la aplicación.
- **Doctores:** la tabla C.2 contiene la información correspondiente a los doctores.
- **Pacientes:** la tabla C.3 contiene la información correspondiente a los pacientes.
- **Vídeos:** la tabla C.4 contiene la información correspondiente a los vídeos de los pacientes pacientes.
- **Medicinas:** la tabla C.5 contiene la información de las medicinas.
- **PacienteMedicinas:** la tabla C.6 contiene la información de las medicinas administradas a cada paciente incluyendo la fecha de inicio del tratamiento así como la fecha de finalización.

Nombre	Tipo	Columna	Descripción
id	INTEGER	PK	Identificador único del paciente.
username	VARCHAR(100)	UNIQUE NOT NULL	Nombre de usuario del paciente. Debe ser único.
password	VARCHAR(100)	NOT NULL	Contraseña del paciente, cifrada usando sha-256.
first_name	VARCHAR(100)	NOT NULL	Primer nombre del paciente.
last_name	VARCHAR(100)	NOT NULL	Apellidos del paciente.
birth_date	DATE	NOT NULL	Fecha de nacimiento del paciente.
handedness	ENUM('right', 'left')	NOT NULL	Mano dominante del paciente (derecha o izquierda).
gender	ENUM('M', 'F')	NOT NULL	Género del paciente (masculino o femenino).
doctor_id	INTEGER	FK(doctor.id)	Clave foránea a la tabla Doctor. Indica el doctor a cargo del paciente.

Tabla C.3: Diccionario de datos. Tabla correspondiente a la clase **Patient**.

Nombre	Tipo	Columna	Descripción
id	INTEGER	PK	Identificador único del video.
patient_id	INTEGER	FK (patient.id)	Clave foránea a la tabla Paciente. Indica el paciente asociado al video.
hand	ENUM('left', 'right')	NOT NULL	Mano usada en el vídeo (izquierda o derecha).
date	DATE	NOT NULL	Fecha de grabación del video.
video_data	LargeBinary	NOT NULL	Datos binarios del video.
amplitude	VARCHAR(100)	NOT NULL	Amplitud clasificada para el video.
slowness	VARCHAR(100)	NOT NULL	Lentitud clasificada para el video.

Tabla C.4: Diccionario de datos. Tabla correspondiente a la clase **Video**.

Nombre	Tipo	Columna	Descripción
id	INTEGER	PK	Identificador único del medicamento.
name	VARCHAR(100)	NOT NULL	Nombre del medicamento.

Tabla C.5: Diccionario de datos. Tabla correspondiente a la clase **Medicine**.

Nombre	Tipo	Columna	Descripción
<code>id</code>	INTEGER	PK	Identificador único de la asociación entre paciente y medicamento.
<code>patient_ - id</code>	INTEGER	FK ( <code>patient.id</code> )	Clave foránea a la tabla Paciente. Indica el paciente que toma el medicamento.
<code>medicine_ - id</code>	INTEGER	FK ( <code>medicine.id</code> )	Clave foránea a la tabla Medicamento. Indica el medicamento que toma el paciente.
<code>dosage</code>	VARCHAR(100)	NOT NULL	Dosis del medicamento para el paciente.
<code>start_ - date</code>	DATE	NOT NULL	Fecha de inicio del tratamiento con el medicamento.
<code>end_ date</code>	DATE	NULL	Fecha de fin del tratamiento con el medicamento (opcional).

Tabla C.6: Diccionario de datos. Tabla correspondiente a la clase `PatientMedicine`.

### C.3. Diseño procedimental

El diseño procedimental en informática es una metodología de desarrollo de software que se centra en la creación de procedimientos o funciones para llevar a cabo tareas específicas dentro de un programa. Durante el desarrollo del proyecto se ha hecho uso de la herramienta Draw.io<sup>1</sup> para representar los procedimientos. Dentro del sistema podemos distinguir los siguientes componentes:

- Doctor: representado por el monigote, representa a un usuario que haya iniciado sesión como doctor.
- Navegador: se refiere al apartado de *frontend* de la aplicación.
- Servidor web: se refiere a la parte de *backend*.
- Paddel: se refiere a la librería de Paddel utilizada durante el proyecto.
- Base de datos: se refiere a la base de datos de la aplicación.

Debido a que los diagramas secuenciales de la aplicación son muy básicos exceptuando el de la clasificación de los vídeos, se ha tomado la decisión de omitirlos.

---

<sup>1</sup>Fuente: [draw.io](https://draw.io)

## Clasificación de los videos

Para esta funcionalidad, el doctor ha de tener al menos un paciente para poder realizar la acción. El doctor solicita la vista de añadir vídeos al usuario. Tras recibirla, el doctor envía la información del vídeo junto con el archivo del mismo a la aplicación. La aplicación lo envía a la librería Paddel para extraer las características y estas características se envían a los modelos entrenados para obtener las clasificaciones. Una vez obtenidas el vídeo es guardado en la base de datos y posteriormente, se envía al usuario de vuelta a la página de gestión de vídeos actualizada del paciente al que se le ha añadido. La figura C.3 muestra el diagrama de secuencia del proceso.

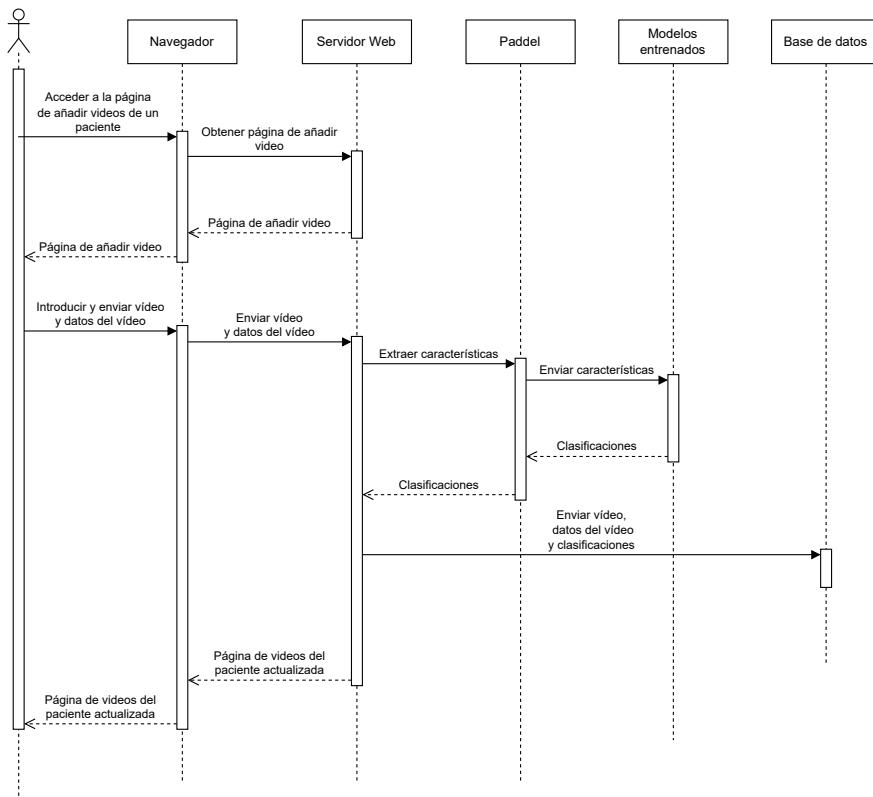


Figura C.3: Diagrama de secuencia de clasificación de los videos

## C.4. Diseño arquitectónico

Para ilustrar la arquitectura de la aplicación se emplea el diagrama de despliegue de la figura C.4. Este tipo de diagrama UML muestra cómo están organizados físicamente los componentes del software en el sistema.

La aplicación no se ha desplegado debido al gran tamaño de los archivos, incluidos los vídeos y los datos de los sensores de los pacientes, por lo que la sección del Web Server no es precisa, ya que la lógica de la aplicación se ejecuta localmente.

El Data Base Server se encarga de gestionar la capa de persistencia del proyecto. Para ello, se ha utilizado SQLite como herramienta de gestión de la base de datos.

El usuario o cliente actúa como el controlador de la interfaz de usuario, interactuando con las páginas HTML a través de su navegador.

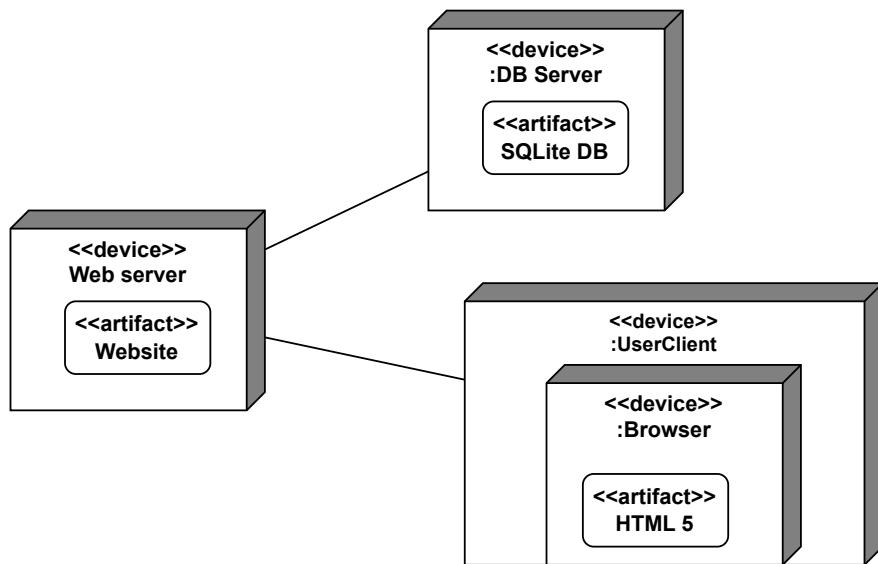


Figura C.4: Diagrama de despliegue



## *Apéndice D*

---

# **Documentación técnica de programación**

---

## **D.1. Introducción**

Esta sección contiene la documentación necesaria para describir el funcionamiento del *software*. Esta documentación es esencial para desarrolladores que pretendan utilizar, mantener o modificar el proyecto.

Se exponen en ella la estructura de directorios, el manual del desarrollador, compilación, instalación y ejecución del proyecto y pruebas del sistema.

## **D.2. Estructura de directorios**

En el repositorio de GitHub <sup>1</sup> esta dividido en los siguientes directorios:

---

<sup>1</sup><https://github.com/JornezMM/PDAIvolution>

/web/  
En este directorio se encuentran todos los subdirectorios y archivos relativos al código fuente de la aplicación.

- |   **controllers/**  
    |    Contiene los controladores de la aplicación.
- |   **instance/**  
    |    Contiene la base de datos de gestionada mediante SQLite.
- |   **models/**  
    |    Contiene todos los modelos de la base de datos de la aplicación.
- |   **app/**  
    |     En el se pueden encontrar el código fuente de la aplicación junto con las *templates* de la aplicación.
- |   **static/**  
    |     Contiene los archivos estáticos de la aplicación entre los que se encuentran hojas de estilos (CSS), imágenes y archivos .js.
- |   **templates/**  
    |     Contiene todos los archivos HTML relativos a la aplicación.
- |   **paddel/**  
    |     Contiene los diferentes archivos de Python que componen la librería PADDEL.
- |   **hyper\_parameters/**  
    |     Contiene los archivos de Python relacionados con la fase de optimización de hiperparámetros.
- |   **preprocessing/**  
    |     Contiene los archivos de Python relacionados con el preprocesado y transformación de los vídeos para reducirlos a un conjunto de características.

/docs/

Proyecto en L<sup>A</sup>T<sub>E</sub>X que contiene este documento junto con la memoria.

  └─ schematics/

    Contiene diferentes ficheros .dawio utilizados durante el desarrollo del proyecto.

  └─ tex/

    Carpeta que contiene todos los fuentes .tex para memoria y anexos.

  └─ img/

    Diferentes imágenes utilizadas en la documentación.

      └─ concepts/

        Diferentes imágenes utilizadas en el apartado de conceptos teóricos.

      └─ graphs/

        Diferentes gráficos utilizados en la documentación.

      └─ introduction/

        Diferentes imágenes utilizadas en el apartado de introducción de la memoria.

      └─ manual/

        Diferentes imágenes utilizadas en el apartado del manual de estos anexos.

/notebooks/

Notebooks de Jupyter que se han utilizado para realizar pruebas, entrenar modelos, optimizar hiperparámetros y generar gráficas.

  └─ datasets/

    Resultados obtenidos tras el entrenamiento y evaluación de los modelos.

### D.3. Manual del programador

Se describen a continuación las herramientas y entornos utilizados durante el desarrollo del proyecto:

- **Sistema operativo:** para llevar a cabo la fase de desarrollo se ha hecho uso del sistema operativo Windows 11. No obstante, en la fase de producción, ha sido necesario el uso de *Windows Subsystem for Linux* (WSL)<sup>2</sup> para poder utilizar la herramienta *gunicorn*. Por lo que para que la fase de desarrollo y producción sea constante se recomienda el uso de Linux o en su defecto WSL desde el inicio.
- **Versión de Python:** la aplicación está desarrollada en Python<sup>3</sup> 3.10.7 debido a incompatibilidades de versión con Numba y CUDA. Se recomienda comprobar la versión de CUDA instalada en el dispositivo. Para CUDA 11, se ha utilizado Numba 0.56.0; para CUDA 12, se ha utilizado Numba 0.59.0
- **Visual Studio Code:** tanto para el control de versiones, mediante el *plugin* de GitHub de Visual Studio Code, como para el desarrollo del código se ha hecho uso de la versión 1.91.0 de Visual Studio Code<sup>4</sup>.

Aunque para la programación y ejecución del código se hizo uso de Visual Studio Code, se puede utilizar cualquier IDE(*Integrated development environment*) que sea compatible con Python.

Durante el desarrollo de la aplicación se hizo uso de un entorno virtual de Python. Antes de crear el entorno virtual, se recomienda consultar la versión de Python con la que se está trabajando mediante el siguiente comando:

```
1 python --version
```

Listing D.1: Comprobación de versión de Python

A su vez, se ha de tener en cuenta que el entorno virtual creado estará en la versión que nos aparezca tras ejecutar el comando anterior.

Para crear un entorno virtual se hará uso del siguiente comando:

---

<sup>2</sup><https://ubuntu.com/desktop/wsl>

<sup>3</sup><https://www.python.org/downloads/>

<sup>4</sup><https://code.visualstudio.com/download>

```
1 python -m venv nombre_del_entorno
```

Listing D.2: Creación de un entorno virtual

Una vez creado, antes de instalar las dependencias, se recomienda comprobar la versión de CUDA instalada. Para ello se utilizará el comando `nvidia-smi` el cual nos mostrará, entre otras cosas la versión de CUDA instalada.

## D.4. Compilación, instalación y ejecución del proyecto

A continuación se explican algunos de los procesos para poder modificar y utilizar el proyecto.

### Obtención de los ficheros fuente

Par obtener los ficheros fuente de la aplicación, se puede hacer uso del comando `git clone` o en su defecto, ir al GitHub<sup>5</sup> y, en el botón de *code*, seleccionar la opción de descargar *ZIP*.

### Instalación del entorno virtual y dependencias

Para instalar las librerías y dependencias necesarias para lanzar la aplicación, se incluye en el repositorio un archivo `requirements.txt` en el que se incluyen tanto librerías como dependencias con sus correspondientes versiones. Para crear este archivo se hizo uso de comando `pip freeze > requirements.txt`.

Para instalar las librerías en el entorno virtual primero deberá ser activado. En Windows, habrá que desplazarse hasta el directorio donde se encuentra el entorno virtual y, posteriormente, ejecutar `./nombre_del_entorno/Scripts/activate`. En Linux el proceso es similar salvo que el comando será `./nombre_del_entorno/bin/activate`. Una vez activado el entorno virtual, se ejecutará el siguiente comando:

```
1 python install -r requirements.txt
```

Listing D.3: Instalación dependencias y librerías

<sup>5</sup><https://github.com/JornezMM/PDAIvolution/tree/main>

Este comando instalará automáticamente las librerías y dependencias con sus correspondientes versiones, incluidas en el archivo previamente mencionado.

## Ejecución de experimentos en remoto

Debido a la gran cantidad de recursos necesarios para lanzar los experimentos, se ha hecho uso de un servidor remoto proporcionado por la UBU. Para ello se han seguido los siguientes pasos:

1. Activar VPN de la UBU<sup>6</sup>. Se necesita autorización previa.
2. Acceder al servidor de manera remota mediante ssh. Para ello se hace uso del comando `ssh -X -p 22 usuario@ipmáquina`. Tras ello solicitará las credenciales del usuario.
3. Descargarse el proyecto ya sea mediante `git clone` o enviarlo por ssh.
4. Crear un entorno virtual. Para ello primero habrá de instalar Miniconda y posteriormente ejecutar estos comandos:
  - a) `bash Miniconda3-latest-Linux-x86_64.sh`
  - b) `conda create -name EntornoProyecto python=3.10.2`
  - c) `conda activate EntornoProyecto`
  - d) `pip install -r requirements.txt`
5. Por último, ejecutar el notebook ModelTraining con el archivo .csv correspondiente a los datos extraídos mediante la librería Paddel.

## Despliegue de la aplicación en local

Para desplegar la aplicación en local en modo *debug*, se deberán seguir los siguientes pasos:

1. Ir al directorio `/web`
2. Activar el entorno virtual como se ha explicado anteriormente.
3. Ejecutar el comando `python run.py`

---

<sup>6</sup><https://www.ubu.es/servicio-de-informatica-y-comunicaciones/documentacion-de-ayuda/manuales-de-usuario/manuales-vpn>

Esto desplegará la aplicación en modo depuración, permitiendo hacer cambios y verlos en tiempo real en vez de tener que relanzar la aplicación.

Para desplegarla en producción se seguirán los siguientes pasos:

1. Ir al directorio `/web`.
2. Activar el entorno virtual como se ha explicado anteriormente.
3. Ejecutar el comando `gunicorn -bind 0.0.0.0:8000 wsgi:application`.

Esto desplegará la aplicación en el puerto 8000.

## Despliegue de la aplicación mediante Docker

Para instalar una imagen de Docker desde Docker Hub<sup>7</sup>, primero se debe asegurar de tener Docker instalado en tu sistema. Luego, abrir una terminal y usar el comando `docker pull` seguido del nombre de la imagen que deseas descargar. Para instalar la imagen de la aplicación, se debe ejecutar `docker push jornez/pdaivolution:tagname`. Esto descargará la imagen más reciente de la aplicación desde Docker Hub. Para lanzar la imagen se recomienda el uso de DockerDesktop ya que automatiza el despliegue de la aplicación. Para poderla desplegar correctamente, se ha de introducir el puerto de salida deseado, en este caso el 8000.

## D.5. Pruebas del sistema

En este apartado se muestran las pruebas para comprobar el correcto funcionamiento de la aplicación.

---

<sup>7</sup><https://hub.docker.com/repository/docker/jornez/pdaivolution/general>

---

Identificador: CP1

Prioridad: Alta

Fecha de ejecución: 05/07/2024

Realización: Manual

Objetivo: Probar el inicio de sesión

Precondiciones: No tener una sesión iniciada

Postcondiciones: Se ha guardado un token de sesión en la memoria local del navegador con el rol seleccionado

Acción	Entrada	Salida Esperada	Salida Real
Acceder a la página de inicio de sesión		Estar en la página de inicio de sesión	Estar en la página de inicio de sesión ✓
Introducir información de acceso	Usuario: admin; Contraseña: <code>defaultpassword</code> Rol:Administrador	Datos introducidos	Datos introducidos ✓
Enviar información mediante el botón «Iniciar sesión»		Estar en el panel de administración	Estar en el panel de administración ✓

Tabla D.1: CP1: Inicio de Sesión

---

Identificador: CP2

Prioridad: Alta

Fecha de ejecución: 05/07/2024

Realización: Manual

Objetivo: Probar el cierre de sesión

Precondiciones: Tener una sesión iniciada

Postcondiciones: Se ha eliminado el token de sesión

Acción	Entrada	Salida Esperada	Salida Real
Pulsar el botón de «Cerrar sesión»		Estar en la de inicio de sesión con la barra de navegación mostrando «Iniciar sesión»	Estar en la página de inicio de sesión con la barra de navegación mostrando «Iniciar sesión» ✓

Tabla D.2: CP2: Cierre de Sesión

---

Identificador: CP3

Prioridad: Alta

Fecha de ejecución: 05/07/2024

Realización: Manual

Objetivo: No poder acceder a una página sin permisos

Precondiciones: No ser tener permisos de la página

Postcondiciones: Ha devuelto al usuario a su página principal si tiene sesión iniciada o a la página de inicio de sesión

Acción	Entrada	Salida Esperada	Salida Real
Iniciar sesión como administrador	Estar en la página principal del administrador	Estar en la página principal del administrador	Estar en la página principal del administrador ✓
Introducir la URL de la página principal del paciente	Volver a la página de principal de administrador	Volver a la página de principal de administrador	Volver a la página de principal de administrador ✓
Cerrar sesión	Estar en la página de inicio de sesión	Estar en la página de inicio de sesión	Estar en la página de inicio de sesión ✓
Introducir la URL de la página principal del administrador	Volver a la página de principal de inicio de sesión	Volver a la página de principal de inicio de sesión	Volver a la página de principal de inicio de sesión ✓

Tabla D.3: CP3: Acceder a una página sin permisos

---

Identificador: CP4

Prioridad: Alta

Fecha de ejecución: 05/07/2024

Realización: Manual

Objetivo: Probar a añadir usuario

Precondiciones: El usuario que se desea añadir no existe y el usuario actual tiene la sesión iniciada como administrador

Postcondiciones: Ver en la página de administración el nuevo usuario

Acción	Entrada	Salida Esperada	Salida Real
Acceder al panel de gestión de usuarios		Estar en el panel del administrador	Estar en el panel del administrador ✓
Pulsar el botón «Añadir usuario»		Estar en la página de añadir usuario	Estar en la página principal de añadir usuario ✓
Introducir información del nuevo usuario	Nombre: Juan; Apellidos: Pérez; Nombre de usuario: admin2; Contraseña: 1234 Rol:Administrador	Datos introducidos	Datos introducidos ✓
Enviar información mediante el botón «Añadir usuario»		Estar en el panel de administración con el nuevo usuario añadido	Estar en el panel de administración con el nuevo usuario añadido ✓

Tabla D.4: CP4: Añadir usuario

---

Identificador: CP5

Prioridad: Media

Fecha de ejecución: 05/07/2024

Realización: Manual

Objetivo: Probar a eliminar a un médico con pacientes

Precondiciones: El usuario actual tiene la sesión iniciada como administrador

Postcondiciones: Aviso de no poder eliminar a un médico con pacientes

Acción	Entrada	Salida Esperada	Salida Real
Acceder al panel de gestión de usuarios		Estar en el panel del administrador	Estar en el panel del administrador ✓
Pulsar el botón «Eliminar usuario» al lado del médico		Mostrar un mensaje de confirmación	Mostrar un mensaje de confirmación ✓
Pulsar el botón de «Confirmar» en el mensaje de confirmación		Mostrar un aviso de que no se ha podido eliminar al médico	Mostrar un aviso de que no se ha podido eliminar al médico ✓

Tabla D.5: CP5: Eliminar médico con pacientes

---

Identificador: CP6

Prioridad: Media

Fecha de ejecución: 05/07/2024

Realización: Manual

Objetivo: Probar a eliminar a un administrador

Precondiciones: El usuario actual tiene la sesión iniciada como administrador

Postcondiciones: No aparece el administrador en el panel de administrador

Acción	Entrada	Salida Esperada	Salida Real
Acceder al panel de gestión de usuarios		Estar en el panel del administrador	Estar en el panel del administrador ✓
Pulsar el botón «Eliminar usuario» al lado del administrador		Mostrar un mensaje de confirmación	Mostrar un mensaje de confirmación ✓
Pulsar el botón de «Confirmar» en el mensaje de confirmación		Estar en el panel del administrador sin el paciente	Estar en el panel del administrador sin el paciente ✓

Tabla D.6: CP6: Eliminar administrador

---

Identificador: CP7

Prioridad: Media

Fecha de ejecución: 05/07/2024

Realización: Manual

Objetivo: Probar a eliminar a un paciente con vídeos y medicinas

Precondiciones: El usuario actual tiene la sesión iniciada como administrador

Postcondiciones: No aparece el paciente en el panel de administrador

<b>Acción</b>	<b>Entrada</b>	<b>Salida Esperada</b>	<b>Salida Real</b>
Acceder al panel de gestión de usuarios		Estar en el panel del administrador	Estar en el panel del administrador ✓
Pulsar el botón «Eliminar usuario» al lado del paciente		Mostrar un mensaje de confirmación	Mostrar un mensaje de confirmación ✓
Pulsar el botón de «Confirmar» en el mensaje de confirmación		Estar en el panel del administrador sin el paciente	Estar en el panel del administrador sin el paciente ✓

Tabla D.7: CP7: Eliminar paciente con vídeos y medicinas

---

Identificador: CP8

Prioridad: Alta

Fecha de ejecución: 05/07/2024

Realización: Manual

Objetivo: Probar a modificar usuario

Precondiciones: El usuario actual tiene la sesión iniciada como administrador

Postcondiciones: Ver en la página de administración el usuario modificado

Acción	Entrada	Salida Esperada	Salida Real
Acceder al panel de gestión de usuarios		Estar en el panel del administrador	Estar en el panel del administrador ✓
Pulsar el botón «Modificar usuario» al lado del administrador		Estar en la página de añadir usuario	Estar en la página principal de añadir usuario ✓
Introducir información del nuevo usuario	Nombre: Juan; Apellidos: Pérez; Nombre de usuario: admin3; Contraseña: Rol:Administrado;	Datos introducidos	Datos introducidos ✓
Enviar información mediante el botón «Modificar usuario»		Estar en el panel de administración con el usuario modificado	Estar en el panel de administración con el usuario modificado ✓

Tabla D.8: CP8: Modificar usuario

---

Identificador: CP9

Prioridad: Alta

Fecha de ejecución: 05/07/2024

Realización: Manual

Objetivo: Probar a añadir un vídeo a un paciente

Precondiciones: El usuario actual tiene la sesión iniciada como médico

Postcondiciones: El vídeo aparece con las predicciones en el listado de vídeos del paciente

Acción	Entrada	Salida Esperada	Salida Real
Acceder al panel de gestión de pacientes		Estar en el panel del médico	Estar en el panel del médico ✓
Pulsar el botón de «Gestionar vídeos» al lado del paciente		Estar en la página de gestión de vídeos	Estar en la página de gestión de vídeos ✓
Pulsar el botón «Añadir vídeo»		Estar en la página de añadir vídeo	Estar en la página de añadir vídeo ✓
Introducir información del nuevo vídeo	Fichero: video.mp4; Mano: introducidos Izquierda; Fecha: 05/07/2024	Datos	Datos ✓ introducidos
Enviar información mediante el botón «Añadir usuario»		Estar en el panel de vídeos del paciente con el nuevo vídeo añadido	Estar en el panel de vídeos del paciente con el nuevo vídeo añadido ✓

Tabla D.9: CP9: Añadir vídeo

---

Identificador: CP10

Prioridad: Media

Fecha de ejecución: 05/07/2024

Realización: Manual

Objetivo: Probar a eliminar un vídeo a un paciente

Precondiciones: El usuario actual tiene la sesión iniciada como médico

Postcondiciones: El vídeo no aparece en el listado de vídeos del paciente

Acción	Entrada	Salida Esperada	Salida Real
Acceder al panel de gestión de pacientes		Estar en el panel del médico	Estar en el panel del médico ✓
Pulsar el botón de «Gestionar vídeos» al lado del paciente		Estar en la página de gestión de vídeos	Estar en la página de gestión de vídeos ✓
Pulsar el botón «Eliminar vídeo» al lado del vídeo		Mostrar un mensaje de confirmación	Mostrar un mensaje de confirmación ✓
Pulsar el botón de «Confirmar» en el mensaje de confirmación		Estar en la página de gestión de vídeos sin el vídeo	Estar en la página de gestión de vídeos sin el vídeo ✓

Tabla D.10: CP10: Eliminar vídeo

---

Identificador: CP11

Prioridad: Media

Fecha de ejecución: 05/07/2024

Realización: Manual

Objetivo: Probar a añadir medicina

Precondiciones: El usuario actual tiene la sesión iniciada como médico

Postcondiciones: Eliminación tanto del paciente como de los vídeos y medicinas asociadas

Acción	Entrada	Salida Esperada	Salida Real
Acceder al panel de gestión de pacientes		Estar en el panel del médico	Estar en el panel del médico ✓
Pulsar el botón de «Gestionar medicinas» al lado del paciente		Estar en la página de gestión de vídeos	Estar en la página de gestión de vídeos ✓
Pulsar el botón «Añadir medicina»		Estar en la página de añadir medicina	Estar en la página de añadir medicina ✓
Introducir información de la nueva medicina	Medicina: Medicine1; Dosis: introducidos 1 pastilla; Fecha inicio: 02/07/2024; Fecha fin: 05/07/2024	Datos	Datos introducidos ✓
Enviar información mediante el botón «Añadir usuario»		Estar en el panel de vídeos del paciente con el nuevo video añadido	Estar en el panel de vídeos del paciente con el nuevo video añadido ✓

Tabla D.11: CP11: Añadir medicina

---

Identificador: CP12

Prioridad: Baja

Fecha de ejecución: 05/07/2024

Realización: Manual

Objetivo: Probar a eliminar un medicina a un paciente

Precondiciones: El usuario actual tiene la sesión iniciada como médico

Postcondiciones: El vídeo no aparece en el listado de medicinas del paciente

Acción	Entrada	Salida Esperada	Salida Real
Acceder al panel de gestión de pacientes	Estar en el panel del médico	Estar en el panel del médico	✓
Pulsar el botón de «Gestionar medicinas» al lado del paciente	Estar en la página de gestión de vídeos	Estar en la página de gestión de vídeos	✓
Pulsar el botón «Eliminar medicina» al lado de la medicina	Mostrar un mensaje de confirmación	Mostrar un mensaje de confirmación	✓
Pulsar el botón de «Confirmar» en el mensaje de confirmación	Estar en la página de gestión de medicinas sin la medicina	Estar en la página de gestión de medicinas sin el medicina	✓

Tabla D.12: CP12: Eliminar medicina

---

Identificador: CP13

Prioridad: Media

Fecha de ejecución: 05/07/2024

Realización: Manual

Objetivo: Probar a visualizar gráfica con 1 o menos vídeos

Precondiciones: El usuario actual tiene la sesión iniciada como paciente

Postcondiciones: Mostrar un aviso de que no se ha podido dibujar la gráfica

Acción	Entrada	Salida Esperada	Salida Real
Acceder a la página principal del paciente	Estar en la página principal del paciente	Estar en la página principal del paciente	✓
Visualizar gráfica	Mostrar un mensaje de vídeos insuficientes	Mostrar un mensaje de vídeos insuficientes	✓

Tabla D.13: CP13: Error en la gráfica del paciente

---

Identificador: CP14

Prioridad: Media

Fecha de ejecución: 05/07/2024

Realización: Manual

Objetivo: Probar a visualizar gráfica con 2 o más videos

Precondiciones: El usuario actual tiene la sesión iniciada como paciente

Postcondiciones: Mostrar un aviso de que no se ha podido dibujar la gráfica

<b>Acción</b>	<b>Entrada</b>	<b>Salida Esperada</b>	<b>Salida Real</b>
Acceder a la página principal del paciente		Estar en la página principal del paciente	Estar en la página principal del paciente ✓
Visualizar gráfica		Mostrar la gráfica del paciente	Mostrar la gráfica del paciente ✓

Tabla D.14: CP14: Visualizar gráfica del paciente

## *Apéndice E*

---

# **Documentación de usuario**

---

## **E.1. Introducción**

En este apartado, se detalla la información necesaria para el uso de la aplicación por un usuario. Además, en este se indican los requisitos necesarios para hacer un uso correcto de la aplicación.

## **E.2. Requisitos de usuarios**

Para poder hacer uso de la aplicación es necesario disponer de conexión a internet y utilizar cualquiera de los siguientes navegadores: Mozilla Firefox, Safari, Opera, Microsoft Edge o Google Chrome.

Para el desarrollo y pruebas de la aplicación se ha hecho uso tanto del navegador de Opera como del de Google Chrome.

## **E.3. Instalación**

Dado que se trata de una aplicación web alojada en un servidor, no es necesario ningún tipo de instalación. No obstante, debido a que la aplicación se encuentra alojada dentro de los servidores de la UBU es necesario tener acceso a la Intranet ya sea mediante el uso de una VPN previamente autorizada o mediante el acceso a internet dentro de los recintos físicos de la universidad. Para poder acceder a la aplicación, una vez dentro de la Intranet, se hará mediante la dirección: <http://10.168.168.34:9000/>.

## E.4. Manual del usuario

Una vez accedido a la aplicación, el usuario tendrá acceso a la página principal. Dentro de la aplicación existen diversas funcionalidades dependiendo del tipo de usuario que tenga iniciada la sesión.

En este apartado se mostrarán tanto las funcionalidades comunes, que son aquellas que se pueden realizar independientemente del tipo de usuario como las funcionalidades específicas de cada tipo de usuario: administrador, médico, paciente.

### Funcionalidades generales

#### Página principal

Al tomar contacto por primera vez con la aplicación, la primera página que se mostrará será la página principal (ver figura E.1), la cual se compone de una barra de navegación donde se encuentra los botones de «iniciar sesión» y «acerca de».



Figura E.1: Página principal

#### Inicio de sesión

Al pulsar en el botón de «inicio de sesión» se redirige al usuario a la página de inicio de sesión (ver figura E.2) la cual consiste en un formulario en el que se solicitan las credenciales y el rol con el que se desea acceder.



Figura E.2: Página de inicio de sesión

### Acerca de

Al pulsar en el botón de «acerca de» se redirige al usuario a la página de información (ver figura E.3) en la que se muestra una breve descripción de la aplicación y del estudiante así como un enlace de contacto.



Figura E.3: Página de información

## Funcionalidades del administrador

### Página de gestión de usuarios

Al iniciar sesión como administrador, se redirige al usuario a la página de gestión de usuarios (ver figura E.4). En esta página se muestra un listado con todos los usuarios de la base de datos junto a las opciones de eliminar y

modificar al usuario. Además se puede encontrar el botón de «añadir usuario».

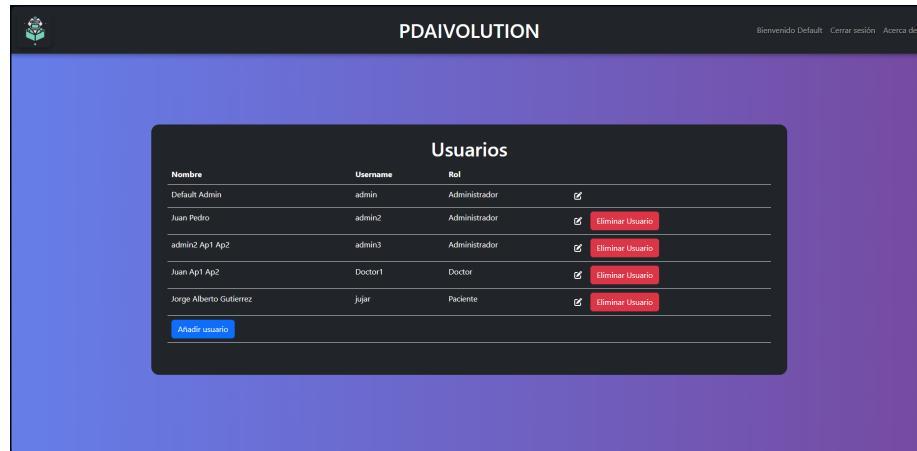


Figura E.4: Página del administrador

### Página de añadir usuarios

Al pulsar el botón de «añadir usuario» se redirigirá a la página de añadir usuarios (ver figura E.5), la cual se compone de un formulario que varía dependiendo del tipo de usuario que se desee añadir. Una vez llenado el formulario, al pulsar el botón de «añadir usuario», se añadirá a la base de datos si cumple los requisitos.

Figura E.5: Página de añadir usuarios

### Página de modificar usuarios

Al pulsar el botón con el símbolo de edición se redirigirá a la página de modificar usuarios con los datos de ese usuario (ver figura E.6), la cual se compone de un formulario que varía dependiendo del tipo de usuario que se desee modificar. El formulario se rellena automáticamente con los datos del usuario que se desea modificar. Si la contraseña se deja en blanco, no se modificará la contraseña actual del usuario a modificar. Una vez llenado el formulario, al pulsar el botón de «modificar usuario», se guardará en la base de datos si cumple los requisito el nuevo usuario modificado.

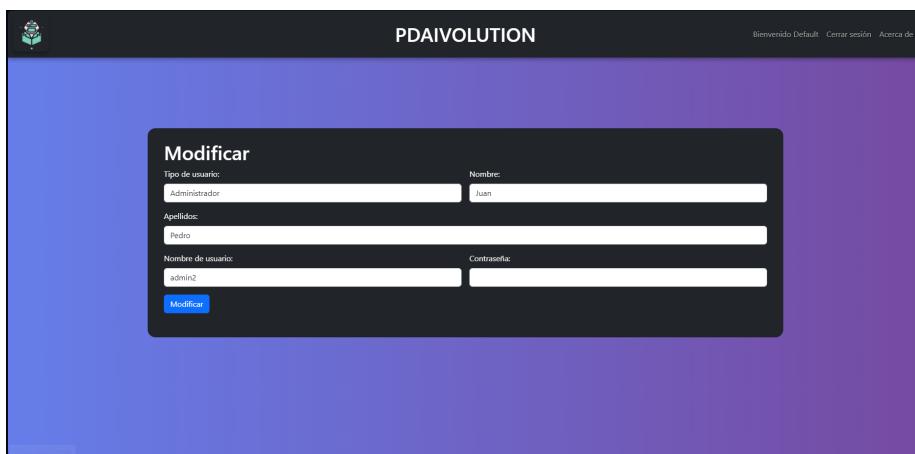


Figura E.6: Página de modificar usuarios

## Funcionalidades del administrador

### Página principal del médico

Al iniciar sesión como médico, se redirige al usuario a la página de gestión de pacientes (ver figura E.7). En esta página se muestra un listado con todos los pacientes asignados al usuario con la sesión iniciada junto a las opciones de «ver usuario», «ver medicinas» y «ver vídeos del usuario». Además se puede encontrar el botón de «añadir usuario».

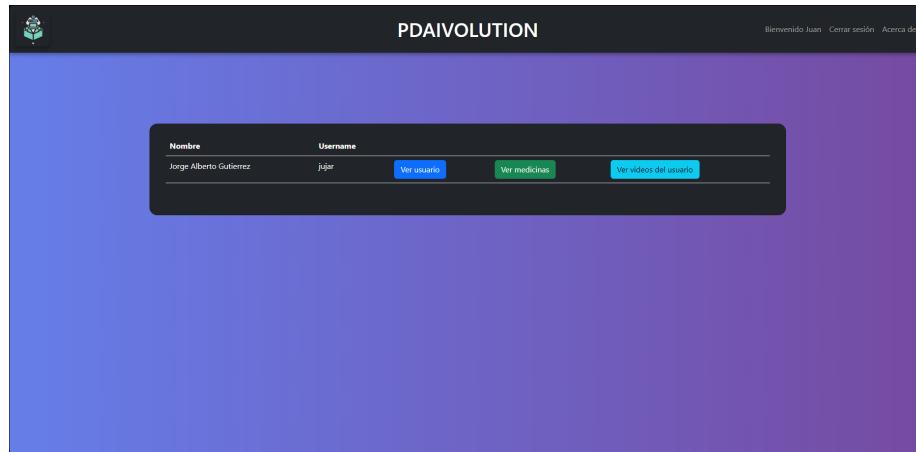


Figura E.7: Página del médico

### Página de evolución del paciente

Al pulsar el botón de «ver usuario» se redirigirá a la página de evolución del paciente (ver figura E.8), en la que se muestran los datos del paciente como su nombre, edad, mano, medicinas, así como la gráfica de evolución del paciente.



Figura E.8: Página de evolución del paciente

### Página de gestión de medicinas

Al pulsar el botón de «ver medicinas» se redirigirá a la página de medicinas del paciente (ver figura E.9), en la que se muestra un listado con la medicina, dosis y fechas de inicio y fin del tratamiento. Además, se incluyen dos botones adicionales, uno «eliminar medicina» que como su texto indica

elimina la medicina de la lista y un botón de «añadir medicinas» que redirige al médico a la página de añadir medicinas.



Figura E.9: Página de medicinas del paciente

### Página de añadir medicinas

Al pulsar el botón de «añadir medicina» se redirigirá a la página de añadir medicinas (ver figura E.10), la cual se compone de un formulario en el que se introducen los datos de la medicina que se desee añadir. Estos datos incluyen el nombre de la medicina, la dosis y la fecha de inicio y fin del tratamiento. Si la fecha de fin se deja en blanco, se interpreta que el tratamiento sigue en curso. Una vez llenado el formulario, al pulsar el botón de «añadir medicina», se añadirá la medicina al listado del paciente.

Figura E.10: Página de añadir medicinas

## Página de gestión de vídeos

Al pulsar el botón de «ver vídeos» se redirigirá a la página de vídeos del paciente (ver figura E.11), en la que se muestra un listado con los vídeos, amplitud, lentitud, mano y fechas grabación. Además, se incluyen tres botones adicionales, uno «eliminar vídeo» que como su texto indica elimina el vídeo de la lista y un botón de «añadir vídeos» que redirige al médico a la página de añadir medicinas y por último el botón de ver vídeo.

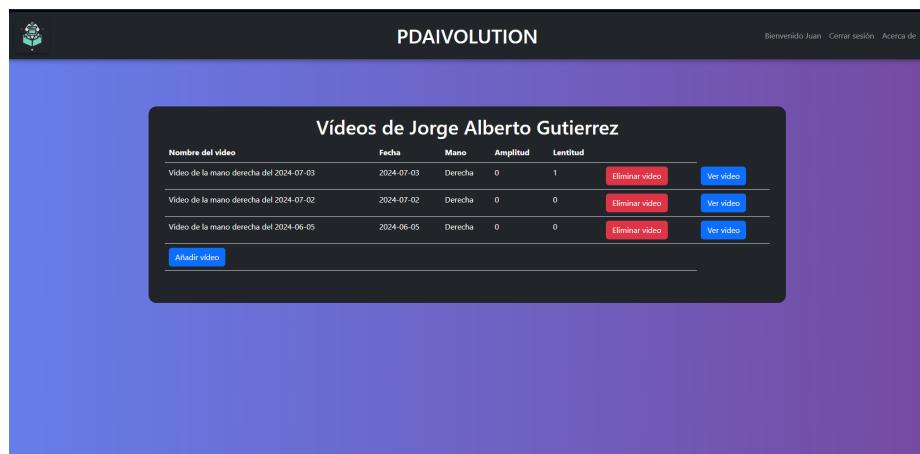


Figura E.11: Página de vídeos del paciente

## Página de añadir vídeos

Al pulsar el botón de «añadir vídeos» se redirigirá a la página de añadir medicinas (ver figura E.12), la cual se compone de un formulario en el que se introducen los datos: archivo de vídeom mano y día en el que fue tomado. Una vez rellenado el formulario, al pulsar el botón de «añadir vídeos», se añadirá el vídeo al listado del paciente.

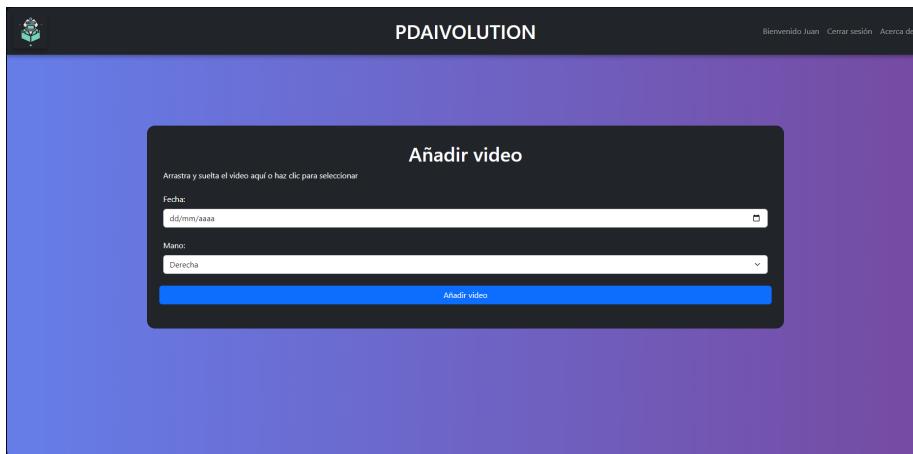


Figura E.12: Página de añadir videos

## Funcionalidades del paciente

### Página principal del paciente

Al iniciar sesión como paciente, se redirige al usuario a la página principal de usuarios (ver figura E.13). En esta página al igual que en la página de vista de evolución del paciente, se muestra una tabla con los datos del paciente, así como la gráfica de su evolución.



Figura E.13: Página del médico



## *Apéndice F*

---

# **Anexo de sostenibilización curricular**

---

## **F.1. Introducción**

Este apartado, aborda los diversos aspectos de sostenibilidad aplicados durante el desarrollo del proyecto. Para la elaboración de este apartado, se ha utilizado como referencia el documento «Directrices para la inclusión de la sostenibilidad en el currículo universitario» publicado por la Conferencia de Rectores de las Universidades Españolas (CRUE)<sup>1</sup>. Este documento proporciona un marco teórico y práctico para integrar la sostenibilidad en los proyectos educativos y de investigación.

Este apartado se divide en tres secciones: impacto del software a nivel de consumo, igualdad y accesibilidad y conclusiones.

## **F.2. Impacto del software a nivel de consumo**

El impacto del software en términos de consumo energético y de recursos es un aspecto crítico a considerar en el desarrollo de aplicaciones sostenibles. Durante el proyecto se ha hecho uso de modelos de aprendizaje y servidores.

---

<sup>1</sup>[https://www.crue.org/wp-content/uploads/2020/02/Directrices\\_Sostenibilidad\\_Crue2012.pdf](https://www.crue.org/wp-content/uploads/2020/02/Directrices_Sostenibilidad_Crue2012.pdf)

## Servidor

Para el despliegue de la aplicación se ha hecho uso del servidor de la UBU. Este servidor genera un consumo constante debido a la necesidad de mantener la infraestructura para poder proporcionar servicios de manera ininterrumpida. Es este uso continuo el que lleva a un consumo energético elevado. Para solucionar esto se podría hacer un estudio para mejorar los procesadores y componentes del servidor para ser más eficientes energéticamente o apagar aquellos componentes y servicios que no estén en uso.

## Aprendizaje automático

Los modelos de aprendizaje automático consumen una cantidad significativa de energía por varias razones, relacionadas principalmente con el proceso de entrenamiento y la infraestructura necesaria para ejecutar estos modelos. Debido a que el entrenamiento de modelos, especialmente los de deep learning, requiere realizar cálculos complejos y numerosas iteraciones sobre grandes conjuntos de datos y las unidades de procesamiento gráfico (GPU) y las unidades de procesamiento tensorial (TPU) son esenciales para el entrenamiento eficiente, pero también son muy intensivas en energía.

Como solución se propone el uso de modelos más pequeños y eficiente, entrenamiento distribuido o adquisición de *hardware* específico. Otras soluciones podría ser entrenamiento fuera de las horas pico programando el entrenamiento de modelos durante períodos de baja demanda energética u optimización de la carga de trabajo balanceando y distribuyendo la carga de trabajo de manera óptima para evitar picos de consumo energético. El desarrollo de este software también ha considerado la sostenibilidad a largo plazo, buscando minimizar su huella de carbono. Se han implementado prácticas de programación eficientes, optimizando el código para reducir el tiempo de procesamiento y, por ende, el consumo energético.

## F.3. Igualdad y accesibilidad

Uno de los objetivos fundamentales del software es su accesibilidad para todos los usuarios, independientemente de sus capacidades o limitaciones. Este enfoque inclusivo garantiza que el software sea una herramienta útil y eficaz para el mayor número posible de personas, alineándose con los principios de igualdad y no discriminación.

El compromiso con la igualdad también se refleja en la estructura de precios del software. Dado que el proyecto está destinado a facilitar el trabajo de pacientes y médicos, se ha decidido que su uso sea completamente gratuito. Esta decisión garantiza que el software esté disponible para todos los usuarios, independientemente de su situación económica, promoviendo así la igualdad de oportunidades .

## F.4. Conclusiones

La integración de la sostenibilidad en el desarrollo de software es esencial para minimizar el impacto ambiental y promover la igualdad de acceso a la tecnología. Este proyecto ha sido diseñado con un enfoque holístico, considerando tanto la eficiencia energética como la accesibilidad y la igualdad. Siguiendo las directrices de sostenibilidad de la CRUE, se ha logrado desarrollar un software que no solo cumple con sus objetivos funcionales, sino que también contribuye a un futuro más sostenible y equitativo.



---

## Bibliografía

---

- [1] Catalin Andrei Cacuci. Github:paddel.  
<https://github.com/cataand/tfg-paddel>, 2024.
- [2] Taha Khan, Dag Nyholm, Jerker Westin, and Mark Dougherty. A computer vision framework for finger-tapping evaluation in parkinson's disease. *Artificial intelligence in medicine*, 60(1):27–40, 2014.
- [3] Marta Palacio. *Scrum Master. Temario troncal 1.* 02 2022.
- [4] Stefan Williams, Samuel D Relton, Hui Fang, Jane Alty, Rami Qahwaji, Christopher D Graham, and David C Wong. Supervised classification of bradykinesia in parkinson's disease from smartphone videos. *Artificial Intelligence in Medicine*, 110:101966, 2020.
- [5] Stefan Williams, Zhibin Zhao, Awais Hafeez, David C Wong, Samuel D Relton, Hui Fang, and Jane E Alty. The discerning eye of computer vision: Can it measure parkinson's finger tap bradykinesia? *Journal of the Neurological Sciences*, 416:117003, 2020.