

**SISTEMA PARA LA MONITORIZACIÓN DE LA
ASIGNACIÓN DE RECURSOS HUMANOS A
PROYECTOS INFORMÁTICOS.**

MEMORIA DEL PROYECTO



**VNiVERSIDAD
D SALAMANCA**

Curso 2021/2022

Alumno:

Sergio Sánchez Antona

Tutores:

Pablo Chamoso Santos
Agustín Moretón García
Lorena Sánchez Pravos

Septiembre de 2022

Resumen

Para las empresas informáticas, el seguimiento del trabajo que realizan sus empleados en cada uno de los proyectos es algo muy importante. Sin embargo, esta gestión puede suponer un esfuerzo importante y requerir de un tiempo del que no se dispone si no se realiza utilizando las herramientas adecuadas.

Dada esta necesidad, en este proyecto se plantea una aplicación web que simplifique el proceso para la gestión de proyectos informáticos desarrollados por una empresa, mostrando el flujo de horas de cada tarea que forma el proyecto y haciendo un seguimiento del trabajo de cada miembro en cada una de las tareas y en el proyecto.

Si bien existen otras herramientas de gestión de tareas, tiempos y proyectos suelen ser mucho más complejas de utilizar y, por ello, requerir más tiempo de gestión.

Además, empresas como Global Exchange, con la que se ha colaborado para el desarrollo de la presente herramienta, prefieren no depender de terceros y tener una herramienta bajo su control, que sea sencilla, rápida de usar y que permita obtener datos muy básicos pero importantes, como las horas dedicadas por proyecto y las posibles desviaciones entre horas estimadas y horas reales.

Palabras clave: gestión de proyectos, gestión de tareas, aplicación web, gestión de trabajo.

ABSTRACT

For IT companies, monitoring the work carried out by their employees in each of the projects is very important. But this management can suppose a significant effort and require time that is not available.

Given this need, this project proposes a web application that simplifies the process for managing IT projects developed by a company, showing the flow of hours for each task that makes up the project and tracking the work of each member in each one of the tasks and in the project.

While there are numerous task, time and project management tools out there, they are often much more advanced and therefore require more time to manage. In addition, companies like Global Exchange prefer not to depend on third parties and have a tool under their control that is simple, quick to use and that allows obtaining very basic but important data, such as the hours dedicated per project and the possible deviations between estimated hours and actual hours.

Keywords: project management, task management, web application, work management.

Índice

Resumen	3
ABSTRACT	5
1. Introducción.....	11
2. Objetivos	13
2.1. Objetivo principal.....	13
2.2. Objetivos específicos.....	13
2.2.1. Objetivos marcados por los requisitos.....	13
2.2.2. Objetivos técnicos.....	14
3. Trabajos relacionados	15
3.1. Explicación de la aplicación elegida	15
3.2. Aplicaciones de control de tareas y tiempos para gestión de proyectos.....	15
3.2.1. Azure Devops	15
3.2.2. Trello	17
3.2.3. Jira	17
4. Técnicas y herramientas.....	19
4.1. Patrones de diseño.....	19
4.1.1. MVC	19
4.1.2. Data Access Object (DAO).....	19
4.2. Lenguajes de programación.....	20
4.2.1. Vue.js	20
4.2.2. HTML.....	20
4.2.3. JavaScript	21
4.2.4. TypeScript	21
4.3. Bases de datos	22
4.3.1. PostgreSQL.....	22
4.4. Entornos de ejecución y servidores	22
4.5. Otras herramientas utilizadas.....	23
4.5.1. Visual Paradigm	23
4.5.2. Microsoft Project	23
5. Aspectos relevantes del desarrollo	25
5.1. Ciclo de vida	25
5.2. Planificación temporal	26
5.3. Arquitectura del sistema	36
5.3.1. Cliente	36
5.3.2. Servidor.....	37
5.3.3. Justificación.....	39
5.4. Especificación de requisitos.....	39

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.

Memoria final

5.5.	Etapa de análisis.....	42
5.6.	Etapa de diseño.....	44
5.7.	Etapa de implementación.....	49
6.	<i>Conclusiones y trabajo futuro.....</i>	59
	<i>Referencias.....</i>	62

Tabla de ilustraciones:

Ilustración 1. Página Azure DevOps. Fuente: https://dev.azure.com/antonasergio/	16
Ilustración 2. Página Azure DevOps. Fuente: https://dev.azure.com/antonasergio/TFG16	
Ilustración 3. Página Trello. Fuente: https://trello.com/es/tour	17
Ilustración 4. Página Jira. Fuente: https://www.atlassian.com/es/software/dev-tools	18
Ilustración 5. Logotipo de Vue.js. [7].....	20
Ilustración 6. Logotipo de HTML. [8].....	20
Ilustración 7. Logotipo de JavaScript. [9]	21
Ilustración 8. Logotipo de TypeScript. [10]	21
Ilustración 9. Logotipo de PostgreSQL. [11]	22
Ilustración 10. Logotipo de node. [12]	22
Ilustración 11. Logotipo de Visual Paradigm. [13]	23
Ilustración 12. Logotipo de Microsoft Project. [14].....	23
Ilustración 13. Diagrama de Gantt con la planificación del proyecto (parte 1). Fuente: elaboración propia	33
Ilustración 14. Diagrama de Gantt con la planificación del proyecto (parte 2). Fuente: elaboración propia	33
Ilustración 15. Diagrama de Gantt con la planificación del proyecto (parte 3). Fuente: elaboración propia	34
Ilustración 16. Diagrama de Gantt con la planificación del proyecto (parte 4). Fuente: elaboración propia	34
Ilustración 17. Diagrama de Gantt con la planificación del proyecto (parte 5). Fuente: elaboración propia	34
Ilustración 18. Diagrama de Gantt con la planificación del proyecto (parte 6). Fuente: elaboración propia	35
Ilustración 19. Diagrama de Gantt con la planificación del proyecto (parte 7). Fuente: elaboración propia	35
Ilustración 20. Diagrama de Gantt con la planificación del proyecto (parte 8). Fuente: elaboración propia	35
Ilustración 21. Diagrama de Gantt con la planificación del proyecto (parte 9). Fuente: elaboración propia	36
Ilustración 22. Diagrama de Gantt con la planificación del proyecto (parte 10). Fuente: elaboración propia	36
Ilustración 23. Esquema con la arquitectura cliente-servidor propuesta. Fuente: elaboración propia	
Ilustración 23. Esquema con la arquitectura cliente-servidor propuesta. Fuente: elaboración propia	
Ilustración 24. Esquema de la parte cliente. Fuente: elaboración propia	37
Ilustración 25. Esquema con los componentes principales de la parte servidora Fuente: elaboración propia	38
Ilustración 26. Diagrama de clases de la etapa de análisis. Fuente: elaboración propia	
Ilustración 27. Arquitectura de capas. Fuente: elaboración propia	44
Ilustración 28. Diagrama de secuencia caso de uso 'Iniciar sesión'	45
Ilustración 29. Diagrama de secuencia caso de uso 'Ver lista de proyectos creados'	45
Ilustración 30. Diagrama de secuencia caso de uso 'Añadir proyecto'	46
Ilustración 31. Diagrama de secuencia caso de uso 'Añadir miembro'	46
Ilustración 32. Diagrama de secuencia caso de uso 'Añadir tarea'	47
Ilustración 33. Diagrama de secuencia caso de uso 'Ver lista de tareas'	47
Ilustración 34. Diagrama de secuencia caso de uso 'Actualizar tarea'	48
Ilustración 35. Diagrama de secuencia caso de uso 'Ver detalles del proyecto'	48

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

Ilustración 36. Diagrama de secuencia caso de uso 'Generar gráficos'	49
Ilustración 37. Pantalla de inicio	50
Ilustración 38. Pantalla de registro	51
Ilustración 39. Pantalla inicial (parte 1).....	51
Ilustración 40. Pantalla inicial (parte 2).....	51
Ilustración 41. Pantalla inicial (parte 3).....	52
Ilustración 42. Pantalla de proyectos	52
Ilustración 43. Pantalla de miembros	52
Ilustración 44. Pantalla de tareas (parte 1)	53
Ilustración 45. Pantalla de tareas (parte 2)	53
Ilustración 46. Pantalla de detalles (parte 1).....	53
Ilustración 47. Pantalla de detalles (parte 2).....	54
Ilustración 48. Pantalla de detalles (parte 3).....	54
Ilustración 49. Pantalla de detalles (parte 4).....	54
Ilustración 50. Pantalla móvil de página inicio	55
Ilustración 51. Pantalla móvil página miembros	56
Ilustración 52. Pantalla móvil página inicial	57
Ilustración 53. Pantalla móvil página cambio de contraseña y barra de navegación	58

1. Introducción

Los proyectos informáticos en las empresas suelen ser grandes procesos de desarrollo con un conjunto de actividades relacionadas entre sí que tienen un tiempo marcado y las cuales son realizadas por miembros de la empresa asignados a las distintas tareas.

Estos proyectos necesitan un seguimiento para cumplir los plazos de tiempo fijados. Para ello existen ya múltiples aplicaciones, las cuales suelen ser demasiado complejas y usan datos que son irrelevantes para muchas empresas. Aun así, no usar estas herramientas puede suponer no tener un buen control sobre los desarrollos y, finalmente, incumplir los plazos asignados a los proyectos o no tener el rendimiento necesario para llevar a cabo estos proyectos.

Por ello, se plantea este proyecto como solución. La propuesta consiste en el análisis, diseño e implementación de una aplicación web para la gestión de proyectos informáticos de la empresa Global Exchange [1]. Esta empresa busca solucionar los problemas descritos anteriormente, por lo que se plantea esta aplicación para simplificar el trabajo de la gestión de tareas de los proyectos y obtener valiosos informes sobre el rendimiento de los miembros que trabajan en el proyecto.

Con la ayuda de Global Exchange y del tutor del TFG se han obtenido los requisitos técnicos y funcionales del software a desarrollar. Por tanto, la aplicación web desarrollada permite gestionar proyectos informáticos, miembros y tareas de forma que se generan los datos necesarios para el seguimiento de estos proyectos. También se incluye en la aplicación la gestión de usuarios para la identificación de los miembros del proyecto con dos roles diferentes: Administrador o Usuario. Los administradores podrán gestionar (crear, eliminar o actualizar) los proyectos, miembros y tareas. Los usuarios podrán ver los datos generados por la aplicación en forma de tablas y gráficas que muestran un seguimiento de los proyectos y el rendimiento de los trabajadores.

Para la realización de esta aplicación se han utilizado distintos lenguajes de programación y diferentes técnicas para crear el proyecto y la base de datos, descritos con mayor nivel de detalle en el capítulo 3 y los anexos.

De forma general, las partes principales del proyecto son:

1. *Frontend*. Esta parte ha sido desarrollada con Vue [2], JavaScript [3], HTML y CSS [4].
2. *Backend*. Ha sido desarrollado con TypeScript, Shell y PLpgSQL [5]. La base de datos creada es PostgreSQL [6] utilizada para gestionar los datos de proyectos, miembros, tareas y usuarios.

Esta aplicación es accesible desde cualquier navegador web, a través de un servidor local accesible a cualquiera que consulte la URL smpi.no-ip.org. Aunque el objetivo final es que la aplicación esté alojada en un servidor propio, de acceso privado y exclusivo de Global Exchange, debido a que es una aplicación desarrollada para esta empresa y únicamente con la que ellos puedan usar sus funcionalidades. Pero por motivos didácticos

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

y para la exposición del trabajo de fin de grado esta aplicación se ha publicado en un servidor accesible desde Internet.

2. Objetivos

En este capítulo se describen los objetivos planteados en este trabajo fin de grado.

2.1. Objetivo principal

El objetivo principal es analizar, diseñar y desarrollar un software que permita controlar, de forma muy sencilla, las tareas y horas realizadas en los proyectos de una empresa, así como el seguimiento del rendimiento de los miembros que realizan estas tareas. El software a desarrollar podría servir para muchas empresas, pero en este trabajo fin de grado se ha desarrollado siguiendo las directrices y requisitos marcados por la empresa Global Exchange.

2.2. Objetivos específicos

2.2.1. Objetivos marcados por los requisitos

1. **Controlar y gestionar los desarrolladores autorizados a participar en los proyectos.** El sistema tiene que ser capaz de gestionar los usuarios que tienen autoridad para participar en los proyectos. Para ello se registran en el sistema y posteriormente deben iniciar sesión. Para ello existirá una página de registro en la que se introduzcan los datos de un nuevo usuario y el sistema sea capaz de introducirlos en la base de datos, además una página de inicio de sesión en la que introduzca cada usuario sus credenciales se compare con las de la base de datos y deje acceso al usuario si coinciden.
2. **Mejorar el control y la gestión de proyectos.** Para ello el software permitirá añadir y eliminar proyectos en el sistema, que se guardarán en la base de datos. Para mejorar el control se debe ir actualizando la información de estos proyectos cuando se añadan tareas (horas que lleva el proyecto y horas estimadas) para posteriormente utilizarla y mostrar detalles al usuario de la aplicación.
3. **Tener un control preciso del tiempo de tareas y de proyectos.** El sistema tiene que contar con una página para crear, eliminar y actualizar tareas. Las tareas se irán añadiendo con las horas que se haya trabajado en estas, cuando se actualicen se suman las horas que se han realizado posteriormente y el día en que se han actualizado con el fin de tener un seguimiento del tiempo que ha llevado realizar las tareas y el proyecto.
4. **Disponer de un control del rendimiento de los miembros en las tareas de proyecto.** El sistema contará con una página para crear miembros que se asocian a cada proyecto. Posteriormente al crear la tarea se le asigna el miembro que la realiza y el tiempo que lleva realizándola. Estos datos se guardan para hacer informes por medio de tablas y gráficos para mostrar el rendimiento que está teniendo el miembro en las tareas y en el proyecto.

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

5. **Simplificar las gestiones.** El sistema tiene que ser lo más simple posible, excluyendo aquellos datos innecesarios que no se vayan a usar en la funcionalidad marcada por los requisitos impuestos por Global Exchange.
Se simplifica la gestión de los proyectos para mostrar únicamente los datos que interesen sobre el rendimiento de los miembros, la gestión de los proyectos y las tareas.

2.2.2. Objetivos técnicos

1. **Tener un software web de tipo responsive.** Diseñar un software que se pueda redimensionar para cada vista de la aplicación, en la que haciendo más pequeña la página, la vista siga siendo la correcta.
2. **Aplicación con formato web.** Diseñar una aplicación web que sea accesible desde cualquier dispositivo a través de un navegador introduciendo el nombre de dominio, accesible desde cualquier dispositivo, desde ordenadores hasta móviles.

3. Trabajos relacionados

El trabajo realizado se basa en la gestión de proyectos informáticos, en el cálculo del trabajo necesario para realizar las tareas de este y en la medida del rendimiento de los trabajadores. Este tipo de aplicaciones ya existen y son utilizadas por muchas empresas. En este capítulo se explica por qué se ha realizado esta aplicación existiendo ya herramientas parecidas y se describen algunas de las aplicaciones más conocidas con funcionalidades similares a las de este proyecto.

3.1. Explicación de la aplicación elegida

En primer lugar, se explica en este apartado el porqué de la elección de este trabajo existiendo aplicaciones similares.

Como se ha indicado en apartados anteriores, se trata de un proyecto realizado para una empresa, Global Exchange, cuya finalidad es el uso exclusivo de la aplicación para solucionar el problema planteado. Este problema se trata de conseguir gestionar proyectos y tareas mostrando datos importantes sobre el rendimiento de los miembros del proyecto. Al tratar este problema un requisito fundamental para la solución es eliminar datos e información que no nos interese y poder gestionar las tareas de los proyectos de la empresa de una forma simple y propia, es decir, que no dependa de herramientas de terceros. Este punto es importante ya que lo que buscan es una aplicación propia ejecutada en un servidor de la empresa para gestionar únicamente sus proyectos sin información almacenada en herramientas públicas o de terceros por aspectos de confidencialidad.

Con estos requisitos se planteó la solución, una herramienta simple sin datos innecesarios orientada a Global Exchange.

3.2. Aplicaciones de control de tareas y tiempos para gestión de proyectos

En este apartado se incluyen algunas herramientas de mercado ya existentes relacionadas con la temática del trabajo.

3.2.1. Azure Devops

Azure DevOps [19] es una herramienta de Microsoft para la gestión de proyectos, con múltiples opciones para planificar y colaborar con un equipo.

Con esta aplicación se puede planificar y seguir el trabajo del equipo en un proyecto. En un principio este TFG se iba a realizar con este sistema, ampliándolo o enlazando la aplicación que se utilizara con esta herramienta, pero debido a sus excesivas opciones y datos innecesarios se deshecho. Además, eran necesarios conocimientos avanzados de lenguajes y aplicaciones que el desarrollador no tenía por tanto se optó por simplificar la aplicación con la solución ya explicada.

En las ilustraciones adjuntas se puede observar la web de DevOps, en este caso cómo se ha explicado en un principio este trabajo se iba a relacionar con esta aplicación y hay una cuenta personal creada con un proyecto con el nombre TFG. Se puede ver en la ilustración 41 la página inicial con la cuenta del usuario y los proyectos que hay creados y en la

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos.

Memoria final

ilustración 42 el proyecto que hay creado con todas las opciones que nos permite esta aplicación.

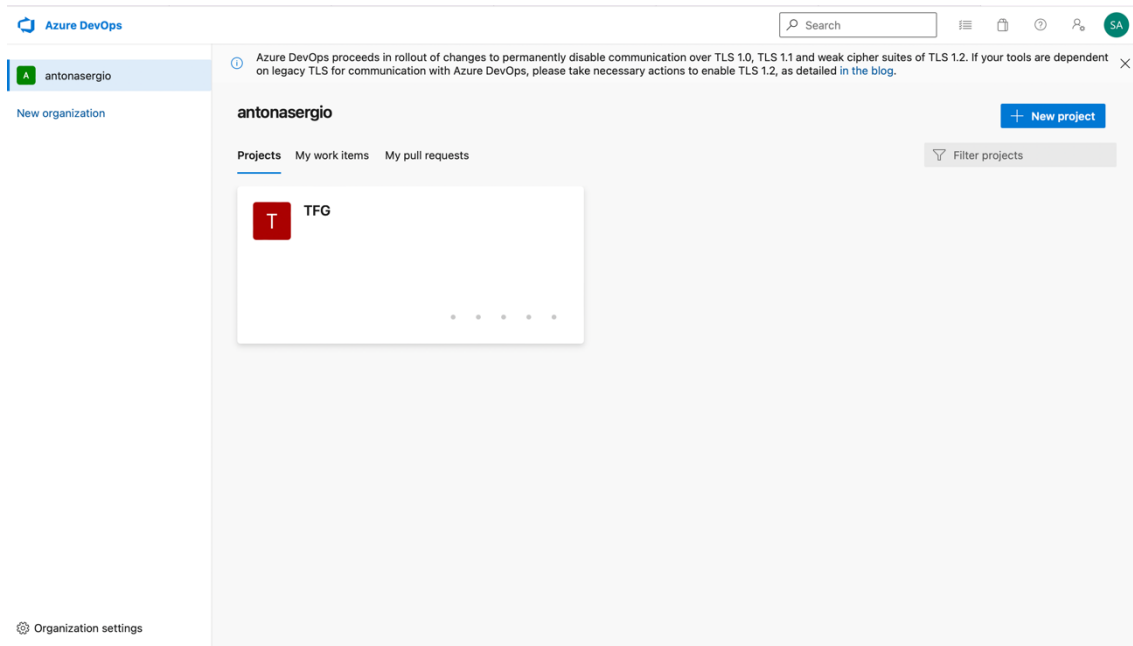


Ilustración 1. Página Azure DevOps. Fuente: <https://dev.azure.com/antonasergio/>

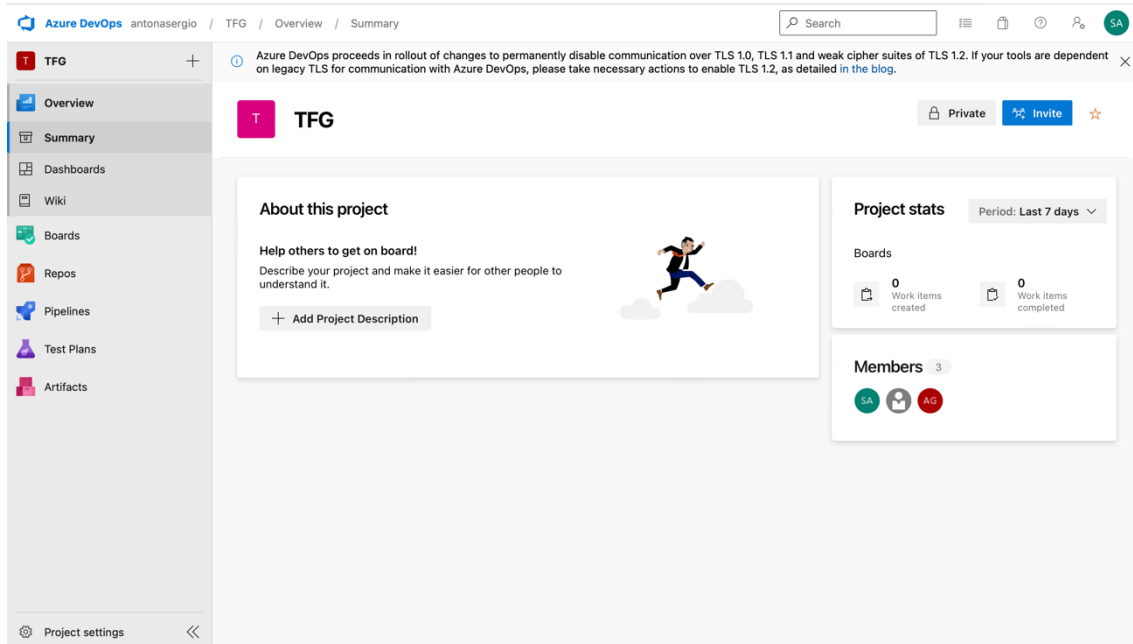


Ilustración 2. Página Azure DevOps. Fuente: <https://dev.azure.com/antonasergio/TFG>

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos. Memoria final

3.2.2. Trello

Trello [20] es una aplicación de gestión de tareas muy utilizada por equipos de desarrollo que utilizan metodologías ágiles, donde existe una elevada actividad colaborativa en los proyectos.

Además, tiene la funcionalidad de configurar equipos y flujos de trabajo mediante un tablero que tiene como interfaz de usuario en el que se organiza el trabajo, cuenta con tarjetas para organizar el trabajo y compartir con los compañeros de equipo el trabajo realizado.

Tal y como se observa en la ilustración adjunta, Trello es una aplicación para gestionar los proyectos que cuenta con este tablero el cual indica que tareas están por hacer, las que se están haciendo y las que ya están hechas, además se pueden incluir comentarios y errores en las tareas. De esta aplicación no se tiene ninguna cuenta creada.

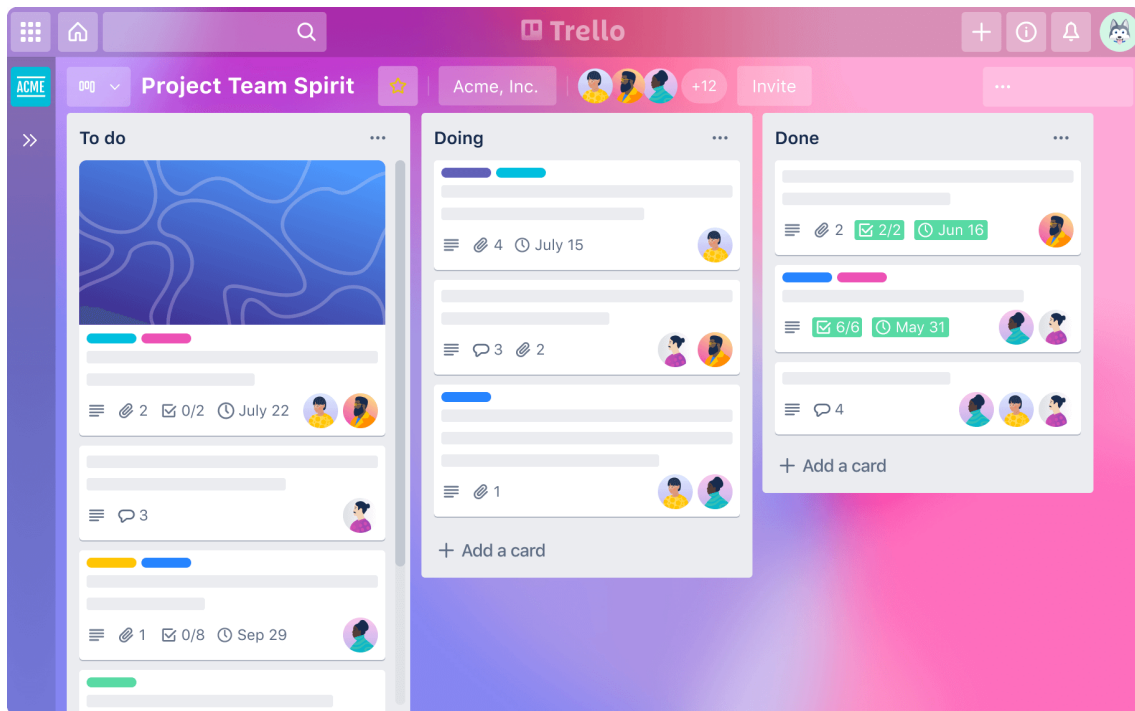


Ilustración 3. Página Trello. Fuente: <https://trello.com/es/tour>

3.2.3. Jira

Jira [21] es un software de gestión de proyectos desarrollado por la empresa Atlassian que permite gestionar las tareas de los proyectos y seguir los errores y las incidencias. Planifica los proyectos dividiendo las tareas en tareas más pequeñas fáciles de gestionar para los equipos de trabajo.

En esta aplicación también se pueden integrar otras herramientas con las que se estén trabajando en el proyecto para hacer más fácil el desarrollo y la gestión del trabajo. Al igual que Trello, tiene una tabla con la que se gestionan las tareas y se puede colaborar con el equipo de desarrollo.

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos.

Memoria final

Es un software que permite elegir cual es la metodología que se está usando.

En la ilustración adjunta se observa que al igual que en Trello se cuenta con un tablero para saber las tareas que se están realizando, en progreso y las que se han realizado, además tiene una columna de revisión de código. Mediante este tablero indicamos al equipo las tareas del proyecto y los demás miembros pueden verlas y poner comentarios o errores.

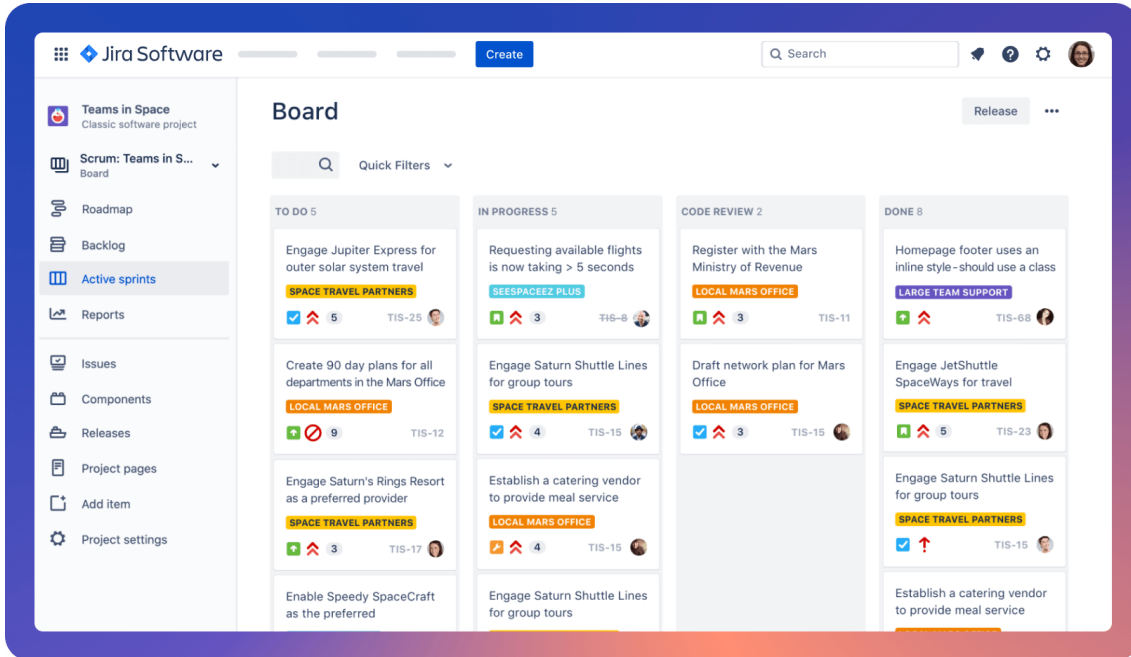


Ilustración 4. Página Jira. Fuente: <https://www.atlassian.com/es/software/dev-tools>

4. Técnicas y herramientas

En este capítulo se exponen las técnicas y herramientas que se han usado para el desarrollo del trabajo, se definen los aspectos más destacados del proyecto.

4.1. Patrones de diseño

Se explican los detalles de los patrones de diseño utilizados durante el desarrollo de la aplicación y vistos en los anexos de la documentación.

4.1.1. MVC

Patrón de diseño Modelo Vista Controlador, para el diseño arquitectónico del software visto en el Anexo 3 de la documentación.

Utilizado para clasificar las clases del diseño entre la información y métodos que controlan la base de datos, los métodos que controla y pasan por el sistema y las interfaces del sistema.

Los componentes son:

- **Modelo:** Componente para gestionar los datos, que van a pasar a la base de datos.
- **Vista:** Encargado de mostrar al usuario las pantallas y vistas de la aplicación, métodos que usará este usuario para controlar las ventanas y la información de las páginas.
- **Controlador:** Componente encargado de controlar y gestionar los métodos que se reciben como instrucciones y es el sistema el que se encarga de realizarlos, por medio de este se comunicarán la vista y el modelo.

4.1.2. Data Access Object (DAO)

Patrón de diseño utilizado para acceder a fuentes de datos, en nuestro caso la base de datos PostgreSQL, suministra la interfaz común entre la aplicación y el almacenamiento de los datos.

4.2. Lenguajes de programación

Se definen en este apartado los lenguajes de programación principales con los que se ha desarrollado la aplicación.

4.2.1. Vue.js



Ilustración 5. Logotipo de Vue.js. [7]

Vue.js (o simplemente Vue) [2] es el lenguaje más utilizado en la aplicación para la parte de *frontend*, se trata de un *framework* progresivo de JavaScript para crear las interfaces de usuario.

Se usa para crear las páginas con un lenguaje parecido a HTML el cual también usamos y de hecho introducimos todos los *scripts* de Vue en las páginas HTML.

4.2.2. HTML



Ilustración 6. Logotipo de HTML. [8]

Es un lenguaje de etiquetas, componente básico de la web. Define la estructura de nuestro sitio web, y es en una página HTML donde introducimos los *scripts* de Vue como ya se ha explicado en el apartado anterior.

4.2.3. JavaScript



Ilustración 7. Logotipo de JavaScript. [9]

Es un lenguaje de programación que permite implementar funciones complejas en páginas web. Es un lenguaje de programación interpretado orientado a objetos que utilizamos en la parte de *frontend* combinado con Vue para gestionar las páginas de *front*. También lo usaremos en *backend* para *scripts* de configuración y ya que usando TypeScript [5] se generará automáticamente el código en este lenguaje en la carpeta /dist.

4.2.4. TypeScript



Ilustración 8. Logotipo de TypeScript. [10]

Lenguaje de programación libre y de código abierto desarrollado como subconjunto de JavaScript, que añade tipos estáticos y objetos basados en clases para extender la sintaxis de JavaScript. Se usa en *backend* como lenguaje de programación principal con el que desarrollamos todo el código del api y de la configuración del servidor.

4.3. Bases de datos

Se explica en este apartado la base de datos utilizada por el sistema.

4.3.1. PostgreSQL

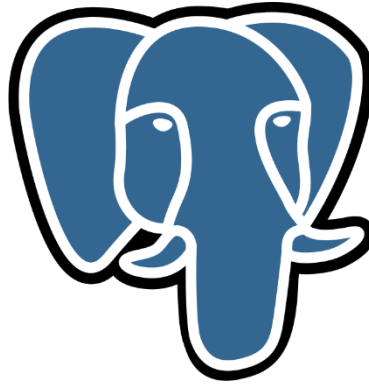


Ilustración 9. Logotipo de PostgreSQL. [11]

Sistema de gestión de bases de datos [6] relacional orientado a objetos y de código abierto con el que gestionamos los datos de nuestra base de datos.

Todos los elementos de esta base de datos van a ser tratados como objetos siendo así más fácil de utilizar, se ha utilizado junto con el servidor de node para gestionar la información.

4.4. Entornos de ejecución y servidores

En este apartado se describe el entorno de ejecución que se ha utilizado para el proyecto, siendo el mismo el entorno de ejecución y el servidor en que se ejecuta la aplicación, Node.js.

Node:



Ilustración 10. Logotipo de node. [12]

Entorno de ejecución [15] para JavaScript multiplataforma y de código abierto basado en el lenguaje de programación JavaScript.

Utilizado como entorno de ejecución de nuestro sistema y como servidor Node.JS Server para servir nuestra aplicación en un puerto de nuestro servidor proporcionando respuesta HTTP.

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

4.5. Otras herramientas utilizadas

En este apartado se describen otras herramientas que se han utilizado mayormente para hacer la documentación.

4.5.1. Visual Paradigm



Ilustración 11. Logotipo de Visual Paradigm. [13]

Herramienta para gestión de proyectos utilizada para hacer los diagramas de la documentación [16] que se han presentado en la documentación técnica.

Se utiliza esta aplicación para realizar los diagramas necesarios de la parte de la documentación del proyecto, todos los diagramas han sido realizados con Visual Paradigm ya que el desarrollador tenía experiencia utilizando esta aplicación y no necesitaba aprender a utilizarla.

4.5.2. Microsoft Project



Ilustración 12. Logotipo de Microsoft Project. [14]

Software de Microsoft utilizado para planificar las tareas y mostrar el diagrama de Gantt [17]. Enfocado a gestión de proyectos para analizar los datos vendido por Microsoft para asistir a administradores de proyectos en el seguimiento del proceso.

Este Software se ha utilizado para la planificación temporal del proyecto, añadiendo cada tarea que se ha realizado en el proyecto para tener una estimación del tiempo que se dedica

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

a cada tarea y al proyecto completo. Se utiliza esta aplicación porque el desarrollador ya tenía experiencia usándola y la planificación del proyecto es necesaria para seguir el desarrollo de este.

5. Aspectos relevantes del desarrollo

En este capítulo se explicarán los aspectos más relevantes del trabajo exponiendo el ciclo de vida utilizado y los detalles de mayor relevancia en las distintas fases del proyecto.

Se explicarán en los siguientes apartados de manera justificada las soluciones que se han implementado para los problemas iniciales. Además, se explicará la arquitectura elegida.

5.1. Ciclo de vida

Para este trabajo se ha seguido un modelo de ciclo de vida clasificado en distintas fases que son Inicio, Elaboración, Construcción y Transición que se definen como:

- **Fase de Inicio:** Fase en la que se empieza este proyecto, se empezará a recaudar información sobre el problema que queremos resolver. Al ser este TFG dirigido a Global Exchange se recogen los requisitos que esta empresa marque en las reuniones.

Se definen también las herramientas que se van a usar para este proyecto, así como los lenguajes de programación y las funcionalidades.

Se empiezan los casos de uso y debido a la inexperiencia del programador en los lenguajes y herramientas que se han elegido también se empieza con tutoriales para el aprendizaje de estos.

- **Fase de Elaboración:** En esta fase se empiezan a refinar los casos de uso y el análisis de estos, se elaboran los objetivos del sistema en base a los requisitos que hemos recogido en la anterior fase y se siguen refinando las funcionalidades gracias a las diferentes reuniones con los tutores que sirven para añadir más funcionalidades o para descartar las que no fueran útiles.

Se sigue aprendiendo de los lenguajes de programación haciendo tutoriales y se hace una primera prueba de un proyecto con estos lenguajes y las herramientas elegidas.

- **Fase de Construcción:** La fase más larga del proyecto ya que es donde construimos e implementamos todo el sistema y los diferentes objetivos. Se van refinando los casos de uso a medida que vamos desarrollando la aplicación.

Además, a la vez que se va implementando se van realizando pruebas individuales de cada objetivo, así como globales de la aplicación para comprobar el correcto funcionamiento.

Continúan las reuniones con los tutores esta vez para mostrar el desarrollo y cómo se va implementando la aplicación, son reuniones más técnicas en las que se habla de métodos, funciones y la forma de usar alguna de las herramientas asignadas en la fase de inicio. También son importantes estas reuniones para según se muestra la idea de la solución al problema planteado, sacar fallos a esta solución o considerar problemas que no se habían tenido en cuenta.

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

El final de esta fase será con la aplicación terminada a falta de pruebas y retoques finales.

- **Fase de Transición:** La fase final del proceso, en la cual se hacen las últimas pruebas ya con la solución del proyecto terminada y ultimando los detalles del código. Se muestran los resultados en las reuniones finales y se cambian en estas reuniones los detalles que se hayan pasado por alto o algunos que no afecten a la funcionalidad del sistema.

Se inicia en esta fase la documentación, que se explicará en detalle en otra reunión, y se realiza después con el código ya terminado y con el servidor listo.

Aquí concluirá el ciclo de vida del proyecto.

Este ciclo de vida se ha dividido en iteraciones e hitos, que a su vez estaban divididos en tareas, las cuales se explican en el Anexo 1 de la documentación.

Las iteraciones y los hitos tenían a su vez una fecha de inicio y finalización fijada, marcada por lo que se esperaba que se tardara en realizar cada tarea como se mostrará en el siguiente apartado de esta memoria. Finalmente se ha cumplido cada tarea propuesta en cada una de las fases, así como el tiempo de inicio y finalización del proyecto desarrollando el proyecto en periodos cortos siguiendo las iteraciones marcadas.

Como se puede ver en el ciclo de vida, mucha parte de este proceso ha sido aprender los lenguajes y las herramientas que se utilizan en el desarrollo del sistema. Este proyecto tiene por tanto una curva de aprendizaje muy grande la cual ha consumido gran parte del tiempo asignado para la realización del trabajo.

Además, la fase de construcción es la más larga ya que es donde se desarrolla la mayor parte del código y debido de nuevo a la inexperiencia del programador en los lenguajes de programación se han tenido que realizar numerosas pruebas para conseguir el correcto funcionamiento del sistema durante toda esta fase de construcción del código.

Por último, la fase de transición es en la que se realiza la documentación la cual ha llevado cerca de un mes acabar. En esta fase el desarrollador ya tenía experiencia y se ha realizado más rápido, aun así, es un proceso que lleva tiempo debido a las múltiples memorias que hay que realizar y a las herramientas que hay que utilizar para completar estas memorias.

5.2. Planificación temporal

Una de las primeras tareas a realizar en el ciclo de vida del proyecto ha sido calcular la duración que va a tener este para así usar como referencia ese tiempo y planificar de la mejor manera las tareas y fases que se van a seguir.

Como se explica en el apartado anterior se ha dividido el ciclo de vida en 4 partes, las cuales han seguido unas iteraciones y un número de tareas.

Cada una de las fases se puede dividir en: Modelado de negocio, requisitos, análisis, diseño, implementación y pruebas. Aunque alguna iteración no tenga todas estas tareas.

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

En total se han realizado 58 tareas en todo el proyecto asignadas a las fases y a las iteraciones. Con esta división de tareas el calendario de trabajo final marca que se empieza el proyecto el día 9 de febrero de 2022 y termina el 15 de agosto de 2022 lo cual implica 6 meses hasta finalizar el proyecto.

Finalmente, esta planificación temporal se ha cumplido, terminando la última tarea cerca del 15 de agosto y añadiendo pruebas finales y ajustes del servidor con el desarrollo ya terminado entregando este trabajo el día 5 de septiembre del 2022.

Las tareas que se han realizado se muestran en esta tabla:

Número	Tarea	Descripción
1	Reunión inicial con los tutores	Reunión inicial con los tutores para plantear el proyecto y tomar los requisitos
2	Identificar actores	Identificar personas que usarán la aplicación como actores
3	Identificar casos de uso	Identificar los casos de uso del sistema
4	Identificar objetivos	Identificar los objetivos del sistema
5	Definir herramientas que se van a utilizar para hacer la aplicación	Definir qué herramientas y qué lenguajes se van a usar para hacer la aplicación
6	Estudio de tutoriales para aprender a usar herramientas y lenguajes que se han definido	Aprendizaje de las herramientas y lenguajes de programación que se han decidido usar
7	Identificar requisitos de información	Identificar los requisitos de funcionalidad del sistema
8	Identificar requisitos no funcionales	Identificar sistemas u operaciones que van a ayudar a la realización del sistema
9	Analizar los objetivos	Analizar los objetivos que se han identificado

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

10	Diseño del modelo de datos	Diseñar el modelo de datos para empezar a ver cómo será el sistema en un futuro
11	Diseño de la arquitectura inicial	Diseñar la primera arquitectura para ver los pasos a seguir y los procesos a realizar
12	Reunión con tutores	Se realizarán varias a partir de aquí para revisar el sistema y hacer cambios o mejoras en las partes que sean necesarias
13	Hacer casos de uso	Realizar las tablas y diagramas de los casos de uso
14	Hacer diagramas de actores	Incluir los actores en los diagramas
15	Hacer requisitos no funcionales	Realización de las tablas de requisitos no funcionales
16	Diagramas de clases	Realizar diagramas de clases del análisis
17	Hacer proyecto inicial	Realizar el proyecto inicial desde cero en base a los lenguajes y herramientas elegidas anteriormente
18	Hacer pruebas de proyecto inicial	Realizar las pruebas del primer proyecto creado en el paso anterior
19	Revisión de los casos de uso y funcionalidad del sistema	Revisar los casos de uso y la funcionalidad de la aplicación dependiendo de la base del proyecto y de las herramientas y lenguajes especificados
20	Elaboración de diagramas de secuencia	Realización de los diagramas de secuencia

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

21	Documentación de las clases	Inicio de la documentación de las distintas clases del sistema
22	Diagramas de diseño	Inicio de elaboración de diagramas de diseño
23	Diseño de la estructura de la base de datos	Diseñar la estructura inicial de la base de datos con los requisitos que se hayan establecido
24	Creación de la base de datos	Creación de la base de datos a partir del diseño de esta que se hizo previamente
25	Pruebas de la base de datos	Pruebas iniciales para familiarizarse con el lenguaje de bases de datos
26	Investigación y pruebas	Investigación y aprendizaje sobre métodos de los nuevos lenguajes y pruebas de las funcionalidades que queremos conseguir
27	Refinamiento de requisitos del sistema	Refinamiento de los requisitos que habíamos tomado revisándolos y cambiando o añadiendo los que sean necesarios
28	Refinamiento de casos de uso	Refinamiento de los casos de uso que habíamos definido
29	Refinamiento del diseño	Refinamiento de la parte del diseño
30	Implementación del objetivo 'Gestión de proyectos'	Implementación del objetivo que hemos definido como 'Gestión de proyectos'
31	Implementación del objetivo 'Gestión de miembros'	Implementación del objetivo que hemos definido como 'Gestión de miembros'

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos.
Memoria final

32	Implementación del objetivo 'Gestión de tareas'	Implementación del objetivo que hemos definido como 'Gestión de tareas'
33	Implementación del objetivo 'Gestión de detalles'	Implementación del objetivo que hemos definido como 'Gestión de detalles'
34	Implementación del objetivo 'Gestión de usuarios'	Implementación del objetivo que hemos definido como 'Gestión de usuarios'
35	Pruebas de la base de datos	Pruebas de almacenamiento de datos en la base de datos
36	Refinar objetivo 'Gestión de proyectos'	Refinar requisitos del objetivo 'Gestión de proyectos' en base a lo que se ha ido implementando
37	Refinar objetivo 'Gestión de miembros'	Refinar requisitos del objetivo 'Gestión de miembros' en base a lo que se ha ido implementando
38	Refinar objetivo 'Gestión de tareas'	Refinar requisitos del objetivo 'Gestión de tareas' en base a lo que se ha ido implementando
39	Refinar objetivo 'Gestión de detalles'	Refinar requisitos del objetivo 'Gestión de detalles' en base a lo que se ha ido implementando
40	Refinar objetivo 'Gestión de usuarios'	Refinar requisitos del objetivo 'Gestión de usuarios' en base a lo que se ha ido implementando
41	Pruebas unitarias del objetivo 'Gestión de proyectos'	Pruebas individuales de este objetivo

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

42	Pruebas unitarias del objetivo 'Gestión de miembros'	Pruebas individuales de este objetivo
43	Pruebas unitarias del objetivo 'Gestión de tareas'	Pruebas individuales de este objetivo
44	Pruebas unitarias del objetivo 'Gestión de detalles'	Pruebas individuales de este objetivo
45	Pruebas unitarias del objetivo 'Gestión de usuarios'	Pruebas individuales de este objetivo
46	Pruebas del funcionamiento del sistema	Pruebas globales del sistema
47	Reunión con los tutores para mostrar desarrollo	Reunión con los tutores del trabajo para mostrar desarrollos y debatir cambios
48	Análisis de los resultados	Analizamos los resultados del desarrollo para hacer cambios
49	Verificación de los requisitos	Verificación de los requisitos ya refinados para saber si todos son correctos
50	Programación de la documentación	Programación de los tiempos de documentación y diseño de las partes que va a tener
51	Aplicar cambios en el desarrollo de los objetivos en base a las pruebas	Aplicar los cambios que hayamos apuntado de los casos anteriores
52	Inicio de la documentación	Iniciar los documentos de documentación y planificar tiempos
53	Pruebas de interfaz	Pruebas de funcionamiento de la interfaz del usuario
54	Reunión final con los tutores del proyecto	Reunión para presentar resultado del sistema y ultimar los detalles

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

55	Realizar la documentación	Realizar la documentación en base a las reuniones con los profesores
56	Refinamiento de la documentación	Últimos detalles de la documentación
57	Pruebas finales de la aplicación	Pruebas del sistema completo
58	Implementación del despliegue de los subsistemas en un servidor	Despliegue del sistema en un servidor

De estas tareas podemos concluir como tareas principales:

- Reunión inicial con los tutores: Es la primera reunión del proyecto en la cual se exponen los requisitos fundamentales para empezar el proyecto, a raíz de esta reunión se empieza el trabajo planteando el problema en las siguientes tareas.
- Estudio de tutoriales para aprender a usar herramientas y lenguajes que se han definido: En esta tarea se empiezan a aprender todas las herramientas que se van a usar para construir la aplicación ya que el desarrollador no conocía los lenguajes elegidos.
- Diseño de la arquitectura inicial: Se realiza la primera arquitectura que es la que se irá siguiendo a lo largo del proyecto marcada por los requisitos.
- Diseño de la estructura de la base de datos: Se empieza a implementar la base de datos definiendo todos los datos que vamos a usar.
- Implementación de los objetivos: La parte de desarrollo del sistema, son varias tareas en las que dejamos hecho el código de la aplicación de cada uno de los objetivos que nos habíamos marcado implementando los casos de uso.
- Pruebas del funcionamiento del sistema: Pruebas con las implementaciones terminadas en las que comprobamos el correcto funcionamiento de toda la aplicación y corregimos los fallos que aparezcan.
- Reunión final con los tutores del proyecto: Reunión en la que se muestra el resultado final del trabajo, los tutores corrigen los errores, añaden funcionalidades y dan las directrices necesarias para iniciar la documentación del proyecto.
- Realizar la documentación: Con todo el sistema terminado y con las indicaciones de los tutores se realiza la documentación de todo el proceso, juntando aquellas partes que se han ido documentando durante la realización del sistema y lo que se indica. Este proceso tiene una duración de un mes.

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos.

Memoria final

- Implementación del despliegue de los subsistemas en un servidor: Un requisito fundamental es que la aplicación sea una aplicación web y para ello tiene que ser accesible desde el navegador de un dispositivo en cualquier red. Para ello se monta un servidor con un dominio 'smpi.no-ip.org' desde el que se accede a la aplicación.

Para comprobar y entender mejor esta planificación del proyecto se encuentra detallada en el Anexo 1 mostrando en este documento cada una de las iteraciones y tareas que las componen junto con la fecha aproximada de inicio y finalización de cada una. Acompañado de un diagrama de Gantt para el seguimiento de este proceso.

Se muestra el diagrama de Gantt con todas las tareas para indicar los tiempos que ha llevado realizar el trabajo:

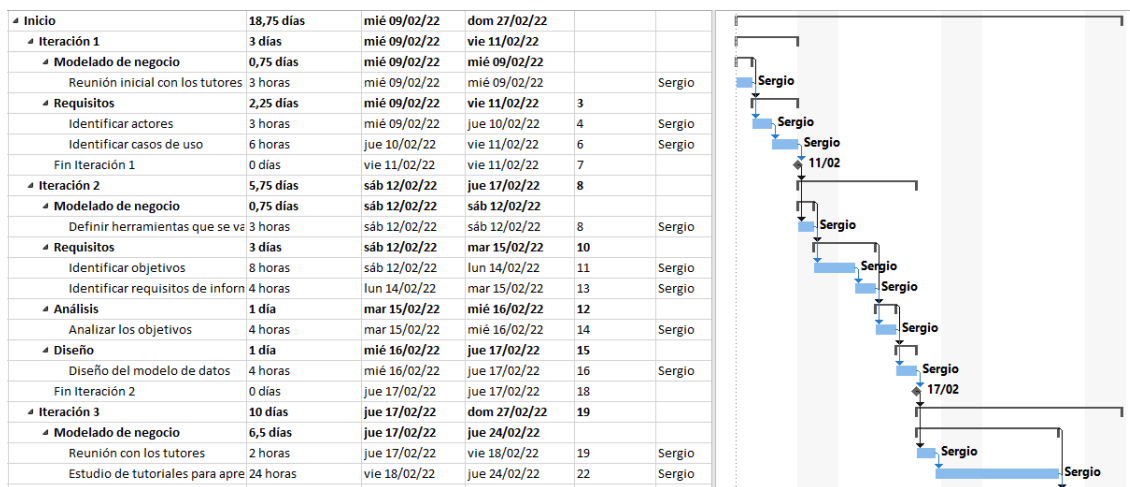


Ilustración 13. Diagrama de Gantt con la planificación del proyecto (parte 1). Fuente: elaboración propia

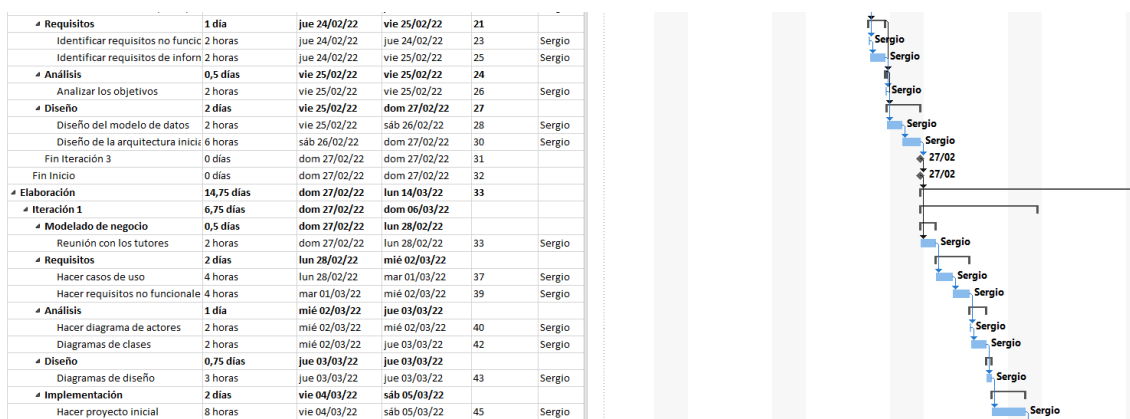


Ilustración 14. Diagrama de Gantt con la planificación del proyecto (parte 2). Fuente: elaboración propia

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos.

Memoria final

Pruebas	0,5 días	dom 06/03/22	dom 06/03/22		
Hacer pruebas de proyecto inici	2 horas	dom 06/03/22	dom 06/03/22	47	Sergio
Fin Iteración 1	0 días	dom 06/03/22	dom 06/03/22	49	
Iteración 2	8 días	dom 06/03/22	lun 14/03/22	50	
Modelado de negocio	1,5 días	dom 06/03/22	lun 07/03/22		
Reunión con los tutores	1 hora	dom 06/03/22	dom 06/03/22	50	Sergio
Estudio de tutoriales para aprende	5 horas	dom 06/03/22	lun 07/03/22	53	Sergio
Requisitos	2 días	mar 08/03/22	mié 09/03/22		
Revisión de los casos de uso y	2 horas	mar 08/03/22	mar 08/03/22	54	Sergio
Hacer diagramas de secuencia	4 horas	mar 08/03/22	mié 09/03/22	56	Sergio
Hacer casos de uso	2 horas	mié 09/03/22	mié 09/03/22	57	Sergio
Análisis	0,75 días	jue 10/03/22	jue 10/03/22		
Documentación de las clases	3 horas	jue 10/03/22	jue 10/03/22	58	Sergio
Diseño	1 día	jue 10/03/22	vie 11/03/22		
Diseño de la estructura de la b	4 horas	jue 10/03/22	vie 11/03/22	60	Sergio
Implementación	1,75 días	vie 11/03/22	dom 13/03/22		
Creación de la base de datos	7 horas	vie 11/03/22	dom 13/03/22	62	Sergio
Pruebas	1 día	dom 13/03/22	lun 14/03/22		
Pruebas de la base de datos	4 horas	dom 13/03/22	lun 14/03/22	64	Sergio
Fin Iteración 2	0 días	lun 14/03/22	lun 14/03/22	66	
Fin Elaboración	0 días	lun 14/03/22	lun 14/03/22	67	

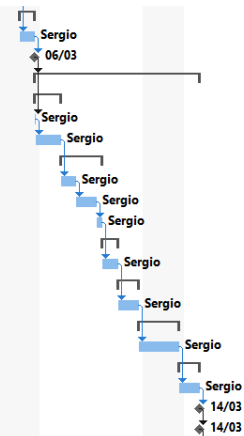


Ilustración 15. Diagrama de Gantt con la planificación del proyecto (parte 3). Fuente: elaboración propia

Construcción	75,25 días	lun 14/03/22	vie 10/06/22	68	
Iteración 1	22,5 días	lun 14/03/22	mar 05/04/22	68	
Modelado de negocio	1,5 días	lun 14/03/22	mar 15/03/22		
Investigación y pruebas	6 horas	lun 14/03/22	mar 15/03/22		Sergio
Requisitos	1,25 días	mié 16/03/22	jue 17/03/22		
Refinamiento de requisitos de	5 horas	mié 16/03/22	jue 17/03/22	72	Sergio
Análisis	1,75 días	jue 17/03/22	vie 18/03/22		
Refinamiento de casos de uso	7 horas	jue 17/03/22	vie 18/03/22	74	Sergio
Diseño	1,25 días	sáb 19/03/22	dom 20/03/22		
Refinamiento del diseño	5 horas	sáb 19/03/22	dom 20/03/22	76	Sergio
Implementación	10,5 días	dom 20/03/22	mié 30/03/22		
Implementación del objetivo 1	25 horas	dom 20/03/22	sáb 26/03/22	78	Sergio
Implementación del objetivo 1	17 horas	sáb 26/03/22	mié 30/03/22	80	Sergio
Pruebas	6,25 días	mié 30/03/22	mar 05/04/22		
Pruebas de la base de datos	8 horas	mié 30/03/22	vie 01/04/22	81	Sergio
Pruebas unitarias del objetivo	8 horas	vie 01/04/22	dom 03/04/22	83	Sergio
Pruebas unitarias del objetivo	8 horas	dom 03/04/22	mar 05/04/22	84	Sergio
Pruebas del funcionamiento d	1 hora	mar 05/04/22	mar 05/04/22	85	Sergio
Fin Iteración 1	0 días	mar 05/04/22	mar 05/04/22	86	

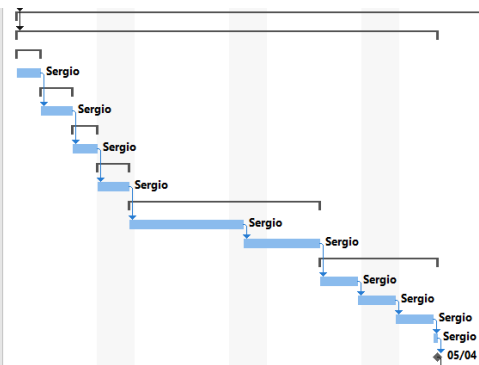


Ilustración 16. Diagrama de Gantt con la planificación del proyecto (parte 4). Fuente: elaboración propia

Iteración 2	16,25 días	mié 06/04/22	jue 28/04/22	87	
Modelado de negocio	1,5 días	mié 06/04/22	jue 07/04/22		
Reunión con los tutores para m	2 horas	mié 06/04/22	mié 06/04/22	87	Sergio
Investigación y pruebas	4 horas	mié 06/04/22	jue 07/04/22	90	Sergio
Requisitos	0,75 días	jue 07/04/22	vie 08/04/22		
Refinamiento de requisitos de	3 horas	jue 07/04/22	vie 08/04/22	91	Sergio
Análisis	1 día	vie 08/04/22	sáb 09/04/22		
Refinamiento del objetivo 1Ge:	2 horas	vie 08/04/22	vie 08/04/22	93	Sergio
Refinamiento del objetivo 1Ge:	2 horas	vie 08/04/22	sáb 09/04/22	95	Sergio
Diseño	1 día	sáb 09/04/22	dom 10/04/22		
Refinamiento del diseño	1 día	sáb 09/04/22	dom 10/04/22	96	Sergio
Implementación	8 días	dom 10/04/22	sáb 23/04/22		
Implementación del objetivo 1	4 horas	dom 10/04/22	lun 11/04/22	98	Sergio
Implementación del objetivo 1	3 horas	lun 11/04/22	lun 11/04/22	100	Sergio
Implementación del objetivo 1	20 horas	sáb 16/04/22	jue 21/04/22	101	Sergio
Implementación del objetivo 1	5 horas	vie 22/04/22	sáb 23/04/22	102	Sergio
Pruebas	4 días	sáb 23/04/22	jue 28/04/22		
Pruebas de la base de datos	2 horas	sáb 23/04/22	sáb 23/04/22	103	Sergio
Pruebas unitarias del objetivo	2 horas	sáb 23/04/22	lun 25/04/22	105	Sergio
Pruebas unitarias del objetivo	2 horas	lun 25/04/22	lun 25/04/22	106	Sergio
Pruebas unitarias del objetivo	6 horas	lun 25/04/22	mié 27/04/22	107	Sergio
Pruebas unitarias del objetivo	4 horas	mié 27/04/22	jue 28/04/22	108	Sergio
Fin iteración 2	0 días	jue 28/04/22	jue 28/04/22	109	

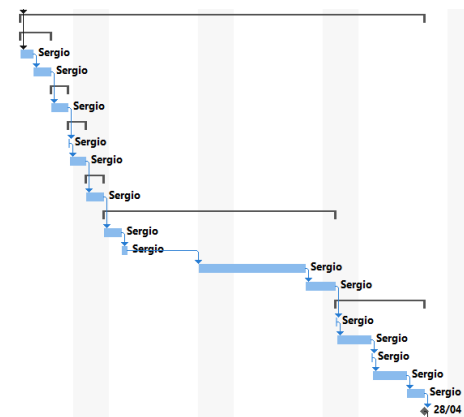


Ilustración 17. Diagrama de Gantt con la planificación del proyecto (parte 5). Fuente: elaboración propia

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos.

Memoria final

Iteración 3	17 días	jue 28/04/22	mié 18/05/22	110	
Modelado de negocio	0,75 días	jue 28/04/22	jue 28/04/22		
Reunión con los tutores para r	1 hora	jue 28/04/22	jue 28/04/22	110	Sergio
Investigación y pruebas	2 horas	jue 28/04/22	jue 28/04/22	113	Sergio
Requisitos	0,25 días	vie 29/04/22	vie 29/04/22		
Refinamiento de requisitos de	1 hora	vie 29/04/22	vie 29/04/22	114	Sergio
Análisis	4 días	vie 29/04/22	mié 04/05/22		
Refinamiento de objetivo 'Ge	8 horas	vie 29/04/22	lun 02/05/22	116	Sergio
Refinamiento de objetivo 'Ge	8 horas	lun 02/05/22	mié 04/05/22	118	Sergio
Diseño	0,25 días	mié 04/05/22	mié 04/05/22		
Refinamiento del diseño	1 hora	mié 04/05/22	mié 04/05/22	119	Sergio
Implementación	7,75 días	mié 04/05/22	vie 13/05/22		
Implementación del objetivo 's	5 horas	mié 04/05/22	jue 05/05/22	121	Sergio
Implementación del objetivo 's	26 horas	jue 05/05/22	vie 13/05/22	123	Sergio
Pruebas	4 días	vie 13/05/22	mié 18/05/22		
Pruebas unitarias del objetivo	6 horas	vie 13/05/22	sáb 14/05/22	124	Sergio
Pruebas unitarias del objetivo	9 horas	sáb 14/05/22	mar 17/05/22	126	Sergio
Pruebas del funcionamiento di	1 hora	mié 18/05/22	mié 18/05/22	127	Sergio
Fin Iteración 3	0 días	mié 18/05/22	mié 18/05/22	128	

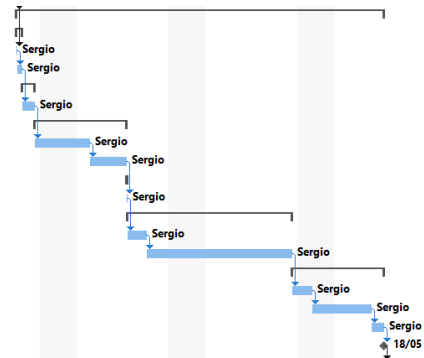


Ilustración 18. Diagrama de Gantt con la planificación del proyecto (parte 6). Fuente: elaboración propia

Iteración 4	19,5 días	mié 18/05/22	vie 10/06/22	129	
Modelado de negocio	1 día	mié 18/05/22	jue 19/05/22		
Reunión con los tutores para r	1 hora	mié 18/05/22	mié 18/05/22	129	Sergio
Investigación y pruebas	3 horas	mié 18/05/22	jue 19/05/22	132	Sergio
Requisitos	0,25 días	jue 19/05/22	jue 19/05/22		
Refinamiento de requisitos de	1 hora	jue 19/05/22	jue 19/05/22	133	Sergio
Análisis	3,75 días	jue 19/05/22	mar 24/05/22		
Refinamiento del objetivo 'Ge	12 horas	jue 19/05/22	lun 23/05/22	135	Sergio
Refinamiento de casos de uso	3 horas	lun 23/05/22	mar 24/05/22	137	Sergio
Diseño	0,5 días	mar 24/05/22	mar 24/05/22		
Refinamiento del diseño	2 horas	mar 24/05/22	mar 24/05/22	138	Sergio
Implementación	7,5 días	mar 24/05/22	jue 02/06/22		
Implementación del objetivo 'C	30 horas	mar 24/05/22	jue 02/06/22	140	Sergio
Pruebas	6,5 días	jue 02/06/22	vie 10/06/22		
Pruebas unitarias del objetivo	10 horas	jue 02/06/22	sáb 04/06/22	142	Sergio
Pruebas del funcionamiento di	16 horas	sáb 04/06/22	vie 10/06/22	144	Sergio
Fin Iteración 4	0 días	vie 10/06/22	vie 10/06/22	145	
Fin Construcción	0 días	vie 10/06/22	vie 10/06/22	146	

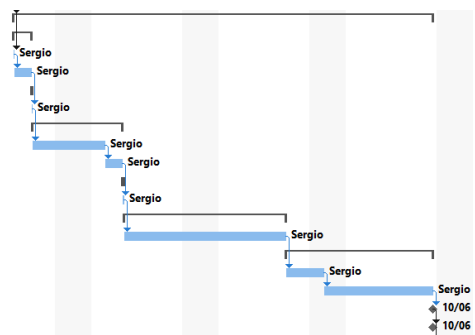


Ilustración 19. Diagrama de Gantt con la planificación del proyecto (parte 7). Fuente: elaboración propia

Transición	56,25 días	vie 10/06/22	lun 15/08/22		
Iteración 1	9 días	vie 10/06/22	mar 21/06/22		
Modelado de negocio	0,75 días	vie 10/06/22	sáb 11/06/22		
Reunión con los tutores para r	3 horas	vie 10/06/22	sáb 11/06/22	146	Sergio
Requisitos	0,75 días	sáb 11/06/22	lun 13/06/22		
Verificación de los requisitos	3 horas	sáb 11/06/22	lun 13/06/22	151	Sergio
Análisis	2 días	lun 13/06/22	mié 15/06/22		
Análisis de los resultados	8 horas	lun 13/06/22	mié 15/06/22	153	Sergio
Diseño	1 día	mié 15/06/22	jue 16/06/22		
Programación de la documenta	4 horas	mié 15/06/22	jue 16/06/22	155	Sergio
Implementación	3 días	jue 16/06/22	lun 20/06/22		
Aplicar cambios en el desarroll	12 horas	jue 16/06/22	lun 20/06/22	157	Sergio
Pruebas	1,5 días	lun 20/06/22	mar 21/06/22		
Pruebas de interfaz	6 horas	lun 20/06/22	mar 21/06/22	159	Sergio
Fin Iteración 1	0 días	mar 21/06/22	mar 21/06/22	161	
Iteración 2	10 días	mar 21/06/22	sáb 02/07/22	162	
Modelado de negocio	0,75 días	mar 21/06/22	mié 22/06/22		
Reunión con los tutores para r	3 horas	mar 21/06/22	mié 22/06/22	162	Sergio
Diseño	2,25 días	mié 22/06/22	vie 24/06/22		
Inicio de la documentación	9 horas	mié 22/06/22	vie 24/06/22	165	Sergio
Implementación	2,25 días	vie 24/06/22	lun 27/06/22		
Implementación del despliegue	9 horas	vie 24/06/22	lun 27/06/22	167	Sergio
Pruebas	3,75 días	mar 28/06/22	vie 01/07/22		
Pruebas finales de la aplicació	15 horas	mar 28/06/22	vie 01/07/22	169	Sergio
Fin Iteración 2	1 día	vie 01/07/22	sáb 02/07/22	171	

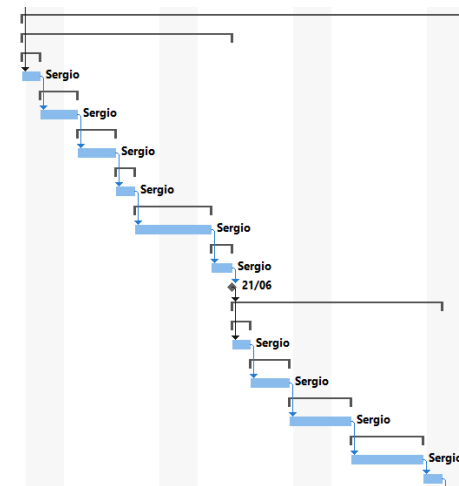


Ilustración 20. Diagrama de Gantt con la planificación del proyecto (parte 8). Fuente: elaboración propia

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos.

Memoria final

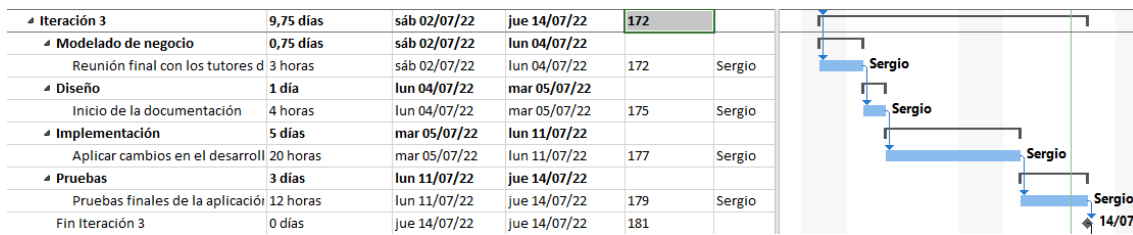


Ilustración 21. Diagrama de Gantt con la planificación del proyecto (parte 9). Fuente: elaboración propia



Ilustración 22. Diagrama de Gantt con la planificación del proyecto (parte 10). Fuente: elaboración propia

5.3. Arquitectura del sistema

En este apartado se explican las tecnologías y herramientas utilizadas para el desarrollo del proyecto, las cuales se justificarán para entender por qué esta arquitectura es la mejor para la solución del problema que plantea Global Exchange. En la ilustración adjunta se describen las partes principales, cliente y servidor, y en los próximos subapartados se explica con mayor detalle cada una de ellas.

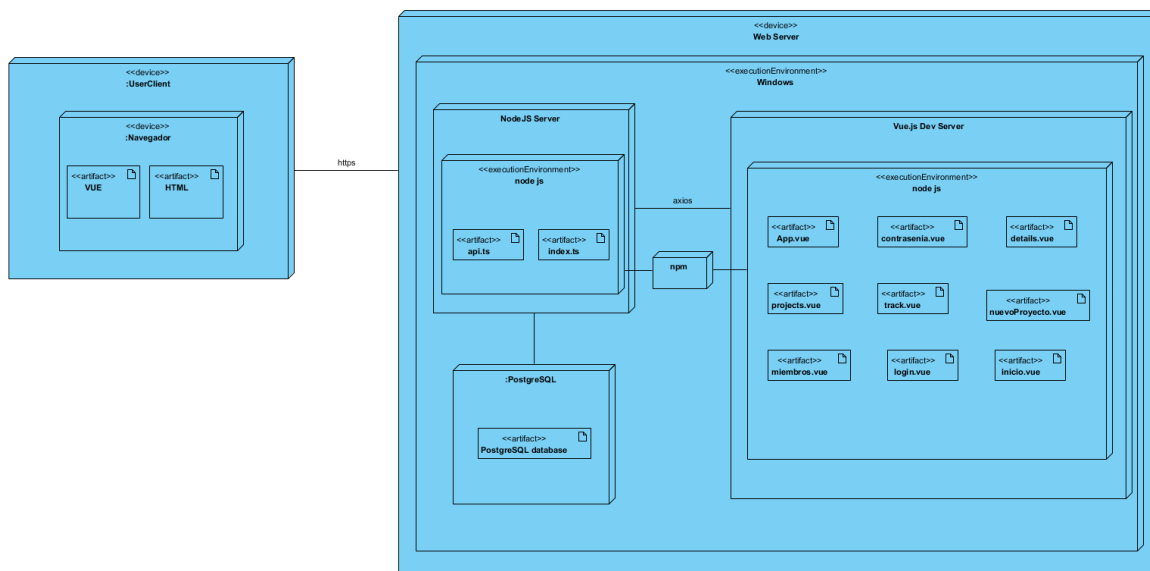


Ilustración 23. Esquema con la arquitectura cliente-servidor propuesta. Fuente: elaboración propia

5.3.1. Cliente

Dado que se trata de una arquitectura Cliente-Servidor [18] que responde a peticiones http, el cliente se conectará al servidor a través de un dominio mediante un navegador web que mostrará al servidor las vistas de Vue que muestre la aplicación.

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

Esta arquitectura es elegida ya que las especificaciones de Global Exchange dejaron clara la necesidad de que esta aplicación fuera un servicio web. Tal y como se observa en la ilustración adjunta, en la parte Cliente se accede desde el navegador de un dispositivo, que mostrará todas las vistas de la aplicación. Estas vistas se sirven con *scripts* de Vue y HTML.

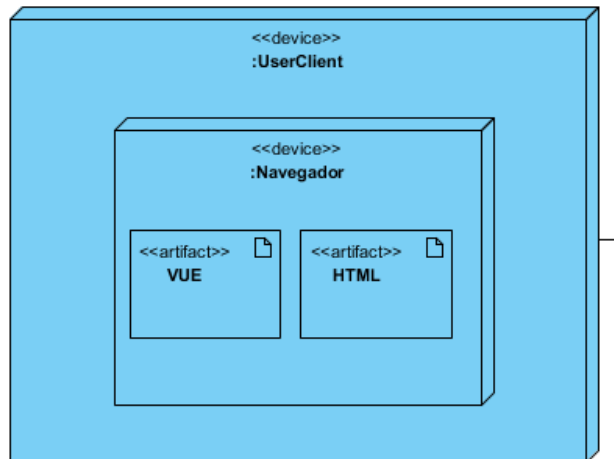


Ilustración 24. Esquema de la parte cliente. Fuente: elaboración propia

5.3.2. Servidor

Este servidor se divide en dos partes: el proyecto *frontend* y el *backend*. Para este trabajo fin de grado, estas partes se han instalado y se ejecutan localmente en un ordenador Windows con los puertos del *router* abiertos y un dominio creado para que sea accesible desde cualquier red a través de internet.

Como ya he explicado, la empresa Global Exchange puso como requisito fundamental que este proyecto fuera una aplicación web. Por ello, se ha montado de tal forma que sea accesible desde cualquier punto.

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos.
Memoria final

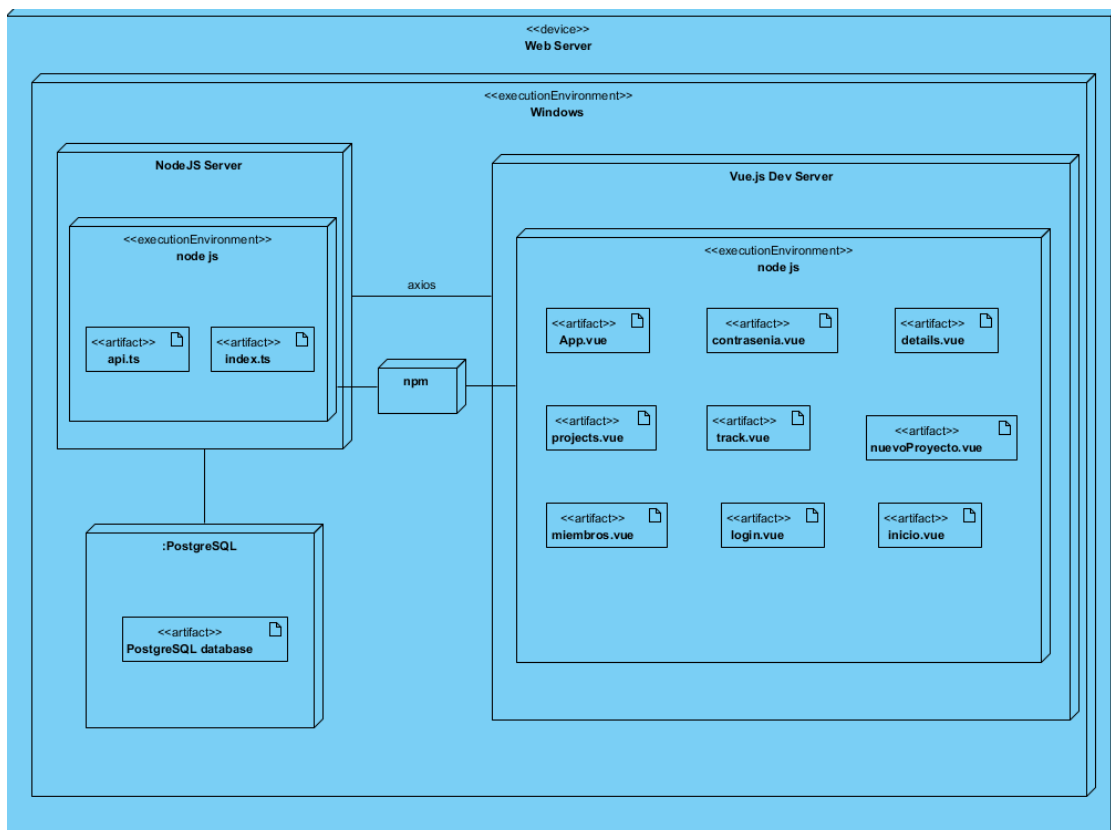


Ilustración 25. Esquema con los componentes principales de la parte servidora Fuente: elaboración propia

Tal y como se aprecia en la ilustración adjunta, los componentes principales de la parte servidora se explican a continuación, agrupándolos en *backend* y *frontend*.

5.3.2.1. Backend

Esta es la capa de acceso y control de los datos y en el esquema de la ilustración se representa mediante el recuadro de NodeJS Server y PostgreSQL. En este proyecto se trata de una API en un servidor NodeJS como se ve en la imagen montado en una carpeta con las configuraciones necesarias para montar el servidor Node y los *scripts* para montar la base de datos PostgreSQL, la cual se gestionará mediante *scripts*, escritos en TypeScript que se ven en los <<artiffact>> de la ilustración *api.ts* y *index.ts*, que gestionan las llamadas http de la aplicación.

La base de datos PostgreSQL es elegida ya que es de código abierto y simple de usar, con código *pgsql* similar a SQL usado anteriormente por el desarrollador, con PostgreSQL podemos gestionar la base de datos como se marca en los requisitos. Se muestra en la imagen junto con el *script* que va a controlarla escrito en lenguaje PLpgSQL.

Usamos un servidor de Node para servir esta parte ya que se adapta perfectamente a lo que queremos recibiendo peticiones HTTP. Gestionándolo por medio de unos *scripts* para montar el api con las diferentes rutas pedidas por el sistema desde la parte de *frontend*, código escrito en TypeScript ya que añade funciones y métodos a JavaScript y Node es el entorno de ejecución de este.

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

5.3.2.2. *Frontend*

Esta es la parte accesible por el usuario, muestra las páginas a las que accede el usuario por medio del navegador web. En el esquema de la ilustración esta parte se representa mediante el recuadro de Vue.js Dev Server que es el servidor que lo va a ejecutar.

El entorno de ejecución se muestra en la imagen dentro de esta parte, Node, esta parte es desarrollada mayormente con Vue ya que es un lenguaje parecido a HTML (también usado para parte de la vista) con el que somos capaces de construir interfaces de usuario. Los *scripts* son los que se muestran en pantalla, que como se ha explicado, todos son *.vue*. Este lenguaje trabaja con JavaScript que lo usamos para gestionar las vistas y las funcionalidades que pueden ofrecer, conectando así esta parte con el servidor de *backend*. Además, tiene su propia sección de estilos, con código CSS con el que definimos el diseño gráfico de cada página.

La elección de estos lenguajes y esta arquitectura en esta parte es de nuevo por ejecutar el código en el mismo entorno de desarrollo, NodeJS, aunque en este caso será con VueJS y ya que para implementar la parte gráfica este código es una buena opción.

Este proyecto tiene 9 *scripts* *.vue* que muestra las vistas de cada una de las páginas.

5.3.3. Justificación

Se ha elegido como entorno de desarrollo y lenguajes aquellos con los que se pudieran cumplir los requisitos fundamentales recogidos al principio del proyecto expuestos por la empresa a la que va destinada este proyecto.

El entorno de desarrollo es Node con el que podemos montar un servidor web.

El lenguaje de programación de la parte gráfica es Vue con el cual conseguimos generar vistas para páginas web y gestionado por JavaScript para controlar la funcionalidad de esta parte *frontend*.

El lenguaje de *backend* utilizado para gestionar los datos de la aplicación es TypeScript para hacer más sencillo el uso de JavaScript y para añadir funcionalidades.

El lenguaje para controlar la base de datos es postgresql, lenguaje de PostgreSQL que es la base de datos elegida por su simplicidad y su código abierto.

5.4. Especificación de requisitos

En este apartado se va a explicar una de las primeras partes del proyecto, los requisitos, que fueron recogidos en las primeras reuniones con los tutores.

En estas reuniones se discutieron cuáles serían las funcionalidades finales del sistema y los requisitos fundamentales que debería tener esta aplicación.

Como ya se ha explicado en esta memoria un requisito fundamental era crear la solución como aplicación web, además de esto se especificaron los requisitos recogidos como objetivos en el Anexo 2 de la documentación.

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

OBJ-0001	Gestión de usuarios
Versión	2.0 (15/04/2022)
Autores	Sergio Sánchez Antona
Fuentes	
Descripción	El sistema debe ser capaz de gestionar los usuarios, registrando sus datos de email, contraseña y ron de este (cifrando la contraseña), también deberá gestionar un inicio de sesión comparando la contraseña que se introduzca con la registrada en el sistema y diferenciando entre los roles para permitir unas opciones u otras.
Subobjetivos	Ninguno
Importancia	Vital
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

OBJ-0002	Gestión de proyectos
Versión	1.0 (15/02/2022)
Autores	Sergio Sánchez Antona
Fuentes	
Descripción	El sistema debe ser capaz de gestionar los proyectos pudiendo el usuario que sea administrador añadir proyectos recogiendo los datos que introduzca el usuario y almacenándolos en la base de datos. Además, el sistema después tiene que ser capaz de leer de la base de datos los proyectos que hay en el sistema para gestionarlos en los demás objetivos.
Subobjetivos	Ninguno
Importancia	Vital
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

OBJ-0003	Gestión de miembros
Versión	1.0 (15/02/2022)
Autores	Sergio Sánchez Antona
Fuentes	
Descripción	El sistema deberá permitir gestionar los miembros de los proyectos garantizando que el usuario introduzca todos los datos, los datos que debe añadir son el proyecto en el que va a estar asignado, el nombre del miembro que no puede coincidir con otro nombre del mismo proyecto y el rol que va a desempeñar en el proyecto. Deberá permitir borrar el miembro que se desee mediante un botón.
Subobjetivos	[OBJ-0002]-Gestión de proyectos
Importancia	Vital
Estado	En construcción

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

Estabilidad	Alta
Comentarios	Ninguno

OBJ-0004	Gestión de tareas
Versión	1.0 (17/02/2022)
Autores	Sergio Sánchez Antona
Fuentes	
Descripción	<p>El sistema deberá permitir gestionar las tareas de cada proyecto, garantizando que el usuario introduzca todos los datos que serán el proyecto del que se va a poner la tarea, el nombre de la tarea, las horas que lleva trabajadas en la tarea, las horas estimadas que llevará la tarea, la prioridad, la fecha de la última modificación de la tarea y el miembro propietario de esta.</p> <p>Además, debe permitir actualizar esta tarea con un botón con el que se pueda añadir horas a la tarea si se ha trabajado después en esta y cambiar la fecha de modificación de esta.</p> <p>El sistema sumará automáticamente las horas añadidas a las horas totales del proyecto.</p> <p>Permitirá borrar una tarea eliminándola de la base de datos y restando las horas que llevaba esa tarea de las horas totales del proyecto.</p>
Subobjetivos	[OBJ-0003]-Gestión de miembros [OBJ-0002]-Gestión de proyectos
Importancia	Vital
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

OBJ-0005	Gestión de detalles
Versión	1.0 (17/02/2022)
Autores	Sergio Sánchez Antona
Fuentes	
Descripción	<p>El sistema deberá generar detalles sobre cada proyecto que ya esté creado mediante un botón en la página principal, este abrirá una nueva página en la que se mostrarán datos relevantes del proyecto y cada una de las tareas, además se generarán unas gráficas para monitorizar el trabajo de cada miembro pudiendo ver cuantas horas ha trabajado cada uno, cuantas tareas ha realizado y el rendimiento que ha tenido en el proyecto.</p>
Subobjetivos	[OBJ-004]-Gestión de tareas [OBJ-0003]-Gestión de miembros [OBJ-0002]-Gestión de proyectos
Importancia	Vital
Estado	En construcción
Estabilidad	Alta

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos.
Memoria final

Comentarios	Ninguno
-------------	---------

5.5. Etapa de análisis

Conocidos y definidos los requisitos fundamentales del sistema, se plantea el análisis del dominio del problema.

Como consecuencia de analizar los requisitos, se definen un diagrama de clases con los objetos las relaciones necesarias para plantear correctamente la aplicación. En la ilustración adjunta se observa el mencionado diagrama de clases obtenido tras la fase de análisis.

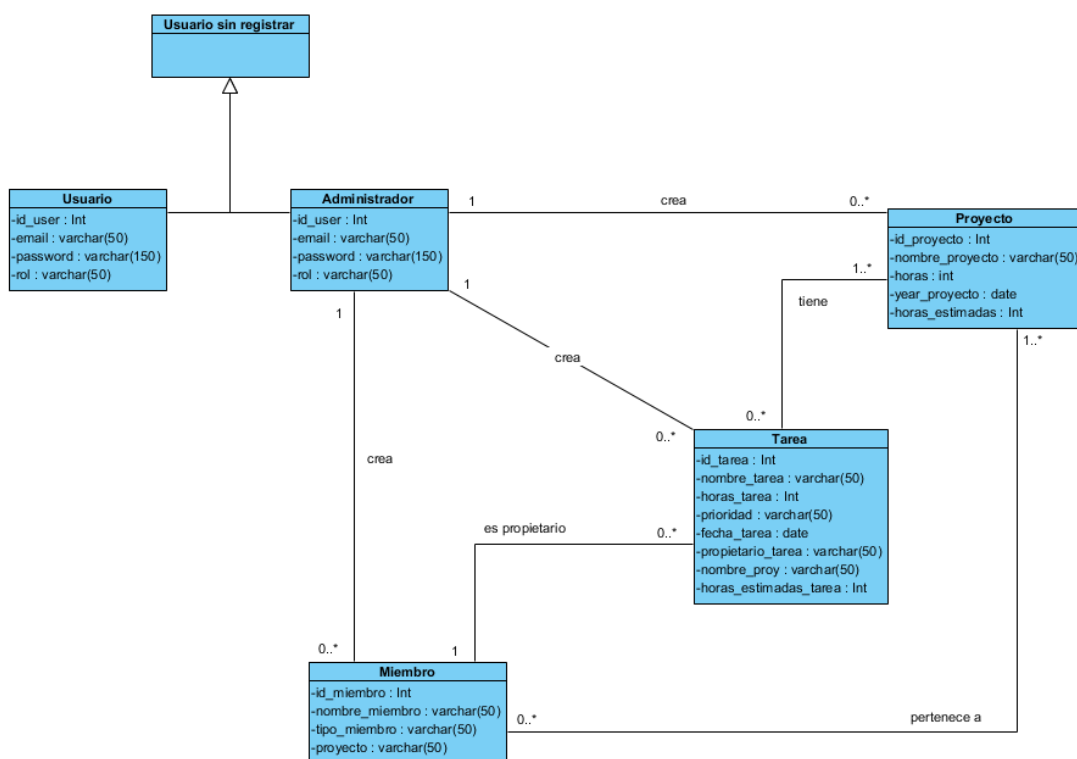


Ilustración 26. Diagrama de clases de la etapa de análisis. Fuente: elaboración propia

Se definen aquí los datos representados en el diagrama de clases:

Usuario: (users) Hace referencia a los usuarios cuando se registran y se guardan sus datos en el sistema con el rol de Usuario, deriva de usuario sin registrar del cual no se recogen datos.

- id_user: identificador único del usuario.
- email: Nombre con el que se identifica al usuario.
- password: Contraseña del usuario para iniciar sesión en el sistema.
- rol: Rol que va a desempeñar el usuario en el sistema (Administrador, Usuario)

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

Administrador: (users) Hace referencia a los usuarios cuando se registran y se guardan sus datos en el sistema con el rol de Administrador, tiene más funcionalidades que el rol Usuario, deriva de usuario sin registrar del cual no se recogen datos.

- id_user: identificador único del usuario.
- email: Nombre con el que se identifica al usuario.
- password: Contraseña del usuario para iniciar sesión en el sistema.
- rol: Rol que va a desempeñar el usuario en el sistema (Administrador, Usuario)

Proyecto: (projects) Clase que almacena la información de los proyectos del sistema

- id_proyecto: Int
- nombre_proyecto: varchar(50)
- horas: Int
- year_proyecto: date
- horas_estimadas: Int

Tarea: (tareass) Clase que almacena la información de las tareas de cada proyecto.

- id_tarea: Int
- nombre_tarea: varchar(50)
- horas_tarea: Int
- prioridad: varchar(50)
- fecha_tarea: date
- propietario_tarea: varchar(50)
- nombre_proy: varchar(50)
- horas_estimadas_tarea: Int

Miembro: (miembros) Clase que almacena la información de los miembros que realizan las tareas de cada proyecto.

- id_miembro: Int
- nombre_miembro: varchar(50)
- tipo_miembro: varchar(50)
- proyecto: varchar(50)

Encontramos información más detallada sobre las clases en el Anexo 3 de la documentación.

En esta fase también encontramos diagramas de secuencia teniendo en cuenta la relación de los objetos y los casos de uso. Se muestran en el apartado 6 del Anexo 3.

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

5.6. Etapa de diseño

Después de detallar el análisis se procede con el diseño. En esta fase es donde ya hemos definido las clases y los objetos, además de tener claros los requisitos y la secuencia que van a seguir los casos de uso.

Se empieza entonces a hacer una arquitectura en forma de paquetes siguiendo el esquema cliente-servidor ya que es el que tenemos que implementar.

Se definen también los patrones arquitectónicos y de diseño, siguiendo las especificaciones marcadas por la empresa. Se muestra en el Anexo 3 esta arquitectura más detallada.

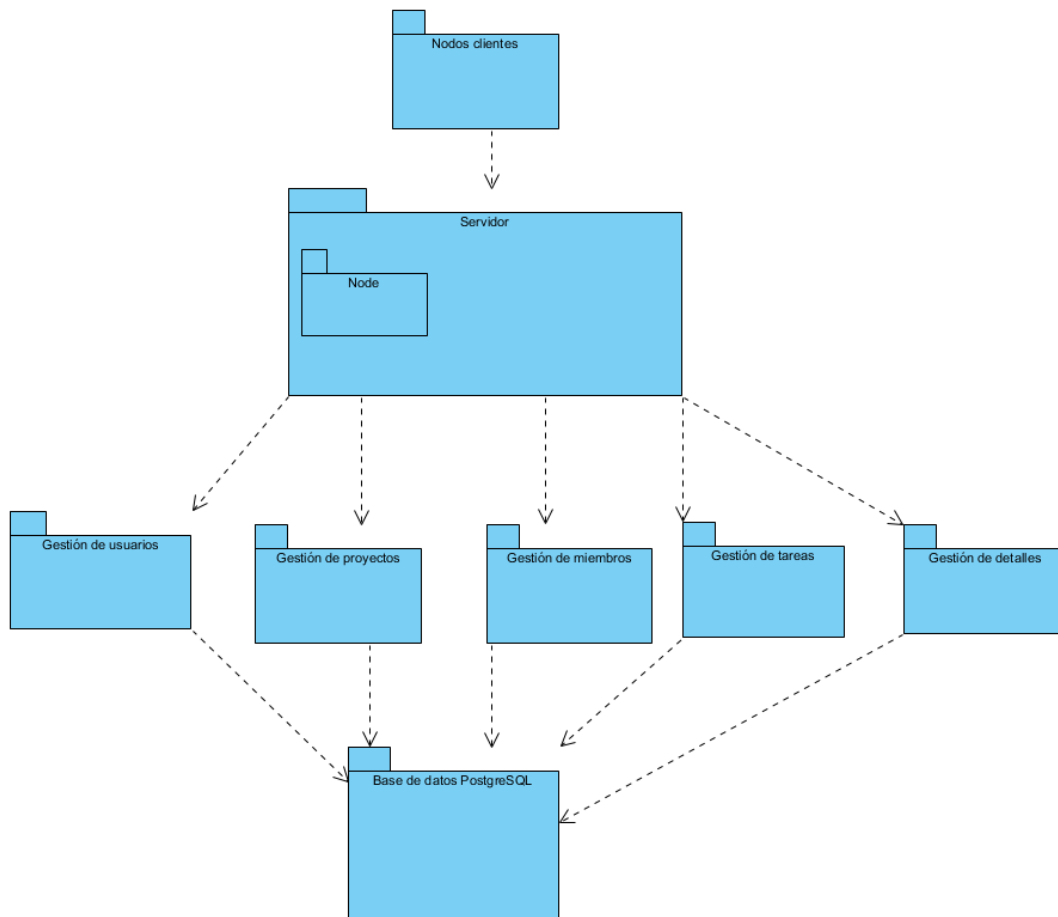


Ilustración 27. Arquitectura de capas. Fuente: elaboración propia

Tal y como se muestra en la ilustración adjunta, se muestra como primero los nodos clientes (que será un dispositivo por medio de un navegador) acceden al servidor de node, este servidor mostrará cada una de las gestiones a través de vistas en las diferentes páginas a las que se pueden acceder y estas acceden a la base de datos PostgreSQL para realizar las gestiones que el usuario indique con la interfaz que se le ha mostrado.

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos.
Memoria final

Para reflejar los casos de uso más importantes del sistema se muestran en las ilustraciones 28-36 los diagramas de secuencia de estos casos de uso que representan las funciones más importantes del sistema.

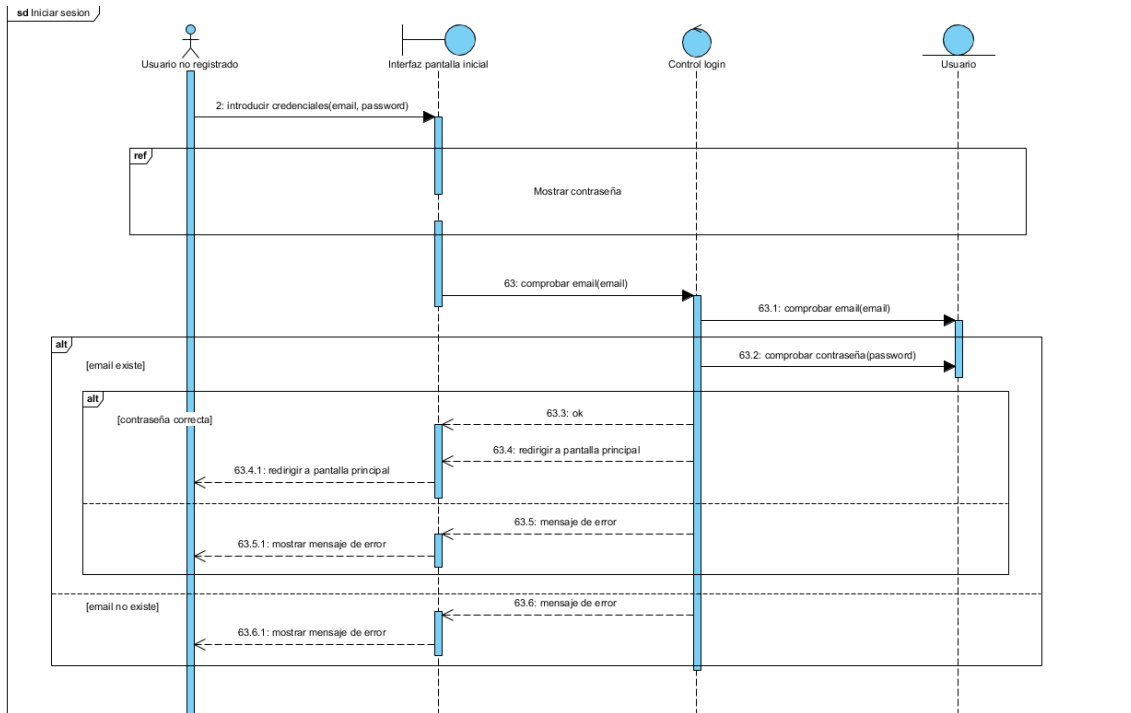


Ilustración 28. Diagrama de secuencia caso de uso 'Iniciar sesión'

Esta ilustración muestra el caso de uso Iniciar sesión, en el que se escriben las credenciales, con opción de mostrar la contraseña, después se comprueba que el email y la contraseña sean correctas y si lo son se redirige al usuario a la página principal.

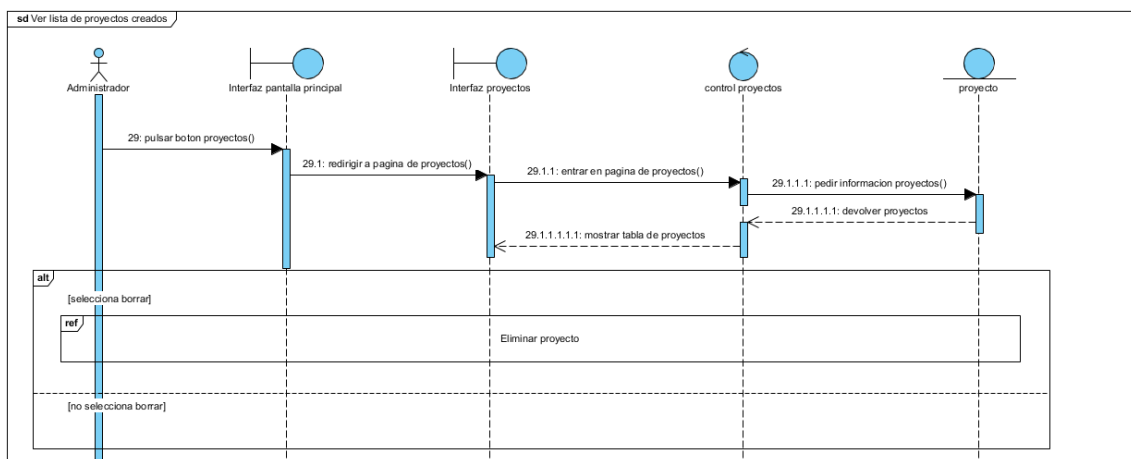


Ilustración 29. Diagrama de secuencia caso de uso 'Ver lista de proyectos creados'

En este caso de uso como se muestra en la ilustración 29 se ven las listas de los proyectos creados, solo puede el usuario con rol Administrador que se redirige a la página de

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos. Memoria final

proyectos mediante el botón de proyectos y el sistema muestra una tabla con los proyectos que hay creados.

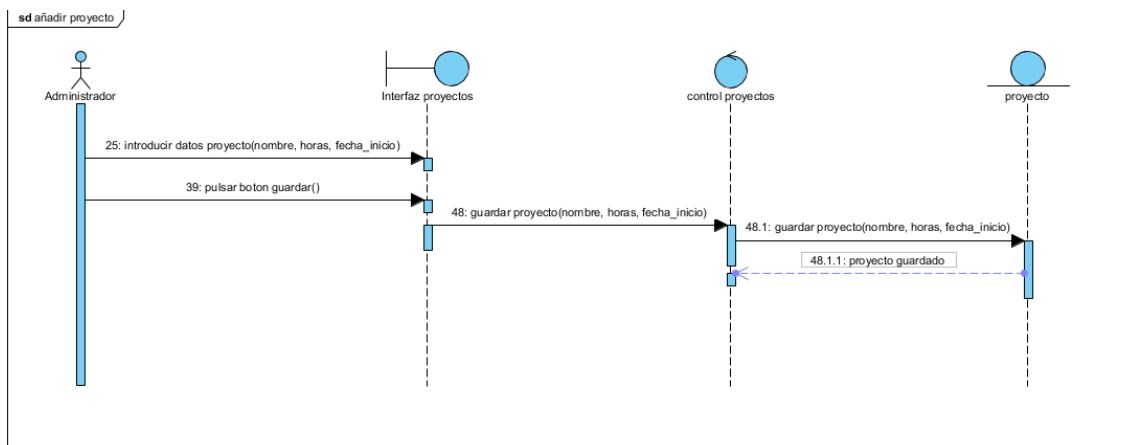


Ilustración 30. Diagrama de secuencia caso de uso 'Añadir proyecto'

En esta imagen vemos el caso de uso Añadir proyecto, en el cual el administrador puede introducir los datos de un nuevo proyecto y guardarlo, el sistema se encarga de guardarlo en la base de datos.

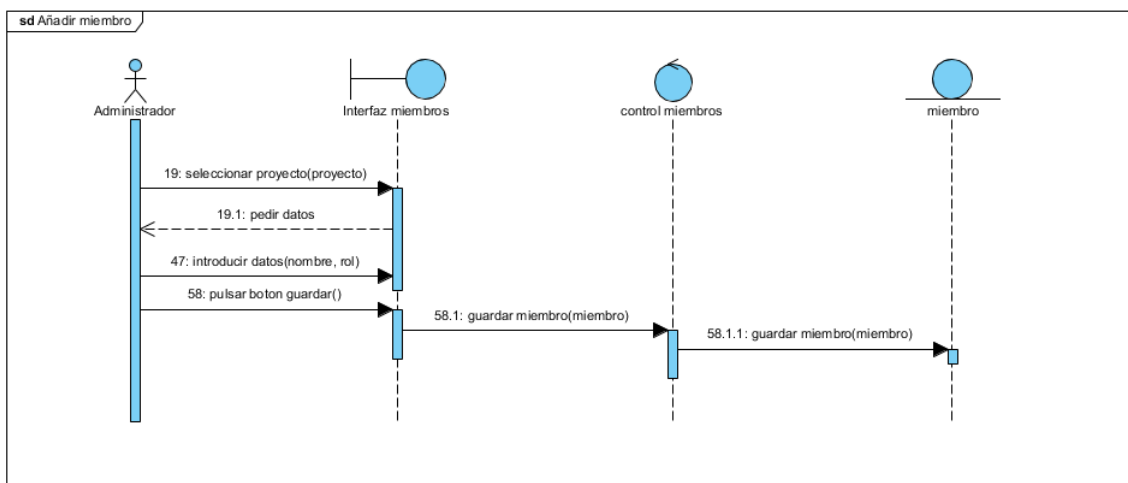


Ilustración 31. Diagrama de secuencia caso de uso 'Añadir miembro'

En la ilustración 31 vemos el caso de uso Añadir miembro en el que el administrador selecciona el proyecto del cual se va a añadir el miembro, introduce los datos y guarda el nuevo miembro.

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos. Memoria final

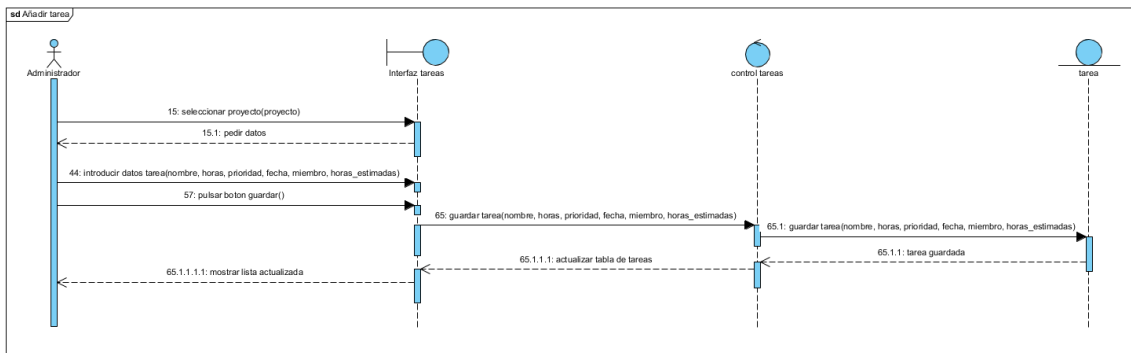


Ilustración 32. Diagrama de secuencia caso de uso 'Añadir tarea'

En el caso de uso Añadir tarea podemos ver en la imagen anterior el flujo de esta funcionalidad, en la que el administrador selecciona el proyecto del que se añadirá la tarea, se introducen los datos de la nueva tarea y se guarda en la base de datos.

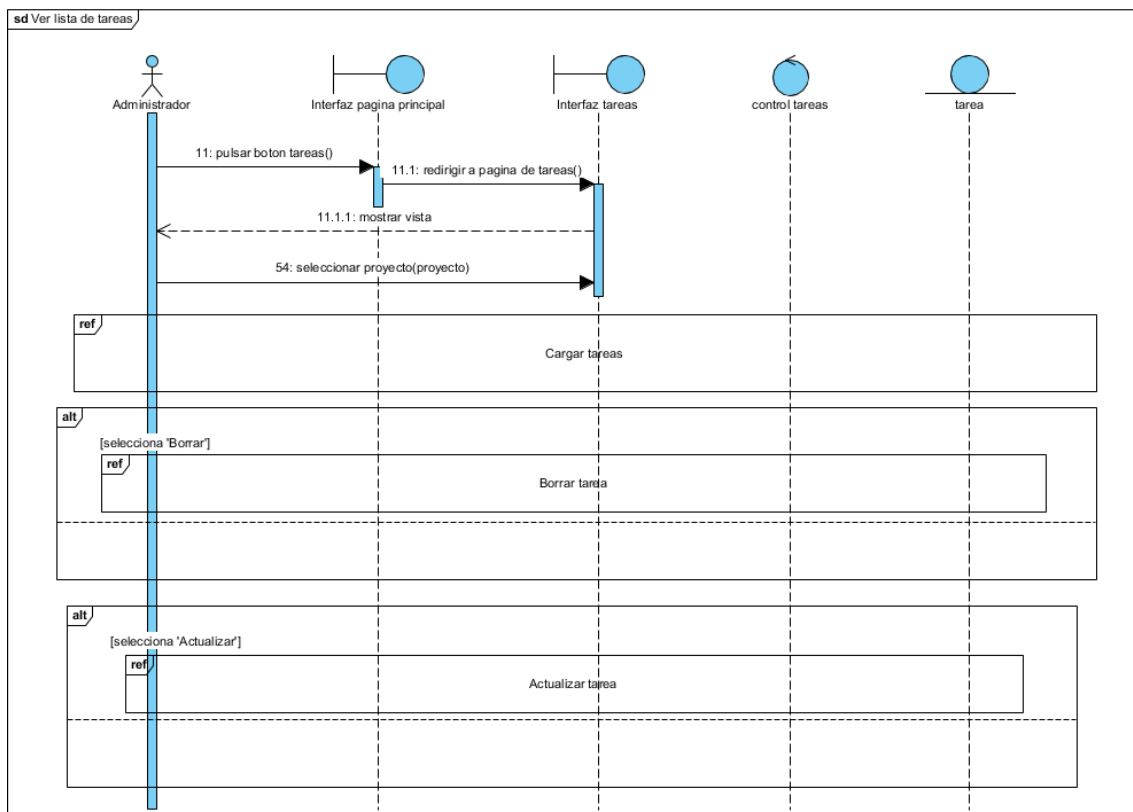


Ilustración 33. Diagrama de secuencia caso de uso 'Ver lista de tareas'

En la ilustración 33 podemos ver el diagrama de secuencia del caso de uso Ver lista de tareas, únicamente puede ver esta lista el administrador, pulsa en el botón de tareas y el sistema le redirige a la página en la que se muestran las tablas de las tareas seleccionando el proyecto del que quiere ver las tareas.

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos.

Memoria final

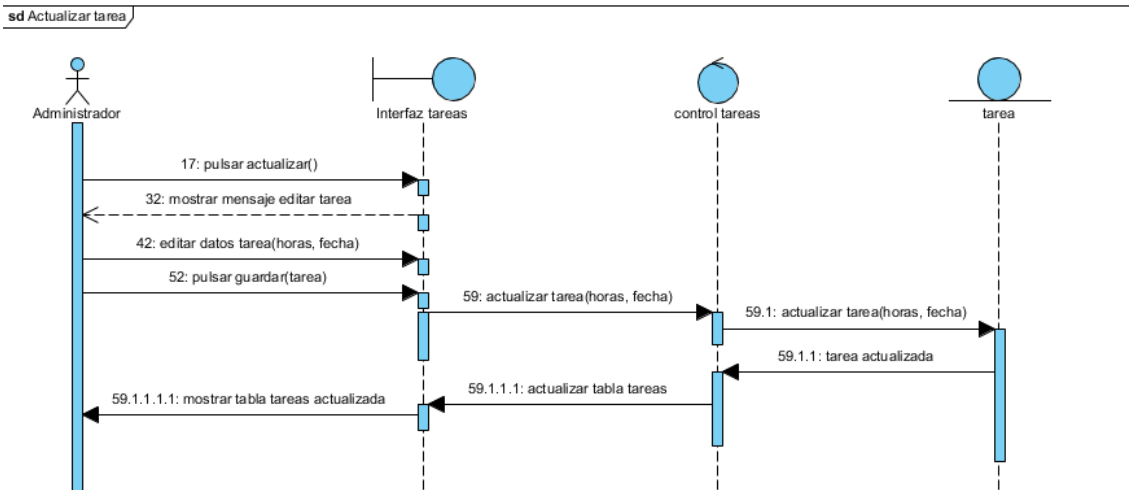


Ilustración 34. Diagrama de secuencia caso de uso 'Actualizar tarea'

En la ilustración 34 vemos el flujo del caso de uso actualizar tarea, el administrador pulsa en actualizar y se muestra un mensaje de editar la tarea, edita esta tarea y guarda los datos.

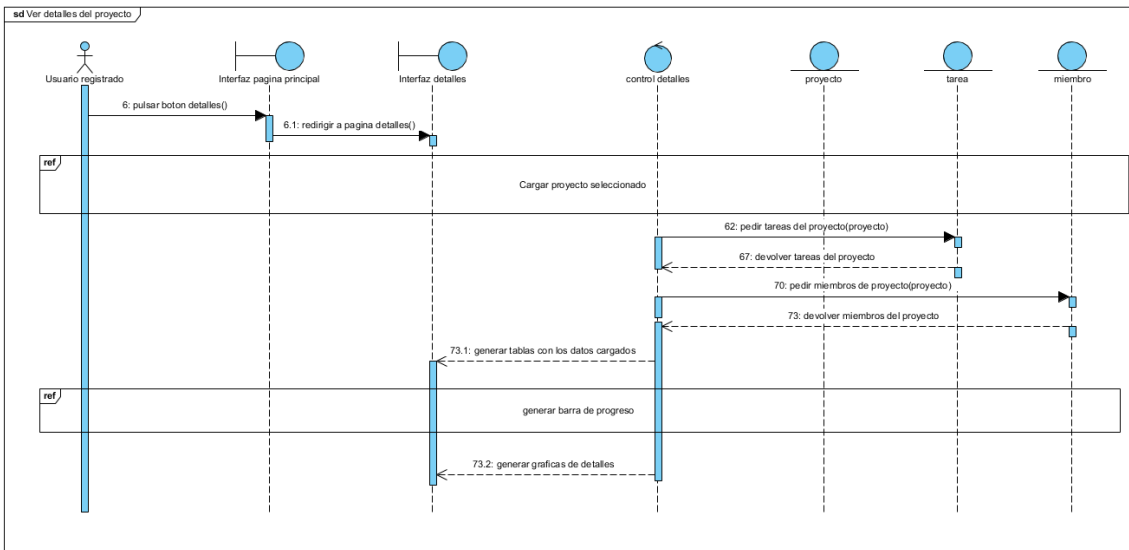


Ilustración 35. Diagrama de secuencia caso de uso 'Ver detalles del proyecto'

En el caso de uso Ver detalles del proyecto como se ve en la ilustración anterior el usuario registrado es el que la realiza, pero también la puede realizar el Administrador, se pone usuario registrado ya que se entiende que cualquier usuario con un rol por encima puede realizar las mismas funciones que los usuarios con un rol inferior, en este caso el Usuario registrado, y más funciones. Pulsa en el botón de detalles y se redirige a la página de detalles, en la cual el sistema genera todos los detalles del proyecto, miembros y tareas y se muestran al usuario.

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

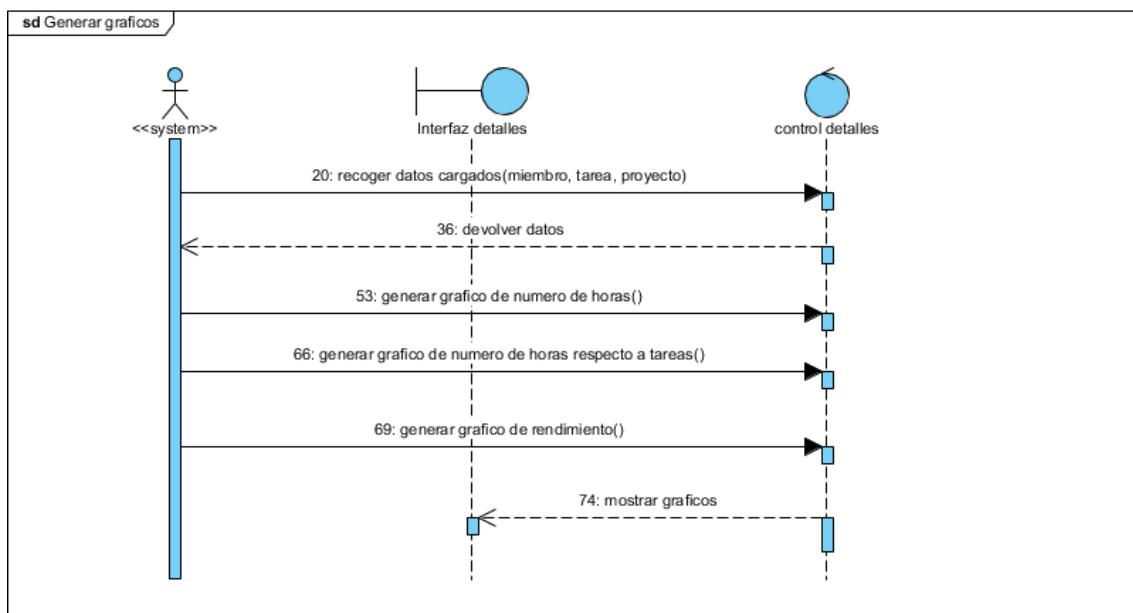


Ilustración 36. Diagrama de secuencia caso de uso 'Generar gráficos'

En la ilustración 36 vemos el diagrama de secuencia del caso de uso Generar gráficos que realiza el sistema, es un caso de uso importante ya que muestran mucha información del proyecto, primero recoge todos los datos que están cargados en la base de datos que necesite del proyecto del que se van a generar los gráficos y genera las tres gráficas que se muestran al usuario.

5.7. Etapa de implementación

En esta etapa se toma como base y referencia toda la información y esquemas obtenidos en los pasos anteriores para saber cómo implementar todo el código de la aplicación. Gracias a los pasos anteriores, cuando realizamos el desarrollo del código ya están claros cuales son los casos de uso y la arquitectura con cada uno de los lenguajes y las herramientas que se van a utilizar para la realización del sistema.

Como se explica en el Anexo 3 con detalle, se utilizan para esta aplicación Vue para la parte de *frontend* combinado con JavaScript, TypeScript para la parte de *backend* junto con PostgreSQL para la gestión de la base de datos. Además, usamos servidores Node.js y npm para la gestión de los paquetes.

Se realiza en esta fase la construcción del código que se ve y explica en el Anexo 4 de la documentación, después de haberlo realizado se muestra un diagrama de despliegue para entender el funcionamiento:

La aplicación web se ha implementado utilizando las partes y lenguajes descritos en el apartado de arquitectura. De cara a entender cómo están organizadas las pantallas que forman la interfaz de la aplicación, en el siguiente listado se enumeran las partes de la interfaz:

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

- Pantalla de inicio. Para iniciar sesión en el sistema y como introducción a este cuando se accede. Ilustración 24
- Pantalla de registro. Utilizada para registrarse en el sistema. Ilustración 25
- Home – Pantalla de bienvenida. Para inicial cuando se inicia sesión en el sistema. Ilustraciones 26, 27, 28.
- Pantalla de proyectos. Para gestionar los proyectos si se inicia sesión como Administrador. Ilustración 29
- Pantalla de miembros. Para gestionar los miembros y ver los ya creados si se inicia sesión como Administrador. Ilustración 30.
- Pantalla de tareas. Para gestionar las tareas si se inicia sesión como Administrador. Ilustraciones 31, 32.
- Pantalla de detalles. Para ver la información generada por el sistema para monitorizar los proyectos y los miembros mediante gráficos y tablas. Ilustraciones 33, 34, 35, 36.

A continuación, se adjuntan en diversas ilustraciones las pantallas de la parte *frontend* obtenidas como resultado de la implementación de la aplicación web.

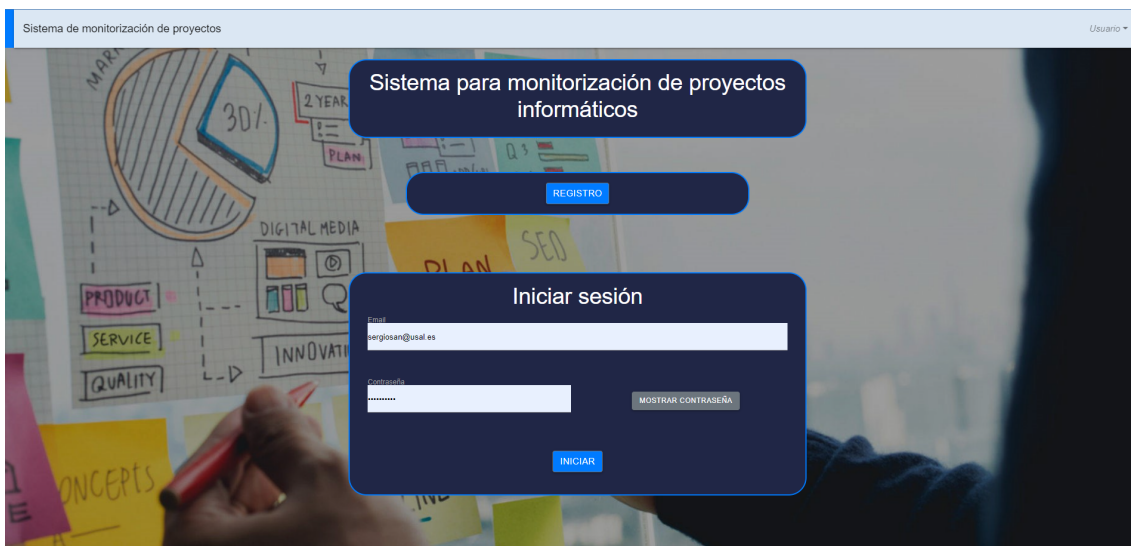


Ilustración 37. Pantalla de inicio

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos. Memoria final

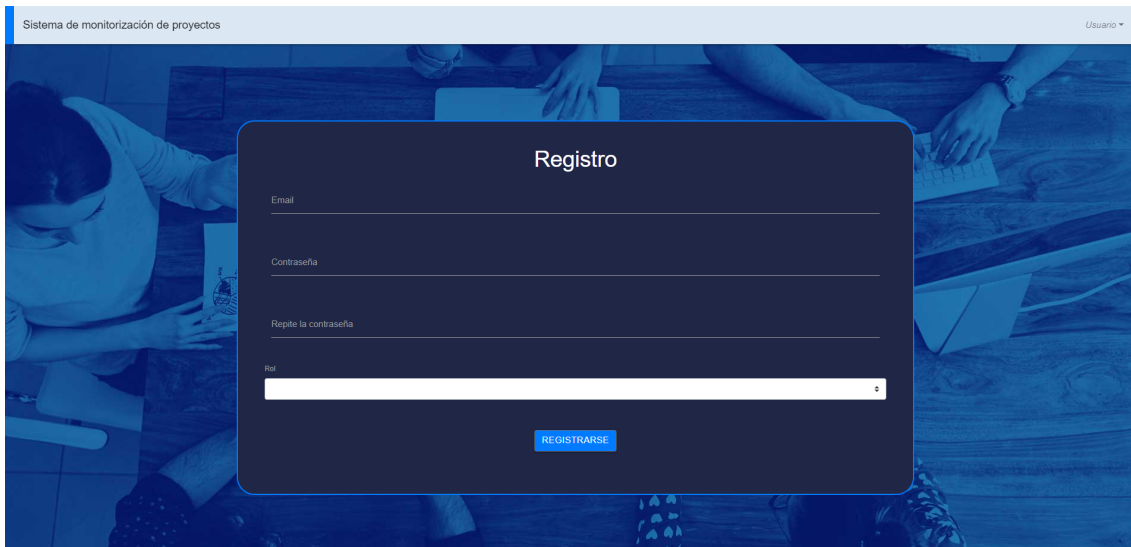


Ilustración 38. Pantalla de registro

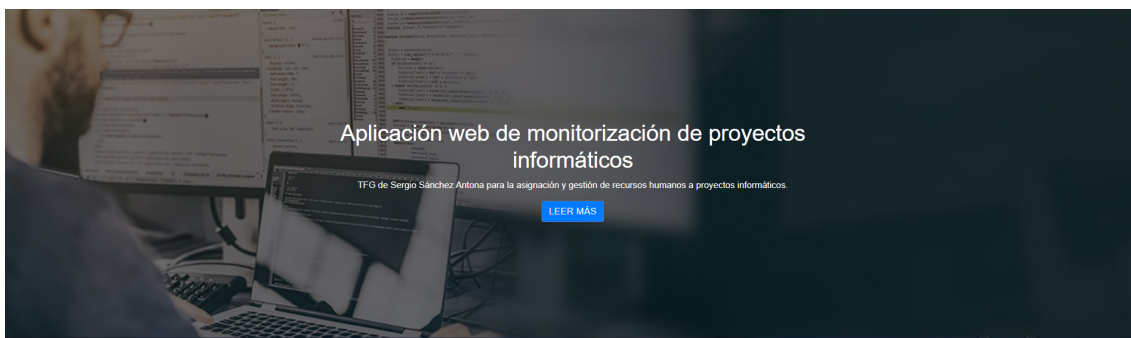


Ilustración 39. Pantalla inicial (parte 1)

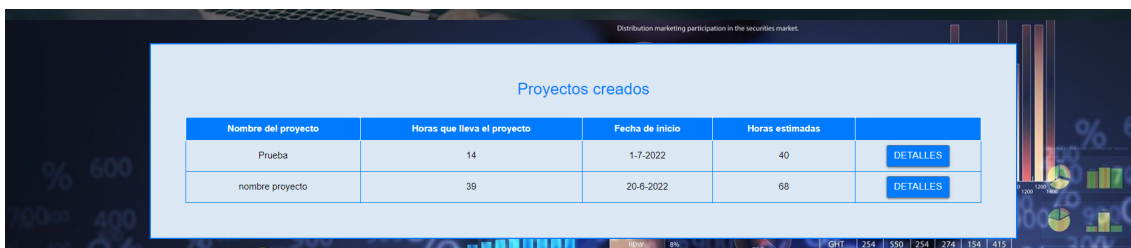


Ilustración 40. Pantalla inicial (parte 2)

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos. Memoria final

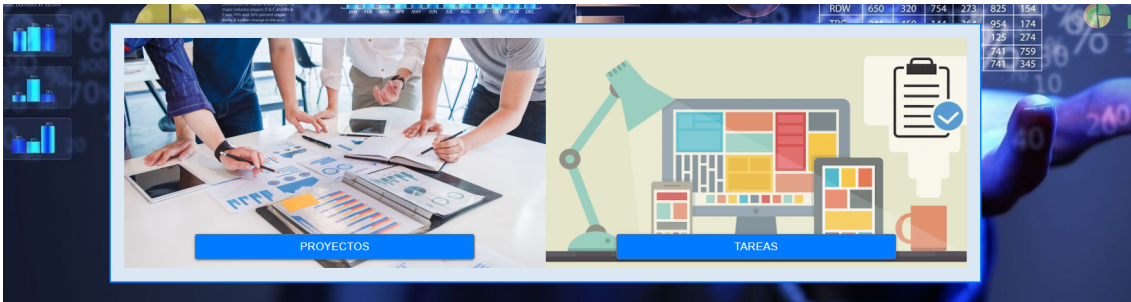


Ilustración 41. Pantalla inicial (parte 3)

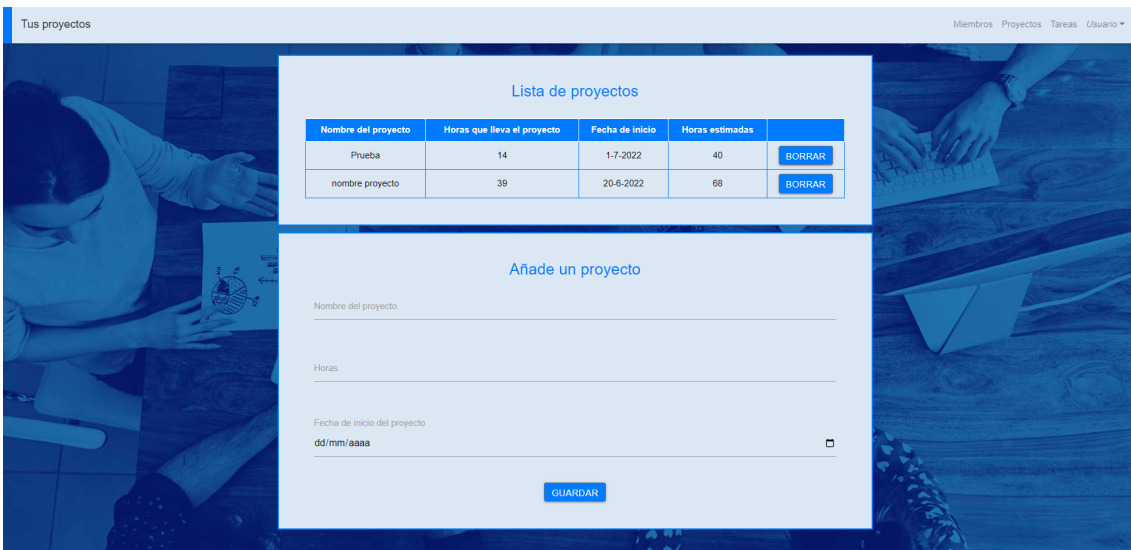


Ilustración 42. Pantalla de proyectos

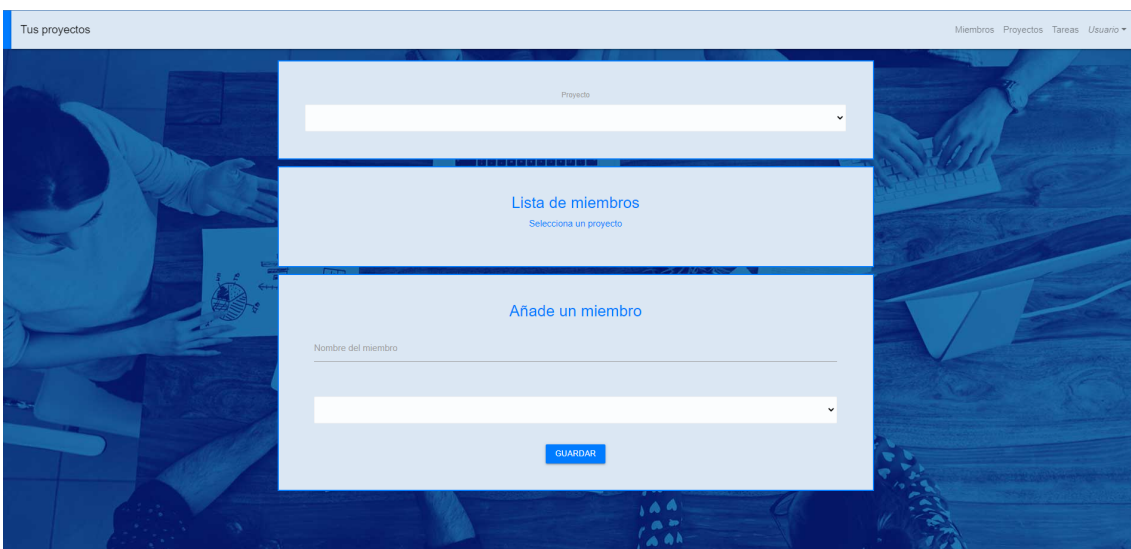


Ilustración 43. Pantalla de miembros

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos.

Memoria final

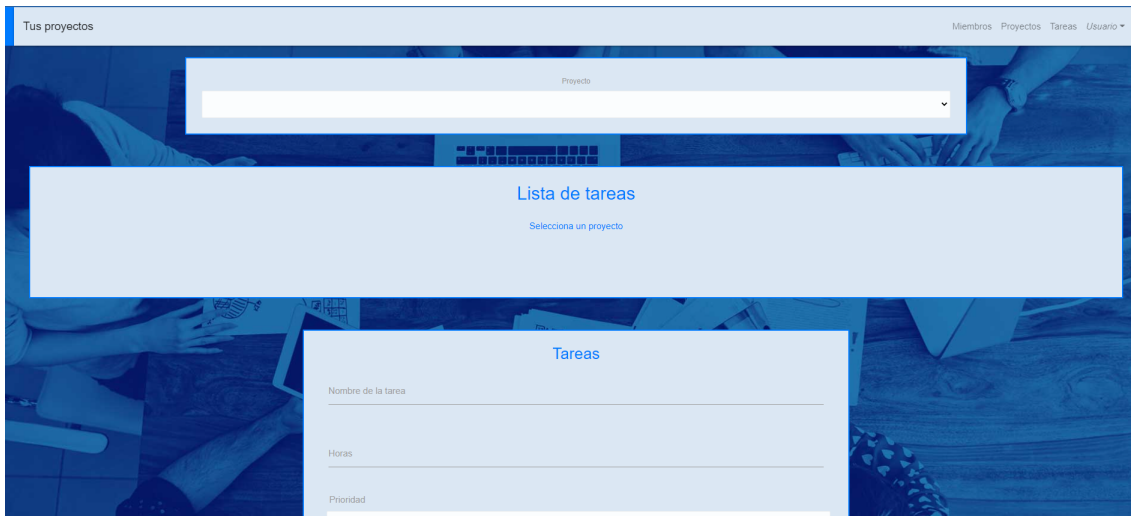


Ilustración 44. Pantalla de tareas (parte 1)

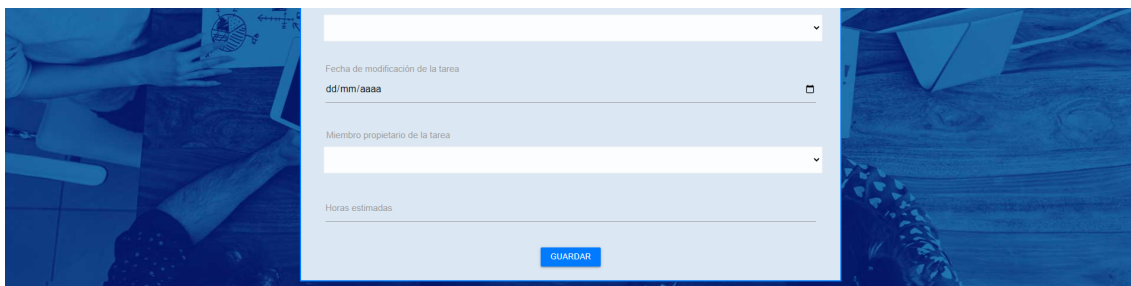


Ilustración 45. Pantalla de tareas (parte 2)



Ilustración 46. Pantalla de detalles (parte 1)

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos.
Memoria final



Ilustración 47. Pantalla de detalles (parte 2)

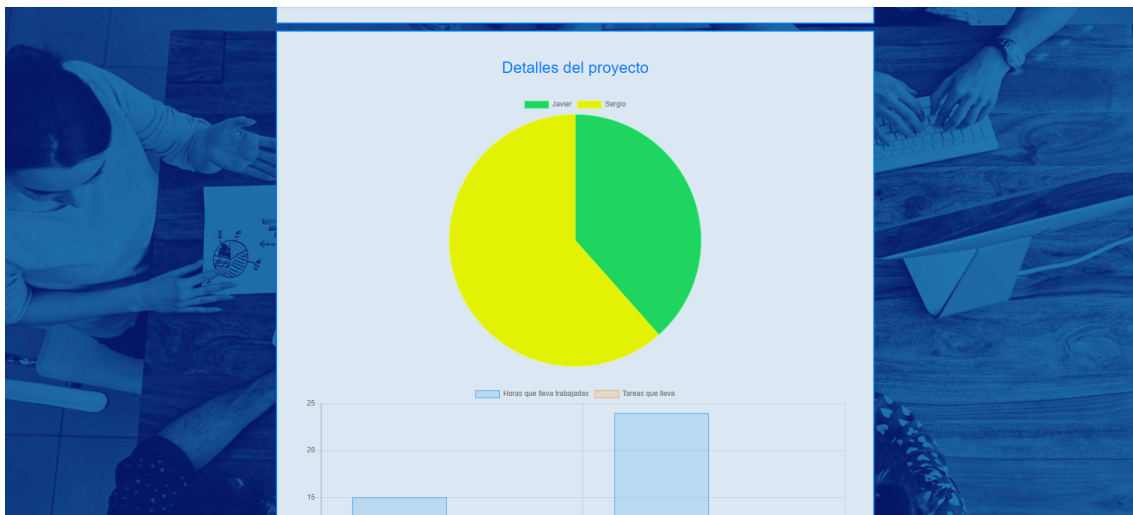


Ilustración 48. Pantalla de detalles (parte 3)

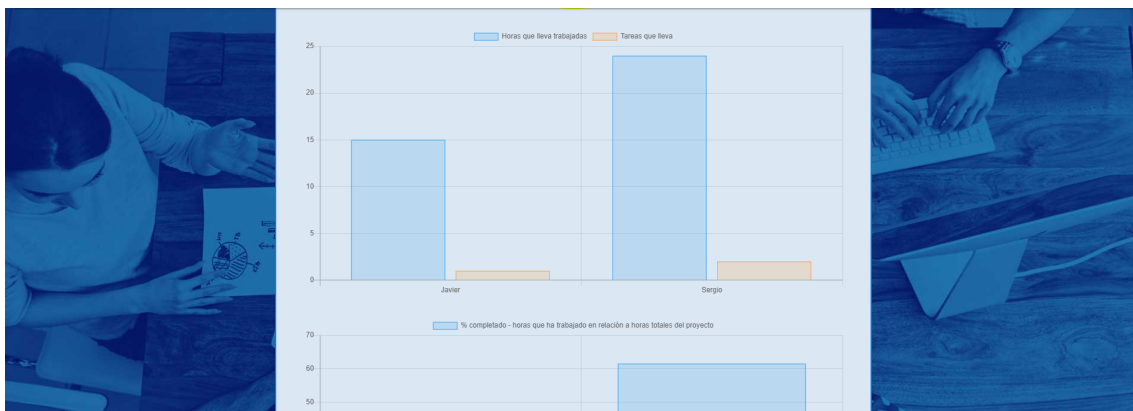


Ilustración 49. Pantalla de detalles (parte 4)

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos.
Memoria final

Cada una de estas vistas se detalla en el Anexo 5.

Dado que la aplicación web se ha diseñado para tener una interfaz responsive, estas mismas pantallas se ajustan a dispositivos de otras características. En las siguientes ilustraciones se muestran algunas imágenes obtenidas de las pantallas equivalentes pero adaptadas a un dispositivo móvil.



Ilustración 50. Pantalla móvil de página inicio

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos.
Memoria final

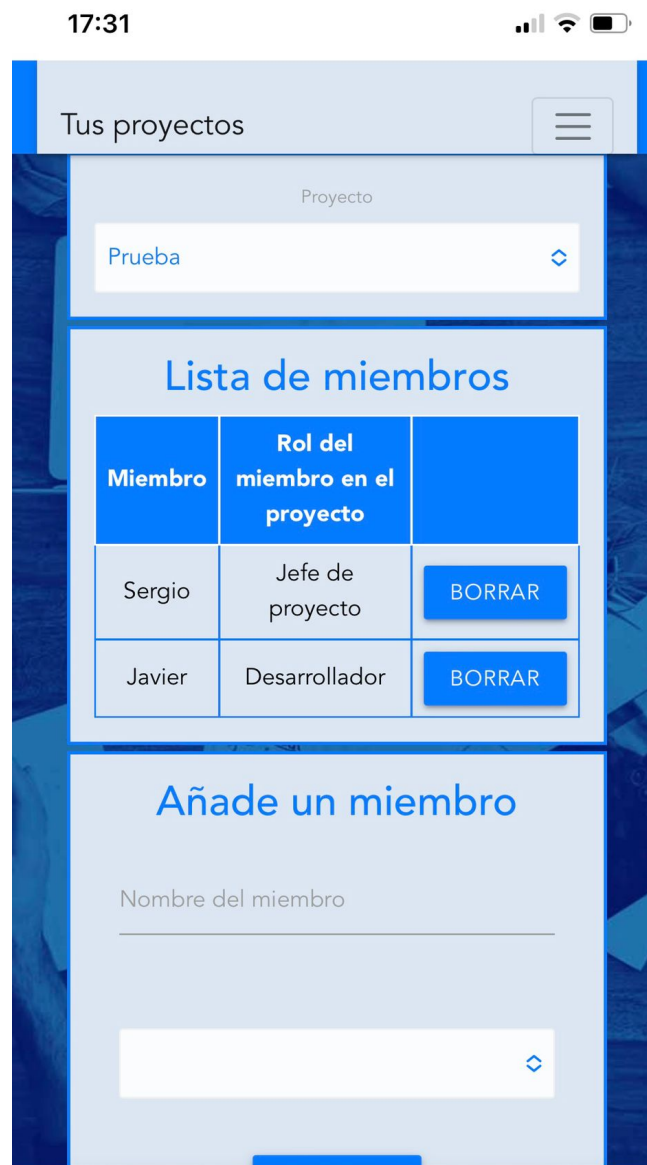


Ilustración 51. Pantalla móvil página miembros

Sistema para la monitorización de la asignación de recursos humanos a proyectos informáticos.
Memoria final



Ilustración 52. Pantalla móvil página inicial

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

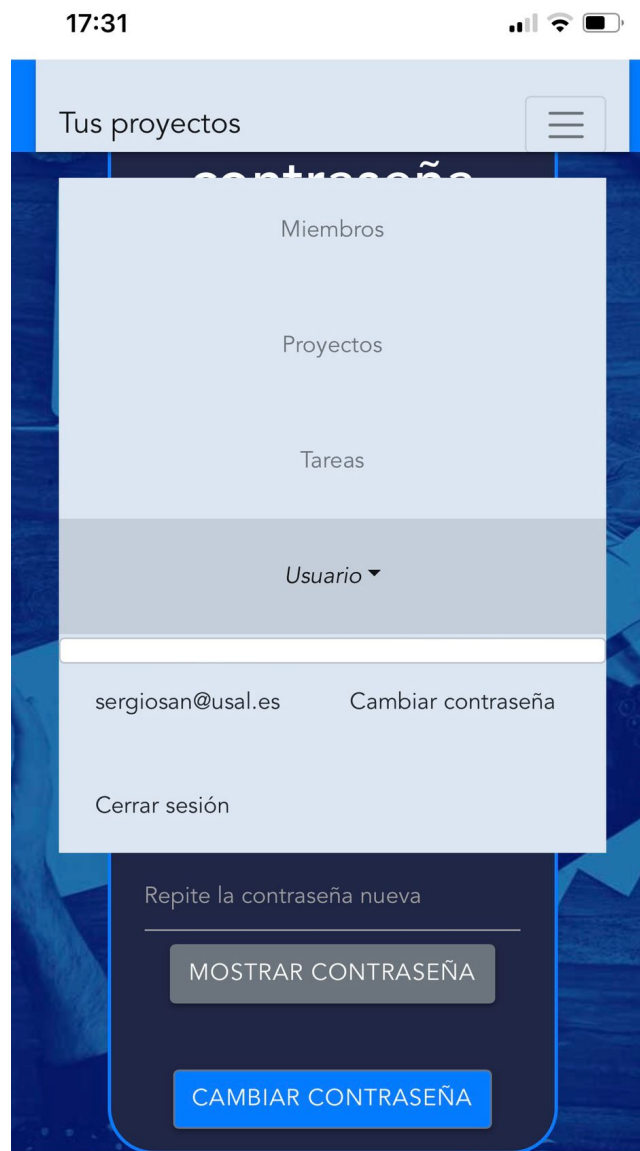


Ilustración 53. Pantalla móvil página cambio de contraseña y barra de navegación

6. Conclusiones y trabajo futuro

Tal y como se recoge en la introducción, el principal objetivo que aborda este proyecto es reducir la complejidad y la información innecesaria de otras herramientas para gestionar y monitorizar el trabajo que se realiza en los proyectos informáticos de una empresa, concretamente de Global Exchange.

Gracias al desarrollo de la aplicación planteada en este proyecto, con una interfaz de usuario simplificada y fácil de usar, es posible tener un control y seguimiento de los proyectos que satisface las necesidades expuestas inicialmente por la empresa. .

Por lo tanto, las principales contribuciones obtenidas tras la realización de este proyecto son las siguientes:

1. Una aplicación web responsive que atiende a las necesidades marcadas inicialmente. Concretamente, la aplicación permite el seguimiento de cualquier proyecto que se esté realizando por la empresa añadiendo tareas realizadas por los distintos miembros del proyecto y viendo los resultados en las múltiples tablas y gráficos de información que muestra el sistema.
2. Un método simplificado y rápido de gestionar y controlar los proyectos. Gracias a la sencillez de la aplicación, el tiempo necesario para la gestión de tareas, proyectos y tiempos, se reduce respecto a otros métodos de control, basados en herramientas más completas, pero que exigen más tiempo (ej. Jira).

Teniendo en cuenta estos resultados, a nuestro juicio queda demostrado en el transcurso del TFG que han sido alcanzados los objetivos planteados ya que:

Objetivo 1: Controlar y gestionar los desarrolladores autorizados a participan en los proyectos. Este objetivo se considera alcanzado ya que, tal y como queda reflejado en el apartado 4.4 en el objetivo gestión de usuarios, y explicado con más detalle en el Anexo 3 mediante una página de registro de los usuarios en la que se dan de alta en el sistema con un email, una contraseña (que se guarda encriptada)y un rol que indica si este usuario es Administrador y va a poder crear, eliminar o modificar proyectos, miembros y tareas o si es Usuario y va a poder ver el progreso que está teniendo en el proyecto junto con el de los demás miembros del equipo.

Objetivo 2: Mejorar el control y la gestión de proyectos. Este objetivo se considera alcanzado ya que, tal y como queda reflejado en el apartado 4.4. en los objetivos gestión de proyectos, miembros y tareas, se han implementado diferentes vistas que permiten crear proyectos que se irán actualizando a través de tareas que se implementen en este proyecto y así actualizando las horas de trabajo realizado que lleva el proyecto y las horas estimadas que se esperan realizar en total para tener un control sobre el esfuerzo que va a llevar este proyecto traducido en horas de trabajo de los distintos miembros en cada uno de los proyectos.

Objetivo 3: Tener un control preciso del tiempo de tareas y de proyectos. Este objetivo se considera alcanzado ya que, tal y como queda reflejado en el apartado 4.4. en el objetivo gestión de tareas, se ha implementado una página exclusivamente de tareas

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

asociadas a un proyecto, las cuales se pueden añadir, modificar y eliminar siendo capaz el Administrador de añadir horas trabajadas por los miembros en las distintas tareas teniendo total control del tiempo que lleva el proyecto en horas ya que estas tareas actualizan el proyecto mostrando las horas totales trabajadas y la estimación de las que se van a trabajar.

Objetivo 4: Disponer de un control del rendimiento de los miembros en las tareas de proyecto. Este objetivo se considera alcanzado ya que, tal y como queda reflejado en el apartado 4.4. en el objetivo 5 gestión de detalles, el sistema cuenta con una página para crear los miembros asociados a un proyecto que son los que realizan cada una de las tareas como se explica en el objetivo anterior. El sistema genera gráficas, tablas y barras de progreso con los datos que se han introducido sobre cada miembro por medio de las tareas mostrando exactamente lo que ha trabajado cada uno y el rendimiento que está teniendo en el transcurso de ese proyecto.

Objetivo 5: Simplificar las gestiones. Este objetivo se considera alcanzado ya que, tal y como queda reflejado a lo largo del apartado 4. Aspectos relevantes del proyecto y como se indica en el apartado 1. Introducción, el sistema es lo más simple posible, simplificando cada una de las vistas haciendo la aplicación muy intuitiva y eliminando los datos innecesarios que no nos sirven para los objetivos principales que son el control y gestión de los proyectos, el seguimiento del trabajo realizado y monitorización de los miembros.

Por consiguiente, se puede concluir que el objetivo principal, desarrollo de un software que permita controlar de forma sencilla las tareas y horas realizadas en los proyectos de una empresa siguiendo el rendimiento del trabajo de los miembros, ha sido alcanzado tras la realización de este proyecto.

Respecto a posibles líneas de trabajo futuro, hay cambios que se podrían realizar o nuevos objetivos que se podrían implementar con más tiempo y recursos.

En primer lugar, el registro e inicio de sesión se podría cambiar si este software fuera entregado a más empresas o fuera realmente subido a internet para el uso de más personas. Ya que el inicio de sesión actual está orientado únicamente para una empresa pudiendo ser controlado por alguna persona de esta, siendo capaz cada usuario de registrarse como Usuario o Administrados sin ninguna restricción.

En relación con este cambio cada usuario debería de tener su propia lista de proyectos creados o un grupo de estas gestionada por la empresa o el equipo de trabajo, pero al ser únicamente este proyecto para una empresa en concreto solo se dispone de un grupo de proyectos.

Se podría añadir también una vista para equipos de trabajo creando y gestionando el equipo con los miembros, controlando el rendimiento de todo el equipo y no solo de un trabajador en concreto.

Otro cambio interesante podría ser el seguimiento de las horas realizadas por los miembros en las tareas de forma más detallada, teniendo un seguimiento de qué día ha realizado las horas que se muestran en la página de tareas y en la de detalles y no solo el

Sistema para la monitorización de la asignación de recursos humanos a proyectos
informáticos.
Memoria final

último día que se ha trabajado en esta, por medio de un historial de tareas siguiendo los días con las horas de ese día.

Referencias

1. Global Exchange (s.f.). Sobre nosotros. Acceso: <https://www.globalexchange.es/>. Fecha de consulta: 16/07/2022
2. Filipova, O. (2016). *Learning Vue.js 2*. Packt Publishing Ltd.
3. Pérez, J. E. (2019). Introducción a JavaScript.
4. Luna, F., Millahual, C. P., & Iacono, M. (2018). *PROGRAMACION WEB Full Stack 13-PHP: Desarrollo frontend y backend-Curso visual y práctico* (Vol. 13). RedUsers.
5. Freeman, A. (2019). *Essential TypeScript*. Springer.
6. Juba, S., & Volkov, A. (2019). *Learning PostgreSQL 11: A beginner's guide to building high-performance PostgreSQL database solutions*. Packt Publishing Ltd.
7. Vue (s.f.). Acceso: <https://vuejs.org>. Fecha de consulta: 19/07/2022
8. HTML (s.f.). Acceso: <https://borjaarandavaquero.com/que-es/html/>. Fecha de consulta: 19/07/2022
9. JavaScript (s.f.). Acceso: <https://lenguajejs.com>. Fecha de consulta: 19/07/2022
10. TypeScript (s.f.). Acceso: <https://www.typescriptlang.org>. Fecha de consulta: 19/07/2022
11. PostgreSQL (s.f.). Acceso: <https://www.postgresql.org>. Fecha de consulta: 22/07/2022
12. NodeJS (s.f.). Acceso: <https://nodejs.org/es/>. Fecha de consulta: 22/07/2022
13. Visual Paradigm (s.f.). Acceso: <https://www.visual-paradigm.com>. Fecha de consulta: 22/07/2022
14. Microsoft Project (s.f.). Acceso: <https://www.microsoft.com/es-es/microsoft-365/project/project-management-software>. Fecha de consulta: 22/07/2022
15. Gómez, D., Flores, Á., & Ureta, R. (2018). Programación Node. JS. *Caribeña de Ciencias Sociales*, (mayo).
16. Peña, D. M., & Baquero, L. (2016). Extensión de la herramienta Visual Paradigm for UML para la evaluación y corrección de Diagramas de Casos de Uso. *Univ. las Ciencias Informaticas*.
17. Gido, J., & Clements, J. (2014). *Successful project management*.
18. Guillén, X. V., & Moldes, L. N. (2019). Arquitectura de aplicaciones web.
19. Rossberg, J. (2019). An overview of azure devops. *Agile Project Management with Azure DevOps*, 37-66.
20. Atlassian (s.f.). Trello. Acceso: <https://trello.com/> Fecha de consulta: 28/07/2022
21. Atlassian (s.f.). Jira Software. Acceso: <https://www.atlassian.com/es/software/jira> Fecha de consulta: 28/07/2022