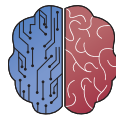




UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería de la Salud



INGENIERÍA
DE LA SALUD

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

**Aplicación web para el
seguimiento de la actividad de
las personas con enfermedad
de Parkinson
Documentación Técnica**

Presentado por Inés Martos Barbero
en Universidad de Burgos

12 de febrero de 2024

Tutor: Guirguis Zaki Guirguis Abdelmessih

Índice general

I

E.1. Planos	49
E.2. Diseño arquitectónico	49
Apéndice F Especificación de Requisitos	57
F.1. Diagrama de Casos de Uso	58
F.2. Explicación Casos de Uso.	58
F.3. Prototipos de interfaz.	60
Apéndice G Estudio experimental	97
G.1. Detalle de resultados.	97
Apéndice H Anexo de sostenibilización curricular	103
Bibliografía	105

Índice de figuras

A.1. Diagrama de Gantt inicial	5
A.2. Diagrama de Gantt final	5
A.3. Fórmula para el Cálculo de la Amortización durante un Periodo Determinado	7
B.1. Proceso de calibrado.	19
B.2. XAMPP Control Panel.	20
B.3. Éxito en la inicialización del servidor.	21
B.4. Instalación de MySQL en el servidor.	21
B.5. Fragmento del script <i>bridge.py</i> en el que se debe modificar el puerto COM.	22
B.6. Correcta colocación del sensor en el tobillo izquierdo.	23
B.7. Interior de la carcasa que protege al sensor MPU6050.	23
B.8. Pantalla para el inicio de una sesión de monitoreo.	24
B.9. Pantalla de inicio para el usuario administrador.	25
B.10. Pantalla de inicio para el usuario profesional.	25
B.11. Pantalla de inicio para el usuario paciente.	26
C.1. Terminal CMD muestra los datos recibidos por el servidor.	38
C.2. Terminal VS Code. Acción ‘realizar actividad’.	39
C.3. Terminal VS Code. Acción ‘iniciar actividad’.	39
C.4. Terminal VS Code. Flujo de datos durante la realización de una actividad de marcha.	40
E.1. Esquema de montaje del prototipo de partida. [González, 2023]	50
E.2. Esquema de montaje del dispositivo final.	51
E.3. Hardware del dispositivo final.	52
E.4. Diagrama de flujo - Inicio y fin de sesión.	53

E.5. Diagrama de flujo. Usuario administrador.	54
E.6. Diagrama de flujo. Usuario profesional.	55
E.7. Diagrama de flujo. Usuario paciente.	56
F.1. CU-Administradores, Pacientes y Profesionales	58
F.2. CU-Administrador	60
F.3. CU-Pacientes y Profesionales	61
F.4. Iniciar sesión	63
F.5. Confirmar operación	63
F.6. Consulta y gestión de usuarios	64
F.7. Crear usuario	90
F.8. Consultar lista de pacientes	91
F.9. Consultar información personal y otras actividades	91
F.10. Iniciar actividad	92
F.11. Consultar actividades y estadísticas (Paciente)	93
F.12. Consultar actividades y estadísticas (Profesional)	94
F.13. Consulta y gestión de pacientes	95
F.14. Crear paciente	96
F.15. Visualizar paciente	96
G.1. Cuestionario SUS. Parte 1.	98
G.2. Cuestionario SUS. Parte 2.	99
G.3. Cuestionario SUS. Parte 3.	100
G.4. Cuestionario SUS. Parte 4.	101

Todas las figuras en las que no se indica lo contrario, han sido elaboradas por Inés Martos Barbero, la autora de este trabajo.

Índice de tablas

A.1. Desglose de costes de contratación	6
A.2. Desglose de costes de los componentes	8
A.3. Herramientas empleadas y sus licencias	11
B.1. RF-01 Iniciar Sesión	15
B.2. RF-02 Consultar pacientes y usuarios	15
B.3. RF-03 Gestionar pacientes y usuarios	15
B.4. RF-04 Realizar actividad	15
B.5. RF-05 Mostrar actividades	16
B.6. RF-06 Consultar Estadísticas	16
B.7. RF-07 Gestionar cuenta	16
B.8. RF-08 Cerrar sesión	16
B.9. Componentes del Prototipo	17
B.10.RF-09 Transmisión de datos sin cables	18
B.11.RF-10 Alimentación externa	18
B.12.RF-11 Encendido y apagado del sistema	18
B.13.RF-12 Seguridad y manejabilidad del prototipo	18
F.1. CU-1 Iniciar sesión	59
F.2. CU-2 Confirmar operación	64
F.3. CU-3 Gestionar cuenta	65
F.4. CU-4 Actualizar correo	66
F.5. CU-5 Cambiar contraseña	67
F.6. CU-6 Cerrar sesión.	68
F.7. CU-7 Consulta y gestión de usuarios	69
F.8. CU-8 Buscar usuario	70
F.9. CU-9 Crear usuario	71
F.10.CU-10 Especificar características usuario profesional	72

F.11.CU-11 Especificar características usuario paciente	73
F.12.CU-12 Gestionar función admin	74
F.13.CU-13 Eliminar usuario	75
F.14.CU-14 Consultar lista de pacientes	76
F.15.CU-15 Consultar información personal y otras operaciones . . .	77
F.16.CU-16 Iniciar actividad	78
F.17.CU-17 Finalizar actividad	79
F.18.CU-18 Consultar actividades y estadísticas	80
F.19.CU-19 Buscar actividad	81
F.20.CU-20 Consulta y gestión de pacientes	82
F.21.CU-21 Buscar paciente	83
F.22.CU-22 Dar de alta paciente	84
F.23.CU-23 Crear paciente	85
F.24.CU-24 Eliminar paciente	86
F.25.CU-25 Visualizar paciente	87
F.26.CU-26 Modificar paciente	88
F.27.CU-27 Eliminar cuenta	89

Todas las tablas en las que no se indica lo contrario, han sido elaboradas por Inés Martos Barbero, la autora de este trabajo.

Apéndice A

Plan de Proyecto Software

A.1. Introducción

Antes de iniciar un proyecto, es fundamental considerar los aspectos relativos al tiempo de desarrollo, los costes económicos y el marco legal que le afecta. Una planificación detallada previene complicaciones tras el inicio, permite la adopción de estrategias efectivas para gestionar tiempo y recursos, y asegura la viabilidad del proyecto conforme a la ley.

A.2. Planificación temporal

GitHub fue la herramienta empleada para controlar el avance del proyecto. En el repositorio se establecieron una serie de tareas (issues), agrupándolas en hitos (milestones). También se definieron diversas etiquetas (labels) para su categorización, destinadas a indicar el tipo de tarea y su estado.

Desglose de milestones e issues definidos:

1. Investigación y planificación.
 - Definir objetivos y comprender requisitos del proyecto
 - Comprender TFG Detección Actividad Muscular
 - Investigar sensor MPU6050
 - 1er contacto con métodos de conexión
 - Crear un plan de trabajo
 - Añadir licencia al proyecto

2. Configuración del sensor y pruebas de conexión.

El objetivo principal es replicar el TFG Detección de Actividad Muscular de Sara González [González, 2023].

- Montaje hardware
- Configuración software
- Pruebas correcto funcionamiento del dispositivo
- Evaluación, comparación y selección del método de conexión

3. Diseño de la aplicación.

- Descripción Casos de Uso (CU)
- Diagramas Casos de Uso
- Descripción Requisitos Funcionales (RF)
- Diseño de la Interfaz de Usuario (UI). Creación de wireframes y mockups

4. Desarrollo página web.

- Cambiar contraseña
- Actualizar correo
- Realizar actividad como profesional
- Consultar actividades
- Incluir paciente (profesional)
- Inicio profesional
- Listado pacientes de cada profesional
- Eliminar usuario (admin)
- Usuario registrado elimina su cuenta
- Quitar paciente
- Consultar paciente
- Crear usuario (admin)
- Inicio admin
- Crear base de datos
- Menú usuarios
- Inicio paciente

- Finalizar actividad
- Inicio actividad
- Confirmar operación
- Cerrar sesión
- Login (1)
- Login (2)

5. Protocolo de conexión.

Incluye las tareas necesarias para realizar con éxito la conexión y envío de datos entre el dispositivo hardware y la página web.

- Enviar datos bluetooth (base)
- Recibir datos bluetooth (base)
- Iniciar actividad desde la web
- Finalizar actividad desde la web
- Ver datos actividad en tiempo real en la web
- Almacenar datos actividad en BD
- Obtener datos para la actividad BD

6. Pruebas finales, análisis y mejoras.

- Funcionamiento del sistema (sin las mejoras hardware)
- Mejoras hardware
- Funcionamiento del sistema (hardware mejorado)

7. Redacción de memoria.

- 0. Resumen y abstract
- 1. Introducción
- 2. Objetivos
- 3.1 Conceptos teóricos
- 3.2 Conceptos teóricos. Estado del arte y trabajos relacionados
- 4.1 Metodología. Descripción de los datos
- 4.2 Metodología. Técnicas y herramientas
- 5.1 Resultados. Resumen
- 5.2 Resultados. Discusión

- 6. Conclusiones
 - 6.1 Conclusiones. Aspectos relevantes
 - 7. Líneas de trabajo futuras
 - 8. Bibliografía
8. Redacción de anexos.
- A.1 Introducción
 - A.2 Planificación temporal
 - A.3 Planificación económica
 - A.4 Viabilidad legal
 - B.1 Requisitos software y hardware para ejecutar el proyecto
 - B.2 Instalación/Puesta en marcha
 - B.3 Manuales y/o Demostraciones prácticas
 - C.1 Estructura de directorios
 - C.3 Pruebas del sistema.
 - C.4 Instrucciones para la modificación o mejora del proyecto
 - D Descripción de adquisición y tratamiento de datos
 - E.1 Planes
 - E.2 Diseño arquitectónico
 - F Especificación de requisitos (completar)
 - H Anexo de sostenibilización curricular
9. Preparación de la presentación.

Diagrama de Gantt

Para la planificación temporal del proyecto, se elaboró un diagrama de Gantt. Esta herramienta permite visualizar los hitos del proyecto y su duración en el tiempo.

El planteamiento inicial está representado en la Figura A.1. Con el transcurso del tiempo, se hicieron evidentes la imposibilidad de cumplir los plazos y la necesidad de modificar esta planificación, ya que no se había realizado de manera realista debido a la falta de experiencia y a la aparición de imprevistos. Se llevó a cabo una segunda planificación, que se ilustra en la Figura A.2.

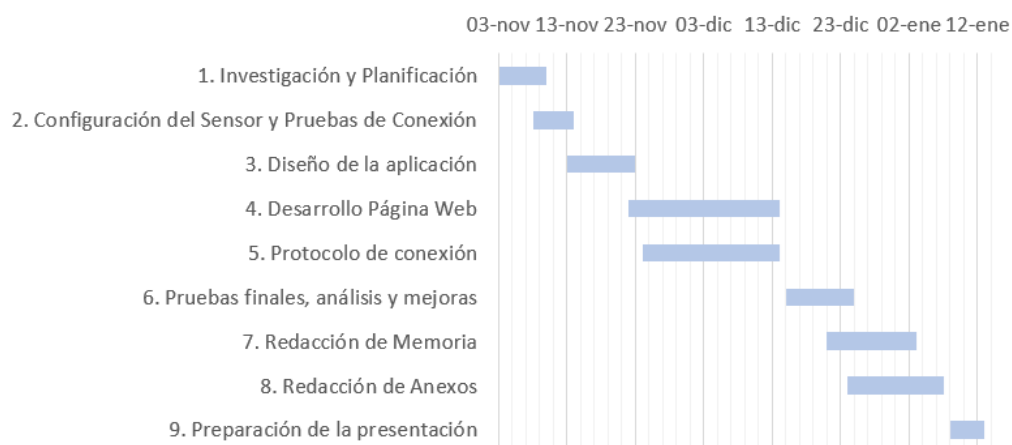


Figura A.1: Diagrama de Gantt inicial

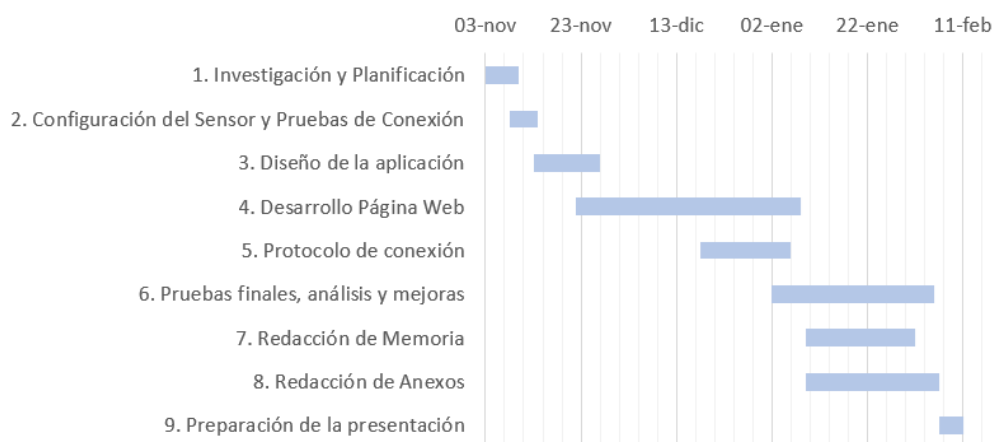


Figura A.2: Diagrama de Gantt final

A.3. Planificación económica

En una planificación económica realista hay que considerar todos los costes de realización del proyecto. Esto incluye desde la valoración económica de herramientas de desarrollo software hasta la de materiales, equipos y sueldos del personal necesario. Además de los costos, también se debe tener en cuenta la rentabilidad a través de diversas opciones de explotación.

En este apartado se presenta la planificación económica del proyecto realizado. Se incluye el desarrollo software de la página web y el montaje hardware de un prototipo anterior sobre el que se han realizado modificaciones.

Sueldo bruto (€/mes)	2.418,33
Costes del Régimen General de la Seguridad Social	
Contingencias Comunes (23,60 %)	570,72
Contrato duración determinada Tiempo Completo (6,7 %)	162,02
FOGASA (0,2 %)	4,83
Formación Profesional (0,6 %)	14,51
Costo total - 1 mes	3.170,41
Costo total - 2 meses	6.340,82 €

Tabla A.1: Desglose de costes de contratación

Coste de personal

Ha habido únicamente una persona involucrada en el proyecto, desde el desarrollo hasta el diseño, que para el cálculo de los costos de contratación será considerada Ingeniero de la Salud sin experiencia. En España, un Ingeniero Biomédico recién egresado o con menos de 3 años de experiencia, que es la situación que más se ajusta a este caso, puede aspirar a un salario medio bruto de 29.020 €/año [Jobted, 2024]. Teniendo en cuenta que el contrato necesario para la realización de este proyecto debe ser de 2 meses de duración a jornada completa, el costo del empleado será de 4.836,66 €, cantidad a la que habrá que añadirle el pago de la Seguridad Social que corre a cargo de la empresa. La cantidad de dinero a pagar según el Régimen General de la Seguridad Social [Seguridad-Social, 2023] se puede calcular a través de unos porcentajes a aplicar sobre el salario bruto como se muestra en la Tabla A.1.

Costes de software

Todas las herramientas software empleadas compartían la característica de ser de código abierto y gratuitas. Visual Studio Code, XAMPP, Node.js y Arduino IDE han sido algunas de las utilizadas.

Amortización de los equipos

El único equipo indispensable para el desarrollo del proyecto ha sido un portátil de la marca HP (modelo HP Pavillion Laptop 15-cs2xxx), cuyo precio de compra fue de 849,45€ y al que se le estima una vida útil de 6 años. La amortización ¹ del portátil durante los dos meses y medio que duró el

¹Amortización es la pérdida de valor de un bien o activo a lo largo del tiempo.

trabajo puede calcularse aplicando la fórmula de la Figura A.3 , obteniendo como resultado 29,49 € [ING, 2022].

$$\text{Amortización del Periodo} = \frac{\text{Precio de Compra}}{\text{Vida Útil en Años}} \times \frac{\text{Meses de Uso}}{12}$$

Figura A.3: Fórmula para el Cálculo de la Amortización durante un Periodo Determinado

Costes de hardware

Se va a estimar el precio del prototipo empleado para la realización de pruebas a lo largo del proyecto. En este caso el cálculo de costos se ha limitado al único dispositivo que todavía requiere mejoras pero, si se obtuviera un producto final susceptible de lanzamiento al mercado, además del cálculo del coste por unidad habría que realizar una estimación de la cantidad necesaria de cada material para la producción del número de dispositivos que se pretenda distribuir.

Tras la búsqueda y comparación de precios para los componentes necesarios en las tiendas online de Amazon, turiBOT y AliExpress, se concluye que AliExpress ofrece los mejores precios. Sin embargo, debido a los problemas reportados con su distribuidora en España, se ha descartado esta opción. Los precios de Amazon [Amazon,] y turiBOT [turiBOT,] son bastante similares, por lo que se ha optado por elaborar el presupuesto considerando a ambas empresas. Los resultados se pueden visualizar en la Tabla A.2.

A.4. Viabilidad legal

En el desarrollo de este proyecto tecnológico, es crucial considerar su viabilidad legal para garantizar no solo la protección del proceso y del producto intelectual, sino también el cumplimiento de las regulaciones vigentes.

Leyes y normativas

La normativa de interés en el desarrollo de un proyecto como el que se presenta, debe abarcar aspectos como la comercialización y derechos de los usuarios. Su cumplimiento va a asegurar la calidad y seguridad del producto y su desarrollo, fortaleciendo la confianza de todas las partes involucradas.

Componente	Coste (€)
Microprocesador Arduino UNO R3	24,68
Cables	1,57
Acelerómetro + Giroscopio MPU-6050	3,92
Módulo bluetooth HC-05	10,00
Display LCD 16x2	7,5
Módulo I2C	2,66
2 pulsadores	0,50
2 resistencias	0,50
Caja para prototipos	7,00
Batería recargable 9V	10,00
Conector para batería	0,35
Cable multihilo flexible	11,99
Interruptor	1,19
Proto shield Arduino	3,19
Conector 5 pines macho-hembra	6,99
Coste Total	92,04 €

Tabla A.2: Desglose de costes de los componentes

Considerando el marco legislativo europeo y español, se procede a describir los principales decretos y leyes que afectan al proyecto.

Protección de datos personales y respeto de la privacidad

Los datos personales deben ser protegidos debido a su valor e importancia, respetando siempre el derecho de las personas a la transparencia en el tratamiento de sus datos. En este caso se trabaja con datos que podrían ser considerados sanitarios, lo que les confiere un carácter más sensible. Aunque hasta el momento no se han empleado datos reales de pacientes ni realizado evaluaciones sobre su evolución, en el futuro las siguientes leyes marcarán cómo se debe realizar dicho tratamiento.

- Reglamento (UE) 2016/679 (Reglamento general de protección de datos, RGPD), aprobado por el Parlamento y Consejo Europeo, establece el marco legal para el desarrollo del tratamiento y la libre circulación de los datos de personas físicas, a las que reserva el derecho de acceso, rectificación y eliminación de sus datos [BOE, 2016].
- Ley Orgánica 3/2018 de Protección de Datos Personales y garantía de los derechos digitales (LOPDGDD). Esta ley complementa el RGPD en

España estableciendo y regulando aspectos que este último no llegaba a concretar, ya que delegaba la oportunidad de hacerlo a cada Estado miembro, como es el caso de los datos en el ámbito digital[BOE, 2018] [BOE, 2016].

- La Ley 41/2002 básica reguladora de la autonomía del paciente y de derechos y obligaciones en materia de información y documentación clínica [BOE, 2002], defiende los derechos básicos de los pacientes, entre los que se encuentra la confidencialidad.

Comercialización de dispositivos médicos

La alta sensibilidad del ámbito sanitario se tiene en cuenta a través de esta normativa que garantiza la seguridad, eficacia y calidad de los productos destinados a emplearse en este campo.

- El Reglamento (UE) 2017/745 sobre los productos sanitarios [BOE, 2017] especifica toda una serie de factores a tener en cuenta para la garantizar la seguridad de los dispositivos y permite que sean los Estados miembro quienes decidan qué es considerado producto sanitario. Añade que los programas informáticos serán considerados productos sanitarios según la función a la que estén destinados. En el caso de la web que se desarrolla en este proyecto, que todavía está en una etapa muy temprana, el enfoque que se le ha dado para que esté destinada al seguimiento y adaptación del tratamiento, la convertiría en un producto sanitario.
- El marcado CE (Conformidad Europea) que indica que el producto cumple con todos los requerimientos, permite que los productos sanitarios disfruten de la libre circulación dentro de la Unión como se describe en el Reglamento 2017/745 [BOE, 2017], donde también se describe cómo debe realizarse y en qué casos.
- Real Decreto 192/2023 por el que se regulan los productos sanitarios [BOE, 2023]. Hace responsable a la Agencia Española de Medicamentos y Productos Sanitarios de decir sobre la clasificación de cada producto y de proporcionar o retirar las licencias pertinentes.

Derechos de autor y propiedad intelectual

Tanto en España como en la Unión Europea existe un conjunto de leyes y regulaciones relacionadas con los derechos de autor y propiedad intelectual aplicables a la protección de software.

- Directiva (UE) 2009/24/CE sobre la protección jurídica de programas de ordenador [BOE, 2009]. Establece que los programas de ordenador de cualquier tipo, incluso los incorporados en el hardware (como el que en este proyecto se carga en el microprocesador Arduino), están protegidos por los derechos de autor. Además, describe los criterios para determinar si se trata de una obra original entre otros aspectos relevantes.
- Real Decreto Legislativo 1/1996 por el que se aprueba el texto refundido de la Ley de Propiedad Intelectual [BOE, 1996]. En el, se definen las bases de la propiedad intelectual y se describen las características y condiciones de los derechos de autor.

Licencias de software

Las licencias de software son una especie de contrato entre el propietario y el usuario interesado, en el que se establecen los términos, cláusulas y responsabilidades que se deben cumplir para respetar los derechos de ambas partes [IEBSCHOOL, 2023].

Licencias de softwares empleados en este proyecto

Se ha hecho uso de una amplia gama de herramientas, todas ellas protegidas con algún tipo de licencia que habría que tener en cuenta a la hora de la distribución de nuestro proyecto. En la tabla A.3 se muestran, agrupadas en categorías, todas las herramientas software junto a sus respectivas licencias.

Licencia de este proyecto

Para este proyecto se ha seleccionado la Licencia Apache 2.0. Esta licencia de software libre y código abierto, que destaca por las escasas restricciones, no proporciona ningún tipo de garantía ni establece restricciones de uso [Apache,]. No obstante, su cumplimiento de los requisitos mínimos y que no supone ningún coste adicional, han sido razones suficientes para su selección.

Categoría	Herramienta	Tipo de Licencia
IDEs	Arduino IDE Visual Studio Code	GLP/LGPL MIT License
Gestión Bases de Datos	MySQL, XAMPP	GPL
Librerías Arduino	I2Cdev, MPU6050	MIT License
	Wire, SoftwareSerial LiquidCrystal_I2C	LGPL/GPL Libre
Paquetes Python	PySerial Requests	BSD License Apache License 2.0
Paquetes Node.js	Express, MySQL, Body-Parser, CORS	MIT License
Diseño	Draw.io Balsamiq Wireframes	Apache License 2.0 Licencia Propietario

Tabla A.3: Herramientas empleadas y sus licencias

Apéndice *B*

Documentación de usuario

B.1. Requisitos software y hardware para ejecutar el proyecto.

Software

El desarrollo de la página web en modo localhost, en lugar de alojarla en un servidor externo, genera unos requerimientos específicos de software. Cumplir con estos requisitos es indispensable para el correcto funcionamiento de la página, asegurando el acceso a esta y permitiendo la comunicación con la base de datos y el Bluetooth del dispositivo hardware.

Descripción de los elementos software necesarios y sus respectivas funciones:

- XAMPP. Es un paquete de software que a través de su panel de control va a permitir iniciar Apache (servidor web) y MySQL (servidor de bases de datos).
- Node.js. Servidor que se inicia desde los comandos de cmd y forma parte del backend para el control de la comunicación entre la web, Arduino y la base de datos.
- Scripts y archivos de *Web_VisualStudio*. Es imprescindible que estén localizados dentro de la carpeta htdocs de XAMPP.

- Visual Studio Code. Para ejecutar el archivo `/bluetooth/ArduinoBridge/bridge.py`, puente para la comunicación entre el dispositivo Arduino y el servidor web.

Además del software que permite el funcionamiento de la web, es indispensable cargar el script correcto en el microprocesador Arduino:

- `version 4.0/v4.0_solicitudBD.ino`. Una vez el código ha sido compilado y almacenado en el microprocesador no será necesario realizar esta operación más veces, es decir, el usuario final no debería realizar este proceso ya que el hardware estaría configurado de fábrica.
- Arduino IDE. Necesario para realizar la subida del script.
- Antes de cualquier uso del prototipo hay que cargar y ejecutar el archivo `MPU6050-lcd16_ic2/MPU6050-lcd16_ic2.ino` para su calibración [González, 2023]. Esta acción debería realizarse en fábrica, antes de la distribución al usuario final.

Para el correcto funcionamiento de todos los softwares mencionados, se requiere la instalación de una variedad de paquetes y librerías de Arduino, python y Node.js.

- Librerías de Arduino: I2Cdev, MPU6050, Wire, SoftwareSerial y LiquidCrystal_I2C.
- Paquetes Python: PySerial, Requests.
- Paquetes Node.js: Express, MySQL, Body-Parser y CORS.

Se procede a la descripción detallada mediante tablas, de los requisitos funcionales específicos del proyecto software que se desarrolla. Van desde la Tabla RF-01 (Tabla B.1) hasta la Tabla RF-08 (Tabla B.8) e incluyen información del funcionamiento de la web y las interacciones con los usuarios.

Hardware

El funcionamiento correcto de todo el proyecto requiere del dispositivo físico adecuado que trabaje en concordancia con el software desarrollado, fin con el que se han realizado cambios y adiciones de componentes sobre el hardware del prototipo inicial. En la Tabla B.9 se muestran todos los componentes empleados.

RF-01	Iniciar sesión
Descripción	Todos los usuarios deben introducir de forma obligatoria su correo electrónico, tipo de usuario y contraseña para poder acceder a la página web.
Importancia	Baja

Tabla B.1: RF-01 Iniciar Sesión

RF-02	Consultar pacientes y usuarios
Descripción	Otorgar acceso a la lista completa de usuarios o pacientes, según los permisos asignados al usuario, y permitir la realización de búsquedas específicas dentro de ella.
Importancia	Media

Tabla B.2: RF-02 Consultar pacientes y usuarios

RF-03	Gestionar pacientes y usuarios
Descripción	Permitir la creación y eliminación de cuentas, así como la modificación de los datos almacenados en las cuentas de pacientes y médicos. La capacidad para realizar estas acciones depende del nivel de acceso que el usuario tenga en el sistema web.
Importancia	Media

Tabla B.3: RF-03 Gestionar pacientes y usuarios

RF-04	Realizar actividad
Descripción	Ofrece las opciones de iniciar y finalizar actividades, así como la opción de guardar o descartar estas mismas.
Importancia	Alta

Tabla B.4: RF-04 Realizar actividad

RF-05	Mostrar actividades
Descripción	Presenta al usuario en una lista las actividades realizadas por el paciente, permitiendo diferentes visualizaciones y llevar a cabo filtrados.
Importancia	Alta

Tabla B.5: RF-05 Mostrar actividades

RF-06	Consultar estadísticas
Descripción	Visualización de los datos relacionados con las actividades realizadas por el paciente, ya sea de una actividad en concreto o de todas en conjunto.
Importancia	Media

Tabla B.6: RF-06 Consultar Estadísticas

RF-07	Gestionar cuenta
Descripción	Facilitar a los usuarios las tareas de cambio de contraseña y actualización del correo electrónico vinculado a su cuenta.
Importancia	Baja

Tabla B.7: RF-07 Gestionar cuenta

RF-08	Cerrar sesión
Descripción	Todos los usuarios tendrán la opción de cerrar sesión desde el menú de inicio. Para prevenir cierres accidentales, se solicitará una confirmación de la acción antes de que esta se complete.
Importancia	Baja

Tabla B.8: RF-08 Cerrar sesión

Componentes	
Arduino UNO R3	2 Resistencias
Display LCD + Interfaz I2C	2 Pulsadores
Bluetooth HC-05	Sensor MPU-6050
Cables	Caja para Prototipos
Batería Recargable 9V + Conector de Batería	Cable Multihilo Flexible
Proto Shield Arduino	Interruptor
Conector 5 pines macho-hembra	

Tabla B.9: Componentes del Prototipo

Los requisitos hardware se basan en el principal elemento que da pie a este proyecto, el Bluetooth, y en la necesidad de implementar mejoras para lograr una mayor facilidad de uso y comodidad. Su descripción se incluye en las tablas RF-09 (Tabla B.10) a RF-12 (Tabla B.13).

B.2. Instalación / Puesta en marcha

Esta sección proporciona una guía detallada para que cualquier usuario pueda instalar y comenzar a utilizar el producto final de este proyecto. Se explican los procedimientos necesarios para la preparación del dispositivo y el despliegue de la web.

Antes de iniciar cualquier proceso detallado en esta sección, es imprescindible tener instalados y configurados correctamente los siguientes programas software con sus respectivas librerías y paquetes: VS Code, Node.js, XAMPP y Arduino IDE. Para más información, consultar el Anexo B.1. *Requisitos software y hardware para ejecutar el proyecto* y el apartado 4.2. *Técnicas y herramientas* de la memoria.

Configuración del dispositivo

1. Proceso de calibración. Abrir el archivo *Arduino/version 4.0/v4.0_solitudBD/v4.0_solitudBD.ino* y cargar el programa en la placa Arduino. Una vez completada la carga, se debe abrir el monitor serial, asegurándose que esté configurado para una velocidad de transmisión de 115200 baudios. Para completar con éxito el proceso de calibración, el usuario debe seguir todas las instrucciones que aparecen en el monitor serial. La Figura B.1 muestra el ejemplo de un proceso de

RF-09	Transmisión de datos sin cables
Descripción	Implementación de la tecnología de comunicación escogida para que el envío de datos del dispositivo no dependa de la conexión por USB al ordenador.
Importancia	Alta

Tabla B.10: RF-09 Transmisión de datos sin cables

RF-10	Alimentación externa
Descripción	Permitir la completa autonomía del prototipo incorporando en él una batería externa recargable.
Importancia	Alta

Tabla B.11: RF-10 Alimentación externa

RF-11	Encendido y apagado del sistema
Descripción	Regulación del flujo eléctrico que recibe el sistema. Se puede lograr de forma sencilla a través de un interruptor.
Importancia	Media

Tabla B.12: RF-11 Encendido y apagado del sistema

RF-12	Seguridad y manejabilidad del prototipo
Descripción	Es imprescindible, tanto para la protección del prototipo como para la seguridad del usuario, que el hardware y el montaje electrónico se encuentren protegidos y aislados. La solución que se implemente debe asegurar un manejo sencillo del dispositivo.
Importancia	Alta

Tabla B.13: RF-12 Seguridad y manejabilidad del prototipo

calibración adecuado, en el que los offsets presentan valores similares a la referencia.

```

COM42
Enviar cualquier caracter.
Enviar cualquier caracter.

MPU6050 Calibration Sketch

El sensor MPU6050 debe estar en posicion horizontal, con las letras hacia arriba.
No tocar hasta que no se muestre el mensaje de fin.

MPU6050 connection successful

Leyendo el sensor ...

Calculando offsets...
...
...
...
...
...

FINISHED!

Lecturas del sensor con offsets:      -7      -1      16374      0      0      0
Tus offsets:      633      -5399      1721      54      -22      2

Se muestra: accelX accelY accelZ giroX giroY giroZ
Comprueba que las lecturas presentan valores aproximados a: 0 0 16384 0 0 0

Autoscroll [ ] Mostrar marca temporal  Ambos NL & CR  115200 baudio  Limpiar salida

```

Figura B.1: Proceso de calibrado.

2. Carga del programa principal. Realizar la subida del archivo *Arduino/version 4.0/v4.0_solicitudBD/v4.0_solicitudBD.ino* desde Arduino IDE a la placa de Arduino.

Despliegue de la aplicación web en entorno local

1. Acciones necesarias con XAMPP. Únicamente es necesario seguir los pasos a continuación descritos la primera vez que se descargan los archivos y se quiere desplegar la web.
 - a) Incluir la carpeta *Web_VisualStudio/* en el directorio *xampp/htdocs* del equipo. La localización de dicho directorio puede variar según

lo seleccionado durante el proceso de instalación. El repositorio de [GitHub](#) del proyecto contiene el archivo *Web_VisualStudio.zip* para facilitar la descarga.

- b) Iniciar Apache y MySQL desde XAMPP Control Panel (Figura B.2). Este paso sí se va a tener que realizar siempre que se quiera utilizar la plataforma web.

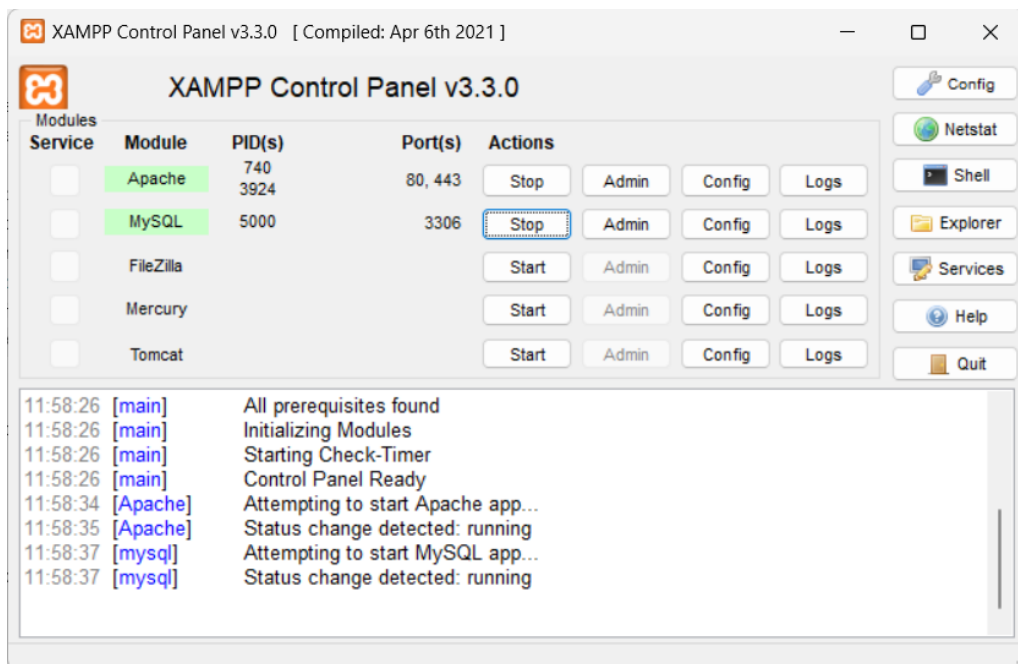
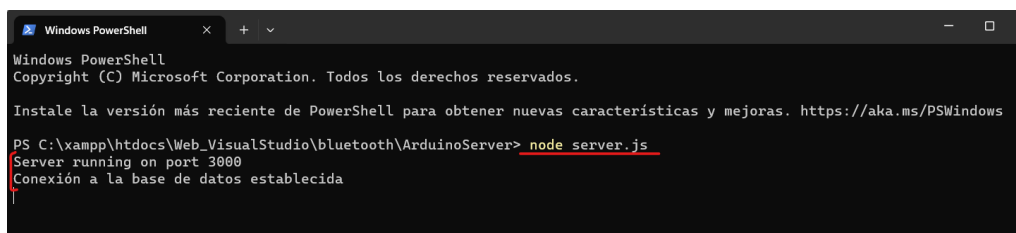


Figura B.2: XAMPP Control Panel.

- c) Crear la base de datos. Para realizar este proceso se debe navegar a la dirección *localhost/phpmyadmin*, crear una nueva base de datos bajo el nombre 'WebParkinson' e importar el archivo *DataBase/DataBase.sql*.
2. Iniciar el servidor. El usuario debe dirigirse al directorio *bluetooth/Ard-duinoServer/* y, una vez entre en él, debe abrir el terminal de comandos (cmd) haciendo clic derecho y seleccionando la opción correspondiente. Dentro del terminal, ejecutar el comando 'node server.js'. Si el proceso se realiza correctamente, aparecerá un mensaje similar al mostrado en la Figura B.3. En caso de que no se haya instalado MySQL en el servidor, se mostrará un mensaje de error y se deberá proceder a ejecutar el comando 'npm install mysql', tal y como se indica en la Figura B.4.



```

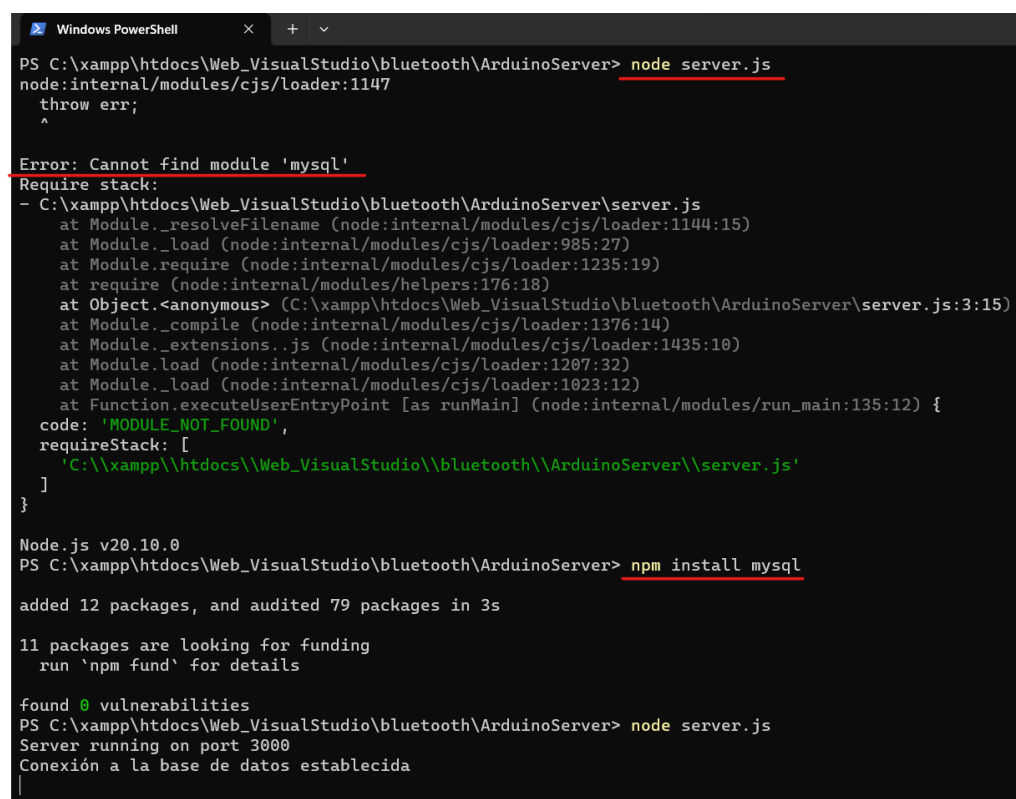
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PSWindows

PS C:\xampp\htdocs\Web_VisualStudio\bluetooth\ArduinoServer> node server.js
Server running on port 3000
Conexión a la base de datos establecida

```

Figura B.3: Éxito en la inicialización del servidor.



```

Windows PowerShell

PS C:\xampp\htdocs\Web_VisualStudio\bluetooth\ArduinoServer> node server.js
node:internal/modules/cjs/loader:1147
  throw err;
  ^

Error: Cannot find module 'mysql'
Require stack:
- C:\xampp\htdocs\Web_VisualStudio\bluetooth\ArduinoServer\server.js
    at Module._resolveFilename (node:internal/modules/cjs/loader:1144:15)
    at Module._load (node:internal/modules/cjs/loader:985:27)
    at Module.require (node:internal/modules/cjs/loader:1235:19)
    at require (node:internal/modules/helpers:176:18)
    at Object.<anonymous> (C:\xampp\htdocs\Web_VisualStudio\bluetooth\ArduinoServer\server.js:3:15)
    at Module._compile (node:internal/modules/cjs/loader:1376:14)
    at Module._extensions..js (node:internal/modules/cjs/loader:1435:10)
    at Module.load (node:internal/modules/cjs/loader:1207:32)
    at Module._load (node:internal/modules/cjs/loader:1023:12)
    at Function.executeUserEntryPoint [as runMain] (node:internal/modules/run_main:135:12) {
  code: 'MODULE_NOT_FOUND',
  requireStack: [
    'C:\\xampp\\htdocs\\Web_VisualStudio\\bluetooth\\ArduinoServer\\server.js'
  ]
}

Node.js v20.10.0
PS C:\xampp\htdocs\Web_VisualStudio\bluetooth\ArduinoServer> npm install mysql

added 12 packages, and audited 79 packages in 3s

11 packages are looking for funding
  run 'npm fund' for details

found 0 vulnerabilities
PS C:\xampp\htdocs\Web_VisualStudio\bluetooth\ArduinoServer> node server.js
Server running on port 3000
Conexión a la base de datos establecida

```

Figura B.4: Instalación de MySQL en el servidor.

3. Ejecutar *bluetooth/ArduinoBridge/bridge.py*. La ejecución se debe realizar desde VS Code. Es importante verificar siempre el número del puerto COM asignado al Bluetooth, porque puede variar y entonces será necesario modificar el script (Figura B.5). En el 'Administrador de dispositivos' de Windows se puede observar el número de puerto.

```
C: > xampp > htdocs > Web_VisualStudio > bluetooth > ArduinoBridge > bridge.py > ...
1  import serial
2  import requests
3  import time
4  import math
5
6  # Configura el puerto serie
7  ser = serial.Serial('COM6', 9600) # Ajusta el puerto COM y el baudrate
8
9  def send_command_to_arduino(command):
10     """Envía un comando al Arduino."""
11     print(f"Enviando a Arduino: {command}") # Imprime el dato a enviar
12     command_saltoLinea = command + '\n'
13     ser.write(command_saltoLinea.encode())
14
15  def send_data_to_server(endpoint, data):
16     """Envía datos al servidor."""
17     url = f'http://localhost:3000/{endpoint}'
18     try:
19         with requests.post(url, json={endpoint: data}) as response:
20             print(f"Enviado a {url}: {data}")
21     except requests.exceptions.RequestException as e:
22         print(f"Error al enviar a {url}: {e}")
23
```

Figura B.5: Fragmento del script *bridge.py* en el que se debe modificar el puerto COM.

B.3. Manuales y Demostraciones prácticas

Para proceder a la utilización del dispositivo de monitorización de la marcha junto a la página web desarrollada, se requiere haber seguido todas las instrucciones descritas en el Anexo B.2. *Instalación/Puesta en marcha*.

Manual de uso

Este apartado incluye las instrucciones necesarias para que cualquier persona pueda utilizar el producto desarrollado, incluso sin haber tenido contacto previo con él.

Preparación

El proceso descrito a continuación se realizará únicamente si se requiere el dispositivo de monitoreo para registrar los datos de una actividad de marcha.

1. Coloque de forma correcta el dispositivo y enciéndalo. El sensor debe sujetarse mediante el velcro en el tobillo izquierdo tal y como se muestra en la Figura B.6. La Figura B.7 permite ver cual es la posición del sensor dentro de la carcasa.



Figura B.6: Correcta colocación del sensor en el tobillo izquierdo.

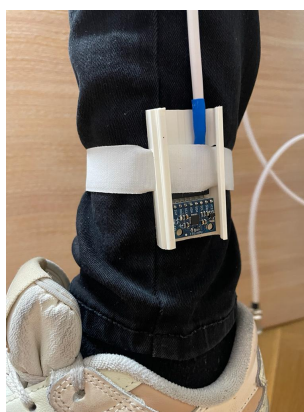


Figura B.7: Interior de la carcasa que protege al sensor MPU6050.

2. Conexión Bluetooth desde el ordenador.
3. Navegue a la dirección 'http://localhost/Web_VisualStudio/common/login.html' e inicie sesión con las credenciales de profesional o paciente.
4. Presione el botón 'Realizar Actividad' desde la ventana de visualización de la información personal del paciente. Se mostrará la pantalla de

la Figura B.8, o una similar, si el usuario registrado es un paciente. Desde aquí podrá operar de forma remota y visualizar los datos de la actividad en tiempo real.



Figura B.8: Pantalla para el inicio de una sesión de monitoreo.

Utilización

La página web ofrece distintas funcionalidades dependiendo del tipo de usuario que acceda. Tras ir a la dirección 'http://localhost/Web_VisualStudio/common/login.html' e iniciar sesión con las credenciales correspondientes, se mostrarán diferentes páginas de inicio y funcionalidades según el tipo de usuario. Las figuras B.9, B.10 y B.11 muestran capturas de pantalla de las páginas de inicio correspondientes a los usuarios de tipo administrador, profesional y paciente, respectivamente. En ellas, se pueden visualizar las principales funcionalidades para cada tipo de usuario.

Demostraciones prácticas

Con el propósito de facilitar el uso del sistema al usuario, se han realizado una serie de vídeos explicativos disponibles [aquí](#) y accesibles desde el repositorio de [GitHub](#) del proyecto.

El vídeo '[manual_Web](#)' sirve como guía al usuario para entender las funcionalidades que ofrece el sistema y cómo navegar por la web, mientras '[manual_RegistroActividad](#)' muestra un ejemplo de cómo proceder para

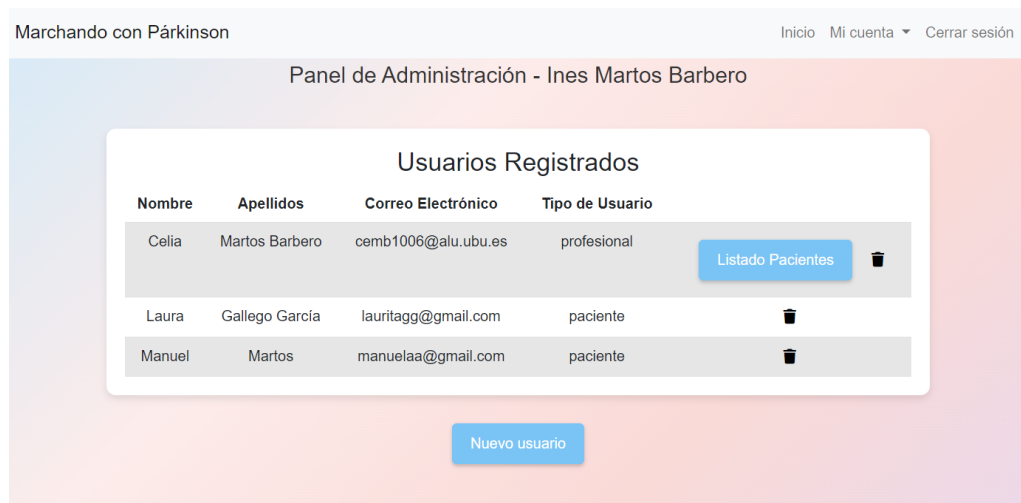


Figura B.9: Pantalla de inicio para el usuario administrador.



Figura B.10: Pantalla de inicio para el usuario profesional.

registrar los datos de monitorización de la marcha y almacenarlos para su posterior consulta.



Figura B.11: Pantalla de inicio para el usuario paciente.

Apéndice C

Manual del desarrollador / programador / investigador.

C.1. Estructura de directorios

Todos los ficheros que forman parte de este trabajo se encuentran alojados en el repositorio de GitHub disponible en [este enlace](#). En este apartado se detalla el contenido de cada uno de ellos para facilitar la comprensión y el análisis del trabajo.

- ***README.md***.
Pequeña introducción y explicación sobre el trabajo y los recursos disponibles en el repositorio de GitHub.
- ***LICENSE***.
Documento de la licencia Apache License 2.0 utilizada para este proyecto.
- ***Arduino/***.
Contiene todo el material, desde scripts hasta librerías, relacionado con el desarrollo del trabajo en lo relativo al software para la placa Arduino.
 - ***/libraries/***.
Carpeta que da acceso a algunas de librerías necesarias para el funcionamiento de los scripts de Arduino. El formato comprimido en el que se proporcionan los archivos permite su importación directa en Arduino IDE.

- *I2Cdev.zip*
- *LiquidCrystal_I2C-1.1.2.zip*
- *MPU6050.zip*
- */pruebas_bluetooth/*.
Contiene scripts empleados al inicio del proyecto para probar de forma simple las funcionalidades de envío y recibo de datos mediante Bluetooth.
 - */01_configuracion_Bluetooth-HC-05/*
01_configuracion_Bluetooth-HC-05.ino: script diseñado para establecer una comunicación bidireccional entre un Arduino y un módulo Bluetooth utilizando dos pines digitales (10 y 11), en lugar de los pines de comunicación predeterminados (TX y RX). Permite comprobar de forma sencilla el funcionamiento del módulo Bluetooth a través de su conexión con un móvil para su uso con la aplicación 'Serial Bluetooth Terminal'. Además, la sencillez del script permite trabajar con el módulo Bluetooth en modo configuración (conectando pin 'EN' a 5V) y enviar comandos AT de forma eficaz.
 - */02_enviar_a_Arduino/02_enviar_a_Arduino.ino*:
permite la comunicación entre Arduino y un dispositivo Bluetooth para controlar el LED integrado en la placa mediante de comandos específicos. Se trata de un enfoque básico que sirve como prueba inicial para futuras implementaciones donde se controlarán los estados en el script de Arduino a través de botones de una página web.
 - */03_recibir_de_Arduino/*
03_recibir_de_Arduino.ino: establece la comunicación para el envío automático de mensajes desde el Arduino al módulo HC-05. Permite comprobar con un ejemplo básico la capacidad de envío de datos de Arduino a dispositivos Bluetooth que será necesaria en el proyecto para la transmisión de datos de actividad recogidos por el sensor MPU-6050.
 - */04_envio_recibo_datos/*
04_envio_recibo_datos.ino: Combina las funcionalidades de los dos archivos anteriores (*02_enviar_a_Arduino.ino* y *03_recibir_de_Arduino.ino*).
 - */05_envioBluetooth_MPU6050/05_envioBluetooth_MPU6050.ino*: presenta pequeños cambios para imple-

mentar la funcionalidad del envío de datos a través del módulo HC-05. Estas modificaciones se realizan sobre el script */version 2.0/MPU6050-lcd16_ic2.ino*.

- ***/version 1.0/***.

Carpeta con los archivos Arduino (extensión .ino) necesarios para el funcionamiento del prototipo inicial obtenido de [González, 2023] sin ninguna modificación.

- ***/MPU6050-dmp/MPU6050-dmp.ino***: (Extra para pruebas) activa el Digital Motion Procesor (DMP) del módulo MPU-6050. Determina y muestra el número de pasos realizados por el paciente en función de los valores pitch, roll y yaw.
- ***/MPU6050-filtro/MPU6050-filtro.ino***: (Extra para pruebas) script que determina la orientación del sensor utilizando unos ángulos de inclinación y rotación calculados previamente.
- ***/MPU6050-lcd20/MPU6050-lcd20.ino***: contiene el código para operar el prototipo usando un módulo LCD 20x4 conectado a través de un módulo I2C.
- ***/MPU6050/MPU6050.ino***: similar a *MPU6050-lcd20.ino* pero adaptado para un LCD 16x2 y eliminando el uso del módulo I2C.
- ***/calibracionDMP/calibracionDMP.ino***: necesario para la calibración del DMP del módulo MPU-6050. Ajusta los offsets del acelerómetro y del giroscopio, un proceso que debe completarse en el Arduino antes de cargar el programa principal.

- ***/version 2.0/***.

Contiene los archivos Arduino con los que se trabajará a lo largo del proyecto, implementando funcionalidades nuevas en versiones siguientes según sea requerido.

- ***/MPU6050-lcd16_ic2/MPU6050-lcd16_ic2.ino***: adaptación y combinación de los scripts */version 1.0/MPU6050/MPU6050.ino* y */version 1.0/MPU6050-lcd20/MPU6050-lcd20.ino* para su funcionamiento con módulo LCD 16x2 e interfaz IC2.
- ***/calibracionDMP/calibracionDMP.ino***: no tiene modificaciones respecto al script desarrollado por [González, 2023] y que también se encuentra en las carpetas */version 1.0/* y

/version 4.0/. Este proceso de calibración del módulo MPU-6050 debe realizarse antes de su primer uso y cuando se detectan errores en las lecturas después de un uso prolongado. Tras cargar el programa en el Arduino, es en el monitor serial dónde se guía al usuario durante el proceso completo.

- */version 3.0/*
 - */v3.0_botonesHTML/v3.0_botonesHTML.ino*: proporciona al script */version 2.0/MPU6050-lcd16_ic2/MPU6050-lcd16_ic2.ino* la funcionalidad de controlar el inicio y detención de la medición del MPU-6050 mediante comandos recibidos por Bluetooth. Permite el control remoto del proceso de registro de datos de una actividad.
 - */v3.1_mostrarDatos/v3.1_mostrarDatos.ino*: modifica el script proporciona al script */version 2.0/MPU6050-lcd16_ic2/MPU6050-lcd16_ic2.ino* para que, tanto durante la realización de la actividad como tras su finalización, los datos correspondientes sean enviados por Bluetooth.
 - */v3.2_ControlBotones-y-MostrarDatos/v3.2_ControlBotones-y-MostrarDatos.ino*: combina en un solo script las funcionalidades implementadas por separado en los dos anteriores (*v3.0_botonesHTML.ino* y *v3.1_mostrarDatos.ino*). Por último, añade el número total de pasos a los datos que se envían al finalizar la actividad.
- */version 4.0/*
 - */calibracionDMP/calibracionDMP.ino*: no tiene modificaciones respecto al script desarrollado por [González, 2023]. Se incluye en la carpeta */version 4.0/* con el único propósito de agrupar en el mismo directorio todos los archivos Arduino requeridos para el funcionamiento del producto final del proyecto.
 - *v4.0_solicitudBD/v4.0_solicitudBD.ino*: es el script final del proyecto para Arduino, el que contiene la implementación de todas las funcionalidades requeridas y el que hay que cargar en la placa de Arduino UNO para el correcto funcionamiento del dispositivo en conjunto con la página web. Parte del archivo */version 3.0/v3.2_ControlBotones-y-MostrarDatos/v3.2_ControlBotones-y-MostrarDatos.ino*, sobre el que se realizan cambios en el código para que los datos

específicos del paciente como la altura y el sexo se reciban por Bluetooth (serán obtenidos de la base de datos).

■ ***Documentacion_Overleaf/***.

Carpeta que contiene todos los archivos empleados en el desarrollo de la memoria con la herramienta Overleaf. Incluye los documentos pdf de la memoria y anexos.

- ***img/***: almacena todas las imágenes que aparecen tanto en la memoria como en los anexos. Se encuentran organizadas en carpetas según su contenido y el apartado en el que han sido empleadas.
- ***tex/***: incluye todos los capítulos de la memoria y los anexos en sus respectivos documentos LaTeX.

■ ***Pruebas_Comunicación/***.

Incluye una serie de archivos y carpetas utilizados en la realización de pruebas de comunicación entre el servidor Node.js y Arduino, un proceso coordinado por el archivo *bridge.py*.

- ***/EnvioDatos_a_Arduino/***.
 - ***/conNode_html fuera/***
- ***/LecturaDatos_de_Arduino/***
 - ***/conNode_html dentro/***: el html que muestra los datos recibidos por Bluetooth desde Arduino, se localiza en la misma dirección que el servidor Node.js.
 - ***/conNode_html fuera/***: el html que muestra los datos recibidos por Bluetooth desde Arduino, se localiza en una dirección diferente a la del servidor Node.js.
 - ***/sinNode_no funciona/***: se intentó regular la comunicación entre la web y Arduino sin emplear un servidor Node.js pero no se logró llevar a cabo esta opción.

■ ***videos_demostracion.md***.

Archivo con el enlace a los vídeos que sirven de manual sobre el funcionamiento del sistema, tanto de la web como del registro de datos con el prototipo.

■ ***Web_VisualStudio.zip***.

Archivo comprimido que tiene el mismo contenido que la carpeta *Web_VisualStudio/*. Se incluye para facilitar la descarga de todos los documentos necesarios para el despliegue de la aplicación web.

■ **Web_VisualStudio/..**

Contiene, almacenados en subcarpetas, todos los archivos y scripts de código necesarios para la creación de la página web.

• **/admin/.**

Carpeta con todos los archivos requeridos únicamente para el usuario administrador.

- **crearUsuario.php**: procesa los datos enviados desde el formulario de *crearUsuarioHTML.php*. Realiza la conexión con la base de datos y, tras una serie de validaciones, inserta la información obtenida en los campos correspondientes. Si el usuario creado es del tipo 'paciente', gestiona la asignación del profesional. Por último, establece mensajes de confirmación del proceso de creación y redirige a la página del formulario.
- **crearUsuarioHTML.php**: página HTML con contenido de PHP, diseñada para mostrar el formulario de creación de usuario, cuyo diseño atractivo y responsivo se establece empleando CSS. Incluye solicitudes de validación del proceso, mensajes de retroalimentación y maneja la redirección según acciones realizadas en el proceso de creación.
- **eliminarUsuario.php**: script que gestiona la eliminación de un usuario tras la confirmación al presionar el icono de papelera. Después de validar el id del usuario que se va a eliminar, se consulta su tipo en la base de datos. Según el tipo de usuario el proceso de eliminación varía: un administrador se borra únicamente de la tabla 'usuarios', un profesional requiere una reasignación de pacientes, y un paciente acarrea la eliminación de todas las actividades asociadas.
- **inicioAdmin.php**: página web desarrollada con HTML, CSS y Bootstrap, diseñada para proporcionar una interfaz para visualizar y gestionar los usuarios registrados en el sistema. Implementa la funcionalidad de confirmación de algunas operaciones. Además de constituir el front-end, interactúa con el back-end para realizar operaciones de recuperación y gestión de datos de usuarios desde la base de datos.
- **listadoPacientes.php**: integra elementos de front-end y back-end para la visualización de pacientes asignados a un profesional específico. Por un lado se proporciona una interfaz de usuario con una tabla detallada de los pacientes y opciones

de navegación web, y por otro se realiza la conexión con la base de datos para la obtención, procesamiento y preparación de la información de aquellos pacientes relacionados con el id del profesional que se está consultando.

- ***menu.php***: componente de navegación destinado a ser incluido en otras páginas web del sistema accesibles por el usuario del tipo 'administrador'. Proporciona una barra de navegación interactiva que incluye enlaces para varias funcionalidades clave: página de inicio, opciones para gestionar la cuenta del usuario y la posibilidad de cerrar sesión.
- ***/bluetooth/***.
 - ***/ArduinoBridge/bridge.py***: script que actúa como intermediario en la comunicación bidireccional entre Arduino y el servidor Node.js. Establece una conexión serial a través de Bluetooth con el Arduino para recibir datos y enviar comandos, utilizando la función 'send_command_to_arduino(command)'. Además, realiza solicitudes HTTP para el intercambio de información con el servidor, empleando la función 'send_data_to_server(endpoint,data)'. El script mantiene una comunicación continua con una pausa mínima para evitar la sobrecarga en la transmisión de datos.
 - ***/ArduinoServer/***.
Directorio del proyecto Node.js.
 - ◇ ***/node_modules/***: carpeta que almacena todas las dependencias requeridas para el proyecto, incluye bibliotecas y frameworks instalados en Node.js.
 - ◇ ***package-lock.json***: esencial para el bloqueo de versiones, optimización de la instalación y registro de la estructura de */node_modules/*. Se genera de forma automática por Node Package Manager (npm) durante la creación del archivo *package.json*.
 - ◇ ***package.json***: creado de forma automática al iniciar el proceso de creación del proyecto, ejecutando 'npm init' desde el terminal cdm estando situado dentro de */ArduinoServer/*. Contiene configuración y dependencias necesarias para el proyecto de Node.js creado.
 - ◇ ***server.js***: archivo del servidor principal, encargado de la gestión de la aplicación web. Define rutas para solicitudes GET y POST que le permiten manejar la lógica

del servidor, la interacción con la base de datos y el procesamiento de datos de entrada/salida, almacenando los datos recibidos (datos de actividades y estados de comandos) en variables esenciales para la lógica de la aplicación.

- */common/*

Almacena una serie de archivos que implementan páginas web y funcionalidades requeridas por más de un tipo de usuario.

- ***actividad.php***: web dinámica que permite a pacientes y profesionales monitorear y controlar actividades en tiempo real a través de botones y una interfaz sencilla de entender. Muestra información relevante de la actividad en curso (los datos que está recogiendo el sensor MPU-6050) o finalizada. Integra un script de JavaScript para manejar la interacción con el servidor y actualizar la interfaz de forma adecuada.
- ***actualizarCorreo.php***: maneja el proceso de actualización del correo electrónico. Tras recibir los datos del formulario, valida la información y, si es correcta, actualiza el correo del usuario en la base de datos. Si la actualización es exitosa, informa al usuario y lo redirige a la página de inicio correspondiente. En caso de errores y discrepancias, informa del problema al usuario y permanece en el formulario.
- ***actualizarCorreoHTML.php***: página web que proporciona un formulario de diseño atractivo y responsivo para que el usuario actualice el correo electrónico asociado a su cuenta. Dependiendo del tipo de usuario, muestra diferentes barras de navegación. Al presionar el botón 'Aplicar Cambios', se activa la función 'confirmarAcción' de JavaScript.
- ***cambiarContraseña.php***: gestiona la actualización de contraseña del usuario. Comprueba que los campos del formulario se hayan rellenado y contengan la información correcta. Si los datos se verifican, procede a la actualización de la contraseña en la base de datos, informando al usuario en cada paso del proceso.
- ***cambiarContraseña.php***: web que presenta un formulario para que el usuario introduzca los datos requeridos para el cambio de contraseña. Al seleccionar la opción 'Aplicar Cambios', se activa la función 'confirmarAcción' de JavaScript para procesar la solicitud.

- ***consultaActividades.php***: proporciona, tanto a profesionales como a pacientes, una interfaz para visualizar las actividades realizadas por cada paciente y unas estadísticas básicas. Tras establecer conexión con la base de datos recupera y muestra las actividades del paciente seleccionado en una tabla detallada, llevando a cabo el cálculo de estadísticas globales como la media de bloqueos, velocidad media, número de pasos y duración media de la actividad.
- ***eliminarCuenta.php***: lleva a cabo la eliminación de cuentas de usuario en la web. A partir de la obtención del id del usuario que quiere eliminar su propia cuenta, procede a la eliminación llevando a cabo las operaciones requeridas según el tipo de usuario. Al finalizar se redirige al usuario al cierre de sesión (*logout.php*).
- ***login.html***: página desarrollada como front-end para proporcionar al usuario el formulario de inicio de sesión para la web. Incluye campos para el correo electrónico, contraseña y un desplegable para la selección del tipo de usuario. Utiliza estilos personalizados y emplea herramientas como HTML, CSS y JavaScript.
- ***login.php***: manejo, desde el back-end, del proceso de inicio de sesión. Realiza la autenticación de las credenciales introducidas en el formulario de inicio de sesión. Si los datos resultan correctos, almacena la información del usuario en la sesión y lo redirige a la página de inicio correspondiente según su rol.
- ***logout.php***: control del proceso de cierre de sesión en la web. Al confirmar el usuario que quiere cerrar sesión se va a ejecutar este script para iniciar la sesión activa, eliminar todas las variables de sesión y destruir la sesión. Finalmente, redirige al usuario a la página de inicio de sesión (*login.html*).
- ***/database/DataBase.sql***: script de SQL para la creación y configuración de la base de datos 'WebParkinson', facilitando su importación a phpMyAdmin. Se encarga de estructurar las tablas 'actividades', 'pacientes', 'profesional_paciente' y 'usuarios', definiendo para cada una campos específicos, tipos de datos y restricciones. Se establecen claves primarias para identificación única y claves extranjeras para definir relaciones entre tablas. Además, se incluye la inserción de datos en la tabla usuarios, creando un

usuario administrador y un usuario profesional, imprescindible para el uso inicial de la web.

- **/js/confirmacion.js**: archivo de JavaScript que proporciona métodos de confirmación y redirección para diversas acciones críticas de la página web. Este script contiene tres funciones principales: *'realizarRedireccion(userType)'*, que redirige al usuario a la página de inicio correspondiente según su tipo (administrador, profesional o paciente), *'confirmarAccion(accion)'*, que muestra mensajes de confirmación para distintas acciones; y *'confirmarAccionConId(accion,id)'*, que es similar a la anterior pero se utiliza para acciones que requieren un identificador específico. Estas funciones mejoran la interactividad y la seguridad de la aplicación web.
- **/paciente/**
 Contiene scripts que serán únicamente empleados por usuarios de tipo paciente.
 - **inicioPaciente.php**: página de inicio para pacientes que muestra su información personal y opciones para iniciar actividad o consultar estadísticas anteriores. Mediante JavaScript interacciona con Arduino para proporcionar datos de altura y sexo.
 - **menu.php**: componente de navegación destinado a ser incluido en otras páginas web del sistema accesibles por el usuario del tipo 'paciente'. Proporciona una barra de navegación interactiva que incluye enlaces para varias funcionalidades clave: página de inicio, opciones para gestionar la cuenta del usuario y la posibilidad de cerrar sesión.
- **/profesional/**
 Carpeta que contiene los scripts con funcionalidades y características exclusivas del usuario de tipo profesional.
 - **asignarPaciente.php**: script que maneja la asignación de pacientes al profesional actualmente en sesión. En primer lugar recibe el id del paciente y tras su validación ejecuta la consulta SQL para realizar la asignación solicitada. Tras finalizar el proceso se proporciona un mensaje de confirmación.
 - **infoPaciente.php**: página web que muestra la información detallada de un paciente específico mediante el empleo de una cookie que recoge el id del paciente. Realiza una conexión con la página web para obtener toda la información requerida

y a través de JavaScript envía a Arduino datos los datos obtenidos de altura y sexo necesarios si se quiere iniciar una actividad. También presenta diferentes botones que redirigen a otras funcionalidades.

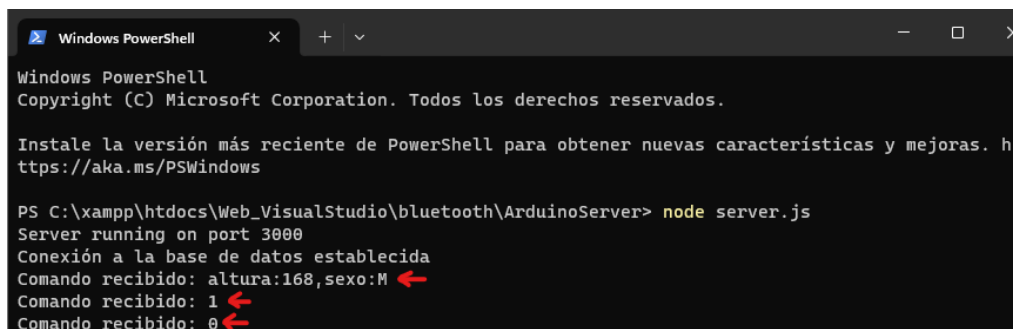
- ***inicioProfesional.php***: página de inicio para profesionales diseñada para mostrar la lista de pacientes asignados junto a funcionalidades para la consulta de información detallada, así como opciones para asignar nuevos pacientes o eliminar existentes. Se utiliza Bootstrap y CSS para un diseño claro y accesible.
- ***menu.php***: componente de navegación destinado a ser incluido en otras páginas web del sistema accesibles por el usuario del tipo 'profesional'. Proporciona una barra de navegación interactiva que incluye enlaces para varias funcionalidades clave: página de inicio, opciones para gestionar la cuenta del usuario y la posibilidad de cerrar sesión.
- ***mostrarPacientes.php***: web que permite a los profesionales visualizar una tabla de pacientes que no les pertenecen y la posibilidad de asignárselos a través de un botón interactivo que redirige a *asignarPaciente.php*.
- ***nuevoPaciente.php***: gestión del proceso de creación de nuevos pacientes. Incluye validaciones de los datos introducidos en cada campo, tras las cuales inserta al nuevo paciente en la base de datos y lo asigna al profesional cuyo id se ha almacenado. La finalización exitosa del proceso se confirma a través de una ventana emergente.
- ***nuevoPacienteHTML.php***: página web que proporciona un formulario con un diseño responsivo para incrementar los datos del nuevo paciente. Se almacena el id del profesional para que la asignación posterior se realice de forma correcta. Presenta una serie de botones que permiten cancelar o continuar con la operación en cualquier momento.
- ***quitarPaciente.php***: script de php que, tras verificar la conexión con la base de datos y el id del paciente a eliminar, elimina la relación con el profesional de la tabla 'profesional_paciente'. Además, si dicho paciente no se encuentra asignado a otros profesionales, se llevan a cabo los siguientes pasos: eliminar actividades asociadas al paciente, eliminar el registro de la tabla 'pacientes' y, eliminar el registro de

la tabla 'usuarios'. Si la acción se lleva a cabo con éxito, se notifica al usuario profesional.

C.2. Pruebas del sistema

Comprobar el correcto funcionamiento del sistema es un paso esencial para garantizar una experiencia satisfactoria para el usuario. Con este objetivo, se han incluido una serie de comandos en los scripts que gestionan la comunicación entre la base de datos, la web y el prototipo, permitiendo consultar información sobre el estado de la comunicación.

Desde el terminal CMD donde se inicializa la conexión con el servidor y la base de datos, se imprime la información que el servidor recibe de la web. En la web, cuando el usuario pulsa el botón 'realizar actividad', el sistema recibe los datos de altura y sexo. Al seleccionar 'iniciar actividad' o 'finalizar actividad', los comandos recibidos son 1 o 0. Esto se muestra en la Figura C.1.



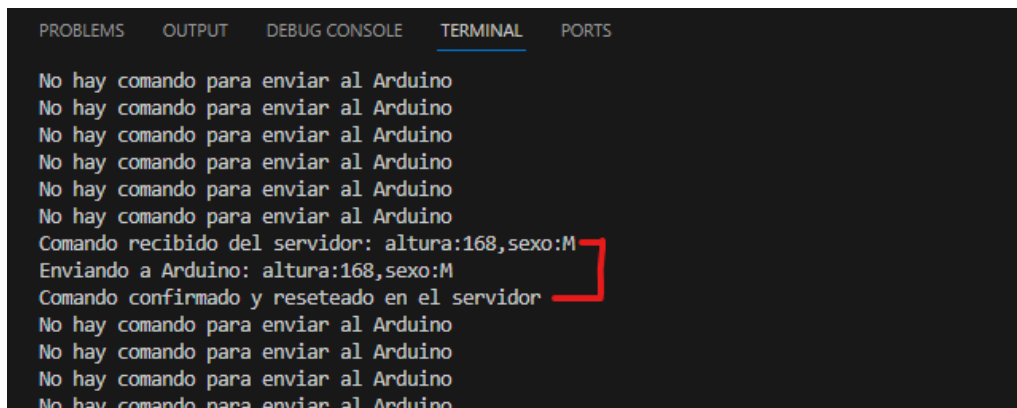
```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. h
https://aka.ms/PSWindows

PS C:\xampp\htdocs\Web_VisualStudio\bluetooth\ArduinoServer> node server.js
Server running on port 3000
Conexión a la base de datos establecida
Comando recibido: altura:168,sexo:M
Comando recibido: 1
Comando recibido: 0
```

Figura C.1: Terminal CMD muestra los datos recibidos por el servidor.

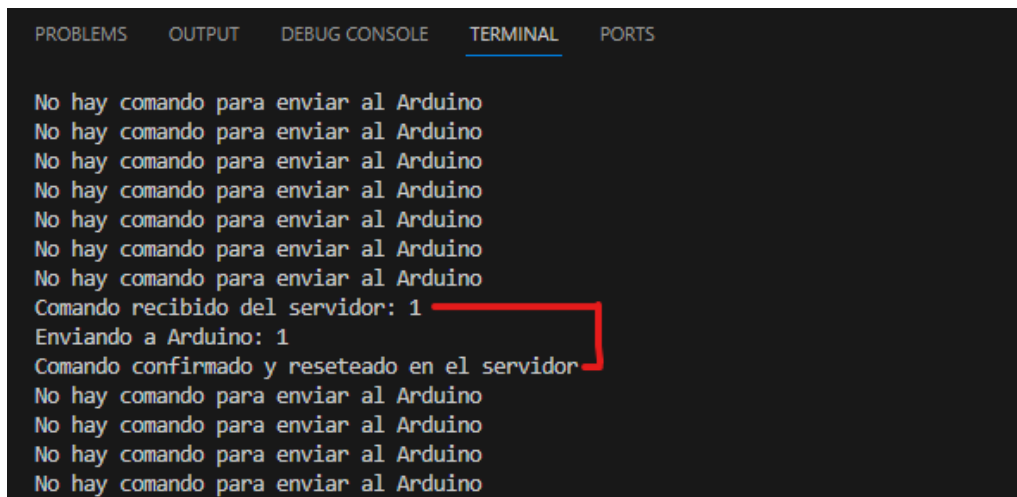
Por otro lado, el terminal de VS Code permitirá comprobar qué información se comparte entre Arduino y el servidor web. Si el funcionamiento es correcto se mostrarán por pantalla las acciones 'realizar actividad' (Figura C.2) e 'iniciar actividad' (Figura C.3) de forma similar a como sucedía en el terminal CMD. Al realizar una actividad de monitorización de la marcha, Arduino envía los datos registrados y estos son inmediatamente redirigidos al servidor, como se refleja en la Figura C.4.



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
Comando recibido del servidor: altura:168,sexo:M
Enviando a Arduino: altura:168,sexo:M
Comando confirmado y reseteado en el servidor
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
```

Figura C.2: Terminal VS Code. Acción ‘realizar actividad’.



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
Comando recibido del servidor: 1
Enviando a Arduino: 1
Comando confirmado y reseteado en el servidor
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
```

Figura C.3: Terminal VS Code. Acción ‘iniciar actividad’.

C.3. Instrucciones para la modificación o mejora del proyecto.

Este anexo comenta una serie de puntos que se deben tener en cuenta para continuar el desarrollo del proyecto software según se ha comentado en el apartado ‘*Lineas de trabajo futuras*’ de la memoria.

En cuanto a la realización de mejoras en la plataforma web, será imprescindible prestar atención a las siguientes áreas:

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Enviado a http://localhost:3000/velocidad: 1.25
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
Recibido de Arduino: contP:6.00
Enviado a http://localhost:3000/izquierda: False
Enviado a http://localhost:3000/contP: 6.0
No hay comando para enviar al Arduino
Recibido de Arduino: tiempo:12
Enviado a http://localhost:3000/izquierda: False
Enviado a http://localhost:3000/tiempo: 12.0
No hay comando para enviar al Arduino
Recibido de Arduino: velocidad:1.25
Enviado a http://localhost:3000/izquierda: False
Enviado a http://localhost:3000/velocidad: 1.25
No hay comando para enviar al Arduino
Recibido de Arduino: IZQUIERDA
Enviado a http://localhost:3000/izquierda: True
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
No hay comando para enviar al Arduino
Recibido de Arduino: IZQUIERDA
Enviado a http://localhost:3000/izquierda: True

```

Figura C.4: Terminal VS Code. Flujo de datos durante la realización de una actividad de marcha.

- Implementar protocolos de seguridad avanzados como HTTPS, medidas de protección contra ataques de inyección SQL y encriptado de contraseñas.
- Analizar la obtención de un diseño web que proporcione al usuario una experiencia más satisfactoria. Además, la plataforma desarrollada tiene un diseño responsivo que puede ser optimizado.
- Facilitar el acceso desde dispositivos externos migrando la web a un servidor en la nube. Al escoger el servidor se debe considerar la capacidad de escalabilidad que proporciona.
- Explorar la implementación de WebSockets para mejorar la comunicación en tiempo real. La plataforma actual implementa API REST como tecnología de comunicación.

- Complementar el sistema con el desarrollo de una aplicación móvil que funciona en paralelo con la plataforma web.

Respecto a las funcionalidades del sistema, sería interesante trabajar en las siguientes direcciones:

- Incrementar el procesamiento de datos para obtener estadísticas más detalladas y útiles para su uso en el ámbito clínico.
- Registrar las fechas y horas de las actividades realizadas, así como los horarios de administración de la medicación, para evaluar con mayor precisión el efecto del tratamiento farmacológico en el paciente.
- Integrar en el software el proceso de calibración del sensor que registra los datos de la actividad, posibilitando al usuario resolver posibles problemas si dicho sensor comienza a presentar errores de lectura.
- Implementar una funcionalidad para el registro de datos de actividad sin conexión, permitiendo el almacenamiento posterior de datos en la base de datos. Esto podría lograrse añadiendo una tarjeta microSD en la que se almacena la información de forma temporal.
- Realizar la combinación con un sistema de control postural, como el desarrollado por [Rodríguez, 2023]. El proyecto mencionado emplea las mismas herramientas que este, y requiere una conexión bluetooth, desarrollo de software y mejoras hardware, todo ello al mismo nivel que requería la versión previa de este proyecto. Esta integración estaría justificada por el objetivo de obtener un dispositivo de monitoreo integral de pacientes con EP que sea económicamente accesible.

Los trabajos futuros de mejoras hardware pueden centrarse en los siguientes aspectos:

- Implementar un sistema de alerta eficaz para episodios de congelación de la marcha. Se podría considerar el uso de vibraciones o alarmas sonoras, y la adición de un laser que proyecte una referencia visual en el suelo para ayudar al paciente a superar la situación.
- Mejorar la precisión y fiabilidad en la lectura de datos del sensor MPU6050, mediante el uso del procesador digital de movimiento (DMP) integrado [González, 2023].

- Ampliar el sistema añadiendo un sensor MPU6050 para la pierna derecha [González, 2023].
- Realizar cambios para mejorar la portabilidad y comodidad del prototipo, aproximándolo a un dispositivo de uso cotidiano.
- Estudiar alternativas de microcontroladores con Bluetooth integrado, podría reducir el tamaño del hardware y mejorar el procesamiento.

Apéndice D

Descripción de adquisición y tratamiento de datos

Los datos de actividad física generados y procesados por el sistema de monitorización son de fuente propia, obtenidos directamente a partir del funcionamiento del programa Arduino, y no extraídos de ninguna base de datos preexistente.

D.1. Descripción formal de los datos

En esta sección se procede a detallar la estructura, procesamiento, manejo, tipo y formato de los datos empleados en la operativa de la aplicación web.

Recopilación de datos en el script de Arduino

- **Datos del sensor MPU6050.** Obtenidos directamente del sensor a través del programa Arduino.
 - Valores del acelerómetro (ax, ay, az). Datos de tipo entero ('int') correspondientes a las mediciones del acelerómetro en los ejes X, Y, Z con un rango de -2g a +2g. 'getAcceleration(&ax, &ay, &az)' es la función encargada de la lectura de valores y la función 'contarPasos()' también emplea valores de acelerómetro para contabilizar los pasos del usuario.
 - Valores del giroscopio (gx, gy, gz). Datos de tipo entero ('int') correspondientes a las mediciones del giroscopio en los ejes X,

Y, Z con un rango de $-250^\circ/\text{seg}$ a $+250^\circ/\text{seg}$. ‘getRotation(&gx, &gy, &gz)’ es la función encargada de la lectura de valores.

- **Datos procesados de la actividad.** Empleo de variables generadas y administradas por el programa Arduino.
 - Variables de actividad. Son de tipo cuantitativo (‘float’ o ‘int’) e incluyen número de pasos (contP y Ptotal), bloqueos, tiempo de actividad (tiempo_actividad), velocidad media (velocidadMedia), frecuencia de pasos (frecuencia) y tiempo de actividad en minutos (actividadMin).
 - Variables auxiliares. Destinadas al control del tiempo, actividad y cálculo de longitud del paso. De tipo ‘unsigned long’ para variables de tiempo (tiempo1, tiempo2, tiempo), ‘float’ para cont_espera y cte e ‘int’ para altura y estado.

Datos de comunicación Bluetooth

El manejo de datos en la comunicación Bluetooth se realiza a diferentes niveles. A continuación, se procede a analizar el proceso.

Arduino

La funcionalidad del Arduino se centra en:

- **Recepción de datos:** Arduino está constantemente pendiente de la recepción de comandos a través de Bluetooth, utilizando la función ‘btSerial.available()’. La información que recibe se lee como cadena de texto (‘String’) y se almacena en la variable ‘command’ para su procesamiento. Si el Arduino recibe ‘1’ o ‘0’, inicia o detiene la actividad. En caso de recibir información con el formato ‘altura:<valor>,sexo:<valor>’, procede a extraer y procesar los valores.
- **Envío de datos:** Se envían datos cuantitativos (‘float’ e ‘int’) que están principalmente relacionados la actividad realizada. Además, se incluye información sobre el estado de la actividad para permitir la actualización correspondiente en la plataforma web cuando el control de la actividad se realiza a través de comandos hardware. Para enviar estos datos, se utiliza la función ‘btSerial.print()’.

Bridge.py

Este script de python actúa como puente entre el Arduino y el servidor web. Se encarga de:

- **Recibir datos de Arduino:** Mediante la biblioteca ‘serial’ comprueba si se están recibiendo datos en el puerto serie. Según el tipo de datos recibido lleva a cabo diferentes acciones. Al recibir ‘IZQUIERDA’ establece un valor booleano ‘true’, mientras que al recibir otro tipo de datos establece el valor ‘false’ y los procesa para obtener la información en un formato que pueda ser manejado posteriormente.
- **Enviar datos al servidor:** Tras haber recibido datos desde el Arduino y haber realizado el procesamiento correcto, se emplea la biblioteca ‘requests’ para proceder al envío de datos al servidor. Los datos pueden incluir estados de la actividad (‘detenida’ o ‘iniciada’), booleanos (flag de izquierda) o datos de actividad procesados como valores numéricos.
- **Enviar comandos a Arduino:** Recibe comandos en formato cadena de texto (‘String’) desde el servidor web y los transmite a Arduino añadiendo un salto de línea para que este los procese de forma correcta.

Server.js

Archivo de JavaScript que implementa el servidor web mediante Node.js y provee una API REST para la gestión de la comunicación y los datos en el sistema. Facilita las siguientes funciones:

1. **Recibir datos de ‘bridge.py’:** Recepción de diferentes tipos de datos y almacenamiento en las variables correspondientes. Se reciben datos numéricos (‘int’ y ‘float’) de la actividad registrada, cadenas de texto (‘String’) sobre el estado de la actividad, y booleanos almacenados en la variable ‘izquierda’ que hacen referencia a la existencia de bloqueos.
2. **Gestionar comandos para Arduino:** Almacenamiento y gestión los comandos en formato cadena de texto que tienen que ser enviados al Arduino a través de ‘bridge.py’. Incluye comandos del estado de la actividad, altura y sexo del usuario.
3. **Interacción con la base de datos:** Uso de MySQL para el almacenamiento y recuperación de datos relacionados con la actividad e información personal del paciente.

Base de datos: estructura y manejo de datos

Este apartado está destinado a la descripción de la base de datos ‘Web-Parkinson’ utilizada por la plataforma. En él, se detalla información relativa a la estructura, los datos almacenados y las relaciones establecidas. Este análisis es fundamental para comprender cómo trabajar con los datos recopilados.

La base de datos está compuesta por varias tablas, cada una de las cuales almacena información específica para cada aspecto relevante del sistema. Las relaciones entre tablas se manejan mediante las claves foráneas.

- **Tabla ‘usuarios’:** Gestiona los datos de los usuarios del sistema, incluyendo administradores, profesionales y pacientes. Contiene los siguientes campos:
 - ‘id_usuario’ (int). Clave primaria con autoincremento que asigna un identificador único a cada usuario.
 - ‘nombre’, ‘apellidos’, ‘contrasena’ (varchar). Información personal y de acceso.
 - ‘correo_electrónico’ (varchar). Clave única para asegurar que cada correo electrónico sea único en la base de datos. Necesario para el acceso a la plataforma.
 - ‘tipo_usuario’ (enum). Indica el rol del usuario en el sistema (administrador, profesional, paciente).
- **Tabla ‘pacientes’:** Almacena información específica de los pacientes. Esta información se requiere para el cálculo preciso de datos durante el análisis de la marcha. Contiene los siguientes campos:
 - ‘id_paciente’. Clave primaria y clave foránea, referenciada desde la tabla ‘usuarios’ para relacionar la información del paciente con su perfil de usuario.
 - ‘altura’ (int). Altura del paciente en centímetros.
 - ‘sexo’ (enum). Indica el sexo del paciente (M o F).
- **Tabla ‘profesional_paciente’:** Realiza las asignaciones de pacientes a profesionales, facilitando el seguimiento de los pacientes por parte de los profesionales. Contiene los siguientes campos:
 - ‘id_profesional’ (int). Clave primaria y foránea, referenciada desde la tabla ‘usuarios’.

- ‘id_paciente’ (int). Clave primaria y foránea, referenciada desde la tabla ‘pacientes’.
- **Tabla ‘actividades’**: Reúne los datos de las actividades físicas de los pacientes.
 - ‘id_actividad’ (int). Clave primaria con autoincremento que asigna un identificador único a cada actividad.
 - ‘id_paciente’ (int). Clave foránea, referenciada desde la tabla ‘pacientes’ para vincular cada actividad a un paciente.
 - ‘numero_bloqueos’, ‘numero_pasos’ (int). Número de bloqueos y total de pasos registrados durante una actividad.
 - ‘velocidad_media’ (decimal). Velocidad media de la actividad.
 - ‘duracion’ (float). Tiempo total de la actividad.

Se incluyen inserciones iniciales de usuarios con rol administrador y profesional para facilitar la puesta en marcha de la plataforma.

D.2. Descripción clínica de los datos.

Significado clínico de las métricas de actividad

Monitorizar la actividad física en personas con EP proporciona los datos para una comprensión detallada del estado físico y evolución de la enfermedad en cada paciente.

La marcha en pacientes con EP presenta un aumento en el número de pasos, debido a la disminución de su longitud, y una reducción en la velocidad, lo que se conoce como bradicinesia. Estas características están vinculadas a la rigidez y alteraciones en el control motor típicas de la EP [di Biase et al., 2020].

Otro factor relevante son los episodios de bloqueo, consistentes en la incapacidad temporal de mover los pies, que se producen en las etapas más avanzadas de la enfermedad. Estos episodios afectan gravemente a la calidad de vida de los pacientes, ya que suponen un alto riesgo de caídas. Por ello, se requiere un seguimiento exhaustivo de los bloqueos para la aplicación de un tratamiento efectivo [di Biase et al., 2020].

Las escalas de medición más comunmente utilizadas en el análisis de la marcha de pacientes con EP, son la Unified Parkinson’s Disease Rating Scale

(UPDRS) y la Movement Disorder Society-Unified Parkinson's Disease Rating Scale (MDS-UPDRS) [di Biase et al., 2020]. Esta última es una revisión de la UPDRS e incluye aspectos motores y no motores [Goetz et al., 2008].

La aplicación web almacena información relativa al número de pasos, bloqueos y velocidad de la marcha, permitiendo el seguimiento constante de estos indicadores de movilidad, clave para evaluar la evolución de la enfermedad y su impacto en las funciones motoras.

Relación con intervenciones terapéuticas

Los datos recopilados proporcionan una base objetiva para el seguimiento de los síntomas, siendo esenciales en la evaluación clínica, la toma de decisiones terapéuticas y la personalización del tratamiento. Tras una visita a la Asociación Párkinson Burgos, en la que se conversó con profesionales especializados en el manejo de esta enfermedad, se confirmó el impacto positivo de la información manejada en este proyecto a través de los siguientes aspectos:

- Evaluación y personalización de la rehabilitación. La monitorización continua permite evaluar la efectividad de las distintas terapias y adaptarlas a las necesidades concretas del paciente. Por ejemplo, ante un aumento de las situaciones de congelación de la marcha puede ser necesario implementar estrategias para prevenir caídas.
- Ajuste de la medicación. El acceso a información detallada y objetiva sobre la evolución del paciente permitirá a los neurólogos hacer los ajustes precisos en la medicación, basándose en las conclusiones extraídas sobre la eficacia del tratamiento actual.

Apéndice *E*

Manual de especificación de diseño

E.1. Planos

El montaje inicial, del que parte la idea de este proyecto con el objetivo principal de desarrollar un dispositivo completamente autónomo, se presenta en la Figura [E.1](#).

Durante el trabajo se ha implementado una fuente de alimentación externa, regulada a través de un interruptor, y un módulo Bluetooth destinado a la transmisión de datos. La instalación del módulo HC-05 requiere únicamente cuatro conexiones: conectar Vcc al polo positivo, GND a toma de tierra, y las conexiones TXD (pin de transmisión) y RXD (pin de recepción) a los correspondientes opuestos en la placa Arduino, que en este caso se han configurado como los pines digitales 10 (RX) y 11 (TX). El montaje final se refleja en la Figura [E.2](#).

El protipo final, ilustrado en la Figura [E.3](#), integra todos los componentes de hardware esenciales para facilitar su manejo.

E.2. Diseño arquitectónico

El diseño de la arquitectura web es el primer paso crítico en el diseño de software. Partiendo de los requisitos del sistema, el diseño arquitectónico proporciona una planificación y organización global de este, estableciendo las relaciones entre sus componentes [[Castro, 2015](#)].

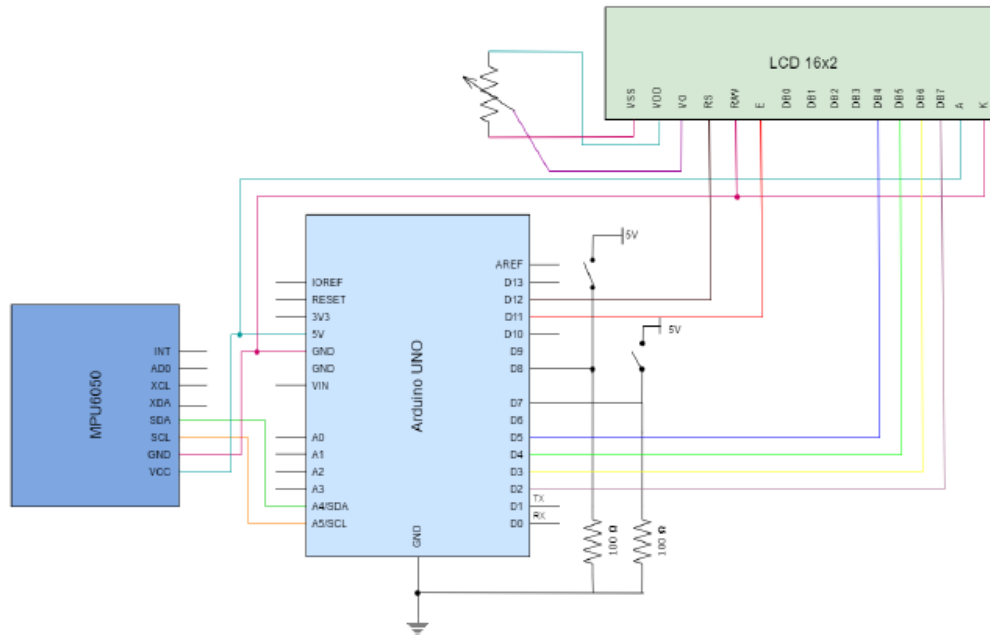


Figura E.1: Esquema de montaje del prototipo de partida. [González, 2023]

Este proceso inicia con la obtención de los requisitos funcionales, detallados en el *Anexo B.1*. Sin embargo, el paso esencial para la arquitectura web es el diseño de la experiencia de usuario. En este caso, se emplean diagramas de flujo que muestran las posibilidades del sistema para cada tipo de usuario.

Descripción del diagrama de flujo:

- Figura E.4. Muestra el proceso de iniciar sesión en la página web y las opciones que comparten los usuarios en el menú. Redirige a otras figuras diferentes donde el diagrama de flujo continúa según el tipo de usuario que haya iniciado sesión.
- Figura E.5. Recorre las opciones presentadas al administrador en el sistema.
- Figura E.6. Presenta el flujo de funcionalidades accesibles para el profesional y la relación entre ellas.
- Figura E.7. Indica la forma de navegación del paciente en la web y la forma de acceder a cada acción posible.

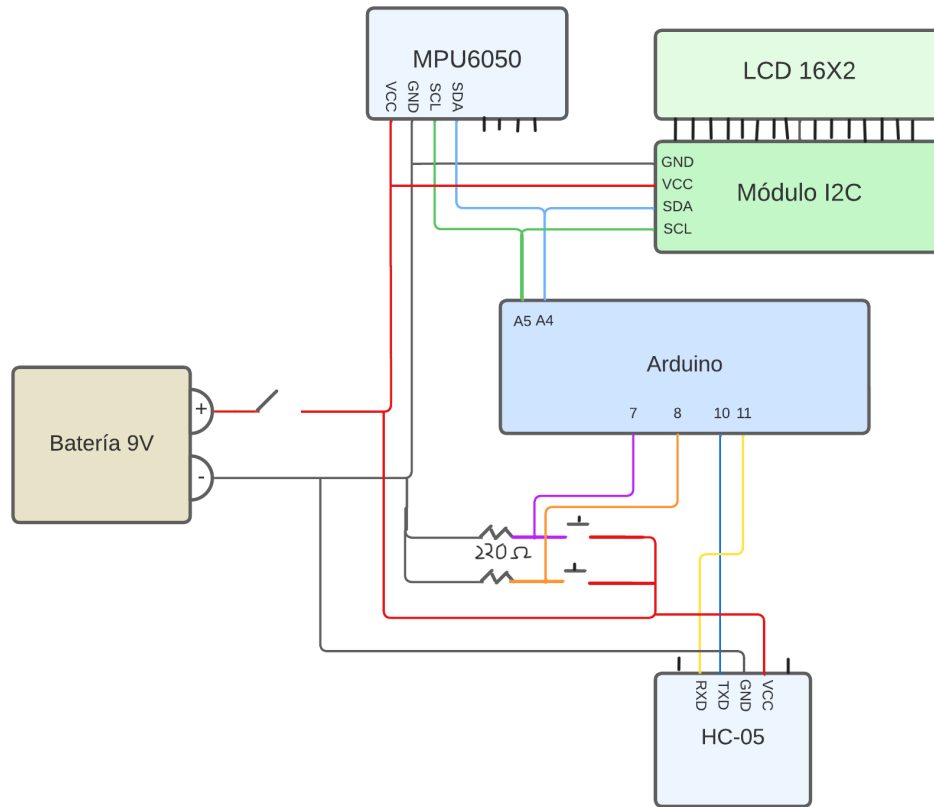


Figura E.2: Esquema de montaje del dispositivo final.

En relación con el diseño de la interfaz de usuario, llevado a cabo mediante los wireframes que se presentan y describen en el *Anexo F.3*, este se asocia principalmente con el desarrollo. Sin embargo, cabe destacar su alineación con la arquitectura web al establecer las pautas para la presentación de los elementos visuales.



Figura E.3: Hardware del dispositivo final.

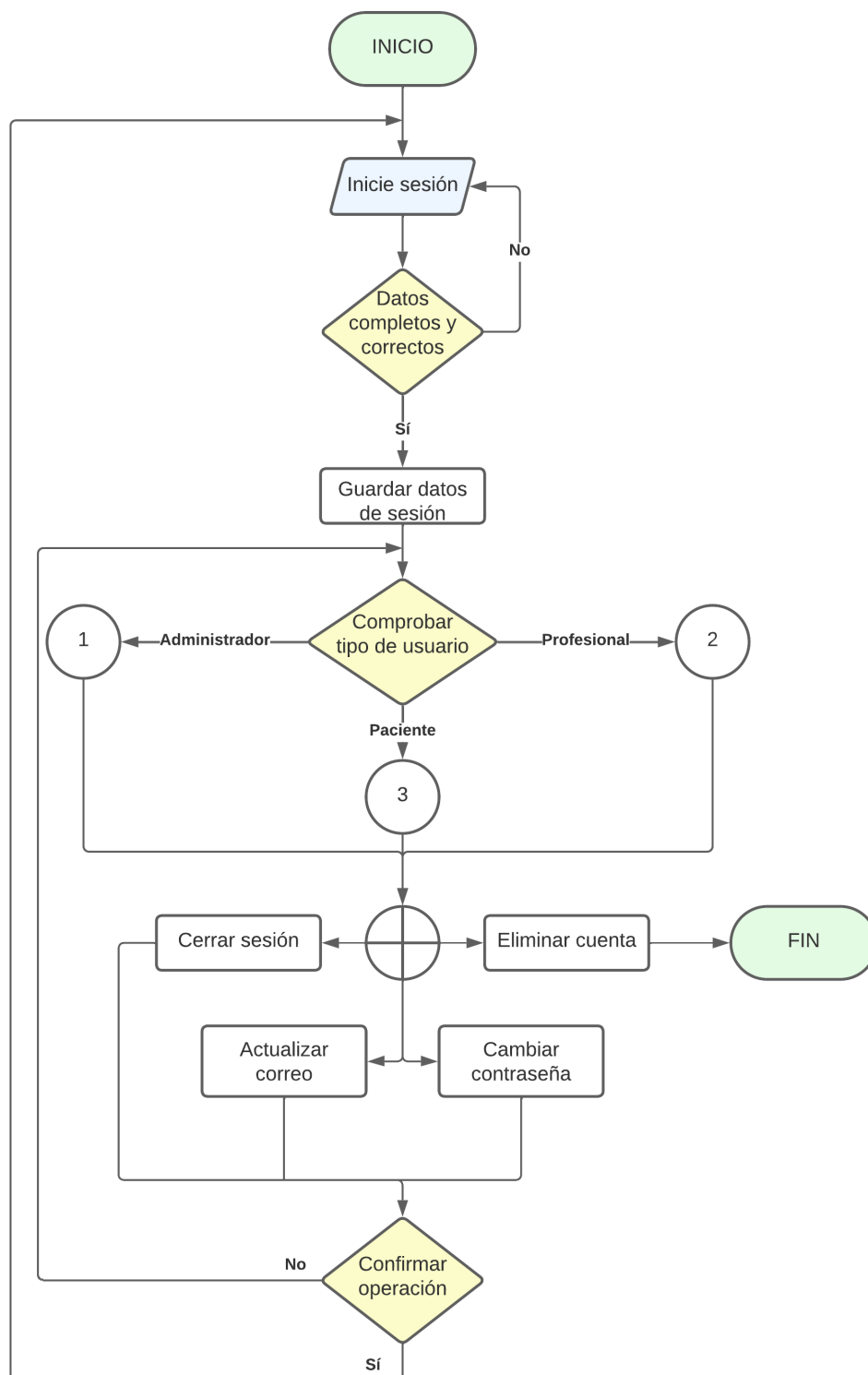


Figura E.4: Diagrama de flujo - Inicio y fin de sesión.

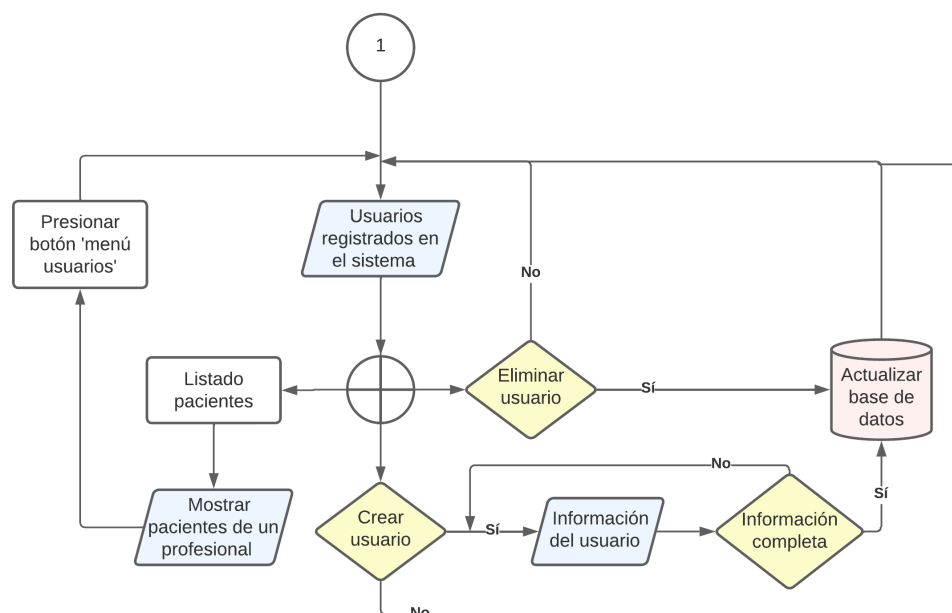


Figura E.5: Diagrama de flujo. Usuario administrador.

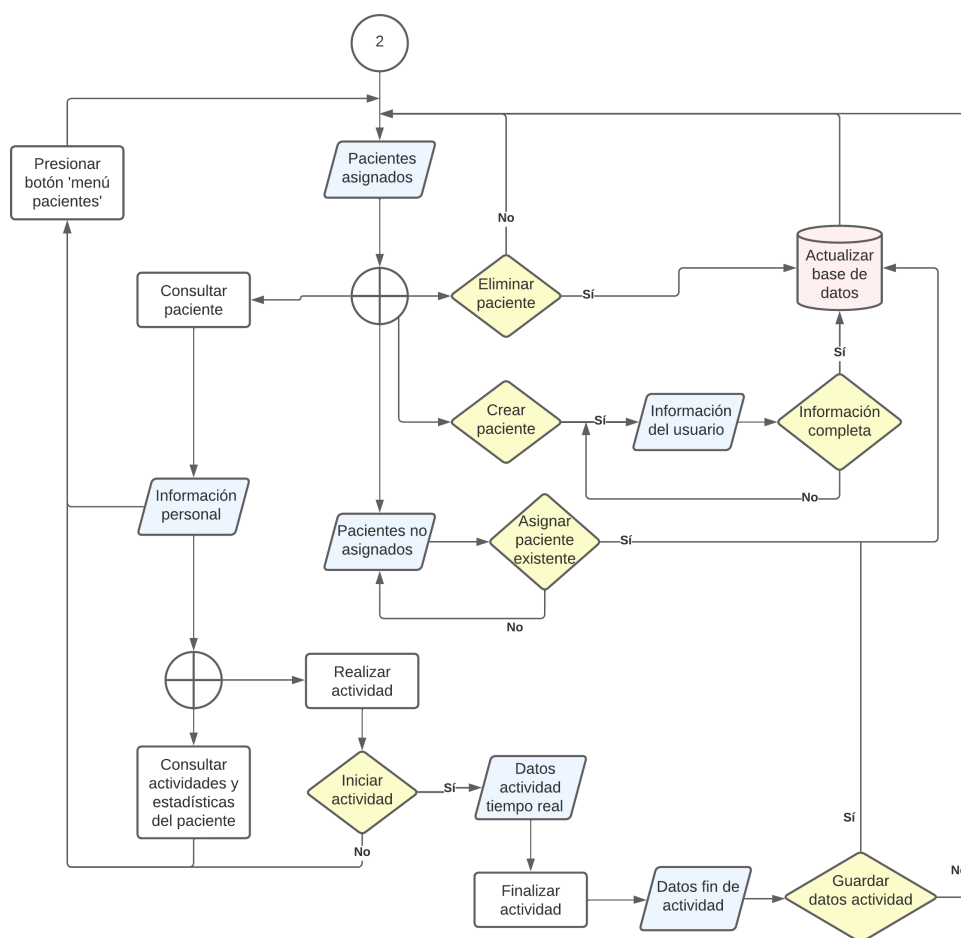


Figura E.6: Diagrama de flujo. Usuario profesional.

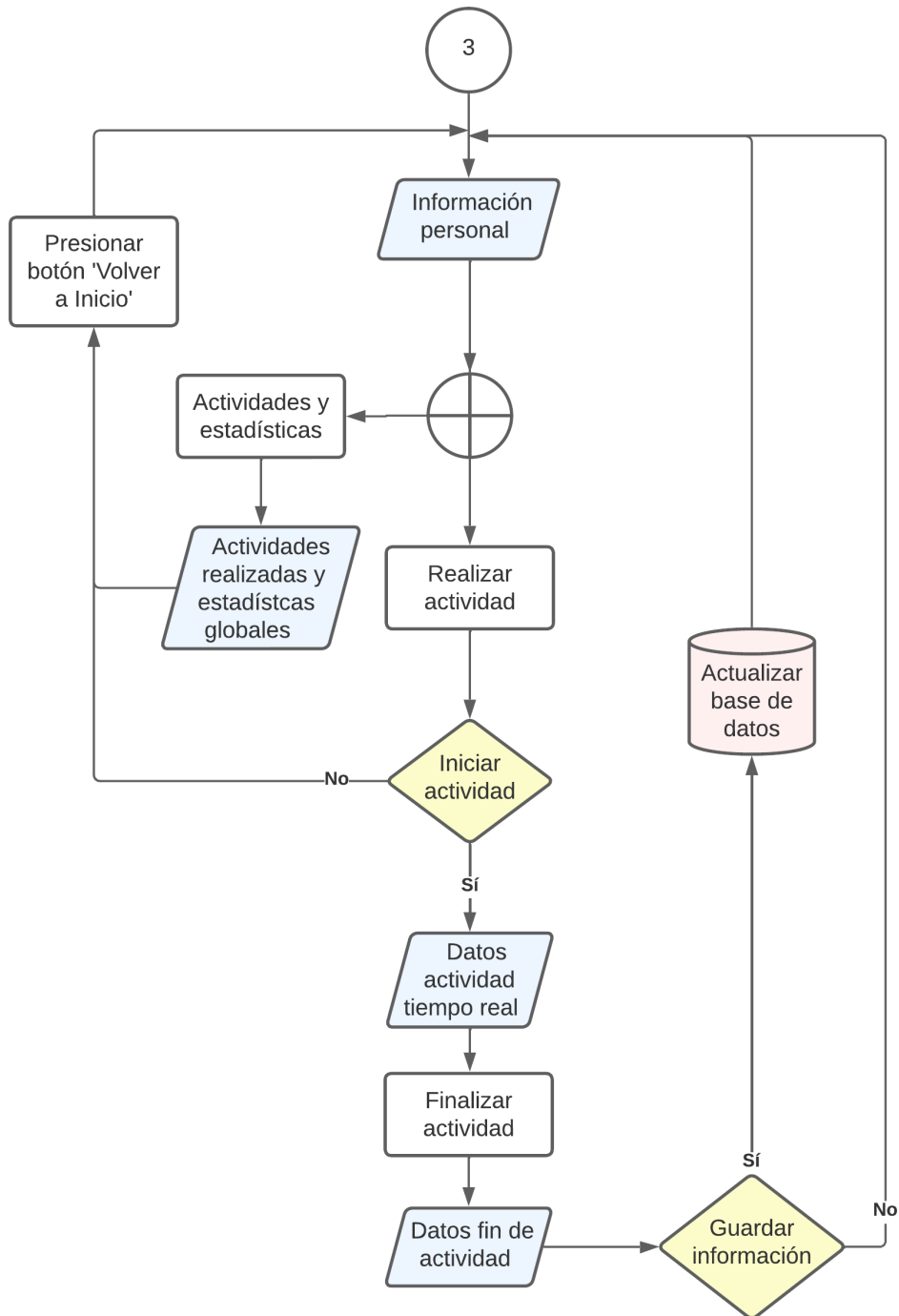


Figura E.7: Diagrama de flujo. Usuario paciente.

Apéndice *F*

Especificación de Requisitos

Un stakeholder es cualquier persona u organización interesada, afectada o implicada en el funcionamiento del software que se desarrolla [Pradel Miquel et al., 2013]. Van a desempeñar un papel clave y activo en la especificación y validación de los requerimientos del software [Oliveros et al., 2014]. Para la página web que se va a desarrollar se prevee la implicación de los tres tipos de stakeholders o usuarios:

- Paciente: usuario que está siendo tratado de EP.
- Profesional: médicos, enfermeros, fisioterapeutas, neurólogos y cualquier otro personal sanitario que sigue de cerca la evolución de la enfermedad o toma decisiones referidas al tratamiento.
- Administrador: usuario con control sobre funciones críticas del sistema no disponibles para usuarios regulares. Estas funciones incluyen crear, consultar y eliminar cualquier tipo de usuario.

Un caso de uso es una descripción de las acciones del sistema desde el punto de vista del usuario, es decir, son las funciones o servicios que el sistema proporciona a los usuarios [Mediavilla,]. La documentación de casos de uso es una herramienta empleada en la obtención de los requerimientos del sistema y requiere un paso previo de identificación de los actores o stakeholders [Cervillos, 2015]. Se han generado un total de 27 Casos de Uso que pretenden abarcar todas las funcionalidades del sistema y las posibles interacciones entre este y los stakeholders.

F.1. Diagrama de Casos de Uso

La representación de los casos de uso en diagramas permite una comprensión global de las posibles relaciones entre ellos y las opciones ofertadas por la web.

- Las funciones comunes a todos los tipos de usuarios se muestran en la Figura F.1.
- Las opciones de trabajo restringidas para el usuario administrador son las presentadas en la Figura F.2.
- La Figura F.3 refleja la relación entre las funcionalidades de pacientes y profesionales.

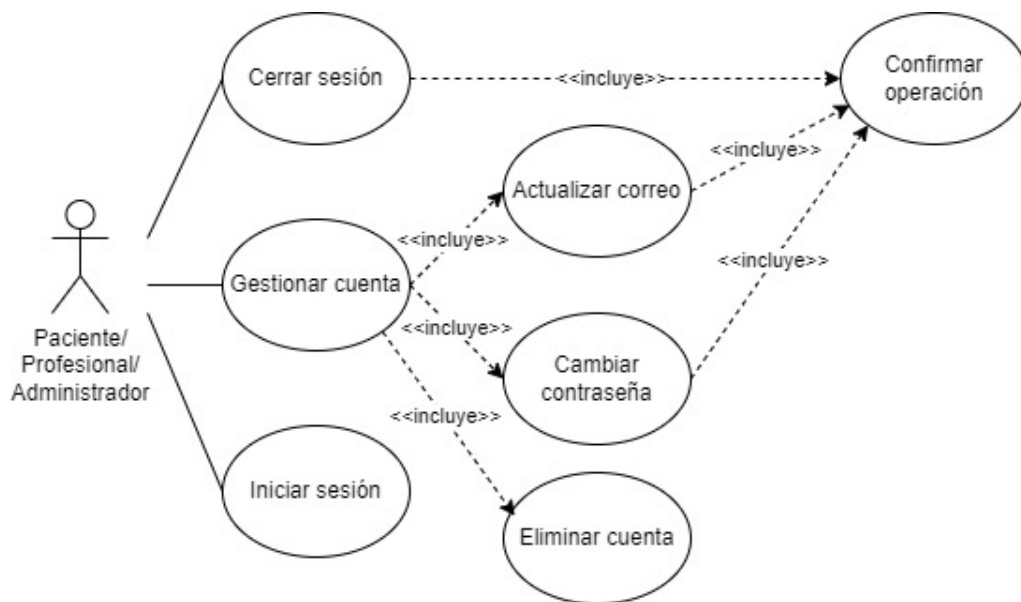


Figura F.1: CU-Administradores, Pacientes y Profesionales

F.2. Explicación Casos de Uso.

Las tablas de explicación de Casos de Uso permiten proporcionar información detallada y organizada relativa a las funcionalidades del sistema y su integración. Facilitan el manejo de excepciones y sirven de guía para el posterior proceso de desarrollo, evitando errores que podrían tener importantes repercusiones en el avance del proyecto.

CU-1	Iniciar sesión
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-01
Descripción	Acceso del usuario al sistema mediante la selección del tipo de usuario y la autenticación de correo electrónico y contraseña.
Precondición	El usuario debe estar dado de alta en el sistema.
Acciones	<ol style="list-style-type: none"> 1. Seleccionar tipo de usuario: paciente/profesional/administrador 2. Ingresar credenciales (correo electrónico y contraseña) 3. Validar credenciales
Postcondiciones	<ol style="list-style-type: none"> 1. Acceso al sistema concedido 2. Inicio de sesión registrado para propósitos de seguimiento y seguridad
Excepciones	<ol style="list-style-type: none"> 1. Tipo de usuario no seleccionado. Alerta indicando que es necesaria la selección 2. Ausencia de credenciales. El sistema pide que se completen los campos de correo electrónico y contraseña 3. Credenciales incorrectas. Mensaje de aviso y permitir hasta 3 nuevos intentos 4. Múltiples intentos fallidos. La cuenta será bloqueada un total de 5 minutos por seguridad
Importancia	Alta

Tabla F.1: CU-1 Iniciar sesión

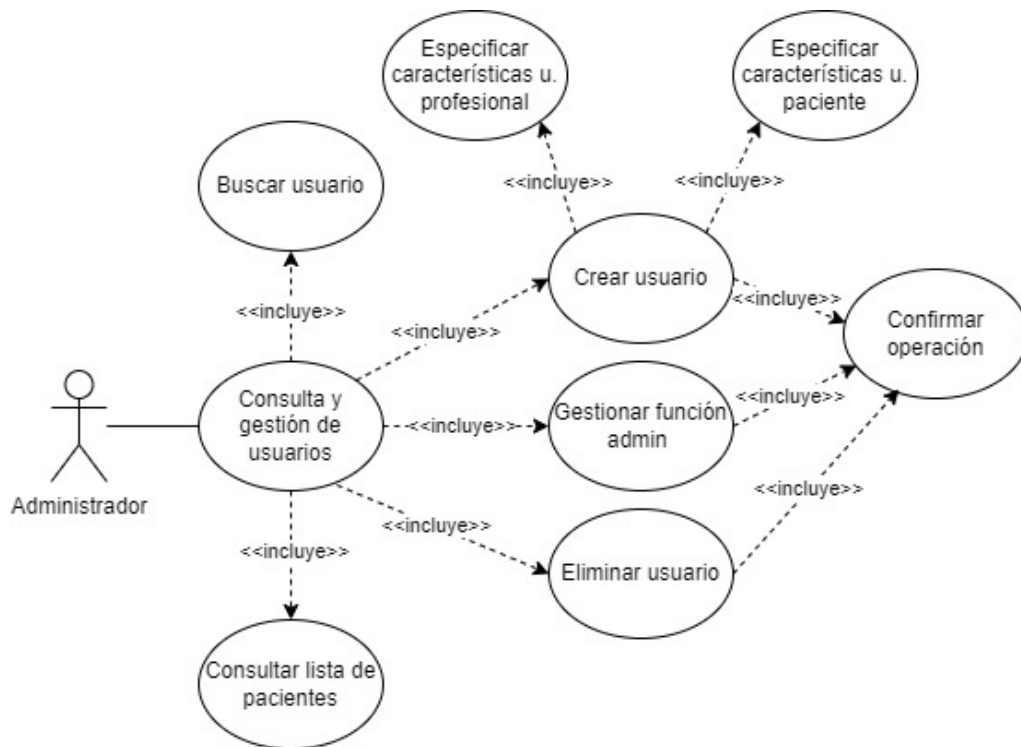


Figura F.2: CU-Administrador

F.3. Prototipos de interfaz.

Una buena práctica antes de comenzar con la programación de la página web es la elaboración de wireframes sencillos, al menos, para las principales pestañas. Esto es esencial por varios motivos:

- Proporcionan mayor claridad y comprensión de los requisitos de la web, ayudando a guiar el desarrollo posterior.
- Permiten la detección temprana de problemas de usabilidad y otorgar importancia para la experiencia del usuario.
- Mejoran la comunicación entre las personas involucradas en el proceso de desarrollo (no aplicable a este proyecto).

En este caso, se han realizado wireframes para algunas de las ventanas más relevantes, mostrando una aproximación de lo que podría ser el resultado final.

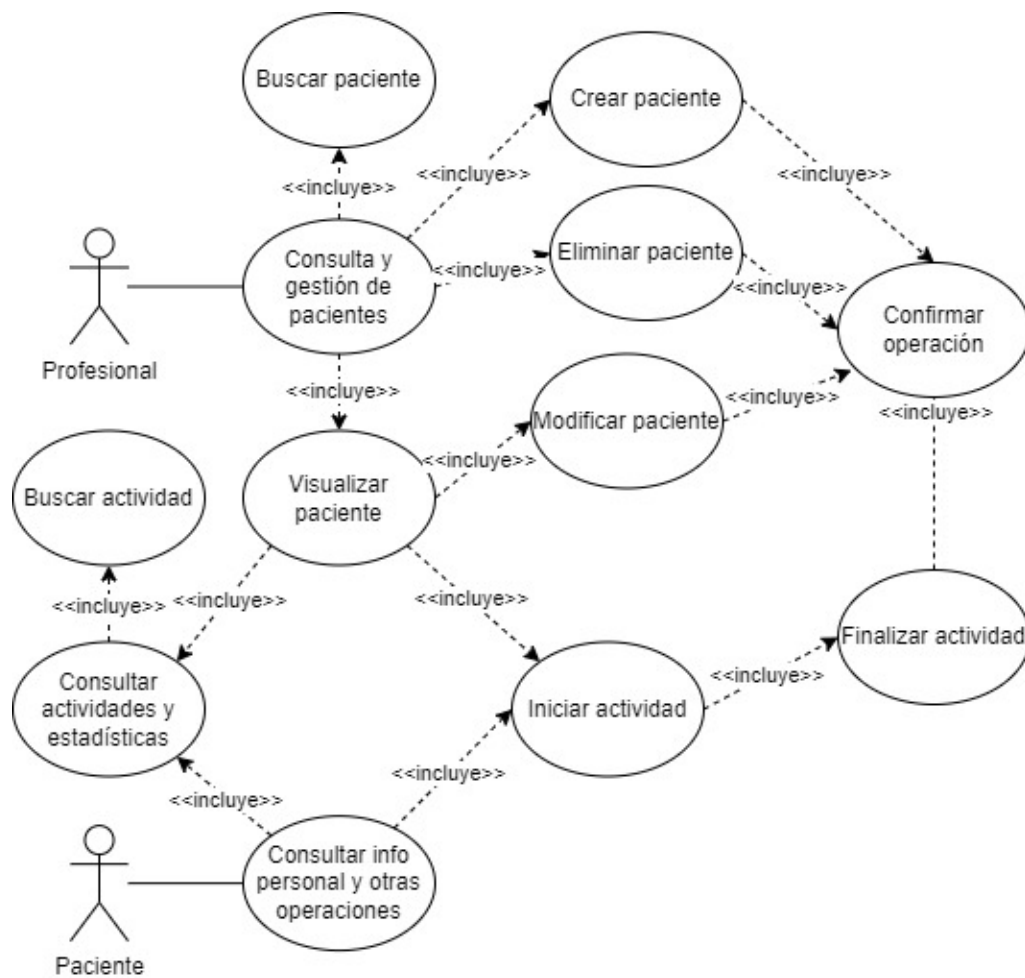


Figura F.3: CU-Pacientes y Profesionales

- Iniciar sesión. En la figura F.4 se muestran los campos a rellenar en la pantalla de acceso común a todos los usuarios.
- Confirmar operación. La notificación F.5 se mostrará ante operaciones delicadas como eliminación de pacientes para confirmar que el usuario realmente quiere llevar a cabo el proceso que ha iniciado (F.2).
- Consulta y gestión de usuarios. El menú de inicio del usuario administrador será similar a la figura F.6 de la forma que indica CU-7 (F.7).
- Crear usuario. Frontend de CU-9 (F.9) representado en la figura F.7. El administrador tendrá que rellenar los datos comunes a todos los

usuarios y, posteriormente, según el tipo de usuario que esté dando de alta será necesario completar unos campos específicos.

- Consultar lista de pacientes. La Figura F.8 es un primer boceto de la pantalla de inicio de un usuario profesional.
- Consultar información personal y otras operaciones. La pantalla de la Figura F.9 es accesible tanto por pacientes (será su pantalla de inicio) como por profesionales cuando seleccionen el paciente que quieren visualizar.
- Iniciar actividad. Al seleccionar la función de iniciar actividad, CU-16 (F.16), el sistema se comunica con Arduino y este le enviará los datos en tiempo real a una interfaz como la de la Figura F.10.
- Consultar actividades y estadísticas. Las imágenes de F.11 y F.12 reflejan la idea inicial de cómo organizar el CU-18 (F.18) para pacientes y profesionales respectivamente.
- Consulta y gestión de pacientes (F.13). Es la pantalla de inicio para profesionales.
- Crear paciente. Acción reservada a administradores mediante la cumplimentación del formulario F.14.
- Visualizar paciente. El profesional sanitario tiene acceso a los datos de cada paciente estructurados como en la pestaña de la Figura F.15.

A web browser window titled "Seguimiento Actividad Parkinson". The address bar shows "https://". The main content area has the heading "Inicio de sesión". Below the heading is a light gray rectangular box containing the login form. The form consists of three input fields: "email", "contraseña", and "Tipo de usuario" (a dropdown menu). Below these fields is a button labeled "Iniciar Sesión".

Figura F.4: Iniciar sesión

A confirmation dialog box with a rounded rectangular border. The text inside reads "¡Atención!" followed by "¿Quiere continuar?". At the bottom, there are two buttons: "Cancelar" on the left and "Aceptar" on the right.

Figura F.5: Confirmar operación

CU-2	Confirmar operación
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-03, RF-04, RF-07, RF-08
Descripción	Verificación y aprobación de acciones críticas o cambios significativos en el sistema.
Precondición	Operación pendiente de confirmación
Acciones	<ol style="list-style-type: none"> 1. El sistema muestra al usuario una ventana emergente para que este confirme la operación 2. Confirmación o rechazo de la operación a través de un clic
Postcondición	Operación confirmada y llevada a cabo
Excepción	Operación cancelada. El sistema devuelve al usuario a la pantalla anterior sin realizar el cambio que se iba a llevar a cabo
Importancia	Media

Tabla F.2: CU-2 Confirmar operación

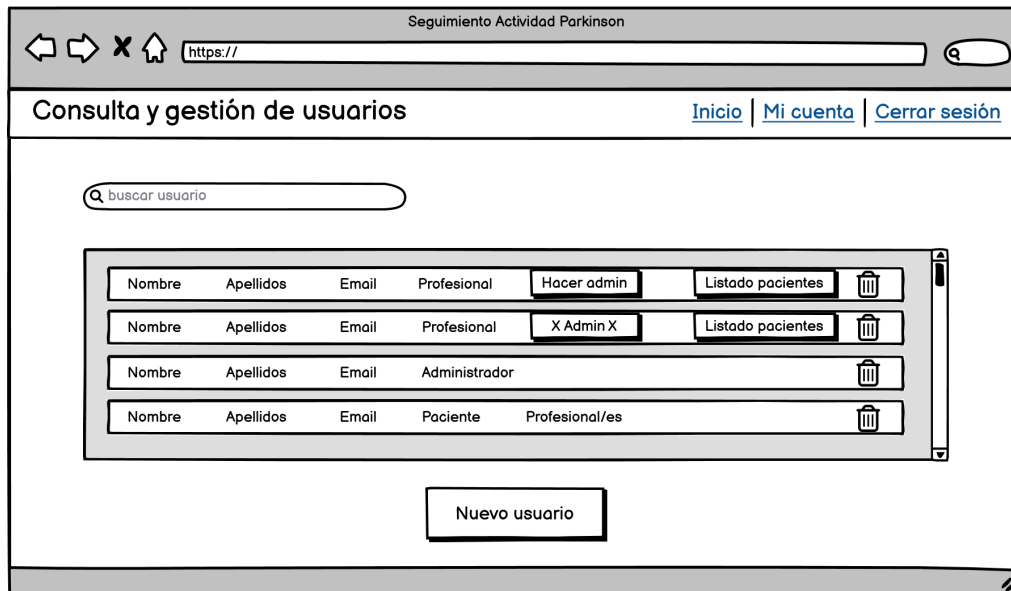


Figura F.6: Consulta y gestión de usuarios

CU-3	Gestionar cuenta
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-07
Descripción	Administración de las credenciales del usuario necesarias para el inicio de sesión
Precondición	Usuario autenticado
Acciones	<ol style="list-style-type: none"> 1. Acceso a la ventana de gestión de la cuenta y presentación de las opciones que esta ofrece 2. Selección de la operación que se quiere realizar; CU-4 Actualizar correo, CU-5 Cambiar contraseña
Postcondiciones	<ol style="list-style-type: none"> 1. Visualización de la información de la cuenta y las opciones disponibles. 2. Redirección a casos de uso específicos
Excepción	El usuario puede decidir no realizar ninguna de las operaciones y regresar a la pantalla anterior, página de inicio o cualquiera de las opciones del menú principal
Importancia	Media

Tabla F.3: CU-3 Gestionar cuenta

CU-4	Actualizar correo
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-07
Descripción	Cambio del correo electrónico asociado a la cuenta del usuario
Precondición	Redirección desde 'Gestionar Cuenta' CU-3
Acciones	<ol style="list-style-type: none"> 1. Introducir la nueva dirección de correo en el campo correspondiente 2. Verificación del nuevo correo introducido. El sistema solicita al usuario que reingrese el correo para confirmar que es el correcto. 3. Seleccionar de la opción 'aplicar cambios' 4. Confirmación de la operación, CU-2
Postcondiciones	<ol style="list-style-type: none"> 1. Actualización del correo electrónico y notificación de operación exitosa 2. Redirección a la ventana de Gestión de la cuenta (CU-2, CU-3)
Excepciones	<ol style="list-style-type: none"> 1. Campo vacío. Solicitar al usuario que introduzca la dirección de correo en los dos campos habilitados para ello 2. Error de verificación: los correos introducidos no coinciden. Situación comunicada al usuario para que este actúe en consecuencia 3. Formato de correo electrónico no válido. El sistema informa al usuario, solicita modificación y no se permite el cambio 4. Correo electrónico en uso. La dirección de correo introducida ya se encuentra asociada a otra cuenta. Se informa del problema al usuario y no se permite el cambio 5. Botón 'Cancelar operación' redirecciona a la pantalla de 'Gestión de la cuenta'
Importancia	Baja

Tabla F.4: CU-4 Actualizar correo

CU-5	Cambiar contraseña
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-07
Descripción	Modificación de la contraseña actual del usuario
Precondición	Redirección desde 'Gestionar Cuenta' CU-3
Acciones	<ol style="list-style-type: none"> 1. Ingresar la contraseña actual para verificar la identidad del usuario 2. Introducir nueva contraseña en el campo correspondiente 3. Repetir contraseña 4. Seleccionar de la opción 'aplicar cambios' 5. Confirmación de la operación, CU-2
Postcondiciones	<ol style="list-style-type: none"> 1. Actualización de la contraseña y notificación de operación exitosa 2. Redirección a la ventana de Gestión de la cuenta (CU-2, CU-3)
Excepciones	<ol style="list-style-type: none"> 1. Campo vacío. Solicitar al usuario que rellene todos los campos 2. Contraseña actual incorrecta. Notificación al usuario y solicitud para que se vuelva a introducir 3. Error de verificación. Las nuevas contraseñas introducidas no coinciden entre sí. Se informa al usuario y se solicita modificación 4. Botón 'Cancelar operación' redirecciona a la pantalla de 'Gestión de la cuenta'
Importancia	Baja

Tabla F.5: CU-5 Cambiar contraseña

CU-6	Cerrar sesión
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-08
Descripción	Desconexión segura del usuario del sistema
Precondición	Usuario autenticado (sesión iniciada)
Acciones	<ol style="list-style-type: none"> 1. Selección de 'Cerrar Sesión' desde la opción del menú principal 2. Confirmar operación, CU-2
Postcondiciones	<ol style="list-style-type: none"> 1. Sesión Cerrada 2. Redirección a la pantalla de inicio de sesión (CU-1)
Excepción	Operación cancelada desde CU-2
Importancia	Alta

Tabla F.6: CU-6 Cerrar sesión.

CU-7	Consulta y gestión de usuarios
Versión	1.1
Autor	Inés Martos Barbero
Requisitos asociados	RF-02, RF-03
Descripción	Revisión y manejo de los perfiles de usuarios registrados en el sistema
Precondición	Usuario con permisos administrativos accede a su pantalla de inicio
Acciones	<ol style="list-style-type: none"> 1. Visualizar todos los datos, diferentes para cada tipo de usuario 2. A través de una barra de búsqueda se puede realizar un filtrado de usuarios: CU-8 Buscar usuario 3. Presionar el botón 'Nuevo usuario' para acceder a CU-9 Crear usuario 4. Existen 2 botones diferentes para usuarios de tipo profesional que permiten controlar sus funciones de administrador a través de CU-12 Gestionar función admin de usuario profesional 5. Hacer click sobre el icono de la papelera: CU-13 Eliminar usuario
Postcondiciones	<ol style="list-style-type: none"> 1. Redirección a alguno de los CU mencionados 2. Actualización de la base de datos de usuarios según la acción realizada
Excepción	Fallos en alguno de los casos de uso que comprende
Importancia	Alta

Tabla F.7: CU-7 Consulta y gestión de usuarios

CU-8	Buscar usuario
Versión	1.1
Autor	Inés Martos Barbero
Requisitos asociados	RF-02
Descripción	Localización de usuarios dentro del sistema a través de nombre, apellidos, email y tipo de usuario
Precondiciones	<ol style="list-style-type: none"> 1. Usuario de tipo administrador autenticado 2. Acceso desde la pantalla de inicio
Acciones	<ol style="list-style-type: none"> 1. Ingreso manual de los criterios de búsqueda: nombre, apellidos, correo electrónico o tipo de usuario 2. Envío de la solicitud de búsqueda. Botón lupa o 'buscar' para la obtención de los resultados
Postcondiciones	<ol style="list-style-type: none"> 1. Presentación de una lista de usuarios que cumplen con los criterios de búsqueda. Se muestra su nombre, apellidos, correo y rol. 2. Se siguen ofreciendo opciones de gestión de usuarios: CU-9, CU-12, CU-13, CU-14 3. Opción de eliminar el filtro aplicado y volver a mostrar todos los usuarios. Botón 'Retroceder en la búsqueda'
Excepciones	<ol style="list-style-type: none"> 1. Búsqueda sin resultados 2. Formato erróneo del correo electrónico introducido 3. Error en la escritura del tipo de usuario
Importancia	Media

Tabla F.8: CU-8 Buscar usuario

CU-9	Crear usuario
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-03
Descripción	Registro de un nuevo usuario en el sistema, ya sea paciente, profesional o administrador
Precondición	Usuario administrador clicca sobre 'Nuevo usuario'
Acciones	<ol style="list-style-type: none"> 1. Completar los campos obligatorios de 'Nombre' y 'Apellidos'. 2. Introducir obligatoriamente el correo electrónico. Es necesario repetirlo para evitar fallos de escritura 3. Introducir contraseña para el acceso del usuario que se está creando. Hay que repetirla para evitar fallos de escritura. Recomendación: introducir el año de nacimiento y que el propio usuario la cambie con posterioridad 4. Seleccionar tipo de usuario: profesional (CU-10), paciente (CU-11) o administrador 5. Clicar en 'Guardar cambios' para continuar el proceso de creación 6. Cancelar la creación del nuevo usuario seleccionando el botón 'Descartar cambios' 7. Redireccionamiento a CU-2
Postcondiciones	<ol style="list-style-type: none"> 1. Usuario registrado en el sistema 2. Notificación de la creación exitosa del usuario
Excepciones	<ol style="list-style-type: none"> 1. Selección de un único tipo de usuario 2. Datos incompletos. Solicitud para que se completen todos los campos 3. Error formato de email no adecuado 4. Datos incorrectos. La información en los campos email/contraseña no coincide entre sí 5. Conflicto de datos. Ya existe un usuario con mismo nombre y apellidos y/o correo electrónico
Importancia	Alta

Tabla F.9: CU-9 Crear usuario

CU-10	Especificar características usuario profesional
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-03
Descripción	Define las características específicas de los usuarios profesionales
Precondiciones	<ol style="list-style-type: none"> 1. Usuario administrador 2. En el proceso de creación de nuevo usuario (CU-9) se ha seleccionado el tipo de usuario como 'profesional'
Acciones	<ol style="list-style-type: none"> 1. Seleccionar la casilla 'proporcionar rol administrador' si se le quieren conceder dichos permisos 2. Escoger el método de asignación de pacientes. Para la selección manual se desplegarán todos los pacientes y también se podrá buscar por nombre y/o apellidos
Postcondición	Los detalles quedan asociados al nuevo perfil
Excepciones	<ol style="list-style-type: none"> 1. Método de selección de pacientes no indicado 2. Selección de pacientes manual no realizada. En caso de que esta sea la opción escogida
Importancia	Alta

Tabla F.10: CU-10 Especificar características usuario profesional

CU-11	Especificar características usuario paciente
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-03
Descripción	Define las características específicas de los pacientes
Precondiciones	<ol style="list-style-type: none"> 1. Usuario administrador 2. En el proceso de creación de nuevo usuario (CU-9) se ha seleccionado el tipo de usuario como 'paciente'
Acciones	<ol style="list-style-type: none"> 1. Completar campos 'altura' y 'sexo' 2. Selección del método de asignación de profesional. La forma manual despliega una lista en la que también se puede buscar por nombre y/o apellidos
Postcondición	Los detalles quedan asociados al nuevo perfil
Excepciones	<ol style="list-style-type: none"> 1. Información incompleta. Falta rellenar el campo altura y/o sexo, lo que se solicitará a través de una notificación 2. Formato erróneo. El formato en el que se ha introducido la información del campo altura y/o sexo es erróneo 3. Método de asignación de profesional no indicado 4. Selección de profesional manual no realizada. En caso de que esta sea la opción escogida
Importancia	Alta

Tabla F.11: CU-11 Especificar características usuario paciente

CU-12	Gestionar función admin
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-03
Descripción	Asignar o revocar roles administrativos a usuarios profesionales a través de un botón que indica el estado actual del rol
Precondiciones	<ol style="list-style-type: none"> 1. Usuario administrador autenticado 2. El administrador debe estar visualizando el perfil de un usuario profesional en su pantalla de inicio
Acciones	<ol style="list-style-type: none"> 1. Identificar el profesional cuyo rol se desea gestionar 2. Según el estado actual, el administrador selecciona el botón correspondiente: 'Hacer admin' para otorgar funciones administrativas, o 'Quitar admin' para revocarlas 3. Tras presionar el botón se redirige a CU-2 para confirmar la operación
Postcondiciones	<ol style="list-style-type: none"> 1. Actualización del perfil profesional reflejando la adición o eliminación de funciones administrativas 2. Notificación de actualización exitosa
Excepciones	<ol style="list-style-type: none"> 1. No se localiza el profesional que se quiere modificar 2. Operación cancelada en CU-2. No se realiza ningún cambio
Importancia	Baja

Tabla F.12: CU-12 Gestionar función admin

CU-13	Eliminar usuario
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-03
Descripción	Facilita la eliminación de usuarios del sistema
Precondición	El usuario administrador debe estar visualizando el usuario que pretende eliminar
Acciones	<ol style="list-style-type: none"> 1. Hacer clic sobre el icono de papelera situado a la derecha de la información perteneciente al usuario de interés 2. CU-2 para confirmación de la operación
Postcondición	Usuario seleccionado es eliminado del sistema. Si se trata de un paciente, desaparece de las listas de profesionales. Si se trata de un profesional, sus pacientes son redistribuidos automáticamente entre otros profesionales
Excepción	Operación CU-2 cancelada, el usuario no se elimina
Importancia	Baja

Tabla F.13: CU-13 Eliminar usuario

CU-14	Consultar lista de pacientes
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-02, RF-03
Descripción	Permite visualizar la lista completa de pacientes asociados a un profesional específico, y sus opciones de gestión
Precondición	Usuario administrador selecciona el botón 'listado de pacientes' correspondiente al profesional de interés, desde la pantalla de inicio (consulta y gestión de usuarios)
Acciones	<ol style="list-style-type: none"> 1. El usuario presiona el botón 'listado de pacientes' 2. Visualización de los pacientes asignados al profesional de interés. Incluye la siguiente información: nombre, apellidos, email y profesionales asignados 3. Barra de búsqueda de paciente CU-21 4. Ofrece la posibilidad de eliminar pacientes CU-13 5. Botón con opción de volver a la pantalla de inicio 6. Botón para realizar una redistribución automática de pacientes si se considera que el profesional tiene demasiados a su cargo. Se repartirán pacientes a otros profesionales. Redirecciona a CU-2
Postcondiciones	<ol style="list-style-type: none"> 1. El administrador obtiene una visión completa de los pacientes asignados a un profesional específico 2. Tras la redistribución automática se muestra la lista actualizada y se envía notificación de operación exitosa
Excepciones	<ol style="list-style-type: none"> 1. Paciente no localizado 2. Redistribución automática no aprobada en CU-2
Importancia	Baja

Tabla F.14: CU-14 Consultar lista de pacientes

CU-15	Consultar información personal y otras operaciones
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-02, RF-03, RF-04, RF-05, RF-06
Descripción	Permite a los usuarios acceder a su información personal y a las funciones principales de su cuenta
Precondición	Paciente accede a su cuenta
Acciones	<ol style="list-style-type: none"> 1. Visualización de la información personal relativa al paciente que ha iniciado sesión: sexo, altura, email y profesionales a cargo de su tratamiento 2. Las opciones que se presentan en forma de botones son: 'iniciar actividad' (CU-16), 'actividades y estadísticas' (CU-18) y 'actividad offline'. 3. Escoger la opción que se desea realizar 4. 'actividad offline' mostrará una pestaña informando al usuario de cómo usar el dispositivo físico para realizar una actividad sin necesidad de conexión a la página web
Postcondición	Visualización de la información y, en caso de pinchar uno de los botones, redirección a una pestaña nueva
Excepción	Tras acceder a alguna de las opciones se podrá regresar a la pantalla de inicio en cualquier momento sin realizar cambios
Importancia	Alta

Tabla F.15: CU-15 Consultar información personal y otras operaciones

CU-16	Iniciar actividad
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-04
Descripción	Comenzar a visualizar en tiempo real y a registrar los datos de una actividad del paciente
Precondición	Usuario profesional o paciente acceden a la opción 'Iniciar actividad'
Acciones	<ol style="list-style-type: none"> 1. El usuario presiona el botón 'Iniciar actividad' 2. Redirección a CU-2 para confirmar la operación 3. Una nueva página muestra en tiempo real los datos de la actividad: distancia recorrida, velocidad, número de pasos y número de bloqueos 4. Opción 'Finalizar actividad' (CU-17) para concluir el registro de datos
Postcondición	Visualizar en tiempo real los datos de la actividad
Excepciones	<ol style="list-style-type: none"> 1. CU-2 cancela la operación 2. Problemas de conectividad que no permiten la visualización en tiempo real
Importancia	Alta

Tabla F.16: CU-16 Iniciar actividad

CU-17	Finalizar actividad
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-04
Descripción	Concluir el registro de datos de una actividad en proceso
Precondición	Que una actividad se esté llevando a cabo y el botón 'Finalizar actividad' esté disponible
Acciones	<ol style="list-style-type: none"> 1. Presionar el botón 'Finalizar actividad' 2. Confirmar operación CU-2 3. Seleccionar opción para la finalización: 'Descartar actividad' si no se quieren almacenar los datos, o 'Guardar actividad' si se quieren guardar los datos para consultas futuras
Postcondiciones	<ol style="list-style-type: none"> 1. Actividad finalizada y almacenada para su posterior consulta 2. Se muestra la pantalla con los datos personales del paciente
Excepciones	<ol style="list-style-type: none"> 1. CU-2 Cancela la operación 2. No se almacenan los datos de la actividad
Importancia	Alta

Tabla F.17: CU-17 Finalizar actividad

CU-18	Consultar actividades y estadísticas
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-05, RF-06
Descripción	Permite acceder al listado de todas las actividades, consultarlas en un calendario y ver las estadísticas globales
Precondición	Acceso presionando el botón 'Actividades y estadísticas' desde la pantalla en la que se visualiza la información personal del paciente
Acciones	<ol style="list-style-type: none"> 1. Visualizar todos los datos diferentes para cada actividad: ID, fecha, hora, número de bloqueos, distancia, velocidad media y número de pasos 2. A través de una barra de búsqueda se puede realizar un filtrado de actividades: CU-19 Buscar actividad 3. Apartado de estadísticas globales: media de bloqueos, distancia total, media de velocidades y media de pasos 4. Calendario de actividades. Cada día almacena el nombre de la actividad realizada (ID); seleccionar una actividad muestra sus datos en una ventana emergente
Postcondición	Visualización de actividades desde diferentes perspectivas
Excepción	No existencia de actividades almacenadas
Importancia	Alta

Tabla F.18: CU-18 Consultar actividades y estadísticas

CU-19	Buscar actividad
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-05, RF-06
Descripción	Localización de actividades a través de ID, fecha, hora o número de bloqueos
Precondición	Usuario situado en la ventana de consulta de actividades y estadísticas
Acciones	<ol style="list-style-type: none"> 1. Ingreso manual de los criterios de búsqueda: ID actividad, fecha, hora o número de bloqueos 2. Envío de la solicitud de búsqueda. Botón lupa o 'buscar' para la obtención de resultados
Postcondiciones	<ol style="list-style-type: none"> 1. Presentación de una lista de actividades que cumplen con los criterios de búsqueda 2. Se siguen visualizando el resto de funcionalidades del apartado de consulta de actividades y estadísticas 3. Opción de eliminar el filtro aplicado y volver a mostrar todas las actividades. Botón 'Retroceder en la búsqueda'
Excepciones	<ol style="list-style-type: none"> 1. Búsqueda sin resultados 2. Formato erróneo de los datos introducidos. Salta una alerta indicando cuales pueden ser los posibles problemas
Importancia	Baja

Tabla F.19: CU-19 Buscar actividad

CU-20	Consulta y gestión de pacientes
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-02, RF-03
Descripción	Revisión y manejo de los perfiles de pacientes registrados en el sistema
Precondición	Usuario profesional autenticado accede a su pantalla de inicio
Acciones	<ol style="list-style-type: none"> 1. Mostrar todos los datos, diferentes para cada paciente 2. A través de una barra de búsqueda se puede realizar un filtrado de pacientes: CU-21 Buscar paciente 3. Opción 'Consultar paciente' para visualizar sus datos personales y acceder a estadísticas, consulta y realización de actividades (CU-25) 4. Presionar el botón 'Dar de alta paciente' para acceder a CU-22 5. Hacer click sobre el icono de la papelera: CU-24 Eliminar usuario
Postcondiciones	<ol style="list-style-type: none"> 1. Visualición de todos los pacientes, sus datos y opciones de gestión 2. Redirección a alguno de los CU mencionados
Excepción	Inexistencia de pacientes asignados al profesional que accede la plataforma
Importancia	Media

Tabla F.20: CU-20 Consulta y gestión de pacientes

CU-21	Buscar paciente
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-02, RF-03
Descripción	Localización de pacientes asignados a un profesional, a través de: nombre, apellidos o email
Precondición	Usuario profesional autenticado accede a su pantalla de inicio (consulta y gestión de pacientes)
Acciones	<ol style="list-style-type: none"> 1. Ingreso manual de los criterios de búsqueda: nombre, apellidos o email 2. Envío de la solicitud de búsqueda. Botón lupa o 'buscar' para la obtención de los resultados
Postcondiciones	<ol style="list-style-type: none"> 1. Presentación de una lista de pacientes que cumplen con los criterios de búsqueda 2. Se siguen ofreciendo opciones de gestión de pacientes: CU-22, CU-24, CU-25
Excepciones	<ol style="list-style-type: none"> 1. Búsqueda sin resultados 2. Formato erróneo del texto de búsqueda
Importancia	Baja

Tabla F.21: CU-21 Buscar paciente

CU-22	Dar de alta paciente
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-03
Descripción	El profesional se auto-asigna un paciente
Precondición	Usuario profesional autenticado selecciona 'Dar de alta paciente' desde su pestaña de inicio (consulta y gestión de pacientes)
Acciones	<ol style="list-style-type: none"> 1. Acceder mediante el botón 'Dar de alta paciente' 2. Seleccionar un paciente ya dado de alta en el sistema o crear un nuevo paciente (CU-23) 3. Presionar botón 'Realizar operación' que redirigirá al profesional a CU-2
Postcondición	Paciente añadido al conjunto de pacientes del profesional que lo está dando de alta
Excepciones	<ol style="list-style-type: none"> 1. Botón 'Cancelar operación' 2. CU-2 operación cancelada
Importancia	Media

Tabla F.22: CU-22 Dar de alta paciente

CU-23	Crear paciente
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-03
Descripción	Profesional registra un nuevo usuario de tipo paciente en el sistema
Precondición	Usuario profesional autenticado que haya escogido la opción 'nuevo paciente' desde el apartado 'dar de alta paciente'
Acciones	<ol style="list-style-type: none"> 1. Completar los campos obligatorios de nombre y apellidos 2. Introducir obligatoriamente el correo electrónico. Es necesario repetirlo para evitar fallos de escritura 3. Introducir contraseña para el acceso del usuario que se está creando. Hay que repetirla para evitar fallos de escritura. Recomendación: introducir el año de nacimiento y que el propio usuario la cambie con posterioridad 4. Rellenar campos altura y sexo. 5. Asignación automática al profesional que está creando el usuario
Postcondiciones	<ol style="list-style-type: none"> 1. Usuario registrado en el sistema 2. Notificación de creación exitosa
Excepciones	<ol style="list-style-type: none"> 1. Datos incompletos. Solicitud para que se completen todos los campos 2. Error formato de email no adecuado 3. Datos incorrectos. La información en los campos email/contraseña no coincide entre sí 4. Conflicto de datos. Ya existe un usuario con mismo nombre y apellidos y/o correo electrónico
Importancia	Alta

Tabla F.23: CU-23 Crear paciente

CU-24	Eliminar paciente
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-03
Descripción	Habilita la eliminación de un paciente de aquellos asignados al profesional que realiza la acción
Precondición	El usuario profesional debe estar visualizando el paciente que pretende eliminar
Acciones	<ol style="list-style-type: none"> 1. Hacer clic sobre el icono de papelera situado a la derecha de la información perteneciente al usuario de interés 2. CU-2 para confirmación de la operación
Postcondición	Paciente seleccionado es eliminado de la lista de pacientes del profesional que lo está eliminando. Si dicho paciente no está asignado a otros profesionales, se verá eliminado del sistema
Excepción	CU-2 operación cancelada, el paciente no se elimina
Importancia	Baja

Tabla F.24: CU-24 Eliminar paciente

CU-25	Visualizar paciente
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-02, RF-03, RF-04, RF-05, RF-06
Descripción	Permite visualizar la información detallada de un paciente
Precondición	Profesional selecciona la opción 'visualizar paciente' del paciente de interés
Acciones	<ol style="list-style-type: none"> 1. Mostrar la información personal del paciente seleccionado: sexo, altura, email y profesionales a cargo de su tratamiento 2. Las opciones de gestión del paciente son: 'modificar datos' CU-26 y eliminar paciente (CU-24) 3. Otras opciones son: 'iniciar actividad' (CU-16), 'actividades y estadísticas' (CU-18)
Postcondición	Visualización de la información y, en caso de pinchar uno de los botones, redirección a una pestaña nueva
Excepción	Tras acceder a alguna de las opciones se podrá regresar a la pantalla de visualización del paciente sin realizar cambios
Importancia	Alta

Tabla F.25: CU-25 Visualizar paciente

CU-26	Modificar paciente
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-03
Descripción	Actualización de los datos de un paciente
Precondición	Profesional selecciona la opción 'modificar paciente' desde la pantalla de visualización de dicho paciente
Acciones	<ol style="list-style-type: none"> 1. Seleccionar 'modificar paciente' del paciente de interés 2. Actualizar los datos que se considere necesario: nombre, apellidos, sexo, altura, email 3. Presionar botón 'Aplicar cambios' 4. Confirmar operación CU-2
Postcondición	Se aplican los cambios. Modificaciones visibles en la página de visualización de datos personales del paciente
Excepciones	CU-2 operación cancelada
Importancia	Alta o Media o Baja...

Tabla F.26: CU-26 Modificar paciente

CU-27	Eliminar cuenta
Versión	1.0
Autor	Inés Martos Barbero
Requisitos asociados	RF-07
Descripción	Borrado de todos los datos del usuario que está ejecutando la acción
Precondición	Usuario registrado
Acciones	<ol style="list-style-type: none"> 1. Seleccionar la opción 'eliminar cuenta' del menú 2. Redireccionamiento a CU-2 3. Si el usuario eliminado es de tipo profesional, todos sus pacientes que no estén asignados a otros profesionales, se readjudicarán de forma automática
Postcondición	El usuario deja de existir por completo. Se elimina toda información relacionada
Excepciones	CU-2 operación cancelada
Importancia	Baja

Tabla F.27: CU-27 Eliminar cuenta

Seguimiento Actividad Parkinson

Nuevo usuario [Inicio](#) | [Mi cuenta](#) | [Cerrar sesión](#)

Información general

* Nombre

* Apellidos

* email

* repetir email

* contraseña

* repetir contraseña

Tipo de usuario

seleccionar tipo de usuario

paciente

* altura

* sexo

☐ Asignar profesional de forma automática

☐ Seleccionar profesional/es

seleccionar profesional

profesional

☐ Proporcionar rol administrador

☐ Asignar pacientes de forma automática

☐ No asignar pacientes

☐ Seleccionar pacientes

seleccionar paciente

administrador

Descartar cambios

Guardar cambios

Figura F.7: Crear usuario

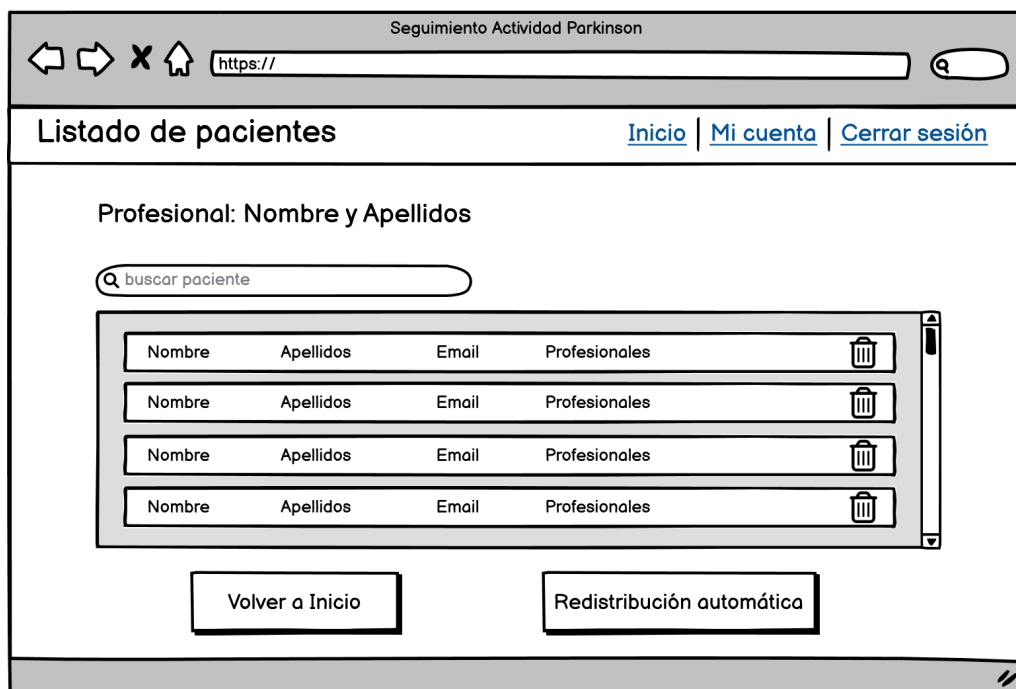


Figura F.8: Consultar lista de pacientes

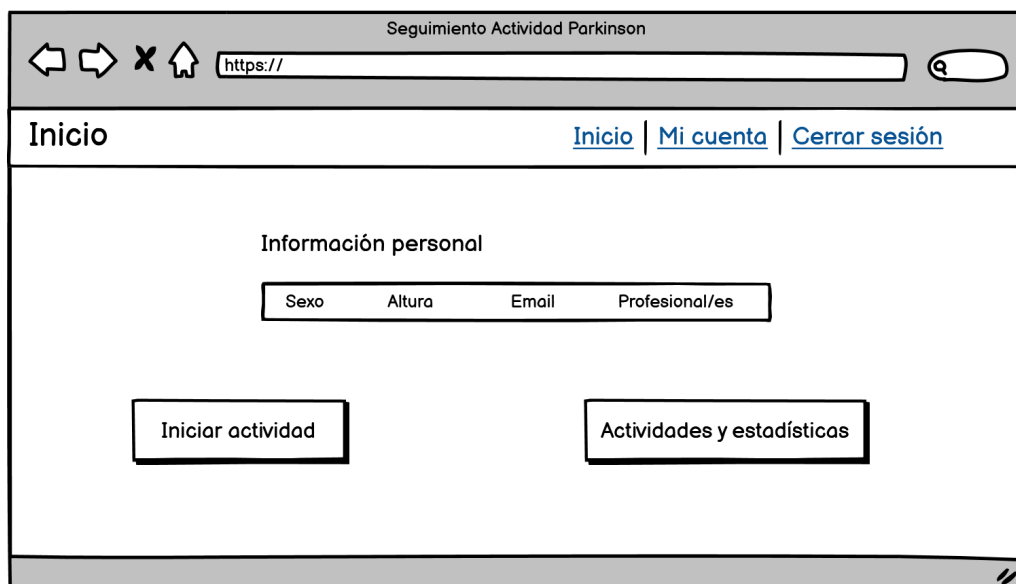
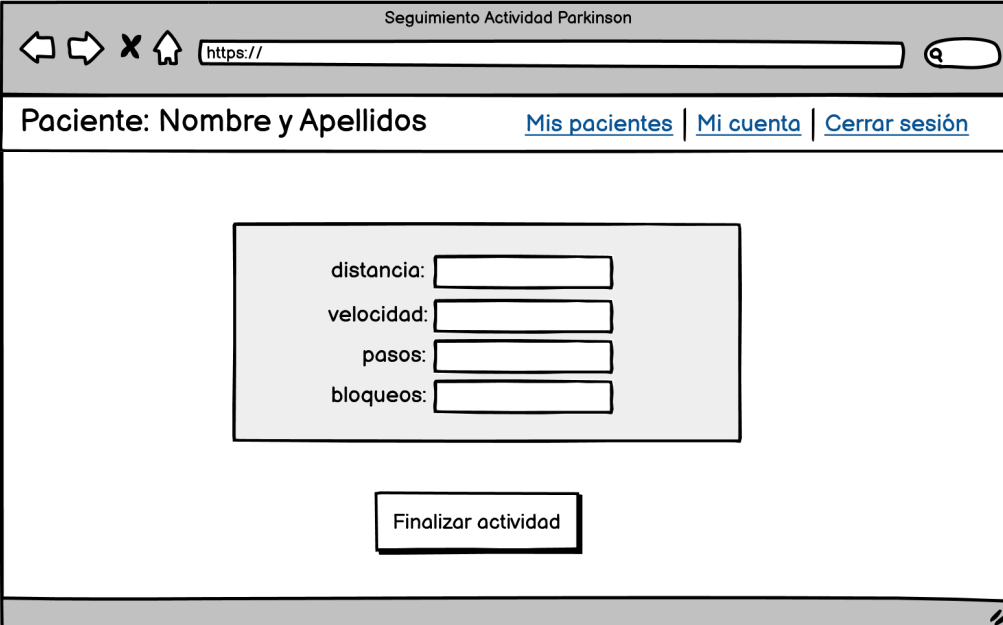


Figura F.9: Consultar información personal y otras actividades



The image shows a web browser window titled "Seguimiento Actividad Parkinson". The address bar contains "https://". The page header includes the text "Paciente: Nombre y Apellidos" followed by three links: "Mis pacientes", "Mi cuenta", and "Cerrar sesión". The main content area features a light gray box with four input fields labeled "distancia:", "velocidad:", "pasos:", and "bloqueos:". Below this box is a button labeled "Finalizar actividad".

Seguimiento Actividad Parkinson

← → × 🏠 https:// 🔍

Paciente: Nombre y Apellidos [Mis pacientes](#) | [Mi cuenta](#) | [Cerrar sesión](#)

distancia:

velocidad:

pasos:

bloqueos:

Figura F.10: Iniciar actividad

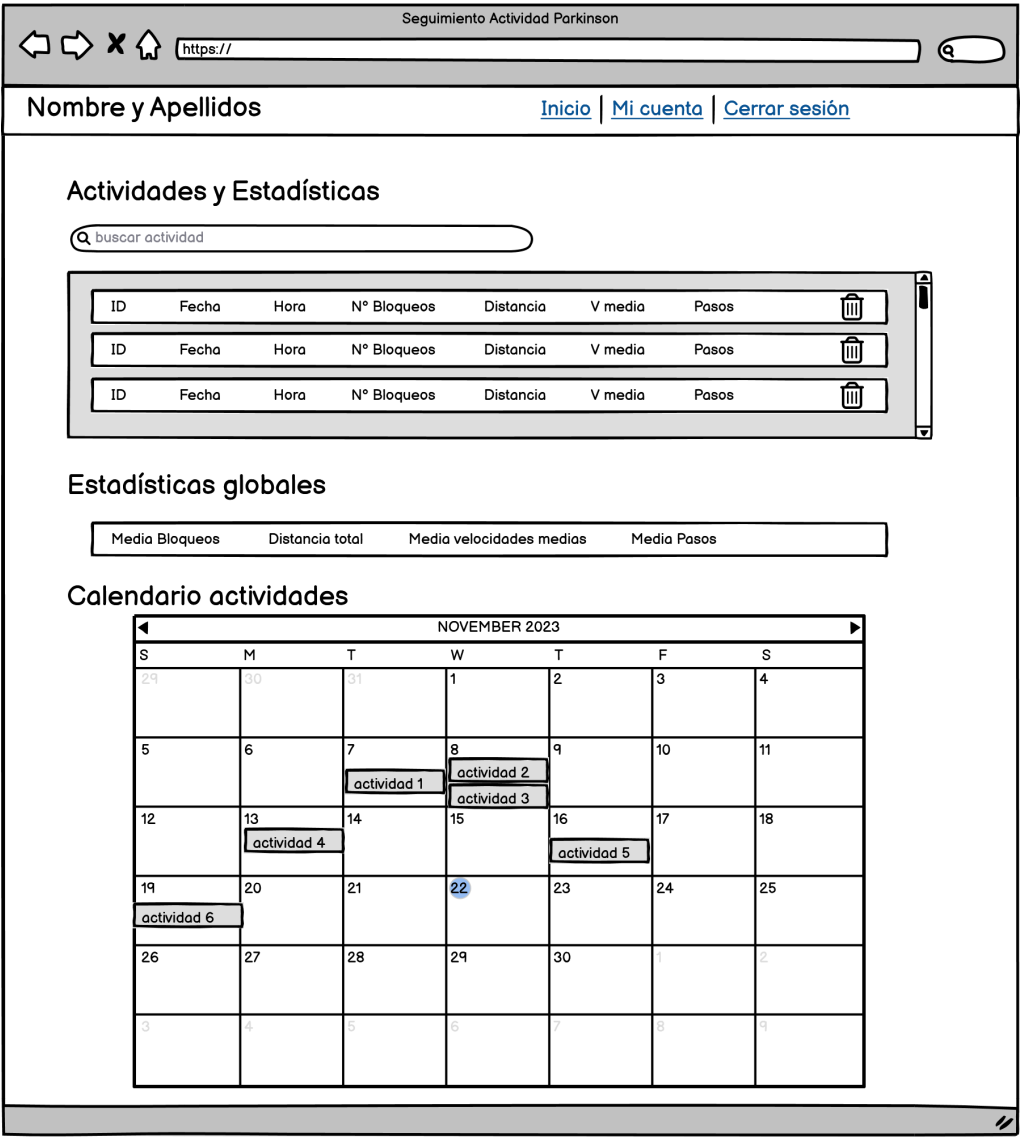


Figura F.11: Consultar actividades y estadísticas (Paciente)

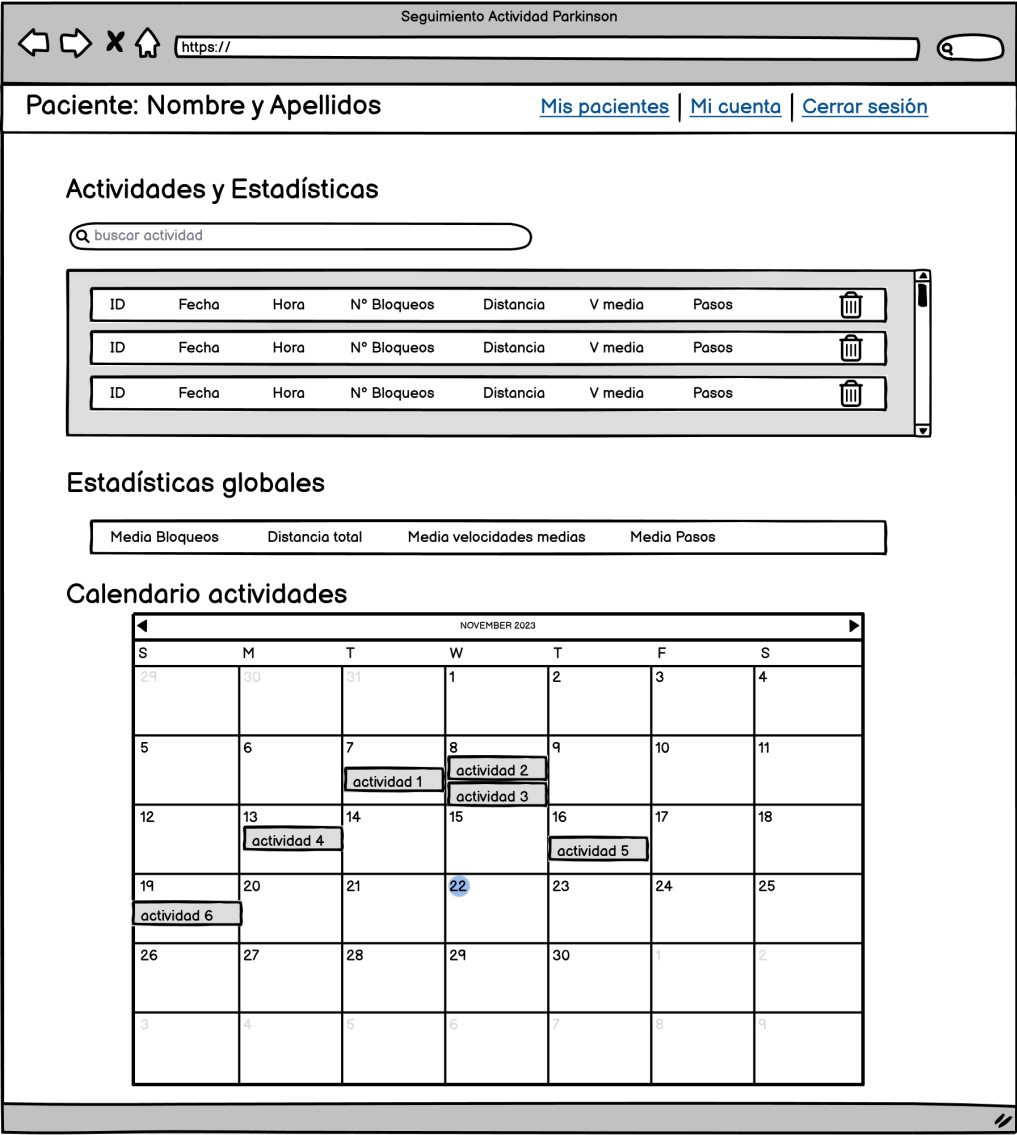


Figura F.12: Consultar actividades y estadísticas (Profesional)

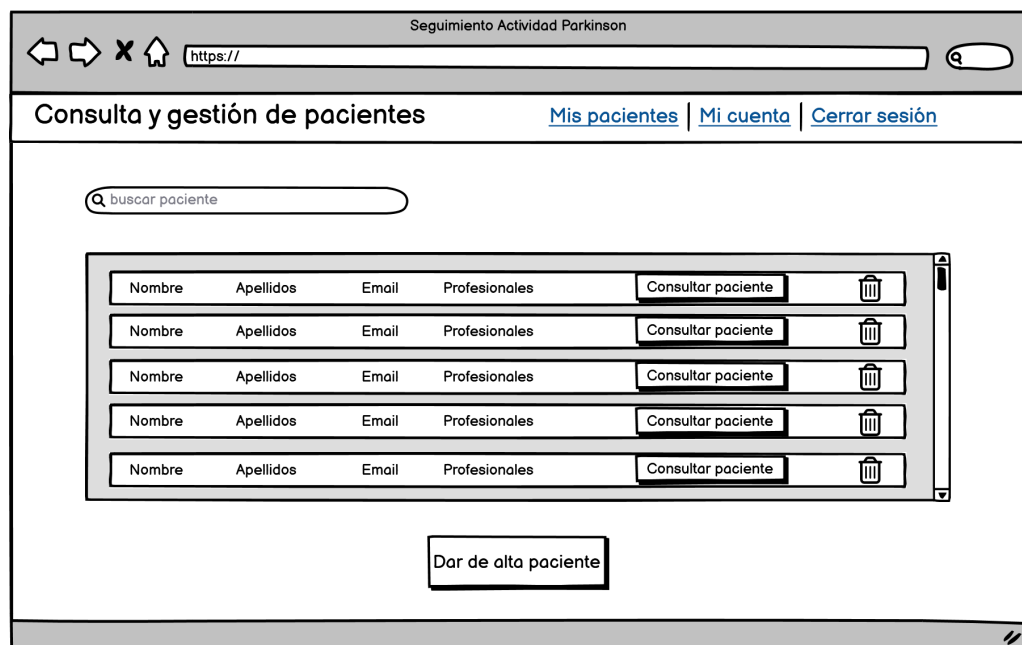


Figura F.13: Consulta y gestión de pacientes

The screenshot shows a web browser window titled "Seguimiento Actividad Parkinson". The address bar shows "https://". The page header includes the title "Nuevo paciente" and navigation links: "Mis pacientes", "Mi cuenta", and "Cerrar sesión". The main content area contains a form with the following fields:

- * Nombre
- * Apellidos
- * email
- * repetir email
- * contraseña
- * repetir contraseña
- * altura
- * sexo

At the bottom of the form are two buttons: "Descartar cambios" and "Guardar cambios".

Figura F.14: Crear paciente

The screenshot shows a web browser window titled "Seguimiento Actividad Parkinson". The address bar shows "https://". The page header includes the title "Paciente: Nombre y Apellidos" and navigation links: "Mis pacientes", "Mi cuenta", and "Cerrar sesión". The main content area displays the patient's information:

Información personal

Sexo	Altura	Email	Profesionales	Modificar datos	
------	--------	-------	---------------	-----------------	--

Below the table are three buttons: "Menú pacientes", "Iniciar actividad", and "Actividades y estadísticas".

Figura F.15: Visualizar paciente

Apéndice G

Estudio experimental

G.1. Detalle de resultados.


Este anexo presenta la encuesta SUS y los resultados obtenidos habiendo sido completada por una persona ajena al proyecto después de su primera experiencia usando el sistema.

Los 10 ítems de la encuesta se responden mediante una escala Likert con opciones de 1 a 5, donde 1 significa “Totalmente en desacuerdo” y 5 “Totalmente de acuerdo”. Los resultados están contenidos en las Figuras G.1, G.2, G.3 y G.4.

Para calcular la puntuación SUS, que pretende proporcionar un resultado objetivo sobre la usabilidad del sistema, se deben seguir los pasos descritos a continuación [Brooke, 1995].


- Suma de la puntuación de cada ítem (varía de 0 a 4) siguiendo las siguientes reglas para obtenerla:
 - Ítems 1, 3, 5, 7 y 9, la puntuación es la posición en la escala menos 1.
 - Ítems 2, 4, 6, 8 y 10, la puntuación es 5 menos la posición en la escala.
- Multiplicar la suma de puntuaciones por 2.5.


En este caso se obtiene un
$$\text{SUS SCORE} = ((3+4+4+4+4) + (4+4+4+4+4)) \times 2.5 = 97.5 / 100,$$



System Usability Scale (SUS)

Este cuestionario de 10 ítems se utiliza para evaluar el grado de acuerdo de los usuarios con algunas afirmaciones sobre la usabilidad del sistema.
Las respuestas se utilizan para calcular una puntuación que refleja la usabilidad general del sistema.

[Cambiar de cuenta](#)  Borrador guardado

 No compartido

* Indica que la pregunta es obligatoria

1. Creo que usaría este sistema con frecuencia. *

	1	2	3	4	5	
Muy en desacuerdo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Muy de acuerdo

Figura G.1: Cuestionario SUS. Parte 1.

resultado que supera ampliamente el punto de referencia (68) necesario para considerar que la web cumple de forma adecuada con su propósito.

The image shows a portion of a SUS questionnaire with three items. Each item has a 5-point Likert scale with radio buttons. The first item is 'El sistema parece innecesariamente complejo.' with a red asterisk, and the second radio button (1) is selected. The second item is 'Encuentro que el sistema es fácil de usar.' with a red asterisk, and the fifth radio button (5) is selected. The third item is 'Creo que necesitaría la asistencia de un experto técnico para usar este sistema.' with a red asterisk, and the first radio button (1) is selected. The labels 'Muy en desacuerdo' and 'Muy de acuerdo' are at the ends of the scales.

2. El sistema parece innecesariamente complejo. *

Muy en desacuerdo 1 2 3 4 5 Muy de acuerdo

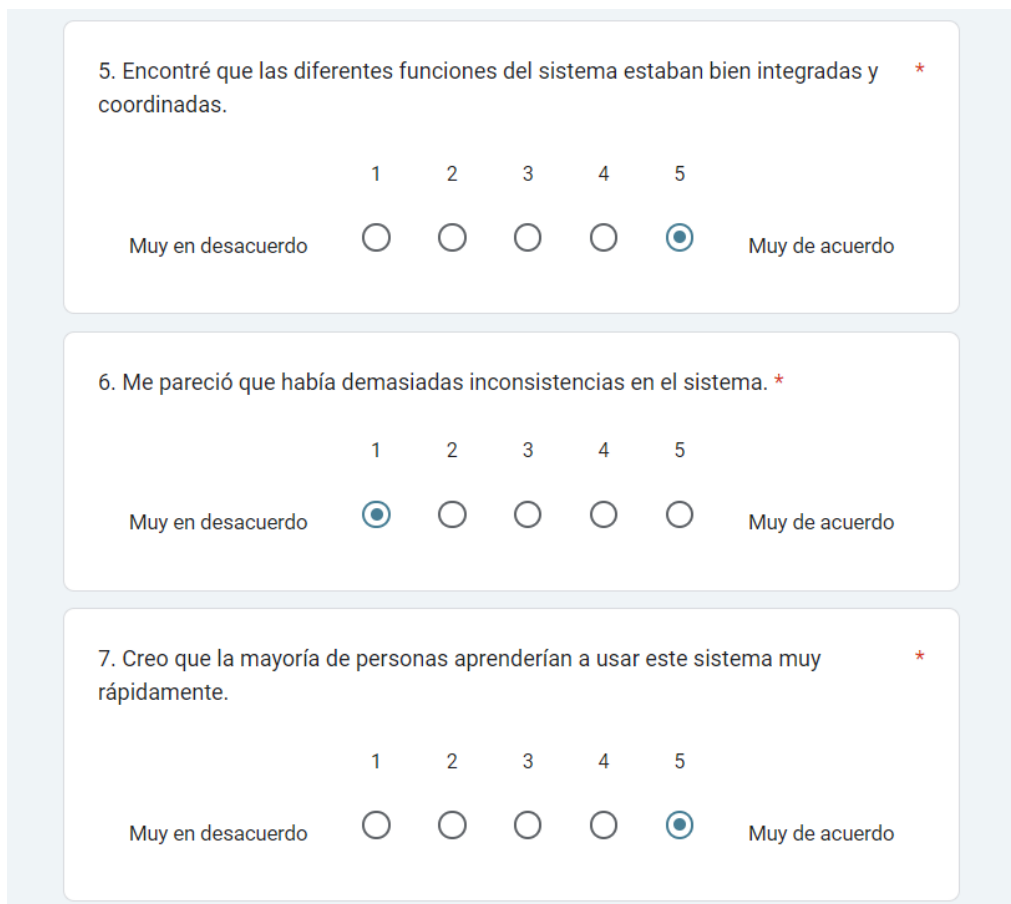
3. Encuentro que el sistema es fácil de usar. *

Muy en desacuerdo 1 2 3 4 5 Muy de acuerdo

4. Creo que necesitaría la asistencia de un experto técnico para usar este sistema. *

Muy en desacuerdo 1 2 3 4 Muy de acuerdo

Figura G.2: Cuestionario SUS. Parte 2.



5. Encontré que las diferentes funciones del sistema estaban bien integradas y coordinadas. *

Muy en desacuerdo 1 2 3 4 5 Muy de acuerdo

6. Me pareció que había demasiadas inconsistencias en el sistema. *

Muy en desacuerdo 1 2 3 4 5 Muy de acuerdo

7. Creo que la mayoría de personas aprenderían a usar este sistema muy rápidamente. *

Muy en desacuerdo 1 2 3 4 5 Muy de acuerdo

Figura G.3: Cuestionario SUS. Parte 3.

The image shows a screenshot of a questionnaire with three items, each with a 5-point Likert scale. The scales are labeled 'Muy en desacuerdo' (Very disagree) on the left and 'Muy de acuerdo' (Very agree) on the right. The items are:

- Item 8: 'Encontré el sistema muy engorroso/incómodo/difícil de usar. *'. The scale shows a selection at 1 (Muy en desacuerdo).
- Item 9: 'Me sentí muy seguro al usar el sistema. *'. The scale shows a selection at 5 (Muy de acuerdo).
- Item 10: 'Necesité aprender muchas cosas para empezar a usar este sistema. *'. The scale shows a selection at 1 (Muy en desacuerdo).

Each item's scale is represented by five radio buttons labeled 1 through 5. The selected option is indicated by a filled blue circle.

Figura G.4: Cuestionario SUS. Parte 4.

Anexo de sostenibilización curricular

En el desarrollo de este Trabajo de Fin de Grado (TFG) he tenido la oportunidad de aplicar los principios de sostenibilidad adquiridos durante mi formación académica, profundizando en su comprensión más allá del ámbito teórico. Esta experiencia me ha permitido entender el impacto y los riesgos que la actividad profesional de un ingeniero de la salud puede tener sobre la sociedad y el medio ambiente.

A continuación, se describe cómo el proyecto respeta y se adhiere a los principios éticos de justicia social y calidad ambiental.

- **Respeto a los derechos humanos y derechos fundamentales.**
El enfoque del proyecto respeta los derechos humanos y fundamentales, particularmente el derecho a la salud y a gozar de los beneficios derivados del progreso científico. El uso de tecnología de bajo coste y la decisión de emplear hardware y software libre promueven la igualdad de todos los individuos para el acceso a la solución tecnológica obtenida.
- **Respeto a la igualdad de género, no discriminación y principios de accesibilidad universal.**
El diseño sencillo e inclusivo de la interfaz de la página web garantiza la accesibilidad a todos los usuarios, independientemente de su género, edad, raza, capacidades físicas y habilidades tecnológicas. Las características del sistema lo convierten en un recurso valioso y accesible para cualquier paciente de Parkinson, contribuyendo a la inclusión. Otro

aspecto que promueve la igualdad de acceso es la elección de software y hardware libre en el desarrollo del proyecto.

■ **Tratamiento de la sostenibilidad y el cambio climático.**

Consciente del impacto ambiental de la tecnología, en el desarrollo se ha querido reducir la huella ecológica en la medida de lo posible. Estas consideraciones condujeron a la elección de una batería recargable y materiales con una larga vida útil.

Este anexo, y el proyecto en su conjunto, reflejan mi compromiso con una evolución tecnológica innovadora que responde a las necesidades sociales, de acuerdo con la protección de los derechos humanos y atendiendo a la actual situación de crisis climática.

Bibliografía

- [Amazon,] Amazon. Amazon.es: compra online de electrónica, libros, deporte, hogar, moda y mucho más. <https://www.amazon.es/>. (Accessed on 01/17/2024).
- [Apache,] Apache. Apache license, version 2.0. <https://www.apache.org/licenses/LICENSE-2.0>. (Accessed on 01/10/2024).
- [BOE, 1996] BOE (1996). Boe-a-1996-8930 real decreto legislativo 1/1996, de 12 de abril, por el que se aprueba el texto refundido de la ley de propiedad intelectual, regularizando, aclarando y armonizando las disposiciones legales vigentes sobre la materia. <https://boe.es/buscar/act.php?id=BOE-A-1996-8930>. (Accessed on 01/10/2024).
- [BOE, 2002] BOE (2002). Boe-a-2002-22188 ley 41/2002, de 14 de noviembre, básica reguladora de la autonomía del paciente y de derechos y obligaciones en materia de información y documentación clínica. <https://www.boe.es/buscar/act.php?id=BOE-A-2002-22188>. (Accessed on 01/10/2024).
- [BOE, 2009] BOE (2009). Boe.es - doue-l-2009-80808 directiva 2009/24/ce del parlamento europeo y del consejo, de 23 de abril de 2009, sobre la protección jurídica de programas de ordenador (versión codificada). <https://www.boe.es/buscar/doc.php?id=DOUE-L-2009-80808>. (Accessed on 01/10/2024).
- [BOE, 2016] BOE (2016). Boe.es - doue-l-2016-80807 reglamento (ue) 2016/679 del parlamento europeo y del consejo, de 27 de abril de 2016, relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos y por el que

- se deroga la directiva 95/46/ce (reglamento general de protección de datos). <https://www.boe.es/buscar/doc.php?id=DOUE-L-2016-80807>. (Accessed on 01/10/2024).
- [BOE, 2017] BOE (2017). Boe.es - doue-l-2017-80916 reglamento (ue) 2017/745 del parlamento europeo y del consejo, de 5 de abril de 2017, sobre los productos sanitarios, por el que se modifican la directiva 2001/83/ce, el reglamento (ce) nº 178/2002 y el reglamento (ce) nº 1223/2009 y por el que se derogan las directivas 90/385/cee y 93/42/cee del consejo. <https://www.boe.es/buscar/doc.php?id=DOUE-L-2017-80916>. (Accessed on 01/10/2024).
- [BOE, 2018] BOE (2018). Boe-a-2018-16673 ley orgánica 3/2018, de 5 de diciembre, de protección de datos personales y garantía de los derechos digitales. <https://www.boe.es/eli/es/lo/2018/12/05/3/con>. (Accessed on 01/10/2024).
- [BOE, 2023] BOE (2023). Boe-a-2023-7416 real decreto 192/2023, de 21 de marzo, por el que se regulan los productos sanitarios. <https://www.boe.es/buscar/act.php?id=BOE-A-2023-7416#dd>. (Accessed on 01/10/2024).
- [Brooke, 1995] Brooke, J. (1995). Sus: A quick and dirty usability scale. *Usability Eval. Ind.*, 189.
- [Castro, 2015] Castro, L. (2015). *Arquitectura del Software*. Cengage Learning Editores.
- [Cervillos, 2015] Cervillos, K. (2015). Uml: Casos de uso – ingeniería del software. <https://ingsotfwarekarlacevallos.wordpress.com/2015/06/04/uml-casos-de-uso/>. (Accessed on 01/11/2024).
- [di Biase et al., 2020] di Biase, L., Di Santo, A., Caminiti, M. L., De Liso, A., Shah, S. A., Ricci, L., and Di Lazzaro, V. (2020). Gait analysis in parkinson's disease: An overview of the most accurate markers for diagnosis and symptoms monitoring. *Sensors (Basel)*, 20(12):3529.
- [Goetz et al., 2008] Goetz, C. G., Tilley, B., Shaftman, S. R., Stebbins, G. T., Fahn, S., Martinez-Martin, P., Poewe, W., Sampaio, C., Stern, M. B., Dodel, R., Dubois, B., Holloway, R., Jankovic, J., Kulisevsky, J., Lang, A. E., Lees, A., Leurgans, S., LeWitt, P. A., Nyenhuis, D., Olanow, C. W., Rascol, O., Schrag, A., Teresi, J. A., van Hilten, J. J., and LaPelle, N. (2008). Escala unificada

de calificación de la enfermedad de parkinson mds (mds-updrs). <https://www.movementdisorders.org/MDS/MDS-Rating-Scales/MDS-Unified-Parkinsons-Disease-Rating-Scale-MDS-UPDRS.htm>. (Accessed on 01/29/2024).

[González, 2023] González, S. (2023). saragonzalezbarcelona/tfg_deteccion_activ_muscular: Trabajo de fin de grado de ingeniería de la salud. https://github.com/saragonzalezbarcelona/TFG_Deteccion_Activ_Muscular. (Accessed on 01/11/2024).

[IEBSCHOOL, 2023] IEBSCHOOL (2023). Tipos de licencias de software: Todo lo que tienes que saber. <https://www.iebschool.com/blog/modelos-negocios-software-libre-open-source-digital-business/>. (Accessed on 01/10/2024).

[ING, 2022] ING (2022). Amortización: qué es, cómo se calcula y qué tipos hay – ing – en naranja. <https://www.ing.es/ennaranja/finanzas-personales/conceptos-utiles/amortizacion-que-es-como-se-calcula/>. (Accessed on 01/10/2024).

[Jobted, 2024] Jobted (2024). ¿cuánto cobra un ingeniero biomédico? (sueldo 2024) | jobted.es. <https://www.jobted.es/salario/ingeniero-biom%C3%A9dico>. (Accessed on 01/09/2024).

[Mediavilla,] Mediavilla, E. Microsoft powerpoint - casos_de_uso_completo_sin_notas.ppt. https://www.ctr.unican.es/asignaturas/mc_oo/doc/casos_de_uso.pdf. (Accessed on 01/11/2024).

[Oliveros et al., 2014] Oliveros, A., Danyans, F. J., and Mastropietro, M. L. (2014). Stakeholders en los requerimientos de aplicaciones web. In *XLIII Jornadas Argentinas de Informática e Investigación Operativa (43JAIIO)-XV Simposio Argentino de Ingeniería de Software (Buenos Aires, 2014)*.

[Pradel Miquel et al., 2013] Pradel Miquel, J., Raya Martos, J. A., Campderrih Falgueras, B., Sánchez Porras, X., Fuertes Royo, C., and Albiñana Bertomeu, R. (2013). Ingeniería del software, febrero 2013.

[Rodríguez, 2023] Rodríguez, N. (2023). Naiaragadea/tfg_dispositivodecontrolpostural. https://github.com/NaiaraGadea/TFG_DispositivoDeControlPostural. (Accessed on 01/29/2024).

- [Seguridad-Social, 2023] Seguridad-Social (2023). Seguridad social: Cotización / recaudación de trabajadores / bases y tipos de cotización 2023. <https://www.seg-social.es/wps/portal/wss/internet/Trabajadores/CotizacionRecaudacionTrabajadores/36537>. (Accessed on 01/09/2024).
- [turiBOT,] turiBOT. Comprar en tienda arduino españa | turibot. <https://www.turibot.es/>. (Accessed on 01/17/2024).