

Desarrollo de una plataforma web para la gestión de centros de Plena Inclusión

Trabajo de Fin de Grado

Ingeniería Informática



VNiVERSIDAD
D SALAMANCA

julio de 2023

Autor:

Lidia Alaejos Herrero

Tutores:

Alicia García Holgado

Andrea Vázquez Ingelmo

CERTIFICADO TUTORES TFG

Dña. Alicia García Holgado, Profesora Contratada Doctora en el Departamento de Informática y Automática de la Universidad de Salamanca y Dña. Andrea Vázquez Ingelmo, Personal Investigador de la Universidad de Salamanca,

CERTIFICAN:

Que el trabajo titulado “Desarrollo de una plataforma web para la gestión de centros de Plena Inclusión”, que se presenta, ha sido realizado por Lidia Alaejos Herrero, con DNI 70961227-V y constituye la memoria del trabajo realizado para la superación de la asignatura Trabajo de Fin de Grado en Ingeniería Informática en esta Universidad.

Salamanca, 2 de julio de 2023

Fdo.:

RESUMEN

El presente Trabajo de Fin de Grado (TFG) se centra en el desarrollo de una página web destinada a los centros de Plena Inclusión, como ASPRODES y otros similares en España. El objetivo principal de esta página web es servir como recurso de gestión para facilitar la administración y organización de los centros, así como establecer una comunicación más cercana entre los trabajadores como son el equipo técnico, encargado de la gestión principal, es decir, cumpliendo de encargados en cada planta o departamento, los cuidadores y conductores, las familias y los usuarios con discapacidad intelectual asociada.

Las principales funcionalidades de la página web incluyen la posibilidad de agregar horarios de los trabajadores, registrar las rutas de autobús para los conductores encargados de recoger a las personas en cada parada, proporcionar a los cuidadores acceso a los protocolos del centro y permitir el acceso a fichas de apoyo individuales que contienen el diario y las necesidades médicas de cada persona con discapacidad, entre otros.

Además de las funcionalidades de gestión interna, se implementará un apartado de cursos y actividades, donde se ofrecerán diferentes programas formativos para los trabajadores, permitiéndoles inscribirse en los cursos de su interés. Asimismo, se dispondrá de un calendario donde se registrarán todos los eventos relacionados con cada usuario, brindando una visión global de las actividades, compromisos y horario laboral.

Como un elemento innovador y para fomentar la interacción entre los centros, las familias y los usuarios con menor grado de discapacidad, se incluirá un blog en la página web. Este blog permitirá a todos los usuarios participar activamente, realizar comentarios y visualizar publicaciones relacionadas con diversos temas, como menús semanales, excursiones, ferias y noticias relevantes para los centros de Plena Inclusión.

En resumen, este TFG se enfoca en el desarrollo de una página web destinada a los centros de Plena Inclusión, como ASPRODES, con el propósito de facilitar la gestión interna, promover la comunicación entre los diferentes actores involucrados y acercar el centro a las familias y los usuarios con menor grado de discapacidad. Las palabras clave asociadas a este proyecto son: página web, centros de Plena Inclusión, ASPRODES, gestión, comunicación, usuarios con discapacidad, horarios, rutas de

autobús, protocolos, fichas de apoyo, cursos, actividades, calendario, blog e interacción.

Palabras clave: plena inclusión, comunicación, discapacidad intelectual, gestión, interacción.

ABSTRACT

The present dissertation focuses on the development of a website aimed at Plena Inclusión centers, such as ASPRODES and similar ones in Spain. The main objective of this website is to serve as a management resource to facilitate the administration and organization of the centers, as well as to establish closer communication among the different stakeholders, including the technical team responsible for the main management tasks on each floor or department, caregivers, drivers, families, and individuals with associated intellectual disabilities.

The key functionalities of the website include the ability to add employee schedules, register bus routes for drivers responsible for picking up individuals at each stop, provide caregivers with access to center protocols, and allow access to individual support files containing the daily records and medical needs of each person with a disability, among others.

In addition to the internal management functionalities, a section for courses and activities will be implemented, offering different training programs for employees and allowing them to enroll in courses of their interest. Furthermore, a calendar will be available to register all events related to each user, providing a comprehensive view of activities, commitments, and work schedules.

As an innovative element to encourage interaction among the centers, families, and individuals with milder disabilities, a blog will be included on the website. This blog will enable all users to actively participate, make comments, and view posts related to various topics such as weekly menus, excursions, fairs, and relevant news for Plena Inclusión centers.

In summary, this dissertation focuses on the development of a website for Plena Inclusión centers, including ASPRODES, with the aim of facilitating internal management, promoting communication among the different stakeholders, and bringing the centers closer to families and individuals with milder disabilities.

Keywords: Plena Inclusión, communication, intellectual disability, management, interaction.

TABLA DE CONTENIDOS

RESUMEN	3
ABSTRACT	5
1.-INTRODUCCIÓN	9
2.-OBJETIVOS DEL PROYECTO	14
2.1.-OBJETIVOS DEL SISTEMA	14
2.2.-OBJETIVOS PERSONALES	14
3.-DESCRIPCIÓN DE LA APLICACIÓN	16
4.-TÉCNICAS Y HERRAMIENTAS	17
4.1.-DISEÑO DE LA SOLUCIÓN	17
4.1.1. Figma	17
4.1.2. Draw.io	17
4.1.3. Lectura fácil	18
4.2.-CÓDIGO DE PROGRAMACIÓN	19
4.2.1.-Python	19
4.2.2.-Django	19
4.2.3.-Bootstrap	20
4.2.4.-HTML, CSS, JavaScript y jQuery	20
4.2.5.-jQuery	20
4.2.5.1.-FullCalendar	21
4.2.5.2.-DataTables	21
4.3.-ENTORNO DE DESARROLLO Y DESPLIEGUE	21
4.3.1.-PostgreSQL	21
4.3.2.-PyCharm	22
4.3.3.-Google Drive	23
4.3.4.-Slack	23
4.3.5.-Zoom	23
4.4.-METODOLOGÍA	23
5.-ASPECTOS RELEVANTES DEL DESARROLLO	26
5.1.-PLANIFICACIÓN TEMPORAL	26
5.2.-ANÁLISIS DE LA COMPETENCIA	28
5.3.-BÚSQUEDA DE NECESIDADES	29
5.4.-DEFINICIÓN DE LA AUDIENCIA	30
5.4.1.-ARQUETIPOS DE USUARIOS	31
5.4.2.-ESCENARIOS DE USO	31
5.5.-ESPECIFICACIÓN DE REQUISITOS	32
4.5.1.-DEFINICIÓN DE ACTORES DEL SISTEMA	36
5.6.-CONCEPTUALIZACIÓN	37
5.7.-GESTIÓN DE DATOS	42
5.8.-FUNCIONALIDAD DEL SISTEMA	45
5.8.1.-GESTIÓN DE RUTAS	45

5.8.2.-GESTIÓN DE CALENDARIO Y SUS EVENTOS	47
5.9.-DESPLIEGUE	48
6.-CONCLUSIONES Y LÍNEAS FUTURAS DE TRABAJO	50
REFERENCIAS	53

ÍNDICE DE FIGURAS

Figura 1. Logo ResiPlus	28
Figura 2. Gerosalus Software	29
Figura 3. Logo IxisSocialGest	29
Figura 4. Diagrama Gestión Tablón De Anuncios	34
Figura 5. Diagrama Gestión Calendario	35
Figura 6. Diagrama Actores	37
Figura 7. Ejemplo Blog Interfaz	38
Figura 8. Ejemplo Calendario Interfaz	39
Figura 9. Figma Inicio Sesión	40
Figura 10. Figma Blog	41
Figura 11. Figma Blog Detalle	41
Figura 12. Diagrama Entidad-Relación No Definitivo	44
Figura 13. Diagrama Entidad-Relación Definitivo	44
Figura 14. Interfaz Rutas	46
Figura 15. Código Rutas I	47
Figura 16. Código Rutas II	47
Figura 17. Código Calendario I	48
Figura 18. Código Calendario II	48
Figura 19. Diagrama Despliegue	49

ÍNDICE DE TABLAS

Tabla 1. Distribución del tiempo a lo largo de todo el proyecto

28

1.-INTRODUCCIÓN

Actualmente, la informática se ha convertido en una presencia constante en nuestro entorno laboral. En casi cualquier trabajo, se hace uso de herramientas informáticas para tareas como la gestión de documentos digitalizados o la administración de información. Sin embargo, no todos los centros de trabajo disponen de una herramienta adecuada para el almacenamiento y procesamiento de datos. En muchos casos, se sigue utilizando una carpeta compartida donde se guardan numerosos archivos de Excel o Word, lo que genera caos, desorganización y pérdida de tiempo.

En este contexto, es relevante destacar la importancia de abordar la optimización y coordinación del trabajo en los centros de Plena Inclusión. La Plena Inclusión es un movimiento social que busca garantizar los derechos y la participación plena de las personas con discapacidad intelectual en todos los ámbitos de la sociedad [1]. Se basa en el principio de que todas las personas, independientemente de sus capacidades, tienen el derecho de ser incluidas y participar en igualdad de condiciones en la comunidad.

En España, existen diversos centros de Plena Inclusión que se dedican a promover la igualdad de oportunidades y la participación activa de las personas con discapacidad intelectual. Uno de estos centros es ASPRODES (Asociación Protectora de Personas con Discapacidad Intelectual de Salamanca), ubicado en la ciudad de Salamanca. ASPRODES es una entidad reconocida y consolidada que gestiona diferentes centros y servicios dirigidos a personas con discapacidad intelectual y sus familias [2].

ASPRODES opera en múltiples frentes, ofreciendo una amplia gama de servicios para satisfacer las necesidades de las personas con discapacidad intelectual y promover su plena inclusión en la sociedad. Entre los servicios que proporciona ASPRODES se encuentran los centros de día, donde se brinda apoyo y actividades de ocio y formación; un colegio de educación especial, enfocado en el desarrollo educativo y social de los alumnos; un centro especial de empleo, que ofrece oportunidades laborales adaptadas; y viviendas y residencias, que brindan apoyo y alojamiento a las personas con discapacidad.

Además de ASPRODES, existen otros centros de Plena Inclusión en España, como AEDIS (Asociación para la Atención y la Integración Social de las personas con Discapacidad Intelectual), Liber, Plena Inclusión Valencia y Plena Inclusión Andalucía, entre otros. Estas organizaciones comparten la misión de garantizar los derechos de

las personas con discapacidad intelectual y promover su inclusión en todos los ámbitos de la sociedad.

En este Trabajo de Fin de Grado, se propone el desarrollo de una página web llamada "Hogares Madrigal", que busca facilitar la gestión de los centros de Plena Inclusión. Esta página web proporcionará herramientas y recursos tecnológicos que permitirán a los empleados de los centros gestionar de manera más eficiente los protocolos internos, las fichas de apoyo individualizadas, los cursos y actividades, los horarios laborales y las rutas de transporte de los usuarios. Además, se incluirá un blog para fomentar la comunicación entre familiares, usuarios y empleados, compartiendo información relevante y promoviendo la interacción.

De esta manera, se logra acercar la tecnología a personas con diferentes grados de discapacidad intelectual, proporcionando una interfaz intuitiva y sencilla de utilizar. Al mismo tiempo, se mejora la organización y gestión de los centros de Plena Inclusión, lo cual tiene un impacto directo en la satisfacción y comodidad de los empleados, así como en la satisfacción de las familias al brindarles una comunicación más cercana y actualizada sobre sus seres queridos que forman parte del centro.

El presente documento constituye la memoria explicativa del Trabajo de Fin de Grado en Ingeniería Informática. En este informe se detallan los aspectos fundamentales del desarrollo del proyecto, los cuales se encuentran estructurados en los siguientes apartados:

- **Objetivos del proyecto:** En este apartado se describen de manera detallada los objetivos que se pretenden alcanzar con el proyecto. Se incluyen los objetivos funcionales, es decir, las funcionalidades específicas que se espera implementar en la aplicación web, los objetivos no funcionales, que se refieren a aspectos como rendimiento, seguridad y usabilidad, y los objetivos personales, que pueden estar relacionados con el aprendizaje o desarrollo profesional del autor.
- **Descripción de la aplicación:** En este apartado se brinda una descripción completa de la aplicación web desarrollada. Se explican su propósito, funcionalidades principales, público objetivo y cualquier otro aspecto relevante que permita comprender la naturaleza y el alcance de la aplicación.
- **Técnicas y herramientas:** En este apartado se documentan las técnicas, herramientas y tecnologías que se han utilizado en el desarrollo del proyecto.

Se explica el motivo de su elección, su contribución al desarrollo y cualquier consideración especial relacionada con su implementación.

- Aspectos relevantes del desarrollo: Aquí se detallan las tareas realizadas durante el desarrollo de la aplicación web, las decisiones tomadas, las problemáticas encontradas y las cuestiones más significativas que han surgido a lo largo del proceso. Se pueden mencionar desafíos técnicos superados, soluciones innovadoras implementadas y cualquier aspecto que destaque por su relevancia en el desarrollo del proyecto.
- Conclusiones y líneas futuras de trabajo: En este apartado se presentan las conclusiones obtenidas tras la finalización del proyecto. Se analizan los logros alcanzados, los aprendizajes adquiridos y se evalúa en qué medida se han cumplido los objetivos establecidos. Además, se plantean posibles líneas futuras de trabajo, sugerencias de mejora o posibles extensiones del proyecto.
- Referencias: En esta sección se citan todas las fuentes bibliográficas, documentales o cualquier otro recurso utilizado como referencia en el desarrollo del trabajo. Se deben incluir adecuadamente las citas y referencias bibliográficas según las normas establecidas.

Además, de la memoria, se añaden y documentan los siguientes anexos complementarios con más información y detalle:

- *Anexo I – Diseño Centrado en el Usuario*: Este anexo proporciona información adicional sobre el enfoque de diseño centrado en el usuario utilizado en el desarrollo de la aplicación web. Puede incluir diagramas de flujo, wireframes, casos de uso o cualquier otro material relacionado con el diseño de la interfaz y la experiencia del usuario.
- *Anexo II – Definición y análisis de requisitos*: En este anexo se detallan los requisitos funcionales y no funcionales identificados para la aplicación web. Se pueden incluir tablas, diagramas o documentos que describan exhaustivamente los requisitos del sistema.
- *Anexo III – Plan de proyecto software*: Aquí se muestra la planificación temporal del proyecto, incluyendo las diferentes etapas, tareas y plazos establecidos para su ejecución. Puede incluir diagramas de Gantt, calendarios u otros recursos visuales que ilustren la planificación temporal.
- *Anexo IV - Manual del usuario*: Este anexo contiene el manual del usuario de la aplicación web, proporcionando instrucciones detalladas sobre cómo utilizarla,

acceder a sus funcionalidades y resolver posibles problemas o dudas que puedan surgir.

- *Anexo V - Manual del programador*. En este anexo se presenta el manual del programador, que incluye información técnica y guías para desarrolladores que deseen mantener, modificar o ampliar la aplicación web. Puede contener información sobre la arquitectura del sistema, patrones de diseño utilizados, convenciones de codificación, entre otros aspectos relevantes.

2.-OBJETIVOS DEL PROYECTO

En esta sección se detallarán los objetivos que se buscan alcanzar mediante la ejecución del proyecto. Se diferenciarán los objetivos establecidos por los requisitos del software, los funcionales de los objetivos personales individuales.

2.1.-OBJETIVOS DEL SISTEMA

En cuanto a los objetivos del sistema, podríamos resumirlos en simples palabras: impulsar, mejorar, potenciar, desarrollar, proseguir... A continuación, detallaremos los principales:

- Mejorar la eficacia/eficiencia a través de la digitalización. Al tener informatizada y en red la información del centro y de su funcionamiento se accede a la misma de una forma más ágil y desde cualquier lugar. Esto permite una mejor coordinación y funcionamiento de todos los departamentos y personas de apoyo implicadas en el centro.
- Agilizar y mejorar los canales de comunicación de ida y vuelta del personal. Se evitan llamadas, correos electrónicos, olvidos, etc. Con esta herramienta se pueden registrar directamente vacaciones, cambios de turno, necesidades de algún residente o familiar.
- Transferir conocimientos de forma ágil y versátil a todos los profesionales, personas con discapacidad y familiares. Con el acceso a la plataforma podrán ver nuevos protocolos de actuación, infografías, talleres relevantes del grupo, pautas individualizadas, etc.
- Informar, acompañar y apoyar a las familias a través de esta herramienta.
- Implantar un sistema de comunicación a través de las nuevas tecnologías en el cual las personas con discapacidad sean las protagonistas de su propia vida.

2.2.-OBJETIVOS PERSONALES

En lo que respecta a los objetivos personales, mi principal aspiración es experimentar una satisfacción personal al saber que he desarrollado una herramienta que puede ayudar y facilitar el trabajo en centros de plena inclusión, especialmente en centros destinados a personas con discapacidad. Desde mi infancia, he tenido contacto directo con personas mayores y con individuos que enfrentan diversas dificultades, y siempre he deseado brindarles apoyo de alguna manera.

Además, en el ámbito de la informática, siento un gran interés por el diseño de páginas web. A través de este Trabajo de Fin de Grado, tengo la oportunidad de crear una página web y, al mismo tiempo, adquirir conocimientos sobre un nuevo lenguaje de programación. Esto me permitirá ampliar mis habilidades y conocimientos para adaptarme a las exigencias del mercado laboral.

Al embarcarme en este proyecto, también persigo otros objetivos personales de desarrollo y crecimiento profesional. Deseo fortalecer mis habilidades técnicas en el campo de la programación web, así como mejorar mi capacidad para abordar desafíos complejos y resolver problemas de manera eficiente. Asimismo, espero ampliar mi comprensión de los principios de usabilidad y diseño centrado en el usuario, lo que me permitirá crear una experiencia en línea más intuitiva y atractiva para los usuarios finales.

Además, a través de este proyecto, aspiro a mejorar mis habilidades de colaboración y comunicación, y así poder comprender las necesidades de otras personas y garantizar que la herramienta desarrollada cumpla con sus expectativas. También espero aprender a gestionar eficazmente el tiempo y los recursos, y a tomar decisiones fundamentadas durante todo el proceso de desarrollo.

En resumen, mi enfoque en este proyecto se basa en lograr tanto una satisfacción personal al contribuir al bienestar de las personas con discapacidad en los centros, como en ampliar mis habilidades y conocimientos en el ámbito de la programación web.

3.-DESCRIPCIÓN DE LA APLICACIÓN

La aplicación implementada en el presente proyecto TFG, es una página web diseñada para los centros de Plena Inclusión, como ASPRODES y otros centros similares en España. Su principal objetivo es facilitar la gestión interna de estos centros, promover la comunicación entre diferentes actores y acercar la tecnología a los usuarios con discapacidad.

La aplicación ofrece diversas funcionalidades que mejoran la organización y administración de los centros. Permite a los empleados agregar y gestionar horarios de trabajo, registrar rutas de autobús para los conductores encargados de recoger a las personas en cada parada, proporcionar acceso a protocolos del centro para los cuidadores, y ofrecer fichas de apoyo individuales que contienen información relevante sobre las necesidades médicas y diario de cada persona con discapacidad.

Además de estas funcionalidades internas, la aplicación también incluye un apartado de cursos y actividades donde se ofrecen programas formativos para los trabajadores, permitiéndoles inscribirse en los cursos de su interés. Asimismo, cuenta con un calendario donde se registran todos los eventos relacionados con cada usuario, brindando una visión global de las actividades y compromisos.

Una característica innovadora de la aplicación es la inclusión de un blog, que fomenta la interacción entre los centros, las familias y los usuarios con menor grado de discapacidad. A través del blog, los usuarios pueden participar activamente, realizar comentarios y acceder a publicaciones relacionadas con diferentes temas, como menús semanales, excursiones, ferias y noticias relevantes para los centros de Plena Inclusión.

En resumen, la aplicación es una página web integral que tiene como objetivo principal facilitar la gestión interna de los centros de Plena Inclusión, promover la comunicación entre los diferentes actores involucrados y acercar el centro a las familias y los usuarios con menor grado de discapacidad. Ofrece herramientas prácticas y funcionales para mejorar la administración, la organización y la interacción en estos entornos, utilizando la tecnología como un medio para promover la inclusión y la participación activa de todos los usuarios.

Adjuntamos el enlace a la página 'Hogares Madrigal':
<https://hogaresmadrigal.grial.eu/login>

4.-TÉCNICAS Y HERRAMIENTAS

En esta sección se detallarán herramientas similares a la propuesta, herramientas de diseño y desarrollo, así como los entornos de desarrollo y librerías utilizadas a lo largo del proyecto.

4.1.-DISEÑO DE LA SOLUCIÓN

4.1.1. Figma

Hemos empleado la herramienta de diseño y prototipado de interfaces de usuario Figma para crear un prototipo digital inicial de la futura interfaz de la página web. La elección de esta herramienta se basó en la experiencia previa y familiaridad con ella, habiéndola utilizado para desarrollar prototipos digitales en otros proyectos académicos y también para fines recreativos y de entretenimiento persona [3].

Además de nuestra experiencia previa y familiaridad con Figma, decidimos utilizar esta herramienta debido a su amplia gama de características y beneficios. Proporcionando una interfaz intuitiva y fácil de usar, lo que agilizó el proceso de diseño y prototipado. Además, su enfoque en la colaboración en tiempo real nos permitió trabajar de manera eficiente, facilitando la comunicación y la sincronización de cambios. También apreciamos la capacidad de Figma para crear prototipos interactivos, lo que nos permitió simular la experiencia de usuario y obtener retroalimentación valiosa antes de avanzar en el desarrollo. En general, el uso de Figma mejoró nuestra productividad, calidad y eficacia en la creación del prototipo de la interfaz de la página web.

4.1.2. Draw.io

Por el mismo caso que con la herramienta comentada anteriormente, se eligió el uso de esta herramienta de diagramación y creación de diagramas en línea, lo cual nos facilitó el almacenamiento de estos en Google Drive.

Mediante draw.io se realizaron los correspondientes diagramas de casos de uso y actores así como el diagrama de entidad utilizado posteriormente para la creación de la base de datos.

4.1.3. Lectura fácil

La "lectura fácil" es una metodología desarrollada por Plena Inclusión y otras organizaciones que busca hacer la información accesible y comprensible para personas con dificultades de comprensión lectora [4].

El objetivo de la lectura fácil es eliminar las barreras que puedan dificultar la comprensión de textos, utilizando un lenguaje sencillo, estructuras gramaticales simples, frases cortas y evitando tecnicismos o términos complejos. Además, se utilizan apoyos visuales, como imágenes o pictogramas, para facilitar la comprensión de los contenidos.

Por ello, para elegir los colores a usar en la interfaz de la página web, hemos seguido las pautas de diseño de 'Lectura fácil: métodos de redacción y evaluación'.

Tras una previa lectura del documento y posteriores análisis, se decidió usar una paleta de colores de gama azul, con fondo blanco y texto en negro, para proporcionar un buen contraste y facilitar así la lectura, además de elementos visuales intuitivos para facilitar la mencionada comprensión de los contenidos.

Remarcar que la elección de la paleta de color de gama azul se debe a puntos clave como:

- Asociaciones positivas: el azul, en particular los tonos más claros y suaves, a menudo se asocian con la calma, la tranquilidad y la serenidad. Estas asociaciones pueden crear un ambiente relajado y agradable para los usuarios al interactuar con el contenido de la web.

Consistencia y coherencia: utilizar una gama de azules en todo el diseño de la página web, incluyendo el texto, los encabezados y los elementos visuales, ayuda a crear una apariencia coherente y unificada. La consistencia visual mejora la legibilidad al proporcionar un diseño equilibrado y fácil de seguir ('lectura fácil').

- Accesibilidad: el color azul suele ser una opción segura en términos de accesibilidad, ya que es fácilmente distinguible para personas con diferentes tipos de discapacidades visuales. Además, al elegir tonos de azul que cumplan con las pautas de contraste de accesibilidad, se garantiza que el contenido del blog sea legible para la mayor cantidad de personas posible.

4.2.-CÓDIGO DE PROGRAMACIÓN

Durante el desarrollo y diseño de la página web, se tuvo la tarea de elegir las herramientas adecuadas para llevar a cabo el proyecto. Se tomaron decisiones en base a las necesidades y requisitos específicos de la aplicación, así como en consideraciones de eficiencia, seguridad y mantenibilidad. A continuación, se describirán las herramientas seleccionadas.

4.2.1.-Python

Python fue seleccionado como el lenguaje de programación principal debido a su versatilidad y capacidad para manejar eficientemente la información del centro y su funcionamiento.

Cuenta con una gran cantidad de bibliotecas y *frameworks* que facilitan el manejo y procesamiento de datos. Esto es especialmente útil para una página web de gestión de datos, ya que Python ofrece herramientas para realizar tareas como el análisis de datos, la manipulación de bases de datos y la implementación de algoritmos de aprendizaje automático.

Al utilizar Python, se ha podido crear una aplicación que permite el acceso ágil y desde cualquier lugar a la información del centro, mejorando así la coordinación y el funcionamiento de todos los departamentos y personas de apoyo implicadas.

Además de por su sintaxis sencilla y legible: Python se destaca por su sintaxis clara y legible, lo que facilita el desarrollo y la comprensión del código. Esto es especialmente valioso en un proyecto de desarrollo web, ya que permite una programación más eficiente y menos propensa a errores.

4.2.2.-Django

Django es un *framework* web de alto nivel que proporciona una estructura sólida y una serie de componentes predefinidos. Esto acelera el proceso de desarrollo al ofrecer características esenciales para la creación de una aplicación web, como el manejo de rutas, el acceso a la base de datos y la autenticación de usuarios.

Con Django, fue posible desarrollar rápidamente un blog funcional y seguro, además, permitió agilizar y mejorar los canales de comunicación gracias a las numerosas funcionalidades que ofrece de forma predeterminada.

4.2.3.-Bootstrap

Bootstrap es una biblioteca de diseño que ofrece componentes y estilos predefinidos, lo que permite crear una página web con un diseño adaptable y receptivo. Esto significa que la página se ajustará automáticamente a diferentes tamaños de pantalla y dispositivos, brindando una experiencia de usuario consistente y agradable en computadoras de escritorio, *tablets* y teléfonos móviles.

Además, proporciona una serie de estructuras y elementos de diseño reutilizables, lo que simplifica el proceso de diseño y ahorra tiempo en la creación de interfaces de usuario, lo cual nos facilitó el diseño en cuanto al front-end, principalmente en elementos como la barra de navegación y diseño de elementos básicos como botones o selectores.

4.2.4.-HTML, CSS, JavaScript y jQuery

Para desarrollar el front-end, además de incorporar Bootstrap, hemos utilizado las siguientes tecnologías básicas de una página web:

- HTML (HyperText Markup Language) es el lenguaje de marcado utilizado para estructurar el contenido de la página web. Proporciona la estructura básica de la página, definiendo los elementos y su disposición.
- CSS (Cascading Style Sheets) se utiliza para definir el aspecto visual de la página, como el color, la tipografía y el diseño.
- JavaScript es un lenguaje de programación que se utiliza para agregar interactividad y funcionalidad dinámica a la página web.

Estas tecnologías son fundamentales en el desarrollo web, ya que permiten crear una experiencia de usuario atractiva y funcional. HTML proporciona la estructura y el contenido de la página, CSS se encarga del diseño y la presentación visual, y JavaScript agrega interactividad y comportamiento dinámico.

4.2.5.-jQuery

jQuery es una biblioteca de JavaScript que simplifica el desarrollo web al proporcionar una serie de funciones y métodos predefinidos. Se basa en JavaScript y se utiliza para facilitar la manipulación del DOM (Modelo de Objetos del Documento), realizar animaciones, gestionar eventos, realizar peticiones AJAX y más. jQuery simplifica la escritura de código JavaScript al proporcionar una sintaxis más sencilla y abstraer algunas complejidades.

4.2.5.1.-FullCalendar

Para el desarrollo del calendario laboral explicado en puntos anteriores, utilizamos FullCalendar, una librería de JavaScript que permite crear y mostrar calendarios interactivos en páginas web. Ofrece una variedad de funciones, como agregar eventos, cambiar de vista (diaria, semanal, mensual), entre otros. FullCalendar se basa en jQuery y aprovecha las facilidades que ofrece para la manipulación del DOM y la interacción con los elementos de la página [5].

4.2.5.2.-DataTables

Para organizar los documentos de las fichas de apoyo y los protocolos, hemos utilizado DataTables, una librería de JavaScript que permite la manipulación y visualización avanzada de tablas de datos en páginas web. Proporciona características como ordenamiento, filtrado, paginación y búsqueda en tablas de datos. DataTables se basa en jQuery y utiliza su funcionalidad para facilitar la manipulación del DOM y ofrecer una experiencia de usuario interactiva [6].

4.3.-ENTORNO DE DESARROLLO Y DESPLIEGUE

La elección de un entorno de desarrollo adecuado es crucial para el desarrollo eficiente y efectivo de una página web. En este proyecto, se han utilizado las siguientes herramientas en el entorno de desarrollo:

4.3.1.-PostgreSQL

La elección de PostgreSQL como base de datos para el desarrollo de nuestra página web se basa en una previa evaluación de sus características y ventajas. PostgreSQL destaca por su estabilidad y confiabilidad, lo que garantiza la integridad de los datos almacenados.

En términos de escalabilidad, PostgreSQL es altamente flexible y capaz de adaptarse al crecimiento de nuestra aplicación web. Su capacidad para manejar grandes volúmenes de datos y su habilidad para escalar vertical y horizontalmente nos brindan la tranquilidad de que nuestro sistema puede manejar un crecimiento sustancial, en el caso de querer mejorar futuramente la página web o añadir nuevas funcionalidades.

La flexibilidad de PostgreSQL también es un factor clave en nuestra elección. Al permitir el almacenamiento de diversos tipos de datos, como actividades, cursos,

documentos PDF y contenido de blog, en una sola base de datos, simplifica nuestra arquitectura y mejora la eficiencia del sistema.

La seguridad es una preocupación fundamental en nuestro proyecto, especialmente al tratar con datos sensibles. PostgreSQL ofrece robustas opciones de seguridad, como autenticación de usuarios, control de acceso basado en roles y encriptación de datos, lo que nos permite proteger la confidencialidad y la integridad de la información almacenada.

Por último, la capacidad de PostgreSQL para integrarse con otras tecnologías y lenguajes de programación populares es un factor decisivo para nosotros. Esta integración sin problemas nos permite aprovechar las herramientas y librerías específicas de nuestro entorno de desarrollo, mejorando la eficiencia y productividad.

4.3.2.-PyCharm

La elección de PyCharm como Entorno de Desarrollo Integrado (IDE) para nuestro proyecto se debe principalmente a ser el entorno más conocido para programar con el lenguaje Python, usado en el desarrollo de nuestra herramienta.

Además de su amplia gama de funcionalidades. Ofrece un conjunto de herramientas completo que incluye un editor de código inteligente, depurador, soporte para pruebas unitarias, administrador de paquetes integrado y un sistema de control de versiones. Estas características nos permiten desarrollar, depurar y mantener nuestro código de manera eficiente y productiva.

PyCharm, desarrollado por JetBrains, es reconocido en la comunidad de Python como uno de los IDE más potentes y versátiles disponibles.

Otro aspecto importante de PyCharm es su soporte y comunidad activa. JetBrains, el equipo detrás de PyCharm, brinda un excelente soporte técnico y mantiene actualizaciones regulares del IDE. Además, la comunidad de usuarios de PyCharm es amplia y colaborativa, lo que nos brinda acceso a recursos, tutoriales y soluciones a problemas comunes.

En resumen, la elección de PyCharm como nuestro IDE se basa en su amplio conjunto de funcionalidades, su integración con el ecosistema de Python, su compatibilidad con frameworks y bibliotecas populares, su personalización y extensibilidad, y su soporte y comunidad activa.

4.3.3.-Google Drive

La necesidad de almacenar y organizar los archivos de nuestro proyecto de manera eficiente nos llevó a utilizar Google Drive. Esta plataforma nos permitió almacenar y acceder a diferentes tipos de documentos, como son los diagramas comentados anteriormente de Draw.io, las distintas versiones del proyecto web y los documentos con anotaciones relacionadas con la memoria del proyecto.

Creamos una carpeta compartida donde almacenar todo esto, de tal forma que las tutoras estarían al tanto en tiempo real de las modificaciones.

4.3.4.-Slack

La comunicación instantánea es fundamental para el éxito de nuestro proyecto, y Slack nos proporciona una interfaz intuitiva y fácil de usar para enviar mensajes en tiempo real. Esto nos permitió resolver dudas concretas relacionadas con el código o aspectos específicos de la memoria de manera rápida y efectiva. Pudimos intercambiar ideas, compartir fragmentos de código y obtener respuestas inmediatas, lo que aumenta la productividad y nos permite avanzar de manera eficiente.

Además de fijar las reuniones online que realizamos mediante Zoom.

4.3.5.-Zoom

Como parte fundamental de la organización durante la realización del proyecto, mantuvimos distintas reuniones online gracias al uso de Zoom.

La elección de Zoom como plataforma de reuniones online para nuestro proyecto se basa en su capacidad para facilitar la comunicación efectiva y la colaboración en equipo, a pesar de estar geográficamente dispersos. Zoom es ampliamente reconocido y utilizado en entornos profesionales debido a su confiabilidad y funcionalidades avanzadas.

4.4.-METODOLOGÍA

Como principal metodología en este proyecto, hemos seguido un enfoque en el usuario, es decir, Diseño Centrado en el Usuario (DCU). Esta metodología se centra en comprender las necesidades, expectativas y comportamientos de los usuarios finales, y utiliza este conocimiento para guiar el diseño y desarrollo de la interfaz y la funcionalidad del sitio web. A continuación, se detallan las ventajas de utilizar esta

metodología en mi proyecto y se explica el proceso que seguí para enfocar mi trabajo siguiendo el DCU [7].

Ventajas del Diseño Centrado en el Usuario (DCU):

- Experiencia de usuario mejorada: el enfoque del Diseño Centrado en el Usuario garantiza que nuestra página web se adapte a las necesidades y expectativas de los usuarios. Esto resulta en una experiencia de usuario más satisfactoria, lo que a su vez aumenta la retención de usuarios y mejora la percepción de calidad del sitio web.
- Aumento de la usabilidad: al comprender cómo los usuarios interactúan con la página web, es posible diseñar una interfaz intuitiva y fácil de usar. El DCU permite identificar puntos problemáticos y realizar mejoras que optimicen la navegación y la interacción del usuario, lo que hace que aumente la eficiencia de uso. Además, nos hemos centrado en adoptar un papel importante en cuanto a la accesibilidad, lo que también beneficia dicha usabilidad, adecuándonos a las necesidades de nuestros distintos tipos de usuarios.
- Mayor eficacia y eficiencia: conociendo las necesidades y objetivos de los usuarios, pudimos diseñar una página web que los ayude a lograr sus tareas de manera más rápida y efectiva. Esto se traduce en una mayor eficacia y eficiencia en la realización de acciones dentro del sitio web, lo que mejora la productividad y la satisfacción de los usuarios.
- Reducción de costos y tiempos de desarrollo: sabiendo las necesidades del usuario desde las etapas iniciales del proyecto, se evita el trabajo innecesario y redundante y los costos asociados con cambios significativos en etapas avanzadas del desarrollo. El DCU permite identificar problemas y realizar ajustes tempranos, lo que acelera el proceso de desarrollo y reduce los costos asociados.

En cuanto a el proceso seguido para enfocar el proyecto siguiendo el DCU podemos hacer un seguimiento mediante los siguientes puntos:

- Investigación de usuarios: inicié el proceso mediante la ejecución de investigaciones centradas en los usuarios finales de mi página web, realizando entrevistas principalmente a los trabajadores de uno de los centros de ASPRODES de Salamanca. Esta fase permitió obtener una comprensión profunda de las necesidades, objetivos y comportamientos de los usuarios. A través de diversas técnicas de investigación, como entrevistas, encuestas y

análisis de datos, obtuve información valiosa para fundamentar las decisiones de diseño y definir los objetivos del sistema vistos anteriormente.

- Definición de requisitos: sobre la base de los datos recopilados durante la investigación de usuarios, procedí a establecer de manera rigurosa los requisitos y objetivos del proyecto combinando el DCU con la elicitación de requisitos de Durán y Bernárdez. Estos requisitos se fundamentaron en las necesidades y expectativas de los usuarios, lo que garantizó un enfoque centrado en su experiencia y maximizó la relevancia y utilidad de la página web.
- Diseño conceptual: con el propósito de generar ideas y conceptos de diseño sólidos y eficaces, utilizando herramientas y metodologías reconocidas, como diagramas de flujo, *wireframes* y prototipos, materialicé visualmente las posibles soluciones para la página web. Este enfoque me permitió explorar diferentes alternativas y evaluar su idoneidad en términos de usabilidad y experiencia del usuario.
- Evaluación y prueba: una vez que se dispuso de prototipos funcionales, llevé a cabo pruebas de usabilidad con usuarios. Recopilé datos objetivos para evaluar la eficacia de las soluciones propuestas. Las conclusiones sacadas mediante las pruebas sirvieron para ajustar y mejorar el diseño de la web.
- Implementación y desarrollo: finalmente, basándome en los resultados de las pruebas y evaluaciones, procedí a la implementación y desarrollo de la página web, asegurándome de mantener la coherencia con los principios fundamentales del DCU. Esta fase involucró la traducción de los diseños refinados en código funcional, la integración de [tecnologías](#) anteriormente analizadas y la optimización del rendimiento general, garantizando así una experiencia de usuario excepcional y cumpliendo con los requisitos establecidos.

5.-ASPECTOS RELEVANTES DEL DESARROLLO

En esta sección se tratará el proceso de desarrollo del Trabajo de Fin de Grado desde la especificación de requisitos, pasando por la búsqueda de necesidades hasta el despliegue en el entorno real, entre otros puntos a destacar.

5.1.-PLANIFICACIÓN TEMPORAL

La planificación temporal ha sido de gran utilidad para conseguir el desarrollo de este Trabajo Fin de Grado a tiempo. Nos ha permitido organizar y estructurar el proyecto de manera efectiva, estableciendo las tareas, los plazos y los recursos necesarios para su realización. Gracias a esta planificación, hemos tenido una visión clara de cómo abordar el trabajo y hemos podido mantenernos en el camino correcto.

La gestión del tiempo ha sido un aspecto clave en nuestro proyecto. Gracias a la planificación temporal, hemos podido asignar tiempos adecuados a cada tarea, estableciendo prioridades y evitando retrasos innecesarios. Esto nos ha permitido mantener un ritmo constante y asegurarnos de cumplir con los plazos establecidos.

Otro beneficio importante de la planificación temporal ha sido la identificación de dependencias entre las diferentes tareas. Al visualizar las relaciones de precedencia, hemos podido coordinar nuestras actividades de manera secuencial y evitar conflictos. Esto ha sido fundamental para mantener la coherencia y el orden en nuestro trabajo.

La planificación temporal también nos ha brindado un mayor control y seguimiento del proyecto. Hemos podido monitorear constantemente nuestro progreso real en comparación con el planificado, lo que nos ha permitido realizar ajustes y tomar decisiones informadas en el camino. De esta manera, hemos podido garantizar una ejecución eficiente y exitosa de nuestro TFG.

Todo esto y la estimación del esfuerzo previa, lo encontramos más detallado en el [Anexo III - Plan de Proyecto Software](#).

A continuación, se presenta el esquema de la planificación en tareas siguiendo el Proceso Unificado [8]:

Semanas

1. FASE DE INICIO

1.1.	Preparación de los lenguajes y herramientas	0.5
1.2.	Planificación temporal	0.5
1.3.	Especificación de requisitos del Software	2
	1.3.1. Definición de los objetivos del proyecto	
	1.3.2. Especificación de requisitos del sistema	
1.4.	Proceso de diseño centrado en el usuario	2
	1.4.1. Búsqueda de necesidades	
	1.4.2. Análisis de la competencia	
	1.4.3. Definición de la audiencia	
	1.4.4. Primer boceto de diseño	
2. FASE DE ELABORACIÓN		
2.1.	Inicio de los primeros anexos	1
2.2.	Refinamientos de los requisitos y actores del sistema	2
2.3.	Diagramas casos de uso	1
2.4.	Conceptualización y diseño	3
	2.4.1. Aspectos del diseño	
	2.4.2. Wireframes	
	2.4.3. Prototipado digital	
	2.4.4. Pruebas de usuario	
3. FASE DE CONSTRUCCIÓN		
3.1.	Continuación anexos	0.5
3.2.	Implementación código	4
3.3.	Pruebas	0.5
4. FASE DE TRANSICIÓN		
4.1.	Revisión final requisitos	0.5
4.2.	Revisión final diagramas	0.5
4.3.	Revisión final funcionalidad	0.5
4.4.	Prueba final de la aplicación	0.5

4.5. Mejoras de la aplicación	1.5
4.6. Elaboración de la memoria final	2.5
4.7. Últimas revisiones y entrega	1

Tabla 1. DISTRIBUCIÓN DEL TIEMPO A LO LARGO DE TODO EL PROYECTO

5.2.-ANÁLISIS DE LA COMPETENCIA

En la fase de análisis de la competencia, hemos realizado un mapeo exhaustivo de diversas aplicaciones utilizadas en centros de personas con discapacidad. El objetivo de este análisis comparativo fue evaluar las herramientas existentes en el mercado y establecer un punto de referencia para nuestra propuesta de mejora en este Trabajo de Fin de Grado.

Hemos identificado varias herramientas utilizadas en centros de mayores o de plena inclusión, como Resiplus, Gerosalus, IxisSocial Gest, Famileo, Soft4care, Tipsa y Ontimize Solutions, las cuales podemos consultar información más detallada en el [Anexo I - Diseño Centrado en el Usuario](#)

Estas herramientas se utilizan para gestionar y facilitar diversas tareas y procesos en dichos centros.

Por ejemplo, Resiplus es una herramienta diseñada específicamente para la gestión de residencias de mayores, permitiendo llevar a cabo actividades como el seguimiento de residentes, control de ingresos, planificación de cuidados, gestión de farmacia y proveedores, entre otros aspectos relacionados con la administración de la residencia [9].



Figura 1. LOGO RESIPLUS

Gerosalus, por su parte, está enfocado en la gestión de clínicas, hospitales y centros médicos, proporcionando funcionalidades para el control y cuidados de salud, la gestión de servicios y el seguimiento individualizado de los usuarios.



FIGURA 2. GEROSALUS SOFTWARE

IxisSocial Gest se centra en la planificación centrada en la persona y ofrece herramientas para la gestión de servicios de entidades, inserción sociolaboral, control y cuidados de salud, así como la organización de actividades y la comunicación con usuarios y familias[10].



FIGURA 3. LOGO IXISSOCIALGEST

Estas herramientas existentes en el mercado proporcionan soluciones para la gestión eficiente de los centros de mayores o de plena inclusión, abordando aspectos como la administración, la planificación de cuidados, la comunicación con las familias y la gestión de servicios.

5.3.-BÚSQUEDA DE NECESIDADES

El proyecto tiene como objetivo mejorar la comunicación y organización de un centro para personas con discapacidad. Surgió a raíz de la experiencia de un familiar que trabaja en uno de estos centros en Salamanca y de las problemáticas que comentaba

sobre la falta de organización, desorden de archivos y la falta de seguridad en los sistemas utilizados.

Tras realizar entrevistas y familiarizarse con el sistema informático existente, se identificaron varias áreas de mejora. La directora del centro expresó la necesidad de contar con un registro de deposiciones para detectar posibles alteraciones en los residentes, así como protocolos actualizados para los cuidadores y facilitar la organización de actividades y el mantenimiento. También se observó que la distribución de información se realizaba a través de una red compartida con numerosos archivos de Excel y Word, lo que planteaba problemas de seguridad, ya que cualquier persona con acceso al ordenador podía modificarlos.

La falta de seguridad y la desorganización generaban inconvenientes para los trabajadores, que a menudo se perdían al buscar archivos específicos. Además, el exceso de llamadas telefónicas por parte de los familiares para obtener información sobre los residentes generaba una carga adicional para el personal, que se veía interrumpido en sus tareas diarias.

Como resultado, surgió la idea de crear una plataforma que permitiera una comunicación eficiente entre técnicos, cuidadores, familiares y residentes. Además, se propuso establecer un nuevo enfoque de trabajo para recopilar toda la documentación y archivos necesarios en la organización del centro, garantizando la seguridad y el acceso controlado para cada tipo de usuario.

En resumen, el proyecto busca implementar una solución que mejore la comunicación, la seguridad y la organización en el centro para personas con discapacidad, brindando a los diferentes actores la capacidad de acceder a la información relevante de manera eficiente y asegurando la confidencialidad de los datos.

Para más información consultar [Anexo I - Diseño Centrado en el Usuario](#)

5.4.-DEFINICIÓN DE LA AUDIENCIA

Mediante las técnicas de la definición de arquetipos de usuarios y escenarios de uso hemos ejemplificado y definido la audiencia del proyecto. A continuación, veremos unos pequeños ejemplos de estos para ponernos en contexto, pero para toda la información y situaciones consultar [Anexo I - Diseño Centrado en el Usuario](#)

5.4.1.-ARQUETIPOS DE USUARIOS

- Terapeuta ocupacional: Alfonso, un terapeuta ocupacional creativo y apasionado del deporte, que se dedica a crear y organizar actividades para trabajar con los residentes.
- Cuidadora: María Teresa, una cuidadora en un centro de día de personas con discapacidad psicosocial, que tiene dificultades para encontrar los documentos de registro y no puede interpretar bien los registros de asistencia en Excel.
- Conductor: Ángel, un conductor de autobús de una residencia de inclusión social, que necesita conocer los horarios y rutas de trabajo con precisión y tiene dificultades para anotar los cambios mientras conduce.
- Familiar de persona con discapacidad: Luisa, una madre que trabaja en horarios incompatibles con las visitas y llamadas al centro, y desea estar más conectada con la vida diaria de su hijo.
- Persona con discapacidad: Manuel, un residente con discapacidad sensorial e intelectual, que busca comunicarse con su cuidadora favorita, Carmen, y compartir sus actividades y sentimientos a través de una plataforma.

5.4.2.-ESCENARIOS DE USO

- Alfonso, como miembro del equipo técnico, utiliza la plataforma para crear y organizar actividades para analizar los cambios y necesidades de las personas.
- María Teresa, como cuidadora, accede a la plataforma para buscar y apuntarse a cursos de formación, ya que a veces no se entera de las novedades debido a la falta de comunicación clara.
- Ángel, como conductor, consulta la plataforma para conocer sus horarios y rutas de trabajo de manera precisa, evitando olvidos y asegurándose de recoger a todas las personas.
- Luisa, como familiar, utiliza la plataforma para acceder a información actualizada sobre las actividades y el bienestar de su hijo residente, ya que sus horarios de trabajo le impiden visitar el centro con regularidad.
- Manuel, como residente, se comunica con su cuidadora favorita, Carmen, a través de la plataforma para compartir sus actividades diarias y sentimientos, manteniéndola informada sobre su vida en el centro.

Teniendo en cuenta estos ejemplos, se puede destacar el caso de Alfonso como representativo de los arquetipos de usuarios, ya que ilustra la importancia de la

plataforma en la creación, organización y análisis de actividades para satisfacer las necesidades de los residentes.

5.5.-ESPECIFICACIÓN DE REQUISITOS

La especificación de requisitos es fundamental para definir y detallar los diferentes aspectos y funcionalidades que se esperan del sistema desarrollado. A través de este apartado, el cual hemos podido definir mejor gracias al previo análisis de la competencia, el needfinding y definición de la audiencia, se establecen los requisitos de información, los requisitos no funcionales y los requisitos funcionales, los cuales describen de manera exhaustiva las necesidades y expectativas del sistema.

En primer lugar, la sección de Requisitos de Información(IRQ) proporciona una descripción detallada de la información que el sistema debe manejar. Esta información puede incluir datos específicos, como nombres, fechas, ubicaciones, entre otros, así como los formatos en los que se deben presentar y almacenar estos datos. Mediante las tablas correspondientes, que se pueden consultar en el [Anexo II - Definición y análisis de los requisitos](#), conseguimos de manera organizada y clara diferenciar los distintos elementos de información requeridos y sus características asociadas. Los requisitos de información definidos son:

- Usuario: correspondiente a los datos de cada usuario registrado (nombre, contraseña, centro, rol)
- Calendario laboral: corresponde al calendario de cada usuario donde se reflejarán los eventos a los que está inscrito o tiene asignados en un horario (nombre, evento, horario, ruta)
- Evento: correspondiente a los tipos de evento que se reflejan en el calendario (calendario, descripción, lugar desarrollo, hora, fecha)
- Ruta: correspondiente a la ruta que sigue un conductor donde recoge a usuarios en cada parada (nombre, fecha, parada)
- Parada: correspondiente a las paradas ordenadas de la ruta que sigue un conductor donde recoge a usuarios (ruta, dirección, hora recogida, usuario)
- Curso: correspondiente a las actividades que se realizarán en el centro (nombre, descripción, lugar desarrollo, hora, duración, aforo)

- Actividad: correspondiente a las actividades que se realizarán en el centro (nombre, descripción, lugar desarrollo, lugar salida, hora salida, hora llegada, duración, aforo)
- Protocolo: correspondiente a los pdfs de protocolos con normas del centro (título, responsable, versión, fecha, descripción)
- Ficha de apoyo: correspondiente a los pdfs de seguimiento diario de cada usuario (nombre usuario, responsable, versión, fecha, descripción)
- Publicación: correspondiente a las entradas del blog (título, texto, fecha, hora, autor)
- Comentario: correspondiente a los comentarios del blog para cada publicación (texto, autor, fecha, hora)

Por otro lado, los Requisitos No Funcionales (NFR) describen las características y atributos que el sistema debe poseer, además de las funcionalidades específicas. Estos requisitos se centran en aspectos como la seguridad, la usabilidad, el rendimiento, la escalabilidad, la disponibilidad y otros criterios relevantes para el funcionamiento eficiente y efectivo del sistema, los cuales vemos más detalladamente en el [Anexo II - Definición y análisis de los requisitos](#). Los definidos para nuestro sistema concretamente son:

- Protección de datos
- Copia de seguridad
- Disponibilidad tecnológica
- Interacción usuario-datos
- Interfaz intuitiva
- Concurrencia
- Mantenibilidad
- Facilidad de manejo y acceso

Finalmente, los Requisitos Funcionales (UC) representan los casos de uso del sistema y describen cómo este debe comportarse en diferentes situaciones o interacciones con los usuarios. Cada caso de uso se detalla de manera individual, especificando las acciones que pueden realizar los usuarios, los resultados esperados y las condiciones o restricciones asociadas, como iniciar sesión, cerrar sesión, CRUD de cursos, actividades, protocolos...Estos requisitos funcionales se presentan de manera estructurada y organizada, facilitando su comprensión y seguimiento. Para información

más detallada de cada caso de uso consultar [Anexo II - Definición y análisis de los requisitos](#).

Además, hemos diseñado diagramas de casos de uso para reflejar de forma gráfica estos UC y así ayudar con la posterior implementación de relaciones en el sistema [11].

Algún ejemplo de los apartados más complejos:

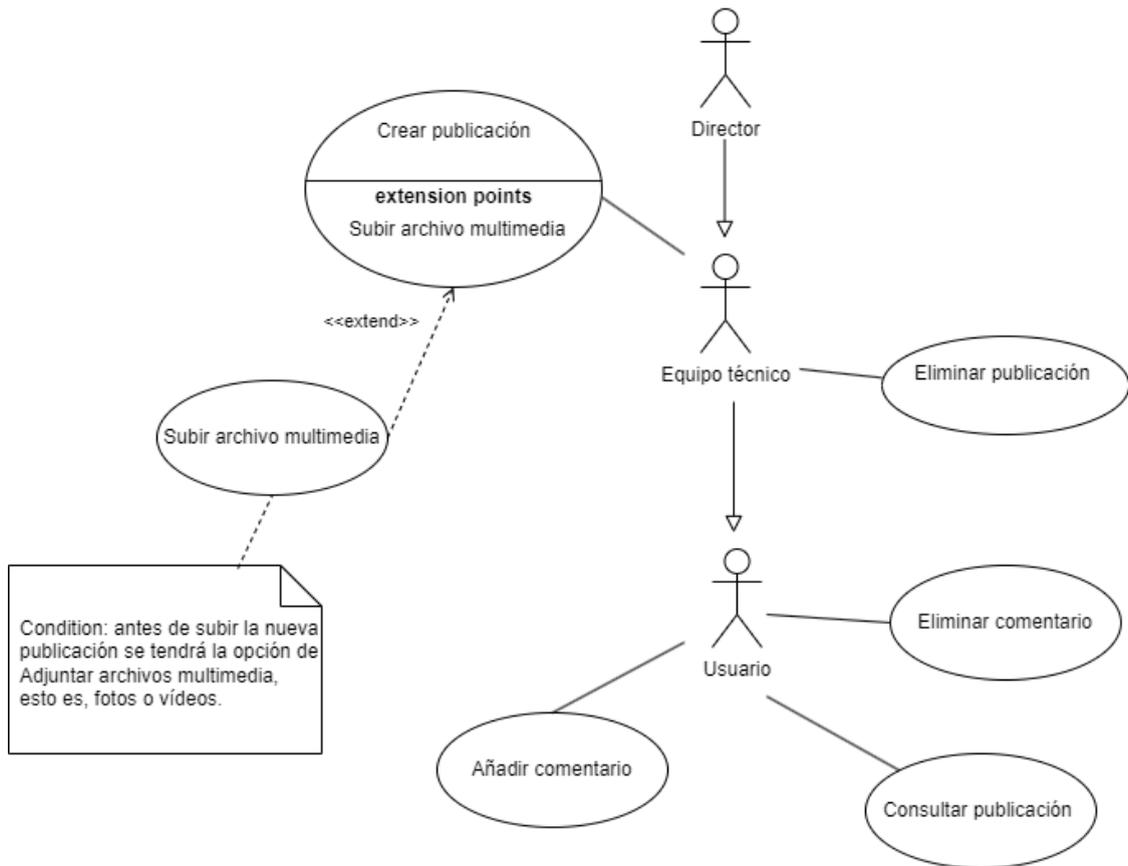


FIGURA 4. DIAGRAMA GESTIÓN TABLÓN DE ANUNCIOS

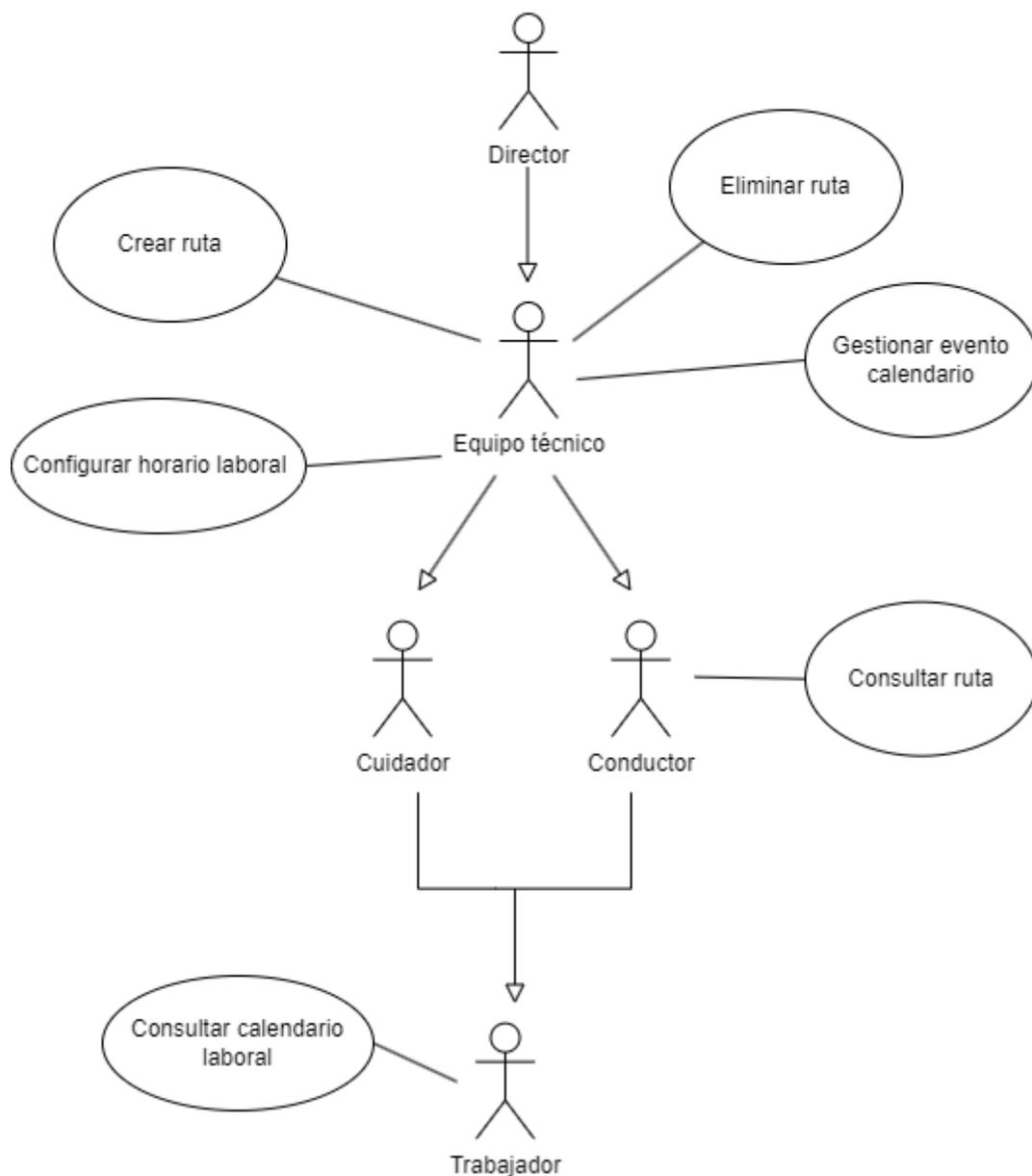


FIGURA 5. DIAGRAMA GESTIÓN CALENDARIO

En conjunto, estas tres secciones proporcionan una visión completa de los requisitos del sistema, abarcando tanto la información necesaria, los aspectos no funcionales que debe cumplir y las funcionalidades específicas que se esperan. Esta especificación detallada asienta las bases para el diseño y desarrollo del sistema, asegurando de esta forma que se cumpla con los criterios y expectativas establecidas. Además, mediante el uso de las tablas definidas en la metodología Durán y Bernárdez se proporciona una forma clara y ordenada de presentar la información, facilitando la comprensión y revisión de los requisitos de forma clara y sencilla.

Cada requisito tiene asociado un actor distinto, ya que en nuestro sistema accederán varios tipos de usuarios según su rol (técnico, cuidador, conductor, familiar, usuario).

4.5.1.-DEFINICIÓN DE ACTORES DEL SISTEMA

Cada requisito visto en el apartado anterior tiene asociado un actor distinto, ya que en nuestro sistema accederán varios tipos de usuarios según su rol (técnico, cuidador, conductor, familiar, usuario).

El actor director corresponde al superusuario de Django, se destacada esa figura porque es el único que puede crear los horarios, ya que los crea para los del equipo técnico inclusive, aunque los técnicos tengan todo el resto de poder sobre el sistema.

El usuario y el familiar tendrán simplemente acceso al blog, con la intención de fomentar la comunicación comentada a lo largo de la memoria.

En cuanto a los actores de conductor y cuidador, son los trabajadores del centro, de los cuales el equipo técnico es responsable, estos no tienen ningún poder en cuanto a la gestión de los distintos apartados. La diferencia fundamental entre cuidador y conductor es que éste último tiene acceso a las rutas que debe seguir como conductor para recoger a los usuarios y llevarlos al correspondiente centro.

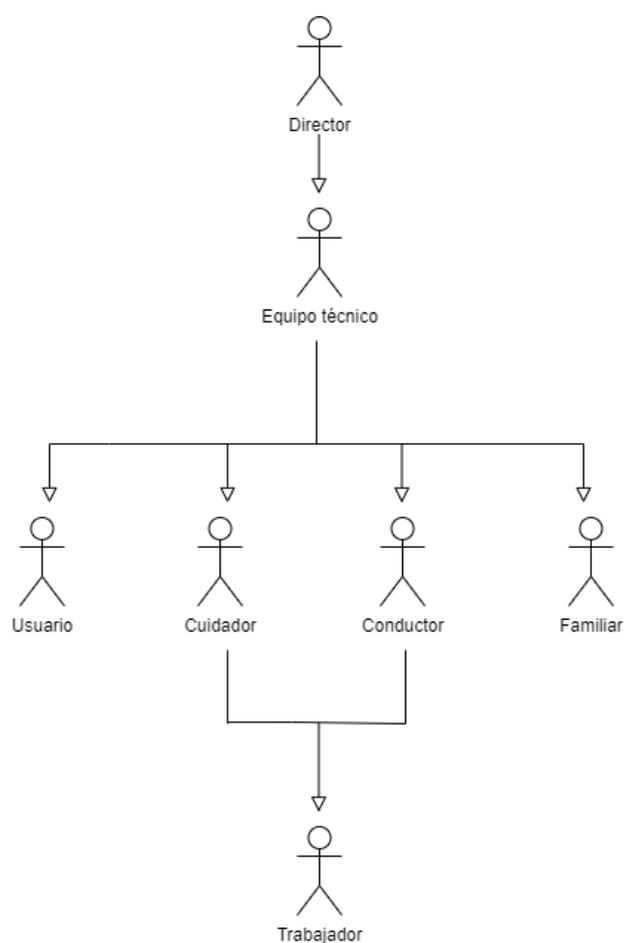


FIGURA 6. DIAGRAMA ACTORES

5.6.-CONCEPTUALIZACIÓN

En la conceptualización de mi página web, además de utilizar la herramienta de diseño Figma para crear un primer prototipo digital, se tomaron decisiones de diseño clave que influyeron en la apariencia y funcionalidad del sitio. A continuación, se explicarán algunas de las decisiones de diseño más relevantes, centrándose especialmente en la parte del blog.

Una de las decisiones importantes fue la inclusión de un blog como parte integral y principal de la página web, queriendo destacar la figura de los usuarios del centro con discapacidad 'NOSOTROS'. Reconociendo la importancia de proporcionar contenido relevante y actualizado, se decidió implementar un espacio dedicado para compartir artículos, noticias, recursos, actividades, menús semanales... cualquier tema que pueda ayudar a la comunicación e integración entre todos los representantes. Esta sección del blog se diseñó con una estructura clara y fácil de navegar, con un título en

cada entrada destacable y un apartado donde aparecen las últimas publicaciones para realizar una búsqueda más sencilla y que permiten a los usuarios encontrar rápidamente el contenido que más les interesa.

Para la presentación visual del blog, se optó por un diseño limpio y moderno que resalte el contenido principal. Se utilizó una disposición de cuadrícula ordenada para mostrar las publicaciones más recientes en la página de inicio del blog, brindando a los visitantes una visión general rápida de los temas y títulos destacados. Además, se incorporaron elementos de navegación intuitivos, como botones de "leer más" para acceder a la información concreta de cada publicación.

En cuanto al diseño de las páginas individuales de los artículos del blog, se enfocó en proporcionar una experiencia de lectura atractiva y agradable. Se seleccionó cuidadosamente una tipografía legible y se ajustaron los tamaños y espaciados de texto para garantizar una lectura cómoda. También se tuvo en cuenta la inclusión de elementos visuales complementarios, como imágenes, gráficos o citas destacadas, para enriquecer el contenido y hacerlo más atractivo visualmente.

Otra decisión relevante fue la integración de funcionalidades de interacción en el blog. Se incluyeron opciones para que los usuarios puedan comentar, compartir y marcar los artículos como favoritos. Esto fomenta la participación de la comunidad, facilita la difusión del contenido y permite a los usuarios guardar los artículos que deseen leer más tarde.

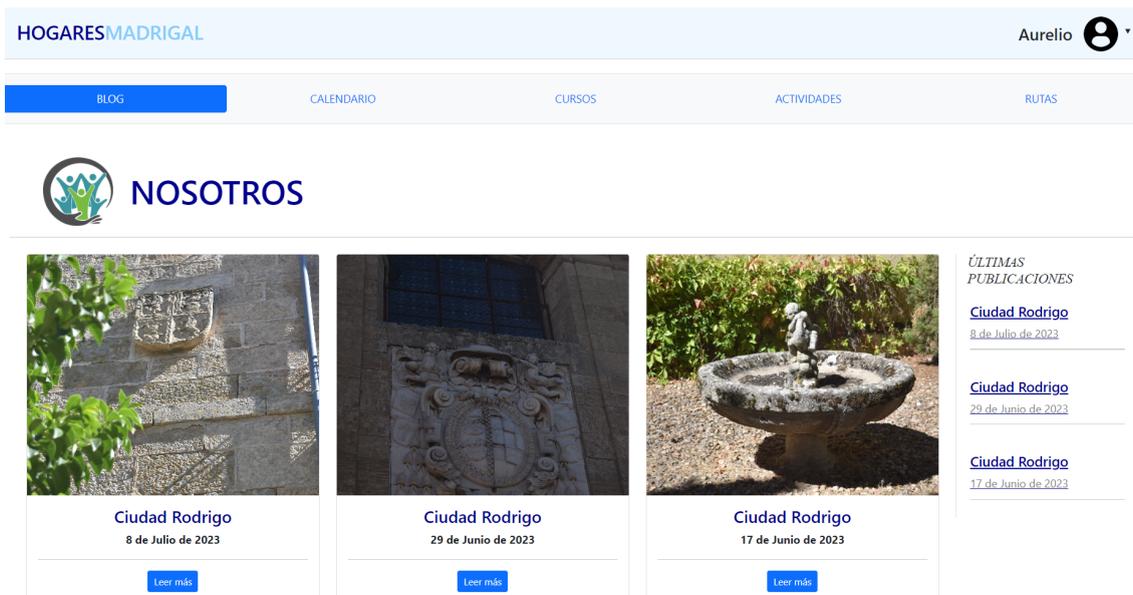


FIGURA 7. EJEMPLO BLOG INTERFAZ

También destacar en el apartado del calendario, la posibilidad de filtrar por categorías, seleccionando o deseleccionando éstas para mostrar en la vista del calendario, de tal modo, que en caso de sólo querer ver el horario, pueda verlo claramente sin necesidad de tener presentes el resto de eventos que puedan llevar a confusión. Aunque, para mejorar la claridad y hacer el calendario más intuitivo, se ha añadido un color representativo a cada evento para identificar cada tipo.

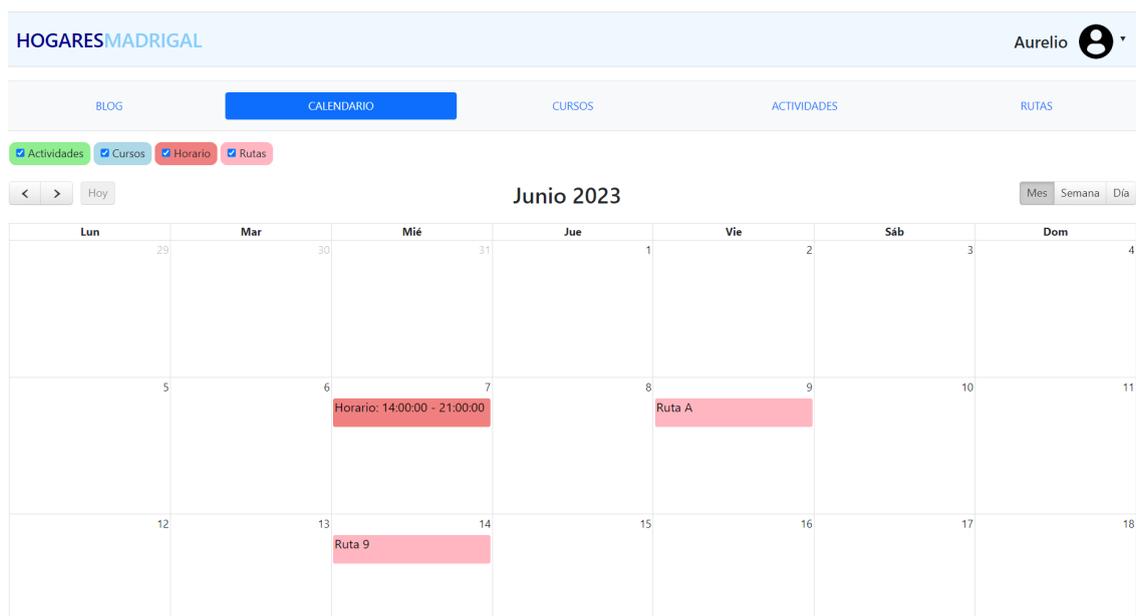


FIGURA 8. EJEMPLO CALENDARIO INTERFAZ

En cuanto al prototipado digital, Figma nos permitió visualizar y experimentar con la estructura, el diseño, la navegación y la distribución de los diferentes elementos de la página antes de iniciar la implementación en Django.

El objetivo principal de esta etapa conceptual fue definir la arquitectura de información y establecer la disposición de los distintos apartados y elementos de la página web. Utilizamos Figma para crear *wireframes* y diseños de baja fidelidad que nos sirvieron como referencia visual y nos ayudaron a comprender cómo se organizarían y presentarían los contenidos.

Mediante Figma, pudimos diseñar las distintas secciones y páginas de la página web, definiendo la ubicación de los elementos clave, como el menú de navegación, el contenido principal, los formularios y los elementos interactivos. También exploramos diferentes opciones de diseño, paletas de colores y tipografías para definir la apariencia visual general de la página.

Al utilizar Figma como herramienta de prototipado, pudimos compartir y colaborar con las tutoras. Esto nos brindó una base sólida para el desarrollo, ya que teníamos una comprensión clara de cómo se organizarían y se distribuirían los distintos apartados y elementos en la página web.

En resumen, la etapa de conceptualización de mi página web involucró el uso de Figma como herramienta para crear un prototipo digital. Esto nos permitió definir la arquitectura de información, establecer la disposición de los apartados y elementos, y tener una referencia visual sólida antes de comenzar la implementación en Django.

En el *Anexo I - Diseño Centrado en el Usuario* se explica con detalle la decisión de utilizar esta herramienta y los pasos seguidos más detalladamente.

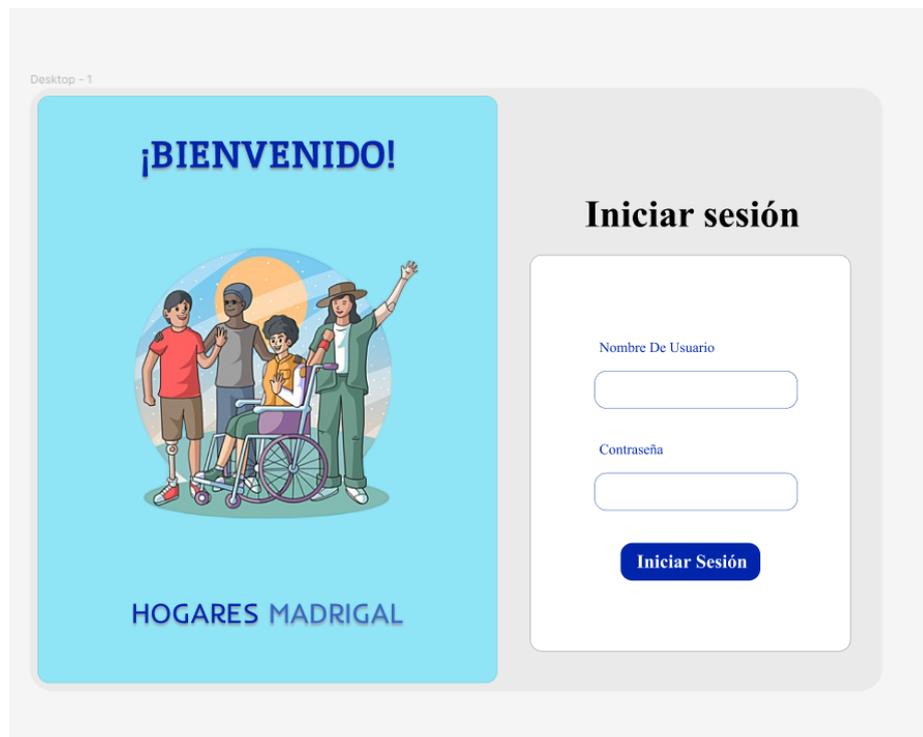


FIGURA 9. FIGMA INICIO SESIÓN

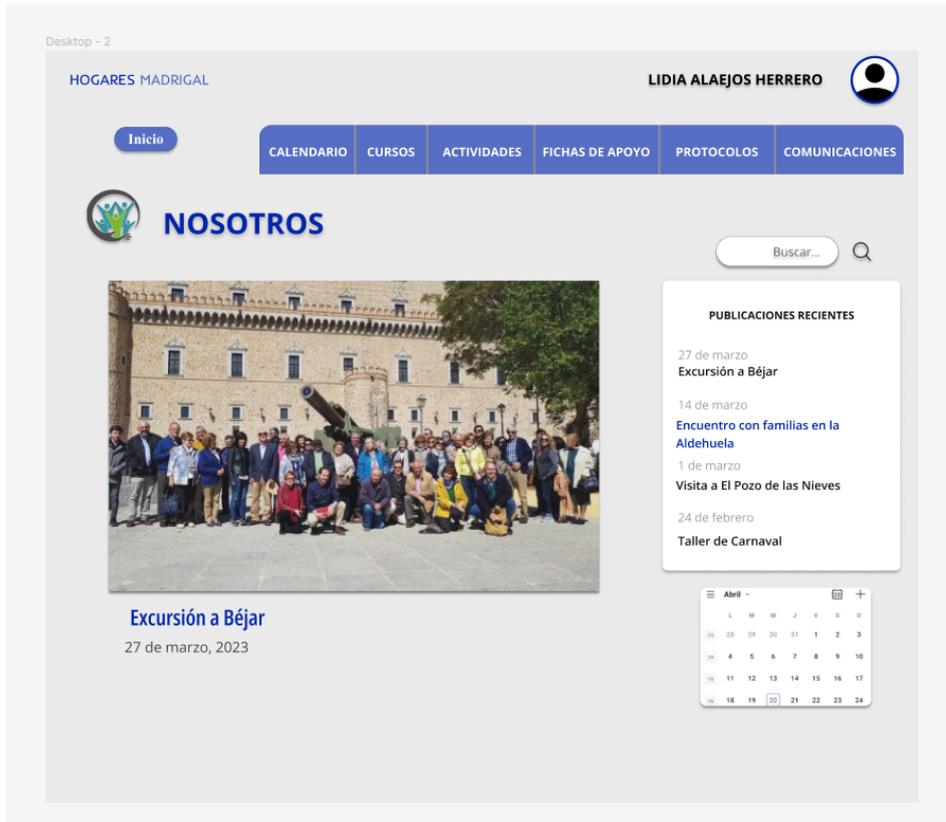


FIGURA 10. FIGMA BLOG

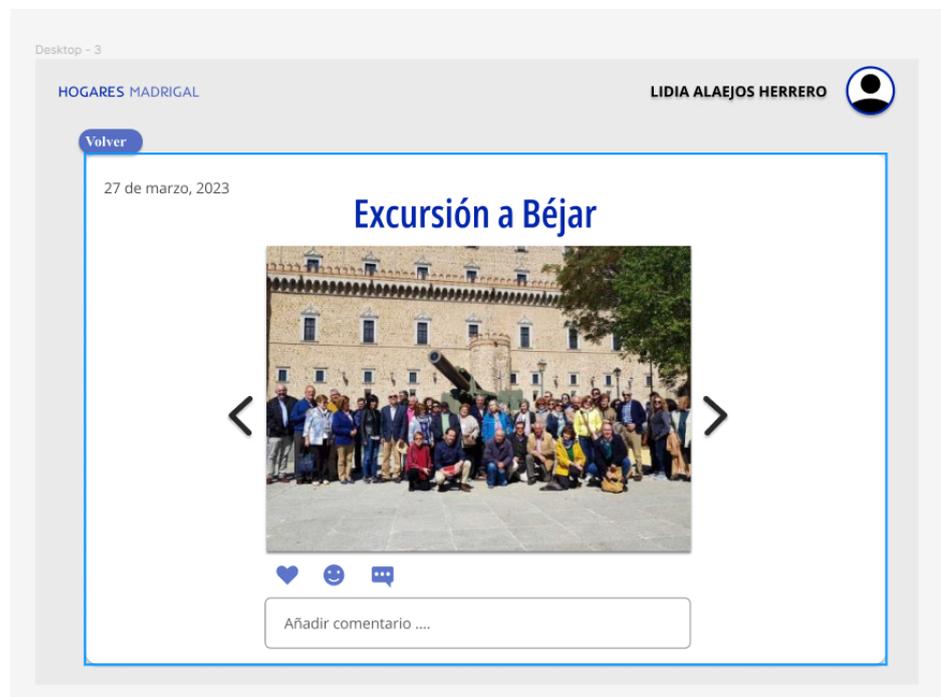


FIGURA 11. FIGMA BLOG DETALLE

5.7.-GESTIÓN DE DATOS

El almacenamiento y tratado de los datos se ha llevado a cabo mediante la base de datos Postgres, explicada anteriormente en el apartado de [técnicas y herramientas](#).

Compuesta por trece tablas, cada una encargada de guardar los atributos relevantes de cada clase, para el correcto funcionamiento de la página web, almacenando todo tipo de datos y archivos multimedia.

La estructura es la siguiente:

- Usuario: almacenará los datos personales de las cuentas de los usuarios registrados.
- Rol: especificación del tipo de usuario para que según el rol que adquiera el usuario tendrá unos u otros contenidos la página.
- Curso: almacenará la información necesaria de los cursos de formación que se realizan en el centro, como su fecha, aforo, etc.
- Actividad: almacenará la información necesaria de las actividades que se realizan en el centro, como su fecha, lugar de realización, aforo, etc.
- Evento calendario: almacenará los tipos de eventos que se añadirán en el calendario de cada usuario según sus datos guardados.
- Turno: almacenará los datos correspondientes al turno de trabajo (mañana, tarde, noche, turno partido) de cada trabajador.

- Ruta: almacenará todo lo necesario para que los conductores conozcan los datos de la ruta que tienen en esa fecha (con paradas y usuarios a recoger en cada una).
- Parada: almacenará los datos relevantes de cada parada que haya en una ruta para un conductor.
- Protocolo: almacenará los datos referentes a los pdfs que especifican los protocolos del centro.
- Ficha de apoyo: almacenará los datos referentes a los pdfs que especifican las fichas de apoyo de cada usuario con discapacidad interno del centro.
- Publicación: almacenará los archivos multimedia e información relevante de cada entrada del blog.
- Comentario: almacenará los datos de un comentario realizado por un usuario registrado a publicar en cada publicación del blog.

Para llegar a la conclusión de dejar definitivamente la base de datos implementada como se ha definido, nos encontramos algún problema de cómo relacionar ciertas clases con otras mediante *ForeignKey* o de cómo asociar cada clase con la de Usuario debido al atributo rol, es decir, que al tener distintos tipos de usuarios tuvimos que revisar en varias ocasiones esas relaciones para darnos cuenta que algunas eran innecesarias dependiendo del tipo de usuario o que por el contrario, se necesitaba establecer una relación directa mediante *ForeignKey*.

En un primer modelo no integramos la clase 'Turno' cuando pensamos en establecer una clase para guardar los datos referentes al horario de cada trabajador, pero avanzando en el desarrollo del código en Django fuimos conscientes de la necesidad de ese modelo, ya que en todo horario laboral se puede tener distintos turnos (de mañana, tarde, noche o partido) según el día o semana, y más en centros como éste que se necesitan cuidadores durante todo el día.

Además, en ese primer modelo, la clase que ahora se llama 'EventoCalendario' se llamó 'Calendario laboral', lo cual al repasar con las tutoras este primer diagrama diseñado, pensamos que era mejor llamarlo 'EventoCalendario' ya que incluyendo la palabra 'laboral' podría causar confusión, puesto que esta clase no sólo almacenaría los horarios de cada usuario del sistema, sino que también serviría para reflejar en el

calendario todo tipo de eventos, como son las actividades, los cursos, y las rutas para los conductores.

A continuación, dejamos imágenes del primer modelo incompleto, y la segunda imagen referente al modelo definitivo del diagrama de entidad-relación [12] con el cual trabajaríamos para definir nuestros modelos en Django y Postgres.

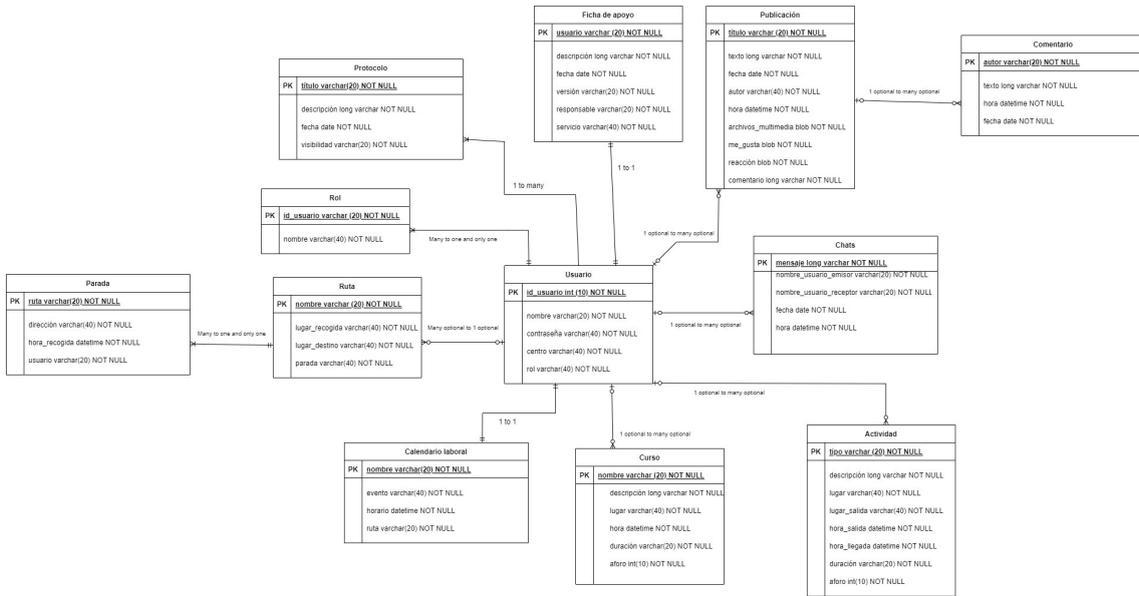


FIGURA 12. DIAGRAMA ENTIDAD-RELACIÓN NO DEFINITIVO

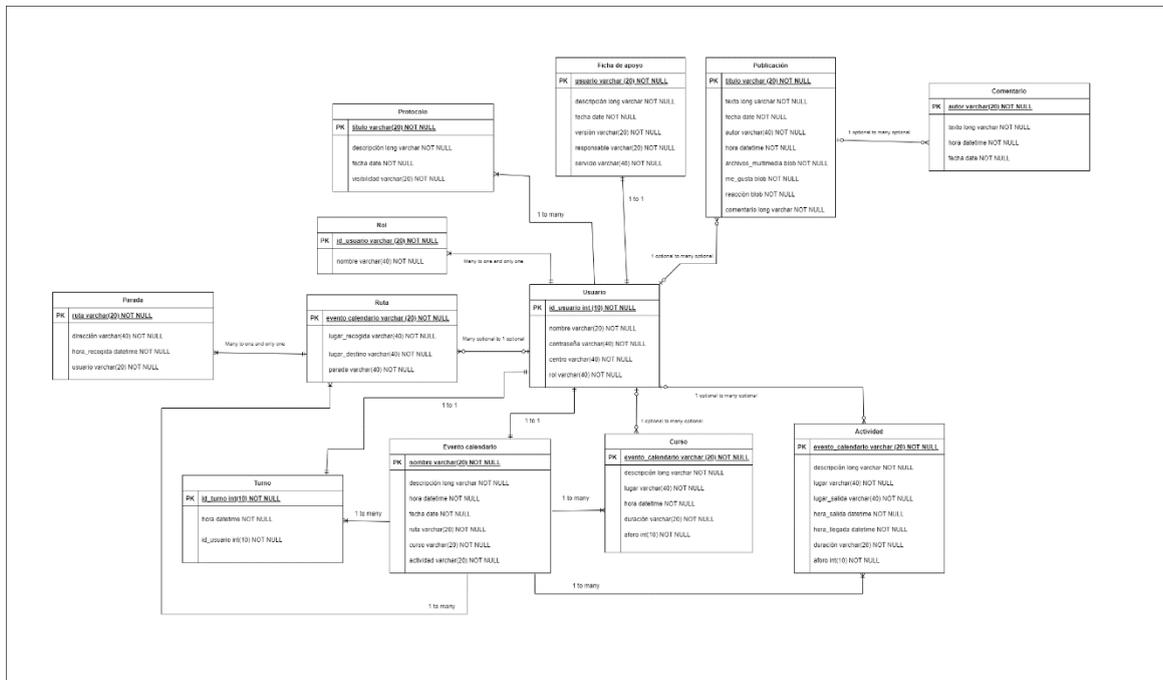


FIGURA 13. DIAGRAMA ENTIDAD-RELACIÓN DEFINITIVO

5.8.-FUNCIONALIDAD DEL SISTEMA

Durante el desarrollo del proyecto hemos encontrado algunas dificultades, a continuación, haremos una puntualización de las principales, explicando la funcionalidad que decidimos agregar para el adecuado funcionamiento de éstos en nuestra web, de manera coherente con los objetivos y necesidades de los usuarios, para brindar una experiencia satisfactoria y eficiente al interactuar con la página.

Como primera instancia, quisimos incorporar a la web un apartado de 'comunicación/chat' en tiempo real, lo cual, a priori no pensamos en la dificultad y el tiempo a dedicar, ya que puntos como el manejo de conexiones en tiempo real y la sincronización y seguridad de mensajes, es algo que requiere mucha dedicación y tiempo, del cual al final, no disponíamos. Tras este inconveniente, decidimos enfocar más el tema de comunicación a focalizar todo en el blog, insistiendo en mantener al día este apartado con las distintas actividades realizadas por los usuarios, y que a su vez los familiares pudieran dejar comentarios en cada publicación del blog para fomentar esta comunicación.

Por otra parte, a la hora de programar, tuvimos dos apartados en los cuales tuvimos que pararnos más y dedicar más tiempo a su lógica: la creación de las rutas y la agregación de eventos al calendario. Podremos ver estos apartados mejor explicados en el [Anexo V - Manual del programador](#). A continuación, veremos cada caso, cómo lo planteamos y cómo acabamos resolviendo los desafíos encontrados en su implementación, diseño y desarrollo.

5.8.1.-GESTIÓN DE RUTAS

En cuanto a las rutas, tuvimos que plantear cómo diseñar de manera eficiente y manteniendo siempre una lectura fácil, la creación de una ruta, ya que ésta a su vez dispone de muchas paradas, y estas paradas, de muchos usuarios, por lo que habría que buscar alguna forma de poder agregar más paradas y más usuarios. Finalmente se optó por agregar dos botones (un '+' para añadir personas a recoger de cada parada, y un 'agregar parada' para añadir una nueva parada con todos sus campos, inclusive esto comentado de añadir más personas a recoger)

RUTA

Introduzca el nombre de la ruta

Fecha de ruta

Conductor asignado

Dirección parada

Personas a recoger



Añadir parada

FIGURA 14. INTERFAZ RUTAS

Respecto a la lógica de programación de este apartado, también tuvimos que pensar esas relaciones ‘muchos a muchos’ para guardar dentro de la ruta muchas paradas y a su vez muchos usuarios por parada. La solución fue el uso de un contador en Javascript, y en la vista con Python, el uso de bucles for para ir guardando cada objeto.

```

<script>
var rutaCounter = 0;

// Funcionalidad para agregar más paradas
document.getElementById('agregar-ruta').addEventListener('click', function() {
    rutaCounter++;
    var rutaGroup = document.createElement('div');
    rutaGroup.className = 'form-group ruta-group';
    rutaGroup.id = 'ruta-group-' + rutaCounter;
    rutaGroup.innerHTML = `
        <p style="font-size: 18px;">Dirección parada</p>
        <input type="text" class="form-control" placeholder="Calle/avenida/rua/paseo..." name="direccion">
        <input type="time" class="form-control" placeholder="Hora" style="margin-bottom: 10px; width: 100px;" name="hora_recogida">
        <hr style="height: 0px;">
        <div class="px-4">
            <p style="font-size: 10px; font-style: italic;">Personas a recoger</p>
            <div class="form-group persona-group d-flex align-items-center">
                <input type="text" class="form-control" placeholder="Nombre y apellidos" name="usuario_${rutaCounter}_0">
                <button class="btn-sm btn-danger eliminar-persona-btn ml-2" type="button"></button>
            </div>
            <button class="btn-sm btn-info agregar-persona-btn mt-2" type="button"></button>
        </div>
        <p></p>
        <button class="btn-sm btn-danger eliminar-parada-btn">Eliminar parada</button>
    `;
    var addButton = document.getElementById('agregar-ruta');
    addButton.parentNode.insertBefore(rutaGroup, addButton);
});
    
```

FIGURA 15. CÓDIGO RUTAS I

```
# Crear la ruta
ruta = Ruta(nombre=nombre, fecha=fecha, conductor=conductor)
ruta.save()
print(ruta)

# Crear las paradas
for i in range(len(direcciones)):
    direccion = direcciones[i]
    hora_recogida = horas_recogida[i]
    parada = Parada(ruta=ruta, direccion=direccion, hora_recogida=hora_recogida)
    parada.save()
    |
    #Guardar personas en cada parada
    personas = []
    for key in request.POST.keys():
        if key.startswith('usuario_' + str(i)):
            personas.append(request.POST[key])
    parada.personas = personas
    parada.save()
    print(parada)
    print(personas)
```

FIGURA 16. CÓDIGO RUTAS II

5.8.2.-GESTIÓN DE CALENDARIO Y SUS EVENTOS

En cuanto a la gestión del apartado del calendario, como comentábamos, está el añadir los eventos al calendario (para el cual utilizamos la librería [FullCalendar](#), explicada en apartados anteriores).

Tras un exhaustivo estudio de esta librería y todas sus funcionalidades, pensamos en añadir un EventoCalendario cada vez que guardásemos un objeto el cual queríamos mostrar posteriormente en el calendario (como los horarios, cursos, actividades o rutas), por ello, cada vez que definimos una función en la vista de Python, guardamos una instancia al calendario, para posteriormente, en una función 'def obtener_eventos' pasarle todos los tipos de eventos al calendario, cada uno con la información requerida.

```
@login_required(login_url='/login/')
def inscribirse_curso(request, curso_id):
    curso = Curso.objects.get(id=curso_id)
    EventoCalendario.objects.create(nombre=curso.nombre, fecha=curso.hora, curso=curso, usuario=request.user)
    curso.inscribir_usuario(request.user)
    return redirect('/cursos/cursos/')
```

FIGURA 17. CÓDIGO CALENDARIO I

```

@login_required(login_url='/login/')
def obtener_eventos(request):
    selected_eventos = request.GET.getlist('selectedEventos[]')
    eventos_actividades = EventoCalendario.objects.filter(actividad__isnull=False, usuario=request.user)
    eventos_cursos = EventoCalendario.objects.filter(curso__isnull=False, usuario=request.user)
    eventos_horario = EventoCalendario.objects.filter(horario__isnull=False, usuario=request.user)
    eventos_ruta = EventoCalendario.objects.filter(ruta__isnull=False, ruta__conductor=request.user)

    print(selected_eventos)
    eventos_data = []

    if 'actividad' in selected_eventos:
        eventos_actividades = eventos_actividades.filter(actividad__isnull=False)
        for evento in eventos_actividades:
            evento_data = {
                'title': evento.nombre,
                'start': evento.fecha.strftime('%Y-%m-%d'),
                'tipo': 'actividad'
            }
            eventos_data.append(evento_data)

    if 'curso' in selected_eventos:
        eventos_cursos = eventos_cursos.filter(curso__isnull=False)
        for evento in eventos_cursos:
            evento_data = {
                'title': evento.nombre,
                'start': evento.fecha.strftime('%Y-%m-%d'),
                'tipo': 'curso'
            }
            eventos_data.append(evento_data)

```

FIGURA 18. CÓDIGO CALENDARIO II

5.9.-DESPLIEGUE

El último paso consistió en llevar a cabo el despliegue de la página web en un servidor. En particular, se utilizó el dominio de GRIAL, el cual está disponible para las tutoras.

Para lograr esto, en primer lugar, se subió el proyecto de Django a GitLab. Se creó un proyecto en GitLab y se ejecutaron los comandos correspondientes de Git, que incluyen establecer la URL del proyecto, agregar todo el contenido del proyecto Django, hacer un commit y realizar el push [13].

Posteriormente, la página web se desplegó en un servidor Ubuntu utilizando Gunicorn como servidor de aplicaciones y Nginx como un proxy inverso [14][15]. Esta configuración permitió que la página web fuera accesible a través del servidor y brindase un rendimiento óptimo.

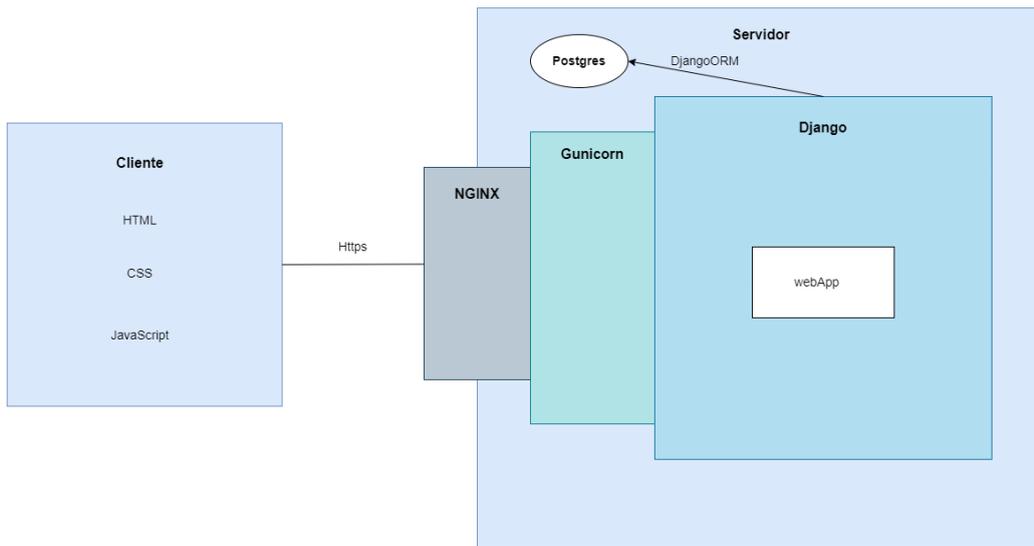


FIGURA 19. DIAGRAMA DESPLIEGUE

6.-CONCLUSIONES Y LÍNEAS FUTURAS DE TRABAJO

En conclusión, el desarrollo de este trabajo ha resultado una prueba de los conocimientos adquiridos durante estos 4 años de carrera, como son:

- La posibilidad de crear una base de datos.
- Tener la lógica y resolución de un programador para realizar una aplicación desde cero sin conocimientos previos de las tecnologías, pero que gracias a los restos de lenguajes estudiados y practicados en estos años, ha resultado una tarea relativamente sencilla.
- Planificación del tiempo, consiguiendo llegar a tiempo a la entrega de la primera convocatoria del TFG.
- Desarrollo de una metodología como completo de la planificación, siguiendo las distintas etapas del desarrollo hasta alcanzar el objetivo final (la creación de la página web y la redacción de la memoria).
- Analizar los requisitos, la audiencia o la competencia para poder llevar a cabo el desarrollo de la aplicación, estableciendo así las funcionalidades y apartados de los que constaría ésta.
- Gestionar el proyecto de una forma adecuada y exitosa, aplicando todos los pasos y conocimientos anteriores.

Demostrando así, haber conseguido superar con éxito todas las asignaturas y abastecerse de dichos conocimientos, aplicándolo a este proyecto final, además, de conseguir una satisfacción propia inmensa por el resultado final, y las felicitaciones de los usuarios que han podido probar previamente la aplicación.

En cuanto a las líneas futuras de trabajo, se vislumbra un amplio potencial para el desarrollo continuo y la expansión de esta página web. La decisión de crearla con miras a una posible implementación en centros como ASPRODES, abre la puerta a futuras mejoras y personalizaciones de acuerdo a los requisitos y funcionalidades específicas que el cliente pueda necesitar.

Una línea de trabajo prometedora sería la implementación de un apartado de comunicación más robusto, como se mencionó en la sección de "comunicación" de la

memoria. Por ejemplo, se podría agregar un chat en tiempo real u otros métodos de comunicación que faciliten la cercanía entre trabajadores, residentes y familiares. Esta mejora tendría un impacto significativo al aliviar la carga de comunicación constante que actualmente se realiza a través de llamadas telefónicas. Además, fomentaría una interacción más fluida, eficiente y satisfactoria para todos los involucrados.

Otra línea de trabajo interesante sería la optimización de la interfaz de usuario y la experiencia de navegación. A través de la recopilación de comentarios y sugerencias de los usuarios, se podría realizar una evaluación continua de la usabilidad de la página web y realizar ajustes para asegurar que sea intuitiva, accesible y atractiva visualmente.

Además, se podría explorar la integración de tecnologías emergentes, como la inteligencia artificial, para mejorar aún más la eficiencia y personalización de la plataforma. Por ejemplo, se podrían implementar sistemas de recomendación basados en preferencias individuales de los residentes, ofreciendo sugerencias de actividades o contenido adaptadas a sus intereses y necesidades.

En resumen, las líneas futuras de trabajo para esta página web abarcan desde mejoras en la comunicación hasta la ampliación de funcionalidades, la optimización de la experiencia de usuario y la incorporación de tecnologías innovadoras. Estas áreas ofrecen un amplio espacio para la innovación y el crecimiento continuo de la plataforma, buscando siempre mejorar el rendimiento, la satisfacción de los trabajadores y residentes, y adaptarse a las necesidades cambiantes del entorno.

REFERENCIAS

- [1] *Conócenos - Plena inclusión.* (n.d.). Plena Inclusión. Recuperado de: <https://www.plenainclusion.org/conocenos>
- [2] *Quiénes somos.* (n.d.). Asprodes. Recuperado de: <https://asprodes.es/quienes-somos/>
- [3] *Design Overview: A Collaborative Product Development Platform.* (n.d.). Figma. Recuperado de: <https://www.figma.com/design-overview/>
- [4] *Lectura fácil: Métodos de redacción y evaluación.* (n.d.). Plena Inclusión. Recuperado de: <https://www.plenainclusion.org/sites/default/files/lectura-facil-metodos.pdf>
- [5] *Tutorial Librería FullCalendar.* (n.d.). Wiktionary. Recuperado de: <https://tutorial101.blogspot.com/2022/05/python-django-ajax-fullcalender-crud.html>
- [6] *Librería DataTables.* (n.d.). DataTables CDN. Recuperado de: <https://cdn.datatables.net/>
- [7] Terón, R., & Universidad de Salamanca. (2021). *Diseño Centrado en el Usuario (Interacción Persona Ordenador).*
- [8] García Peñalvo, D. F. J., García Holgado, A., Vázquez Ingelmo, A., & Universidad de Salamanca. (2021). *Introducción al Proceso Unificado (Ingeniería de Software).*
- [9] *ResiPlus.* (n.d.). Software Gestión Residencias y Centros de Mayores, ResiPlus [En línea]. Recuperado de: <https://addinformatica.com/>
- [10] *IxisSocialGest.* (n.d.). [En línea]. Recuperado de: <https://www.ixissocialgest.com/>
- [11] García Peñalvo, D.F. J., García Holgado, A., & Universidad de Salamanca. (2018). *Ingeniería de Requisitos (Ingeniería de Software).*
- [12] *Diagrama entidad relación: ¿Qué es y cómo crearlo? Con ejemplos.* (n.d.). Miro. Recuperado de: <https://miro.com/es/diagrama/que-es-diagrama-entidad-relacion/>
- [13] *Repositorio TFG.* (2023, Junio 28). GitLab [En línea]. Recuperado de: <https://gitlab.grial.eu/lidiaah/tfg-lidia>
- [14] Unicorn, «Unicorn,» [En línea]. Recuperado de: <https://unicorn.org/>.
- [15] Nginx, «Nginx,» [En línea]. Recuperado de: <https://www.nginx.com/>.

- [16] Alaejos Herrero, María (2022). *Documentos ejemplos*. Centro Madrigal. Asprodes
- [17] *Otros proyectos de Trabajo de Fin de Grado de Ingeniería Informática de ejemplos* (2022). Repositorio Documental Gredos. Universidad de Salamanca
- [18] *Guía de realización y documentación Proyecto Fin de Carrera en la Ingeniería Técnica Informática*. Diaweb. Universidad de Salamanca.