

Plataforma web de inteligencia de negocio orientada a recursos humanos

Trabajo de Fin de Máster

MÁSTER UNIVERSITARIO EN INGENIERÍA INFORMÁTICA



**VNiVERSiDAD
D SALAMANCA**

Septiembre de 2021

Autor

D. Miguel Ángel Bernardo García

Tutores

Dña. Ana Belén Gil González

D. Laureano Ayuso Gómez

Dña. Ana Belén Gil González, profesora del Departamento de Informática y Automática de la Universidad de Salamanca, y D. Laureano Ayuso Gómez, jefe del departamento I+D de la empresa Aplifisa S.L.,

CERTIFICAN:

Que el trabajo titulado "Plataforma web de inteligencia de negocio orientada a recursos humanos" ha sido realizado por D. Miguel Ángel Bernardo García con DNI 70900407D y constituye la memoria del trabajo realizado para la superación de la asignatura Trabajo de Fin de Máster de la Titulación Máster Universitario en Ingeniería Informática de esta Universidad.

Y para que así conste a todos los efectos oportunos.

En Salamanca, a 3 de Septiembre de 2021

Dña. Ana Belén Gil González
Dpto. de Informática y Automática
Universidad de Salamanca

D. Laureano Ayuso Gómez
Departamento de I+D
Aplifisa S.L.

RESUMEN

La gestión de recursos humanos es una de las áreas que conforman la administración de empresas. Entre los procesos de recursos humanos, se incluye la gestión de la contratación, nóminas, relaciones laborales, control horario, evaluación de los empleados y el desarrollo de estrategias enfocadas a mejorar su desempeño. De la necesidad de evaluar a los empleados de cara a desarrollar estas estrategias y disponer de herramientas que faciliten los procesos de contratación, surge la idea de desarrollar una plataforma web.

Esta plataforma, desarrollada en un contexto profesional para Aplifisa, hace uso de la inteligencia de negocio para ayudar a las empresas a detectar problemas de retención de personal y absentismo y así las permita desarrollar estrategias para atajarlos. Además, permite gestionar los conocimientos de los empleados, facilitando el desarrollo de planes de formación, la detección de necesidades de personal y la gestión de proyectos desde el punto de vista de los recursos humanos.

El trabajo realizado durante el desarrollo de este proyecto, además, servirá de base a Aplifisa con el objetivo de crear nuevas líneas de producto capaces de profundizar en el análisis de indicadores clave de rendimiento como soporte a la toma de decisiones, así como en la gestión de los procesos de contratación en forma de sistema de seguimiento de candidatos.

Palabras clave: inteligencia de negocio, recursos humanos, indicadores clave de rendimiento, sistema de seguimiento de candidatos.

ABSTRACT

Human resources management is one of the areas that form part of the business administration. This area includes processes such as the management of contracts, payrolls, labour relations, worked hours control, performance assessment of the employees and the development of strategies focused on improve their performance. From the need of evaluating the employees' performance for developing these strategies and dispose tools that ease the contract processes, comes the idea of developing a web platform.

This platform, developed for Aplifisa in a professional context, makes use of business intelligence to help businesses to detect employee retention and absenteeism problems and thus they can develop strategies to tackle these problems. In addition, the platform allows to manage employees' knowledge, easing the development of formation plans, the detection of personnel needs and project management from a human resources perspective.

All the work done during the development of this project, moreover, is a base that allows Aplifisa to create new product lines capable of deepening into the key performance indicators analysis as support to decision making and, also, the management of hiring processes as an applicant tracking system.

Keywords: business intelligence, human resources, key performance indicators, applicant tracking system.

TABLA DE CONTENIDO

Resumen.....	i
Abstract	iii
Tabla de contenido.....	v
Índice de figuras	viii
Índice de tablas.....	x
1. Introducción	1
2. Motivación del proyecto	3
3. Estado del arte.....	5
3.1. Análisis de mercado.....	5
3.1.1. Análisis de la competencia	6
3.1.2. Análisis de otros productos similares	8
3.1.3. Resoluciones fruto del análisis	9
3.2. Sistema de inteligencia de negocio.....	11
3.2.1. Definición	11
3.2.2. Técnicas de inteligencia de negocio.....	11
3.2.3. Encaje del proyecto como sistema de inteligencia de negocio	13
4. Técnicas y herramientas	17
4.1. Lenguajes y frameworks	17
4.1.1. Back end	17
4.1.2. Front end.....	17
4.2. Entorno de desarrollo.....	19
4.2.1. Docker.....	19
4.2.2. Laradock	19
4.2.3. Visual Studio Code	20
4.2.4. MySQL Workbench	20
4.2.5. Deployer.....	20
4.2.6. Auditoría y pruebas.....	21
4.3. Entorno de pre-producción	21
5. Propuesta de solución	23
5.1. Metodología de trabajo.....	23

5.2.	Arquitectura del sistema de información	24
5.3.	Gestión de currículums	29
5.4.	Métodos de visualización de datos	33
5.5.	Indicadores clave de rendimiento.....	35
5.5.1.	Indicadores para la retención de personal	36
5.5.2.	Indicadores para el cumplimiento horario.....	38
5.5.3.	Indicadores para el análisis de competencias.....	40
5.6.	Proceso de diseño y desarrollo.....	40
6.	Resultados.....	43
6.1.	Resultados: gestión de currículums	46
6.2.	Resultados: análisis de indicadores	56
6.2.1.	Retención de personal.....	56
6.2.2.	Cumplimiento horario.....	59
6.2.3.	Competencias.....	63
6.2.4.	Generación de informes y navegabilidad entre KPI	65
6.3.	Métricas de rendimiento y accesibilidad.....	67
7.	Conclusiones y líneas de trabajo futuras	71
7.1.	Conclusiones.....	71
7.2.	Líneas de trabajo futuras.....	73
	Referencias	75
Anexo A.	Especificación de requisitos	81
A.1.	Participantes.....	81
A.2.	Catálogo de requisitos	83
A.2.1.	Requisitos generales del sistema.....	83
A.2.2.	Requisitos funcionales	84
A.2.3.	Requisitos de información	88
A.2.4.	Requisitos no funcionales.....	90
A.3.	Matriz de rastreabilidad.....	91
Anexo B.	Diseño del sistema	93
B.1.	Diseño de la interfaz.....	93
B.2.	Diseño software	103
B.2.1.	Definición de rutas y API	103

B.2.2.	Modelo de clases.....	105
B.2.3.	Comunicación entre vista y controlador.....	109
Anexo C.	Informes de auditoría.....	113

ÍNDICE DE FIGURAS

FIGURA 3.1: TÉCNICAS DE LA INTELIGENCIA DE NEGOCIO [14]	11
FIGURA 3.2: MÉTODO INMON (<i>TOP-DOWN</i>) [15]	12
FIGURA 3.3: MÉTODO KIMBALL (<i>BOTTOM-UP</i>) [15].....	12
FIGURA 4.1: COMPARATIVA ENTRE DOCKER Y MÁQUINAS VIRTUALES [25]	19
FIGURA 5.1: DISEÑO CONCEPTUAL DEL SISTEMA DE INFORMACIÓN	25
FIGURA 5.2: FUNCIONAMIENTO DEL MECANISMO DE TABLAS FEDERADAS [29]	26
FIGURA 5.3: MODELO ENTIDAD-RELACIÓN DE LA BASE DE DATOS.....	28
FIGURA 5.4: CONSULTA DE CREACIÓN DE LA TABLA FEDERADA	29
FIGURA 5.5: EJEMPLO DE LA MATRIZ DE COMPETENCIAS, APLICADO A UN GRUPO DE TRABAJADORES DE APLIFISA, REALIZADO DURANTE EL CURSO DE FORMACIÓN.	32
FIGURA 5.6: GRÁFICAS DISPONIBLES EN CHART.JS.....	33
FIGURA 6.1: MAPA DE NAVEGACIÓN	43
FIGURA 6.2: VISTA INICIAL.....	44
FIGURA 6.3: LISTADO DE TRABAJADORES DE LA EMPRESA	46
FIGURA 6.4: EJEMPLO DE FILTRO POR CADENA O PALABRA.....	47
FIGURA 6.5: EJEMPLO DE FILTRO RÁPIDO POR ESTADO DEL TRABAJADOR	47
FIGURA 6.6: VISTA INICIAL DE LA REDACCIÓN DEL CURRÍCULUM	48
FIGURA 6.7: MENSAJE DE ERROR AL ACCEDER AL CURRÍCULUM DE UN EMPLEADO QUE NO FORMA PARTE DE LA EMPRESA	48
FIGURA 6.8: AUTOCOMPLETADO DE PROVINCIA/MUNICIPIO.....	49
FIGURA 6.9: DETALLE DEL PROCESO DE AÑADIR LAS TITULACIONES OBTENIDAS Y LA EXPERIENCIA LABORAL.....	50
FIGURA 6.10: DETALLE DEL PROCESO DE AÑADIR COMPETENCIAS E IDIOMAS.....	50
FIGURA 6.11: GUARDAR EL CURRÍCULUM O VOLVER AL LISTADO DE TRABAJADORES	51
FIGURA 6.12: CURRÍCULUM GUARDADO CON ÉXITO	51
FIGURA 6.13: LECTURA DEL CURRÍCULUM.....	52
FIGURA 6.14: ELECCIÓN DE CENTRO Y DEPARTAMENTO.....	53
FIGURA 6.15: MUESTRA DE LA MATRIZ DE COMPETENCIAS	53
FIGURA 6.16: MATRIZ DE COMPETENCIAS UTILIZANDO CÓDIGO NUMÉRICO	54
FIGURA 6.17: FILTRADO POR TRABAJADOR EN LA MATRIZ DE COMPETENCIAS.....	54
FIGURA 6.18: FILTRADO POR COMPETENCIAS EN LA MATRIZ DE COMPETENCIAS.....	54
FIGURA 6.19: ORDENAR POR COMPETENCIAS DE TRABAJADOR Y REORDENAMIENTO DE TRABAJADORES.....	55
FIGURA 6.20: MATRIZ DE COMPETENCIAS EN FORMATO CSV ABIERTA DESDE MICROSOFT EXCEL	55
FIGURA 6.21: GRÁFICA DEL ÍNDICE DE ROTACIÓN DE PERSONAL TRIMESTRAL.....	56
FIGURA 6.22: GRÁFICA DEL ÍNDICE DE ROTACIÓN DE PERSONAL ANUAL DE LA EMPRESA Y LOS DEPARTAMENTOS DEL CENTRO SELECCIONADO	57
FIGURA 6.23: PERMANENCIA Y GRÁFICA DE PERMANENCIA POR GRUPOS DE AÑOS TRABAJADOS	58

FIGURA 6.24: GRÁFICA DE LA TASA DE ABSENTISMO POR CENTROS.....	60
FIGURA 6.25: ELECCIÓN DE CENTRO Y/O DEPARTAMENTO PARA VER EL ABSENTISMO DETALLADAMENTE.....	60
FIGURA 6.26: GRÁFICA DE LA TASA DE ABSENTISMO POR DEPARTAMENTOS EN EL CENTRO SELECCIONADO.....	60
FIGURA 6.27: GRÁFICAS DE ABSENTISMO EN UN DEPARTAMENTO	61
FIGURA 6.28: ALGORITMO PARA EL CÁLCULO DEL FACTOR BRADFORD.....	62
FIGURA 6.29: MENSAJE INDICANDO QUE LOS DATOS DE LA GRÁFICA SE ESTÁN CARGANDO	63
FIGURA 6.30: GRÁFICA DE COMPETENCIAS EN EL CENTRO	63
FIGURA 6.31: GRÁFICA DE COMPETENCIAS EN EL DEPARTAMENTO	64
FIGURA 6.32: GRÁFICA DE COMPETENCIAS DESGLOSADAS POR NIVEL.....	64
FIGURA 6.33: UTILIDADES EN LAS PÁGINAS DE ANÁLISIS DE INDICADORES.....	65
FIGURA 6.34: INFORME DE RETENCIÓN DE PERSONAL	66
FIGURA 6.35: AUDITORÍA DE CALIDAD REALIZADA CON GOOGLE LIGHTHOUSE SOBRE LA PÁGINA "VER TRABAJADORES"	67
FIGURA 6.36: ANÁLISIS DE ACCESIBILIDAD DE LA MATRIZ DE COMPETENCIAS CON "AXE DEVTOOLS"	68
FIGURA 6.37: CONVERSIÓN A ESCALA DE GRISES DE LA GRÁFICA DE PERMANENCIA POR GRUPOS DE AÑOS TRABAJADOS	69
FIGURA B.1: PROTOTIPO DE LA INTERFAZ PRINCIPAL	94
FIGURA B.2: PROTOTIPO DE LA INTERFAZ DEL LISTADO DE TRABAJADORES	95
FIGURA B.3: PROTOTIPO DE LA INTERFAZ DE REDACCIÓN DE CURRÍCULUMS	96
FIGURA B.4: PROTOTIPO DE LA INTERFAZ DE LA MATRIZ DE COMPETENCIAS.....	98
FIGURA B.5: PROTOTIPO DE LA INTERFAZ DE LOS INDICADORES DE RETENCIÓN DE PERSONAL ..	99
FIGURA B.6: PROTOTIPO DE LA INTERFAZ DE INDICADORES DE CUMPLIMIENTO HORARIO.....	101
FIGURA B.7: PROTOTIPO DE LA INTERFAZ DE INDICADORES DE COMPETENCIAS.....	102
FIGURA B.8: DEFINICIÓN DE RUTAS CON CLAUSURA Y CON MÉTODO DE CONTROLADOR.....	103
FIGURA B.9: MODELO FÍSICO DE LA BASE DE DATOS	105
FIGURA B.10: DIAGRAMA DE CLASES – CONTROLADORES.....	107
FIGURA B.11: DIAGRAMA DE CLASES – MODELOS	107
FIGURA B.12: DIAGRAMA DE CLASES – REPOSITORIOS	108
FIGURA B.13: DIAGRAMA DE CLASES – COMPONENTES.....	108
FIGURA B.14: DIAGRAMA DE CLASES – PETICIONES.....	108
FIGURA B.15: US-001 LISTA DE TRABAJADORES	109
FIGURA B.16: US-002 REDACTAR CV.....	109
FIGURA B.17: US-003 CONSULTAR CV	110
FIGURA B.18: US-004 CREAR MATRIZ DE COMPETENCIAS.....	110
FIGURA B.19: US-005 VER INDICADORES DE RETENCIÓN DE PERSONAL.....	110
FIGURA B.20: US-006 VER INDICADORES DE CUMPLIMIENTO HORARIO.....	111
FIGURA B.21: US-007 VER INDICADORES DE COMPETENCIA	111
FIGURA C.1: AUDITORÍA - PÁGINA PRINCIPAL.....	114
FIGURA C.2: AUDITORÍA - LISTADO DE TRABAJADORES	115

FIGURA C.3: AUDITORÍA - REDACCIÓN DE CURRÍCULUM.....	116
FIGURA C.4: AUDITORÍA - CONSULTA DE CURRÍCULUM	117
FIGURA C.5: AUDITORÍA - MATRIZ DE COMPETENCIAS	118
FIGURA C.6: AUDITORÍA - INDICADORES DE RETENCIÓN DE PERSONAL.....	119
FIGURA C.7: AUDITORÍA - INDICADORES DE CUMPLIMIENTO HORARIO.....	120
FIGURA C.8: AUDITORÍA - INDICADORES DE COMPETENCIAS.....	121

ÍNDICE DE TABLAS

TABLA 5.1: CALENDARIO DEL PROYECTO.....	24
TABLA 5.2: US-001 LISTAR TRABAJADORES.....	30
TABLA 5.3: US-002 REDACTAR CURRÍCULUM.....	31
TABLA 5.4: US-003 CONSULTAR CURRÍCULUM	31
TABLA 5.5: US-004 CREAR MATRIZ DE COMPETENCIAS.....	33
TABLA 5.6: COMPARATIVA DE LOS TIPOS DE GRÁFICAS.....	34
TABLA 5.7: US-005 VER INDICADORES DE RETENCIÓN DE PERSONAL.....	38
TABLA 5.8: US-006 VER INDICADORES DE CUMPLIMIENTO HORARIO.....	39
TABLA 5.9: US-007 VER INDICADORES DE COMPETENCIAS.....	40
TABLA A.1: PARTICIPANTE AUTOR.....	81
TABLA A.2: PARTICIPANTE TUTOR 1.....	82
TABLA A.3: PARTICIPANTE TUTOR 2.....	82
TABLA A.4: OBJ-001 GESTIÓN DE EMPLEADOS Y CURRÍCULUMS	83
TABLA A.5: OBJ-005 GESTIÓN DE COMPETENCIAS	83
TABLA A.6: OBJ-003 ANÁLISIS DE INDICADORES.....	83
TABLA A.7: US-001 LISTAR TRABAJADORES.....	84
TABLA A.8: US-002 REDACTAR CURRÍCULUM	85
TABLA A.9: US-003 CONSULTAR CURRÍCULUM	85
TABLA A.10: US-004 CREAR MATRIZ DE COMPETENCIAS.....	86
TABLA A.11: US-005 VER INDICADORES DE RETENCIÓN DE PERSONAL.....	86
TABLA A.12: US-006 VER INDICADORES DE CUMPLIMIENTO HORARIO.....	87
TABLA A.13: US-007 VER INDICADORES DE COMPETENCIAS.....	87
TABLA A.14: IRQ-001 EMPRESAS	88
TABLA A.15: IRQ-002 EMPLEADOS	88
TABLA A.16: IRQ-003 CURRÍCULUMS.....	89
TABLA A.17: IRQ-004 FICHAJES	89
TABLA A.18: NFR-001 SEGURIDAD	90
TABLA A.19: NFR-002 RENDIMIENTO	90
TABLA A.20: NFR-003 ACCESIBILIDAD.....	90

TABLA A.21: NFR-004 USABILIDAD.....	91
TABLA A.22: MATRIZ DE RASTREABILIDAD OBJ-US.....	91
TABLA B.1: RUTAS DE ACCESO A VISTAS.....	103
TABLA B.2: RUTAS DE LA API.....	104

1. INTRODUCCIÓN

La gestión de recursos humanos se define como el conjunto de procesos destinados a la administración de los trabajadores de la empresa. Estos procesos, entre otros, son la contratación y la gestión de las obligaciones que se generan fruto de este proceso (nóminas, cotizaciones sociales...), el control de los trabajadores (horas trabajadas, evaluación del rendimiento...) y el desarrollo de políticas de empresa respecto a los mismos como pueden ser las políticas para aumentar la motivación de los trabajadores [1].

Las aplicaciones *software* destinadas a ayudar a las empresas en la gestión de sus recursos humanos, entre las que se encuentran las aplicaciones desarrolladas por Aplifisa, la empresa beneficiaria del presente Trabajo Fin de Máster, suelen centrarse en los procedimientos más cotidianos como son el alta y baja de trabajadores en la empresa, el control horario o el cálculo de nóminas, cotizaciones sociales y otros impuestos y obligaciones como las retenciones.

Aprovechando esta situación del mercado, que tiende a ignorar los aspectos más estratégicos como son el desarrollo de políticas de empresa y la toma de decisiones, se ha decidido desarrollar una plataforma web que dé respuesta a este tipo de necesidades. Además, aplicar la inteligencia de negocio a esta plataforma permite contribuir a otorgar los medios necesarios para que la empresa usuaria pueda descubrir los problemas que tiene y el estado de su situación. Este estado de la situación de la empresa viene dado por una serie de indicadores denominados “indicadores clave de rendimiento”. El análisis de estos indicadores, facilitado por las técnicas propias de la inteligencia de negocio, pretende detectar y/o prever los problemas que la empresa tenga con el fin de diseñar y llevar a cabo estrategias y políticas capaces de solventarlos o impedir que lleguen a suceder.

Otro de los objetivos que se pretenden alcanzar es el de adquirir información relativa a los conocimientos de los trabajadores. Esta información es de interés como base para gestionar los procesos de contratación y gestionar mejor, desde el punto de vista de los recursos humanos, los proyectos.

Además de estos objetivos, propios del producto a desarrollar, se ha establecido un objetivo secundario: mejorar diversos aspectos de los procesos de desarrollo *software* realizados por Aplifisa. Para ello, se explorarán diversas técnicas y herramientas con el objetivo de optimizar estos procesos y mejorar la forma de trabajo de la empresa, así como evaluar posibles mejoras de la infraestructura *software*.

La presente memoria describe el proceso de ideación del producto: partiendo del análisis del mercado y el estudio de los sistemas de inteligencia de negocio, se

evaluará qué funcionalidades ofrecer y cómo diseñarlas, culminando este proceso en una versión de demostración que pueda optar a ser la versión inicial de este proyecto que se implante en Aplifisa.

Para describir este proceso, la memoria se ha estructurado de la siguiente manera:

1. Introducción: descripción de los objetivos a alcanzar.
2. Motivación del proyecto: explicación del origen del proyecto y su utilidad para la empresa.
3. Estado del arte: análisis de la situación en el mercado del *software* de gestión empresarial y del estado del marco teórico de los sistemas de inteligencia de negocio.
4. Técnicas y herramientas: descripción de los recursos empleados para realizar el proceso de desarrollo. Durante este apartado, además, se presentarán algunas de las herramientas con las que se pretende mejorar los procesos de desarrollo.
5. Propuesta de solución: explicación de la metodología, arquitectura del sistema de información y presentación de cómo alcanzar los objetivos.
6. Resultados: muestra del producto desarrollado y explicación tanto de su funcionamiento como de las decisiones realizadas para obtener el resultado.
7. Conclusiones y líneas de trabajo futuras: análisis del resultado, el cumplimiento de objetivos y la presentación de las diversas líneas de trabajo por las que puede seguir evolucionando el proyecto.

Como complemento a la información contenida en la memoria, se incluyen los siguientes anexos:

- Anexo A: Especificación de requisitos.
- Anexo B: Diseño del sistema.
- Anexo C: Informes de auditoría.

2. MOTIVACIÓN DEL PROYECTO

Aplifisa es una empresa salmantina de ámbito nacional nacida en el año 1990 dedicada al desarrollo de aplicaciones informáticas para asesorías y empresas. Estas aplicaciones están enfocadas a la gestión fiscal y contable de las empresas, así como de las relaciones laborales con sus empleados en cuanto a nóminas y cotizaciones sociales.

Desde hace unos años, además de mejorar las aplicaciones ya existentes, Aplifisa se ha enfocado en la gestión laboral añadiendo a su oferta de productos aplicaciones capaces de llevar el control horario y las horas trabajadas de los empleados. Dicho control horario, además, es obligatorio en España desde el año 2019 [2]. Todo este *software* entra dentro del área empresarial de la gestión de recursos humanos.

En esta línea, para el proyecto, la empresa pretendía desarrollar un producto capaz de gestionar los currículums de los trabajadores para poder controlar su formación, experiencia y los cursos a los que, tanto fuera como dentro de la empresa, han asistido.

El área de gestión de recursos humanos, por lo general, las empresas competencia de Aplifisa y la propia Aplifisa la ven desde una perspectiva económica entre empresa y trabajador: las empresas tienen trabajadores y establecen con ellos relaciones contractuales a través de contratos, donde se detallan salarios, horas semanales a realizar e implican pagos (nóminas y cotizaciones a la Seguridad Social). Una perspectiva más ignorada en las soluciones informáticas que estas empresas ofrecen es la de las personas y cómo estas afectan a la empresa.

Fruto de esta visión, se pensó en, más que crear un gestor de currículums (que encajaría más en el apartado de gestión documental), desarrollar una herramienta que permita analizar los efectos que las personas tienen en la empresa y cómo optimizar diversos aspectos de la gestión empresarial respecto a los recursos humanos.

Además, se consideró beneficioso para Aplifisa marcar como objetivo el desarrollo del proyecto como un sistema de inteligencia de negocio: por una parte permitiría modernizar y paliar diversos problemas de la empresa (entre otros, trabajar al día en función de los problemas y necesidades que surgen y no planificar más a largo plazo, sin desarrollar una infraestructura que dé soporte y facilite el proceso de desarrollo y la adaptación al cambio) y, por otra, conseguir un producto más atractivo y que sea de mayor utilidad para los clientes.

Este nuevo proyecto encaja en la misión de Aplifisa: acercar la tecnología a los usuarios ofreciendo funcionalidades novedosas y que pueden estar presentes en programas a los que no podrían acceder. Esto además permitiría a Aplifisa abrirse a nuevos mercados y a nuevos tipos de clientes, en especial a empresas de mayor tamaño. Dentro del acercamiento de la tecnología a los usuarios, el proyecto también permitiría facilitar la labor de su trabajo a los clientes, simplificando algunas tareas que realizan de manera manual o que, por su naturaleza, sean difíciles de realizar manualmente.

3. ESTADO DEL ARTE

Este apartado, al ser parte de un Trabajo Fin de Máster dirigido al desarrollo de un producto para una empresa, se abordará mediante un análisis de mercado enfocado a analizar la competencia y productos que, no siendo de la competencia, tienen relevancia para el proyecto. Por último, se analizarán las herramientas propias de los sistemas de inteligencia de negocio.

3.1. Análisis de mercado

El análisis de mercado se abordará desde dos puntos de vista: empresas competencia de Aplifisa (competencia directa) y empresas que desarrollan herramientas *software* de inteligencia de negocio aplicada a la gestión de recursos humanos (competencia indirecta). De este modo, se pretende:

- Analizar la viabilidad del proyecto, es decir, la posibilidad de que, en el mercado al que se dirige Aplifisa, el producto *software* a desarrollar tenga acogida. Para ello, se debe analizar la competencia de la empresa y ver si hay soluciones similares y, en caso de haberlas, de qué forma diferenciarse de ellas.
- Analizar productos y su mercado objetivo que, no siendo de empresas realmente competidoras, puedan afectar a la viabilidad del presente proyecto. Una vez sabido si la empresa sería capaz de vender el producto respecto a la competencia directa, hay que saber si el *software* a desarrollar colisionaría en el mercado con otros productos fuera de la competencia normal de Aplifisa y, en su caso, evaluar los problemas que conlleva este hecho y cómo solventarlos.
- Adicionalmente, analizar otras herramientas de inteligencia de negocio ayudará a la hora de enfocar el producto. Palabras grandilocuentes como “inteligencia de negocio” o “gestión de recursos humanos” no significan nada si no se es capaz de concretar qué utilidad tiene el producto ni de qué manera son útiles. En este sentido, analizar los productos ya existentes, estén o no en la competencia directa, ayudará a la hora de definir el producto a desarrollar.

A modo de resumen, se podría decir que el objetivo final de este apartado es conocer si nos enfrentamos a un océano rojo o a un océano azul [3] y cómo se va a hacer: saber si el producto desarrollado va a competir, en un fiero océano rojo de sangre, superando a los muchos rivales compitiendo en precios/costes o si, por el contrario, se va a competir buscando un océano azul, inexplorado y con nuevas oportunidades, creando nuevas oportunidades de negocio.

3.1.1. Análisis de la competencia

Aplifisa se dedica al desarrollo de *software* para asesorías y empresas, centrándose en la parte financiera (facturación, contabilidad, gestión fiscal...). En lo que a este proyecto es relevante, el *software* de gestión de recursos humanos que desarrolla también se centra en el plano económico (nóminas y contratos) y, recientemente, también en el control horario de los trabajadores [4].

Teniendo esto en cuenta, para el análisis hay que fijarse en empresas que desarrollen *software* para asesorías y empresas y ver qué servicios relativos a la inteligencia de negocio y/o relativos a la gestión de recursos humanos ofrecen. De estas empresas y productos, se han identificado como relevantes a la hora de analizar NCS Software, Wolters Kluwer, Next Generation Fintech Services y Sage. Se podrían haber utilizado otras empresas y productos (ContaNet, Datisa, Anfix, Altai...), pero el análisis hecho con NCS se puede aplicar a estos.

Los servicios que ofrecen Wolters Kluwer y NCS Software, en esencia, son bastante similares a los que desarrolla Aplifisa (de hecho, NCS ofrece prácticamente los mismos servicios que ofrece Aplifisa [5]). Entre ellas se observa que:

- Las tres empresas ofertan servicios de control/registro horario.
- Por lo general, los productos de Wolters Kluwer ofrecen servicios con más prestaciones. Por ejemplo, oferta productos compatibles con Microsoft Office o sistemas de recomendación en función de los datos de los declarantes útiles para la asesoría fiscal.
- Wolters Kluwer dispone de un sistema de seguimiento de candidatos (incluido en su producto a3innuva [6]) del que no disponen las otras dos. Este sistema está enfocado meramente al proceso de selección, incluye un sistema de filtrado de candidatos, y no aplica el conocimiento que obtiene a otros aspectos de la gestión de recursos humanos como puede ser la gestión de proyectos.
- Wolters Kluwer ofrece herramientas para gestionar la evaluación del personal mediante la creación de formularios y encuestas, a diferencia de Aplifisa y NCS.
- Ninguna de las tres compañías indica que su *software* entre en la categoría de inteligencia de negocio.

Fuera de los servicios ofertados por cada una, mientras que Aplifisa y NCS son empresas que pertenecen a la categoría de pequeña y mediana empresa, Wolters Kluwer es una multinacional. Además de en los productos, esto también se ve reflejado en el mercado objetivo de cada una: los clientes de

Aplifisa son pequeñas y medianas empresas y Wolters Kluwer tiene, además, grandes empresas como clientes.

Por otro lado, están las empresas Next Generation Fintech Services y Sage (multinacional británica). A diferencia de las otras dos, estas sí que o bien especifican que tienen productos que hacen uso de la inteligencia de negocio (Sage) o bien usan servicios externos para ofrecer herramientas típicas de la inteligencia de negocio.

Next ofrece next-bs. Este producto es una solución *software* que hace uso de los servicios de Microsoft y se adapta al negocio del cliente (empresa final de cualquier tamaño, especialmente pequeña y mediana empresa, o asesoría) para dar soluciones en cuanto a administración contable, gestión de proyectos, gestión de finanzas (información financiera, presupuestos, ventas, etc.) y gestión de almacenes y stocks [7, 8]. Aparte de la gestión de proyectos, no tiene nada más relativo a la gestión de recursos humanos.

Lo que más afecta al presente proyecto es lo relativo a los servicios de Microsoft utilizados. Next utiliza Microsoft Azure, en concreto y a juzgar por la apariencia visual del producto, Azure Portal para crear portales de gestión individualizados para las empresas y, con más repercusión para el proyecto, Power BI para la generación de informes personalizables. Power BI sería interesante como añadido al proyecto de cara al futuro: si bien integrado con Azure (servicio que no utiliza Aplifisa actualmente) sería lo idóneo, sí que, una vez el proyecto esté en fases más avanzadas, podría ser una integración interesante. Fuera del uso Power BI como parte de una estructura de un sistema de inteligencia de negocio, en principio, no debería ser un competidor que afecte al proyecto.

Por último, está Sage. Sage, como el resto de empresas de la competencia, ofrece servicios de gestión de recursos empresariales (finanzas, contabilidad, nóminas...) y, de manera similar a los productos de Wolters Kluwer, tiene un apartado de gestión de recursos humanos. Este sistema de gestión de recursos humanos incluye también la gestión de objetivos y un sistema de seguimiento de candidatos estrictamente enfocado al reclutamiento, aunque más simple que el de Wolters Kluwer. En cuanto a inteligencia de negocio, la aplica a la administración financiera (contabilidad y finanzas), cadena de suministro (compras, ventas y gestión de almacén) y gestión de producción (calidad de producto y planificación de la producción), pero no a la gestión de recursos humanos [9].

3.1.2. Análisis de otros productos similares

Dentro del *software* de gestión de recursos humanos, fuera del *software* típico que ofrece la competencia de Aplifisa (gestión laboral en cuanto a nóminas y contratos), están:

- Las herramientas de análisis de indicadores. Estos indicadores permiten analizar aspectos como la retención de personal, la satisfacción de los trabajadores, la productividad o los costes asociados a la contratación y los empleados (duración de los procesos de selección, entrenamiento de nuevos trabajadores...). Estas herramientas son utilizadas para detectar problemas y como ayuda a la hora de tomar decisiones.
- Los sistemas de seguimiento de candidatos (ATS, del inglés *Applicant Tracking System*). Son herramientas para gestionar los procesos de selección de personal, facilitando la contratación y ahorrando costes.

La parte de análisis de indicadores se puede hacer dentro de una estructura de inteligencia de negocio. En este sentido se puede hablar de empresas/productos como Power BI o Qlik, enfocados a la gestión de datos y su análisis, o SAP, que es una empresa que ofrece soluciones informáticas a empresas. Las herramientas como Power BI hacen uso de los datos de la empresa para su análisis en cualquier aspecto, mientras que SAP se centra en dar soluciones directas y adaptables a las empresas.

Las soluciones de SAP son muy similares en cuanto a funcionalidad a las que ofrecen Sage o Wolters Kluwer, aunque más desde una perspectiva general. Lo importante de SAP es que sus productos se adaptan a las empresas: de manera similar a como hace Next, los clientes (y no SAP directamente, a diferencia de Next) pueden, o bien ellos directamente o a través de consultores especializados en SAP, modificar y adaptar el código de SAP (es código abierto) a las necesidades propias de la empresa. Llegar a competir directamente con SAP es difícil, principalmente por su tamaño como empresa, el modelo de negocio y el tipo de empresas que suelen contratar sus servicios, pero sí que es útil para conocer modelos de negocio alternativos.

Los sistemas de seguimiento de candidatos son herramientas enfocadas a gestionar todo el proceso de reclutamiento de la empresa, desde la recepción de currículums hasta la contratación. De las empresas de la lista Fortune 500 (las quinientas empresas con mayor volumen de ventas de Estados Unidos), este tipo de sistemas son utilizados por, según datos de 2019, más del 95% de ellas. En cuanto a las propias herramientas, las más utilizadas por las empresas de Fortune 500 son Workday, Taleo (Oracle) y SuccessFactors (SAP) [10].

De los ATS, más que entrar en detalle de los productos existentes y las empresas que los ofrecen, lo más interesante es analizar cómo trabajan, en concreto, a la hora de clasificar las solicitudes de los aspirantes: mientras que la gestión laboral es algo que ya hacen Aplifisa y su competencia y que el análisis de indicadores es una tarea de gestión de datos y visualización de los mismos (tarea más estándar), la clasificación de currículums no es trivial. Taleo, aunque el funcionamiento es válido para todos los ATS, clasifica los currículums utilizando tareas de descarte y de análisis [11].

Para el descarte de currículums utiliza preguntas y respuestas (“¿tienes carnet de conducir?”) y, en función de la respuesta, valida (respuesta “sí”) o descarta (respuesta “no”) los currículums. Otra de las formas de filtrar (con el fin de descartar) currículums es utilizando búsquedas booleanas del tipo “(es ingeniero informático O de grado superior) Y sabe Python” o “es ingeniero informático Y NO de grado superior”, que es un método análogo al anterior pero más complejo.

El análisis de currículums sirve para otorgar una puntuación a los candidatos, esta puntuación permitirá ordenar la lista de candidatos de cara a seleccionar a los mejores currículums aceptados para entrevistarlos. Este análisis de currículums hace uso de técnicas de procesamiento de lenguaje natural y puntúa analizando la similitud entre el resumen del currículum y la oferta (porcentaje de similitud) o bien según la aparición o no de palabras clave puntuadas (calificación en función de los puntos que otorgue la presencia de cada palabra clave). Para estas tareas se podrían utilizar bibliotecas como Natural Language Toolkit [12] o Gensim [13] bajo el lenguaje de programación Python.

3.1.3. Resoluciones fruto del análisis

Como se explica en el análisis de la competencia, hay muchas empresas dedicadas al desarrollo de *software* de gestión para asesorías y empresas. Siguiendo la teoría del océano azul, en la actualidad, Aplifisa (y sus productos) está en un océano rojo y busca aproximarse a uno más azul dando un servicio de atención al cliente individualizado y personalizado que tiene una alta valoración por parte de los clientes.

En cuanto al apartado de gestión de recursos humanos la mayoría de empresas, a excepción de empresas como Sage o Wolters Kluwer, se enfoca en una vista financiera. Ofreciendo un producto que se centre en el rendimiento de los empleados, otros indicadores clave de rendimiento y la gestión de los procesos de contratación Aplifisa podría diferenciarse de la competencia estándar.

Al evaluar el *software* de gestión de recursos humanos de empresas más grandes como Wolters Kluwer y Sage se observa que hay módulos en los que no hay conexión: tienen por un lado el análisis de indicadores de rendimiento y la gestión de proyectos y por otro lado, completamente separado, el sistema de seguimiento de candidatos. En este sentido, se pueden cruzar los datos de los diversos productos y crear nuevas funcionalidades: por ejemplo, se puede añadir a la gestión temporal y de los recursos de los proyectos una visión del conocimiento de los empleados para poder crear equipos de trabajo más optimizados; siguiendo en esta línea, datos de productos actuales de Aplifisa también se podrían utilizar para los indicadores clave de rendimiento y mejorar las funcionalidades existentes en El Portal del Empleado. Por lo tanto, a pesar de la dificultad de competir con estas empresas, sí que hay una vía para poder distinguirse de ellos y navegar en ese océano azul.

Añadir una arquitectura de inteligencia de negocio también facilitaría dicho cruce de información, así como mejoraría la función de ayuda a la toma de decisiones que representa el análisis de los indicadores clave de rendimiento.

Por lo tanto y en resumen, se obtienen las siguientes conclusiones:

- Para salirse de lo normal en la competencia directa y que el producto sea viable, se deben abarcar nuevas áreas como el análisis de indicadores o los sistemas de seguimiento de candidatos.
- Cruzar datos entre aplicaciones y utilidades para crear nuevas funcionalidades diferenciaría al presente proyecto del resto de programas de la competencia más fuerte.
- No se trata de competir directamente con la competencia más fuerte, sino de ofrecer nuevas funcionalidades a los clientes actuales de Aplifisa y atraer a nuevos clientes que no puedan permitirse los programas de la competencia.
- Entender el producto a desarrollar como un sistema de negocio permite potenciar las funcionalidades que ofrezca y facilitará la implementación de determinados aspectos.

3.2. Sistema de inteligencia de negocio

3.2.1. Definición

A la hora de desarrollar un sistema de inteligencia de negocio, hay que conocer qué es la inteligencia de negocio y qué técnicas comprende.

La inteligencia de negocio se define como un “conjunto de procesos, técnicas y herramientas de ayuda a las decisiones de negocio basadas en la tecnología de la información” [14]. Por lo tanto, son sistemas que, en un contexto empresarial, ayudan a los directivos o responsables de sección a tomar decisiones. Estas decisiones pueden ser a nivel de operaciones y transacciones (actividades diarias/rutinarias, con procedimientos bien definidos), operativo (decisiones a corto plazo con consecuencias limitadas como la gestión de *stocks*), táctico (decisiones a medio plazo para mejorar el rendimiento de la empresa) y estratégico (definen las líneas maestras de la estrategia de la empresa).

3.2.2. Técnicas de inteligencia de negocio

Las técnicas empleadas en los sistemas de negocio se pueden representar en forma de pirámide según el potencial de soporte a las decisiones, siendo el soporte a la toma de decisiones el último nivel de la pirámide. En la siguiente figura se muestran dichas técnicas:

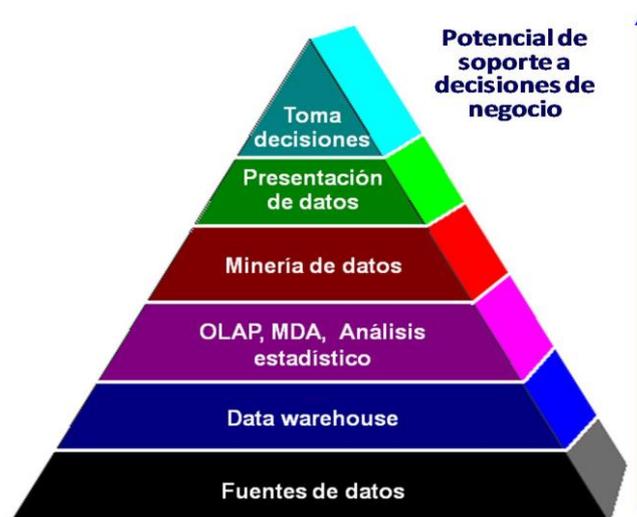


Figura 3.1: Técnicas de la inteligencia de negocio [14]

En el primer nivel se encuentran las fuentes de datos. Estos datos son heterogéneos (proceden de bases de datos de programas fuera del sistema de información de la aplicación, aplicaciones externas a la empresa, documentos...) y se requiere su extracción, transformación y carga para su uso (conocidos por sus

siglas en inglés como procesos ETL: *extract, transform, load*). Dada su heterogeneidad y cantidad (son datos en bruto), no ofrecen excesivo soporte a la toma de decisiones.

En el segundo nivel está lo que se considera como el sistema de información: el almacén de datos (del inglés *data warehouse*). Como su nombre indica, es donde se almacenan los datos que se emplean en el sistema; estos datos son datos homogéneos y proceden bien del uso de la aplicación por parte del usuario (a través de una interfaz gráfica) o bien de los datos transformados procedentes de las mencionadas en el párrafo anterior.

A la hora de implementar el almacén de datos, hay dos grandes arquitecturas en función de cómo se almacena la información en el almacén: de arriba a abajo (*top-down*, método Inmon) y de abajo a arriba (*bottom-up*, método Kimball) [15]:

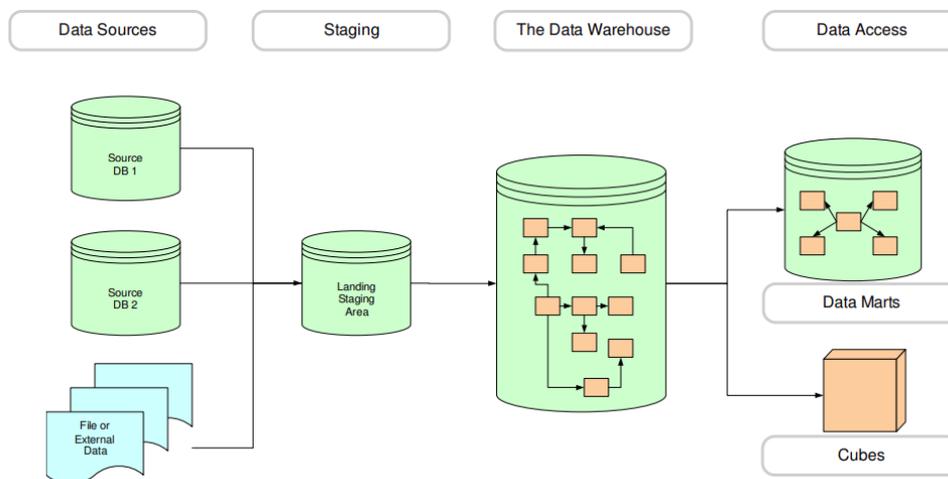


Figura 3.2: Método Inmon (*top-down*) [15]

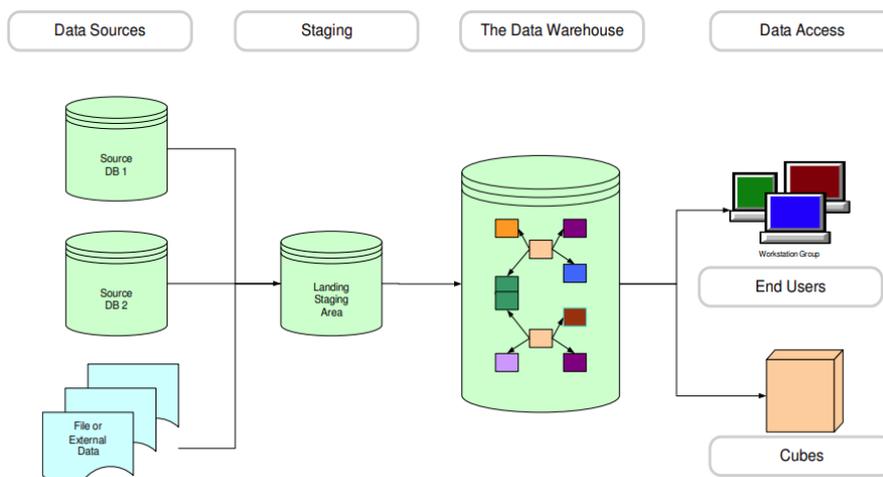


Figura 3.3: Método Kimball (*bottom-up*) [15]

En el método Inmon los datos obtenidos se almacenan en el *data warehouse* y del almacén pueden ser derivados a *data marts* secundarios en función del uso que se vaya a dar a los datos (por ejemplo, en función de los departamentos de la empresa). En el método Kimball los datos se almacenan en los *data marts* y estos constituyen el almacén de datos, pudiéndose agregar utilizando un esquema en estrella.

En el tercer nivel se encuentran herramientas de análisis y exploración de los datos. Estas herramientas permiten obtener (de manera optimizada) y analizar los datos y representar métricas, pudiendo para ello representarse los datos de manera multidimensional (varias dimensiones como costes, departamentos, etcétera en la misma representación) en vez de bidimensional (filas con las tuplas de información y columnas representando los atributos).

El cuarto nivel lo ocupan las técnicas de minería de datos. La minería de datos permite ver relaciones y encontrar patrones en los datos que, mirando los datos en bruto y/o resumidos, no son fáciles de encontrar de manera manual. Son técnicas que descubren conocimiento: no demuestran ni validan ideas sino que, dados los datos, son capaces de encontrar las ideas, las relaciones que hay entre los datos y permiten explotar dicho conocimiento.

Por último, en el quinto nivel se encuentra la presentación de datos. Este nivel sirve de apoyo a los niveles anteriores, pues permite ver mejor la información que se obtiene de estos niveles e identificar con más facilidad las relaciones presentes en los datos y otra información útil.

Dentro de la inteligencia de negocio, de manera adicional se pueden añadir otras herramientas tales como la generación de informes (reportes de la información para su consulta o presentación a terceros y otros niveles/sectores de la organización) o los sistemas de alertas (controlar y notificar cuando determinados parámetros excedan un determinado valor).

3.2.3. Encaje del proyecto como sistema de inteligencia de negocio

En un sistema como el propuesto, para la empresa Aplifisa, se tienen:

- Datos de las aplicaciones propias de escritorio, aplicaciones como Contabilidad o Gestión Laboral. Además, las bases de datos que los contienen tienen fallos de diseño de las tablas (principalmente asociados a la baja normalización de las bases de datos).

- Datos de aplicaciones externas a la empresa. Dentro de esta categoría se tendrían, por ejemplo, currículums de trabajadores o candidatos disponibles en portales de búsqueda de empleo o redes sociales profesionales.
- Datos de otras aplicaciones web y móviles propias, aplicaciones como El Portal del Empleado (versión web y versión móvil) o Mi Centro de Fichajes.
- Los propios datos del producto desarrollado en este proyecto una vez esté en explotación.

De estos cuatro tipos de datos, los dos primeros son de tipo heterogéneo: son datos procedentes de diversas fuentes (cada cliente tiene sus propios datos en las aplicaciones de escritorio instaladas en los ordenadores de su empresa) y/o están en diversos formatos (cada portal tiene su forma de ofrecer los currículums). Dada su heterogeneidad, son datos sobre los que habría que realizar operaciones ETL para su uso.

Una vez hechas estas consideraciones y teniendo en cuenta que, hasta la fecha, el sistema de información de Aplifisa está formado por una base de datos relacional (MySQL) por aplicación (no hay un sistema de información constituido como tal), se puede empezar a visualizar la forma del sistema de información a desarrollar. Lo que se estima más adecuado en costes es el aprovechamiento de la arquitectura actual para seguir un esquema *bottom-up* en el que:

- Los datos de las aplicaciones de escritorio, una vez cargados, se almacenen en la base de datos de la aplicación web con la que los datos tengan relación.
- Los datos del resto de aplicaciones web se almacenan en su propia base de datos.
- Los datos de aplicaciones externas y los que el producto a desarrollar genere se almacenen en una base de datos propia del producto desarrollado.
- Conceptualmente, cada base de datos en los diversos servidores utilizados por las aplicaciones web se puede entender como un *data mart*.

Dado el estado del proyecto y a las herramientas utilizadas actualmente por Aplifisa, no se considera necesario el uso de una herramienta especializada de *data warehousing* ni de herramientas de análisis y exploración de datos: hasta que el proyecto no esté en un estado avanzado, implementar una herramienta de análisis y exploración de datos como OLAP (del inglés *online analytical processing*) que haga uso de una herramienta de *data warehousing* es demasiado costoso, pues se podría tener pocos datos y no poder aprovechar el potencial que se puede extraer de OLAP y del *data warehousing*.

A pesar de este aspecto:

- La arquitectura del sistema de información escogida es fácil de evolucionar: al tener implementados los *data mart*, a mayores introducir en la arquitectura una herramienta de *data warehousing* que los agregue y nutrir al almacén de datos es trivial.
- Las herramientas de análisis y exploración de datos no dejan de ser un añadido que accede a los datos del almacén, siendo igualmente sencillo introducirlos en la arquitectura.
- Hay productos como Apache Kylin que proveen de funcionalidad OLAP y son compatibles con múltiples sistemas de almacenamiento (Hadoop, Hive, MySQL...), lo que facilita más añadir OLAP.

En definitiva, sería una arquitectura que hace uso de la infraestructura ya existente en Aplifisa, permitiendo reducir costes y con la capacidad de evolucionar.

El último aspecto a considerar es el uso de la minería de datos. Al igual que sucede con las herramientas de análisis y exploración de datos, en principio, habrá pocos datos, problema que se suma a la heterogeneidad de los clientes: hay clientes asesoría y hay clientes empresa, y dentro de los clientes empresa, no todas las empresas se dedican al mismo sector ni son del mismo tamaño. Estos hechos hacen difícil añadir la minería de datos al sistema de inteligencia de negocio por lo que, una vez el producto esté en explotación, habría que realizar pruebas para ver de qué manera y qué funcionalidades ofrecer haciendo uso de minería de datos.

Por lo tanto, y como conclusión, hacer uso de un sistema de negocio en el proyecto tiene las siguientes ventajas:

- Permite dotar a Aplifisa de una infraestructura reconocible, modernizando la empresa.
- Facilita y sirve de base a líneas de producto nuevas enfocadas a la toma de decisiones.
- Aunque el desarrollo del proyecto en la actualidad no contemple usar todo el potencial de la inteligencia de negocio sí que, a futuro, se puede llegar a aprovechar en mayor medida dicho potencial.
- El uso de la inteligencia de negocio está en línea con el propósito: ayudar a los procesos de toma de decisiones, en este caso, respecto a la gestión de recursos humanos.

4. TÉCNICAS Y HERRAMIENTAS

Todas las herramientas utilizadas o bien son de código abierto o bien se distribuyen bajo licencia MIT, lo que permite ahorrar en costes a la empresa.

4.1. Lenguajes y frameworks

4.1.1. Back end

Por restricciones de la empresa, se ha utilizado el *framework* Laravel (lenguaje de programación PHP).

4.1.1.1. PHP

Es un lenguaje de código abierto, de propósito general e interpretado con soporte a la programación orientada a objetos. El código PHP se puede ejecutar a través de un intérprete, dentro de un servidor web o bien incrustado en ficheros HTML [16]. Principalmente se utiliza en desarrollo web.

4.1.1.2. Laravel

Laravel es un entorno de trabajo (*framework*) PHP para el desarrollo web; pensado especialmente para la parte servidor, ofrece soporte también a la parte cliente mediante Laravel Blade y, por lo tanto, es un *framework* web *full-stack* [17]. Se caracteriza por ser:

- Progresivo: tiene una curva de aprendizaje aceptable para nuevos desarrolladores.
- Escalable: usa PHP, que es compatible con sistemas de caché.
- Es utilizado por un gran número de usuarios y hay multitud de recursos y materiales de consulta.

4.1.2. Front end

4.1.2.1. HTML

HTML es un lenguaje de marcado (los elementos se representan mediante etiquetas) que describe la estructura de una página web, es *renderizado* por un navegador web [18].

En conjunción con HTML, se ha utilizado Laravel Blade.

4.1.2.2. CSS y Sass

CSS permite definir el estilo (aspecto visual) de los elementos que se muestran en la página web [19]. Puede ser utilizado dentro de la página HTML o bien enlazado como un fichero auxiliar.

Sass es un lenguaje de hojas de estilo compilable en CSS que facilita la creación de estilos mediante el uso de variables y herencia de estilos [20].

4.1.2.3. Javascript

Javascript es un lenguaje de programación interpretado o compilado en tiempo de ejecución utilizado en el desarrollo de páginas web para dotarlas de funcionalidad [21].

Además, se dispone de infinidad de bibliotecas que facilitan el desarrollo. En concreto, las más importantes que se han utilizado son Chart.js y DataTables:

Chart.js es una biblioteca de código abierto que permite la creación de gráficos de una manera sencilla. A diferencia de otras bibliotecas como D3, tiene una curva de aprendizaje óptima para principiantes, sacrificando la capacidad de crear gráficas más avanzadas y formas de interacción más complejas [22]: para el desarrollo de un prototipo, e incluso utilizado en producción, se ha preferido Chart.js sobre D3 por la sencillez de uso. Respecto a Chartist, la biblioteca utilizada por Aplifisa actualmente, se ha preferido Chart.js por considerarse que permite crear gráficas con una mejor visualización y por el número de usuarios: según la fuente consultada, Chart.js posee una comunidad de usuarios muy superior a la de Chartist [23], por lo que se considera que tendrá un mayor tiempo de vida y soporte y una mayor cantidad de recursos de consulta.

Por su parte, DataTables es una herramienta que facilita la creación de tablas interactivas. En conjunción con el paquete Yajra DataTables en la parte servidor, se pueden crear y cargar con suma facilidad las tablas.

4.1.2.4. Plantilla front end

Para el desarrollo de la parte visual se ha utilizado la plantilla Tim Material Dashboard. Es una plantilla inspirada en Material Design construida sobre Bootstrap 4 y distribuida bajo licencia MIT [24].

Usar esta plantilla permite ahorrar tiempo de desarrollo y ofrece un aspecto visual similar al del resto de aplicaciones web de Aplifisa.

4.2. Entorno de desarrollo

4.2.1. Docker

Docker es una plataforma de contenedores utilizada para el despliegue de aplicaciones de manera automatizada en elementos denominados contenedores.

A diferencia de las habituales máquinas virtuales, los contenedores aíslan las aplicaciones sobre un *software* (maquinaria Docker) en vez de compartir el *hardware*, lo que facilita la interacción entre contenedores y el rendimiento [25].



Figura 4.1: Comparativa entre Docker y máquinas virtuales [25]

Docker permite crear un entorno de desarrollo con todas las herramientas necesarias para trabajar con Laravel como se explica en el siguiente punto.

4.2.2. Laradock

Laradock se define como un “entorno de desarrollo completo para PHP utilizando Docker” [26]. Es un conjunto de contenedores que disponen de todas las herramientas necesarias para trabajar con Laravel: servidor web (Apache, NGINX y Caddy), sistemas gestores de bases de datos (entre otros, MySQL, PostgreSQL y MongoDB), sistemas de caché como Redis, depurador...

Su principal ventaja es que todas las herramientas vienen preconfiguradas y se pueden configurar a la hora de crear el contenedor. Además, al estar dentro de contenedores Docker, las herramientas que provee se integran correctamente con editores de código como Visual Studio Code o PHPStorm.

Estas ventajas y herramientas que integra lo convierten en un entorno de desarrollo muy completo y de gran utilidad para el desarrollador. Las alternativas tradicionales como XAMPP (entorno de desarrollo para cualquier sistema operativo formado por Apache, MariaDB, PHP y Perl), son de muy fácil instalación [27] y no requieren conocer Docker. Como desventaja, no proveen de herramientas

adicionales e incluso básicas como es el depurador, que mientras en XAMPP deben ser instaladas y configuradas aparte, con Laradock ya vienen instaladas y configuradas.

Para el proyecto, se ha utilizado NGINX como servidor web y MySQL como sistema gestor de bases de datos.

4.2.3. Visual Studio Code

Visual Studio Code es un editor de código propiedad de Microsoft para el desarrollo de aplicaciones *software*. De código abierto (por lo que la propia comunidad de usuarios puede mejorar el editor), es gratuito y altamente personalizable a través de extensiones.

Está adaptado para poderse utilizar en desarrollo remoto, como es el caso cuando se utilizan contenedores Docker, proveyendo para ello además de diversas utilidades útiles a la hora de trabajar con Docker (gestión de puertos, gestionar los contenedores...). A través de extensiones, se puede utilizar para desarrollar en casi cualquier lenguaje de programación para el que el editor no ofrezca soporte directo como es el caso de PHP.

4.2.4. MySQL Workbench

MySQL Workbench es una herramienta visual para trabajar con bases de datos y realizar labores de administración de manera sencilla.

Además, incluye una herramienta de modelado que permite generar un diagrama entidad-relación. Este diagrama se puede transformar en código SQL para trasladar el modelo a la base de datos.

4.2.5. Deployer

Deployer es una herramienta para el despliegue automático de aplicaciones PHP realizadas utilizando *frameworks* como Laravel [28]. Se basa en la creación de recetas, que definen el conjunto de tareas y comandos a ejecutar (como puede ser realizar una migración en el servidor, utilidad de Laravel explicada en el apartado 5.2).

Se caracteriza por utilizar SSH, permitir la ejecución paralela de tareas y por implementar un sistema de control de versiones de producción en el servidor, permitiendo volver a la versión de producción anterior. Para desplegar la aplicación permite tanto mover los ficheros al servidor como utilizar un repositorio en la nube desde el que subir los cambios.

Esta herramienta ha sido utilizada para desplegar los cambios implementados en el entorno de pre-producción. Además, es de utilidad en los procesos de desarrollo de la empresa para automatizar tareas que, en la actualidad, se hacen de forma manual.

4.2.6. Auditoría y pruebas

Para evaluar el rendimiento y la calidad (accesibilidad, SEO...) se han utilizado las herramientas Lighthouse y “axe DevTools”.

Lighthouse viene incluida por defecto en los navegadores Chromium (como Google Chrome o Microsoft Edge) y permite obtener puntuaciones en los distintos apartados. Además de ofrecer puntuaciones, indica los problemas que hay y hace sugerencias de mejora.

La herramienta “axe DevTools” se ha utilizado como complemento para evaluar la accesibilidad. A diferencia de Lighthouse, axe no recarga la página y comprueba la accesibilidad de lo que se ve en pantalla, siendo útil para ver la accesibilidad de elementos no mostrados al cargar la página.

4.3. Entorno de pre-producción

Para demostraciones y pruebas de rendimiento, se utiliza una máquina virtual VMware con el servidor web NGINX y MySQL como sistema gestor de bases de datos. Esta máquina virtual simula la configuración del entorno de producción.

5. PROPUESTA DE SOLUCIÓN

5.1. Metodología de trabajo

El trabajo se ha desarrollado con metodología ágil basada en Scrum. Se han realizado las siguientes modificaciones para adaptarlo a las necesidades del Trabajo Fin de Máster:

- Al ser un proyecto realizado de manera individual, se han suprimido las reuniones y dinámicas del equipo de desarrollo (sí se han realizado reuniones equiparables a las de planificación y revisión del esprint con los tutores) que lleva asociada la metodología Scrum.
- Los sprints no tienen una duración fija: al haberse acordado con la empresa que el proyecto se desarrollase, mientras sea como Trabajo Fin de Máster, con un horario variable en función de las necesidades de la misma, se ha optado por usar un periodo orientativo (no fijo) de dos semanas para realizar cada esprint.
- Al final de cada esprint y en función de la disponibilidad de los tutores, se ha realizado una reunión con fines retrospectivos y de validación.

En cuanto al tiempo empleado, se acordó con la empresa realizar el presente Trabajo Fin de Máster empleando un 20-25%, como máximo y priorizando el resto de proyectos de la empresa, del horario laboral; además del tiempo dedicado fuera de horario laboral.

Antes de iniciar el propio proceso de desarrollo se realizó una primera fase de entrenamiento, destinada a aprender a utilizar Laravel y mejorar en el desarrollo de interfaces web. Durante dicha fase de entrenamiento se realizaron dos versiones: una primera con el inicio de sesión y registro (particular para el Trabajo Fin de Máster, Aplifisa sigue procedimientos propios para el alta de usuarios) y una segunda añadiendo la interfaz base.

La captura de requisitos se ha realizado mediante tarjetas representando las historias de usuario indicando por un lado el usuario que la realiza, qué realiza y qué fin pretende conseguir y, por otro lado, los criterios necesarios para dar por completada la historia de usuario (criterios de aceptación). El proceso de captura de requisitos, junto al catálogo de requisitos, se detalla en el Anexo A.

En suma, el calendario de trabajo efectivo realizado para el presente Trabajo Fin de Máster, excluyendo el entrenamiento, se muestra en la Tabla 5.1.

Tipo de elemento	Número de esprint	Nombre	Comienzo	Fin
Esprint	1	v0.3 (Listado de trabajadores)	vie 26/03/21	mar 30/03/21
Elemento de trabajo	1	Lista de trabajadores	vie 26/03/21	mar 30/03/21
Esprint	2	v0.4 (Gestión de currículums)	dom 11/04/21	mar 27/04/21
Elemento de trabajo	2	Redactar currículum	dom 11/04/21	sáb 24/04/21
Elemento de trabajo	2	Consultar currículum	dom 25/04/21	mar 27/04/21
Esprint	3	v0.5 (Matriz de competencias)	mié 28/04/21	dom 02/05/21
Elemento de trabajo	3	Crear matriz de competencias	mié 28/04/21	dom 02/05/21
Esprint	4	v0.6 (Análisis de indicadores)	lun 10/05/21	sáb 22/05/21
Elemento de trabajo	4	Rotación de personal	lun 10/05/21	jue 13/05/21
Elemento de trabajo	4	Permanencia	vie 14/05/21	dom 16/05/21
Elemento de trabajo	4	Competencias	dom 16/05/21	dom 16/05/21
Elemento de trabajo	4	Absentismo	mié 19/05/21	sáb 22/05/21

Tabla 5.1: Calendario del proyecto

Respecto a las reuniones, con Ana Belén Gil se han mantenido reuniones los días 24 de febrero, 9 de abril y los días 6 y 26 de mayo. En la primera reunión se debatió acerca del proyecto a realizar. Ideado el proyecto e iniciado el proceso de desarrollo, el resto de reuniones, coincidentes en el tiempo con el fin de los sprints 1, 3 y 4, se evaluó el trabajo realizado y se fijaron en el tiempo los objetivos.

Con Laureano Ayuso como representante de Aplifisa, además de dos reuniones iniciales (una a principios de febrero y otra a mediados de marzo) para plantear y acordar el proyecto, se mantuvieron reuniones al final de los sprints 1, 3 y 4 para que la empresa validase el proyecto conforme se iba desarrollando cada funcionalidad.

Además de las reuniones, se ha mantenido comunicación con ambos tutores vía correo electrónico para la comunicación de los avances realizados en el proyecto y el envío de documentación.

5.2. Arquitectura del sistema de información

Por un lado hay que definir la estructura general que tendrá el sistema de información en el marco de Aplifisa y, por otro lado, la estructura de la base de datos propia del proyecto.

Como se explica en la discusión de los diversos datos que puede haber en el proyecto y sus orígenes, hay:

- Datos de las aplicaciones de escritorio contenidas en bases de datos locales respecto a los clientes.
- Posibles datos de fuentes ajenas a la empresa, como datos de currículums en portales web de búsqueda de empleo.
- Datos generados por las aplicaciones web y móviles de Aplifisa. Estos datos se almacenan en bases de datos alojadas en servidores contratados por Aplifisa y son de fácil acceso.
- Datos que genere el proyecto una vez esté en explotación. Al igual que en el caso anterior, se alojarán en una base de datos propia y de fácil acceso.

Siguiendo el esquema *bottom-up*, se ha diseñado desde un punto de vista conceptual el sistema de información. Este diseño se puede consultar en la Figura 5.1 y parte del sistema de información ya existente (parte relativa al Portal del Empleado de Aplifisa) para añadir lo relativo al presente proyecto y modelarlo como un sistema de información propio de la inteligencia de negocio.

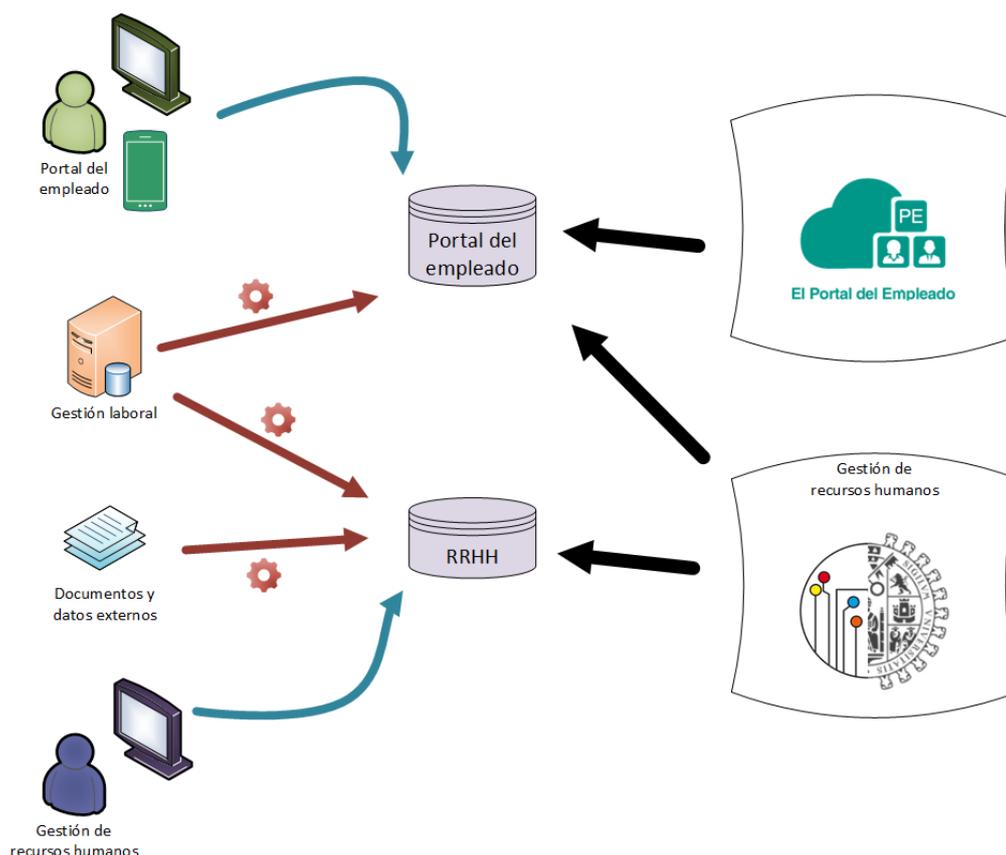


Figura 5.1: Diseño conceptual del sistema de información

Por un lado, los usuarios, al usar las aplicaciones, generan datos que se almacenan en un *data mart* (base de datos propia de la aplicación). También, en el *data mart* correspondiente, se cargan los datos (previamente sometidos a

procesos de extracción y transformación) de las aplicaciones de escritorio y de documentos y datos externos. Por el otro, cada aplicación accede a los *data mart* para consumir los datos que necesitan.

De este diseño cabe destacar varios aspectos:

- Cada base de datos contiene datos asociados (currículums, fichajes de los trabajadores...), por lo que son, en efecto, *data marts*.
- Es escalable respecto a la funcionalidad: se pueden añadir aplicaciones al sistema (lado derecho de la figura), nuevos *data marts* (centro) y nuevas fuentes de datos tanto de aplicaciones propias de Aplifisa como de documentos (lado izquierdo).
- Es capaz de evolucionar: se puede añadir Apache Hive u otro almacén de datos para agrupar y mejorar el rendimiento al leer los datos (Hive es infraestructura de *data warehouse* basada en Hadoop). Este añadido permitiría utilizar herramientas OLAP como Apache Kylin (compatible con Hive).

Con este diseño conceptual en mente, se diseña la base de datos (*data mart*) propia de la aplicación.

Para poder enlazar los *data mart*, que es la parte más crítica a la hora de implementar el sistema de información, se ha optado por el uso del mecanismo de tablas federadas presente en MySQL: dada una tabla en una base de datos, este mecanismo permite crear una tabla en otra base de datos (presente en el mismo equipo o en otro) y federarla para que realice las consultas, de manera remota, sobre la tabla con la que se enlaza.

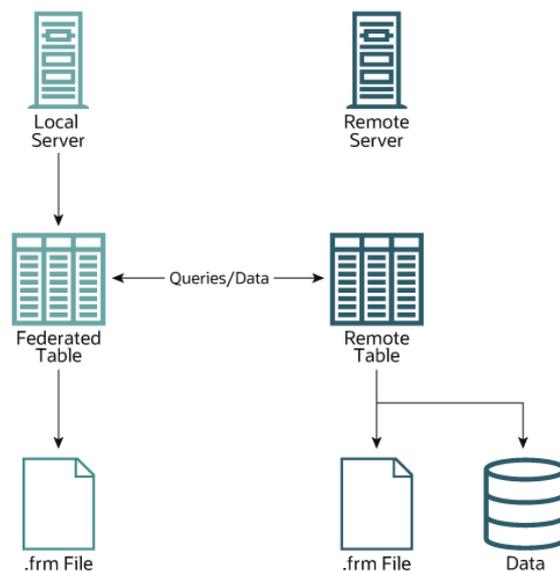


Figura 5.2: Funcionamiento del mecanismo de tablas federadas [29]

Una de las desventajas del uso de la federación de tablas es que este mecanismo no soporta el sistema de transacciones de MySQL. Esta limitación, tal y como está pensado el sistema, no debería ser ningún problema: son tablas a las que se accede solo para leer información, operación en la que no usar transacciones no es crítico, y no para modificarla, operaciones en las que sí es crítico.

La otra desventaja está en el rendimiento. El acceso a los datos de la tabla remota devuelve a la tabla federada todas sus tuplas a través de la red para que esta los procese, si bien este efecto se puede paliar aplicando a la tabla remota los índices adecuados.

El *data mart* de este producto, en este estado de su desarrollo, contendrá información relativa a los currículums de los trabajadores.

Aparte de las necesidades propias del *data mart*, en el modelo entidad-relación de la base de datos realizado se muestra además la tabla federada: si bien no forma parte de la base de datos, se ha decidido incluir para ofrecer más información utilizando un solo diagrama. Dicho modelo, realizado con la herramienta MySQL Workbench, se muestra en la Figura 5.3.

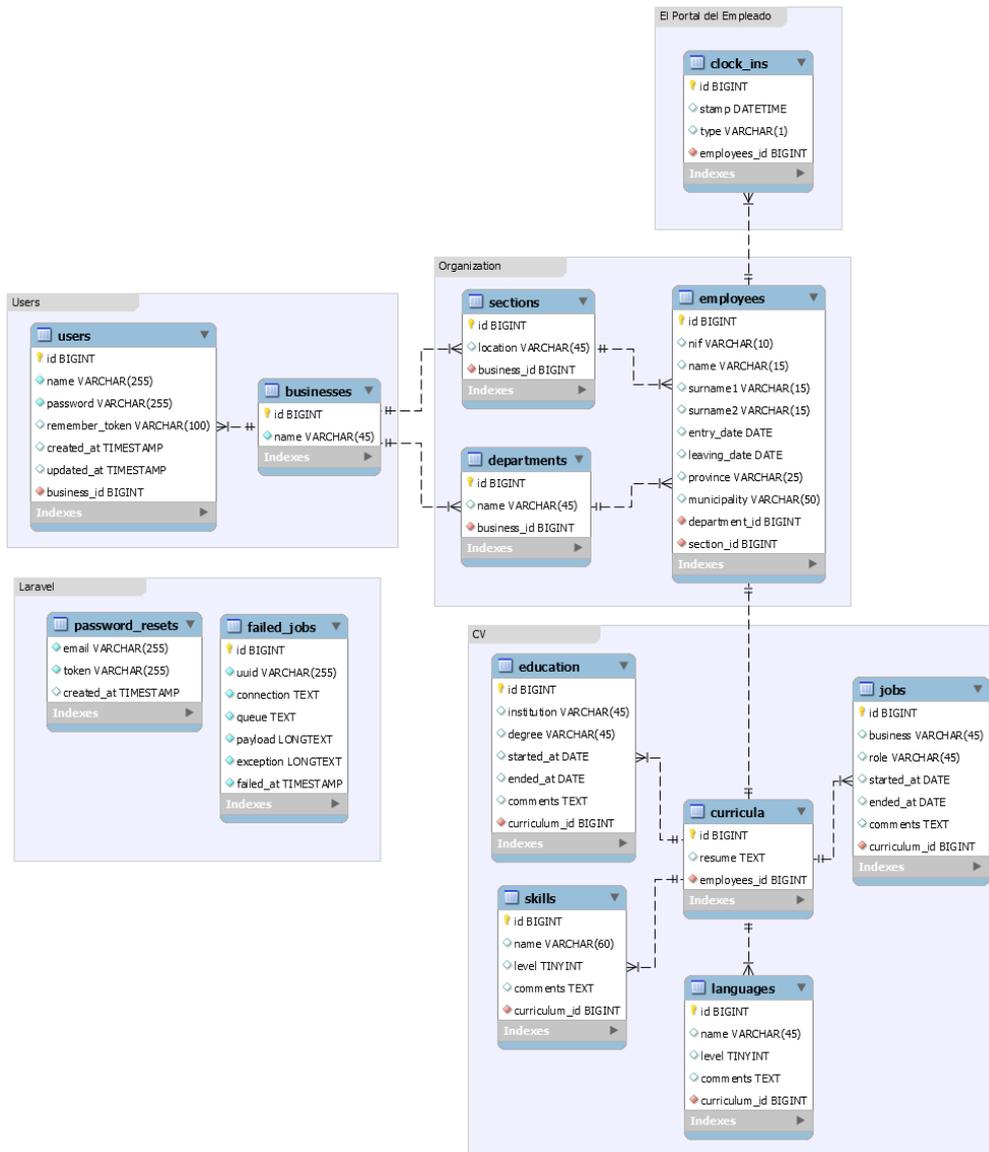


Figura 5.3: Modelo entidad-relación de la base de datos

Hay que tener en cuenta que MySQL Workbench no permite la creación de tablas federadas a partir del modelo y que tampoco se puede definir una migración con el sistema de migraciones de Laravel para especificar la federación, por lo que la tabla federada que se conectará a la tabla de fichajes de El Portal del Empleado se deberá crear de manera manual.

La tabla federada debe tener los mismos atributos que la tabla original¹ y especificar cómo conectarse a dicha tabla. La Figura 5.4 muestra la consulta SQL necesaria para poder crear la tabla federada.

```
1 CREATE TABLE `clock_ins` (  
2   `id` bigint unsigned NOT NULL AUTO_INCREMENT,  
3   `stamp` datetime DEFAULT NULL,  
4   `type` varchar(1) CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci DEFAULT NULL,  
5   `employee_id` bigint unsigned NOT NULL,  
6   PRIMARY KEY (`id`),  
7   KEY `fk_check_ins_employees_idx` (`employee_id`)  
8 ) ENGINE = FEDERATED DEFAULT CHARSET = utf8 CONNECTION = 'mysql://root:root@127.0.0.1:3306/TFM_2/clock_ins'
```

Figura 5.4: Consulta de creación de la tabla federada

El resto de tablas no han sido creadas manualmente: se ha hecho uso de la funcionalidad “ingeniería directa” descrita en el apartado 4.2.4, excluyendo la tabla federada de fichajes del proceso, para generar el código SQL necesario para crearlas y ejecutarlo en la base de datos, resultando este procedimiento en su creación.

Adicionalmente, se ha hecho uso del paquete Laravel Migrations Generator para generar las migraciones [30]. Internamente, lo que hace este paquete es conectarse a la base de datos para obtener las descripciones de las tablas (DDL en forma de código SQL) y transformarlas a código PHP necesario para crear la migración.

En Laravel, las migraciones permiten crear un sistema de control de versiones de la base de datos y sirven de alternativa para crear y modificar el esquema de la base de datos. Su mayor utilidad es la de poder modificar adecuadamente el esquema de la base de datos en cualquier entorno.

Haciendo uso las migraciones generadas por este paquete, junto a la herramienta de modelado provista por MySQL Workbench, el proceso de crear la estructura de la base de datos se simplifica, puesto que apenas se dedica esfuerzo a la programación y el proceso se realiza, mayoritariamente, de manera gráfica y visual.

5.3. Gestión de currículums

La gestión de los currículums se realizará partiendo de un listado de trabajadores de manera análoga a como se hace en El Portal del Empleado, aplicación propia

¹ Para poder probar el proyecto, se ha optado por crear una tabla ficticia (no asociada al Portal del Empleado real) en una segunda base de datos en el mismo servidor de ejecución.

de Aplifisa: es una manera fácil de implementar y, además, similar al resto de aplicaciones de la empresa. La funcionalidad se realiza como se indica en la historia de usuario contenida en la Tabla 5.2.

US-001 Listar trabajadores
<p>Descripción:</p> <p>El usuario quiere visualizar los trabajadores de su empresa para poder ver su estado e información básica, así como para poder escoger uno y redactar o consultar su currículum.</p>
<p>Criterios de aceptación:</p> <p>El usuario puede visualizar todos los trabajadores de su empresa. De cada trabajador de su empresa, puede ver su: estado (activo, no dado de alta o dado de baja), centro de trabajo y departamento al que está adscrito, N.I.F., nombre completo y su fecha de alta y de baja en la empresa.</p> <p>El usuario puede filtrar y buscar trabajadores por cualquiera de los campos.</p> <p>El usuario puede, a partir del listado, redactar o consultar el currículum de los trabajadores.</p>

Tabla 5.2: US-001 Listar trabajadores

Se entiende que un trabajador está en estado activo si tiene fecha de alta y no tiene fecha de baja, dado de baja si tiene fecha de alta y de baja, o bien no dado de alta si no tiene fecha de alta ni de baja. La posibilidad de que un trabajador no esté dado de alta, en principio, está reservada a que sea un candidato a entrar en la empresa por lo que, aunque no está pensado en el estado actual del desarrollo que exista este tipo de trabajadores, sí es interesante dejar planteada la posibilidad de cara al futuro.

Respecto a la acción de redactar y consultar el currículum, se podrá realizar en cualquiera de los casos:

- Si es un trabajador en activo, tanto la redacción como la consulta del currículum son necesarios para el desempeño de las funciones de gestor de recursos humanos.
- Si es un trabajador no dado de alta, la redacción, que tiene funciones de alta y edición, sirve para realizar correcciones y la consulta permite analizar el currículum del candidato.
- Si es un trabajador dado de baja, dicho trabajador puede volver a la empresa, por ejemplo, cuando ha solicitado una excedencia y esta finaliza: dejar el currículum guardado evitaría tener que redactar otra vez su currículum y permitiría poderlo consultar y realizar modificaciones como podría ser en el caso de que haya adquirido nuevas competencias o trabajado para otra entidad.

Desde el empleado, se accederá o bien a la redacción del currículum o bien a su consulta: si bien esto podría ser una única historia de usuario con una sola interfaz, se ha decidido dividir historia e interfaz en dos para detallar mejor cada aspecto. La Tabla 5.3 contiene la historia de usuario que describe la funcionalidad de redacción de currículum y la Tabla 5.4 la correspondiente a la edición.

US-002 Redactar currículum
Descripción: El usuario quiere redactar el currículum de un trabajador de su empresa para poder tener registrada en la aplicación su información básica (nombre, apellidos y localización), presentación, estudios académicos, experiencia laboral, competencias y los idiomas que conoce y su nivel en ellos.
Criterios de aceptación: El usuario puede introducir datos en cada campo. El usuario puede añadir y eliminar registros de cada sección del currículum. El sistema carga en los campos la información del currículum que tenga guardada. El usuario no puede modificar la información básica de un trabajador desde la aplicación porque esta información proviene de la aplicación de escritorio. El usuario no puede redactar el currículum de un trabajador que no pertenezca a su empresa.

Tabla 5.3: US-002 Redactar currículum

US-003 Consultar currículum
Descripción: El usuario quiere leer el currículum de un trabajador para ver la información de dicho trabajador.
Criterios de aceptación: El sistema carga en los campos la información del currículum que tenga guardada. El usuario no puede modificar ningún campo del currículum. El usuario no puede consultar el currículum de un trabajador que no pertenezca a su empresa.

Tabla 5.4: US-003 Consultar currículum

La interfaz debe ser dinámica y ajustarse a las necesidades del usuario descritas en las respectivas historias: los trabajadores, a lo largo de su vida laboral, evolucionan y, por lo tanto, la interfaz deberá permitir dicha evolución poniendo especial atención a que tenga una usabilidad adecuada.

Para finalizar este apartado, está el desarrollo digital de la matriz de competencias. Esta idea tiene su origen en un curso de formación en Management 3.0 recibido en la empresa durante la realización del proyecto. Esta matriz, aplicada al Management 3.0 permite al equipo (equipo de trabajo, miembros de un proyecto, trabajadores de un departamento...) conocer el nivel de las competencias de cada miembro de cara a poder encontrar deficiencias que impidan progresar al equipo y entrenar las competencias que lo necesiten hasta un nivel adecuado [31]. En la práctica y según indica la teoría, cada trabajador marca para cada competencia su nivel (novato/básico, aprendiz/intermedio y experto/avanzado) y lo anota en la matriz.

		MIEMBROS																
		APRENDIZ	PROCT.	MAESTRO														
LENGUAJES / SOFTWARE	PYTHON																	
	PHP	●	●	●	●	●	●	●	●	●								
	JAVASCRIPT	●	●	●	●	●	●	●	●	●								
	C++	●		●	●	●	●	●	●									
	HTML/CSS	●	●	●	●	●	●	●	●									
	MAYA			●	●	●												
	CLIP STUDIO			●	●													
	PHOTOSHOP			●	●													
	MONGO BBDD																	
	SQL	●	●	●	●	●	●											

Figura 5.5: Ejemplo de la matriz de competencias, aplicado a un grupo de trabajadores de Aplifisa, realizado durante el curso de formación.

Si bien la matriz de competencias está orientada al objetivo de Management 3.0, conseguir equipos autoorganizados, sí que puede ser utilizada en la gestión de recursos humanos y gestión de proyectos y no solo por el propio equipo. Como se vio en el apartado 3.1.3, añadir un apartado de gestión de proyectos desde el punto de vista de los conocimientos de los trabajadores es un aspecto interesante para el proyecto: por medio de la matriz de competencias, el gestor podría crear equipos optimizados para conseguir los objetivos del proyecto y balanceados que permitan a los trabajadores desarrollar las competencias más bajas sin que afecte negativamente al proyecto. Además, los currículums dados de alta en la aplicación permiten obtener las competencias de los empleados para su uso en la matriz. Con este objetivo en mente, se define la historia de usuario de la Tabla 5.5.

US-004 Crear matriz de competencias

Descripción:

El usuario quiere ver las competencias de los trabajadores en forma de matriz de competencias para gestionar mejor los diversos equipos de la empresa.

Criterios de aceptación:

El sistema muestra correctamente el nivel de las competencias de los trabajadores.

El usuario puede filtrar por competencias.

El usuario puede filtrar por trabajadores.

El usuario puede mover de posición en la matriz a los trabajadores.

El usuario puede seleccionar a qué centro y a qué departamento pertenecen los trabajadores a mostrar en la matriz de competencias.

Tabla 5.5: US-004 Crear matriz de competencias

La matriz de competencias está pensada para adaptarse a lo que sería la forma estándar de trabajo de un gestor de proyectos o de recursos humanos en un entorno real, por lo que permite: escoger un grupo de trabajadores (selección de centro y departamento y filtrado de trabajadores), seleccionar competencias relevantes (filtrado de competencias) y reordenar los trabajadores en función de las necesidades y preferencias del usuario.

5.4. Métodos de visualización de datos

La visualización de datos es la representación de estos de manera visual. Una de las formas más habituales son las gráficas, si bien existen otras formas como las infografías, las tablas u otros gráficos más elaborados.

Chart.js, la biblioteca de gráficas escogida, ofrece los siguientes tipos de gráficas:



Figura 5.6: Gráficas disponibles en Chart.js

Además de estos tipos, existen cuatro tipos derivados más:

- De área: gráfica de líneas con sombreado.
- De áreas apiladas: derivada de la anterior, representa varios conjuntos de datos. Ejemplo: mes del año, tipo de producto (tipo apilado sobre el mes) y unidades vendidas.
- De barras apiladas: dos o más variables categóricas que se puedan apilar y una numérica. Similar al tipo anterior, deriva de la gráfica de barras.
- Mixta: mezcla de varias gráficas. Ejemplo: ventas estimadas representadas como gráfica de líneas y ventas reales como gráfica de barras.

El uso de cada una se muestra en la Tabla 5.6 en forma de comparativa [32, 33, 34]:

Gráfica	Uso principal	Uso adicional	Dimensiones
Líneas	Evolución		2
Barras	Comparación		2 (3 si apiladas)
Burbujas	Correlación	Distribución	3
Dispersión	Correlación	Distribución	2
Radar	Comparación		2
Área polar	Composición		2
Anillos	Composición		2
Circular	Composición		2
Área	Evolución	Composición	2 (3 si apiladas)

Tabla 5.6: Comparativa de los tipos de gráficas

A la hora de escoger el tipo de gráfica adecuado habrá que tener en cuenta el objetivo que se pretende conseguir con la gráfica y el tipo de datos necesarios para la gráfica.

Además del tipo de gráfica, otro aspecto importante en la visualización de datos es la elección de colores. Los colores, además de embellecer el aspecto visual, ayudan a entender los datos representados y lo que estos indican. A este respecto, hay una serie de pautas generales que seguir [35]:

- Los colores deben ser consistentes: para no dificultar la lectura y comprensión de los datos, que es el objetivo de la visualización de datos, es recomendable que haya una progresión en la escala de colores y que estos no desvíen el foco de atención.
- La consistencia también se puede aplicar al significado de los datos, pues los colores deben amoldarse a cómo el usuario visualiza, mentalmente, la información. Este aspecto se ve claramente a la hora de evaluar resultados electorales: representar dos partidos políticos con el color del otro partido

puede confundir al usuario, pues espera que el color del resultado del partido político se asemeje al color con el que se asocia el partido.

- Hay que procurar que los colores no generen problemas de accesibilidad para que cualquier usuario sea capaz de entender los datos. Si bien es complicado adaptarse a los estándares de accesibilidad cuando se trabaja con visualización de datos, sí que hay que evitar que el usuario no pueda distinguir de los colores. Un buen método para conseguir esto, está en realizar pequeñas variaciones de la tonalidad del color sin cambiar el color en sí para que el usuario, aunque no sea capaz de distinguir correctamente los colores de dos elementos, pueda comprender que son dos colores distintos y no parte de la misma categoría.

Por último, y como ayuda a la hora de facilitar la comprensión, “los datos deben contar una historia” [36]. Esta frase indica que, a la hora de mostrar la información, hay que construir una narrativa en línea con el propósito de cada dato visualizado. Aplicada a la inteligencia de negocio, los datos deben contar historias: situación actual (“¿cómo estamos ahora mismo?”), evolución del dato (“¿nuestras medidas funcionan?”, “¿mejoramos o empeoramos?”, “¿se está creando un problema y en qué medida de cara a implantar o no medidas?”) o cómo este está distribuido en la empresa (“¿qué situación hay en esta sección?”, “¿es un problema general o particular de algún trabajador o departamento?”).

5.5. Indicadores clave de rendimiento

Los indicadores clave de rendimiento (*key performance indicators*, KPI) son métricas empleadas para conocer la situación de una empresa y poder tomar medidas y decisiones en base a la situación de la empresa.

Los indicadores se definen en función de la meta que se pretenda alcanzar (por ejemplo, ejecutar un proyecto en un tiempo determinado) y su elección se basa en uno de dos modelos: el modelo SMART (específico, medible, alcanzable, relevante, tiempo limitado) y el modelo FAST (discutido con frecuencia, ambicioso, específico, transparente).

Según el modelo SMART [37], los objetivos (y por lo tanto los indicadores) deben ser:

- Específicos: son objetivos concretos y para un área determinada, no generales.
- Medibles: se pueden medir y cuantificar.
- Alcanzables: son posibles de satisfacer.
- Relevantes: importantes para la empresa.
- Tiempo limitado: el objetivo se puede cumplir en un tiempo determinado.

Mientras que según el modelo FAST [38]:

- Discutidos con frecuencia: son debatidos y utilizados periódicamente.
- Ambiciosos: complejos pero posibles de alcanzar.
- Específicos: medibles y métricas concretas.
- Transparentes: objetivos y métricas públicos para todos los empleados.

En ambos casos, deberán ser indicadores cuantificables y, al ser un proyecto general (no asociado a clientes con un modelo de negocio determinado) permitir que los clientes puedan amoldarse a estos esquemas. Respecto al modelo, se van a usar indicadores pensados para trabajar con un modelo SMART, pues el uso de la plataforma está destinado a gestores de recursos humanos y puestos directivos de empresas con diversos modelos de negocio.

Para la versión inicial del proyecto, se han escogido tres objetivos o aspectos: retención de personal, cumplimiento horario y análisis de competencias. Dentro de cada objetivo, se definen varios indicadores KPI que permitan alcanzar mejoras en dichas áreas de acción. El usuario será el responsable de concretar el objetivo a alcanzar, para lo cual la plataforma desarrollada le mostrará los indicadores de forma que sea capaz de entender los datos, satisfacer dichos objetivos, detectar posibles problemas y evaluar el impacto de las acciones desarrolladas para alcanzar los objetivos.

5.5.1. Indicadores para la retención de personal

Para analizar la capacidad de retener al personal hay dos indicadores principales: la tasa de rotación y el tiempo de permanencia media en la empresa. Además de estos, hay otros indicadores más específicos como puede ser la tasa de rotación del personal en función de la causa de la baja [39].

La tasa o índice de rotación de personal (IRP), tal y como se define en la Ecuación 1, es el cociente entre el número de empleados que abandonan la empresa en un periodo de tiempo (S) y el promedio de empleados al inicio (I) y al final del periodo (F) en dicho periodo, expresado en porcentaje [40].

$$IRP = \frac{S}{(I + F) / 2} * 100$$

Ecuación 1: Índice de rotación de personal

Es un indicador con impacto en las finanzas de la empresa debido al coste asociado a la bajada de rendimiento producida por la marcha del trabajador y la contratación de su sustituto (costes de entrenamiento, tiempo en alcanzar el nivel

de aprendizaje adecuado...). Dicho de otra manera, señala la capacidad de la empresa de retener a los trabajadores y evitar sustituciones. También puede llegar a ser un indicativo de problemas en el clima laboral (bajos salarios, falta de motivación...) [41]. Por norma general, se intenta tener en un nivel bajo de rotación de personal; este nivel depende, principalmente, del sector y de las circunstancias de los trabajadores que abandonan la empresa.

Por su parte, la permanencia en la empresa mide el tiempo en que, de media, los trabajadores forman parte de la empresa. Este indicador se usa, además de para evaluar la retención del talento y de manera similar al indicador anterior, para evaluar la satisfacción de los empleados (si la permanencia es alta, la satisfacción general es buena). Otro aspecto interesante de la permanencia es evaluarla, además de por la media, por franjas de años de permanencia, en especial para el caso de permanencia inferior a un año: un trabajador que se va antes de cumplir su primer o primeros años en la empresa es una inversión de capital con poco retorno de la inversión, pues se invierten recursos (tiempo y dinero) en formar a personal que no va a ser capaz de devolver la inversión realizada [42].

Cabe destacar de estos dos factores que, mientras la rotación de personal se trata desde una perspectiva temporal (rotación por trimestres/años), la permanencia es una foto fija que señala el estado de la empresa en un momento dado. Esta circunstancia implica que la rotación de personal es un indicador para la toma de decisiones a medio o largo plazo y la permanencia para las decisiones a largo plazo (en el corto plazo, incluso en el medio, es difícil que varíe sustancialmente como para requerir acciones en dichos plazos).

La historia de usuario que define esta funcionalidad se puede consultar en la Tabla 5.7.

US-005 Ver indicadores de retención de personal
<p>Descripción:</p> <p>El usuario quiere ver los indicadores de rotación de personal y permanencia media para evaluar la capacidad de la empresa de retener al personal.</p>
<p>Criterios de aceptación:</p> <p>El sistema muestra el índice de rotación de personal.</p> <p>El sistema muestra la permanencia media y el porcentaje de empleados en cada franja de permanencia en la empresa.</p> <p>El usuario puede filtrar por centro para consultar cada indicador por departamento de la empresa.</p> <p>El usuario puede seleccionar si desea visualizar la rotación de personal de manera anual o trimestral.</p>

Tabla 5.7: US-005 Ver indicadores de retención de personal

5.5.2. Indicadores para el cumplimiento horario

Un trabajador que no realiza las horas por las que está contratado es un trabajador que provoca pérdidas. Uno de los factores principales en el éxito de una empresa es la productividad (producción esperada respecto al tiempo empleado en producir) de sus empleados [43] por lo que, si un empleado no trabaja por las horas por las que está contratado (menos tiempo empleado en producir), está perjudicando a la empresa.

El cumplimiento horario, por lo tanto, es un factor importante que, además, puede indicar problemas estructurales (como la falta de motivación). Para evaluarlo se pueden utilizar dos indicadores: tasa de absentismo y factor Bradford.

La tasa de absentismo, como su nombre indica, mide la diferencia, expresada como porcentaje, entre las horas efectivas de trabajo de los empleados respecto a las horas contratadas en un periodo de tiempo (normalmente mensual o anual). Este indicador se puede calcular tanto de manera general (por empresa, sección...) como individual (por trabajador). Por lo general, este indicador mide exclusivamente horas no trabajadas sin justificar para excluir ausencias y bajas con derecho a retribución (con deber legal del empleador a retribuir) y justificables como, por ejemplo, las ausencias para acudir a una cita médica, enfermedad o por el fallecimiento de un familiar.

El factor Bradford es otro de los indicadores existentes [44, 45]. Este indicador tiene tres principales diferencias respecto a la tasa de absentismo: es individual, evalúa días (no horas) no trabajados, siempre evalúa ausencias en el último año (no otros periodos) y no es un porcentaje, sino que es un factor calculado que da una puntuación. Dicho factor está diseñado para priorizar ausencias con mayor

coste, asumiendo que es mejor para la empresa tener pocas ausencias pero de larga duración (planificables y de efectos mitigables) que muchas ausencias de poca duración (no planificables y de efectos poco mitigables). También, a diferencia de la tasa de absentismo, suele incluir en el cálculo los días no trabajados por ausencia justificable por lo que, al analizarlo, hay que tener en cuenta las circunstancias personales del trabajador para, entre otros problemas, evitar cualquier tipo de discriminación laboral. El factor Bradford viene determinado por la Ecuación 2, en la que B es el factor Bradford, S el número de ventanas (periodos) de ausencia y D el número total de días de ausencia.

$$B = S^2 * D$$

Ecuación 2: Cálculo del factor Bradford

Para ambos indicadores, cuanto más cercanos a cero mejor situación para la empresa, si bien es cierto que el factor Bradford (y la tasa de absentismo si no contempla ausencias justificables), por su naturaleza, es más complicado que sea cero. La empresa cliente será la que, en función de sus protocolos, defina los rangos de valores de cara a tomar las medidas que considere oportunas (reforzar la motivación, dar toques de atención a los empleados correspondientes si es un problema con trabajadores individuales...).

La funcionalidad desarrollada se describe en la historia de usuario mostrada en la Tabla 5.8.

US-006 Ver indicadores de cumplimiento horario
<p>Descripción:</p> <p>El usuario quiere ver los indicadores de tasa de absentismo y factor Bradford para analizar el grado de cumplimiento horario de sus trabajadores.</p>
<p>Criterios de aceptación:</p> <p>El sistema muestra correctamente la tasa de absentismo.</p> <p>El usuario puede filtrar centro y/o departamento.</p> <p>El sistema muestra el factor Bradford de cada trabajador solo si se ha seleccionado centro y departamento.</p>

Tabla 5.8: US-006 Ver indicadores de cumplimiento horario

5.5.3. Indicadores para el análisis de competencias

La matriz de competencias se utiliza para el análisis de competencias desde el punto de vista de la gestión de proyectos y la organización de equipos. Además de la matriz se ha considerado conveniente y de utilidad realizar una serie de indicadores para visualizar cómo se distribuyen las competencias en el centro de trabajo y por los distintos departamentos por un lado y, por otro, para evaluar de modo general el nivel de las competencias. Si bien tiene cierta aplicación a la gestión de proyectos y equipos, es de más utilidad que la matriz de competencias a la hora de conocer un estado general del centro y/o del departamento.

Otra ventaja de utilizar este punto de vista, más estadístico, utilizando visualización de datos es la posibilidad de aplicar la narratividad a los datos que se explicó en el apartado anterior (5.4). Utilizando la narrativa adecuada, se puede realizar mejor el análisis competencial y obtener más lecturas que utilizando solo la matriz de competencias para este propósito.

A partir de la historia de usuario, mostrada en la Tabla 5.9, se pueden esbozar las líneas maestras de la narrativa.

US-007 Ver indicadores de competencias
<p>Descripción:</p> <p>El usuario quiere ver estadísticas de las competencias de los trabajadores para analizar las competencias que existen en la empresa y los departamentos.</p>
<p>Criterios de aceptación:</p> <p>El usuario puede seleccionar un centro y un departamento.</p> <p>El sistema muestra las competencias que existen tanto en el centro como en los departamentos del centro seleccionados por el usuario.</p> <p>El sistema muestra las competencias desglosadas por nivel para el centro y departamento seleccionado por el usuario.</p>

Tabla 5.9: US-007 Ver indicadores de competencias

5.6. Proceso de diseño y desarrollo

El desarrollo se realiza partiendo de la interfaz de usuario. Tras el análisis de los requisitos, se diseña la interfaz utilizando prototipos en papel y, a partir de la interfaz, se definen las vistas, métodos principales y sus controladores, rutas de acceso a las vistas y controladores y se modelan las clases (entidades,

controladores y otras clases). Finalizado el diseño, se prosigue con la implementación.

Esta forma de desarrollar permite enfocarse en el usuario de la aplicación y en la búsqueda de un diseño atractivo y que sea capaz de satisfacer sus necesidades. Utilizar un enfoque centrado en el usuario permite aumentar las posibilidades de éxito de la plataforma: un diseño que no sea atractivo y que no se adapte a sus necesidades (resulte difícil de utilizar, no se comprenda qué funcionalidad ofrece...) no es capaz de atraer usuarios y, por lo tanto, puede hacer fracasar al proyecto. Además, desarrollar diseñando primero la interfaz facilita el resto del proceso de diseño y su posterior implementación, pues a partir de los prototipos se identifican los elementos básicos que sirven de base para, mediante refinamiento, ir completando el diseño.

Este apartado, que tiene un carácter técnico, se aborda en detalle en el Anexo B y se complementa con el apartado 6 de la memoria. En el anexo referido se pueden consultar tanto el diseño de la interfaz y los prototipos como el diseño *software*, que incluye la arquitectura del sistema, la definición de rutas, el diseño de clases y el flujo de mensajes intercambiados entre la vista y el controlador. Además, la documentación de cada clase implementada se encuentra en el manual del programador.

6. RESULTADOS

La forma de acceder a cada apartado y el resumen de la funcionalidad que ofrece la aplicación se recoge a través del mapa de navegación de la Figura 6.1.

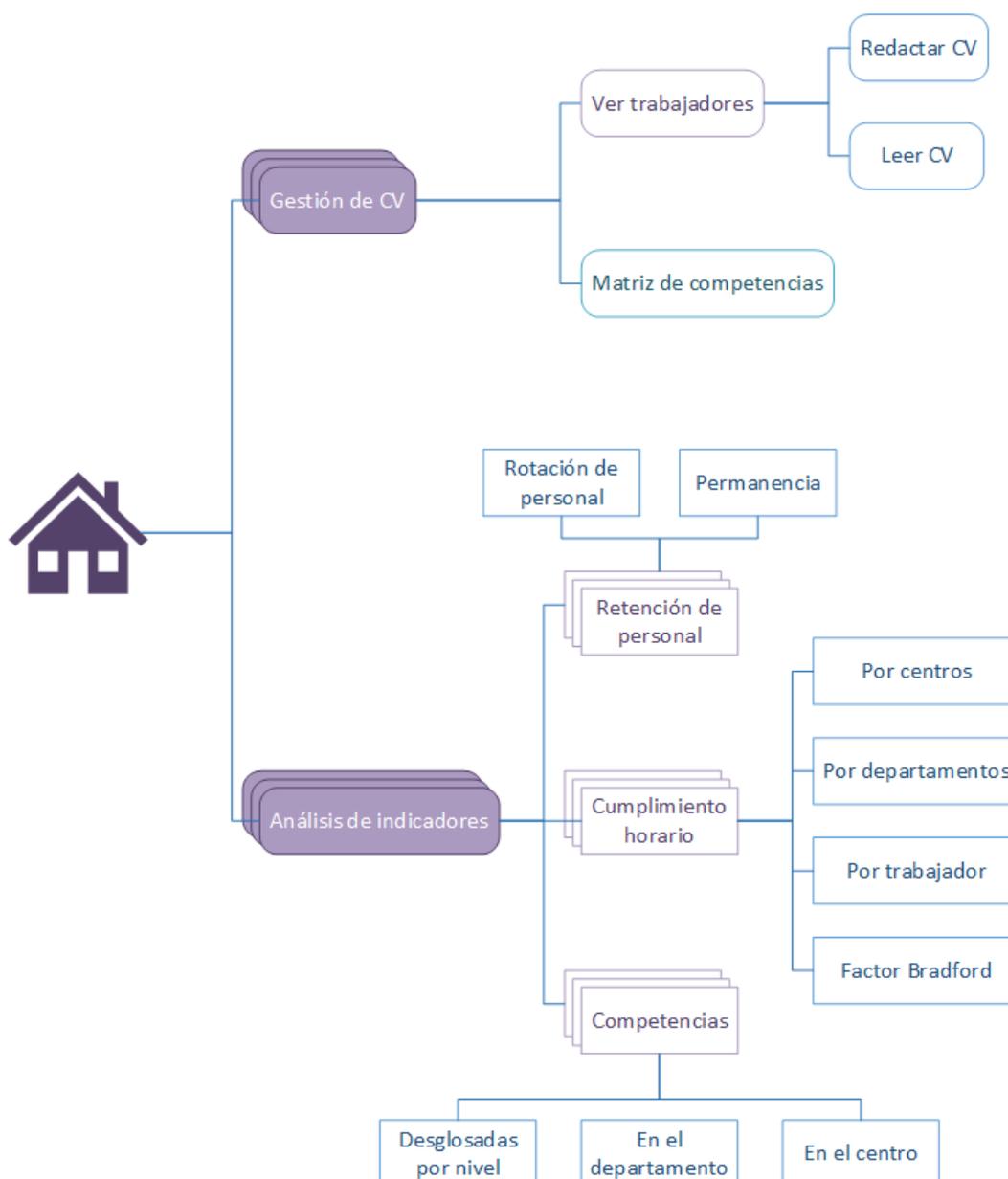


Figura 6.1: Mapa de navegación

La página de inicio muestra información básica como en el futuro podría ser una serie de anuncios con los problemas de la empresa más urgentes que la aplicación detecte (por ejemplo, el factor Bradford de los trabajadores que excedan alguno de los límites que la empresa fije). Desde ella se muestran las dos grandes categorías (elementos no navegables) de funcionalidad: la gestión de currículums y el análisis de indicadores.

La categoría de gestión de currículums permite acceder a la página con el listado de trabajadores (ver trabajadores) y la página con la matriz de competencias. A su vez, desde el listado de trabajadores se accede a las páginas de consulta y redacción del currículum del trabajador seleccionado.

La categoría de análisis de indicadores contiene una página por cada grupo de indicadores:

- Retención de personal. Página con dos elementos:
 - Rotación de personal.
 - Permanencia.
- Cumplimiento horario. Página con cuatro elementos:
 - Absentismo por centros.
 - Absentismo por departamentos.
 - Absentismo por trabajador.
 - Factor Bradford.
- Competencias. Página con tres elementos:
 - Competencias en el centro.
 - Competencias en el departamento.
 - Competencias desglosadas por nivel.

A las páginas de primer nivel se accede desde una barra lateral. Esta barra se muestra, como se puede observar en la Figura 6.2, junto a la pantalla inicial (se accede a ella tras iniciar sesión como usuario).



Figura 6.2: Vista inicial

En las páginas con varios elementos, como se verá, se ha añadido un mecanismo de navegación interno que permite un acceso rápido a cada elemento.

El mapa navegacional se caracteriza por no ser excesivamente profundo, evitando que el usuario se pierda al navegar por el sitio web, y por delimitar la amplitud a través del uso de categorías de elementos. Si bien es una versión inicial que se irá expandiendo en el futuro, no se espera añadir más de un nivel a la profundidad actual.

Antes de empezar a mostrar los resultados obtenidos, hay que señalar que para la demostración y para esta sección de la memoria se ha generado una empresa Aplifisa ficticia con cuarenta y un trabajadores repartidos en los departamentos de programación (15 trabajadores) y de atención al cliente (26). Los trabajadores han sido creados aleatoriamente (en fase de producción provendrán o bien de los datos en las aplicaciones de escritorio o bien de la conexión a la base de datos del Portal del Empleado), aunque para alguno de ellos se han modificado sus datos con el objetivo de tener más casos representables a la hora de mostrar las gráficas. Los datos de los currículums, por su parte, se generan de manera manual al utilizar la aplicación. Por último, los fichajes han sido generados automáticamente procurando que haya ausencias de diversa duración; la cantidad de fichajes y otras consideraciones a los mismos se abordarán al ver los resultados de los indicadores de cumplimiento horario.

Otro aspecto a destacar es que cada usuario de la aplicación está asociado a una empresa y una empresa puede tener varios usuarios (varios gestores de recursos humanos, directivos de la empresa...). Un usuario no puede acceder en ningún caso a otra empresa ni sus datos: Laravel gestiona a nivel interno la autenticación y permite, en cada petición, conocer qué usuario está haciendo la petición por lo que, salvo en caso de robo de cuenta (problema del usuario), consultando en la base de datos se consigue el identificador de la empresa a la que está asociado, este identificador es la base a la hora de obtener cualquier información de la empresa.

El acceso a cada página (a excepción del inicio de sesión y el registro) está protegido y requiere autenticarse mediante el inicio de sesión. En caso de acceder a una ruta protegida, se redirige al usuario a la página de inicio de sesión.

6.1. Resultados: gestión de currículums

El primer paso para empezar a gestionar los conocimientos y currículums es el acceso a la lista de empleados de la empresa. Esta lista muestra la información básica de todos los trabajadores y permite el acceso a la funcionalidad de redactar y consultar currículums (columna “acciones”).

Estado	Centro	Departamento	N.I.F.	Nombre	Fec. alta	Fec. baja	Acciones
✓	Salamanca	Programación	72584228F	Anara Gutiérrez Sobrio	19/04/2016	-	[icon] [icon]
✓	Salamanca	Programación	31337547R	Ariadna Mateo Nieves	26/03/2021	-	[icon] [icon]
✓	Salamanca	Programación	6923392V	David Cedillo Carbonell	19/02/2021	-	[icon] [icon]
✓	Salamanca	Programación	55743719H	Elena Olivo Jiménez	26/11/2012	-	[icon] [icon]
✗	Salamanca	Programación	77532381W	Jesús Lorenzo Ceballos	09/06/2020	09/07/2020	[icon] [icon]
✓	Salamanca	Programación	28606474V	Joel Meraz Gilman	27/03/2021	-	[icon] [icon]
✓	Salamanca	Programación	64022167A	Juan Matías Crespo	22/01/2014	-	[icon] [icon]
✓	Salamanca	Programación	93082205V	Lara Colón Urbina	30/09/2017	-	[icon] [icon]
✓	Salamanca	Programación	57084454B	Lucas Posada Otero	04/07/2020	-	[icon] [icon]
✓	Salamanca	Programación	23216768J	Manuel Fernández Benbez	24/07/2011	-	[icon] [icon]

Figura 6.3: Listado de trabajadores de la empresa

La tabla dispone de varias funcionalidades para agilizar la búsqueda de trabajadores:

- Por defecto, los trabajadores se ordenan por estado (no dado de alta, activo o dado de baja) y por nombre. Para ordenar por estado, siempre que no se filtre por dicho campo, se hace gracias al valor interno de cada estado (0, 1 y 2 respectivamente).
- Ordenamiento tanto ascendente como descendente al pinchar en el encabezado de la columna. Es posible ordenar por varios campos si, al hacer clic, se mantiene presionada la tecla Shift. En la figura anterior, los trabajadores están ordenados primero de manera descendente por departamento y, segundo, de manera ascendente por nombre.
- Buscar en cualquier campo utilizando el campo de texto de la esquina superior derecha de la tabla.

- Filtrar por uno o varios campos. Para este filtrado se escribe la cadena o palabra por la que filtrar en el campo de texto que aparece en el encabezado correspondiente.

Mostrar 10 registros Buscar: _____

Estado	Centro	Departamento	N.I.F.	Nombre	Fec. alta	Fec. baja	Acciones
	Buscar Centro	cliente	Buscar N.I.F.	pa	Buscar Fec. alta	Buscar Fec. baja	
✓	Salamanca	Atención al cliente	15435373G	Marco Pantoja Reséndez	22/07/2020	-	
✓	Salamanca	Atención al cliente	93567882H	Rosa Pardo Arteaga	16/09/2020	-	
✓	Salamanca	Atención al cliente	87879040L	Antonio Rodrigo Zapata	19/04/2021	-	
✓	Salamanca	Atención al cliente	71475329T	Asier Hernández Aparicio	18/12/2020	-	

Mostrando registros del 1 al 4 de un total de 4 registros (filtrado de un total de 41 registros) Anterior **1** Siguiente

Figura 6.4: Ejemplo de filtro por cadena o palabra.

- Filtro rápido por estado del trabajador (permite filtrar por varios estados simultáneamente) haciendo uso de los botones activables que aparecen en la esquina superior izquierda.

SOLO ACTIVOS **SOLO NO DADOS DE ALTA** SOLO DADOS DE BAJA

Mostrar 10 registros Bu

Estado	Centro	Departamento	N.I.F.	Nombre	Fec. alta	Fec. baja
	Buscar Centro	Buscar Departamento	Buscar N.I.F.	Buscar Nombre	Buscar Fec. alta	Buscar Fec. baja
	Salamanca	Programación	72584028F	Ainara Gutiérrez Solorio	-	-
	Salamanca	Programación	77532381W	Jesús Lorenzo Ceballos	09/06/2020	09/07/2020

Figura 6.5: Ejemplo de filtro rápido por estado del trabajador

Todas estas operaciones y la propia carga inicial de la tabla se realizan haciendo peticiones al servidor de manera asíncrona para obtener los datos a representar en la tabla, optimizando así el rendimiento de la página.

Tanto el estado del trabajador como las acciones se muestran de manera visual y se acompañan de una descripción emergente textual. Para cada caso, los iconos son los siguientes:

- Estados: marca verde dentro de círculo para el estado “activo”, marca roja tachada dentro de un círculo para el estado “dado de baja” y portafolios amarillo con una exclamación dentro para el estado “no dado de alta”. Además de la descripción emergente, la accesibilidad de los iconos mejora al no depender la comprensión de los mismos del color sino de la forma del icono.
- Acciones: ficha emulando a un currículum redactado para la consulta de currículums y lápiz para la redacción del currículum.

Para redactar y consultar el currículum de un trabajador, se hace clic en la acción correspondiente de su fila.

Al acceder a la edición de currículums se cargan todos los datos que se tengan del empleado en la base de datos (si es currículum nuevo, solo su nombre completo y lugar de residencia).

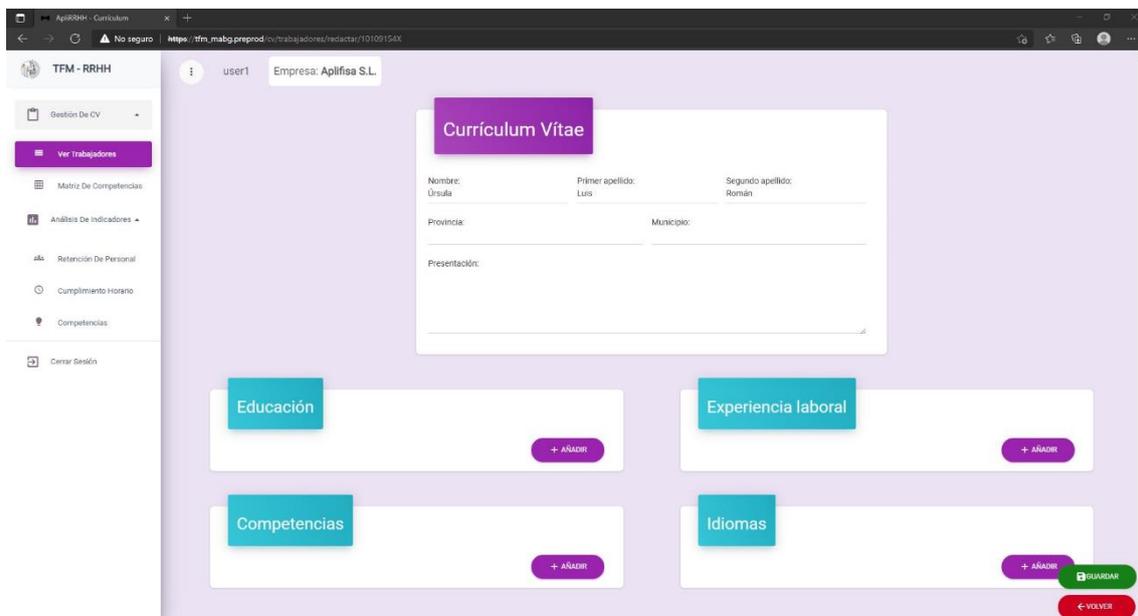


Figura 6.6: Vista inicial de la redacción del currículum

El N.I.F. de la URL se puede modificar como ayuda para usuarios avanzados a la hora de querer editar varios currículums si conocen el N.I.F. del empleado. En caso de intentar acceder, ya sea para lectura o redacción, al currículum de un trabajador que no pertenezca a la empresa del usuario (o no exista, en cuyo caso sucederá lo mismo para no dar detalles de qué trabajadores tiene registrados el sistema) se muestra un mensaje de error y se redirige al listado de empleados.

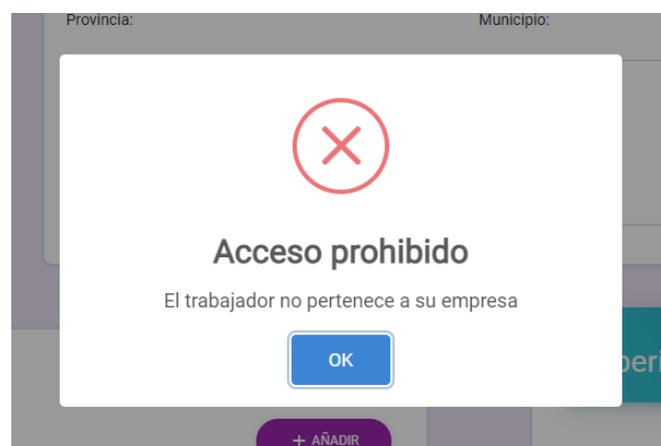


Figura 6.7: Mensaje de error al acceder al currículum de un empleado que no forma parte de la empresa

Aunque elimine este aviso editando el código desde el navegador, el usuario nunca podrá obtener ni grabar un currículum ajeno a su empresa: cuando el servidor recibe la petición valida que el trabajador pertenezca a la empresa y antes de grabar la información en la base de datos realiza también una validación previa de todos los datos (validar y grabar son dos métodos distintos en la API).

Los datos identificativos (nombre, apellidos y localización) no se pueden modificar. En caso de que falte la localización, esta se puede introducir mediante la escritura en el cuadro de texto correspondiente a la provincia o al municipio. Estos cuadros de texto, además, disponen de autocompletado: mientras el usuario escribe, muestra las provincias o municipios válidos (todas las provincias de España y todos los municipios de cada provincia) y el usuario puede seleccionar la opción adecuada. Además de facilitar la introducción de datos al usuario, disminuye la probabilidad de que introduzca información errónea.

Provincia: Salamanca	Municipio: Sala			
Presentación:	<table border="1"><tr><td>Doñinos de Salamanca</td></tr><tr><td>Salamanca</td></tr><tr><td>Valsalabroso</td></tr></table>	Doñinos de Salamanca	Salamanca	Valsalabroso
Doñinos de Salamanca				
Salamanca				
Valsalabroso				

Figura 6.8: Autocompletado de provincia/municipio

En cada sección es posible añadir tantas cajas como el usuario necesite haciendo uso de los botones “añadir” y “eliminar” presentes en las tarjetas. En las cajas de educación hay que introducir la titulación obtenida, institución que la otorga y, de manera opcional, los años de inicio y finalización (o año de finalización previsto) y una descripción para poder realizar las consideraciones se crean oportunas (qué se ha hecho, nota media...). Para la experiencia laboral lo mismo pero, en vez de la institución y la titulación, se introduce la empresa donde se estuvo y el puesto desempeñado. La Figura 6.9 muestra un ejemplo de su funcionamiento y de las validaciones que se realizan.

The image shows two side-by-side form panels. The left panel is titled 'Educación' and contains fields for 'Universidad de Salamanca', 'Grado en Ingeniería Informática', and dates '2013' and '2017'. It has a description field with the text 'Trabajo Fin de Grado: Trabajo gradual del grado graduable gradualmente.' and an 'ELIMINAR' button. The right panel is titled 'Experiencia laboral' and contains fields for 'Consultora de consultoría consultiva S.L.', 'Programador', and dates '2017' and '2016'. It has a description field with the text 'Participé como programadora en varios proyectos importantes de la empresa utilizando el lenguaje de programación Go.' and an 'ELIMINAR' button. Both panels have a '+ AÑADIR' button at the bottom right.

Figura 6.9: Detalle del proceso de añadir las titulaciones obtenidas y la experiencia laboral

Añadir competencias e idiomas es una tarea similar a la anterior: añadir/eliminar cajas en función de las necesidades e introducir los datos correspondientes. A diferencia de la formación y la experiencia laboral, en este caso se añade un selector que permite escoger el nivel (básico, intermedio o avanzado) de la competencia/idioma; otros datos relevantes (comentarios, certificaciones...) se añaden en la sección de observaciones/comentarios (de manera análoga a la tarea previa).

The image shows two side-by-side form panels. The left panel is titled 'Competencias' and contains a 'Go' field, a 'Nivel' dropdown menu set to 'Avanzado', and a 'Comentarios' field with the text 'Tres años de experiencia desarrollando aplicaciones utilizando Go. El mayor proyecto en el que lo he utilizado ha sido la creación de un optimizador de recursos empresariales mediante inteligencia artificial.' and an 'ELIMINAR' button. The right panel is titled 'Idiomas' and contains an 'Idioma' field, a 'Nivel' dropdown menu set to 'Avanzado', and an 'Observaciones IELTS' field with a dropdown menu showing 'Básico', 'Intermedio', and 'Avanzado' (highlighted). It also has an 'ELIMINAR' button. Both panels have a '+ AÑADIR' button at the bottom right.

Figura 6.10: Detalle del proceso de añadir competencias e idiomas

Como se puede observar en la Figura 6.9 y la Figura 6.10, es una interfaz completamente dinámica y que otorga al usuario mucha libertad: añadir la información que se considere necesaria, hacer comentarios a cada caja de información, eliminar las cajas creadas por error o que ya no sean necesarias, escribir comentarios o información que pueda ser relevante, etcétera.

Para terminar de redactar o bien cancelar el proceso y volver, el usuario dispone de dos botones flotantes (siempre son visibles), mostrados en la Figura 6.11.



Figura 6.11: Guardar el currículum o volver al listado de trabajadores

El botón “volver” retrocede al listado de trabajadores. Al guardar el currículum (utilizando el botón “guardar”), primero se valida el currículum y, si es válido, se guarda. Internamente, el método “guardar” de la API vuelve a validar los datos para asegurar que no se haya modificado el código ni utilizado el método de manera maliciosa. Para clarificar este mecanismo, se recomienda consultar el apartado B.2.3 del Anexo B.

Una vez guardado el currículum correctamente, se informa al usuario y se le redirige a la página de lectura de currículums.

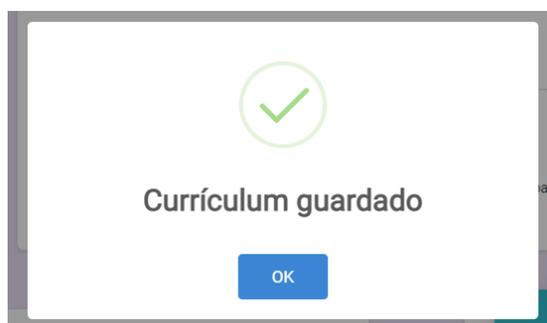


Figura 6.12: Currículum guardado con éxito

user1 Empresa: Aplifesa S.L.

Currículum Vítae

Nombre: Úrsula Primer apellido: Luis Segundo apellido: Román

Provincia: Salamanca Municipio: Salamanca

Presentación:
Soy Úrsula Luis Román, soy una mujer joven y que se toma la vida como un reto para poder mejorar.

Educación

Universidad de Salamanca
Grado en Ingeniería Informática
- Desde 2013 hasta 2017 (4 años)

Descripción:
Trabajo Fin de Grado: Trabajo gradual del grado graduable gradualmente.

Experiencia laboral

Consultora de consultoria consultiva S.L.
Programador
- Desde 2017 hasta 2019 (2 años)

Descripción:
Participé como programadora en varios proyectos importantes de la empresa utilizando el lenguaje de programación Go.

Empresa de alta calidad seguramente asegurada
Ingeniero QA
- Desde 2019 hasta ____

Competencias

Go
Nivel: avanzado

Comentarios:
Tres años de experiencia desarrollando aplicaciones utilizando Go. El mayor proyecto en el que lo he utilizado ha sido la creación de un optimizador de recursos empresariales mediante inteligencia artificial.

Idiomas

Inglés
Nivel: avanzado

Comentarios:
IELTS 8.0

← VOLVER

Figura 6.13: Lectura del currículum

Esta página de lectura, mostrada en la Figura 6.13, es una versión simplificada de la redacción: sustituye todos los cuadros de texto y selectores por párrafos y no permite agregar ni eliminar cajas. Además, muestra, para la formación académica y la experiencia laboral, la duración del estudio o trabajo. Mostrar la duración evita que el usuario la deba calcular de manera manual, reduciendo el tiempo que este dedica a leer el currículum.

Por último, en este apartado está incluida la matriz de competencias, a la que se accede desde la barra lateral.

Primero, se pide al usuario que seleccione el centro y el departamento del cual quiere generar la matriz de competencias. Una vez seleccionados, se genera la matriz de competencias. Este proceso se muestra en la Figura 6.14 y la Figura 6.15.

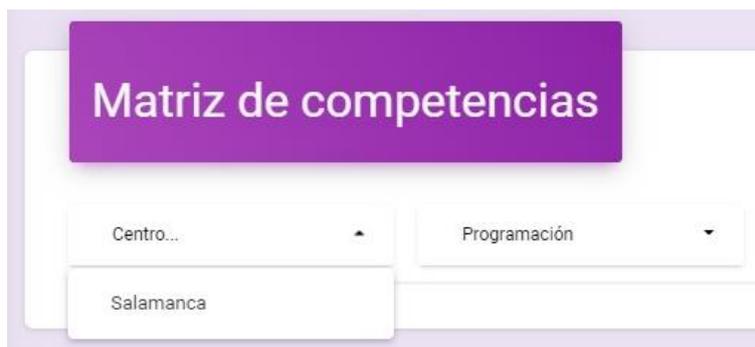


Figura 6.14: Elección de centro y departamento

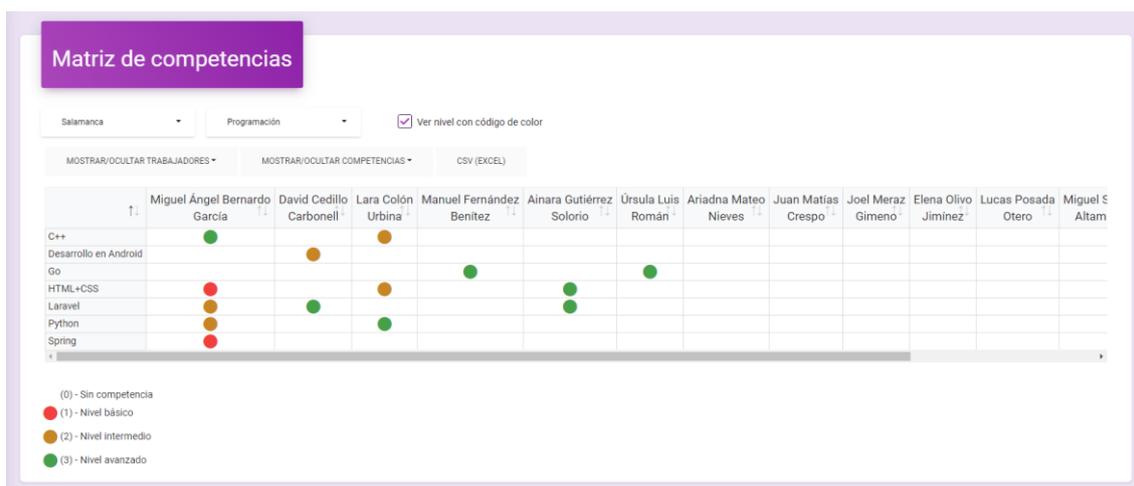


Figura 6.15: Muestra de la matriz de competencias

Internamente, al recibir la petición el servidor crea una matriz con filas las competencias (el primer elemento es el nombre y el segundo es el nivel) y columnas los nombres de los trabajadores. Esta matriz se devuelve al cliente y se representa en forma de tabla.

La matriz, tal y como se comentó en el apartado 5.3, permite al gestor de recursos humanos o de proyectos conocer el nivel de las competencias de los empleados de cara a poder asignar la persona más adecuada y balancear niveles y competencias en cada proyecto. El código de color empleado emula a un semáforo: rojo si el nivel de la competencia es básico, ámbar si es intermedio y verde si es nivel avanzado, no mostrando ningún color si el trabajador no tiene la competencia. Alternativamente, en vez de mostrar los niveles utilizando el código de color, puede usar un código numérico (activando o desactivando la casilla “Ver nivel con código de color”): 0 si no posee la competencia, 1 si el nivel de la competencia es básico, 2 si es intermedio y 3 si es nivel avanzado. Para mejorar la accesibilidad, además del código numérico, se añade una leyenda a la matriz indicando a qué nivel corresponde cada código y una descripción emergente cuando el usuario posa el ratón encima de la competencia del empleado.

Salamanca Programación Ver nivel con código de color

MOSTRAR/OCULTAR TRABAJADORES MOSTRAR/OCULTAR COMPETENCIAS CSV (EXCEL)

	Miguel Ángel Bernardo García	David Cedillo Carbonell	Lara Colón Urbina	Manuel Fernández Benítez	Ainara Gutiérrez Solorio
C++	3	0	2	0	0
Desarrollo en Android	0	2	0	0	0
HTML+CSS	1	0	2	0	3
Laravel	2	3	0	0	3
Python	2	0	3	0	0

Figura 6.16: Matriz de competencias utilizando código numérico

Para ayudar al usuario en su tarea, se han añadido a la tabla filtros de trabajadores y competencias para que pueda quitar, virtualmente, personas del proyecto o competencias innecesarias.

Matriz de competencias

Salamanca Programación Ver nivel con código de color

MOSTRAR/OCULTAR TRABAJADORES MOSTRAR/OCULTAR COMPETENCIAS CSV (EXCEL)

Miguel Ángel Bernardo García
David Cedillo Carbonell
Lara Colón Urbina
Manuel Fernández Benítez
Ainara Gutiérrez Solorio
Úrsula Luis Román
Ariadna Mateo Nieves
Juan Matías Crespo
Joel Meraz Gimeno
Elena Olivo Jiménez
Lucas Posada Otero
Miguel Saldaña Altamirano
Yago Saldiver Bueno
Rosa María Valdés Concepción

	Miguel Ángel Bernardo García	David Cedillo Carbonell	Lara Colón Urbina	Manuel Fernández Benítez	Ainara Gutiérrez Solorio	Úrsula Luis Román	Ariadna Mateo Nieves	Juan Matías Crespo	Joel Meraz Gimeno	Lucas Posada Otero
C++	●		●							
Desarrollo en Android		●								
HTML+CSS	●		●	●	●	●				
Laravel	●									
Python	●		●							

Figura 6.17: Filtrado por trabajador en la matriz de competencias

Matriz de competencias

Salamanca Programación Ver nivel con código de color

MOSTRAR/OCULTAR TRABAJADORES MOSTRAR/OCULTAR COMPETENCIAS CSV (EXCEL)

C++
Desarrollo en Android
Go
HTML+CSS
Laravel
Python
Spring

	Miguel Ángel Bernardo García	Lara Colón Urbina	Manuel Fernández Benítez	Ainara Gutiérrez Solorio	Úrsula Luis Román	Ariadna Mateo Nieves	Juan Matías Crespo	Joel Meraz Gimeno	Lucas Posada Otero
C++	●								
Desarrollo en Android		●							
HTML+CSS	●	●		●					
Laravel	●								
Python	●	●							

Figura 6.18: Filtrado por competencias en la matriz de competencias

El resto de herramientas de ayuda permiten:

- Ordenar las filas de la matriz en función de las competencias de uno o varios empleados. Para ello, se sigue el mismo procedimiento que en el listado de trabajadores.
- Mover trabajadores en la tabla pinchando y arrastrando columnas en la tabla.

	David Cedillo Carbonell	Ainara Gutiérrez Solorio	Lara Colón Urbina	Miguel Ángel Bernardo García
C++			●	●
Laravel	●	●		●
Python			●	●
HTML+CSS		●	●	●
Desarrollo en Android	●			●

Figura 6.19: Ordenar por competencias de trabajador y reordenamiento de trabajadores

En la Figura 6.19, se han ordenado las competencias en orden descendente en función de las del trabajador Miguel Ángel Bernardo García y reordenado los trabajadores según las competencias requeridas para cada sección de un hipotético proyecto que requiere competencias de desarrollo Android, desarrollo web y desarrollo de aplicaciones de escritorio.

La última funcionalidad de la matriz de competencias es la exportación a un fichero en formato CSV, funcionalidad que puede ser catalogada como parte de la categoría “generación de informes”. El fichero CSV puede ser abierto desde un programa de hojas de cálculo como Microsoft Excel y le permitiría al usuario poder hacer filtros más avanzados o crear, con ayuda de Excel, funcionalidades que no están incluidas en la aplicación. Un ejemplo de estas funcionalidades es la posibilidad de definir proyectos y crear submatrices de competencias en función del proyecto y el personal asociado a cada uno.

	David Cedillo	Ainara Gutiérrez	Lara Colón	Miguel Ángel	Manuel Fern	Úrsula Luis R	Ariadna Mat	Juan Matías	Joel Meraz	Elena Olivo	Lucas Posada	Miguel Salda	Yago Saldiva	Rosa María Valdés	Concepción
C++	0	0	2	3	0	0	0	0	0	0	0	0	0	0	0
Laravel	3	3	0	2	0	0	0	0	0	0	0	0	0	0	0
Python	0	0	3	2	0	0	0	0	0	0	0	0	0	0	0
HTML+CSS	0	3	2	1	0	0	0	0	0	0	0	0	0	0	0
Spring	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
Desarrollo en Android	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Go	0	0	0	0	3	3	0	0	0	0	0	0	0	0	0

Figura 6.20: Matriz de competencias en formato CSV abierta desde Microsoft Excel

6.2. Resultados: análisis de indicadores

Como aclaración antes de abordar este apartado, todas las gráficas y otros datos a mostrar en las páginas se cargan de manera asíncrona. Dada la alta interactividad de estas páginas, se ha optado por hacer llamadas a la API de manera asíncrona, según se vayan necesitando los datos, para evitar cargar todos los datos en el navegador del usuario.

Además y como se observará, todas las gráficas disponen de las siguientes funcionalidades:

- Resaltado del elemento sobre el que se sitúa el ratón:
 - o En la gráfica de líneas se resalta el punto aumentándole el tamaño.
 - o En la gráfica de barras se resalta la barra usando un color de tono más fuerte.
- Descripción emergente mostrando la información contenida en el elemento sobre el que se sitúa el ratón.
- Cuando hay más de un conjunto de datos, se muestra una leyenda indicando el color de cada conjunto.
- Posibilidad de mostrar u ocultar conjuntos de datos haciendo clic en su nombre en la leyenda.

6.2.1. Retención de personal

Tal y como se observa en la barra de navegación lateral, el primer grupo de indicadores son los relacionados con la retención de personal y contiene los KPI “índice de rotación de personal” (IRP) y “permanencia”.

La historia de la rotación de personal narra la evolución de este indicador durante los últimos cuatro años por lo que, al mostrar una evolución, la gráfica escogida es la de líneas.

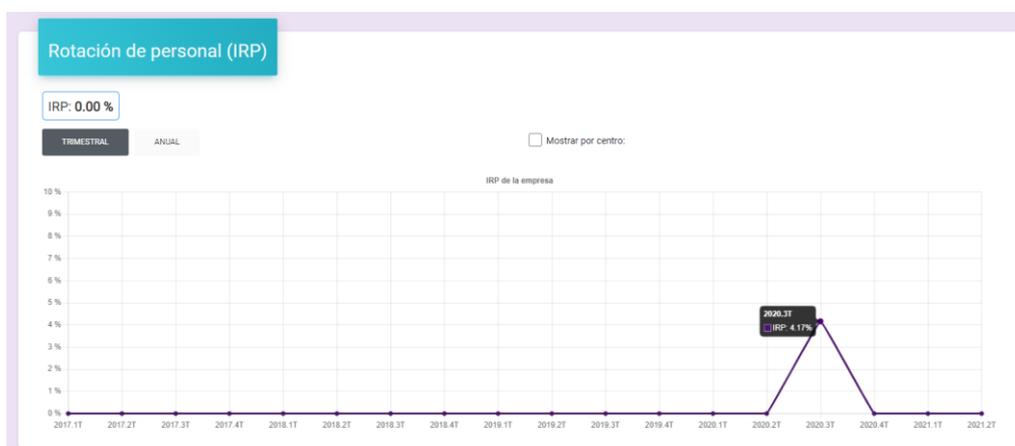


Figura 6.21: Gráfica del índice de rotación de personal trimestral

La interfaz, tal y como se aprecia en la Figura 6.21, muestra el índice de rotación de personal del periodo actual (en el caso representado, segundo cuatrimestre del año 2021, “2021.2T”) y la gráfica mostrando cómo ha evolucionado el indicador entre los últimos cuatro años (2017) y el periodo actual. La periodicidad del índice se puede cambiar de anual a trimestral y viceversa utilizando los dos botones que hay encima de la gráfica.

Dentro de la historia, además, se puede consultar qué rotación de personal ha habido en cada departamento de uno de los centros para poder analizar si es un problema general de la empresa o es un problema específico de un departamento en concreto. Para verlo, se activa la casilla “mostrar por centro” y se selecciona el centro deseado.

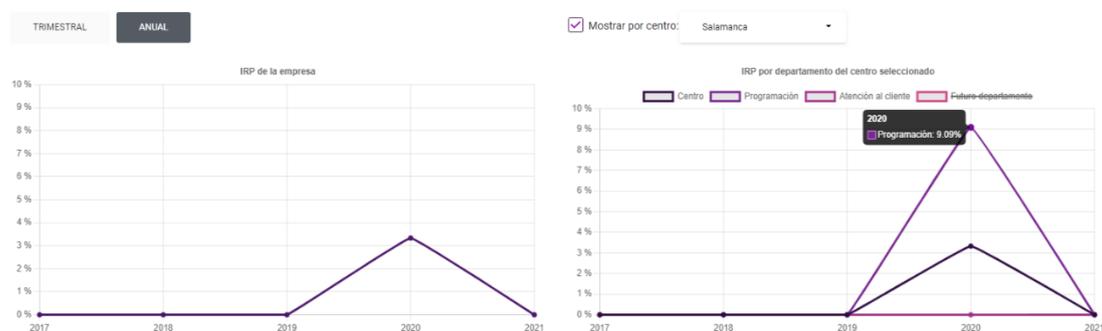


Figura 6.22: Gráfica del índice de rotación de personal anual de la empresa y los departamentos del centro seleccionado

Por su parte, la permanencia (promedio de la diferencia entre la fecha de entrada y de salida de todos los trabajadores) se ha optado por no representar una gráfica indicando su evolución y mostrar los años de media.

Para poder analizar mejor los datos de permanencia, se ha creado una gráfica que muestra la permanencia por grupos de años trabajados y qué porcentaje de empleados hay en cada franja. Para ello, se han creado diversos grupos de años de permanencia (menos de un año, uno a dos años, dos a cuatro...) y calculado los porcentajes. Esta gráfica, al mostrar la composición (reparto de la permanencia por grupos de años), se representa con una gráfica de barras y, como en el caso anterior, en la narrativa se añade posibilidad de, además, analizar la permanencia por departamentos de un centro determinado.

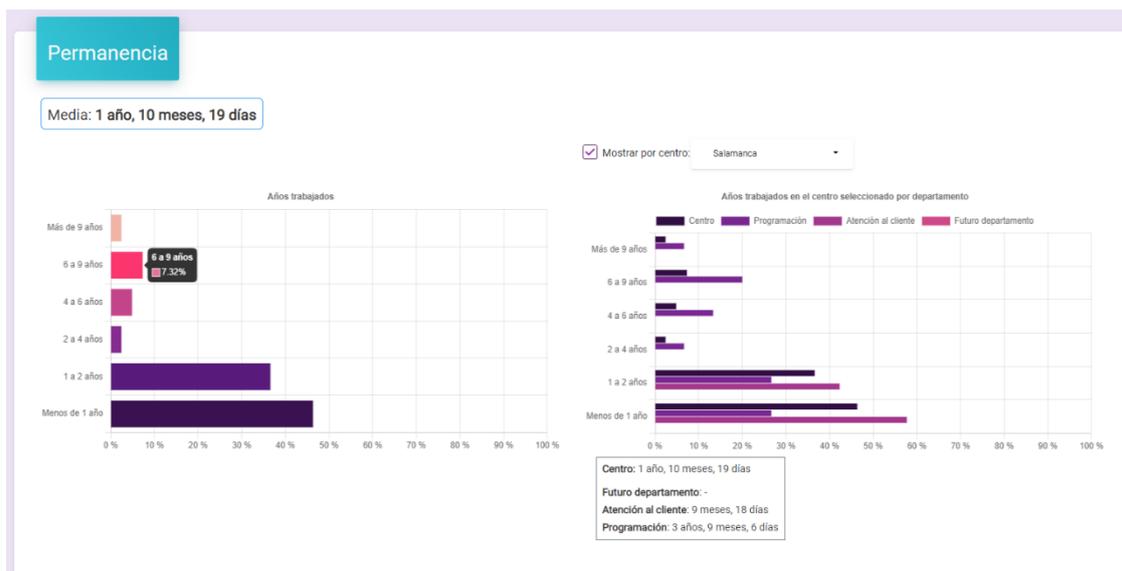


Figura 6.23: Permanencia y gráfica de permanencia por grupos de años trabajados

Además de las gráficas, en los dos casos se ha añadido un cuadro de texto indicando el valor numérico (en años, meses y días) de la permanencia. Para el caso de la permanencia en un centro, la permanencia del centro se muestra en un lugar destacado (en primer puesto y espaciada del resto) y la permanencia en los departamentos ordenada de menor a mayor (menor permanencia, mayor interés).

Como se observa en la Figura 6.23, para representar los grupos de permanencia se han creado grupos pequeños inicialmente (un año de diferencia) y después más grandes (de dos a tres años de diferencia): al analizar la permanencia, los datos más críticos son los de trabajadores que abandonan la empresa en menos de uno o dos años, por lo que los primeros grupos abarcan periodos de tiempo menores que los últimos. Además de seguir un orden lógico (menor permanencia abajo y mayor arriba), dentro de la paleta de colores, entre el grupo “menos de un año” y “uno a dos años” se han saltado varios tonos para hacer que este grupo llame más la atención al usuario, reflejando mejor la importancia que tiene dicha franja. En el caso de los años trabajados en el centro por departamentos, la referencia es el centro de trabajo, por lo que se resalta el centro.

Otro aspecto a señalar de estas dos gráficas es la representación mostrando las categorías (grupos de permanencia) en el eje de ordenadas y los porcentajes en el de abscisas, al contrario de lo que suele ser habitual. Si bien la gráfica se podría entender correctamente con los ejes en la disposición tradicional, se ha optado por usar los ejes de esta manera para que se asemeje a una pirámide de población, de forma que el usuario pueda entender mejor los datos e invierta menos tiempo para ello: al acercarse a lo que es el modelo mental del usuario, puesto que la pirámide poblacional es una forma de representación poblacional ampliamente conocida y esta gráfica tiene un significado similar a la pirámide, se facilita la comprensión de los datos.

6.2.2. Cumplimiento horario

Para el conjunto de datos empleado en este apartado, se ha generado, de manera aleatoria, un total aproximado de 21320 fichajes anuales (la cantidad varía en función de la disposición de los días del año y si el año es bisiesto o no). Estos fichajes corresponden a un entorno ideal en el que no hay festivos ni vacaciones y cada día son 8 horas laborales (40 semanales) de manera seguida, registradas con un fichaje de entrada y otro de salida. En caso de que un día el empleado falte, los fichajes de entrada y salida se realizan en la misma hora. Para tener unas horas trabajadas (diferencia entre la hora del fichaje de salida y el de entrada) más o menos realistas, se han aplicado pesos para simular que los trabajadores realizan cada día entre 0 y 8 horas, siendo las 8 horas (jornada completa) lo más probable y 0 (ausencia en el puesto de trabajo) lo menos probable. Por otro lado, los cálculos realizados se han implementado teniendo en cuenta las circunstancias de este entorno ideal y deberán ser modificados o sustituidos por los cálculos implementados en El Portal del Empleado antes de que el proyecto esté en explotación.

En la página de cumplimiento horario se muestran los indicadores “tasa de absentismo” y “factor Bradford” descritos en el apartado 5.5.2. La narrativa diseñada se basa en conocer si existe un problema de cumplimiento horario en alguno de los centros y la localización del problema, es decir, saber qué centro, qué departamento y qué trabajadores tienen problemas de absentismo para poder tomar las medidas oportunas: si es un problema de uno o varios trabajadores, apercebir a dichos trabajadores y, si es un problema más general (como que el absentismo se dé en un departamento), buscar el problema, por ejemplo, evaluando la motivación de los empleados y desarrollando estrategias para mejorarla si se asocia el problema a la motivación.

En esta narrativa, el primer paso es conocer en qué estado, en lo que a absentismo se refiere, se encuentran los centros de la empresa y cómo ha evolucionado, por mes, en el último año. Se ha escogido utilizar el año como unidad de referencia debido a que el cumplimiento horario se evalúa de manera más frecuente y muchas veces puede ser corregido en un lapso corto de tiempo (especialmente cuando requiere un apercebimiento a un empleado). Esta evolución se representa con una gráfica de líneas tal y como se muestra en la Figura 6.24.

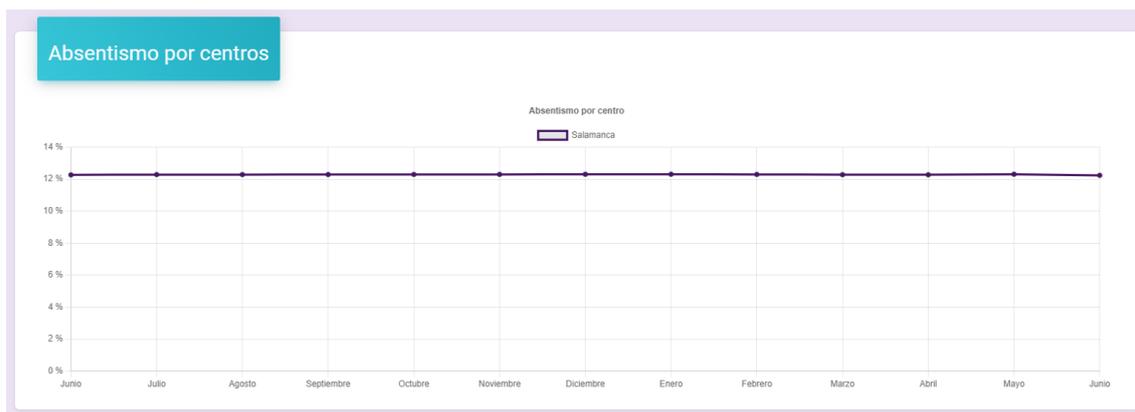


Figura 6.24: Gráfica de la tasa de absentismo por centros

En la empresa de prueba, se observa que es un absentismo con pequeñas variaciones.

Para avanzar en la historia, el usuario debe seleccionar el centro para ver el absentismo por departamentos y, si lo quiere ver por trabajadores, el departamento.

Seleccione:

- Centro para ver el absentismo por departamentos
- Centro y departamento para ver el absentismo de los trabajadores del departamento

Centro... ▲ Departamento... ▲

Figura 6.25: Elección de centro y/o departamento para ver el absentismo detalladamente

La gráfica de absentismo por departamento en el centro de trabajo seleccionado (Salamanca) se muestra en la Figura 6.26.

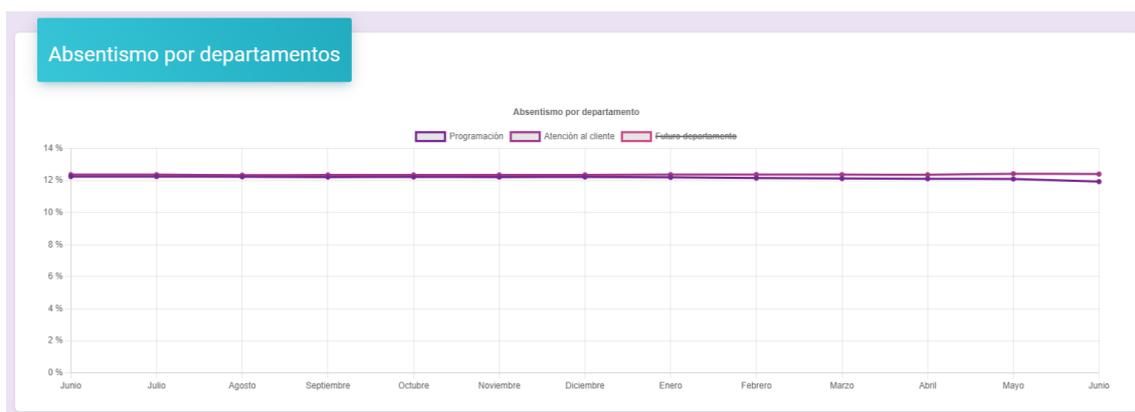


Figura 6.26: Gráfica de la tasa de absentismo por departamentos en el centro seleccionado

Esta gráfica permite analizar la situación en cada departamento. En la Aplifisa ficticia, si bien la tasa de absentismo por departamentos es pareja, se observa que en el departamento de atención al cliente es ligeramente mayor, concretamente en un 0'47%.

Dado que la tasa de absentismo mide horas trabajadas, el usuario puede convertir manualmente la tasa de absentismo en horas, que es una forma más natural de ver el impacto del absentismo. Este cálculo es trivial: se resta a las horas semanales nominales el resultado de multiplicar la tasa de absentismo por dichas horas. Aplicado a este caso y teniendo en cuenta que la jornada laboral es de 40 horas semanales, en el centro de Salamanca se trabajan 35 horas y 6 minutos mientras que, por departamento, en el departamento de programación se trabajan 35 horas y 14 minutos y, en el de atención al cliente, 35 horas y 2 minutos.

El último paso en esta narrativa es la evaluación del absentismo de los empleados de un departamento del centro. Esta evaluación se puede realizar o bien con la tasa de absentismo de cada uno o bien con el factor Bradford. La primera forma se muestra en forma de evolución con una gráfica de líneas mientras que, el factor Bradford, dada su naturaleza (ver explicación en el apartado 5.5.2), se representa en una gráfica de barras que permite comparar a los trabajadores y detectar qué trabajadores exceden cada valor de referencia.

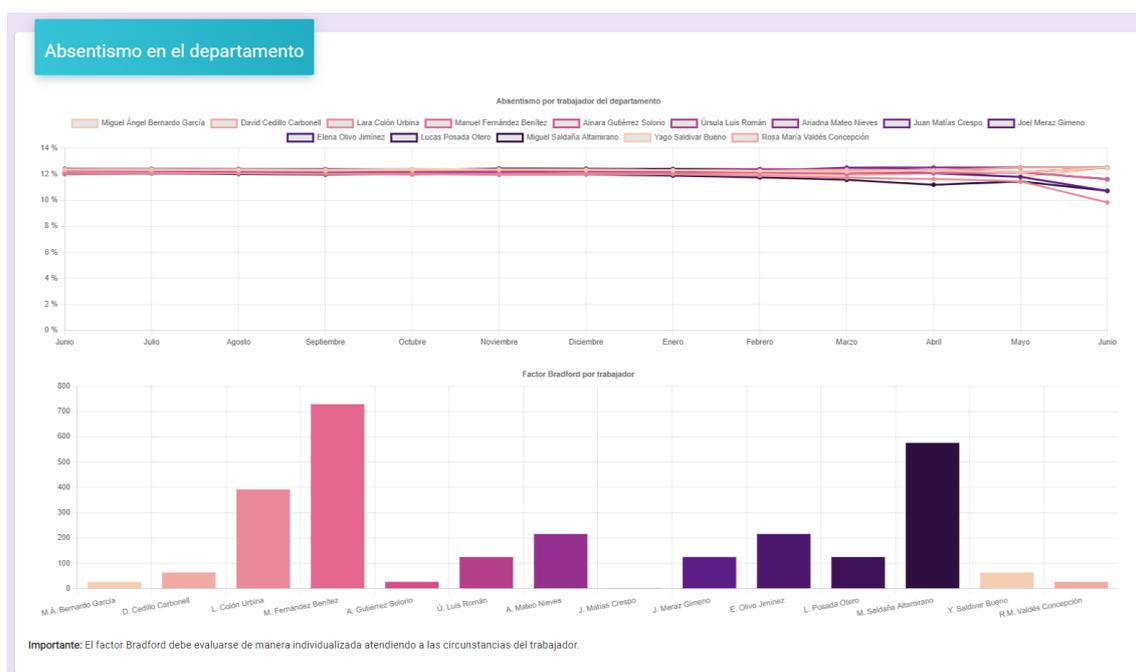


Figura 6.27: Gráficas de absentismo en un departamento

Además de las gráficas, se ha decidido añadir un recordatorio en la parte inferior de que el factor Bradford debe ser analizado de manera individual y teniendo en cuenta las circunstancias del trabajador.

En este caso, observable en la Figura 6.27, se observa que hay cuatro grupos de trabajadores en función de la tasa de absentismo y que entre el primer grupo (12'5%) y el último (9'82%) hay una diferencia del 2'68%, 1h y 4min si se convierte la tasa en horas. Analizando el factor Bradford, se observa que, en tres trabajadores, el problema de absentismo no se debe tanto al número de horas

trabajadas cada día sino a la cantidad de días completos que el trabajador se ausenta y, por lo tanto, hay un problema de absentismo que deberá ser analizado de manera individual.

Por último, en estos indicadores se observa un problema relativo al tiempo empleado por el servidor en realizar los cálculos, en especial para el factor Bradford. Debido a la cantidad de información que se analiza (aproximadamente 21320 fichajes por año para una empresa de 41 trabajadores), son indicadores que requieren un mayor tiempo de procesamiento; esto se deberá evaluar en producción o pruebas con los algoritmos y datos reales y actuar si fuera necesario para optimizar aspectos como el algoritmo implementado o el tiempo de carga de los datos desde la base de datos.

Como demostración de esto último, se evalúa la complejidad algorítmica. Para la implementación del algoritmo de cálculo del factor Bradford (ver Figura 6.28) se observa que el tiempo de cómputo depende del tiempo (latencia, l) en obtener los datos de la tabla federada, pues el número de empleados (e) y el número de días (d) no se pueden modificar (son constantes). En términos de complejidad computacional, es un problema de orden lineal en el que, para optimizarlo, habrá que reducir esta latencia o modificar el algoritmo para que la latencia no dependa de los días (ver Ecuación 3).

```

foreach ($employees as $employee) {
    $current = $start_date->copy();
    $last = $end_date->copy();
    $absent = array();
    while ($current <= $last) {
        if ($current->isWeekend()) {
            $current->addDay();
            continue;
        }
        $in = $employee->ClockIns()->whereDate("stamp", $current)->where("type", "E")->get();
        $out = $employee->ClockIns()->whereDate("stamp", $current)->where("type", "S")->get();
        if ($in[0]->stamp === $out[0]->stamp) {
            array_push($absent, true);
        } else {
            array_push($absent, false);
        }
        $current->addDay();
    }
    $spells = 0;
    for ($i = 0; $i < count($absent); $i++) {
        if ($absent[$i] === true) {
            if (($i + 1) >= count($absent) || $absent[$i + 1] !== true) {
                $spells += 1;
            }
        }
    }
    $days = count(array_keys($absent, true));
    $factor[$name] = pow($spells, 2) * $days;
}

```

Figura 6.28: Algoritmo para el cálculo del factor Bradford

$$O(l) = e(dl + 2d)$$

Ecuación 3: Complejidad del algoritmo para el cálculo del factor Bradford

Para evitar que el usuario no piense que hay un error y sepa que se están cargando los datos, en la leyenda de la gráfica se muestra un mensaje (*dataset* vacío de nombre “Cargando...”) indicando que la gráfica se está cargando y debe esperar a que la carga termine.

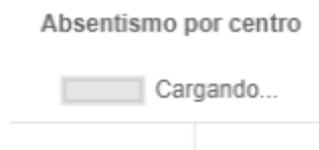


Figura 6.29: Mensaje indicando que los datos de la gráfica se están cargando

6.2.3. Competencias

Los indicadores de las competencias, porcentaje de trabajadores de un centro y/o departamento que tiene cada competencia y porcentaje de trabajadores de un centro y departamento por cada nivel de competencia, forman parte de una misma narrativa que consiste en evaluar cómo se distribuyen las competencias del centro en cada departamento, culminando en la vista de cuántos trabajadores tienen cada nivel de las competencias.

Al seleccionar el centro y el departamento (la narrativa tiene sentido cuando se analiza el departamento de un centro), se muestran las tres gráficas.

Los dos primeros indicadores, competencias en el centro (Figura 6.30) y competencias en el departamento (Figura 6.31), sirven para comparar las competencias entre sí, por lo que se representan mediante una gráfica de barras.

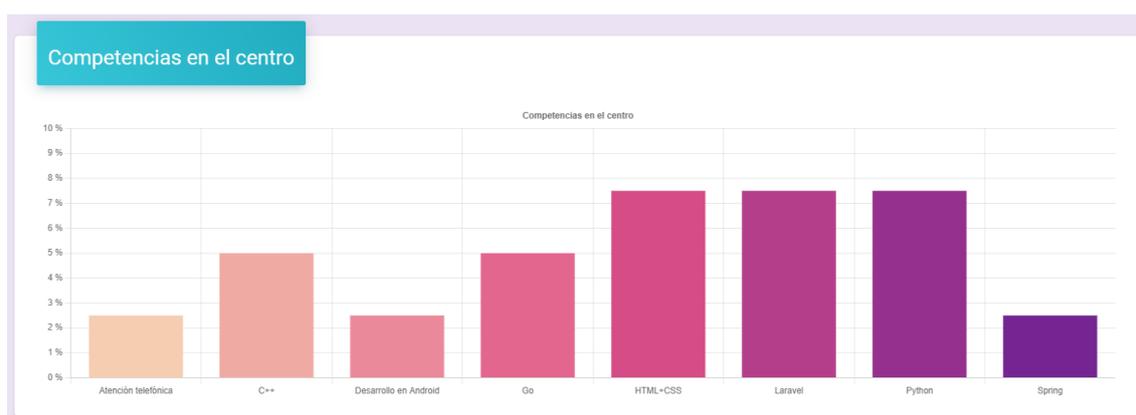


Figura 6.30: Gráfica de competencias en el centro



Figura 6.31: Gráfica de competencias en el departamento

Estos dos primeros indicadores, en el caso representado, permiten ver que hay que trabajadores en el departamento de atención al cliente que tienen competencias (Python) que pueden ser de utilidad en el departamento de programación.

El ejemplo anterior es representativo de un hecho que puede darse en un entorno real y que es uno de los principales motivos para representar las competencias de esta manera: puede haber trabajadores que sean capaces de trabajar en otros departamentos e interesar destinar dichos trabajadores a estos departamentos ya sea de forma puntual o con carácter general. Es decir: estos dos indicadores ayudan a repartir mejor los trabajadores entre los departamentos.

Por último, está el indicador de competencias desglosadas por nivel en el departamento. Este indicador permite, además de comparar la cantidad de trabajadores que tienen cada competencia, comparar por niveles, por lo que la representación será a través de una gráfica de barras apiladas.

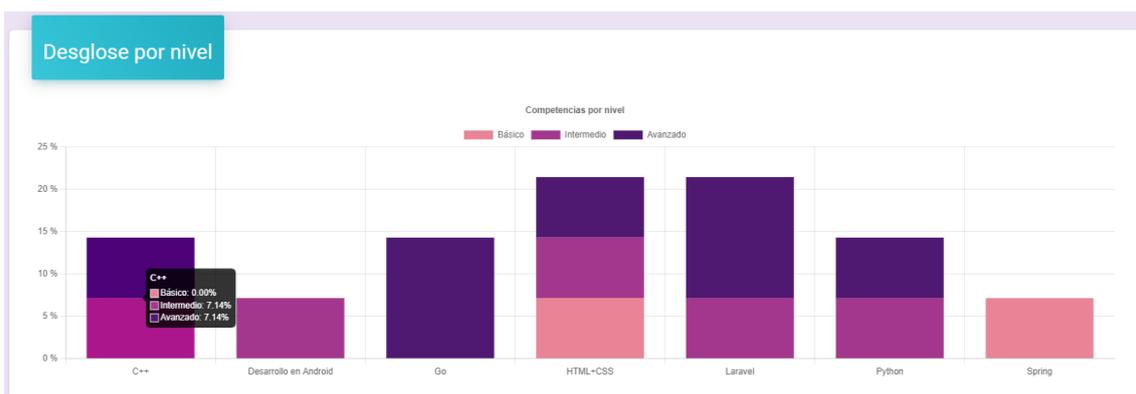


Figura 6.32: Gráfica de competencias desglosadas por nivel

Mientras que la matriz de competencias tiene un enfoque orientado a proyectos y planificación de recursos humanos, este indicador está más enfocado a diseñar acciones para, por ejemplo, mejorar las competencias de los trabajadores a través de cursos de formación impartibles a todo el departamento.

En esta representación, además de las gráficas y el espacio ocupado por cada nivel en las barras, se ha empleado un código de color para representar una metáfora que ayude a comprender mejor la información que muestra la gráfica: la base de un edificio. Los niveles se han asemejado a lo que serían las bases de un edificio, donde hay pilares poco robustos (bajo nivel de competencia) y pilares más sólidos (nivel de competencia alto). A través del tono de color se ha representado este hecho, yendo de un tono suave para el nivel básico a un tono fuerte para el nivel avanzado, pasando por un tono intermedio para el nivel intermedio. De esta forma, se ayuda a comprender en qué estado se encuentra la empresa en cuanto a las competencias de los trabajadores: si las bases son poco sólidas (los tonos más claros predominan en las barras) se requerirá implantar planes de mejora para convertir los niveles bajos en intermedios y altos, si la tonalidad general es más fuerte significa que los trabajadores de la empresa tienen las competencias a un buen nivel.

6.2.4. Generación de informes y navegabilidad entre KPI

En las páginas de análisis de indicadores se han incorporado dos utilidades: una de ayuda a la navegación en cada página y otra de generación de informes. Estas dos utilidades se pueden observar en la Figura 6.33.

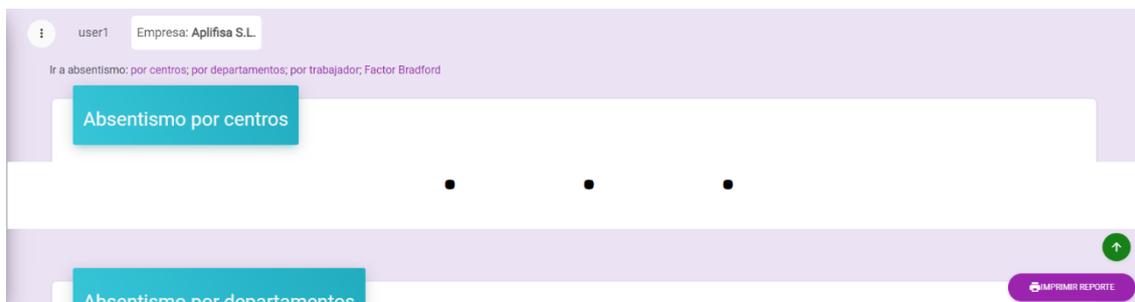


Figura 6.33: Utilidades en las páginas de análisis de indicadores

Cabe destacar que los dos botones que se muestran en la parte inferior derecha de la página son flotantes: son visibles en cualquier parte de la página.

La utilidad de navegabilidad tiene dos funciones: por un lado, el acceso rápido al indicador y, por otro, el acceso rápido a la parte superior de la página. Haciendo clic sobre el nombre del indicador se desplaza la página hacia la zona del indicador. Para volver arriba del todo y consultar otro indicador, el usuario puede pinchar sobre el botón verde con la flecha que apunta hacia arriba de la esquina inferior de la pantalla. Si bien en el estado actual del desarrollo del producto hay pocos indicadores en cada página y por lo tanto la navegabilidad en las páginas no

es muy complicada y no resulta tan necesario, conforme se añadan indicadores a las páginas sí será de gran ayuda al usuario y le facilitará el uso de las páginas.

Al igual que se realizó con la matriz de competencias, para el análisis de indicadores se ha añadido la posibilidad de generar un informe PDF que se pueda imprimir y presentar o sirva para reutilizar las gráficas en documentos propios copiándolas del fichero PDF. Esta utilidad está pensada para facilitar la labor de compartir y presentar información con los mandos directivos de la empresa. En la Figura 6.34, se enseña como ejemplo el informe de retención de personal, mostrando los distintos indicadores, el nombre del usuario que genera el informe y el nombre de la empresa.

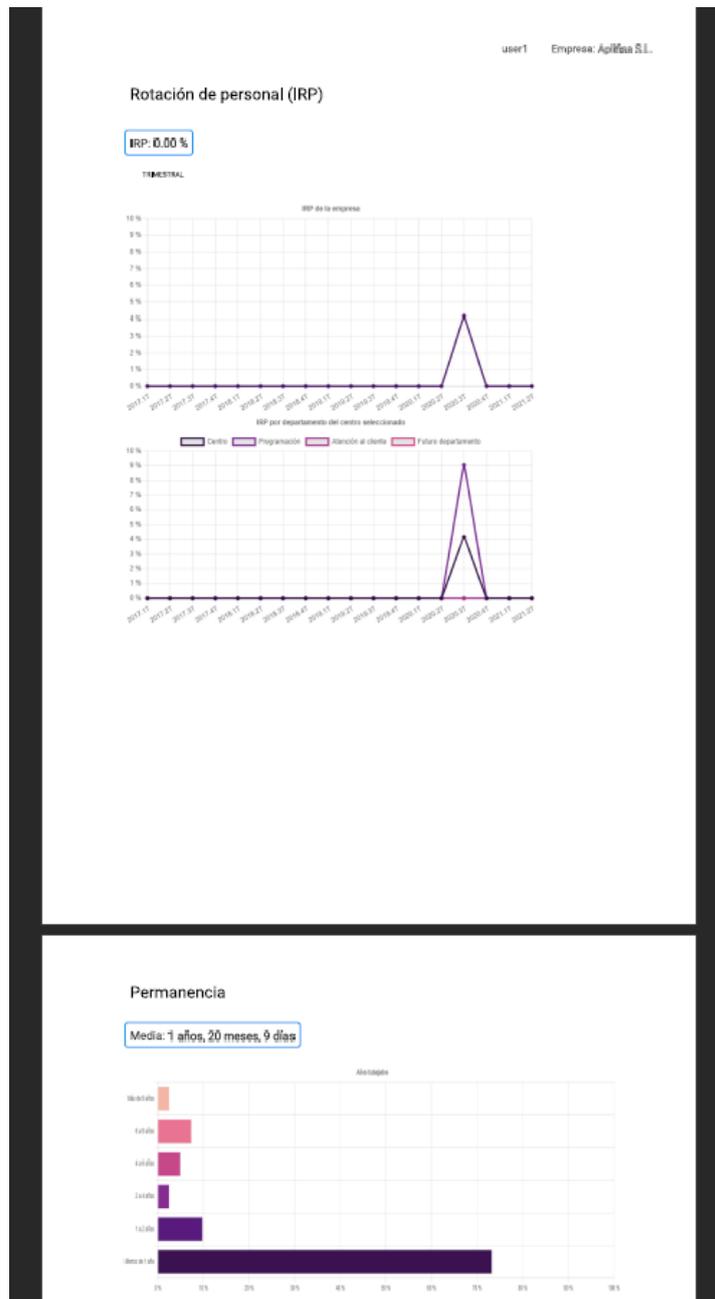


Figura 6.34: Informe de retención de personal

6.3. Métricas de rendimiento y accesibilidad

Para evaluar la calidad de las páginas desarrolladas se ha utilizado la extensión de navegador Lighthouse. Esta extensión, desarrollada por Google, realiza pruebas automatizadas para analizar y evaluar las métricas de rendimiento, accesibilidad y SEO.

En todas las páginas se ha conseguido la máxima puntuación (99-100 puntos). Aunque faltaría revisar las comprobaciones manuales y realizar pruebas con usuarios, los resultados obtenidos en las pruebas automatizadas son buenos.

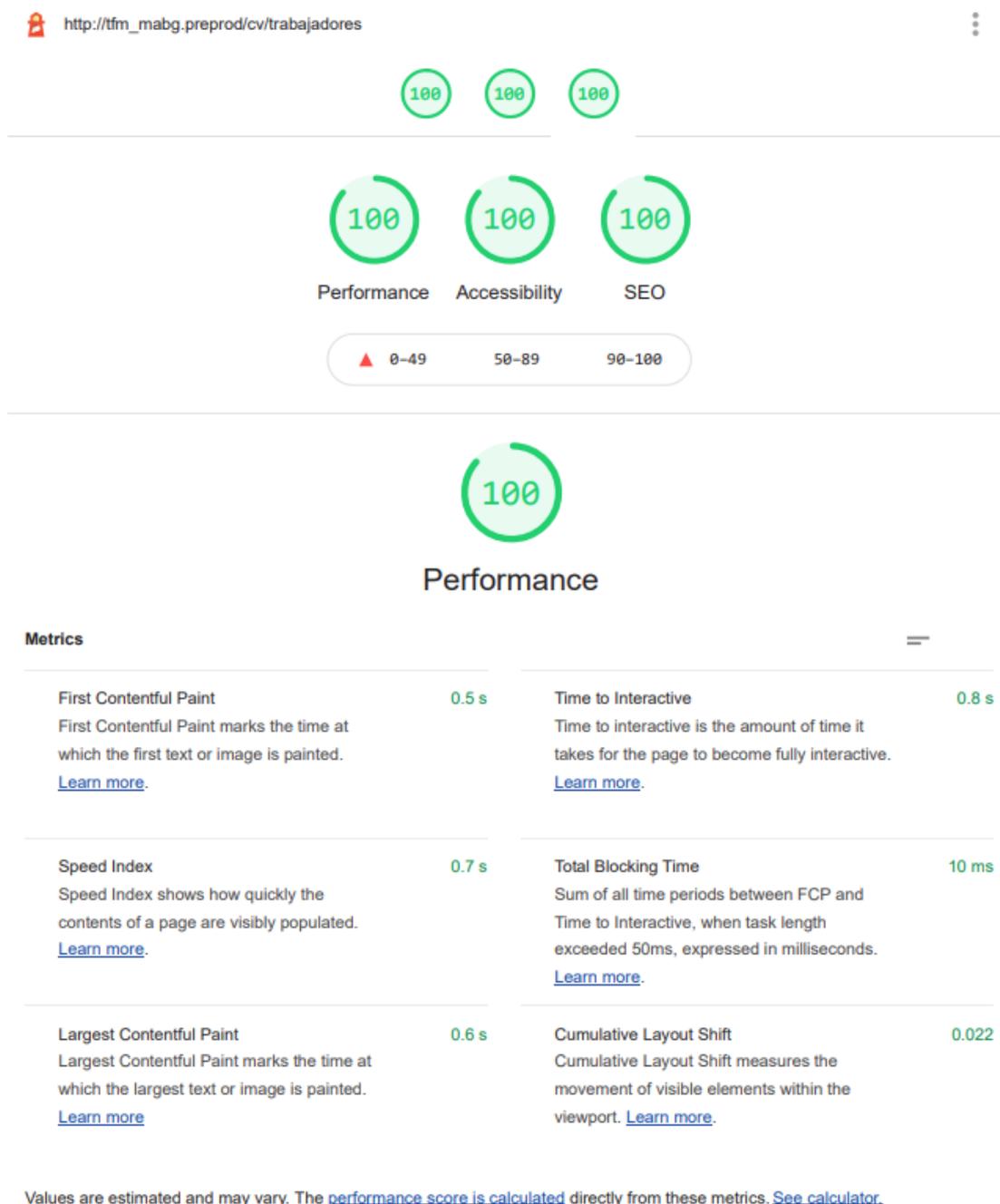


Figura 6.35: Auditoría de calidad realizada con Google Lighthouse sobre la página "Ver trabajadores"

Los resultados de la auditoría con Lighthouse se pueden consultar en el Anexo C, donde se muestran para cada página los resultados obtenidos y la lista de pruebas a realizar de forma manual.

Para un análisis más completo de la accesibilidad, pues Lighthouse recarga la página y no evalúa la accesibilidad de elementos que aparecen conforme el usuario interactúa con la página, se ha utilizado la extensión “axe DevTools” desarrollada por la empresa Deque. Esta extensión, además, permite ver en qué medida cumple con las guías de accesibilidad WCAG del World Wide Web Consortium, aunque Lighthouse también usa estas guías como referencia para sus pruebas automatizadas.

En este sentido, se ha conseguido que casi todas las páginas puedan alcanzar, como mínimo, el nivel AA. Una excepción al cumplimiento íntegro de los distintos niveles está en la matriz de competencias, como se muestra en la Figura 6.36.

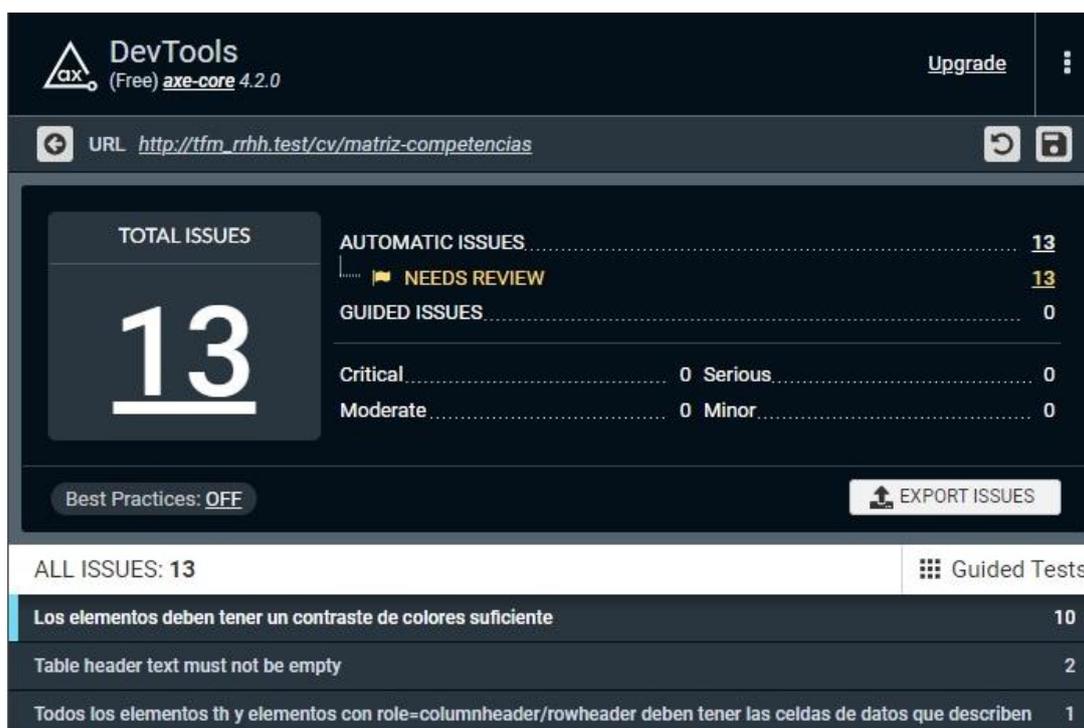


Figura 6.36: Análisis de accesibilidad de la matriz de competencias con "axe DevTools"

De las incidencias encontradas:

- Las incidencias de contraste suceden en elementos que no se han podido comprobar (degradado o elementos superpuestos). Todos los elementos que muestra ya han sido comprobados con Lighthouse, por lo que no se tienen en consideración.

- Las incidencias de encabezado de columna sin texto afectan a la columna de competencias y es una incidencia menor que se recomienda solventar. No se actúa sobre esta incidencia para dejar el encabezado de la matriz solo con los nombres.
- La última incidencia afecta a uno de los componentes que crea la biblioteca DataTables. Al ser parte de la biblioteca, sobre la incidencia no se realiza ninguna acción.

Para terminar con el análisis de la accesibilidad, hay que evaluar de manera manual los colores empleados en las gráficas debido a que “axe DevTools” no comprueba el *canvas* creado por Chart.js. Para evaluar los colores de las gráficas, se efectúa la prueba de escala de grises, consistente en convertir los colores a escala de grises y evaluar si todos los colores son discernibles.

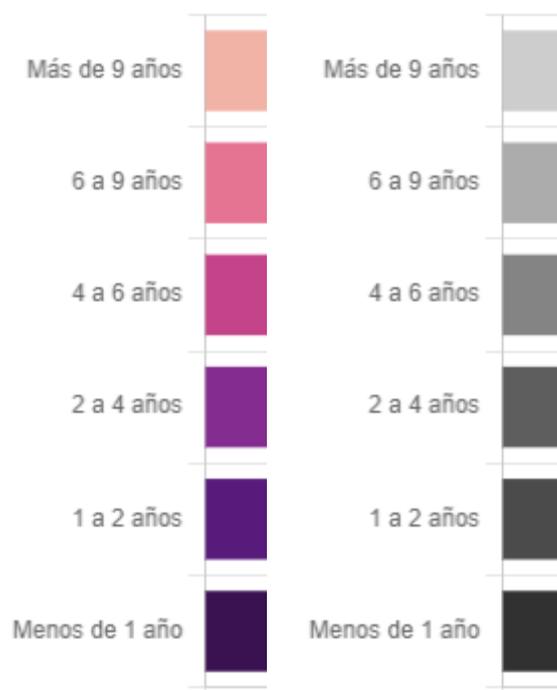


Figura 6.37: Conversión a escala de grises de la gráfica de permanencia por grupos de años trabajados

La Figura 6.37 muestra la prueba de escala de grises aplicada a una de las gráficas, donde se puede observar que sí se pueden discernir correctamente todos los colores.

7. CONCLUSIONES Y LÍNEAS DE TRABAJO FUTURAS

7.1. Conclusiones

El objetivo principal del proyecto ha sido el desarrollo de una plataforma web para la empresa Aplifisa que, basada en la inteligencia de negocio, es útil para facilitar a sus clientes diversas tareas de gestión de recursos humanos. La plataforma desarrollada, a nivel funcional, permite:

- Gestionar los currículums de los empleados para poder controlar las competencias que tiene la plantilla. Además, dispone de una matriz de competencias utilizable para ayudar a distribuir, de manera óptima, a los trabajadores en proyectos según sus conocimientos.
- Analizar los indicadores de retención de personal, cumplimiento horario y competencias para que los clientes puedan detectar problemas en sus empresas y desarrollar estrategias para atajarlos.
- Exportar a diversos formatos la información que ofrece la plataforma. Esta exportación sirve para generar informes y para que, con programas externos, puedan crear nuevas funcionalidades que les sean de utilidad.

Además, para el desarrollo de la plataforma, se ha investigado sobre la inteligencia de negocio y cómo poder implementarla, haciendo énfasis en cómo y con qué tecnologías crear el sistema de información.

Del trabajo realizado se pueden extraer dos tipos de conclusiones. Por un lado, están las consideraciones respecto al proyecto en sí y su potencial como producto de interés para los clientes y para la propia Aplifisa y, por otro lado, las relativas a mejoras que puede obtener la empresa fruto de la realización de este trabajo.

El producto desarrollado, en primer lugar, es una herramienta de gestión de recursos humanos novedosa para los clientes actuales y que es capaz de atraer a nuevos clientes, ya sean asesorías o empresas clientes finales, así como una mejora de las diversas soluciones informáticas que provee Aplifisa. Es un producto enfocado a satisfacer las necesidades de los clientes teniendo en cuenta siempre la accesibilidad y la facilidad de uso; esto permite que, además de satisfacer necesidades, se haga de una forma que ayude a facilitar el trabajo del usuario y no añada complejidad que lo dificulte.

Este proyecto, en segundo lugar, permite a Aplifisa distanciarse de lo que su competencia estándar realiza, pues estas empresas realizan productos que no

llegan a un nivel tan detallado como el nivel que se alcanza con este proyecto: si bien ofrecen lo tradicional en gestión de recursos humanos (control horario y gestión de contratos y nóminas), no llegan a ser productos que influyen tanto en el cliente como este producto, que permite tomar decisiones a medio y largo plazo y no solo en el corto plazo. Esto se aprecia especialmente en el análisis de los indicadores de retención de personal, que no ayudan tanto a resolver un problema inmediato como el mero hecho del abandono de un trabajador, sino a decidir e implantar acciones destinadas a evitar, en el medio y largo plazo, que los trabajadores abandonen la empresa.

Así como permite diferenciarse de la competencia más estándar y tradicional, ayuda y orienta a Aplifisa a cómo acercarse a sus competidores más fuertes: no solo ofertando lo mismo que ellos, como podría ser un futuro sistema de seguimiento de candidatos, sino también diferenciándose de estos competidores haciéndolo desde un punto de vista integral, que se pueda conectar con el resto de aplicaciones para ofrecer nuevas funcionalidades y además ofrezca soluciones novedosas como es lo que se ha conseguido con la gestión de proyectos desde el punto de vista de los conocimientos y competencias de los empleados a través de la matriz de competencias y los indicadores para el análisis de competencias.

Además, en tercer y último lugar de las conclusiones propias del producto, la aplicación de un enfoque de inteligencia de negocio permite construir un producto capaz de, en mayor medida, influir y facilitar los procesos de toma de decisiones mediante el uso de las técnicas de visualización de datos, generación de informes o el uso en el futuro de la inteligencia artificial y el procesamiento analítico.

A raíz de estas conclusiones, se puede afirmar que el proyecto realizado es un valor adicional para Aplifisa no solo en el corto plazo, ya sea como producto individual o añadiendo alguna de las funcionalidades desarrolladas a otros productos de la empresa, sino en el largo plazo, pues añade y permite crear nuevas líneas de productos.

Por otro lado, fuera de lo que es el producto realizado y lo que a partir de él se puede llegar a hacer, están las conclusiones obtenidas respecto a las diversas formas de trabajo utilizadas y las investigaciones realizadas para el desarrollo del proyecto:

- La inteligencia de negocio no solo es una forma de entender las necesidades de los clientes y de qué forma y con qué medios satisfacerlas, sino un marco que permite, entre otros aspectos, crear una infraestructura para el sistema de información. La infraestructura propuesta sirve de base que dé soporte a los proyectos web de Aplifisa, aspecto del que carece actualmente, pues prima una visión a corto plazo en el desarrollo.

- El uso de la herramienta Deployer es un buen avance en la implantación de una estrategia de integración y distribución continuas que permite a la empresa reducir el tiempo empleado en desplegar los cambios de los proyectos y facilitar una entrega más continuada y rápida a los clientes.
- Utilizar MySQL Workbench y el paquete de generación de migraciones permiten reducir tiempo de desarrollo, siendo interesante para la empresa su adopción. Otros cambios, como usar Chart.js en vez de Chartist, también deberían ser tenidos en consideración, pues Chart.js ha ofrecido buenos resultados visuales.
- La accesibilidad es un aspecto que Aplifisa debería tener más en consideración, pues en páginas como El Portal del Empleado Lighthouse muestra margen de mejora. Haciendo uso de las herramientas de pruebas automatizadas, así como pruebas manuales, permitiría a la empresa mejorar sus aplicaciones.

Por todos los motivos anteriormente expuestos, se concluye finalmente que se ha conseguido el objetivo principal: desarrollar una plataforma de gestión de recursos humanos, haciendo uso de la inteligencia de negocio, que dé soporte a la toma de decisiones de esta área más allá de lo que ofrecen las herramientas que tanto Aplifisa como la competencia ofrecen y con aspectos novedosos que permiten al producto diferenciarse de otros similares. Adicionalmente, se han planteado diversas sugerencias de mejora, en especial para los procesos, de gran utilidad para Aplifisa.

7.2. Líneas de trabajo futuras

El producto desarrollado no es un producto final, sino que es una base sobre la que seguir construyendo y ofrecer nuevas líneas de producto.

Como se ha visto durante el análisis de mercado realizado, hay dos líneas por las cuales seguir avanzando: el análisis de indicadores como forma de soporte a las decisiones y el sistema de seguimiento de candidatos.

En primer lugar, como apoyo a estas dos líneas de trabajo, se debe analizar la viabilidad y la necesidad de expandir la arquitectura del sistema de información. Dentro de esta expansión, se incluye el uso de tecnologías como Apache Hive para crear un *data warehouse* y poder utilizar herramientas de análisis y exploración de los datos OLAP como Apache Kylin, facilitando además el acceso a los datos, en especial en lo que a velocidad se refiere. Estas mejoras, que son mejoras tanto técnicas como funcionales, permiten seguir ahondando en la inteligencia de negocio.

Respecto a la línea de análisis de indicadores, se propone avanzar en estas líneas derivadas:

- Implementar un sistema automatizado de realización de encuestas a los empleados. Estas encuestas, además de evaluar la respuesta de los empleados a los planes de acción de la empresa, permiten ofrecer nuevos indicadores; entre otros, indicadores para el análisis de la motivación del personal.
- Usar herramientas de análisis y exploración de los datos e inteligencia artificial para ofrecer nuevos indicadores y otras funcionalidades.
- Avanzar en la gestión de proyectos, por ejemplo, desarrollando un sistema de gestión de tareas u objetivos que permita tanto organizar estas tareas como crear indicadores de rendimiento.
- Analizar la posibilidad de utilizar datos económicos para optimizar costes haciendo uso de la inteligencia artificial y crear nuevos indicadores para analizar el rendimiento de los empleados.

De la línea del sistema de seguimiento de candidatos, la propuesta de líneas derivadas es la siguiente:

- Mejorar y añadir nuevos campos al currículum:
 - o Adjuntar documentación adicional como podría ser un currículum en formato PDF o un portfolio.
 - o Añadir como nuevos campos la foto del candidato, certificaciones obtenidas y la formación recibida. Esta formación va desde formación adicional (cursos en línea) hasta formación impartida en la empresa (cursos de formación bonificada para empresas).
- Permitir la integración de la gestión de currículums en los procesos de reclutamiento de los clientes, pudiendo personas externas a la empresa del cliente subir sus currículums y postularse como candidatos.
- Como alternativa a la integración anterior, permitir que un usuario ajeno a la empresa pueda subir su currículum al sistema utilizando directamente el producto fruto del presente proyecto.
- Desarrollar el sistema de seguimiento de candidatos. Este sistema permitirá gestionar todo el proceso de reclutamiento: desde el postulado del candidato hasta su alta en la empresa, incluyendo la planificación y gestión de entrevistas y la comunicación con los candidatos.
- Dentro de la línea anterior, implementar un sistema de filtrado de currículums haciendo uso de tecnologías como el procesamiento de lenguaje natural.

REFERENCIAS

- [1] University of Minnesota, «1.1 What Is Human Resources? – Human Resource Management,» 2011. [En línea]. Available: <https://open.lib.umn.edu/humanresourcemanagement/chapter/1-1-what-is-human-resources/>.
- [2] Boletín Oficial del Estado, «Disposición 3481 del BOE núm. 61 de 2019,» 12 Marzo 2019. [En línea]. Available: <https://www.boe.es/boe/dias/2019/03/12/pdfs/BOE-A-2019-3481.pdf>.
- [3] K. W. Chan y R. Mauborgne, «Blue Ocean Strategy,» 2005.
- [4] Aplifisa S.L., «▷ Asesorías y despachos: programas de gestión | Aplifisa.com,» [En línea]. Available: <https://www.aplifisa.com/asesorias/>.
- [5] NCS Software, «NCS Software,» [En línea]. Available: <https://www.ncs.es/>.
- [6] Wolters Kluwer N.V., «Software online de gestión - a3innuva | Wolters Kluwer | Wolters Kluwer,» [En línea]. Available: <https://www.wolterskluwer.com/es-es/solutions/a3innuva>.
- [7] Next Generation Fintech Services, S.L., «Software para asesorías - next-bs,» [En línea]. Available: <https://www.next-bs.com/next-para-asesores/>.
- [8] Next Generation Fintech Services, S.L., «Software para pymes - next-bs,» [En línea]. Available: <https://www.next-bs.com/next-software-para-pymes/>.
- [9] Sage Spain, «Sage X3 - Sage Enterprise Intelligence | Sage España,» [En línea]. Available: <https://www.sage.com/es-es/sage-business-cloud/sage-x3/soluciones-complementarias/sage-enterprise-intelligence/>.
- [10] R. Kelly, «Top Applicant Tracking Systems (ATS Software) in 2020 | Ongig Blog,» 18 Noviembre 2020. [En línea]. Available: <https://blog.ongig.com/applicant-tracking-system/top-ats-systems-used-by-the-fortune-500-2019/>.

- [11] J. Shields, «Taleo: 4 Ways the Most Popular ATS Ranks Your Job Application,» 8 Marzo 2018. [En línea]. Available: <https://www.jobscan.co/blog/taleo-popular-ats-ranks-job-applications/>.
- [12] NLTK Project, «Natural Language Toolkit — NLTK 3.6.2 documentation,» 20 Abril 2021. [En línea]. Available: <https://www.nltk.org/>.
- [13] R. Řehůřek, «similarities.docsim – Document similarity queries — gensim,» 29 Abril 2020. [En línea]. Available: <https://radimrehurek.com/gensim/similarities/docsim.html>.
- [14] M. N. Moreno García, *Introducción a la inteligencia de negocio*, 2020.
- [15] I. Abramson, *Data Warehouse: The Choice of Inmon versus Kimball*, 2004.
- [16] PHP Group, «PHP: Prefacio - Manual,» 6 Junio 2021. [En línea]. Available: <https://www.php.net/manual/es/preface.php>.
- [17] Laravel LLC, «Installation - Laravel - The PHP Framework For Web Artisans,» 8 Diciembre 2020. [En línea]. Available: <https://laravel.com/docs/8.x>.
- [18] W3Schools, «Introduction to HTML,» [En línea]. Available: https://www.w3schools.com/html/html_intro.asp.
- [19] W3Schools, «CSS Introduction,» [En línea]. Available: https://www.w3schools.com/css/css_intro.asp.
- [20] Sass, «Sass: Documentation,» [En línea]. Available: <https://sass-lang.com/documentation>.
- [21] Mozilla, «JavaScript | MDN,» [En línea]. Available: <https://developer.mozilla.org/es/docs/Web/JavaScript>.
- [22] P. Cook, «D3 or Chart.js for Data Visualisation?,» 22 Enero 2019. [En línea]. Available: <https://www.createwithdata.com/d3js-or-chartjs/>.
- [23] StackShare, Inc., «What are the best Charting Libraries Tools?,» [En línea]. Available: <https://stackshare.io/charting-libraries>.

- [24] Creative Tim, «Material Dashboard: Free Bootstrap 4 Material Admin @ Creative Tim,» 2020. [En línea]. Available: <https://www.creative-tim.com/product/material-dashboard>.
- [25] J. M. Alarcón Aguíñ, «¿Qué diferencia hay entre Docker (Contenedores) y Máquinas virtuales (VMWare, VirtualBox...)? | campusMVP.es,» 14 Junio 2018. [En línea]. Available: <https://www.campusmvp.es/recursos/post/que-diferencia-hay-entre-docker-contenedores-y-maquinas-virtuales.aspx>.
- [26] Laradock, «Introduction - Laradock,» 25 Mayo 2021. [En línea]. Available: <https://laradock.io/introduction/>.
- [27] Apache Friends, «XAMPP Installers and Downloads for Apache Friends,» [En línea]. Available: <https://www.apachefriends.org/index.html>.
- [28] Deployer, «Deployer – A deployment tool for php,» [En línea]. Available: <https://deployer.org/#what-is-deployer>.
- [29] MySQL, «MySQL :: MySQL 5.7 Reference Manual :: 15.8.1 FEDERATED Storage Engine Overview,» [En línea]. Available: <https://dev.mysql.com/doc/refman/5.7/en/federated-description.html>.
- [30] K. Loong y B. Breytenbach, «GitHub - kitloong/laravel-migrations-generator: Laravel Migrations Generator: Automatically generate your migrations from an existing database schema.,» 1 Noviembre 2020. [En línea]. Available: <https://github.com/kitloong/laravel-migrations-generator>.
- [31] J. Appelo, «Team Competency Matrix - Management 3.0 Practice,» 14 Mayo 2015. [En línea]. Available: <https://management30.com/practice/competency-matrix/>.
- [32] J. Gulbis, «Data Visualization – How to Pick the Right Chart Type?,» 1 Marzo 2016. [En línea]. Available: <https://easybi.com/blog/data-visualization-and-chart-types>.
- [33] Infogram, «How to Choose the Right Chart for Your Data,» 19 Junio 2017. [En línea]. Available: <https://infogram.com/page/choose-the-right-chart-data-visualization>.

- [34] Ferdio, «Data Viz Project | Collection of data visualizations to get inspired and finding the right type.,» 2 Marzo 2010. [En línea]. Available: <https://datavizproject.com/>.
- [35] S. McGuire, «How to Choose the Best Colors For Your Data Charts,» 19 Octubre 2016. [En línea]. Available: <https://www.lifehack.org/484519/how-to-choose-the-best-colors-for-your-data-charts>.
- [36] S. Woods, «Information for Grant Writers and Data Users,» 17 Agosto 2010. [En línea]. Available: <https://digitalcommons.unomaha.edu/cgi/viewcontent.cgi?article=1017&context=datausers>.
- [37] G. T. Doran, «There's a S.M.A.R.T. way to write management's goals and objectives,» vol. 70, n° 11, pp. 35-36, 1981.
- [38] D. Sull y C. Sull, «With Goals, FAST Beats SMART,» 5 Junio 2018. [En línea]. Available: <https://sloanreview.mit.edu/article/with-goals-fast-beats-smart/>.
- [39] J. C. McElroy, P. C. Morrow y S. N. Rude, «Turnover and organizational performance: A comparative analysis of the effects of voluntary, involuntary, and reduction-in-force turnover.,» *Journal of Applied Psychology*, vol. 86, n° 6, pp. 1294-1299, 2001.
- [40] A. Zojceska, «HR Metrics: How and Why to Calculate Employee Turnover Rate? | TalentLyft,» 16 Diciembre 2018. [En línea]. Available: <https://www.talentlyft.com/en/blog/article/242/hr-metrics-how-and-why-to-calculate-employee-turnover-rate>.
- [41] D. Ofosuhene Kwenin, S. Muathe y R. Nzulwa, «The influence of employee rewards, human resource policies and job satisfaction on the retention of employees in Vodafone Ghana Limited,» *European Journal of Business and Management*, vol. 5, n° 12, pp. 13-20, 2011.
- [42] R. Karsan, «Calculating the cost of turnover,» *Employment Relations Today*, vol. 34, n° 1, pp. 33-36, 2007.
- [43] H. Gunawan y R. Amalia, «Wages and Employees Performance: The Quality of Work Life as Moderator,» *International Journal of Economics and Financial Issues*, vol. 5, n° 1, pp. 349-353, 2015.

- [44] B. Chignell, «Understanding Bradford factor scores - CIPHR,» 23 Abril 2018. [En línea]. Available: <https://www.ciphr.com/advice/understanding-the-bradford-factor-score/>.
- [45] O. Mikulec y A. Špačková, «Assessing and Managing Absenteeism with Bradford Factor Score,» *European Financial Systems 2017*, pp. 70-77, 2017.

Los enlaces de todas las referencias han sido consultados por última vez el día 1 de septiembre del año 2021.

ANEXO A. ESPECIFICACIÓN DE REQUISITOS

El presente anexo define los objetivos y funcionalidades que ofrece el proyecto. Para definirlos, se seguirá la siguiente estructura:

- Participantes en el proyecto.
- Catálogo de requisitos:
 - o Requisitos generales del sistema u objetivos:
 - o Requisitos funcionales: funcionalidades ofrecidas por el sistema.
 - o Requisitos de información: información necesaria para satisfacer los requisitos funcionales.
 - o Requisitos no funcionales: criterios a seguir y evaluar necesarios para obtener un producto de calidad.
- Matriz de rastreabilidad: identificación de objetivos con los requisitos funcionales.

A.1. Participantes

Participante	Miguel Ángel Bernardo García
Organización	<i>Aplifisa S.L., Universidad de Salamanca</i>
Rol	<i>Desarrollador</i>
Es desarrollador	<i>Sí</i>
Es cliente	<i>No</i>
Es usuario	<i>No</i>
Comentarios	-

Tabla A.1: Participante autor

Participante	Ana Belén Gil González
Organización	<i>Universidad de Salamanca</i>
Rol	<i>Tutor</i>
Es desarrollador	<i>No</i>
Es cliente	<i>No</i>
Es usuario	<i>No</i>
Comentarios	-

Tabla A.2: Participante tutor 1

Participante	Laureano Ayuso Gómez
Organización	<i>Aplifisa S.L.</i>
Rol	<i>Tutor</i> <i>Representante de la empresa</i>
Es desarrollador	<i>No</i>
Es cliente	<i>No</i>
Es usuario	<i>No</i>
Comentarios	-

Tabla A.3: Participante tutor 2

La labor de cada rol es la siguiente:

- Desarrollador del proyecto: encargado de realizar y gestionar el proyecto.
- Tutor: supervisión y validación del proyecto.
- Representante de la empresa: aprueba y valida, por parte de la empresa, el proyecto.

A.2. Catálogo de requisitos

A.2.1. Requisitos generales del sistema

El sistema debe alcanzar los siguientes objetivos:

OBJ-001	Gestión de empleados y currículums
Descripción	<i>El sistema deberá gestionar los currículums de los empleados, que serán introducidos por el usuario.</i>
Subobjetivos	-
Importancia	<i>Alta</i>
Prioridad	<i>Vital</i>
Comentarios	-

Tabla A.4: OBJ-001 Gestión de empleados y currículums

OBJ-002	Gestión de competencias
Descripción	<i>La herramienta permitirá gestionar las competencias de los empleados de forma que el usuario pueda identificar el estado de los conocimientos de los empleados.</i>
Subobjetivos	-
Importancia	<i>Alta</i>
Prioridad	<i>Vital</i>
Comentarios	-

Tabla A.5: OBJ-005 Gestión de competencias

OBJ-003	Análisis de indicadores
Descripción	<i>El sistema permitirá al usuario analizar diversos indicadores clave de negocio propios de la gestión de recursos humanos. Para ello, se deberán utilizar métodos de visualización gráficos, en los casos en los que sea posible, y se deberá facilitar la comprensión de los datos.</i>
Subobjetivos	-
Importancia	<i>Alta</i>
Prioridad	<i>Vital</i>
Comentarios	-

Tabla A.6: OBJ-003 Análisis de indicadores

A.2.2. Requisitos funcionales

La documentación de los requisitos funcionales se realiza mediante historias de usuario. Se ha optado por el uso de historias de usuario frente a los casos de uso debido a la naturaleza del proyecto y la forma en la que los requisitos han sido capturados: al ser la ideación e inicio de una nueva línea de producto, los requisitos funcionales se han centrado en qué necesidades del usuario satisfacer, sin entrar en detalle en la forma exacta en la que el usuario interactúa con el sistema.

Para todas las historias de usuario, se ha identificado el actor Usuario. Este usuario puede ser un directivo de cualquier nivel de la empresa, gestor de recursos humanos o un gestor de proyectos. Todo el apartado de gestión de usuarios (tipos de usuario, permisos de acceso al sistema y de acceso a cada funcionalidad, etcétera), es ajeno al proyecto (son procedimientos y sistemas propios de Aplifisa), por lo que el rol de cada usuario se abstraerá y se considerará a todos como “usuario”.

US-001 Listar trabajadores
<p>Descripción:</p> <p>El usuario quiere visualizar los trabajadores de su empresa para poder ver su estado e información básica, así como para poder escoger uno y redactar o consultar su currículum.</p>
<p>Criterios de aceptación:</p> <p>El usuario puede visualizar todos los trabajadores de su empresa. De cada trabajador de su empresa, puede ver su: estado (activo, no dado de alta o dado de baja), centro de trabajo y departamento al que está adscrito, N.I.F., nombre completo y su fecha de alta y de baja en la empresa.</p> <p>El usuario puede filtrar y buscar trabajadores por cualquiera de los campos.</p> <p>El usuario puede, a partir del listado, redactar o consultar el currículum de los trabajadores.</p>

Tabla A.7: US-001 Listar trabajadores

US-002 Redactar currículum

Descripción:

El usuario quiere redactar el currículum de un trabajador de su empresa para poder tener registrada en la aplicación su información básica (nombre, apellidos y localización), presentación, estudios académicos, experiencia laboral, competencias y los idiomas que conoce y su nivel en ellos.

Criterios de aceptación:

El usuario puede introducir datos en cada campo.

El usuario puede añadir y eliminar registros de cada sección del currículum.

El sistema carga en los campos la información del currículum que tenga guardada.

El usuario no puede modificar la información básica de un trabajador desde la aplicación porque esta información proviene de la aplicación de escritorio.

El usuario no puede redactar el currículum de un trabajador que no pertenezca a su empresa.

Tabla A.8: US-002 Redactar currículum

US-003 Consultar currículum

Descripción:

El usuario quiere leer el currículum de un trabajador para ver la información de dicho trabajador.

Criterios de aceptación:

El sistema carga en los campos la información del currículum que tenga guardada.

El usuario no puede modificar ningún campo del currículum.

El usuario no puede consultar el currículum de un trabajador que no pertenezca a su empresa.

Tabla A.9: US-003 Consultar currículum

US-004 Crear matriz de competencias**Descripción:**

El usuario quiere ver las competencias de los trabajadores en forma de matriz de competencias para gestionar mejor los diversos equipos de la empresa.

Criterios de aceptación:

El sistema muestra correctamente el nivel de las competencias de los trabajadores.

El usuario puede filtrar por competencias.

El usuario puede filtrar por trabajadores.

El usuario puede mover de posición en la matriz a los trabajadores.

El usuario puede seleccionar a qué centro y a qué departamento pertenecen los trabajadores a mostrar en la matriz de competencias.

Tabla A.10: US-004 Crear matriz de competencias

US-005 Ver indicadores de retención de personal**Descripción:**

El usuario quiere ver los indicadores de rotación de personal y permanencia media para evaluar la capacidad de la empresa de retener al personal.

Criterios de aceptación:

El sistema muestra el índice de rotación de personal.

El sistema muestra la permanencia media y el porcentaje de empleados en cada franja de permanencia en la empresa.

El usuario puede filtrar por centro para consultar cada indicador por departamento de la empresa.

El usuario puede seleccionar si desea visualizar la rotación de personal de manera anual o trimestral.

Tabla A.11: US-005 Ver indicadores de retención de personal

US-006 Ver indicadores de cumplimiento horario

Descripción:

El usuario quiere ver los indicadores de tasa de absentismo y factor Bradford para analizar el grado de cumplimiento horario de sus trabajadores.

Criterios de aceptación:

El sistema muestra correctamente la tasa de absentismo.

El usuario puede filtrar centro y/o departamento.

El sistema muestra el factor Bradford de cada trabajador solo si se ha seleccionado centro y departamento.

Tabla A.12: US-006 Ver indicadores de cumplimiento horario

US-007 Ver indicadores de competencias

Descripción:

El usuario quiere ver estadísticas de las competencias de los trabajadores para analizar las competencias que existen en la empresa y los departamentos.

Criterios de aceptación:

El usuario puede seleccionar un centro y un departamento.

El sistema muestra las competencias que existen tanto en el centro como en los departamentos del centro seleccionados por el usuario.

El sistema muestra las competencias desglosadas por nivel para el centro y departamento seleccionado por el usuario.

Tabla A.13: US-007 Ver indicadores de competencias

A.2.3. Requisitos de información

IRQ-001	Empresas
Dependencias	-
Descripción	El sistema deberá almacenar la información correspondiente a <i>las empresas</i> .
Datos específicos	<ul style="list-style-type: none"> • <i>Nombre de la empresa</i> • <i>Secciones</i> • <i>Departamentos</i>
Importancia	<i>Vital</i>
Prioridad	<i>Alta</i>
Estado	<i>Validado</i>
Comentarios	-

Tabla A.14: IRQ-001 Empresas

IRQ-002	Empleados
Dependencias	-
Descripción	El sistema deberá almacenar la información correspondiente a <i>los empleados de cada empresa</i> .
Datos específicos	<ul style="list-style-type: none"> • <i>N.I.F.</i> • <i>Nombre completo</i> • <i>Fecha de alta</i> • <i>Fecha de baja</i> • <i>Localización</i>
Importancia	<i>Vital</i>
Prioridad	<i>Alta</i>
Estado	<i>Validado</i>
Comentarios	-

Tabla A.15: IRQ-002 Empleados

IRQ-003	Currículums
Dependencias	-
Descripción	El sistema deberá almacenar la información correspondiente a <i>los currículums de cada trabajador.</i>
Datos específicos	<ul style="list-style-type: none"> • <i>Presentación</i> • <i>Formación académica</i> • <i>Experiencia laboral</i> • <i>Idiomas</i> • <i>Competencias</i>
Importancia	<i>Vital</i>
Prioridad	<i>Alta</i>
Estado	<i>Validado</i>
Comentarios	-

Tabla A.16: IRQ-003 Currículums

IRQ-004	Fichajes
Dependencias	-
Descripción	El sistema deberá almacenar la información correspondiente a <i>los fichajes diarios de los trabajadores.</i>
Datos específicos	<ul style="list-style-type: none"> • <i>Tipo de fichaje</i> • <i>Hora</i>
Importancia	<i>Vital</i>
Prioridad	<i>Alta</i>
Estado	<i>Validado</i>
Comentarios	-

Tabla A.17: IRQ-004 Fichajes

A.2.4. Requisitos no funcionales

NFR-001	Seguridad
Descripción	<i>Todas las comunicaciones (entre el sistema y el usuario y del sistema con otros sistemas tales como API de terceros u otras aplicaciones de la empresa) deberán ser protegidas (uso de protocolos de comunicación seguros como HTTPS, protección de las sesiones...). Además, los datos sensibles, tales como las contraseñas de los usuarios, deberán almacenarse cifrados.</i>
Importancia	<i>Vital</i>
Prioridad	<i>Alta</i>
Estado	<i>Validado</i>
Comentarios	-

Tabla A.18: NFR-001 Seguridad

NFR-002	Rendimiento
Descripción	<i>Los tiempos de carga de las páginas deberán ser cortos. Dentro del tiempo de carga, deberá contemplarse el tiempo en poderse utilizar cada página, es decir, poder interactuar con el contenido.</i>
Importancia	<i>Vital</i>
Prioridad	<i>Alta</i>
Estado	<i>Validado</i>
Comentarios	-

Tabla A.19: NFR-002 Rendimiento

NFR-003	Accesibilidad
Descripción	<i>Las personas con discapacidad, especialmente discapacidades de tipo visual, deben ser capaces de utilizar el sistema. Para ello, se deberán seguir estándares de accesibilidad como WCAG y aplicar buenas prácticas.</i>
Importancia	<i>Vital</i>
Prioridad	<i>Alta</i>
Estado	<i>Validado</i>
Comentarios	-

Tabla A.20: NFR-003 Accesibilidad

NFR-004	Usabilidad
Descripción	<i>El sitio web deberá ser fácil de utilizar de forma que el usuario pueda utilizar satisfactoriamente la funcionalidad ofrecida aplicando el menor esfuerzo posible.</i>
Importancia	<i>Vital</i>
Prioridad	<i>Alta</i>
Estado	<i>Validado</i>
Comentarios	-

Tabla A.21: NFR-004 Usabilidad

A.3. Matriz de rastreabilidad

Para poder identificar cada requisito funcional con cada objetivo, se ofrece la matriz de rastreabilidad contenida en la Tabla A.22.

	OBJ-001	OBJ-002	OBJ-003
US-001	✓		
US-002	✓	✓	
US-003	✓	✓	
US-004	✓	✓	
US-005	✓		✓
US-006	✓		✓
US-007	✓	✓	✓

Tabla A.22: Matriz de rastreabilidad OBJ-US

ANEXO B. DISEÑO DEL SISTEMA

Este anexo describe el diseño de la plataforma. Si bien este contenido, en especial lo relativo al sistema de información y su arquitectura, se trata en la memoria, en este anexo se analizará más en detalle cada parte del proceso.

Para ello, este anexo se divide en dos apartados principales:

- Diseño de la interfaz, prototipada en papel. Además de permitir un diseño inicial rápido, sirve de base para, junto a los requisitos, realizar el diseño *software*.
- Diseño *software*, analizando la arquitectura general del sistema y detallando el diseño de clases y el flujo de mensajes entre la vista y el controlador.

Para ello, se explica la división realizada entre la vista y el controlador y se refleja, mediante diagramas de secuencia, la forma en la que interfaz e interfaz de programación de aplicaciones (API) interactúan.

Como complemento de este anexo, se adjunta por separado un manual del programador que detalla la implementación de cada clase y sus métodos.

Debido al carácter iterativo de la metodología de desarrollo seguida, en el anexo se muestra el resultado final del proceso de desarrollo. En el proceso de desarrollo se ha repetido el siguiente proceso de diseño en cada iteración: diseño de la interfaz y diseño *software* definiendo las rutas y métodos de la API, modelado y comunicación entre vista y controlador.

B.1. Diseño de la interfaz

Para el diseño de la interfaz se han realizado prototipos en papel. Estos prototipos permiten, antes de implementar la interfaz, diseñar de manera rápida el aspecto e idear los diversos mecanismos de interacción que tendrá cada página de la plataforma. Como norma, se debe mantener el aspecto visual del resto de programas de Aplifisa, si bien se pueden introducir pequeñas mejoras siempre y cuando no sean cambios radicales que provoquen una ruptura con el aspecto visual del resto de programas.

Antes de realizar el diseño de las interfaces, se realiza un mapa de navegación básico para recoger cada sección en páginas. Este mapa se muestra en forma de lista:

- Gestión de CV: categoría para todo lo relacionado con la gestión de currículums, esta incluye las siguientes páginas:
 - Listado de trabajadores.
 - Consulta de currículums.
 - Redacción de currículums.
 - Matriz de competencias.
- Análisis de indicadores: categoría para la visualización de los indicadores clave de rendimiento. Esta categoría incluye tres páginas, una por conjunto de indicadores:
 - Retención de personal.
 - Cumplimiento horario.
 - Competencias.

A partir de este mapa navegacional, se realiza el prototipo de la interfaz inicial (ver Figura B.1). Esta interfaz es el punto inicial de la aplicación y sirve, principalmente, para acceder al resto de páginas. Actualmente se ha diseñado como una pantalla de bienvenida, si bien en el futuro puede albergar contenidos y funcionalidades como un panel de alertas u otro tipo de información que se desee destacar nada más iniciar sesión.

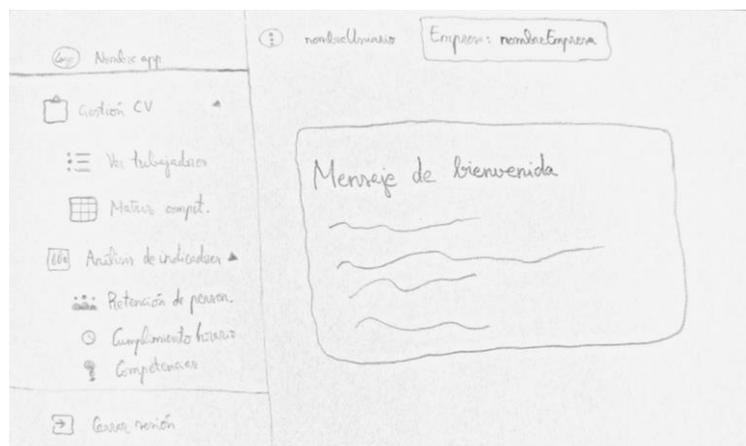


Figura B.1: Prototipo de la interfaz principal

A partir de esta interfaz, además, se pueden definir los diversos módulos de la interfaz que servirán, en la implementación, de componentes reutilizables (principalmente evitan la duplicación de código). Se han definido cuatro módulos:

- Base: contiene el cuerpo HTML básico de la plataforma. Permite definir la cabecera de la misma e incluir archivos como bibliotecas de Javascript u hojas de estilo. Sobre esta base se añade el resto de componentes.
- Barra de navegación lateral: permite al usuario acceder a las diversas páginas. El enlace a la página en la que está el usuario se recalcará de manera especial para indicar al usuario en qué apartado está.
- Barra superior: identificación del usuario y la empresa a la que pertenece.
- Contenido de la página.

Definidos estos módulos, en el resto de prototipos se mostrará solo la parte relativa a la interfaz de la página, no repitiendo la interfaz básica.

La siguiente interfaz es el listado de trabajadores (ver Figura B.2). Este listado es la primera parte de los procesos de consulta y redacción del currículum de los empleados. En el listado se muestran todos los empleados de la empresa junto a sus datos esenciales (estado, centro, departamento, fecha de alta y de baja e identificación) y una columna de acciones que permite acceder a la consulta o redacción del currículum de cada trabajador. Para facilitar la labor de búsqueda al usuario, se han añadido filtros en cada columna y varios filtros predefinidos para mostrar trabajadores en función de su estado (solo activos, sin dar de alta y dados de baja).

Hand-drawn prototype of the employee list interface. The interface includes a title "Listado de empleados" and three filter buttons: "Solo activos", "Solo no dados de alta", and "Solo dados de baja". Below the filters is a table with the following structure:

Estado	Centro Buscan cent	Departamento Buscan depart	N.I.F. Buscan NIF	Nombre Buscan nombre	Fec alta Buscan alta	Fec baja Buscan baja	Acciones
☺	C1	D1	~~~~~	~~~~~	~/~/~	~/~/~	☰ / ✎
☹	C1	D2	~~~~~	~~~~~	~/~/~	~/~/~	☰ / ✎
⚠	C2	D3	~~~~~	~~~~~	~/~/~	~/~/~	☰ / ✎

Figura B.2: Prototipo de la interfaz del listado de trabajadores

El prototipo de la interfaz para la redacción del currículum se muestra en la Figura B.3.

The image shows a hand-drawn prototype of a CV writing interface. It consists of several sections:

- Currículum Vitae** (Main Form):
 - Name: First name, Last name, Middle name.
 - Address: Province, Municipality.
 - Presentations: A large text area for writing.
- Educación** (Education):
 - Field: Title, Institution.
 - Level: Bachelor's, Master's.
 - Description: A text area.
 - Buttons: ELIMINAR, + Añadir.
- Experiencia laboral** (Professional Experience):
 - Field: Company, Position.
 - Level: Bachelor's, Master's.
 - Description: A text area.
 - Buttons: ELIMINAR, + Añadir.
- Idiomas** (Languages):
 - Field: Language.
 - Level: Bachelor's, Master's.
 - Description: A text area.
 - Buttons: ELIMINAR, + Añadir.
- Competencias** (Skills):
 - Field: Competency.
 - Level: Bachelor's, Master's.
 - Description: A text area.
 - Buttons: ELIMINAR, + Añadir.
- Navigation**:
 - Buttons: [≡] GUARDAR, ← VOLVER.

Figura B.3: Prototipo de la interfaz de redacción de currículums

En esta interfaz:

- La colocación de los elementos se hace según importancia: el apartado de datos básicos es lo primero que se ve y está más destacado que el resto, los datos relativos a formación educativa y experiencia profesional se muestran en segundo lugar y, por último, las competencias e idiomas.
- Dentro de cada apartado, a excepción del de datos básicos, se deberá poder añadir (botón “añadir” externo y general del apartado) y quitar entradas (botón “eliminar” por cada entrada), cada entrada aparecerá en una caja. Esto permitirá adaptarse a cada currículum y a los cambios en el currículum (nueva formación, aumento o bajada de nivel de una competencia...), así como poder corregir errores (cajas añadidas por error).

- Cada entrada dispone de los campos básicos de su apartado y, como complemento, se añade un campo de comentarios/observaciones para poder ampliar información.

Para terminar el proceso se incluyen dos botones: uno para guardar y otro para cancelar el proceso y volver al listado de trabajadores. Estos botones deberán estar siempre visibles.

La interfaz de redacción de currículums sirve de base para la de consulta. En la interfaz de consulta de currículums se eliminarán los botones para añadir o quitar entradas, el de guardar currículum y se sustituirán los campos que permiten su modificación por campos meramente textuales; de esta forma, se reduce el tamaño de la interfaz y se facilita la lectura (motivo principal para separar la interfaz de redacción de la de consulta).

Por último en la categoría de gestión de currículums, está la matriz de competencias. Esta matriz representa el nivel de las competencias (filas) de los empleados (columnas) de un centro y departamento, que se seleccionan utilizando los desplegados.

Para mejorar la usabilidad, se han ideado los siguientes mecanismos:

- Filtrado de trabajadores y competencias, permite al usuario visualizar solo los trabajadores y competencias que le interesen.
- Exportación a fichero CSV para conservarla en el equipo del usuario, así como poder hacer usos alternativos de la información.
- Reordenación de trabajadores (arrastrando la columna a la posición deseada) y ordenación tanto en función del nivel de las competencias de un trabajador como del nombre de la competencia (clic en el encabezado de la columna del trabajador o en la del nombre de las competencias).
- La columna de competencias será siempre visible.

Además, para mejorar la accesibilidad:

- Se permite elegir la forma de visualización del nivel de las competencias: o bien mediante un código de color (círculo rojo si es nivel básico, ámbar si es intermedio o rojo si es avanzado) o bien mediante un código numérico. De esta forma se da libertad de elección al usuario en función de sus preferencias o de sus necesidades (en especial, si derivan de problemas de visualización debidos a, por ejemplo, el daltonismo).
- Debajo de la matriz de competencias, se incluye una leyenda indicando qué significa cada código.

El prototipo de esta interfaz se puede consultar en la Figura B.4.

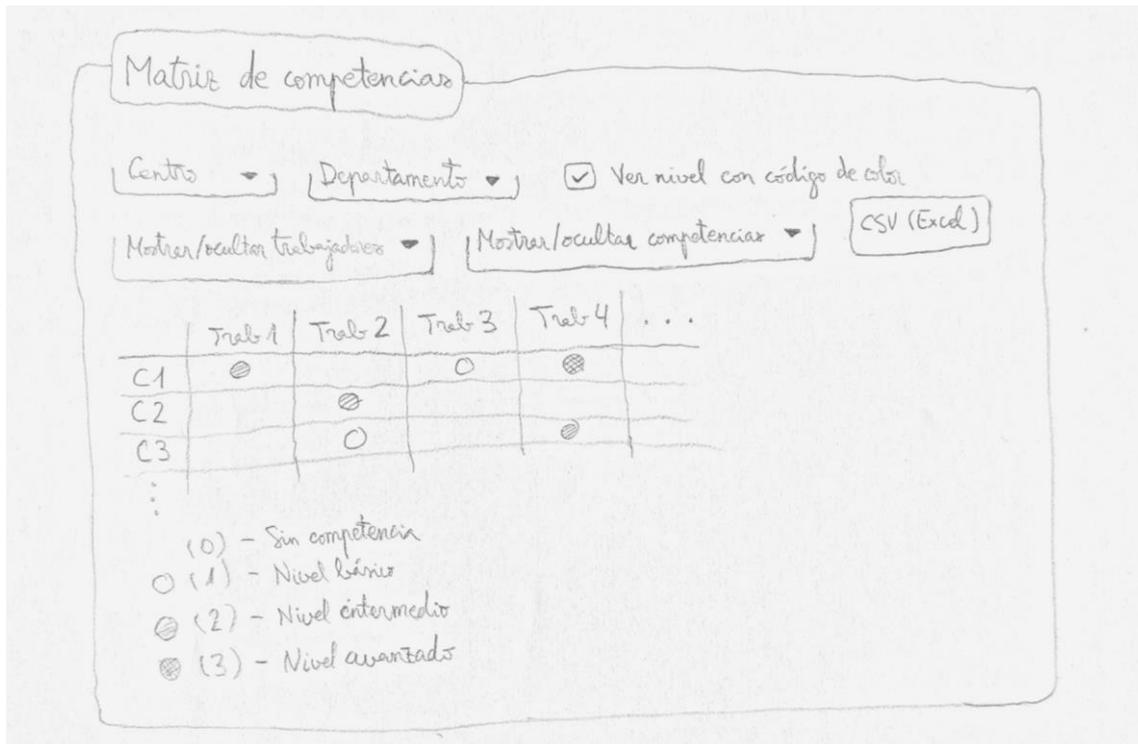


Figura B.4: Prototipo de la interfaz de la matriz de competencias

En la segunda categoría, el análisis de indicadores, cada página muestra varios indicadores asociados. Debido a que es un apartado que puede crecer (inclusión de nuevos indicadores) y contiene gráficas, que son elementos que pueden alcanzar un gran tamaño en la página, a todas estas páginas se les ha añadido un mecanismo de navegabilidad para ir de manera rápida a cada indicador y para volver arriba en la página. Este mecanismo facilita el acceso a los indicadores y permite al usuario consultar de manera rápida el indicador que le interese. Además del mecanismo de navegabilidad, en todas estas páginas se añade un botón para generar un informe imprimible o exportable a formato PDF.

Además, todas las gráficas seguirán el mismo esquema de colores, siguiendo una paleta de color violeta similar a la que se empleará en la interfaz, prestando atención a que todos los colores de las gráficas sean distinguibles. Las gráficas, para todo punto de datos, mostrarán el valor del dato en una descripción emergente. Por último, siempre que se muestren varios conjuntos de datos, se añadirá una leyenda indicando qué representa cada color.

De esta categoría, el primer prototipo es el correspondiente a la interfaz de retención de personal (ver Figura B.5).

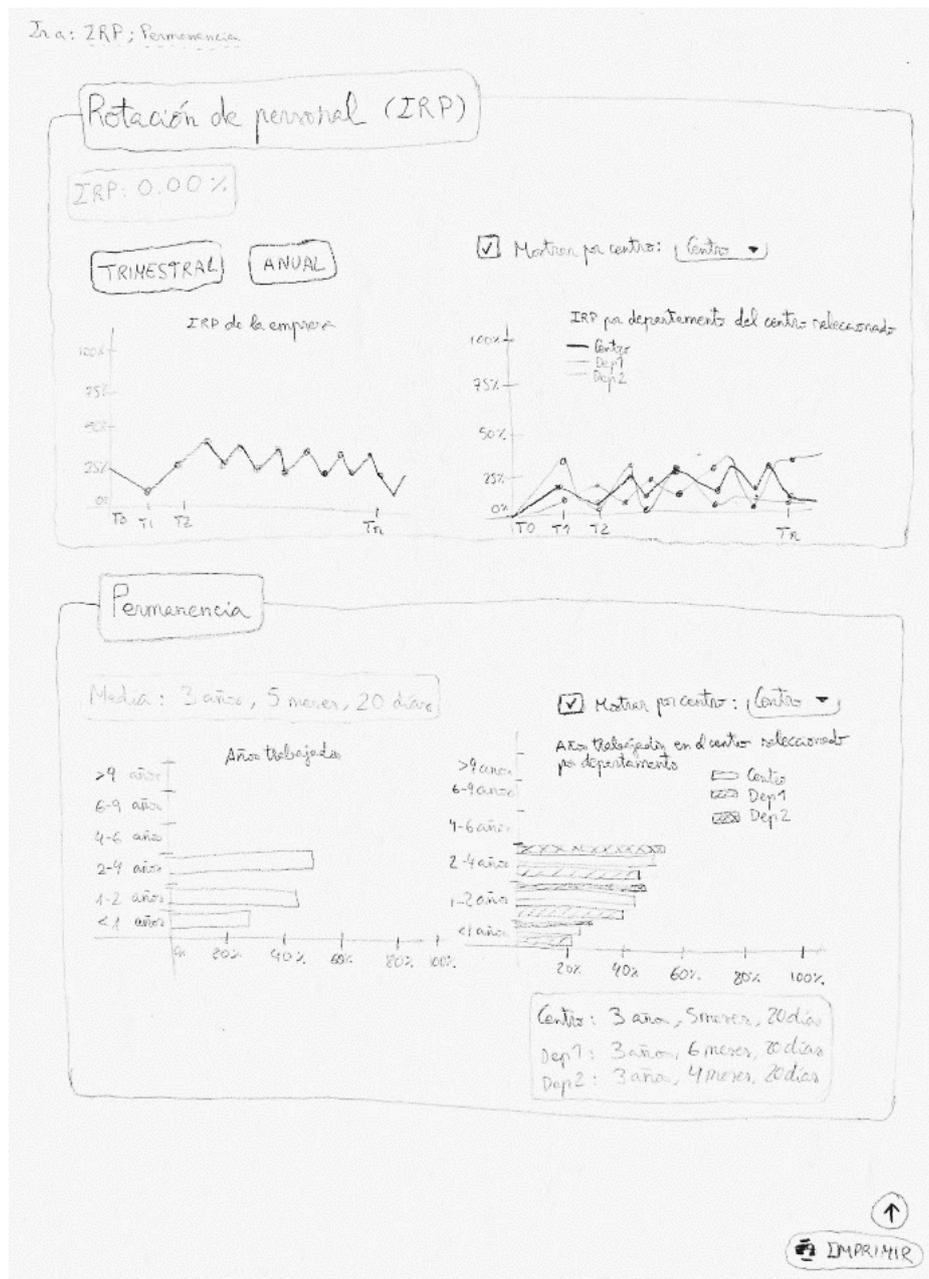


Figura B.5: Prototipo de la interfaz de los indicadores de retención de personal

En la interfaz se visualizan los indicadores de rotación de personal y la permanencia tanto para toda la empresa como por departamento de un centro seleccionado; para facilitar la comparación del dato general con los específicos, las dos visualizaciones se muestran alineadas en la misma fila. En el caso de la rotación de personal, dado que es un valor que se consulta por años o por trimestres, se han añadido dos botones para elegir si se quiere ver de manera anual o de manera trimestral. Además, en ambos indicadores se muestra el valor del indicador en el momento de consulta para poderlo tomar como referencia.

Respecto a los tipos de gráficas empleados en estos indicadores:

- Para la rotación de personal, al mostrar la evolución del dato, se utiliza una gráfica de líneas.
- Para facilitar la interpretación de los datos, se ha optado por una gráfica de líneas emulando una pirámide de población para el indicador de permanencia (en las gráficas se muestra el porcentaje de empleados en cada franja de años de permanencia, equiparable a los grupos etarios de la pirámide).

En la página de cumplimiento horario se muestra la tasa de absentismo por centros. De manera opcional, el usuario puede seleccionar un centro o bien un centro y departamento; de la primera forma se muestra el absentismo en cada departamento, mientras que, de la segunda, se muestra el absentismo tanto por la tasa como por el factor Bradford para cada empleado del centro y departamento.

Las tasas de absentismo sirven para analizar la evolución del absentismo y , por lo que se utilizan gráficas de líneas. En cambio, el factor Bradford sirve para ver el estado del factor de cada empleado y poder comparar y detectar los valores más altos, por lo que se utiliza una gráfica de barras.

El prototipo de la interfaz de esta página se muestra en la Figura B.6.

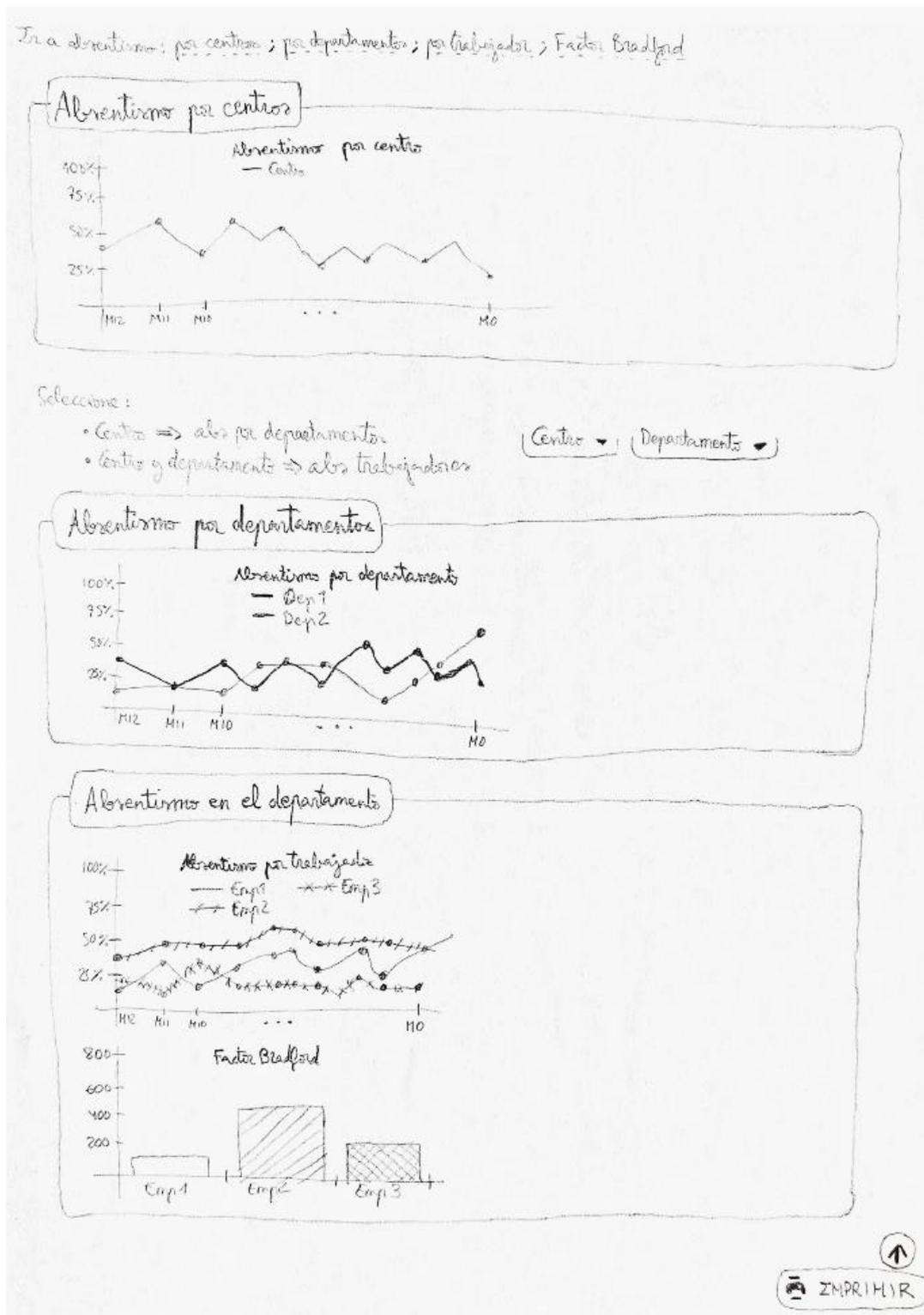


Figura B.6: Prototipo de la interfaz de indicadores de cumplimiento horario

El último prototipo realizado es el de la página de indicadores de competencias (ver Figura B.7). Todos los indicadores de competencias sirven para evaluar la cantidad de empleados que tiene cada competencia y cuántos hay por cada nivel de competencia, por lo que se representan con gráficas de barras. Como el objetivo es analizar la situación yendo desde una perspectiva global (centro)

hacia una más concreta (centro y departamento y, además, desglose por nivel), al seleccionar el centro y el departamento, se mostrarán todos los indicadores (a diferencia de la página de indicadores de cumplimiento horario, donde se podía seleccionar solo el centro).

Mientras que para las competencias sin desglosar cada competencia tiene su color, cuando aparecen desglosadas es cada nivel el que tendrá su propio color, dado en función del nivel (niveles más bajos tendrán menor tono de color). Esta forma de representar los niveles pretende asemejarse a los cimientos de un edificio, donde si son más sólidos (tonos más fuertes) indicará que son mejores cimientos (buen nivel general de la competencia).

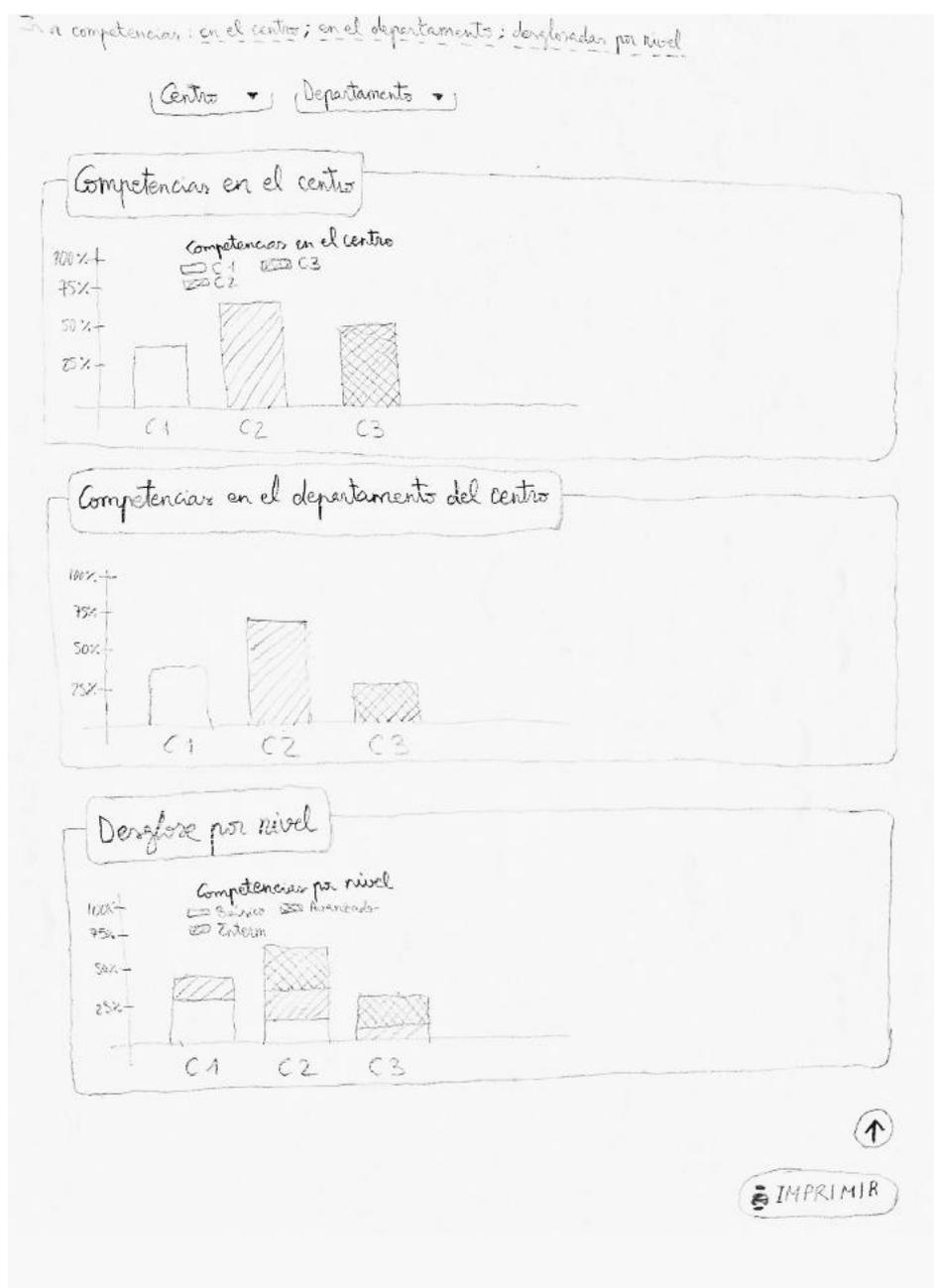


Figura B.7: Prototipo de la interfaz de indicadores de competencias

B.2. Diseño software

B.2.1. Definición de rutas y API

Todas las rutas de la web se definen en el fichero “routes/web.php”, donde se enlaza cada ruta con el método que gestiona la petición. Este método puede ser una clausura (*closure*, actúa implícitamente como un método de un controlador) o bien un método implementado en un controlador.

```
Route::get('/cv/trabajadores/redactar/{id}', [ViewControllerEmployeeCV::class, "writeCV"]->name("writeCV"));
Route::get("/cv/matriz-competencias", function () {
    return view("/cv/skills_matrix");
})->name("skillsMatrix");
```

Figura B.8: Definición de rutas con clausura y con método de controlador

El fichero de rutas se ha dividido en dos secciones: una sección de rutas para devolver las vistas y otra para la propia API que permite acceder a la funcionalidad ofrecida por los controladores. A esta API las vistas requerirán la información que requieran en función de la interacción del usuario. Todas las rutas están protegidas y requieren autenticación para poder acceder a ellas.

Las rutas de vistas, mostradas en la Tabla B.1, utilizan una clausura para su definición, si bien se ha creado un controlador específico (ViewControllerEmployeeCV) para los métodos de la API relacionados con el currículum para poder validar, antes de devolver la vista, el N.I.F. del empleado del que se desea acceder a su currículum. Las rutas que se definen mediante clausura se indica el nombre de la vista asociada entre paréntesis.

Método HTTP	URI	Controlador y método
GET	/	<i>Closure</i>
GET	/dashboard	<i>Closure</i> (dashboard)
GET	/en-desarrollo	<i>Closure</i> (placeholder)
GET	/cv/trabajadores	<i>Closure</i> (cv/employees)
GET	/cv/trabajadores/leer/{id}	ViewControllerEmployeeCV@readCV
GET	/cv/trabajadores/redactar/{id}	ViewControllerEmployeeCV@writeCV
GET	/cv/matriz-competencias	<i>Closure</i> (cv/skills_matrix)
GET	/indicadores/retencion	<i>Closure</i> (kpi/retention_kpis)
GET	/indicadores/cumplimiento-horario	<i>Closure</i> (kpi/schedule_compliance)
GET	/indicadores/competencias	<i>Closure</i> (kpi/skills_kpis)

Tabla B.1: Rutas de acceso a vistas

La ruta raíz (“/”) redirige al panel de control (ruta “/dashboard”) si el usuario tiene una sesión iniciada y a la página de inicio de sesión en caso contrario.

Todos los métodos de la API, con sus respectivas rutas y el método del controlador asociado, se muestran en la Tabla B.2.

Método HTTP	URI	Controlador y método
GET	/business/getOrganization	BusinessController@getOrganization
GET	/CV/getCV/{id}	CVController@getCV
POST	/CV/postCV/{id}	CVController@postCV
POST	/CV/validateCV/{id}	CVController@validateCV
GET	/employees/loadList	EmployeesController@loadList
GET	/skills/getMatrixColumns/{sec}/{dept}	SkillsController@getMatrixColumns
GET	/skills/getMatrixData/{sec}/{dept}	SkillsController@getMatrixData
GET	/KPI/absenteeismPerCenter	KPIController@absenteeismPerCenter
GET	/KPI/absenteeismPerDepartment/{sec}	KPIController@absenteeismPerDepartment
GET	/KPI/absenteeismPerDepartmentEmployees/{sec}/{dept}	KPIController@absenteeismPerDepartmentEmployees
GET	/KPI/averageTenure	KPIController@averageTenure
GET	/KPI/bradfordFactor/{sec}/{dept}	KPIController@bradfordFactor
GET	/KPI/detailedAverageTenure/{sec}	KPIController@detailedAverageTenure
GET	/KPI/detailedTurnover/{sec}	KPIController@detailedTurnover
GET	/KPI/turnover	KPIController@turnover
GET	/skills/getSectionDepartmentLevels/{sec}/{dept}	SkillsController@getSectionDepartmentLevels
GET	/skills/getSectionDepartmentSkills/{sec}/{dept}	SkillsController@getSectionDepartmentSkills
GET	/skills/getSectionSkills/{sec}	SkillsController@getSectionSkills

Tabla B.2: Rutas de la API

Tanto la vista como el controlador son completamente independientes y cada controlador tiene métodos asociados en cuanto a funcionalidad. Esta independencia y la agrupación de métodos en los distintos controladores facilitarían, si se considerara necesario, trasladar diversas partes a otros nodos de despliegue dentro del marco de una arquitectura web distribuida como podría ser una arquitectura de microservicios.

B.2.2. Modelo de clases

El patrón arquitectónico empleado es el modelo-vista-controlador, donde:

1. El servidor, cuando el usuario accede a las rutas de acceso a vistas, devuelve al usuario la vista correspondiente.
2. Las vistas reaccionan a la interacción del usuario y solicitan datos a los controladores.
3. Los controladores acceden al modelo y devuelven a la vista la información solicitada.
4. La vista recibe los datos y se actualiza acordeamente.

Para el modelo y en base a los requisitos funcionales, el modelo físico de la base de datos se representa por medio del diagrama entidad-relación mostrado en la Figura B.9.

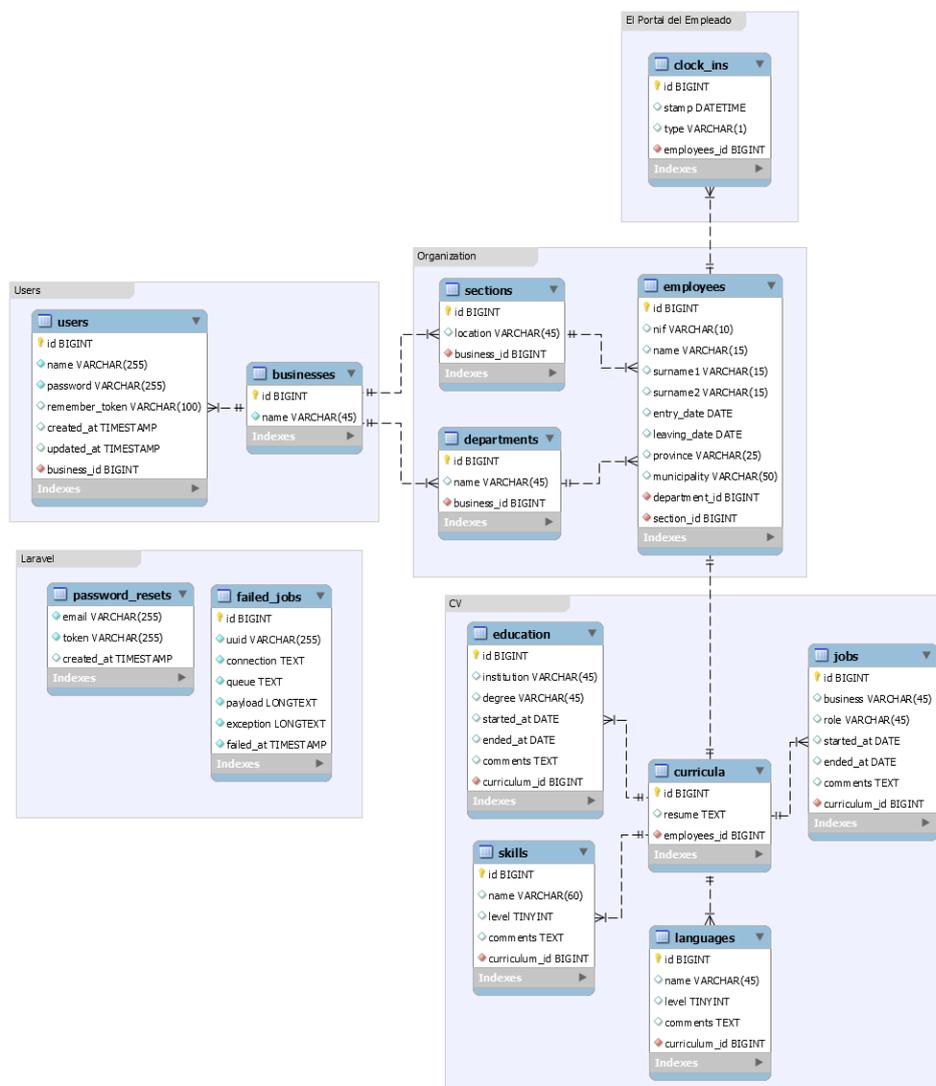


Figura B.9: Modelo físico de la base de datos

Laravel, para poder acceder a los datos, utiliza el patrón *active record* a través de Eloquent. La forma que tiene Eloquent de interpretar qué es una tabla de la base de datos es a través del nombre. Las condiciones para que se produzca una interpretación correcta son las siguientes:

- A cada tabla se accede con una clase que hereda de Model.
- Para saber de qué tabla tomar los datos y qué atributos tiene cada tabla:
 - o El nombre de la tabla en base de datos debe ir en plural y minúsculas utilizando letra de caja de serpiente (*snake case*).
 - o El nombre de la clase debe ser el nombre de la tabla en singular y utilizando letra de caja de camello (*camel case*) empezando con mayúscula.

Estas clases del modelo en la práctica funcionan tanto de modelo (una entidad con sus atributos entre los que se incluyen también los pertenecientes a tablas asociadas, se accede a ellos a través de propiedades) como de objeto de persistencia (proveen de métodos para el alta, consulta, modificación y borrado de tuplas en la base de datos). Además, para acceder a datos de tablas asociadas, Eloquent permite definir métodos que declaren la relación entre las tablas y a partir de ellos se generan propiedades que permiten acceder a estos datos.

Además de las clases propias del modelo junto al patrón *active record* de Eloquent, se ha utilizado el patrón repositorio para abstraer el acceso a los datos de los objetos del dominio. En concreto, se ha creado un repositorio para el acceso a las empresas y sus empleados (BusinessRepository) y otro para el acceso a los datos de los empleados (EmployeeRepository).

Para el modelado del controlador se utilizan como base los métodos y controladores definidos en las rutas de la API, añadiendo los métodos que se consideren necesarios.

En el caso de la vista no es necesario realizar un modelado de clases: es HTML a cuyos elementos se les añade el comportamiento requerido utilizando código JavaScript. Sin embargo, para facilitar la reutilización de elementos visuales como se señaló en el apartado B.1, se han creado tres clases que heredan de Component con el fin de renderizar la vista base (AppLayout), la barra superior (Navbar) y la barra lateral (Sidebar). Además, de cara también a facilitar la implementación, se crea una clase (CVRequest) que hereda de FormRequest y se encarga de recoger en la clase los datos contenidos en el formulario de redacción de currículum.

Con estas consideraciones, se realiza el diagrama de clases. Para respetar la estructura de paquetes de Laravel (bastante similar a la división en modelo, vista y controlador) y facilitar su comprensión, el diagrama se ha dividido en los siguientes diagramas: controladores (ver Figura B.10), modelos (ver Figura B.11), repositorios (ver Figura B.12), componentes visuales (ver Figura B.13) y peticiones (ver Figura B.14).

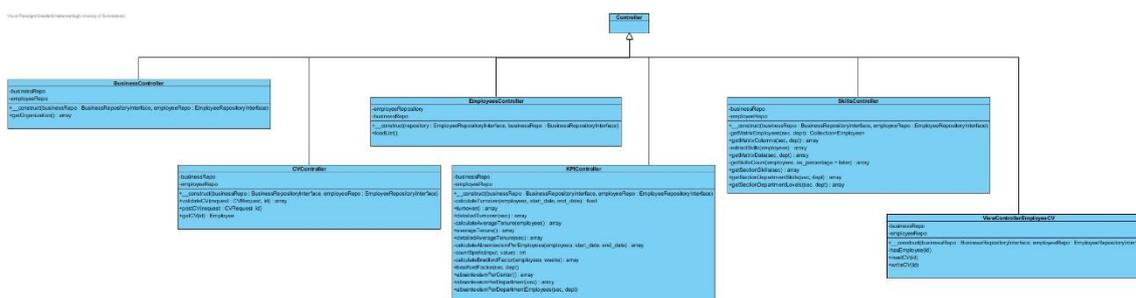


Figura B.10: Diagrama de clases – Controladores

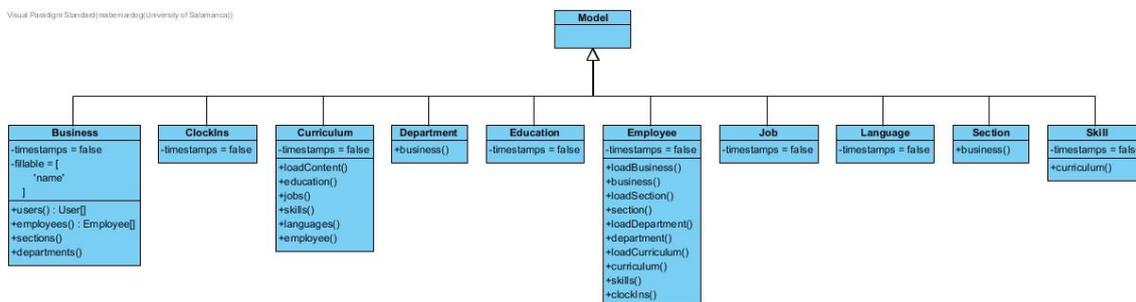


Figura B.11: Diagrama de clases – Modelos

En el diagrama de clases del modelo, no se ha reflejado relación alguna entre cada clase. Se ha optado por esta opción debido a que Eloquent se encarga internamente de generar las relaciones entre las clases; estas relaciones se reflejan en implementación en los métodos que, dentro de las entidades, llevan el nombre de la tabla asociada. Para consultar las relaciones entre las entidades se debe consultar la Figura B.9.

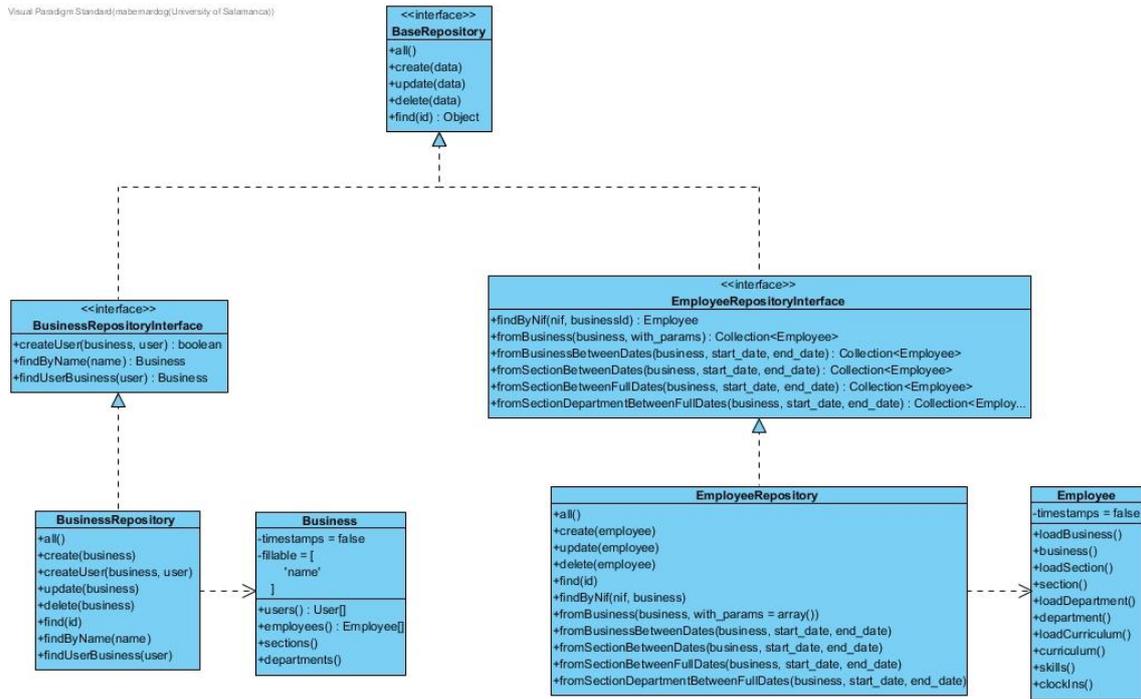


Figura B.12: Diagrama de clases – Repositorios

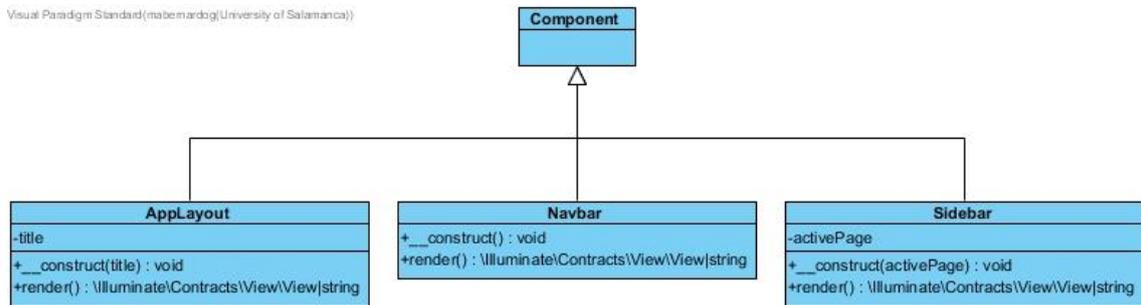


Figura B.13: Diagrama de clases – Componentes

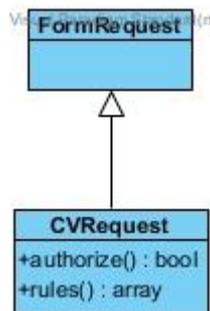


Figura B.14: Diagrama de clases – Peticiones

B.2.3. Comunicación entre vista y controlador

Por último, para cada historia de usuario se ha realizado un diagrama de secuencia (ver desde la Figura B.15 hasta la Figura B.21). Estos diagramas reflejan la interacción del usuario con las vistas y el flujo de llamadas entre las vistas y los controladores. Dada la naturaleza de los diagramas de secuencia, permiten además detallar la secuencia de pasos que debe realizar el usuario para alcanzar sus objetivos, aspecto que se omite al redactar las historias de usuario.

En los diagramas se ha optado por no incluir la comunicación entre el controlador y el modelo, que se hará en función de la lógica de negocio y los algoritmos implementados utilizando como referencia la documentación de las clases.

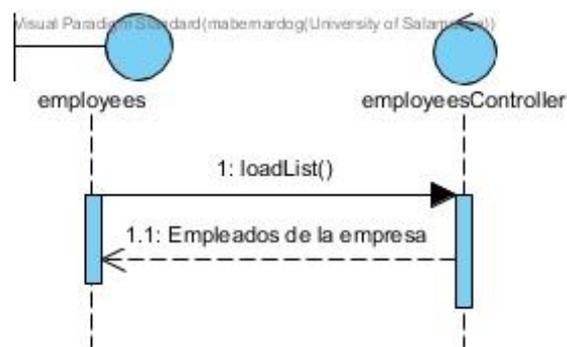


Figura B.15: US-001 Lista de trabajadores

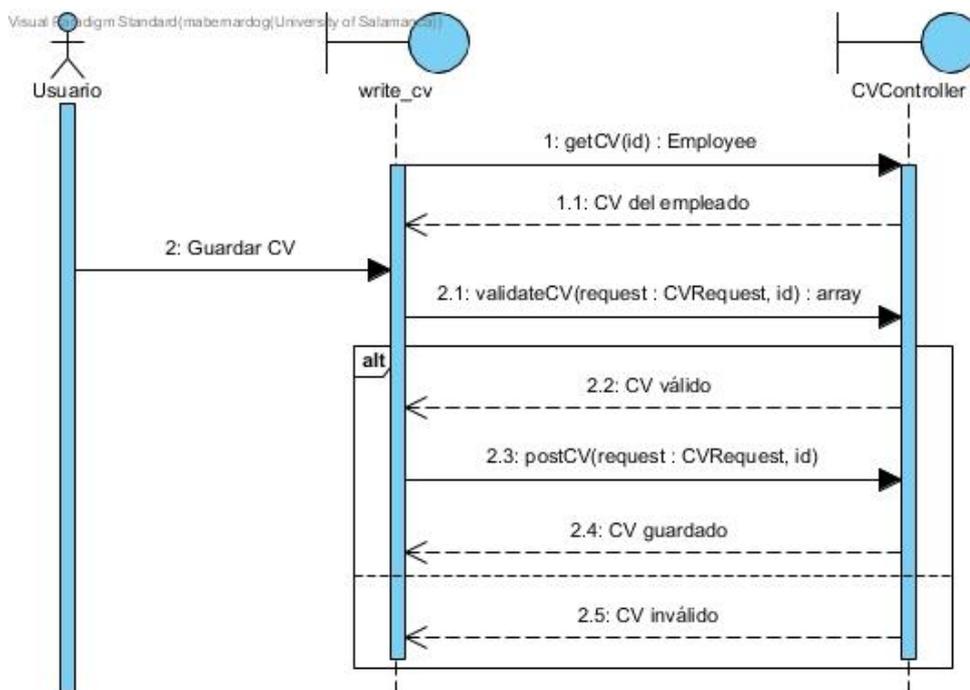


Figura B.16: US-002 Redactar CV

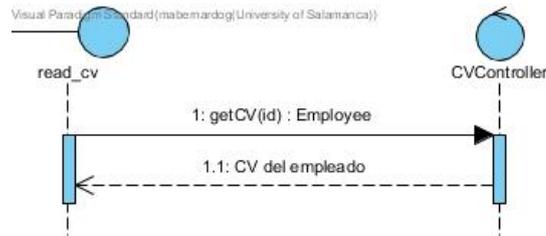


Figura B.17: US-003 Consultar CV

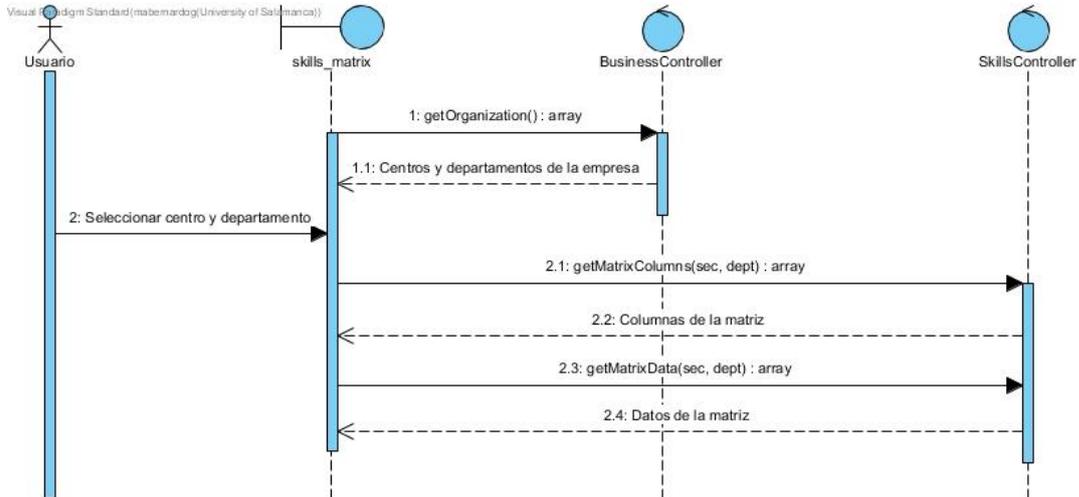


Figura B.18: US-004 Crear matriz de competencias

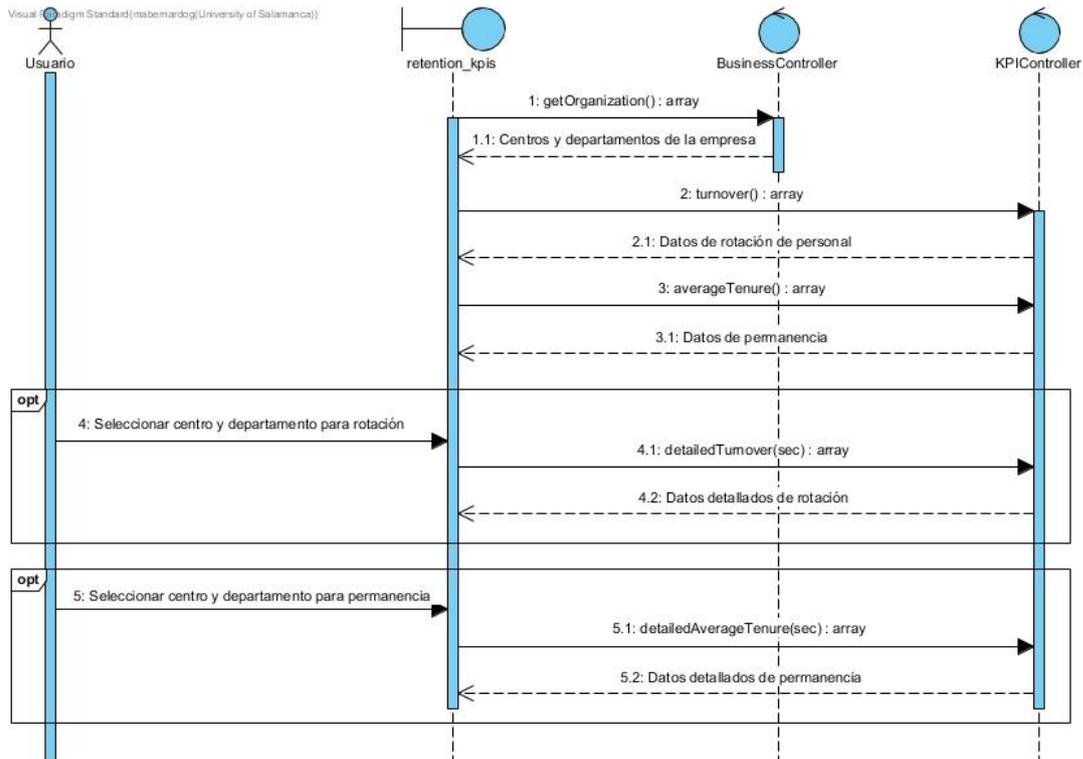


Figura B.19: US-005 Ver indicadores de retención de personal

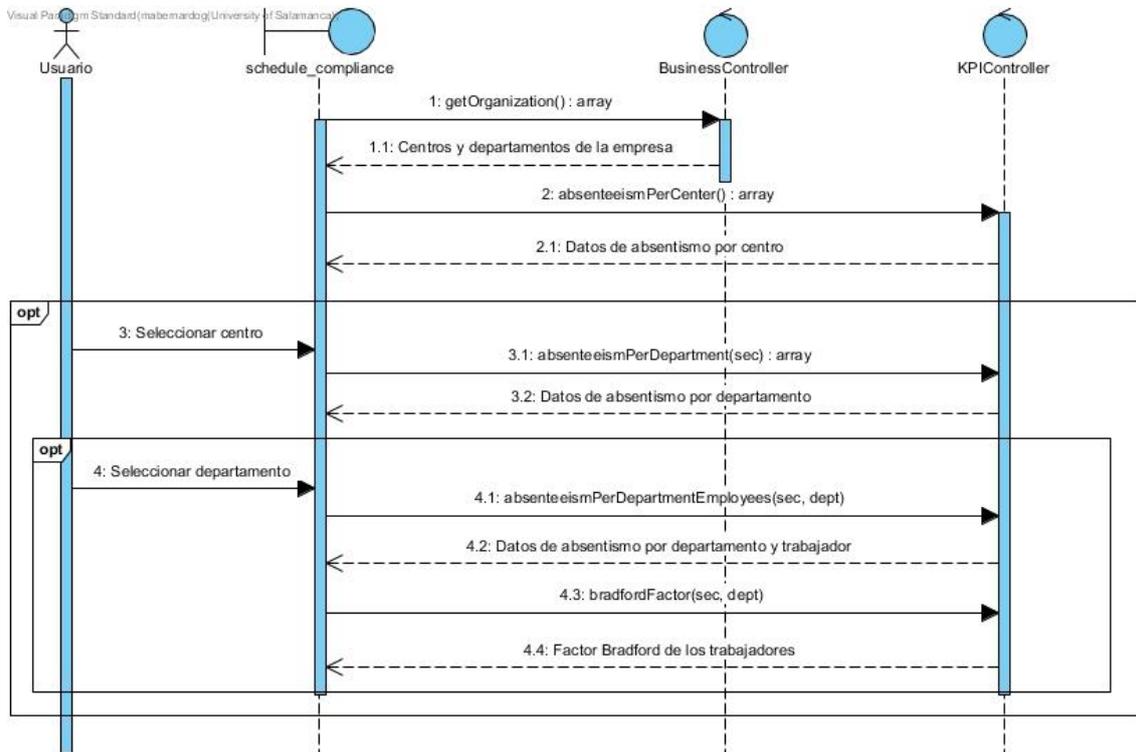


Figura B.20: US-006 Ver indicadores de cumplimiento horario

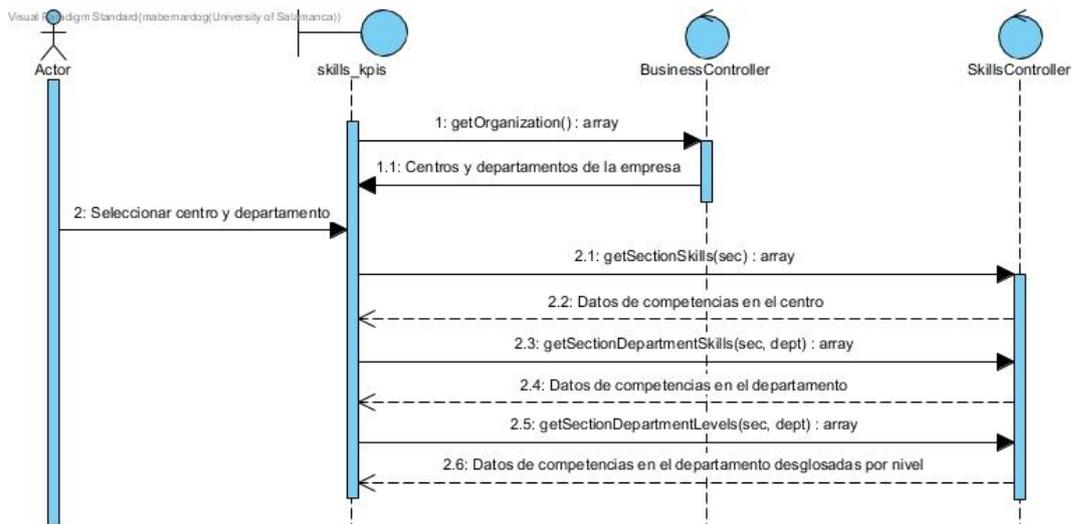


Figura B.21: US-007 Ver indicadores de competencia

ANEXO C. INFORMES DE AUDITORÍA

Como complemento a la información mostrada en la memoria, se presentan para cada página los resultados obtenidos al analizar el rendimiento y la accesibilidad con la herramienta de auditoría Lighthouse.

Para cada página se muestra tanto la puntuación dada por la herramienta para cada apartado como, además, los valores de las métricas de rendimiento y las pruebas que hay que realizar manualmente para comprobar la accesibilidad (el resto de pruebas o bien se han pasado con éxito o bien no eran aplicables).

Las pruebas se han realizado con la siguiente configuración:

- Dispositivo: ordenador de escritorio, emulado.
- Conexión:
 - o Latencia: 40ms.
 - o Velocidad: 10240 Kbps.
- Agente de usuario:
 - o Cliente: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/92.0.4515.159 Safari/537.36 Edg/92.0.902.84.
 - o Red: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4420.0 Safari/537.36 Chrome-Lighthouse
- Versión de Axe: 4.1.3.
- Versión de Lighthouse: 7.5.0.

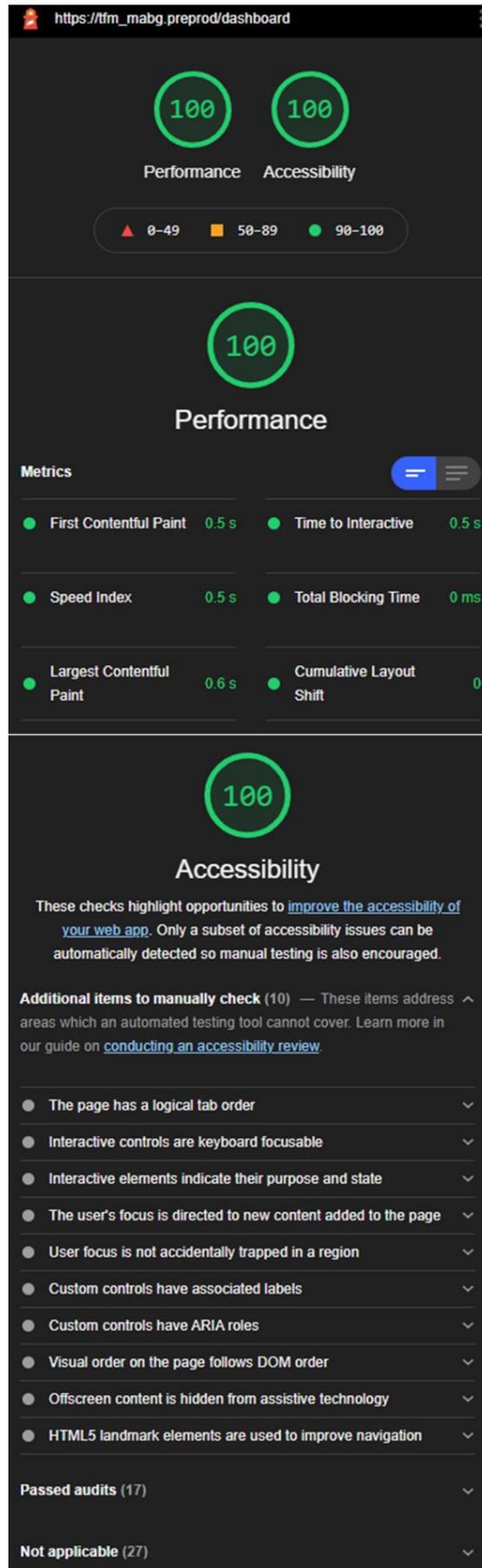


Figura C.1: Auditoría - Página principal

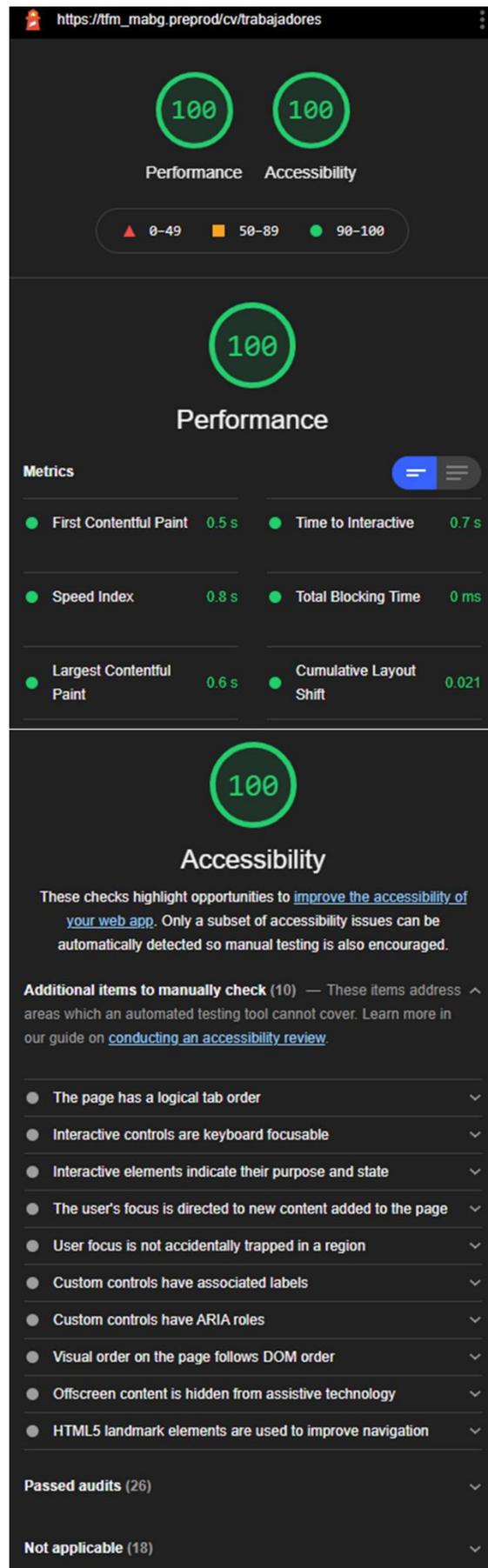


Figura C.2: Auditoría - Listado de trabajadores

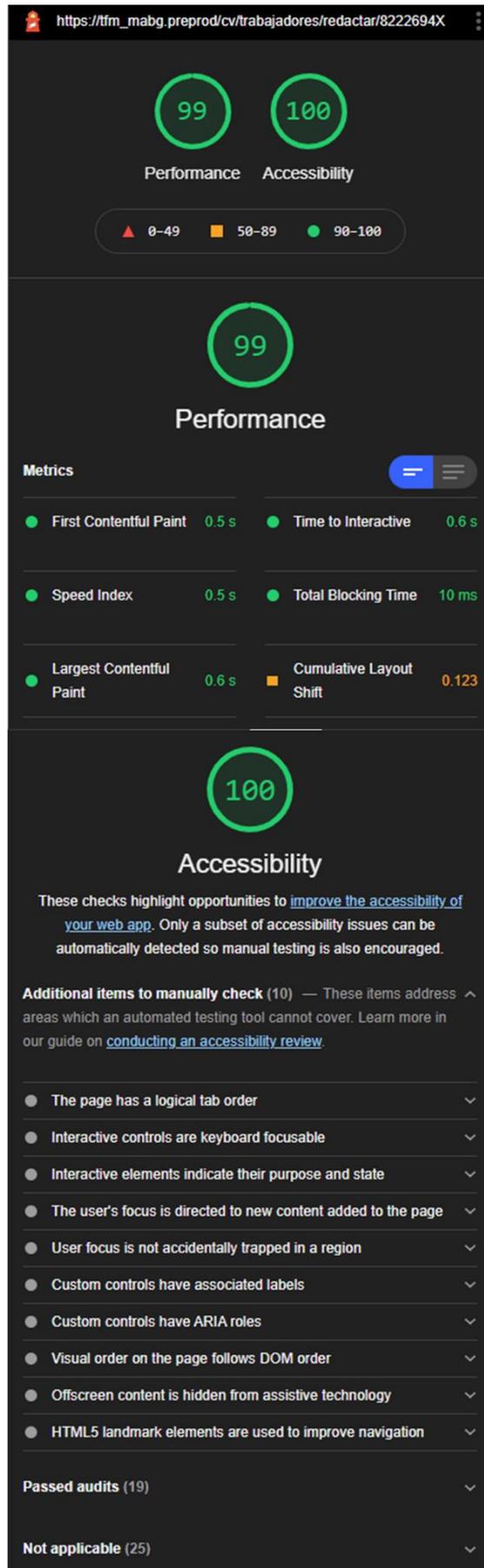


Figura C.3: Auditoría - Redacción de currículum

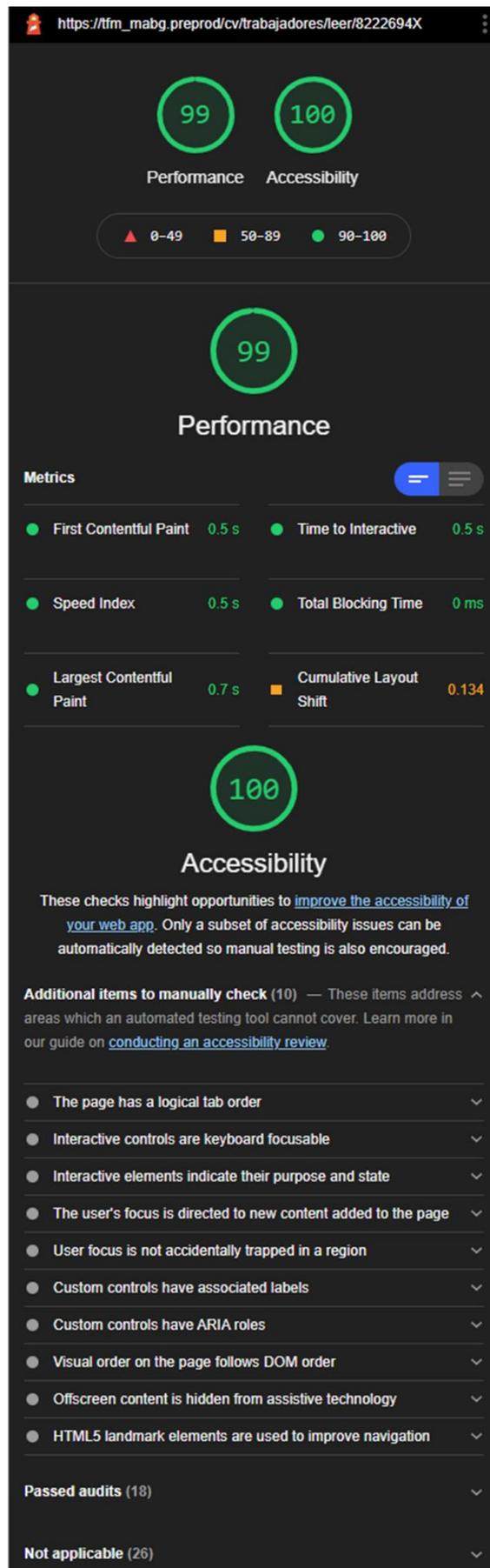


Figura C.4: Auditoría - Consulta de currículum

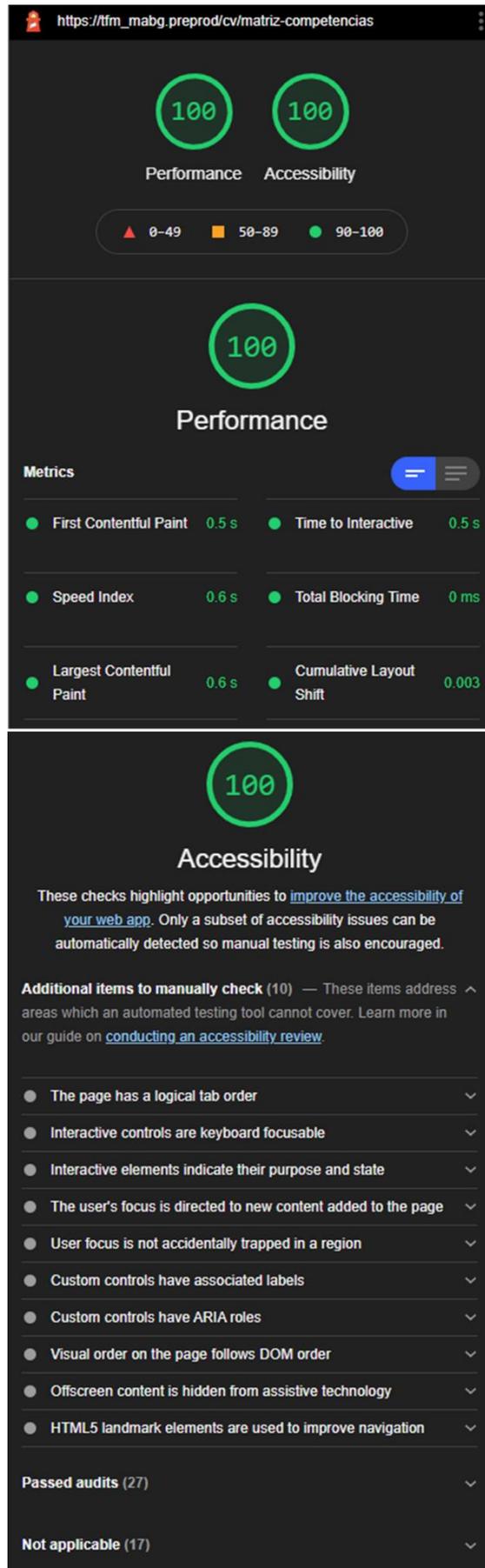


Figura C.5: Auditoría - Matriz de competencias

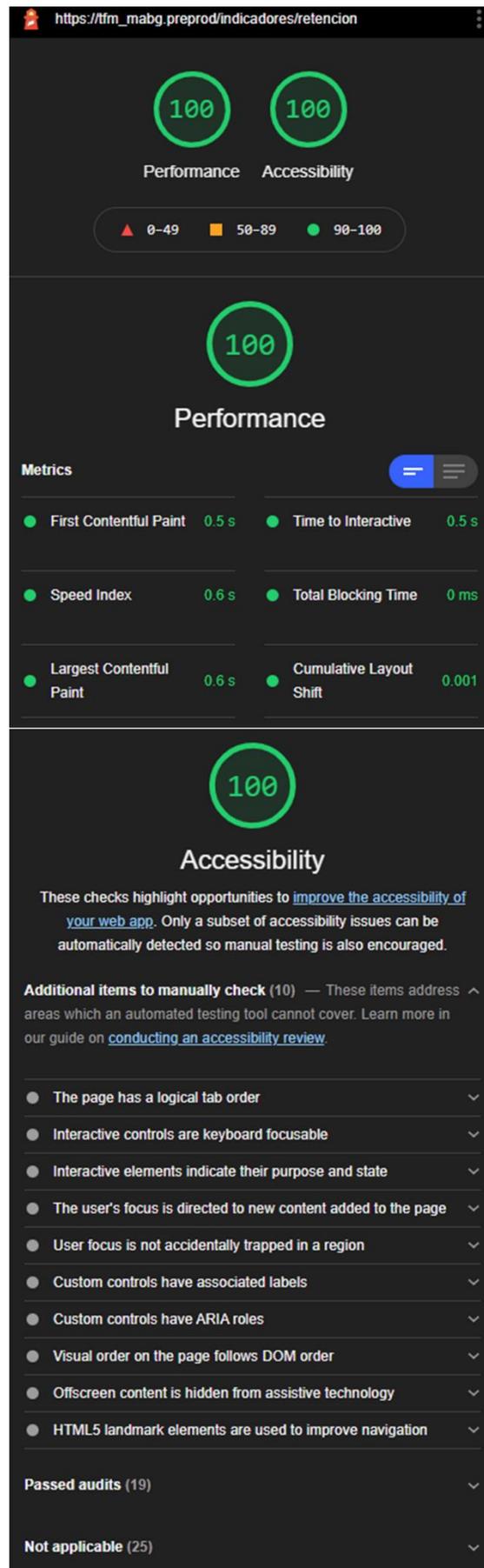


Figura C.6: Auditoría - Indicadores de retención de personal

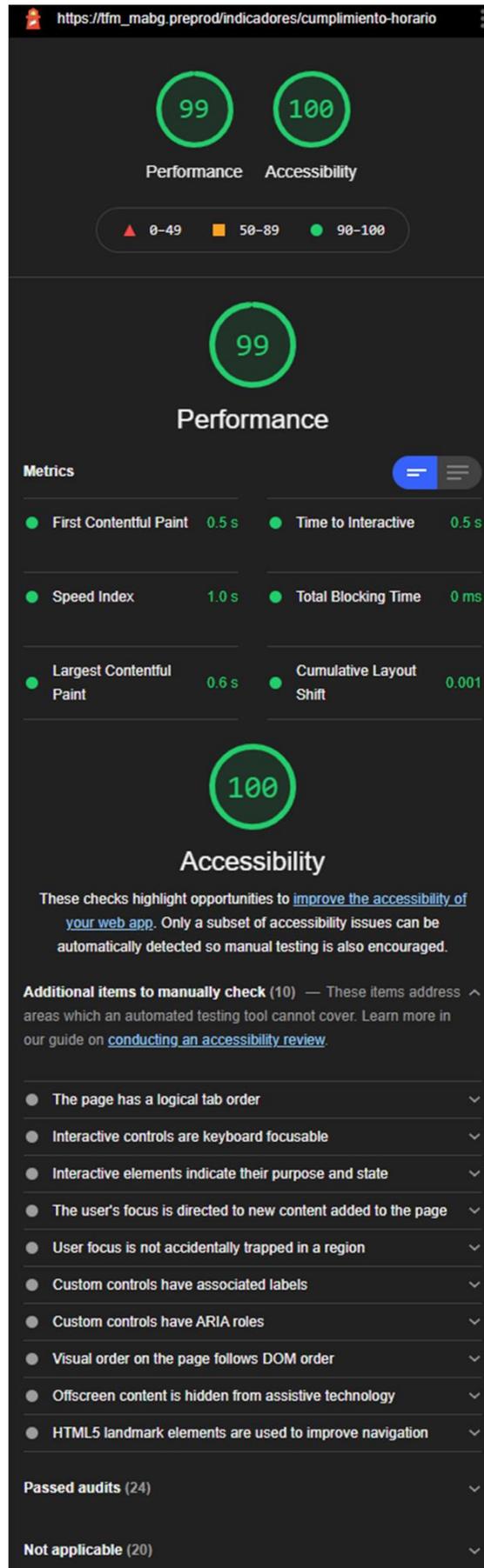


Figura C.7: Auditoría - Indicadores de cumplimiento horario

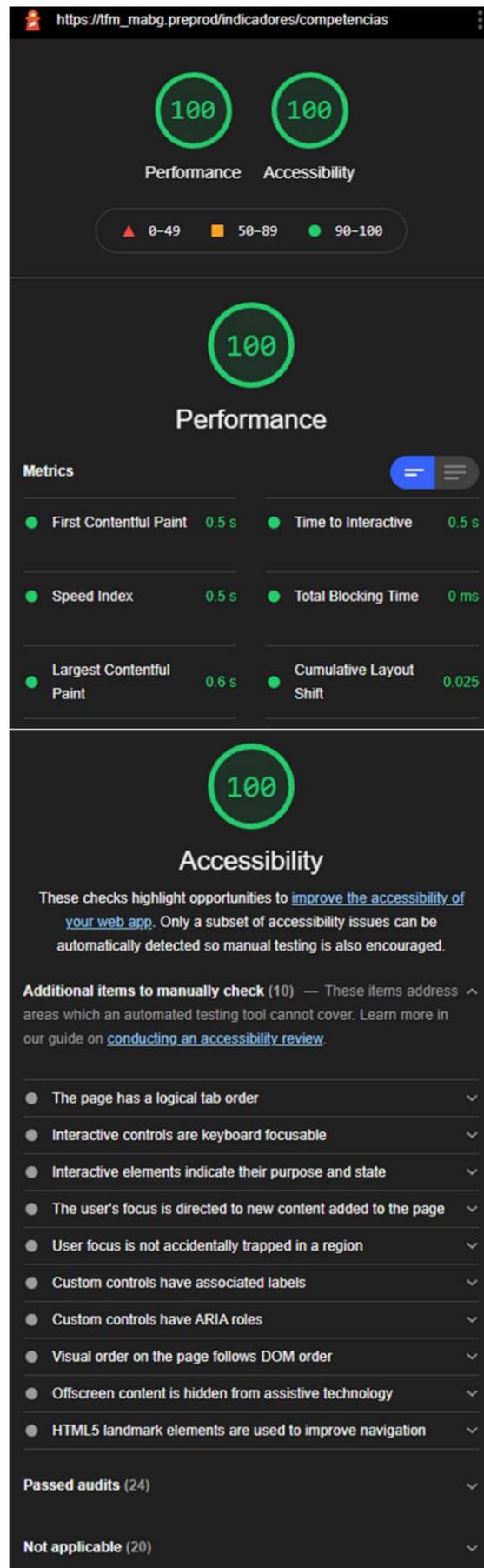


Figura C.8: Auditoría - Indicadores de competencias