



Universidad de Valladolid

E.T.S Ingeniería Informática

TRABAJO FIN DE GRADO

Grado en Ingeniería Informática
(Tecnologías de la información)

**DESARROLLO DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE LA
JORNADA HORARIA Y PERIODO DE VACACIONES EN UNA EMPRESA**

Autor:

D. Saúl Muñoz Galán

Tutores:

D. Miguel Ángel Laguna Serrano

D.^a Vanesa Martín Casado

Agradecimientos:

A mi hermano, Juan,
por motivarme y guiarme en estudiar el grado de Ingeniería informática.

A mis padres, Antonio y M.^a Belén,
por permitirme estudiar aquello que deseaba y apoyarme en todo momento.

A mis compañeros de curso,
por el gran compañerismo y el esfuerzo conjunto en tantos momentos.

A mi tutor, Miguel Ángel Laguna,
por apoyarme en la realización de este trabajo.

A la empresa Luce Innovative Technologies,
por ser la promotora de la idea y prestar todo aquello necesario para la realización del proyecto.

A mis compañeros de la empresa Luce Innovative Technologies,
por ayudarme con cualquier duda en torno al proyecto y por hacer mi estancia en la empresa tan satisfactoria tanto profesional como personalmente.

A todos ellos,
muchas gracias.

Resumen

El proyecto consiste en el desarrollo de una aplicación web capaz de gestionar la jornada horaria y los periodos de vacaciones de los distintos empleados existentes en una empresa. Dicha aplicación web servirá como modelo para la empresa Luce Innovative Technologies en su gestión de los aspectos indicados.

La aplicación web permitirá a los empleados tanto fichar en su jornada horaria como solicitar días de vacaciones que tengan disponibles para su posterior aprobación. Esta acción, también se gestionará mediante la aplicación por parte del encargado de dicha función dentro de la empresa.

La aplicación web será compatible con distintos navegadores y, además, será adaptada para poder ser utilizada a través de los navegadores de dispositivos móviles como pudiera ser un smartphone o una tablet.

Abstract

The project consists in the development of a web application capable of managing the working hours and vacation periods of the different employees existing in a company. This web application will serve as a model for the Luce Innovative Technologies company in its management of the indicated aspects.

The web application will allow employees to sign in their working hours as well as request vacation days that are available for later approval. This action will also be managed through the application by the person in charge of said function within the company.

The web application will be compatible with different browsers and, in addition, it will be adapted to be used through mobile device browsers such as a smartphone or tablet.

ÍNDICE

CAPÍTULO I: INTRODUCCIÓN	13
1.1. VISIÓN GENERAL	15
1.2. OBJETIVOS	16
1.3. METODOLOGÍA	16
1.4. CONTEXTO DE DESARROLLO	17
1.4.1. <i>Plataforma de desarrollo</i>	17
1.4.2. <i>Entorno de desarrollo</i>	17
CAPITULO II: PLANIFICACIÓN INICIAL.....	19
2.1. ORGANIZACIÓN DEL PROYECTO	21
2.1.1. <i>Participantes</i>	21
2.1.2. <i>Roles y responsabilidades</i>	22
2.2. RECURSOS DEL PROYECTO	23
2.3. OBJETIVOS DEL SISTEMA	25
2.4. PRODUCT BACKLOG	26
2.5. ESTIMACIÓN DE COSTES INICIAL.....	28
2.6. GESTIÓN DE RIESGOS.....	30
CAPÍTULO III: DESARROLLO.....	31
3.1. SPRINT 1	33
3.1.1. <i>Sprint planning</i>	33
3.1.2. <i>Sprint backlog</i>	33
3.1.3. <i>Desarrollo</i>	34
3.1.4. <i>Sprint review</i>	34
3.2. SPRINT 2	34
3.2.1. <i>Sprint planning</i>	34
3.2.2. <i>Sprint backlog</i>	35
3.2.3. <i>Desarrollo</i>	36
3.2.4. <i>Sprint review</i>	36
3.3. SPRINT 3	36
3.3.1. <i>Sprint planning</i>	36
3.3.2. <i>Sprint backlog</i>	37
3.3.3. <i>Desarrollo</i>	37
3.3.4. <i>Sprint review</i>	38
3.4. SPRINT 4	38
3.4.1. <i>Sprint planning</i>	38
3.4.2. <i>Sprint backlog</i>	38
3.4.3. <i>Desarrollo</i>	38
3.4.4. <i>Sprint review</i>	39
3.5. SPRINT 5	39
3.5.1. <i>Sprint planning</i>	39
3.5.2. <i>Sprint backlog</i>	39
3.5.3. <i>Desarrollo</i>	40
3.5.4. <i>Sprint review</i>	40
3.6. VISIÓN GLOBAL DE LOS SPRINTS.....	40

3.7.	ACTUALIZACIÓN DE COSTES	41
CAPÍTULO IV: MODELOS DE ANÁLISIS Y DISEÑO		43
3.2.	MODELO DE DOMINIO. DIAGRAMA DE CLASES	45
3.3.	MODELO-DISEÑO DE INTERFAZ DE USUARIO.....	46
3.4.	DIAGRAMA DE PAQUETES.	47
3.5.	MODELO DE INTERACCIÓN. DIAGRAMAS DE SECUENCIA.....	49
3.6.	MODELO DE ARQUITECTURA. DIAGRAMA DE DESPLIEGUE.....	50
CAPÍTULO V: IMPLEMENTACIÓN		51
5.1.	DIAGRAMA DE ARQUITECTURA/INFRAESTRUCTURA	53
5.2.	MODELO DE RELACIONAL.....	54
5.2.1.	<i>Descripción del modelo de relacional.....</i>	<i>54</i>
5.3.	CICLO DE VIDA DE LA APLICACIÓN	58
5.4.	DECISIONES DE IMPLEMENTACIÓN	59
5.4.1.	<i>Interfaz de web</i>	<i>59</i>
5.4.2.	<i>Spring Boot + Primefaces</i>	<i>59</i>
5.4.3.	<i>Base de datos Oracle.....</i>	<i>59</i>
5.4.4.	<i>LDAP</i>	<i>60</i>
5.4.5.	<i>Despliegue en servidor Glassfish</i>	<i>60</i>
CAPÍTULO VI: PRUEBAS		61
6.1.	PRUEBAS REALIZADAS EN LA APLICACIÓN.....	63
CAPÍTULO VII: CONCLUSIONES		73
7.1.	CONCLUSIONES	75
7.2.	FUTURAS MEJORAS.....	76
7.2.1.	<i>Nuevas HU tras despliegue en entorno de producción</i>	<i>76</i>
7.2.2.	<i>App móvil en vez de acceder por navegador.....</i>	<i>76</i>
7.2.3.	<i>Vista de los datos en una gráfica</i>	<i>76</i>
BIBLIOGRAFÍA		77
ANEXO I		83
I.	MANUAL DE USUARIO	85
1.	<i>Instalación.....</i>	<i>85</i>
2.	<i>Uso de la aplicación</i>	<i>90</i>
ANEXO II		99
I.	CONTENIDO DEL CD	101
1.	<i>TremProject.....</i>	<i>101</i>
2.	<i>Figuras</i>	<i>101</i>
3.	<i>Archivo de instalación.....</i>	<i>101</i>
4.	<i>Memoria</i>	<i>101</i>

ÍNDICE DE TABLAS

TABLA 1. PARTICIPANTE 1.....	21
TABLA 2. PARTICIPANTE 2.....	21
TABLA 3. PARTICIPANTE 3.....	21
TABLA 4. PARTICIPANTE 4.....	22
TABLA 5. ROLES Y RESPONSABILIDADES	22
TABLA 6. RECURSOS	23
TABLA 7. ORDENADOR DE DESARROLLO.....	23
TABLA 8. SMARTPHONE DE DESARROLLO.....	24
TABLA 9. TABLET DE DESARROLLO	24
TABLA 10. OBJETIVO 1	25
TABLA 11. OBJETIVO 2	25
TABLA 12. OBJETIVO 3	25
TABLA 13. OBJETIVO 4	26
TABLA 14. OBJETIVO 5	26
TABLA 15. HISTORIAS DE USUARIO	26
TABLA 16. ESTIMACIÓN COSTE HARDWARE	28
TABLA 17. ESTIMACIÓN COSTE SOFTWARE.....	29
TABLA 18. ESTIMACIÓN COSTE RECURSOS HUMANOS	29
TABLA 19. ESTIMACIÓN COSTE TOTAL	29
TABLA 20. RIESGO 1.....	30
TABLA 21. RIESGO 2.....	30
TABLA 22. SPRINT BACKLOG – SPRINT 1	33
TABLA 23. SPRINT BACKLOG – SPRINT 2	35
TABLA 24. SPRINT BACKLOG – SPRINT 3	37
TABLA 25. SPRINT BACKLOG – SPRINT 4	38
TABLA 26. SPRINT BACKLOG – SPRINT 5	39
TABLA 27. ETAPAS CICLO DE VIDA	58
TABLA 28. PRUEBA 1.....	63
TABLA 29. PRUEBA 2.....	63
TABLA 30. PRUEBA 3.....	63
TABLA 31. PRUEBA 4.....	64
TABLA 32. PRUEBA 5.....	64

TABLA 33. PRUEBA 6	64
TABLA 34. PRUEBA 7	65
TABLA 35. PRUEBA 8	65
TABLA 36. PRUEBA 9	65
TABLA 37. PRUEBA 10	66
TABLA 38. PRUEBA 11	66
TABLA 39. PRUEBA 12	66
TABLA 40. PRUEBA 13	67
TABLA 41. PRUEBA 14	67
TABLA 42. PRUEBA 15	67
TABLA 43. PRUEBA 16	67
TABLA 44. PRUEBA 17	68
TABLA 45. PRUEBA 18	68
TABLA 46. PRUEBA 19	68
TABLA 47. PRUEBA 20	69
TABLA 48. PRUEBA 21	69
TABLA 49. PRUEBA 22	69
TABLA 50. PRUEBA 23	70
TABLA 51. PRUEBA 24	70
TABLA 52. PRUEBA 25	70
TABLA 53. PRUEBA 26	71
TABLA 54. PRUEBA 27	71
TABLA 55. PRUEBA 28	71
TABLA 56. HU FUTURAS	76

ÍNDICE DE FIGURAS

FIGURA 1. DIAGRAMA DE CLASES	45
FIGURA 2. INTERFAZ USUARIO PC.....	46
FIGURA 3. INTERFAZ USUARIO SMARTPHONE	46
FIGURA 4. DIAGRAMA PAQUETES.....	47
FIGURA 5. DIAGRAMA SECUENCIA LOGIN.....	49
FIGURA 6. DIAGRAMA SECUENCIA FICHAJE	49
FIGURA 7. DIAGRAMA SECUENCIA LISTADO FICHAJES	50
FIGURA 8. DIAGRAMA DESPLIEGUE	50
FIGURA 9. DIAGRAMA DE ARQUITECTURA-INFRAESTRUCTURA.....	53
FIGURA 10. MODELO RELACIONAL	54
FIGURA 11. ENTIDAD ACCESOS	54
FIGURA 12. ENTIDAD EQUIPO	55
FIGURA 13. ENTIDAD FICHAJE.....	55
FIGURA 14. ENTIDAD MODULO PERMISO	55
FIGURA 15. ENTIDAD PERMISO.....	55
FIGURA 16. ENTIDAD PERSONA	56
FIGURA 17. ENTIDAD ROL	56
FIGURA 18. ENTIDAD TIPO ESTADO VACACIONES.....	57
FIGURA 19. ENTIDAD TIPO FICHAJE	57
FIGURA 20. ENTIDAD USUARIO.....	57
FIGURA 21. ENTIDAD VACACIONES.....	57
FIGURA 22. CICLO DE VIDA	58
FIGURA 23. INTERFAZ WEB SERVIDOR GLASSFISH	85
FIGURA 24. INTERFAZ APPLICATIONS SERVIDOR GLASSFISH.....	86
FIGURA 25. INTERFAZ DEPLOY SERVIDOR GLASSFISH	86
FIGURA 26. INTERFAZ SELECCIÓN ARTEFACTO SERVIDOR GLASSFISH	87
FIGURA 27. INTERFAZ APPLICATIONS DEPLOY SERVIDOR GLASSFISH.....	88
FIGURA 28. INTERFAZ ARTEFACTO SERVIDOR GLASSFISH	88
FIGURA 29. INTERFAZ WEB APPLICATIONS LINK SERVIDOR GLASSFISH	89
FIGURA 30. INTERFAZ LOGIN TRAS DESPLIEGUE	89
FIGURA 31. INTERFAZ LOGIN	90
FIGURA 32. INTERFAZ FICHAJE	91

FIGURA 33. INTERFAZ LISTADO FICHAJES	91
FIGURA 34. INTERFAZ INFORME HORAS REGISTRADAS	92
FIGURA 35. INTERFAZ CALENDARIO VACACIONES	92
FIGURA 36. INTERFAZ LISTADO VACACIONES	93
FIGURA 37. INTERFAZ CHANGELOG	93
FIGURA 38. INTERFAZ LOGIN (SMARTPHONE).....	94
FIGURA 39. INTERFAZ FICHAJE (SMARTPHONE)	94
FIGURA 40. INTERFAZ LISTADO FICHAJE (SMARTPHONE).....	95
FIGURA 41. INTERFAZ INFORME HORAS REGISTRADAS (SMARTPHONE)	95
FIGURA 42. INTERFAZ CALENDARIO VACACIONES (SMARTPHONE).....	96
FIGURA 43. INTERFAZ LISTADO VACACIONES (SMARTPHONE).....	96
FIGURA 44. INTERFAZ CHANGELOG (SMARTPHONE).....	97
FIGURA 45. INTERFAZ ADAPTACIÓN MENÚ (SMARTPHONE).....	97

CAPÍTULO I: INTRODUCCIÓN

1.1. Visión general

El control de la jornada horaria y los periodos de vacaciones dentro de una empresa son aspectos ya reconocidos ampliamente, pero es en los últimos años cuando estos aspectos han empezado a tomar mayor relevancia con el fin de evitar el incumplimiento de los contratos y el real decreto legislativo.

Con efectos al 12 de mayo de 2019 el Real Decreto obliga a las empresas a garantizar el registro de la jornada horaria, la cual debe realizarse mediante negociación colectiva, acuerdo de la empresa y decisión del empresario previa consulta con los representantes legales del trabajador, así como conservar los registros durante al menos 4 años y tenerlos disponibles para su acceso.

En este documento se describen las tareas realizadas para el desarrollo del proyecto, las cuales incluyen la introducción, la planificación inicial, el desarrollo, la implementación, las pruebas y las conclusiones.

En el capítulo I, correspondiente con la introducción del proyecto, nos centraremos en describir su ámbito más global, con la descripción de los objetivos, la metodología utilizada y el contexto de desarrollo del proyecto.

En el capítulo II, correspondiente con la planificación inicial, se desarrollarán los aspectos relacionados con la organización del proyecto, incluyendo tanto la descripción de los participantes como sus roles y responsabilidades dentro del mismo. Además, en este capítulo contendrá la descripción de los recursos del proyecto, las tareas a realizar para el desarrollo el mismo y los objetivos planteados. Para finalizar el capítulo, se describirán los principales riesgos y su gestión, y se describirá una estimación inicial de los costes del proyecto con el fin de dar una visión más económica de los aspectos relacionados con el mismo.

En el capítulo III, correspondiente con el desarrollo, se explicarán los aspectos relacionados con el desarrollo del proyecto a través de la metodología ya indicada, y se incluirán los aspectos más relevantes de cada una de las fases (sprints). En este apartado, también se incluirán aquellos aspectos relacionados con el dominio de la aplicación.

En el capítulo IV, se definen los modelos y diagramas desarrollados durante el proyecto.

En el capítulo V, correspondiente con la implementación, se detallarán las decisiones más importantes en cuanto a implementación que se tomaron a la hora de desarrollar el proyecto.

En el capítulo VI, correspondiente con las pruebas, se detallarán las distintas pruebas que se hicieron en la aplicación para verificar su correcto desarrollo.

Por último, en el capítulo VII, correspondiente con las conclusiones, se expondrán las distintas conclusiones extraídas tras el desarrollo del proyecto, y, además, se plantearán una serie de propuestas futuras para la aplicación con el fin de seguir el desarrollo comenzado en este proyecto.

Cabe destacar también, que este documento incluye dos anexos, el anexo I que contiene el manual de usuario de la aplicación y el anexo II que incluye la descripción del contenido almacenado en el soporte digital.

1.2. Objetivos

Este proyecto tiene dos objetivos principales: el primero es permitir a la empresa y a sus trabajadores realizar el fichaje de la jornada horaria de una manera sencilla y clara, pudiendo llevar un seguimiento de estos, y el segundo, permitir a todos ellos la solicitud y su posterior aprobación o denegación de los periodos de vacaciones aclarados en el contrato de cada trabajador.

La aplicación deberá ser capaz de registrar los distintos fichajes de la jornada y las solicitudes de periodos de vacaciones de los empleados, así como el seguimiento y las acciones de aprobación o denegación de estos por parte de los encargados de dicha tarea dentro de la empresa.

Por último, el sistema desarrollado deberá ser capaz de ejecutarse dentro de un servidor Glassfish 4 y conectarse con una base de datos Oracle en la que almacenará y de la que recuperará posteriormente los datos para su uso.

1.3. Metodología

La planificación de este proyecto se ha basado en una metodología ágil, puesto que actualmente son las más utilizadas dentro de las de las empresas para el desarrollo de sus proyectos. Las metodologías ágiles surgen como alternativa a las metodologías tradicionales como RUP o la metodología en cascada, puesto que dichas metodologías son demasiado burocráticas y por tanto rígidas para las actuales características del mercado.

En este caso se ha utilizado una variante de SCRUM adaptada a la empresa. Se trata de un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente (aunque en este caso el desarrollo haya sido por parte de una única persona, existen más personas involucradas en el proyecto, como, por ejemplo, Vanesa Martin Casado, Directora de Operaciones y en este caso, cliente de la aplicación, o Beatriz Del Pino, Team Leader del proyecto), y obtener el mejor resultado posible de proyectos, caracterizado por:

- Adoptar una estrategia de desarrollo incremental, en lugar de la planificación completa al comienzo del proyecto.
- Colaboración cercana con el cliente y predisposición a posibles cambios durante el desarrollo del proyecto.
- Solapar las diferentes fases del desarrollo, en lugar de realizar una tras otra en un ciclo secuencial o en cascada.

De acuerdo con la metodología elegida para la realización del proyecto, se dividirá el desarrollo del proyecto en varios sprints de 14 días, los cuales contendrán una serie de funcionalidades para llevar a cabo por orden de prioridad. Dichos sprints estarán compuestos por una serie de eventos o tareas:

- Product backlog: Lista total de las historias de usuarios que están en el alcance del proyecto ordenados por prioridad.

- **Sprint planning:** Reunión al comienzo de cada sprint donde se fijan los objetivos de este, se identifican los requisitos y las tareas a realizar para completar cada requisito.
- **Sprint backlog:** Trabajo acordado (obtenido del Product backlog) que se va a realizar durante la ejecución de dicho sprint.
- **Daily scrum:** Reunión acordada por los miembros del proyecto con el objetivo de ver los progresos del proyecto, los posibles obstáculos encontrados y el trabajo que se va a realizar en el momento.
- **Sprint review:** Reunión al final de cada sprint, supervisada por el cliente, donde se analiza la funcionalidad del producto conseguida durante el sprint.

Al final de cada sprint obtenemos un incremento en la funcionalidad del proyecto mediante la realización de las tareas o eventos incluidos en dicho sprint.

Finalmente, cabe destacar que en dicha metodología se utiliza una manera de estimar el trabajo a realizar mediante los denominados puntos de historia (PH). Dichos puntos de historia sirven para estimar el esfuerzo y complejidad que conllevaría realizar la tarea descrita. Aunque lo que realmente se estime es la complejidad, en la metodología utilizada se tiene en cuenta también el tiempo que conllevaría realizar la tarea, teniendo en cuenta que el responsable de esta será una sola persona con un perfil profesional de desarrollador junior, por lo que podemos decir que la relación entre puntos de historia y el tiempo en este proyecto es de aproximadamente 10 horas por cada punto de historia.

1.4. Contexto de desarrollo

1.4.1. Plataforma de desarrollo

El proyecto ha sido desarrollado para ser accesible desde un navegador web ya que será una aplicación que se usará desde distintos dispositivos, tanto ordenadores de sobremesa como dispositivos móviles tales como smartphones o tables.

La aplicación está diseñada para ejecutarse y ser visible desde cualquier tipo de pantalla y desde cualquier navegador, pero siendo recomendable el uso de Google Chrome puesto que se tomó como navegador de referencia para el desarrollo y pruebas de la aplicación.

1.4.2. Entorno de desarrollo

El IDE elegido para el desarrollo de la aplicación fue IntelliJ IDEA, entorno de desarrollo para aplicaciones basadas en Spring Boot dentro de la empresa. Fue elegido por delante de Eclipse o VisualStudio por su mayor potencia y opciones de desarrollo, además de por ser el más utilizado por mi parte.

Para la parte de planificación se utilizó Jira, una herramienta en línea para la administración de tareas de un proyecto, el seguimiento de errores e incidencias y para la gestión operativa de proyectos. Fue elegida por ser la herramienta de administración de la empresa para sus distintos proyectos.

En cuanto a la parte de análisis y diseño se utilizaron Astah y draw.io, un software online libre, para la representación de diagramas.

Para la parte de documentación se eligió Microsoft Word 2016 y para la presentación Microsoft PowerPoint 2016, ambos por su familiaridad y experiencia previa.

Otras herramientas que se utilizaron fueron DataGrip, para todos aquellos aspectos relacionados con la base de datos de la aplicación; Servidor Glassfish 4, para la ejecución de la aplicación; LDAP, un protocolo ligero de acceso a directorios hace referencia a un protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

Por último, para el control de versiones del código se usó Azure DevOps Server (GIT), un producto de Microsoft que proporciona control de versiones, informes, gestión de requisitos, gestión de proyectos, compilaciones automatizadas, gestión de laboratorio, pruebas y capacidades de gestión de versiones. Para la ejecución de los pipelines del ciclo de vida se utilizó Jenkins, para el control de versiones de los artefactos (.war) Nexus y para la calidad del código SonarQube.

CAPITULO II: PLANIFICACIÓN INICIAL

2.1. Organización del proyecto

El proyecto ha sido desarrollado por tan solo una persona, Saúl Muñoz Galán, con la ayuda de su tutor de proyecto Miguel Ángel Laguna Serrano y de la empresa Luce Innovative Technologies.

2.1.1. Participantes

A continuación, se definen los participantes que han hecho posible el desarrollo del proyecto (tablas 1, 2, 3 y 4).

TABLA 1. PARTICIPANTE 1

Participante-1	
Nombre	Saúl Muñoz Galán.
Organización	Alumno del Grado en Ingeniería Informática de la UVa. Luce Innovative Technologies.
Rol	Analista, diseñador y desarrollador.

TABLA 2. PARTICIPANTE 2

Participante-2	
Nombre	Miguel Ángel Laguna.
Organización	Departamento de Informática de la UVa.
Rol	Tutor del TFG.

TABLA 3. PARTICIPANTE 3

Participante-3	
Nombre	Vanesa Martín Casado.
Organización	Luce Innovative Technologies.
Rol	Ciente de la aplicación.

TABLA 4. PARTICIPANTE 4

Participante-4	
Nombre	Beatriz Del Pino.
Organización	Luce Innovative Technologies.
Rol	Team Leader del proyecto.

2.1.2. Roles y responsabilidades

A continuación, se detallan los roles y responsabilidades del proyecto (tabla 5).

TABLA 5. ROLES Y RESPONSABILIDADES

Rol	Persona encargada	Responsabilidades
Team Leader del proyecto	Beatriz Del Pino	Planificar las historias de usuario (HU) a realizar en cada Sprint.
Cliente	Vanesa Martín Casado	Definir los requisitos de la aplicación.
Analista	Saúl Muñoz Galán	Analizar los requisitos del cliente y definir las HU a realizar.
Diseñador	Saúl Muñoz Galán	Plantear el diseño a implementar una vez conocido el detalle de cada HU.
Desarrollador	Saúl Muñoz Galán	Implementar el diseño planteado para cada HU y documentar la aplicación para su uso.

2.2. Recursos del proyecto

En la siguiente tabla, se muestran los recursos utilizados durante el desarrollo del proyecto (tabla 6).

TABLA 6. RECURSOS

Recursos humanos	Recursos hardware	Recursos software
Saúl Muñoz Galán	Ordenador de desarrollo	IntelliJ, DataGrip, Microsoft Office 2016, GitBash, Jenkins, Sonar, Jira, Nexus, Asthah Google Chrome, Firefox
Saúl Muñoz Galán	Tablet de desarrollo (Honeywel EDA70-3)	Google Chrome Firefox
Saúl Muñoz Galán	Smartphone de desarrollo (Samsung s7)	Google Chrome Firefox

A continuación, se muestran las características más relevantes de los recursos hardware utilizados en el proyecto (tablas 7, 8 y 9).

TABLA 7. ORDENADOR DE DESARROLLO

Ordenador de desarrollo – Servidor Multipoint	
CPU	Intel® Core™ i7-7800X CPU @ 3.50GHz (12 CPUs).
RAM	64 GB.
Almacenamiento	315 GB.
SO	Windows Server 2016 Standard 64 bits (10.0, compilación 14393).
Pantalla	Tamaño: 27". Resolución: 1920 x 1080 píxeles.

TABLA 8. SMARTPHONE DE DESARROLLO

Smartphone de desarrollo – Samsung Galaxy s7	
CPU	Exynos 8890 Octa 2.3GHz / Snapdragon 820 2.15GHz.
RAM	8 GB.
Almacenamiento	32 GB.
SO	Android 6.0.1.
Pantalla	Tamaño: 5.1". Resolución: 1440 x 2560 píxeles.

TABLA 9. TABLET DE DESARROLLO

Tablet desarrollo – Honeywell EDA70	
CPU	Qualcomm MSM8916, 1.2GHz.
RAM	2 GB.
Almacenamiento	16 GB.
SO	Android 7.1.1.
Pantalla	Tamaño: 7". Resolución: 1280x720 píxeles.

2.3. Objetivos del sistema

A continuación, se describen cuáles son los principales objetivos del sistema a desarrollar y qué relevancia tiene cada uno en el proyecto (tablas 10, 11, 13, 13 y 14).

TABLA 10. OBJETIVO 1

OBJETIVO-1	
Nombre	Fichaje de la jornada horaria.
Autor	Saúl Muñoz Galán.
Descripción	El sistema deberá ser capaz de gestionar el fichaje de la jornada horaria de los trabajadores de la empresa.
Relevancia	Alta.

TABLA 11. OBJETIVO 2

OBJETIVO-2	
Nombre	Solicitud de periodos vacacionales.
Autores	Saúl Muñoz Galán.
Descripción	El sistema deberá permitir a los usuarios solicitar periodos vacacionales dentro de las políticas de la empresa.
Relevancia	Alta.

TABLA 12. OBJETIVO 3

OBJETIVO-3	
Nombre	Aprobación-Denegación de los periodos vacacionales.
Autores	Saúl Muñoz Galán.
Descripción	El sistema deberá permitir a los usuarios aprobar y denegar las solicitudes de periodos vacacionales siempre y cuando su rol disponga de dicha funcionalidad.
Relevancia	Alta.

TABLA 13. OBJETIVO 4

OBJETIVO-4	
Nombre	Consulta, modificación y borrado de fichajes horarios.
Autores	Saúl Muñoz Galán.
Descripción	El sistema deberá permitir a los usuarios la consulta, modificación y borrado de sus fichajes ya registrados en el sistema.
Relevancia	Alta.

TABLA 14. OBJETIVO 5

OBJETIVO-5	
Nombre	Consulta, modificación y borrado de las solicitudes de periodos vacacionales.
Autores	Saúl Muñoz Galán.
Descripción	El sistema deberá permitir a los usuarios la consulta, modificación y borrado de sus solicitudes de periodos vacacionales ya registrados en el sistema.
Relevancia	Alta.

2.4. Product backlog

El proyecto se desarrollará inicialmente en aproximadamente 11 puntos de historia (PH), cada uno de ellos se corresponde con aproximadamente 10h de trabajo. El alumno realizará una serie de PH en cada sprint hasta completar el total del proyecto. Las HU (historias de usuario), los requisitos y reglas de negocio asociadas incluidas inicialmente en el alcance del proyecto se detallan en la tabla 10.

TABLA 15. HISTORIAS DE USUARIO

HU	Nombre	Detalle	PH
1	Creación y configuración inicial del proyecto	Creación del proyecto de Spring Boot con la configuración inicial.	2
2	Adaptar la autenticación con LDAP y gestión de Roles	Adaptación del sistema de autenticación de la empresa (LDAP) a la aplicación.	1

		<p>Gestión de los roles de la aplicación mediante permisos sobre las acciones que puede realizar cada rol dentro de la aplicación. Los roles que se utilizarán en la aplicación web serán:</p> <ul style="list-style-type: none"> • ADMINISTRADOR: Aquellos usuarios • ADMINISTRACION: Aquellos empleados relacionados con las tareas de administración de la empresa. • RRHH: Aquellos empleados relacionados con las tareas de recursos humanos de la empresa. • POR DEFECTO (empleado): Aquellos empleados de la empresa que no se identifican en ningunos de los otros roles, por ejemplo, los desarrolladores. • DIRECCION: Aquellos empleados relacionados con las tareas de dirección de la empresa. • BUSINESS MANAGER: Aquellos empleados relacionados con las tareas de venta de productos de la empresa. • TEAM LEADER: Aquellos empleados relacionados con las tareas de organización de los equipos de desarrollo de la empresa. 	
3	Creación-Modificación-Consulta fichaje horario	<p>Creación de la lógica y pantallas web del alta, modificación, consulta y eliminación de los fichajes horarios del día (persona, tipo, fecha de fichaje, fecha de comunicación, observaciones).</p> <ul style="list-style-type: none"> • No se pueden realizar dos fichajes en la misma hora (mínimo un minuto después). • No se pueden realizar un fichaje en una fecha y hora anterior a la del último fichaje. • No se pueden realizar dos fichajes del mismo tipo consecutivamente. • El usuario debe tener asociada una persona (pueden existir varios usuarios con la misma persona asociada) • Cada usuario solo podrá ver los fichajes asociados a su persona. 	2
4	Creación-Modificación-Consulta vacaciones	<p>Creación de la lógica y pantallas web del alta, modificación, consulta y eliminación de las solicitudes de periodos vacacionales (persona, fecha de inicio, fecha de fin, estado, observaciones, razón).</p> <ul style="list-style-type: none"> • Los periodos no pueden comenzar o terminar en fin de semana. Los fines de semana no se contarán como días de vacaciones. • No pueden existir varias solicitudes solapadas para una misma persona a menos que las otras solicitudes fueran rechazadas. • La fecha de inicio ha de ser posterior a la fecha de fin. 	2

		<ul style="list-style-type: none"> Solo se pueden eliminar las solicitudes de periodos vacacionales asociados a la persona del usuario que estén en estado de "Solicitud". Cada usuario solo podrá ver las solicitudes asociados a su persona, excepto los usuarios con rol Business Manager o Team Leader que podrán ver todas las solicitudes. 	
5	Aprobación-Denegación de vacaciones	<p>Creación de la lógica y pantallas web de la aprobación y denegación de los periodos vacacionales.</p> <ul style="list-style-type: none"> Solo los usuarios con rol Business Manager o Team Leader podrán realizar estas funciones. 	1
6	Ciclo de vida	Inclusión de la aplicación dentro del ciclo de vida de las aplicaciones desarrolladas en la empresa (Jenkins, Sonar, Nexus...).	1
7	Diseño gráfico y estilos de la aplicación	<p>Diseño de la aplicación web siguiendo los estilos y convenciones de la empresa.</p> <p>Debe ser responsive, es decir, la interfaz de usuario debe adaptarse a la pantalla del dispositivo en el que se utiliza.</p>	1
8	Despliegue en entorno VALIDACIÓN	Despliegue de la aplicación en el entorno de VALIDACION para realizar las primeras pruebas del cliente.	1

2.5. Estimación de costes inicial

A continuación, se muestra una estimación de los costes totales del proyecto (tablas 16, 17, 18 y 19). En estos costes se incluyen tanto las herramientas hardware y software, como el coste de las horas de trabajo de los recursos humanos. Se ha de tener en cuenta que tanto los recursos hardware como software no solo se utilizan para este proyecto, por lo que el coste de estos será prorrateado en función del uso (4 meses aproximadamente) y de su amortización en 4 años, es decir, su coste se multiplicara por 4/48.

TABLA 16. ESTIMACIÓN COSTE HARDWARE

Coste hardware			
Hardware	Coste Unitario (Euros)	Cantidad	Coste Total (Euros)
Ordenador desarrollo	1500	1	1500
Samsung s7	180	1	180
Tablet Honeywell EDA70	600	1	600
Total hardware			2280

TABLA 17. ESTIMACIÓN COSTE SOFTWARE

Coste de software			
Software	Coste Unitario (Euros)	Cantidad	Coste Total (Euros)
Intelij	630	1	630
Microsoft Office 2016	69	1	69
DataGrip	549	51	549
Jenkins	0 (open source)	1	0
Sonar	0 (open source)	1	0
Jira	0 (open source)	1	0
Astah	224	1	224
Nexus	120	1	120
Total software			1592

TABLA 18. ESTIMACIÓN COSTE RECURSOS HUMANOS

Coste recursos humanos			
Persona	Coste Unitario Hora (Euros)	Cantidad (horas)	Coste Total (Euros)
Saul Muñoz Galán	18	110 (10h/PH)	1980
Total recursos humanos			1980

TABLA 19. ESTIMACIÓN COSTE TOTAL

Coste total del proyecto	
Apartado	Coste Total (Euros)
Hardware	$2280 * 4 / 48 = 190$
Software	$1592 * 4 / 48 = 132.66$
Recursos humanos	1980
Total proyecto (Euros)	
	2302.66

2.6. Gestión de riesgos

Dada la metodología utilizada, y a diferencia de como ocurre en metodologías más antiguas como pudiera ser Proceso Unificado, el análisis y gestión de los riesgos no se realiza de manera completa y exhaustiva en la planificación inicial del proyecto, sino que estos riesgos se asumen y es en las estimaciones de las historias donde se tienen en cuenta de manera indirecta, de tal manera que en caso de existir algún riesgo se estima al alza la historia por si el riesgo se convirtiera en una realidad durante el desarrollo. A continuación, se muestran 2 riesgos que sí se tuvieron en cuenta al inicio del proyecto, dada su relevancia (tablas 20 y 21).

TABLA 20. RIESGO 1

Formulario de gestión de riesgos		
Identificador: 1	Título: Fallos de hardware	Fecha:
Categoría: Riesgos del proyecto	Fase: Todo el proyecto	Probabilidad: Baja
Consecuencia: Retraso en la finalización del proyecto y aumento de los costes para sustituir o reparar el componente dañado.		
Valoración de riesgos		
Descripción del riesgo	Los fallos en el hardware pueden afectar a la disponibilidad.	
Contexto del riesgo	El riesgo está presente durante todo el proyecto.	
Análisis del riesgo	Puede producir un retraso en las tareas y aumentar el coste del proyecto.	
Planificación de riesgos		
Estrategia	Aceptación del riesgo	
Plan de acción del riesgo	El coste de la redundancia del hardware es muy caro. En caso de que ocurra tan solo se sustituirá el hardware por dispositivos similares o se utilizarán emuladores, si es posible.	

TABLA 21. RIESGO 2

Formulario de gestión de riesgos		
Identificador: 2	Título: Carga de trabajo elevada	Fecha:
Categoría: Riesgos del proyecto	Fase: Todo el proyecto	Probabilidad: Baja
Consecuencia: retraso en la finalización del proyecto y aumento de los costes por el incremento de tiempo de uso de los recursos para el proyecto		
Valoración de riesgos		
Descripción del riesgo	Durante los sprints la carga de trabajo de otros proyectos más relevantes podría ser más elevada y el desarrollo se trasladaría a siguientes sprints.	
Contexto del riesgo	El riesgo está presente durante todo el proyecto.	
Análisis del riesgo	Puede producir un retraso en las tareas y aumentar el coste del proyecto.	
Planificación de riesgos		
Estrategia	Aceptación del riesgo	
Plan de acción del riesgo	Si el riesgo llegara a materializarse se trasladaría el desarrollo del proyecto a sprints posteriores	

CAPÍTULO III: DESARROLLO

3.1. Sprint 1

3.1.1. Sprint planning

Tras la primera reunión a cerca del proyecto, la cual se alargó más de lo normal puesto que se plantearon distintos escenarios, objetivos y tareas para este primer sprint, se llegó a la conclusión de que este primer sprint únicamente tendría que servir como forma de inicio del proyecto, sin llegar todavía a plantear cuestiones acerca de cómo desarrollar los aspectos relacionados con los objetivos del proyecto (el fichaje y las vacaciones), de manera que se realizaran únicamente las tareas de inicio y configuración inicial del proyecto y la adaptación del sistema de autenticación de la empresa (LDAP) para todas sus herramientas, junto con la gestión de los distintos roles dentro de la empresa.

Por otro lado, este sprint planning sirvió como modelo para el resto, puesto que se decidió repartir la carga de trabajo del proyecto en sprints equilibrados en cuanto a PH. Además, en este primer sprint planning se analizaron distintas opciones de desarrollo de manera conjunta con algunos miembros de la empresa con el fin de asesorar al alumno sobre cuales podrían ser las tecnologías más adaptadas al proyecto, pero dejando que él mismo estudiase los distintos escenarios y decidiera cual podría adaptarse mejor al desarrollo del proyecto.

3.1.2. Sprint backlog

TABLA 22. SPRINT BACKLOG – SPRINT 1

HU	Nombre	Detalle	PH
1	Creación y configuración inicial del proyecto	Creación del proyecto de Spring Boot con la configuración inicial.	2
2	Adaptar la autenticación con LDAP y gestión de Roles	Adaptación del sistema de autenticación de la empresa (LDAP) a la aplicación. Gestión de los roles de la aplicación mediante permisos sobre las acciones que puede realizar cada rol dentro de la aplicación. Los roles que se utilizarán en la aplicación web serán: <ul style="list-style-type: none">• ADMINISTRADOR: Aquellos usuarios• ADMINISTRACION: Aquellos empleados relacionados con las tareas de administración de la empresa.• RRHH: Aquellos empleados relacionados con las tareas de recursos humanos de la empresa.• POR DEFECTO (empleado): Aquellos empleados de la empresa que no se identifican en ningunos de los otros roles, por ejemplo, los desarrolladores.	1

		<ul style="list-style-type: none"> • DIRECCION: Aquellos empleados relacionados con las tareas de dirección de la empresa. • BUSINESS MANAGER: Aquellos empleados relacionados con las tareas de venta de productos de la empresa. • TEAM LEADER: Aquellos empleados relacionados con las tareas de organización de los equipos de desarrollo de la empresa. 	
--	--	---	--

3.1.3. Desarrollo

Este sprint, como ya se ha mencionado antes, serviría como base para el resto, y es por ello por lo que el desarrollo se centró más en los aspectos relacionados con el estudio de los escenarios y tecnologías que se utilizarían, que en el desarrollo en sí. Durante este sprint se configuró el proyecto mediante la utilización de la herramienta web proporcionada por Spring Boot, Spring Initializr, la cual crea un proyecto base de spring boot con la configuración inicial que nosotros deseamos. Una vez obtenido dicho proyecto base, se configuraron los aspectos relacionados con la comunicación de acceso a la base de datos del proyecto y la comunicación con el LDAP de la empresa para la autenticación de los usuarios. En este primer sprint se llevó a cabo también la configuración y la creación de la pantalla de login de la aplicación.

3.1.4. Sprint review

En esta primera reunión tras el desarrollo de un sprint, se presentó al cliente de la aplicación las funcionalidades obtenidas tras el desarrollo de este, las cuales no se centraban en aspectos visuales de la aplicación, sino en aspectos más técnicos y de configuración. El cliente quedó conforme con la funcionalidad conseguida, y el resultado y valoración del sprint fue muy positivo, tanto por parte del cliente como de todos los implicados en el proyecto puesto que se había obtenido una configuración inicial del proyecto sobre la que poder comenzar a desarrollar los objetivos más relevantes de manera segura.

3.2. Sprint 2

3.2.1. Sprint planning

En la reunión correspondiente al sprint planning del sprint 2, se llegó a la conclusión de que sería este el sprint más relevante a la hora de conseguir los objetivos marcados, puesto que en él se incluirían las HU más relacionadas con los principales objetivos del sistema. Es por ello, por lo que en este sprint se decidió incluir mayor número de PH a realizar, coincidiendo también con un periodo en el que yo mismo me encontraba más libre en cuanto a otros proyectos dentro de la empresa. El principal objetivo de este sprint era lograr tener la funcionalidad básica relacionada con los fichajes y las vacaciones, y como se puede observar en el sprint backlog del sprint se incluyeron las HU relacionadas con ello.

3.2.2. Sprint backlog

TABLA 23. SPRINT BACKLOG – SPRINT 2

HU	Nombre	Detalle	PH
3	Creación-Modificación-Consulta fichaje horario	<p>Creación de la lógica y pantallas web del alta, modificación, consulta y eliminación de los fichajes horarios del día (persona, tipo, fecha de fichaje, fecha de comunicación, observaciones).</p> <ul style="list-style-type: none"> • No se pueden realizar dos fichajes en la misma hora (mínimo un minuto después). • No se pueden realizar un fichaje en una fecha y hora anterior a la del último fichaje. • No se pueden realizar dos fichajes del mismo tipo consecutivamente. • El usuario debe tener asociada una persona (pueden existir varios usuarios con la misma persona asociada) • Cada usuario solo podrá ver los fichajes asociados a su persona. 	2
4	Creación-Modificación-Consulta vacaciones	<p>Creación de la lógica y pantallas web del alta, modificación, consulta y eliminación de las solicitudes de periodos vacacionales (persona, fecha de inicio, fecha de fin, estado, observaciones, razón).</p> <ul style="list-style-type: none"> • Los periodos no pueden comenzar o terminar en fin de semana. Los fines de semana no se contarán como días de vacaciones. • No pueden existir varias solicitudes solapadas para una misma persona a menos que las otras solicitudes fueran rechazadas. • La fecha de inicio ha de ser posterior a la fecha de fin. • Solo se pueden eliminar las solicitudes de periodos vacacionales asociados a la persona del usuario que estén en estado de "Solicitud". • Cada usuario solo podrá ver las solicitudes asociados a su persona, excepto los usuarios con rol Business Manager o Team Leader que podrán ver todas las solicitudes. 	2
5	Aprobación-Denegación de vacaciones	<p>Creación de la lógica y pantallas web de la aprobación y denegación de los periodos vacacionales.</p> <ul style="list-style-type: none"> • Solo los usuarios con rol Business Manager o Team Leader podrán realizar estas funciones. 	1

3.2.3. Desarrollo

Durante el desarrollo de este sprint se incluyeron dos fases, puesto que los objetivos del sistema eran muy grandes y las funcionalidades a desarrollar contenían la parte más relevante de la aplicación, pero se podían diferenciar en dos claramente, fichaje y vacaciones.

En primer lugar, se estudió la parte relacionada con los fichajes puesto que era de mayor prioridad en cuanto al sprint y al proyecto. El sistema debía de ser capaz de guardar ciertos datos acerca del fichaje, pero a su vez, debía ser lo más sencillo y cómodo posible para el usuario, por lo que se decidieron aspectos como que el sistema completara la fecha del fichaje con la fecha actual de manera automática y que el usuario únicamente tuviera que elegir el tipo de fichaje (entrada/salida). A su vez, se decidió utilizar mensajes de aviso al usuario cuando sus fichajes no fueran coherentes, como por ejemplo fichar dos entradas seguidas, sin una salida intermedia.

Por otro lado, se estudió la parte relacionada con las vacaciones, la cual incluía una primera parte de alta, modificación, consulta y borrado de los periodos solicitados y una segunda en la que el rol correspondiente dentro de la empresa pudiera cambiar el estado de la solicitud. En este estudio y desarrollo de los aspectos mencionados se decidieron cosas como que, aparte de tener una forma de consulta similar a los fichajes, existiera otra compuesta por una pantalla con un calendario, en la que además de poder realizar estas mismas operaciones se pudiera solicitar los periodos eligiendo las fechas de manera más visual e intuitiva.

3.2.4. Sprint review

Al final de este sprint, y como en todos los llevados a cabo en el proyecto, se mostró la funcionalidad obtenida al cliente. En este sprint, y a diferencia del anterior, las funcionalidades si eran más visuales, y es por ello, por lo que el cliente, como él mismo afirmó durante dicha reunión, observó un gran avance en el proyecto. Los objetivos del sprint eran muy ambiciosos y tras el desarrollo de este, y como se comentó en dicha reunión, se habían cumplido todos ellos de manera correcta, por lo que el sprint fue un gran éxito y el proyecto había tomado un camino y forma correctos.

3.3. Sprint 3

3.3.1. Sprint planning

Teniendo en cuenta los resultados del sprint anterior y que el proyecto ya disponía de una estructura y contenido bastante amplio en cuanto a los objetivos del sistema, se decidió en esta reunión que este tercer sprint debía tener como objetivo que la aplicación estuviera desplegada en un entorno en el que el cliente pudiera probar por sí mismo las funcionalidades desarrolladas, puesto que hasta el momento tan solo se habían mostrado sin que este pudiera realizar sus pruebas oportunas. Por ello, en este sprint se decidió incluir el resto de HU definidas en el alcance inicial, las cuales incluían la integración en el ciclo de vida, el diseño de los estilos de las páginas web y el despliegue en el entorno de validación.

3.3.2. Sprint backlog

TABLA 24. SPRINT BACKLOG – SPRINT 3

HU	Nombre	Detalle	PH
6	Ciclo de vida	Inclusión de la aplicación dentro del ciclo de vida de las aplicaciones desarrolladas en la empresa (Jenkins, Sonar, Nexus...).	1
7	Diseño gráfico y estilos de la aplicación	Diseño de la aplicación web siguiendo los estilos y convenciones de la empresa. Debe ser responsive.	1
8	Despliegue en entorno VALIDACIÓN	Despliegue de la aplicación en el entorno de VALIDACION para realizar las primeras pruebas del cliente.	1

3.3.3. Desarrollo

El desarrollo de este sprint fue el más complicado de todos los llevados a cabo durante el desarrollo del proyecto porque supuso un constante estudio y toma de decisiones en periodos de tiempo reducidos.

En cuanto a la inclusión del proyecto en el ciclo de vida, el cual se describe en el siguiente capítulo, supuso un estudio en herramientas como Jenkins, SonarQube y Nexus, herramientas muy potentes a la hora de obtener a partir de un código fuente un artefacto que poder desplegar en un servidor.

Referido a los estilos de la aplicación, el desarrollo fue más sencillo puesto que la empresa contaba con una serie de guías y estilos que se aplican dentro de los proyectos internos de la empresa. Sin embargo, como la aplicación debía de ser responsive, fue en este aspecto donde se encontraron obstáculos a la hora de llevar a cabo el desarrollo.

Por último, se llegó al momento más importante del sprint, en el cual se debía desplegar lo desarrollado hasta el momento para cumplir el objetivo del sprint. Fue en este punto donde se encontraron los mayores obstáculos de todo el proyecto, puesto que aparentemente el artefacto (.war) generado y desplegado en el servidor no había encontrado problema para ello, pero, sin embargo, el servidor no era capaz de mostrar la aplicación web. Tras horas de esfuerzo y trabajo y con la ayuda del departamento de sistemas de la empresa, se encontró la causa de este incidente, la cual se debía a una configuración dentro del servidor y no a un fallo en el artefacto generado. Esto supuso un empleo de mayor esfuerzo del estimado para la realización de la HU, llegando a ser casi del doble (18h de desarrollo).

3.3.4. Sprint review

Una vez concluido el sprint, se realizó la habitual reunión en la que se presentó al cliente todo lo desarrollado, la inclusión de la aplicación en el ciclo de vida y el diseño de las páginas web de la aplicación. Esta reunión fue diferente, puesto que fue donde el cliente probó la funcionalidad, y a partir de los resultados obtenidos en las mismas, planteó una serie de cambios y nuevas funcionalidades a desarrollar con el fin de hacer una aplicación más completa y atractiva para el usuario. A pesar de esto, el cliente quedó muy contento acerca del objetivo del sprint, puesto que se había cumplido a pesar de los inconvenientes y obstáculos encontrados en el desarrollo de este.

3.4. Sprint 4

3.4.1. Sprint planning

Como se ha mencionado, tras las pruebas del cliente en el entorno de validación, surgieron una serie de nuevas HU a desarrollar. Según se decidió en esta reunión, estas nuevas HU se desarrollarían en dos nuevos sprints. En cuanto al primer sprint de los dos, se decidió incluir las nuevas funcionalidades que, según se había estimado, requirieran un mayor esfuerzo, con el objetivo de que tras la conclusión de este sprint el núcleo más importante de la aplicación quedara completo, y en el segundo, incluir las HU más sencillas con el fin de dar por terminado el proyecto.

3.4.2. Sprint backlog

TABLA 25. SPRINT BACKLOG – SPRINT 4

HU	Nombre	Detalle	PH
9	[Informe horas] Informe horas fichadas por día	Creación de la lógica y pantallas web para visualizar el computo de horas fichadas por día o por periodo de tiempo de un usuario.	2
10	[Fichaje] Grabar la ubicación fichaje	En el fichaje, grabar la ubicación de la persona en el momento del fichaje.	1

3.4.3. Desarrollo

Al igual que en el sprint anterior, este sprint tuvo algunos obstáculos a la hora de su desarrollo. No fue en la realización de del nuevo informe, la cual fue desarrollada sin problema, sino en el grabado del fichaje por parte de la aplicación. Este problema se debió principalmente a que la aplicación se visualizaba en un navegador web, el cual no tiene los mismos sistemas de ubicación que pudieran utilizarse si se tratara de una aplicación Android, por ejemplo. Esto hizo que en esa HU se empleara mucho más esfuerzo (16h de desarrollo) del estimado.

3.4.4. Sprint review

Tras la realización del sprint, se mostró al cliente la nueva funcionalidad que se había desarrollado, siendo esta el informe de las horas fichadas por un usuario y la grabación de la ubicación del fichaje. El cliente quedó conforme con la nueva funcionalidad y aclaró en esta reunión que le gustaría probar en el siguiente sprint review la aplicación al completo.

3.5. Sprint 5

3.5.1. Sprint planning

Esta reunión fue más rápida de lo habitual ya que en ella se decidió que se desarrollaría el resto de las funcionalidades y modificaciones de la aplicación propuestas por parte del cliente para dar por finalizado el proyecto, hasta el punto de desplegarlo en producción para su uso.

3.5.2. Sprint backlog

TABLA 26. SPRINT BACKLOG – SPRINT 5

HU	Nombre	Detalle	PH
11	[Informe horas] Filtro de persona para los roles ADMINISTRADOR y RRHH	Creación del filtro de persona en la pantalla de informe de horas para los roles de ADMINISTRADOR y RHH.	0.5
12	[Ver fichajes] Filtro de persona para los roles ADMINISTRADOR y RRHH	Creación del filtro de persona en la pantalla de ver fichajes para los roles de ADMINISTRADOR y RHH.	0.5
13	Pantalla error amigable	Creación de las pantallas de error tras los distintos fallos posibles de la aplicación.	0.5
14	[Fichaje] Redirección tras fichaje	Redirección al listado de los fichajes del usuario tras la grabación de un fichaje.	0.25
15	Cambio componente fecha fichaje	Cambiar el componente de fecha de fichaje de la hora para permitir introducir la fecha de manera manual con un formato predefinido (dd/mm/aaaa hh:mm).	0.5
16	Rediseño estilos	Rediseño del menú y la cabecera de la aplicación.	0.5
17	Incluir el changelog de versión en el pie	Añadir un enlace en footer de la aplicación que redirija a una pantalla en la que se muestre los cambios que se incluyen en cada versión de la aplicación.	0.5
18	Despliegue en entorno de VALIDACIÓN	Despliegue de la aplicación en el entorno de VALIDACIÓN.	0.5

3.5.3. Desarrollo

Aunque a simple vista parecía un sprint largo y complicado puesto que en él que había muchas HU, el desarrollo de este sprint fue muy sencillo ya que se trataba de pequeñas modificaciones y mejoras en la aplicación con el fin de hacer la aplicación web más sencilla y agradable para el usuario.

3.5.4. Sprint review

Tras la finalización de este último sprint, se presentó al cliente la nueva funcionalidad desde el entorno de validación, según se había aclarado y planteado en reuniones anteriores. Fue en esta reunión en la cual el cliente probó tanto que la funcionalidad ya existente siguiera cumpliendo con los requisitos, como que la nueva funcionalidad y modificaciones en la aplicación tuvieran un correcto funcionamiento.

Una vez realizada las pruebas sobre toda la aplicación, y tras la aprobación por parte del cliente, el cual quedó muy satisfecho con lo conseguido tanto en este último sprint como en todos los sprint que compusieron el total del desarrollo del proyecto, se acordó desplegar el artefacto generado en el entorno de producción, con el fin de poder comenzar a usar la aplicación por parte de los usuarios.

Para finalizar, en esta reunión se trataron temas como próximas funcionalidades y mejoras de la aplicación, las cuales ya no recaerían sobre el desarrollo del proyecto actual, sino se plantearían a futuro, y las cuales describiremos en el capítulo VI.

3.6. Visión global de los sprints

Tras la finalización de los sprints, podemos mencionar varios aspectos de carácter global entre los cuales cabe destacar la correcta planificación de las historias de usuario, el desarrollo gradual partiendo de una base segura y la continua revisión y conversación con el cliente.

En primer lugar, cabe destacar la correcta planificación de las historias de usuario, puesto que como se puede observar a lo largo del proyecto y salvo en alguna excepción, las historias se realizaron dentro del esfuerzo y tiempo estimado para las mismas. El total de las historias de usuario estaba planificado en 17,75 puntos de historia (177,5 horas aproximadamente) y el desarrollo de estas se llevó a cabo en 203,5 horas.

Por otro lado, el desarrollo gradual partiendo de una base segura permitió que durante el desarrollo del proyecto no surgieran prácticamente ningún tipo de contratiempo desde el punto de vista de configuración y funcionalidad, lo cual permitió que el proyecto se desarrollara de manera gradual sin ningún tipo de problema.

Por último, la continua revisión y conversación con el cliente fue uno de los grandes apoyos para el desarrollo del proyecto, puesto que en estas reuniones y revisiones es donde se aclararon ciertas dudas en cuanto al desarrollo de negocio y donde se podía verificar las funcionalidades obtenidas a lo largo de los sprint.

3.7. Actualización de costes

Tras la ejecución del proyecto y teniendo en cuenta las distintas peticiones del cliente durante el desarrollo de este, hemos de actualizar los costes del proyecto, los cuales se ven claramente incrementados en la parte de recursos humanos. Estos costes aumentados tienen que ver con las distintas reuniones que se realizaron para la planificación de los sprints, los correspondientes sprint reviews, las historias de usuario propuestas tras el sprint review del sprint 3 y la parte de documentación del proyecto (diagramas, manuales, memoria...). Este coste adicional supone un número total de horas empleadas en la realización del proyecto de aproximadamente 345,5, las cuales están divididas en 203,5 horas en la parte de desarrollo de las historias de usuario, 22 horas en reuniones (sprint planning y sprint reviews), 29 horas en realización de documentación, manuales de usuario e instalación y 91 horas en la realización de la memoria presente. Esto conlleva un coste total de los recursos humanos de:

- Coste recursos humanos = $345,5h * 18 \text{ euros/h} = 6219 \text{ euros}$.

Por lo que el coste total del proyecto sería:

- Coste total = 190 euros hardware + 132,66 euros software + 6219 euros recursos humanos.

Coste total = 6541,66 euros.

CAPÍTULO IV: MODELOS DE ANÁLISIS Y DISEÑO

3.2. Modelo de dominio. Diagrama de clases

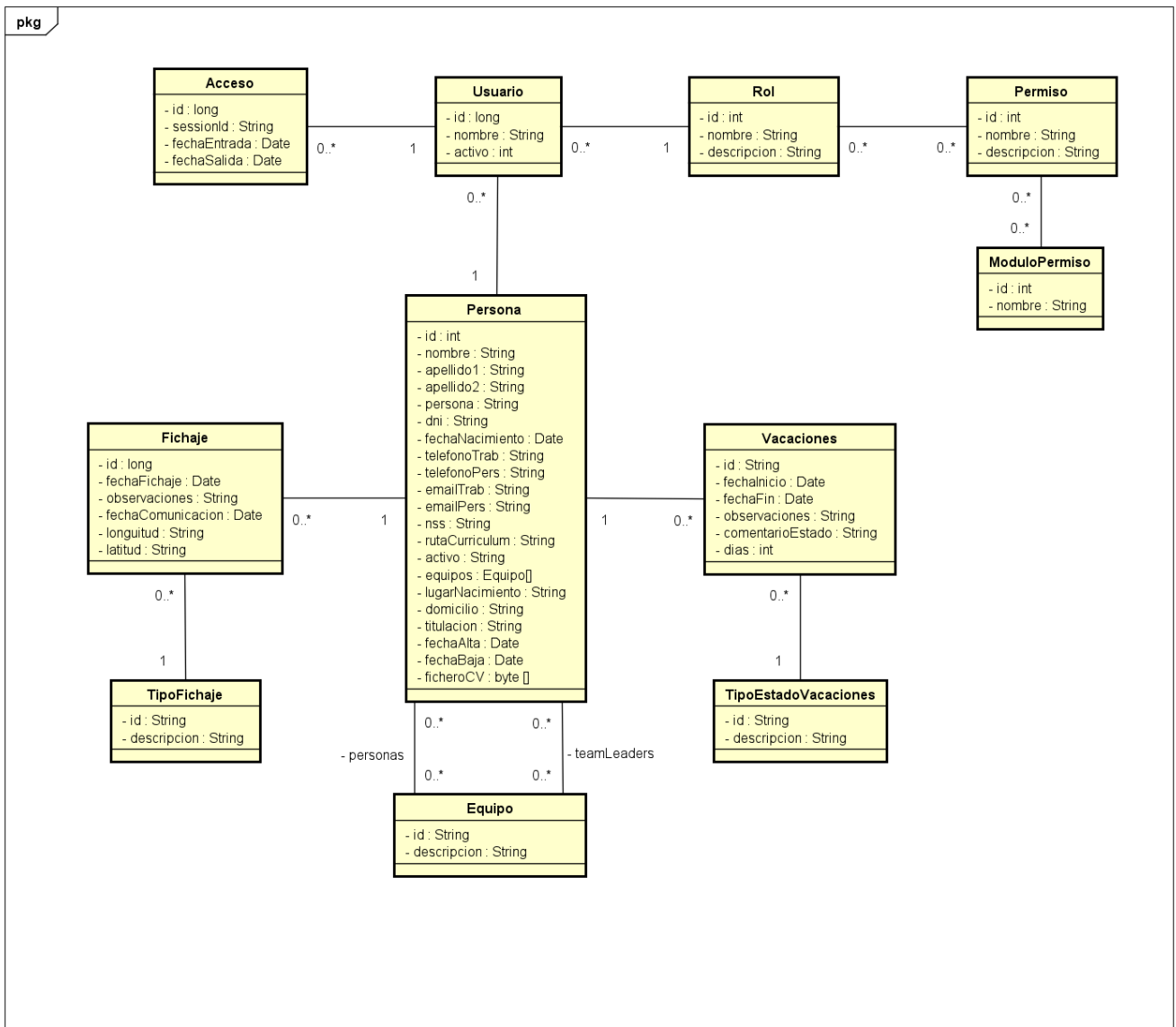


FIGURA 1. DIAGRAMA DE CLASES

3.3. Modelo-Diseño de interfaz de usuario.

Como se indicó en la planificación inicial y en los requisitos y objetivos del proyecto, la interfaz de usuario debía de ser sencilla, agradable, y, sobre todo, responsive, es decir, debe ser capaz de adaptarse a la pantalla desde la que se está visualizando, de manera que la interfaz siga proporcionando toda la funcionalidad de una manera visible y adaptada en caso de pantallas más pequeñas como pudiera ser un smartphone o una table. A continuación, se muestran ejemplos de pantallas de distintos dispositivos.

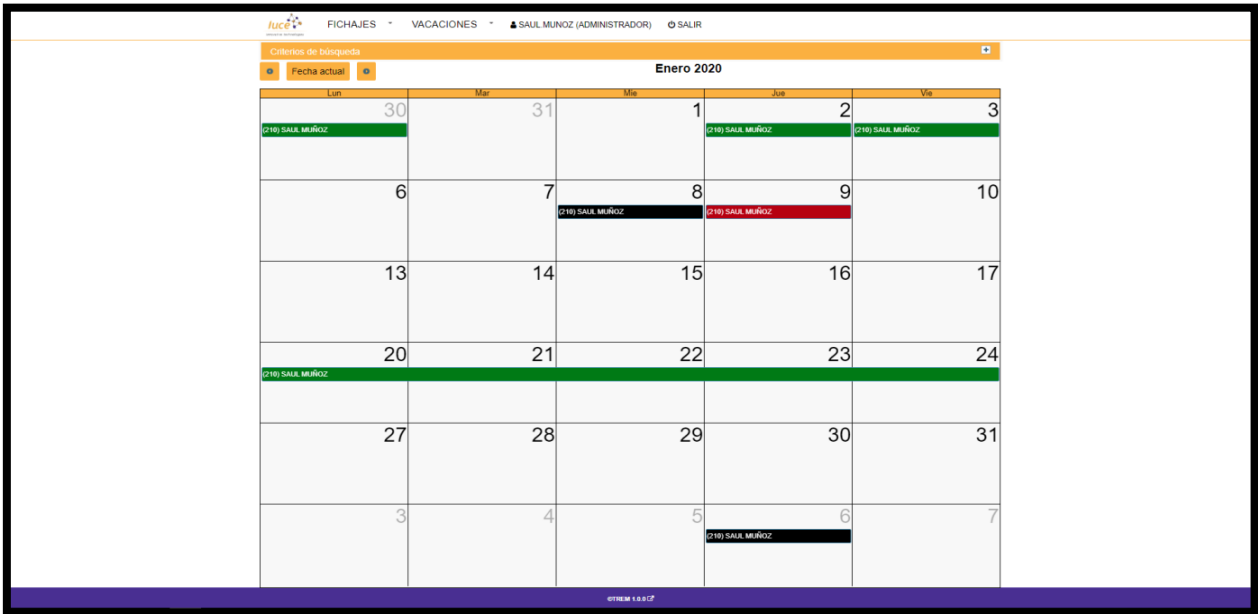


FIGURA 2. INTERFAZ USUARIO PC

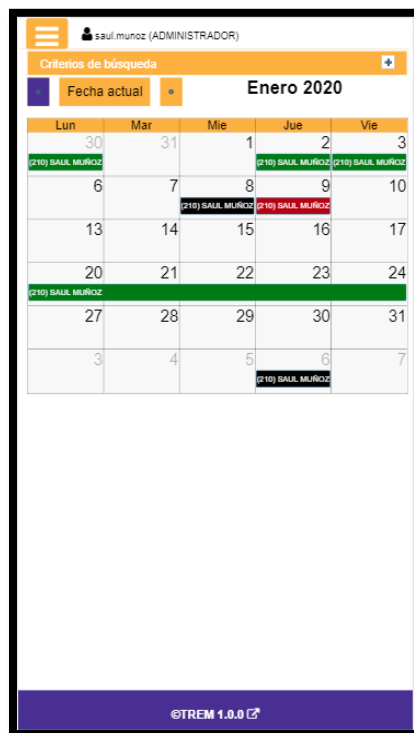


FIGURA 3. INTERFAZ USUARIO SMARTPHONE

3.4. Diagrama de paquetes.

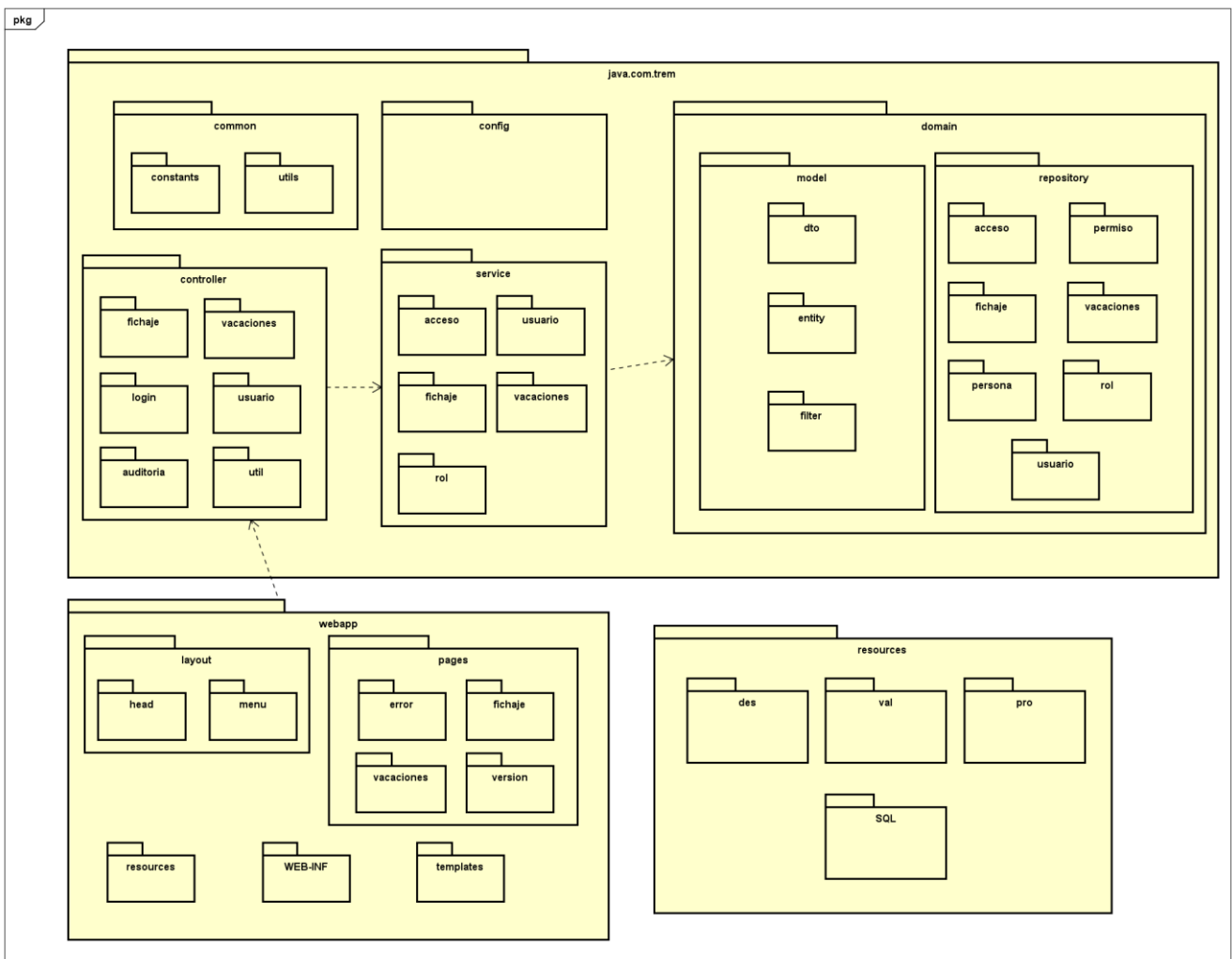


FIGURA 4. DIAGRAMA PAQUETES

Como se puede ver en el diagrama anterior, la aplicación se divide en 3 paquetes principales, que a su vez se dividen en varios paquetes:

- **java.com.trem:** contiene aquellos paquetes relacionados con el desarrollo de las clases java necesarias para el proyecto.
 - **common:** contiene las clases java de constantes y utilidades del proyecto.
 - **config:** contiene las clases java de configuración del proyecto como pueden ser CustomAuthenticationSuccessHandler, encargada de definir el acceso cuando se ha validado a través del LDAP ya mencionado; CustomAuthenticationFailureHandler, encargada de definir las acciones cuando el acceso no ha sido validado a través del LDAP; DefaultViewConfig, encargada de definir la pagina por defecto; ErrorPageRegister, encargada de definir las acciones a realizar cuando ha ocurrido un error en la aplicación.; JSFConfig, encargada de definir los principales parámetros de configuración de las interfaces; MessagesConfig, encargada de definir los principales

parámetros de configuración de los mensajes mostrados al usuario en la interfaz web; PersistenceSigpConfig, encargada de definir el acceso a la base de datos; y WebSecurityConfig, encargada de definir la seguridad de acceso a la aplicación web;

- **controller:** contiene las clases java que se encargan de recoger las peticiones http generadas a partir de la interfaz y gestionadas por Spring Boot a través de sus distintos tipos como @Component o @Controller. Por cuestiones de nomenclatura interna de la empresa, estas clases java contienen al final de su nombre “Bean”. Además, también contiene las clases java de conversión de tipos.
- **service:** contiene las clases java encargadas de definir los servicios de lógica de negocio de la aplicación.
- **domain:** contiene las clases java que definen los modelos de dominio de la aplicación y el acceso a sus representaciones relacionales en base de datos.
 - **model:** contiene las clases java que definen los modelos de dominio de la aplicación. Se divide a su vez en:
 - **dto:** contiene las clases java que definen las distintas representaciones del modelo relacional.
 - **entity:** contiene las clases java que definen el modelo relacional.
 - **filter:** contiene las clases java que definen las representaciones de los filtros de la aplicación.
 - **repository:** contiene las clases java que definen el acceso a la base de datos y esta dividida a su vez en las distintas representaciones del modelo relacional.
- **resources:** contiene aquellos paquetes relacionados con los archivos de propiedades de los entornos de desarrollo (des), validación (val) y producción (pro) y los archivos de cambios necesarios en la base de datos para la actualización de la aplicación.
 - **des:** contiene los archivos de configuración del entorno de desarrollo.
 - **val:** contiene los archivos de configuración del entorno de validación.
 - **pro:** contiene los archivos de configuración del entorno de producción.
 - **SQL:** contiene los archivos de cambios necesarios en la base de datos para la actualización de la aplicación.
- **webapp:**
 - **layout:** contiene los principales archivos (.xhtml) que componen la estructura de la aplicación web.

- **pages:** contiene los archivos (.xhtml) que componen las distintas interfaces de interacción de la aplicación web.
- **resources:** contiene los archivos de imágenes, ejecutables (.js) y estilos (.css) de la aplicación web.
- **WEB-INF:** contiene los archivos de configuración de la aplicación web.
- **templates:** contiene archivos (.xhtml) que son comunes en varias interfaces y sirven como plantillas.

3.5. Modelo de interacción. Diagramas de secuencia

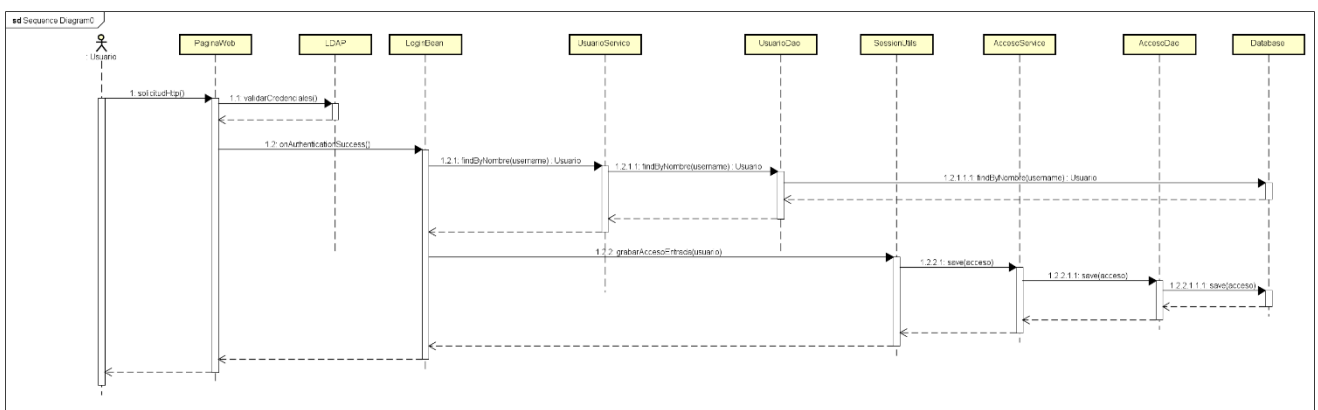


FIGURA 5. DIAGRAMA SECUENCIA LOGIN

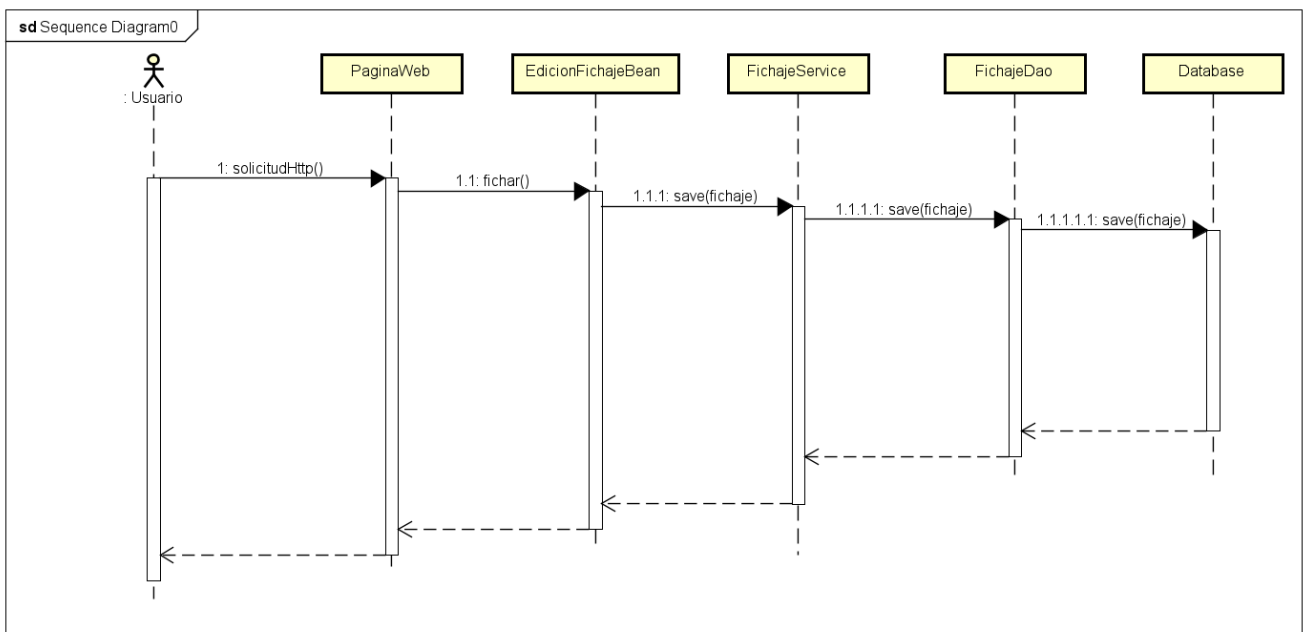


FIGURA 6. DIAGRAMA SECUENCIA FICHAJE

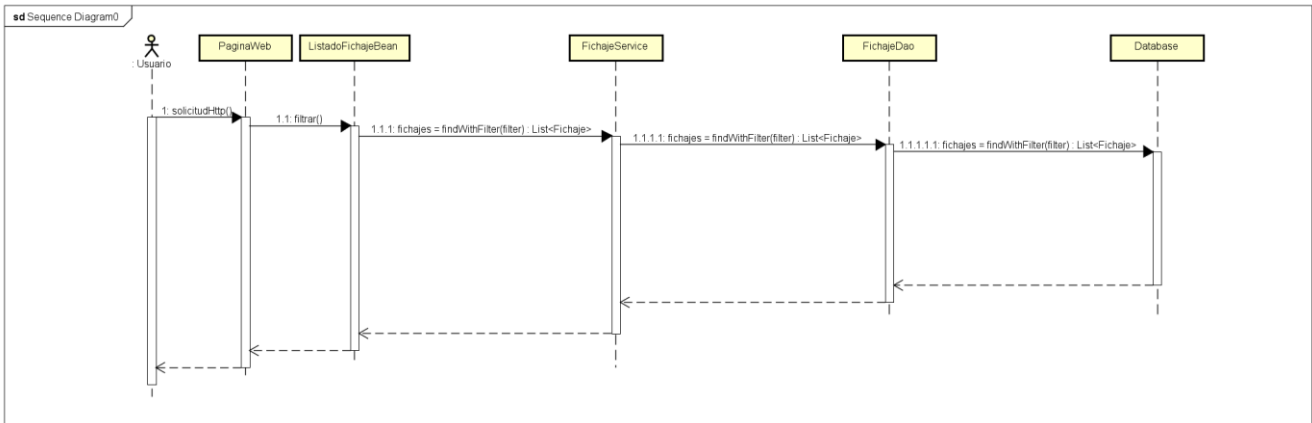


FIGURA 7. DIAGRAMA SECUENCIA LISTADO FICHAJES

3.6. Modelo de arquitectura. Diagrama de despliegue

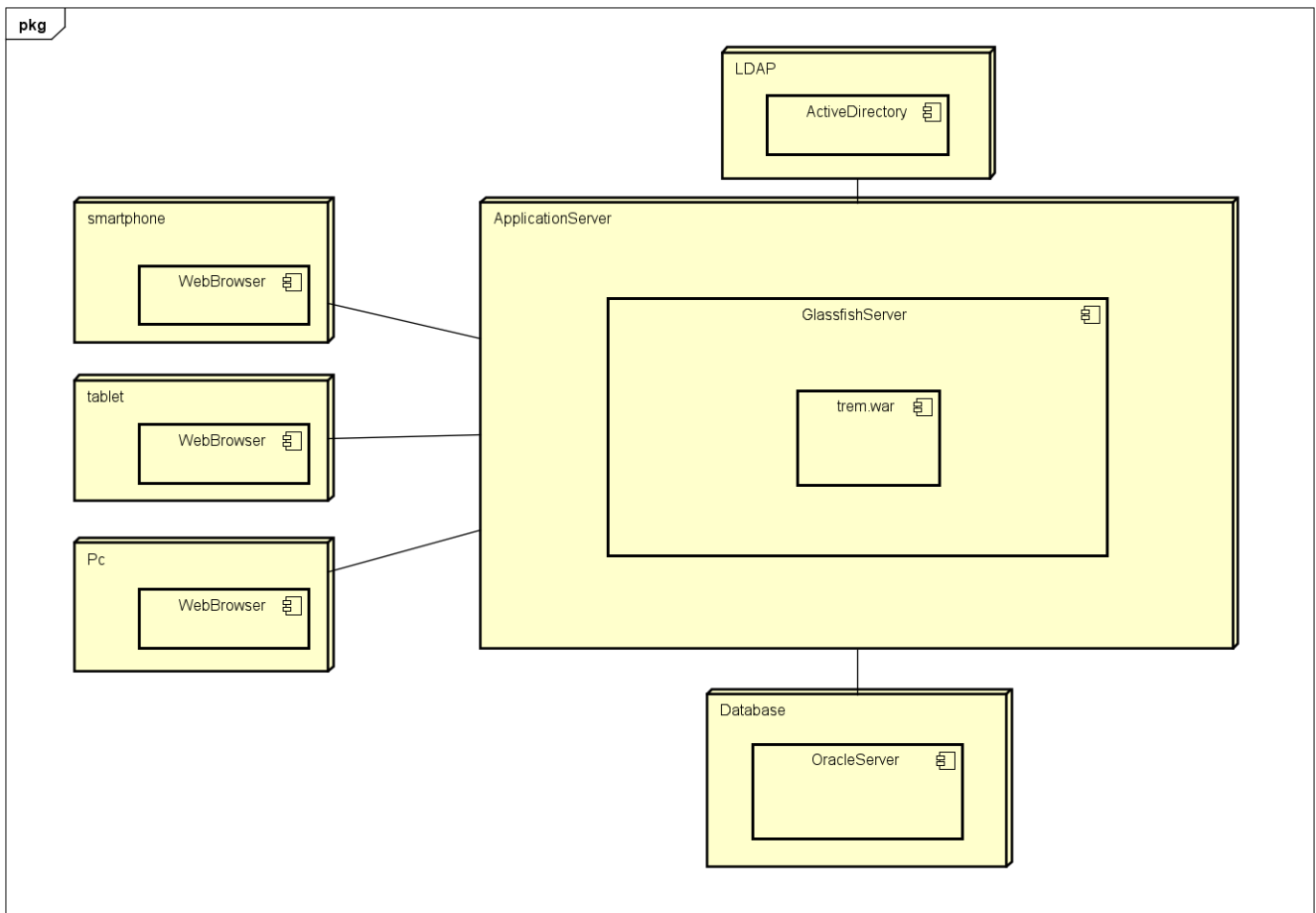


FIGURA 8. DIAGRAMA DESPLIEGUE

CAPÍTULO V: IMPLEMENTACIÓN

5.1. Diagrama de arquitectura/infraestructura

Diagrama de arquitectura-infraestructura del sistema (figura 1).

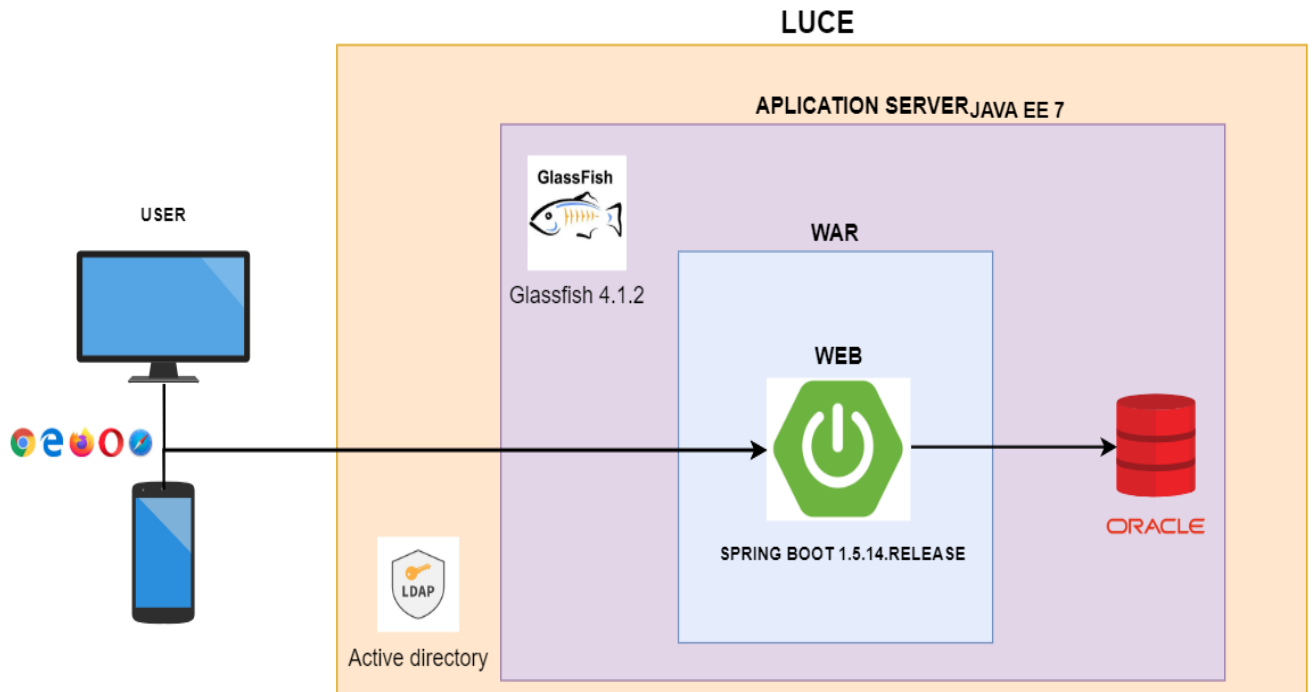


FIGURA 9. DIAGRAMA DE ARQUITECTURA-INFRAESTRUCTURA

5.2. Modelo de relacional

En el modelo de dominio del sistema se detallan las entidades del sistema, sus atributos y las relaciones que existen entre ellas (figura 3).

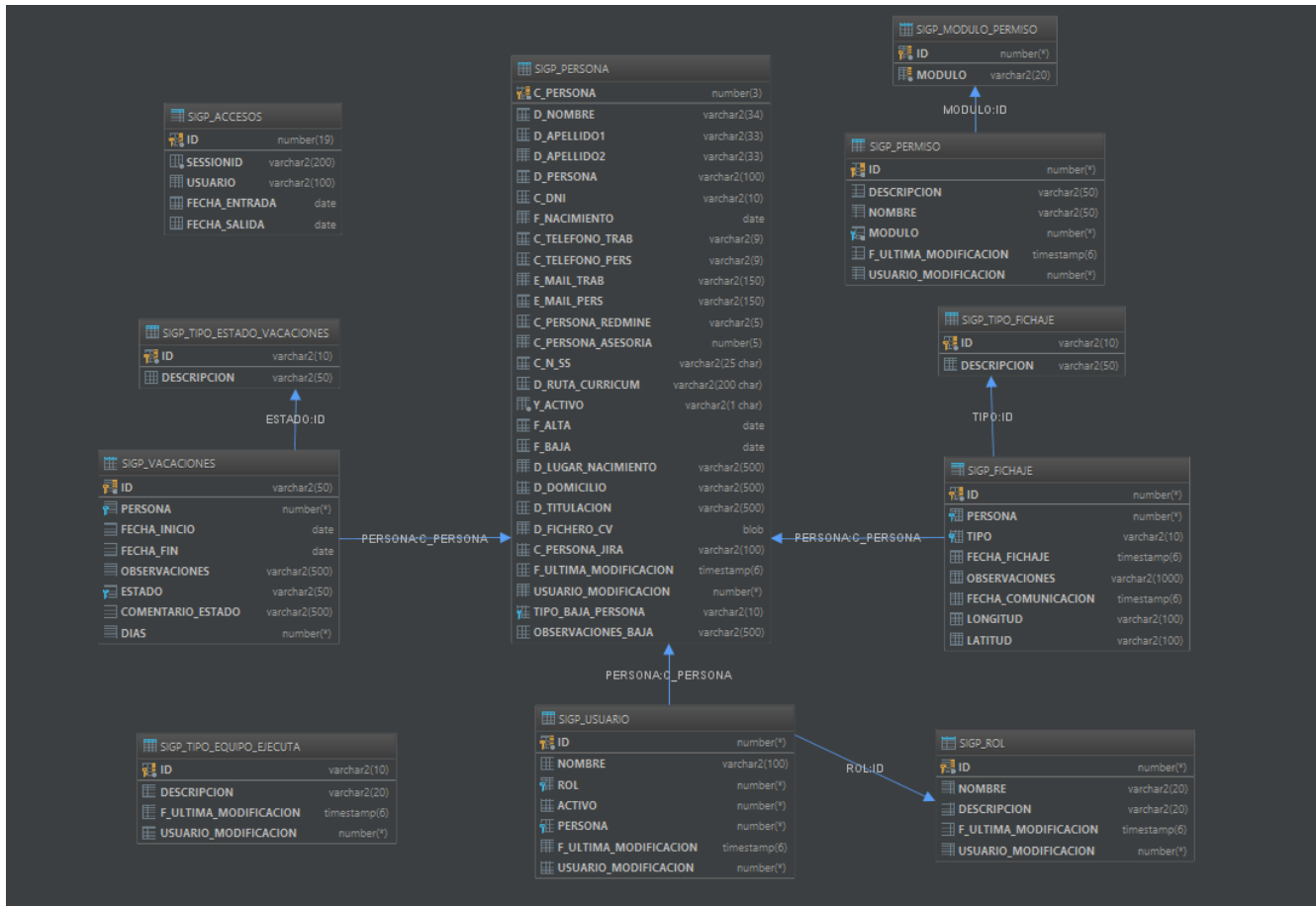


FIGURA 10. MODELO RELACIONAL

5.2.1. Descripción del modelo de relacional

- **Acceso:** Representa los accesos que realizan los usuarios a la aplicación.

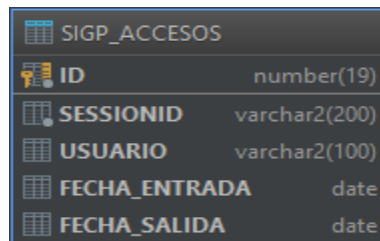


FIGURA 11. ENTIDAD ACCESOS

- **Equipo:** Representa los distintos equipos existentes dentro de la empresa (Sugus, Bigdata, Servicios ...).

SIGP_TIPO_EQUIPO_EJECUTA	
ID	varchar2(10)
DESCRIPCION	varchar2(20)
F_ULTIMA_MODIFICACION	timestamp(6)
USUARIO_MODIFICACION	number(*)

FIGURA 12. ENTIDAD EQUIPO

- **Fichaje:** Representa el fichaje horario de la jornada.

SIGP_FICHAJE	
ID	number(*)
PERSONA	number(*)
TIPO	varchar2(10)
FECHA_FICHAJE	timestamp(6)
OBSERVACIONES	varchar2(1000)
FECHA_COMUNICACION	timestamp(6)
LONGITUD	varchar2(100)
LATITUD	varchar2(100)

FIGURA 13. ENTIDAD FICHAJE

- **Modulo permiso:** Representa los distintos módulos de la aplicación.

SIGP_MODULO_PERMISO	
ID	number(*)
MODULO	varchar2(20)

FIGURA 14. ENTIDAD MODULO PERMISO

- **Permiso:** Representa los distintos permisos de la aplicación.

SIGP_PERMISO	
ID	number(*)
DESCRIPCION	varchar2(50)
NOMBRE	varchar2(50)
MODULO	number(*)
F_ULTIMA_MODIFICACION	timestamp(6)
USUARIO_MODIFICACION	number(*)

FIGURA 15. ENTIDAD PERMISO

- **Persona:** Representa las distintas personas dadas de alta en la aplicación.

SIGP_PERSONA	
C_PERSONA	number(3)
D_NOMBRE	varchar2(34)
D_APELLIDO1	varchar2(33)
D_APELLIDO2	varchar2(33)
D_PERSONA	varchar2(100)
C_DNI	varchar2(10)
F_NACIMIENTO	date
C_TELEFONO_TRAB	varchar2(9)
C_TELEFONO_PERS	varchar2(9)
E_MAIL_TRAB	varchar2(150)
E_MAIL_PERS	varchar2(150)
C_PERSONA_REDMINE	varchar2(5)
C_PERSONA_ASESORIA	number(5)
C_N_SS	varchar2(25 char)
D_RUTA_CURRICUM	varchar2(200 char)
Y_ACTIVO	varchar2(1 char)
F_ALTA	date
F_BAJA	date
D_LUGAR_NACIMIENTO	varchar2(500)
D_DOMICILIO	varchar2(500)
D_TITULACION	varchar2(500)
D_FICHERO_CV	blob
C_PERSONA_JIRA	varchar2(100)
F_ULTIMA_MODIFICACION	timestamp(6)
USUARIO_MODIFICACION	number(*)
TIPO_BAJA_PERSONA	varchar2(10)
OBSERVACIONES_BAJA	varchar2(500)

FIGURA 16. ENTIDAD PERSONA

- **Rol:** Representa los distintos roles de la aplicación.

SIGP_ROL	
ID	number(*)
NOMBRE	varchar2(20)
DESCRIPCION	varchar2(20)
F_ULTIMA_MODIFICACION	timestamp(6)
USUARIO_MODIFICACION	number(*)

FIGURA 17. ENTIDAD ROL

- **Tipo estado vacaciones:** Representa los distintos estados en los que se puede encontrar la solicitud de un periodo vacacional.

SIGP_TIPO_ESTADO_VACACIONES	
ID	varchar2(10)
DESCRIPCION	varchar2(50)

FIGURA 18. ENTIDAD TIPO ESTADO VACACIONES

- **Tipo fichaje:** Representa los distintos tipos de fichaje de la jornada horaria.

SIGP_TIPO_FICHAJE	
ID	varchar2(10)
DESCRIPCION	varchar2(50)

FIGURA 19. ENTIDAD TIPO FICHAJE

- **Usuario:** Representa a los usuarios de la aplicación.

SIGP_USUARIO	
ID	number(*)
NOMBRE	varchar2(100)
ROL	number(*)
ACTIVO	number(*)
PERSONA	number(*)
F_ULTIMA_MODIFICACION	timestamp(6)
USUARIO_MODIFICACION	number(*)

FIGURA 20. ENTIDAD USUARIO

- **Vacaciones:** Representa las solicitudes de los periodos vacacionales.

SIGP_VACACIONES	
ID	varchar2(50)
PERSONA	number(*)
FECHA_INICIO	date
FECHA_FIN	date
OBSERVACIONES	varchar2(500)
ESTADO	varchar2(50)
COMENTARIO_ESTADO	varchar2(500)
DIAS	number(*)

FIGURA 21. ENTIDAD VACACIONES

5.3. Ciclo de vida de la aplicación

Dado que se trata de un proyecto que servirá como base en la empresa para realizar el fichaje y la solicitud de los periodos vacacionales de los trabajadores, y en vistas de que será un proyecto que requerirá un desarrollo continuo para adaptar los cambios de la empresa e incluso añadir nuevas funcionalidades en el futuro, el desarrollo de la aplicación está ligado y dirigido por un ciclo de vida en el que se incluyen una serie de etapas para garantizar su seguridad, robustez, calidad, versionado y seguimiento.

Este ciclo de vida se lleva a cabo mediante el versionado de las ramas de trabajo en Azure DevOps Server y su posterior uso del código fuente del repositorio en la herramienta de Jenkins, la cual permite ejecutar un “jenkinsfile” en el que se describen una serie de pasos configurables a nuestro gusto para la construcción del artefacto final.

A continuación, se muestra un gráfico de las etapas incluidas en el ciclo de vida de la aplicación web (figura 14).

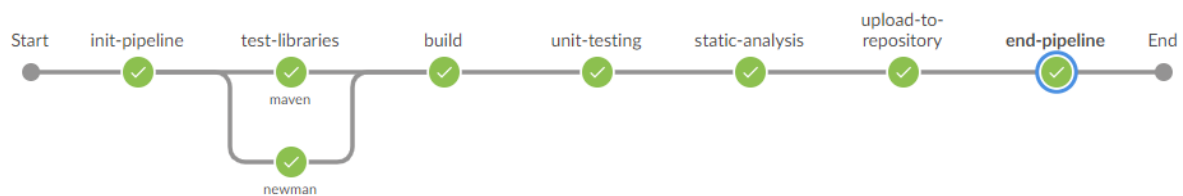


FIGURA 22. CICLO DE VIDA

A continuación, se describen las etapas incluidas en el ciclo de vida (tabla 25).

TABLA 27. ETAPAS CICLO DE VIDA

Ciclo de vida		
ETAPA	SOFTWARE	DESCRIPCIÓN
1- Test de librerías y dependencias (Test libraries)	Maven	Una vez obtenido el código fuente de la aplicación del repositorio de Azure DevOps Server, se comprueban las dependencias de las librerías utilizadas.
2- Construcción (Build)	Maven	Se construye el artefacto (war) mediante la ejecución de los comandos proporcionados por Maven.
3- Test unitarios (Unit-testing)	Maven	Se ejecutan los test de la aplicación mediante los comandos de Maven.
4- Análisis estático (Static-analysis)	SonarQube	Se comprueba la calidad del código mediante bugs, código duplicado, las vulnerabilidades, cobertura y malas prácticas de programación.
5- Subida al repositorio (Upload-repository)	Nexus	Se sube el artefacto generado al repositorio de Nexus.

5.4. Decisiones de implementación

5.4.1. Interfaz de web

La interfaz de web de la aplicación ha sido diseñada de manera que sea lo más sencilla e intuitiva posible para el usuario, con el fin de hacer agradable el uso de la aplicación web.

En cuanto a los estilos aplicados en la interfaz web cabe destacar que siempre se ha seguido una guía de estilos interna y propia de la empresa en la cual se utilizan tanto los colores como las fuentes utilizadas por la empresa en el desarrollo de su portal web.

Cabe destacar que se trata de una aplicación web responsive, es decir, se adapta a cualquier tipo de pantalla con el fin de hacer la aplicación accesible no solo desde la pantalla de un ordenador sino también desde una tablet o un smartphone (siempre a través del navegador).

5.4.2. Spring Boot + Primefaces

Dentro de las diferentes opciones que se manejaron en cuanto a cuáles debían de ser las tecnologías que implementarían tanto la parte de backend como de frontend, se decidió basar la aplicación en un backend desarrollado con el framework de Spring boot basado en Java, el cual permite la fácil inyección de dependencias y librerías en el proyecto y se construye utilizando la herramienta de Maven. Spring Boot dispone de un portal web (Spring Initializr) en el cual podemos construir nuestro proyecto basado en Spring Boot desde cero, eligiendo el tipo de proyecto, el lenguaje y versión con el que vamos a desarrollar nuestro proyecto, la versión de Spring Boot, el tipo de empaquetado y sus principales nombres de paquetes, artefactos. Además, este portal nos permite añadir las dependencias de las que hará uso el proyecto. Una vez elegidos los parámetros de nuestro proyecto, el portal web nos permite descargar un archivo (.zip) que contiene el proyecto y el cual ya podemos abrir desde cualquier IDE de desarrollo. Cabe destacar que inicialmente se pensó en una versión más actualizada de Spring Boot, pero que, debido a problemas de compatibilidad con los diferentes componentes externos de la aplicación, se utilizó una versión más antigua.

Por otro lado, se decidió utilizar el motor de plantilla de Primefaces ya que este ofrecía una amplia variedad de componentes y, además, por el conocimiento por mi parte, puesto que mis proyectos desarrollados hasta el momento en la empresa utilizaban todos ellos este motor de plantillas.

5.4.3. Base de datos Oracle

El tema de la base de datos fue una de las decisiones más difíciles del proyecto, y en la cual se encontraron las opiniones más distintas y enfrentadas por parte de los miembros y personas externas afectadas por el proyecto.

Por mi parte, estaba decidido y convencido que la mejor manera de abordar la base de datos era mediante un sistema de base de datos PostgreSQL, el cual es de tipo relacional y de código abierto. Este sistema de base de datos ofrecía grandes ventajas en cuanto a rendimiento y capacidad de almacenamiento.

Por parte del encargado de la administración de sistemas de bases de datos dentro de la empresa, y con el fin de ahorrar costes, el sistema de base de datos que se debía utilizar era una base de datos Oracle, debido a dos factores, en primer lugar porque los costes serían mucho menos debido a que la empresa ya disponía de este sistema integrado en sus soluciones de desarrollo y, en segundo lugar y mucho más importante, porque la aplicación, como bien aclaró el cliente y dio su aprobación para que fuera este el sistema que finalmente se utilizaría, utilizaría una serie de tablas comunes a otra aplicación ya desarrollada en la empresa (Trinity, una herramienta interna de gestión económica y de proyectos). Este último argumento hizo que la aplicación finalmente utilizara dicho sistema Oracle, con una serie de tablas comunes para ambas aplicaciones.

5.4.4. LDAP

En cuanto al método de autenticación basado en un LDAP, mediante el software de Active Directory la empresa centraliza el acceso a las aplicaciones de las cuales es responsable. Active Directory es un servicio de directorio desarrollado por Microsoft para redes de dominio de Windows que autentica y autoriza a todos los usuarios y equipos de un software tipo de dominio de red de asignar y hacer cumplir las políticas de seguridad para todos los equipos y la instalación o actualización.

Mediante esta tecnología implantada en la empresa, la aplicación utiliza dicho servicio para autenticar a los usuarios de la aplicación, evitando riesgos de almacenamiento y encriptación de contraseñas de los usuarios. Una vez el usuario ha introducido sus credenciales en la pantalla de login de la aplicación, ésta se conecta con el servicio mencionado para validar dichas credenciales.

5.4.5. Despliegue en servidor Glassfish

Por último, otra de las decisiones importantes que se debían tomar en el proyecto era el servidor en el cual se desplegaría la aplicación. Cabe destacar que aquí también hubo algunas opiniones dispares, había quien defendía que se debía utilizar un servidor Tomcat puesto que era el que utiliza Spring Boot en su desarrollo en local. Pero finalmente se decidió utilizar un servidor GlassFish 4 ya que era un servidor mucho más configurable en cuanto a parámetros de conexiones y seguimiento de la actividad de la aplicación.

CAPÍTULO VI: PRUEBAS

6.1. Pruebas realizadas en la aplicación

Las pruebas son un aspecto fundamental a la hora de desarrollar cualquier sistema ya que nos permiten detectar posibles fallos o vulnerabilidades. A continuación, se detallan las pruebas a las que ha sido sometida la aplicación web una vez se dio por concluida la misma y se desplegó entorno de producción.

TABLA 28. PRUEBA 1

Login válido	
Descripción	El usuario introduce un username y password correctos.
Entrada	Username, password y pulsación de botón "iniciar sesión".
Resultado esperado	El sistema comprueba los datos introducidos y deja que el usuario acceda al interior de la aplicación.
Resultado prueba	Correcto.

TABLA 29. PRUEBA 2

Login incorrecto (username)	
Descripción	El usuario introduce un username incorrecto.
Entrada	Username, password y pulsación de botón "iniciar sesión".
Resultado esperado	El sistema comprueba los datos introducidos y al comprobar que son erróneos, impide que el usuario acceda al interior de la aplicación y muestra un mensaje de error.
Resultado prueba	Correcto.

TABLA 30. PRUEBA 3

Login incorrecto (password)	
Descripción	El usuario introduce un password incorrecto.
Entrada	Username, password y pulsación de botón "iniciar sesión".
Resultado esperado	El sistema comprueba los datos introducidos y al comprobar que son erróneos, impide que el usuario acceda al interior de la aplicación y muestra un mensaje de error.
Resultado prueba	Correcto.

TABLA 31. PRUEBA 4

Responsive tablet	
Descripción	El usuario introduce los datos acerca del fichaje.
Entrada	Datos de fichaje y pulsación de botón “fichar”.
Resultado esperado	El sistema comprueba los datos introducidos por el usuario, registra el fichaje y redirige al usuario a la pantalla del listado de fichajes.
Resultado prueba	Incorrecto. El sistema no guarda las coordenadas del fichaje correctamente.

El fichaje no guardaba de manera correcta las coordenadas debido a que el navegador debía de utilizar el protocolo https o si se estaba trabajando sobre el proyecto de manera local, ‘localhost’ y no la dirección IP. Tras la solución de este error se ejecutó de nuevo la prueba, prueba 5.

TABLA 32. PRUEBA 5

Fichaje válido	
Descripción	El usuario introduce los datos acerca del fichaje.
Entrada	Datos de fichaje y pulsación de botón “fichar”.
Resultado esperado	El sistema comprueba los datos introducidos por el usuario, registra el fichaje y redirige al usuario a la pantalla del listado de fichajes.
Resultado prueba	Correcto.

TABLA 33. PRUEBA 6

Fichaje erróneo (fecha)	
Descripción	El usuario introduce los datos acerca del fichaje, pero la fecha del fichaje es anterior a la de su último fichaje.
Entrada	Datos de fichaje y pulsación de botón “fichar”.
Resultado esperado	El sistema comprueba los datos introducidos por el usuario, y al detectar que la fecha es anterior a la de su último fichaje, muestra un mensaje con el error y no registra el fichaje.
Resultado prueba	Correcto.

TABLA 34. PRUEBA 7

Fichaje erróneo (tipo fichaje)	
Descripción	El usuario introduce los datos acerca del fichaje, pero el tipo de fichaje es el mismo que el de su último fichaje.
Entrada	Datos de fichaje y pulsación de botón “fichar”.
Resultado esperado	El sistema comprueba los datos introducidos por el usuario, y al detectar que el tipo de fichaje es el mismo que el de su último fichaje, muestra un mensaje con el error y no registra el fichaje.
Resultado prueba	Correcta.

TABLA 35. PRUEBA 8

Carga informe horas	
Descripción	El usuario accede a la opción de menú “informe horas” y el sistema precarga las horas registradas por el usuario en los últimos 7 días.
Entrada	Acceso a la opción de menú “informe horas”.
Resultado esperado	El sistema precarga las horas registradas por el usuario en los últimos 7 días.
Resultado prueba	Correcto.

TABLA 36. PRUEBA 9

Carga informe horas con fichajes sin cerrar	
Descripción	El usuario accede a la opción de menú “informe horas” y el sistema precarga las horas registradas por el usuario en los últimos 7 días.
Entrada	Acceso a la opción de menú “informe horas”.
Resultado esperado	El sistema precarga las horas registradas por el usuario en los últimos 7 días y detecta que existe algún día en el que usuario no ha completado los fichajes (Entrada/Salida). El sistema muestra el informe y avisa al usuario del día detectado.
Resultado prueba	Correcto.

TABLA 37. PRUEBA 10

Carga informe horas con fichajes erróneos	
Descripción	El usuario accede a la opción de menú “informe horas” y el sistema precarga las horas registradas por el usuario en los últimos 7 días.
Entrada	Acceso a la opción de menú “informe horas”.
Resultado esperado	El sistema precarga las horas registradas por el usuario en los últimos 7 días y detecta que existe algún día en el que usuario ha registrado fichajes de manera errónea. El sistema muestra el informe y avisa al usuario del día detectado.
Resultado prueba	Correcto.

TABLA 38. PRUEBA 11

Carga calendario de vacaciones	
Descripción	El usuario accede a la opción del submenú de vacaciones “calendario” y el sistema precarga el calendario en el mes actual con las solicitudes registradas.
Entrada	El usuario accede a la opción del submenú de vacaciones “calendario”.
Resultado esperado	El sistema precarga el calendario en el mes actual con las solicitudes registradas.
Resultado prueba	Correcto.

TABLA 39. PRUEBA 12

Solicitud periodo vacaciones	
Descripción	El usuario registra una solicitud de periodo de vacaciones.
Entrada	Datos de la solicitud y pulsación del botón “Solicitar”.
Resultado esperado	El sistema registra la solicitud del usuario.
Resultado prueba	Correcto.

TABLA 40. PRUEBA 13

Eliminación de solicitud periodo vacaciones	
Descripción	El usuario elimina una solicitud de periodo de vacaciones.
Entrada	Datos de la solicitud y pulsación del botón "Eliminar".
Resultado esperado	El sistema elimina la solicitud registrada por el usuario.
Resultado prueba	Correcto.

TABLA 41. PRUEBA 14

Solicitud periodo vacaciones erróneo (fin de semana)	
Descripción	El usuario registra una solicitud de periodo de vacaciones.
Entrada	Datos de la solicitud y pulsación del botón "Solicitar".
Resultado esperado	El sistema comprueba los datos de la solicitud y al comprobar que la fecha de inicio o fin es fin de semana, muestra un mensaje de error al usuario y no registra la solicitud.
Resultado prueba	Correcto.

TABLA 42. PRUEBA 15

Aprobar solicitud periodo vacaciones	
Descripción	El usuario (Rol con dicho permiso) aprueba una solicitud de periodo de vacaciones.
Entrada	Datos de la solicitud y pulsación del botón "Aprobar".
Resultado esperado	El sistema cambia el estado de la solicitud a "Aprobada".
Resultado prueba	Correcto.

TABLA 43. PRUEBA 16

Rechazar solicitud periodo vacaciones	
Descripción	El usuario (Rol con dicho permiso) rechaza una solicitud de periodo de vacaciones.
Entrada	Datos de la solicitud y pulsación del botón "Rechazar".
Resultado esperado	El sistema cambia el estado de la solicitud a "Rechazada".
Resultado prueba	Correcto.

TABLA 44. PRUEBA 17

Filtro fechas en listado fichajes	
Descripción	El usuario introduce unas fechas en los criterios de búsqueda.
Entrada	Fechas del filtro y pulsación de botón “filtrar”.
Resultado esperado	El sistema comprueba que los valores del filtro son correctos y muestra el listado de fichajes que cumplen los criterios de búsqueda.
Resultado prueba	Correcto.

TABLA 45. PRUEBA 18

Filtro tipo fichaje en listado fichajes	
Descripción	El usuario selecciona el tipo de fichaje en los criterios de búsqueda.
Entrada	Tipo de fichaje del filtro y pulsación de botón “filtrar”.
Resultado esperado	El sistema comprueba que los valores del filtro son correctos y muestra el listado de fichajes que cumplen los criterios de búsqueda.
Resultado prueba	Correcto.

TABLA 46. PRUEBA 19

Filtro persona en listado fichajes	
Descripción	El usuario introduce una persona en los criterios de búsqueda.
Entrada	Persona del filtro y pulsación de botón “filtrar”.
Resultado esperado	El sistema comprueba que los valores del filtro son correctos y muestra el listado de fichajes que cumplen los criterios de búsqueda.
Resultado prueba	Correcto.

TABLA 47. PRUEBA 20

Filtro fechas en informe horas	
Descripción	El usuario introduce unas fechas en los criterios de búsqueda.
Entrada	Fechas del filtro y pulsación de botón “filtrar”.
Resultado esperado	El sistema comprueba que los valores del filtro son correctos y muestra el listado de horas registradas que cumplen los criterios de búsqueda.
Resultado prueba	Correcto.

TABLA 48. PRUEBA 21

Filtro persona en informe horas	
Descripción	El usuario introduce una persona en los criterios de búsqueda.
Entrada	Persona del filtro y pulsación de botón “filtrar”.
Resultado esperado	El sistema comprueba que los valores del filtro son correctos y muestra el listado de horas registradas que cumplen los criterios de búsqueda.
Resultado prueba	Correcto.

TABLA 49. PRUEBA 22

Filtro fechas en listado vacaciones	
Descripción	El usuario introduce unas fechas en los criterios de búsqueda.
Entrada	Fechas del filtro y pulsación de botón “filtrar”.
Resultado esperado	El sistema comprueba que los valores del filtro son correctos y muestra el listado de solicitudes de periodos vacacionales que cumplen los criterios de búsqueda.
Resultado prueba	Correcto.

TABLA 50. PRUEBA 23

Filtro persona en listado vacaciones	
Descripción	El usuario introduce una persona en los criterios de búsqueda.
Entrada	Persona del filtro y pulsación de botón “filtrar”.
Resultado esperado	El sistema comprueba que los valores del filtro son correctos y muestra el listado de solicitudes de periodos vacacionales que cumplen los criterios de búsqueda.
Resultado prueba	Correcto.

TABLA 51. PRUEBA 24

Filtro estado solicitud en listado vacaciones	
Descripción	El usuario selecciona un estado de solicitud en los criterios de búsqueda.
Entrada	Estado de solicitud del filtro y pulsación de botón “filtrar”.
Resultado esperado	El sistema comprueba que los valores del filtro son correctos y muestra el listado de solicitudes de periodos vacacionales que cumplen los criterios de búsqueda.
Resultado prueba	Correcto.

TABLA 52. PRUEBA 25

changelog	
Descripción	El usuario accede al changelog de la aplicación.
Entrada	Pulsación del enlace de changelog en el footer de la aplicación.
Resultado esperado	El sistema muestra al usuario la página con el changelog de la aplicación.
Resultado prueba	Correcto.

TABLA 53. PRUEBA 26

Responsive smartphone	
Descripción	El usuario accede a la aplicación a través del navegador web un smartphone.
Entrada	Acceso a la aplicación desde el navegador del smartphone.
Resultado esperado	El sistema muestra al usuario las mismas acciones que se mostrarían si se accede desde cualquier navegador y la interfaz de la aplicación se adapta a la pantalla del smartphone.
Resultado prueba	Incorrecto. El menú de la aplicación no se muestra de manera correcta.

El menú no se mostraba de manera correcta ya que al ser demasiado grande este ocupaba la mayor parte de la pantalla y hacía que esta no fuera manejable desde el punto de vista del usuario. Tras la solución de este error, creando un nuevo menú para los dispositivos cuya pantalla fuera de un tamaño reducido, se ejecutó de nuevo la prueba, prueba 27.

TABLA 54. PRUEBA 27

Responsive smartphone	
Descripción	El usuario accede a la aplicación a través del navegador web un smartphone.
Entrada	Acceso a la aplicación desde el navegador del smartphone.
Resultado esperado	El sistema muestra al usuario las mismas acciones que se mostrarían si se accede desde cualquier navegador y la interfaz de la aplicación se adapta a la pantalla del smartphone.
Resultado prueba	Correcto.

TABLA 55. PRUEBA 28

Responsive tablet	
Descripción	El usuario accede a la aplicación a través del navegador web una tablet.
Entrada	Acceso a la aplicación desde el navegador de una tablet.
Resultado esperado	El sistema muestra al usuario las mismas acciones que se mostrarían si se accede desde cualquier navegador y la interfaz de la aplicación se adapta a la pantalla de la tablet.
Resultado prueba	Correcto.

CAPÍTULO VII: CONCLUSIONES

7.1. Conclusiones

Una vez concluido el proyecto, podemos decir que este ha cumplido con los objetivos que se planificaban al comienzo de manera satisfactoria, llegando a cumplir con creces los dos objetivos principales, el registro de la jornada horaria y las solicitudes de periodos vacacionales por partes de los empleados de una empresa.

En primer lugar, me gustaría destacar que este proyecto ha sido un muy gratificante tanto en el ámbito laboral como en el ámbito académico, puesto que ha supuesto la realización de este desde cero por mi parte, lo cual conlleva una serie de estudios, análisis, problemas y soluciones desconocidos hasta el momento y que gracias a este proyecto he sido capaz de enfrentarme a ellos y, lo más importante, desarrollar una serie de soluciones apropiadas para los mismos.

Otra de las conclusiones a las que he llegado con este proyecto, es que la metodología ágil que se usó para el desarrollo de este ha sido muy útil y acertada, puesto que el proyecto por momentos era cambiante, ya que, al tratarse de un proyecto interno de la empresa, la capacidad de comunicación, y por tanto de cambios repentinos, era por momentos continua.

En cuanto a las estimaciones de las HU de usuario, se puede concluir que en general han sido acertadas en todo el proyecto, aunque también me gustaría destacar que la primera vez que se despliega una aplicación, y aunque existan aplicaciones similares en cuanto a tecnología de despliegue, existen muchos otros detalles relacionado con temas de configuración que hacen que el proyecto se retrase más de lo deseado.

Por otro lado, en el aspecto más negativo en cuanto a las conclusiones obtenidas del desarrollo del proyecto, aunque también sirvan como aprendizaje, me gustaría destacar que la adaptación, o como es en este caso, el uso de un recurso compartido por parte de varias aplicaciones, aunque al principio se pueda ver como un aspecto que ahorra costes, puede suponer que la manera en que se desarrolla una parte de la funcionalidad del proyecto se vea gravemente afectada, y esto haga que el proyecto en gran parte dependa o tenga que estar ligado a dicho recurso compartido.

Por último, me gustaría destacar que la inclusión de la aplicación en el ciclo de vida de desarrollo, a pesar de que requiera de gran parte de tiempo del proyecto para el estudio de las distintas herramientas utilizadas, supone un gran beneficio a medida que el proyecto avanza, puesto que agiliza de manera muy eficaz la parte de desarrollo de este y mejora tanto la calidad como el backup ante posibles contratiempos.

7.2. Futuras mejoras

Como se ha mencionado anteriormente, este proyecto servirá como base para la creación de un proyecto más amplio en el que la empresa pueda registrar y utilizar los datos relacionados con el fichaje de la jornada horaria y periodos vacacionales. A continuación, se muestran una serie de futuras mejoras ya detectadas en la aplicación.

7.2.1. Nuevas HU tras despliegue en entorno de producción

Tras el despliegue de la aplicación en el entorno de producción y el uso de la misma por mayor número de usuarios, surgieron una serie de HU (tabla 52) que a corto plazo se incluirán en el proyecto para hacer de este un proyecto más completo y que aporte mayor valor tanto a la empresa como al conjunto de los usuarios que la utilicen.

TABLA 56. HU FUTURAS

HU	Nombre	Detalle	PH
19	Resumen fichaje	Enviar un correo tras el fichaje con los datos grabados y otro a final de semana con un resumen de los fichajes de esta.	2
20	Activar o desactivar correos cada fichaje	Permitir activar o desactivar el envío de correo tras el fichaje.	0.5
21	Marcar tipo de fichaje por defecto	Marcar por defecto el tipo de fichaje cuando se entra en la pantalla según el último fichaje registrado por el usuario.	0.25

7.2.2. App móvil en vez de acceder por navegador

Aunque ya se planteó en un inicio la posibilidad de creación de una aplicación android, siempre teniendo en cuenta que al igual que ahora la aplicación web fuera responsive, se denegó en un principio puesto que el proyecto debía ser desarrollado en un periodo de tiempo más reducido del que supondría la creación a mayores de la aplicación android. Quizá en un futuro, ya sea cercano o no, dicha posibilidad se plantee de nuevo y esta vez sí sea aceptada puesto que en paralelo los usuarios podrían utilizar la aplicación.

7.2.3. Vista de los datos en una gráfica

Otra de las mejoras futuras que ya se han mencionado en las distintas reuniones acerca del proyecto es la de que los datos registrados puedan ser visualizados de una manera mucho más intuitiva y que de un vistazo rápido se pueda obtener información. Esto se ha planteado de manera que los datos sean visibles a través de gráficos con herramientas como Power BI.

BIBLIOGRAFÍA

[1] Baeldung. Intro to Jenkins 2 and the Power of Pipelines. Disponible en

<https://www.baeldung.com/jenkins-pipelines>

(último acceso: 24/03/2020)

[2] Bloch, Joshua. "Effective Java. Third edition". Addison-Wesley (2018).

[3] De las Heras del Dedo, Rafael; Lasa Gómez, Carmen; Álvarez García, Alonso. "Métodos ágiles: Scrum, Kanban, Lean." Anaya (2017).

[4] Github. GlassFish Server deployment guide. Disponible en

<https://javaee.github.io/glassfish/doc/4.0/deployment-planning-guide.pdf>

(último acceso: 03/04/2020)

[5] Github. GlassFish Server installation guide. Disponible en

<https://javaee.github.io/glassfish/doc/4.0/installation-guide.pdf>

(último acceso: 03/04/2020)

[6] Github. GlassFish Server quick start guide. Disponible en

<https://javaee.github.io/glassfish/doc/4.0/quick-start-guide.pdf>

(último acceso: 03/04/2020)

[7] Google. Ubicación en navegador web. Disponible en

<https://developers.google.com/web/fundamentals/native-hardware/user-location?hl=es>

(último acceso: 21/02/2020)

[8] Jenkins. Jenkins User documentation. Disponible en

<https://jenkins.io/doc/>

(último acceso: 24/03/2020)

[9]. Primefaces. Primefaces showcase. Disponible en

<https://www.primefaces.org/showcase/>

(último acceso: 26/04/2020)

[10] Programación.net. Ubicación en navegador web. Disponible en

https://programacion.net/articulo/obtener_la_localizacion_actual_del_usuario_con_jquery_1477

(último acceso: 22/02/2020)

[11] SonarQube. SonarQube documentation. Disponible en

<https://docs.sonarqube.org/pages/>

(último acceso: 21/03/2020)

[12] Sonatype. Nexus repository manager 3. Disponible en

<https://help.sonatype.com/repomanager3>

(último acceso: 21/03/2020)

[13] Spring. Spring inicializr. Disponible en

<https://start.spring.io/>

(último acceso: 06/12/2019)

[14] Stackoverflow. Resolución de problemas. Disponible en

<https://stackoverflow.com/>

(último acceso: 26/04/2020)

[15] Trigás Gallego, Manuel. "Metodología scrum." (2012). Disponible en

<http://openaccess.uoc.edu/webapps/o2/bitstream/10609/17885/1/mtrigasTFC0612memoria.pdf>

(último acceso: 19/02/2020)

[16] Udey. Universidad Java: De Cero a Máster. Disponible en

<https://www.udemy.com/course/universidad-java-especialista-en-java-desde-cero-a-master/>

(último acceso: 03/12/2019)

[17] Udey. Desarrollo Web con Spring Boot - De Cero a Ninja. Disponible en

<https://www.udemy.com/course/desarrollo-web-con-spring-framework-4-de-cero-a-ninja/>

(último acceso: 16/12/2019)

ANEXO I

I. Manual de usuario

1. Instalación

A continuación, se presenta una lista de los pasos a seguir para desplegar nuestro artefacto en un servidor Glassfish 4:

- Descargar el servidor de la página oficial de Glassfish, en mi caso la versión 4.1.2.
- Descomprimir el archivo .zip descargado
- Ejecutar desde consola el archivo “../glassfish-4.1.2\glassfish4\glassfish\bin\startserv.bat
- Acceder a la interfaz web de administración del servidor en “localhost:4848”
- Cuando el servidor se inicie, accederá a la interfaz web de administración y mostrará una página como la siguiente:

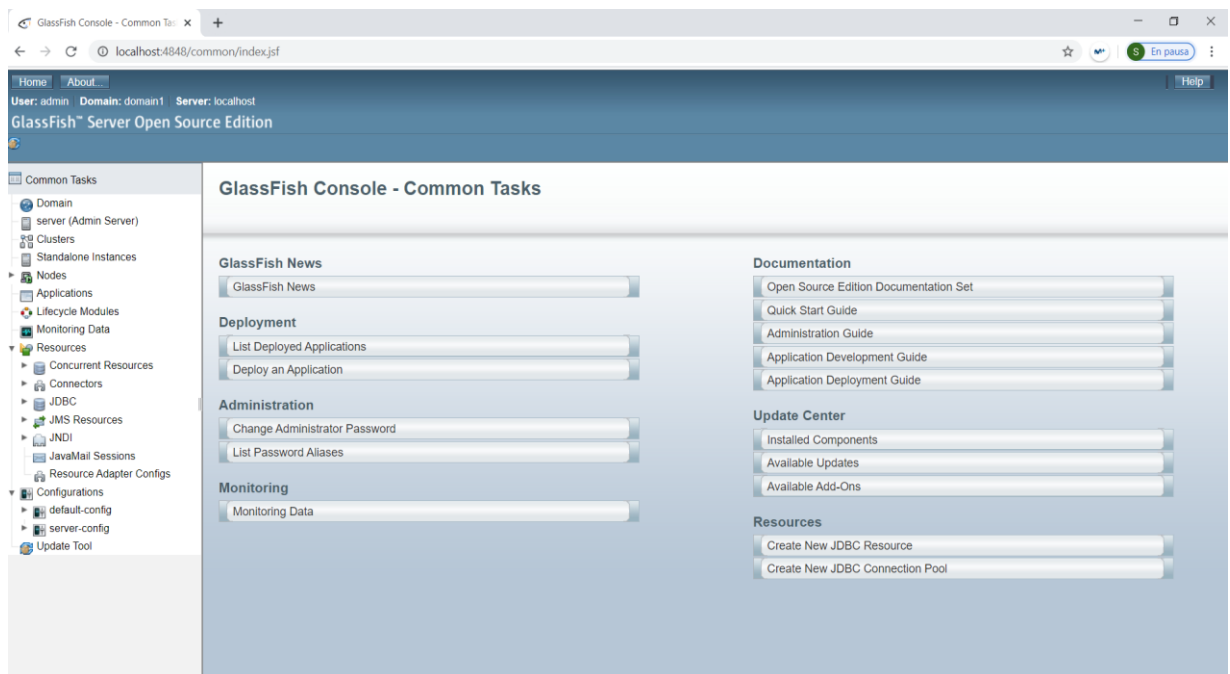


FIGURA 23. INTERFAZ WEB SERVIDOR GLASSFISH

- Accedemos a la opción de “Applications”

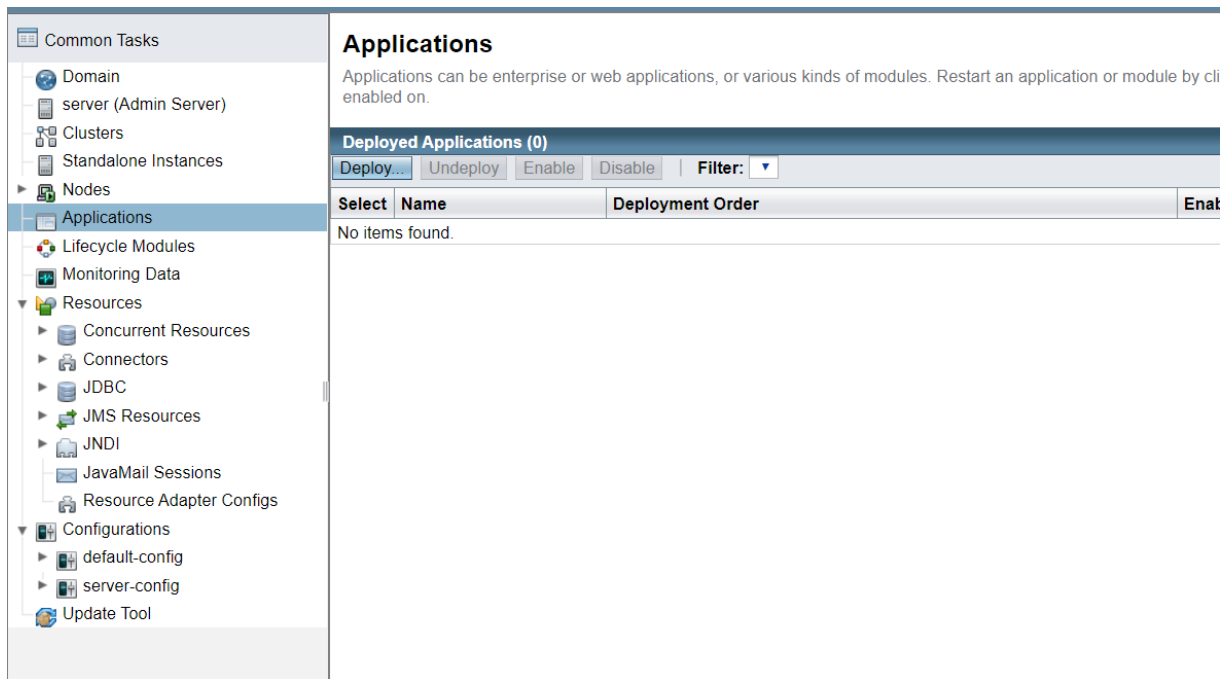


FIGURA 24. INTERFAZ APPLICATIONS SERVIDOR GLASSFISH

- Pulsamos el botón “Deploy” para desplegar nuestro artefacto y nos mostrará la siguiente pantalla:

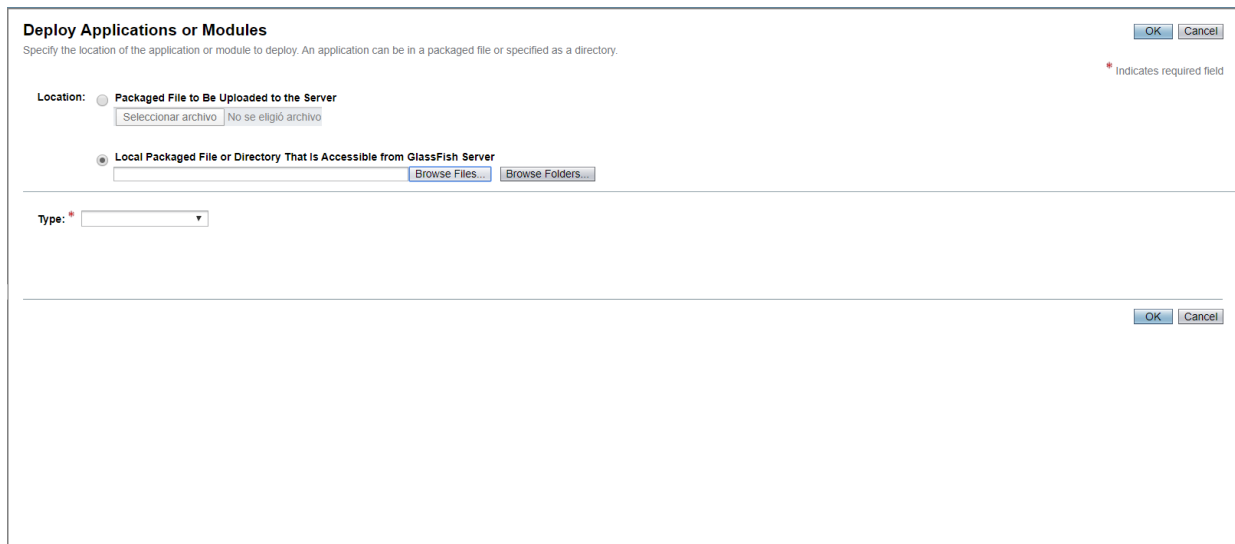


FIGURA 25. INTERFAZ DEPLOY SERVIDOR GLASSFISH

- Seleccionamos la opción de “Local Packaged File or Directory That Is Accessible from GlassFish Server” y buscamos el artefacto en nuestro sistema:

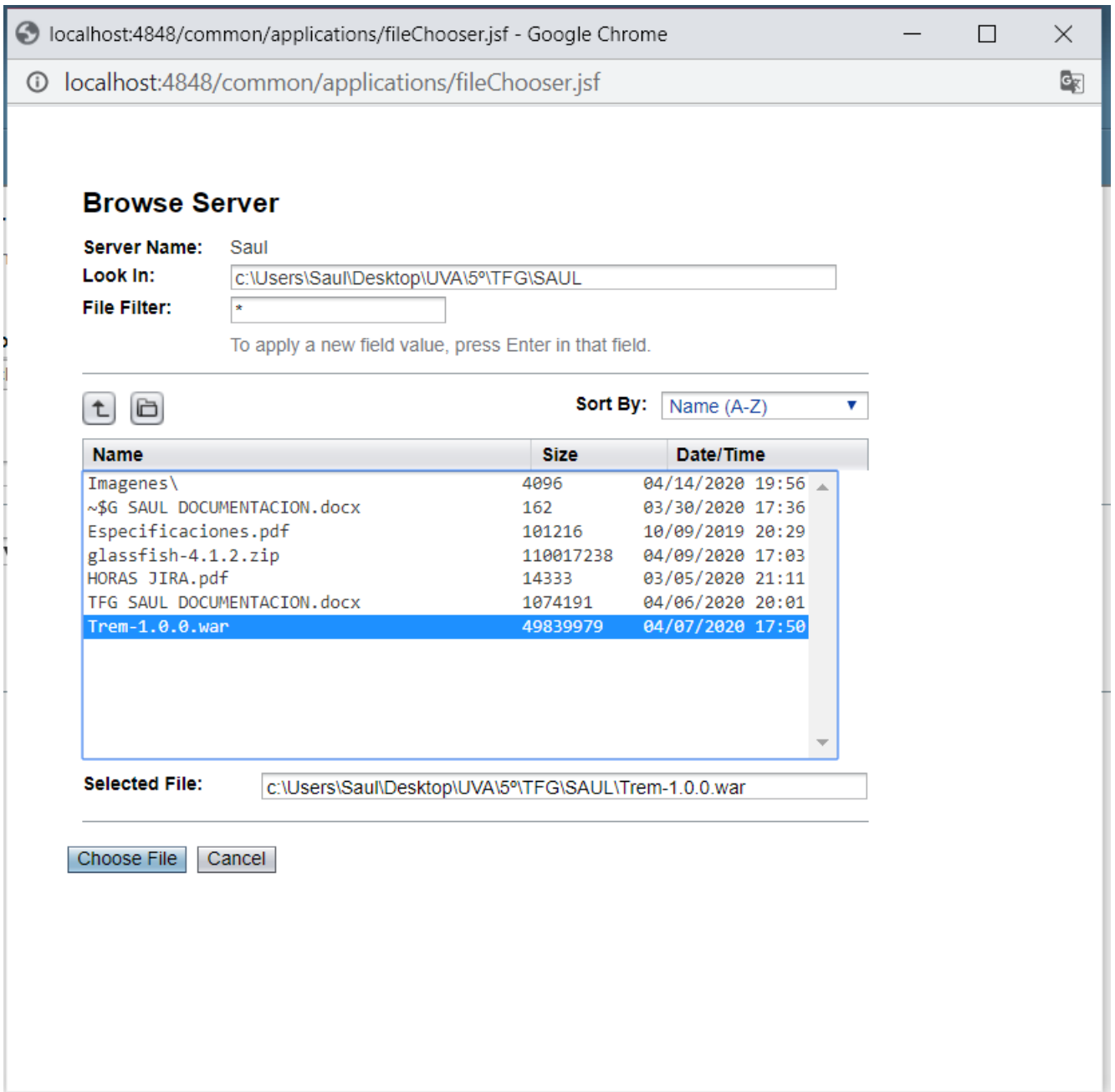


FIGURA 26. INTERFAZ SELECCIÓN ARTEFACTO SERVIDOR GLASSFISH

- Pulsamos en “OK” y esperamos a que el servidor despliegue nuestra aplicación. Cuando el proceso termine nos mostrará una pantalla como la siguiente:

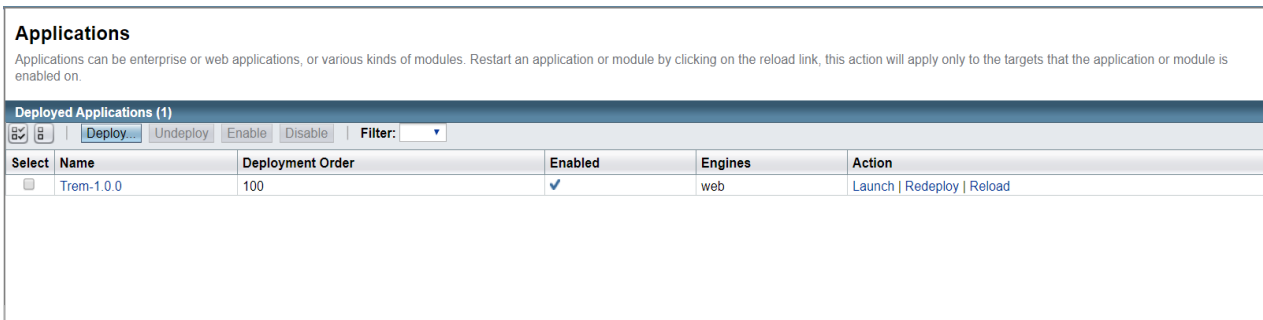


FIGURA 27. INTERFAZ APPLICATIONS DEPLOY SERVIDOR GLASSFISH

- Accedemos al interior de la aplicación pulsando sobre el nombre y nos mostrará la siguiente página:

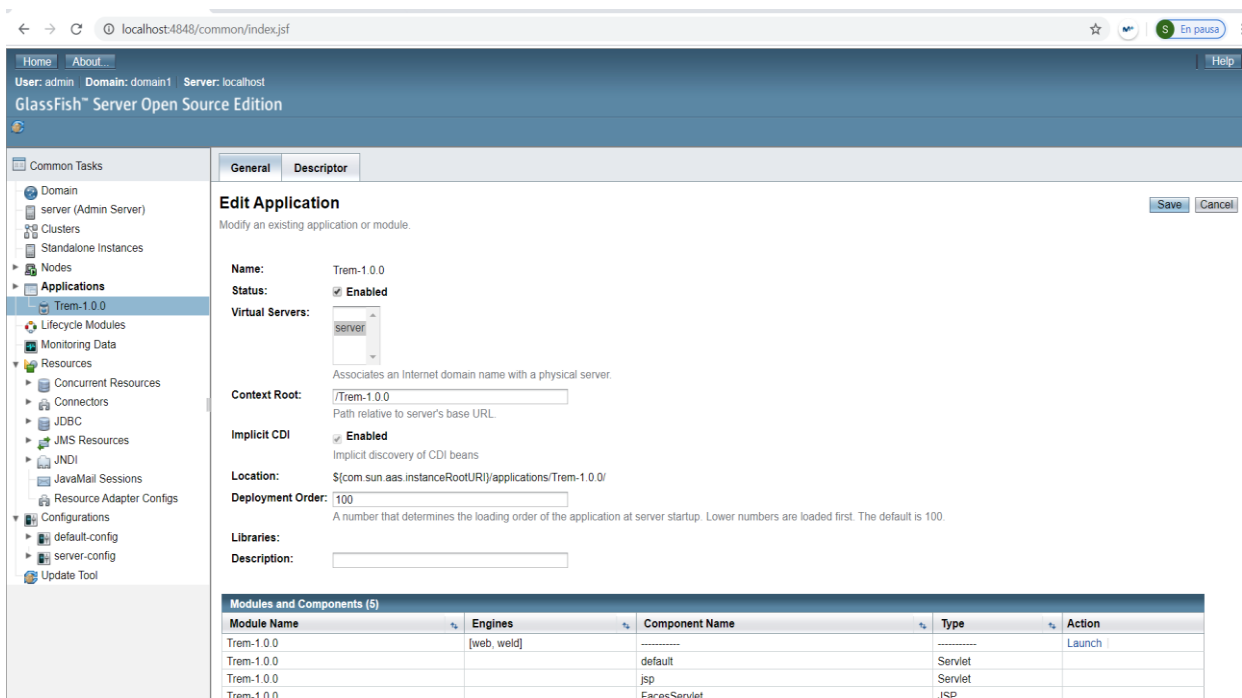


FIGURA 28. INTERFAZ ARTEFACTO SERVIDOR GLASSFISH

- En el paginar de abajo pulsamos sobre el enlace de “Launch” y nos mostrará las distintas opciones de acceso a la aplicación web:

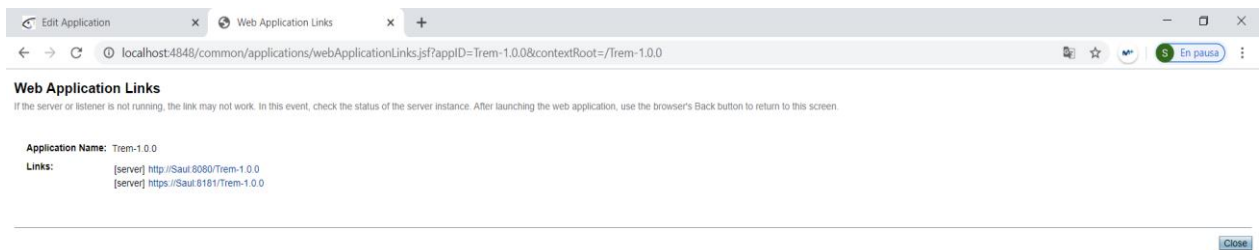


FIGURA 29. INTERFAZ WEB APPLICATIONS LINK SERVIDOR GLASSFISH

- Accedemos (preferiblemente por la opción https) a la aplicación y se mostrará la página de login de la aplicación:

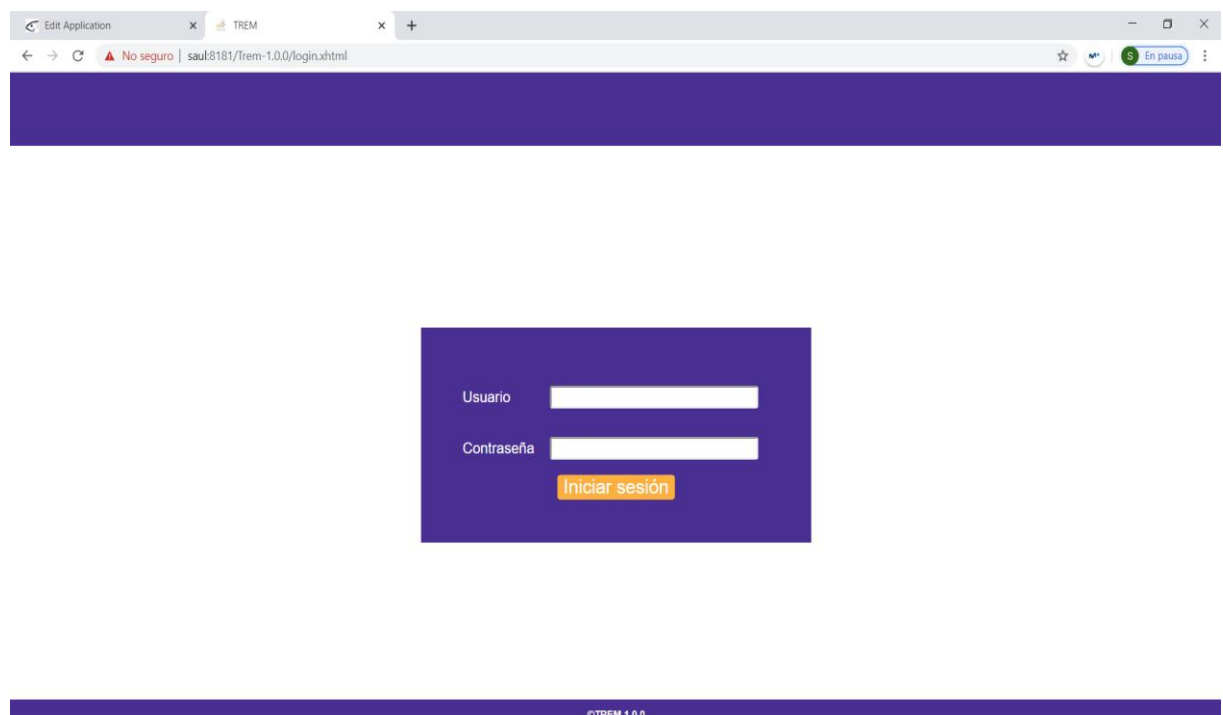
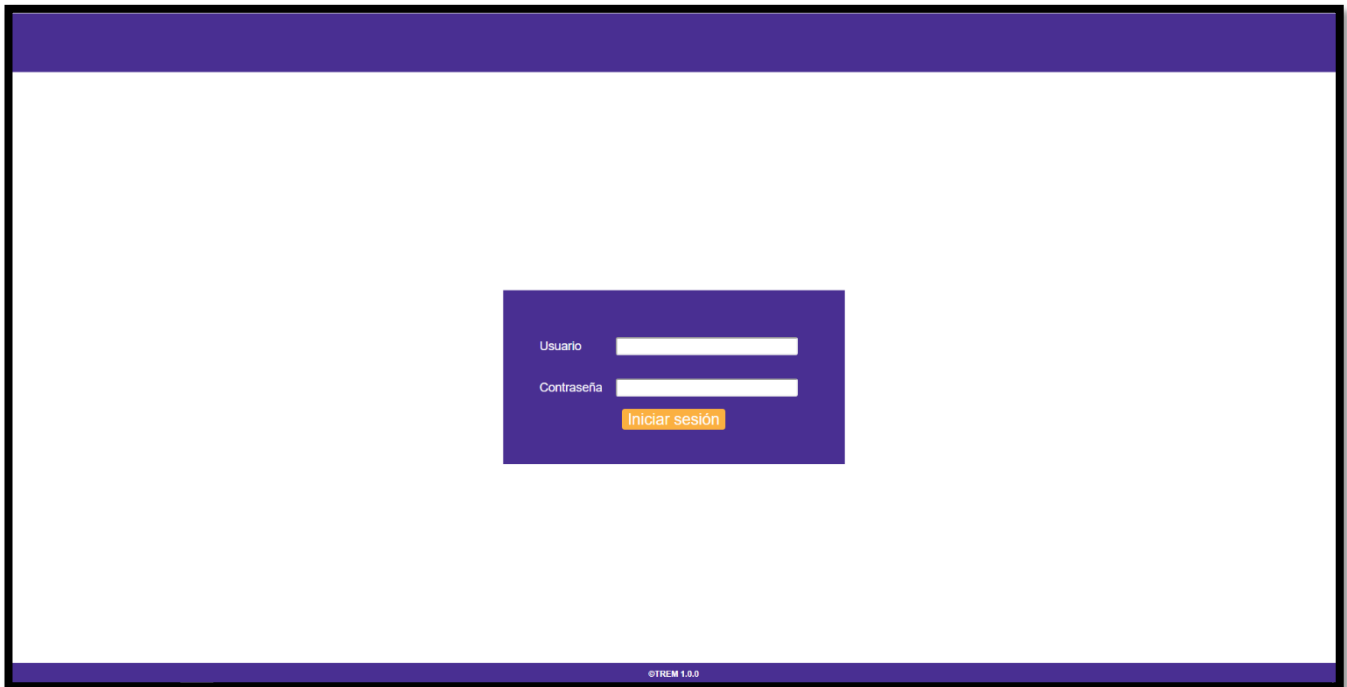


FIGURA 30. INTERFAZ LOGIN TRAS DESPLIEGUE

2. Uso de la aplicación

A continuación, se presenta la interfaz web de cada una de las páginas de las que dispone la aplicación. En primer lugar, se muestra la página de inicio o “login” en la cual introduciendo un username y password podremos acceder al interior de la aplicación:



The image shows a web application login page. It features a purple header bar at the top and a purple footer bar at the bottom. The footer contains the text "©TREM 1.0.0". In the center of the page, there is a white login form with a purple border. The form contains two input fields: "Usuario" and "Contraseña". Below the "Contraseña" field is a yellow button with the text "Iniciar sesión".

FIGURA 31. INTERFAZ LOGIN

Una vez hemos accedido a la aplicación web, se nos redirige por defecto a la página de fichaje en la cual se puede realizar el fichaje horario rellenando un formulario y pulsando en el botón “fichar”:

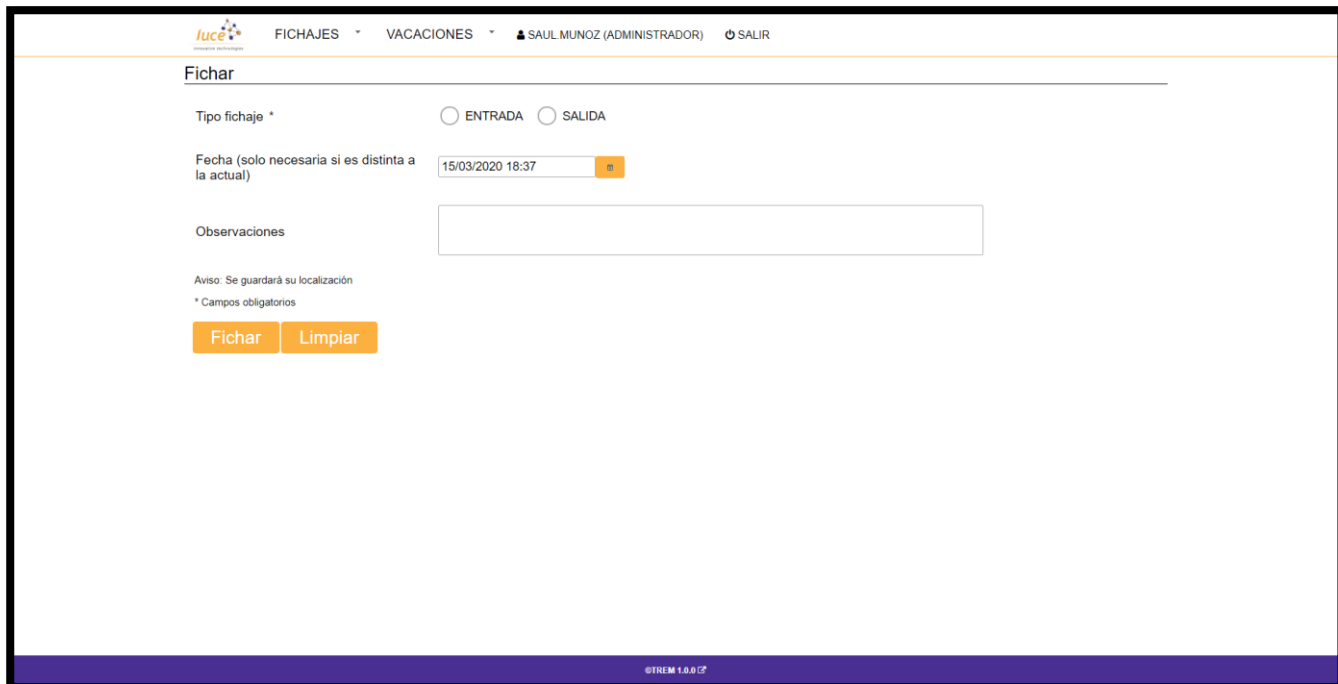


FIGURA 32. INTERFAZ FICHAJE

Una vez registrado nuestro fichaje horario, automáticamente se nos redirigirá a la pantalla de consulta de nuestros fichajes ya realizados, en la cual por defecto se nos filtran los fichajes de los últimos 7 días, pero en la que podemos filtrar a nuestro gusto abriendo el desplegable de los “criterios de búsqueda”:

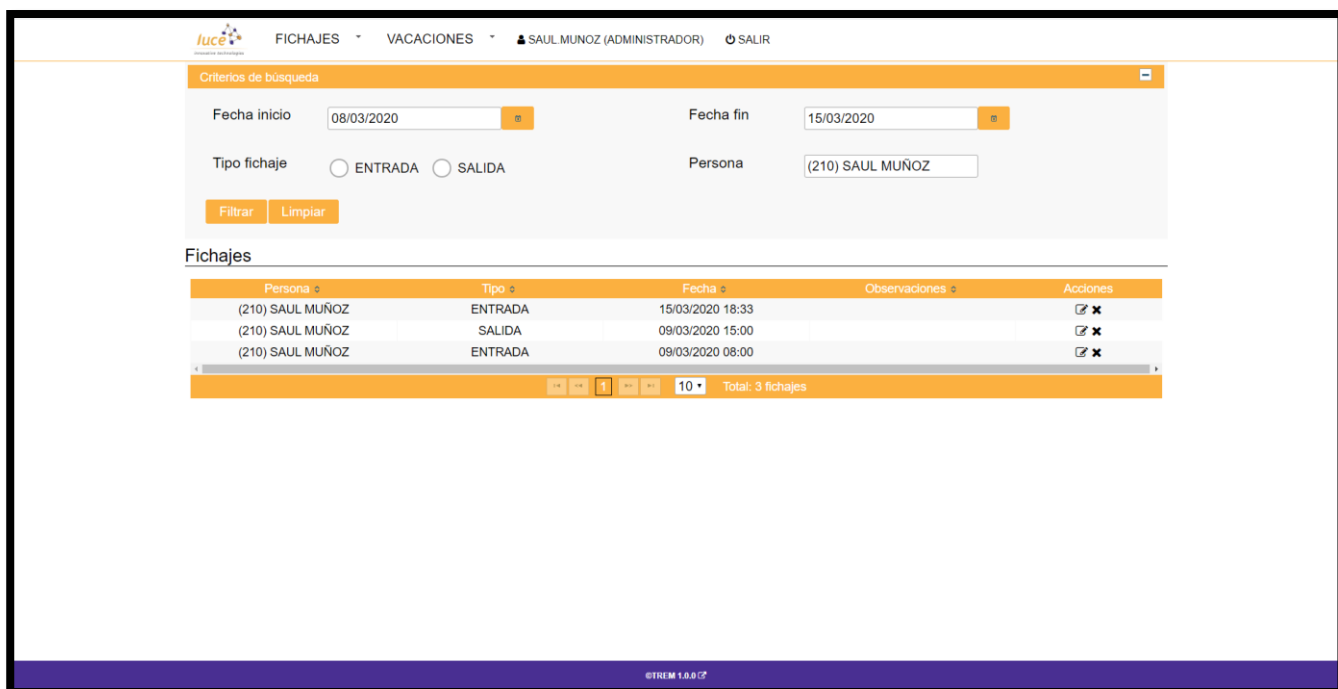


FIGURA 33. INTERFAZ LISTADO FICHAJES

Dentro del mismo menú de fichajes, podremos acceder a un informe de las horas registradas en los diferentes días y también como en la página anterior podremos realizar un filtrado a nuestro gusto:

Criterios de búsqueda

Fecha inicio: 02/03/2020 Fecha fin: 09/03/2020

Persona: (210) SAUL MUÑOZ

Filtrar Limpiar

Informe horas fichaje

Persona	Fecha	Horas	Observaciones	Acciones
(210) SAUL MUÑOZ	09/03/2020	7 h 0 m		✎
(210) SAUL MUÑOZ	06/03/2020	7 h 0 m		✎
(210) SAUL MUÑOZ	05/03/2020	7 h 0 m		✎
(210) SAUL MUÑOZ	04/03/2020	7 h 0 m		✎
(210) SAUL MUÑOZ	03/03/2020	9 h 34 m		✎
(210) SAUL MUÑOZ	02/03/2020	9 h 30 m		✎

Total: 47 h 4 m

© TREM 1.0.0

FIGURA 34. INTERFAZ INFORME HORAS REGISTRADAS

En cuanto al otro apartado del que dispone la aplicación, vacaciones, podemos encontrar dos páginas; la primera, se trata de un calendario en el que el usuario puede solicitar sus vacaciones y ver de manera gráfica en el mismo los estados en los que se encuentran las mismas:

Criterios de búsqueda

Fecha actual

Enero 2020

Lun	Mar	Mie	Jue	Vie
30	31	1	2	3
(210) SAUL MUÑOZ			(210) SAUL MUÑOZ	(210) SAUL MUÑOZ
6	7	8	9	10
		(210) SAUL MUÑOZ	(210) SAUL MUÑOZ	
13	14	15	16	17
20	21	22	23	24
(210) SAUL MUÑOZ	(210) SAUL MUÑOZ	(210) SAUL MUÑOZ	(210) SAUL MUÑOZ	(210) SAUL MUÑOZ
27	28	29	30	31
3	4	5	6	7
		(210) SAUL MUÑOZ		

© TREM 1.0.0

FIGURA 35. INTERFAZ CALENDARIO VACACIONES

En relación con la segunda página del menú de vacaciones, se trata de una página de consulta (similar a la de los fichajes registrados) de los periodos de vacaciones solicitados:

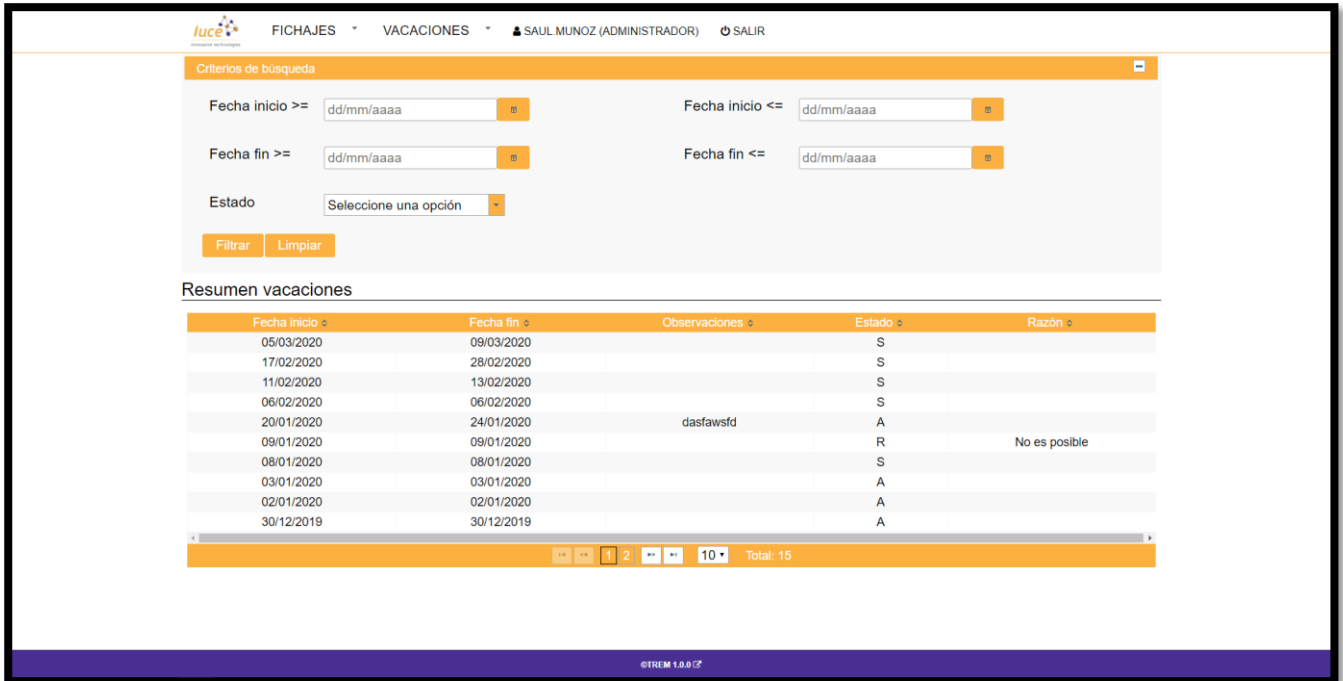


FIGURA 36. INTERFAZ LISTADO VACACIONES

Finalmente, la aplicación cuenta con una página denominada “changelog” en la cual se pueden consultar las actualizaciones o nuevas funcionalidades presentes en la aplicación según su versión y accesible desde el enlace que se encuentra en el footer de las páginas:

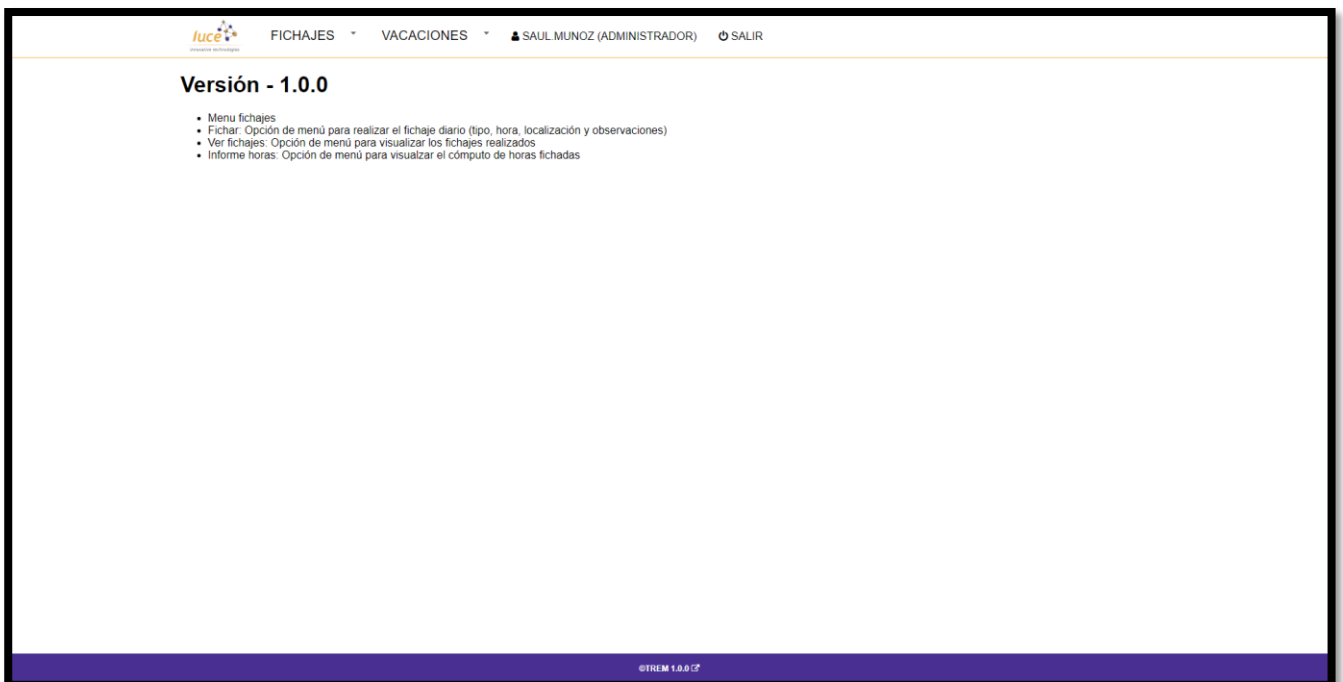


FIGURA 37. INTERFAZ CHANGELOG

A continuación, se presentan las interfaces de la web desde un dispositivo smartphone:



FIGURA 38. INTERFAZ LOGIN (SMARTPHONE)



FIGURA 39. INTERFAZ FICHAJE (SMARTPHONE)



FIGURA 40. INTERFAZ LISTADO FICHAJE (SMARTPHONE)

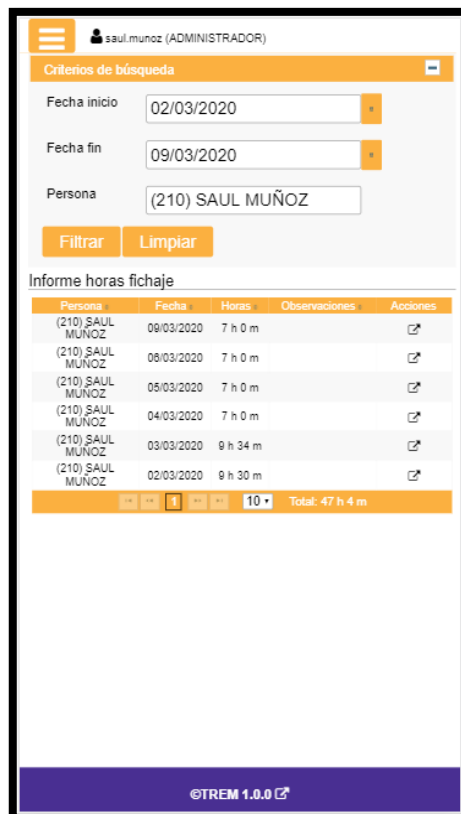


FIGURA 41. INTERFAZ INFORME HORAS REGISTRADAS (SMARTPHONE)

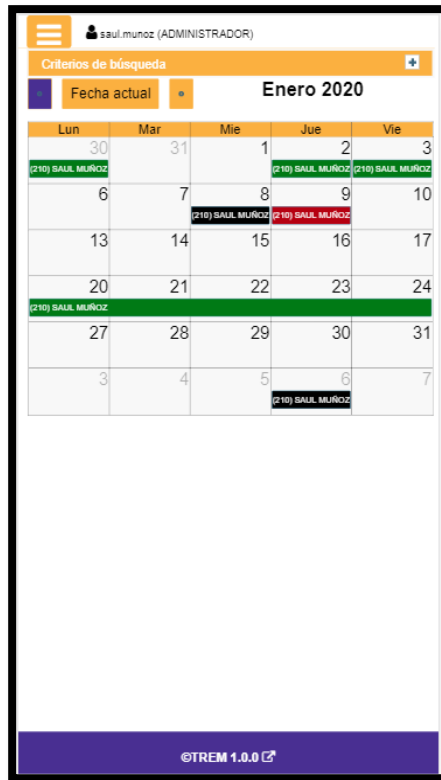


FIGURA 42. INTERFAZ CALENDARIO VACACIONES (SMARTPHONE)



FIGURA 43. INTERFAZ LISTADO VACACIONES (SMARTPHONE)

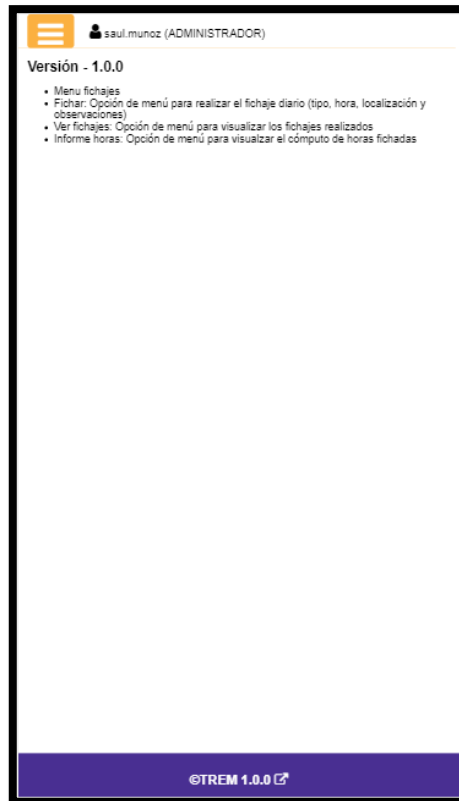


FIGURA 44. INTERFAZ CHANGELOG (SMARTPHONE)

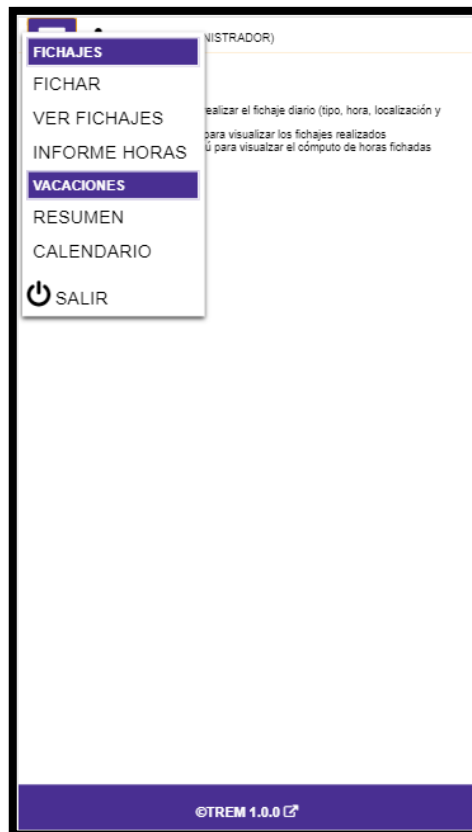


FIGURA 45. INTERFAZ ADAPTACIÓN MENÚ (SMARTPHONE)

ANEXO II

I. Contenido del CD

En el CD se encuentran los directorios que se describen a continuación:

1. TremProject

Carpeta que contiene el proyecto de IntelliJ. En ella podemos encontrar todo el código fuente desarrollado durante el proyecto.

2. Figuras

En esta carpeta se encuentran las figuras incluidas en la memoria en un tamaño mayor para facilitar su visualización.

3. Archivo de instalación

En este directorio se encuentra el archivo trem.war, es el archivo que se utiliza para instalar la aplicación.

4. Memoria

Contiene una copia en formato PDF de esta memoria.