



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

DSIC
DEPARTAMENT DE SISTEMES
INFORMÀTICS I COMPUTACIÓ

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Dpto. de Sistemas Informáticos y Computación

ProSafe: Mejora de la capacitación en prevención de
riesgos laborales

Trabajo Fin de Máster

Máster Universitario en Ingeniería y Tecnología de Sistemas
Software

AUTOR/A: Codoñer Gil, Nieves

Tutor/a: Penadés Gramage, María Carmen

Director/a Experimental: QUIROZ PALMA, PATRICIA ALEXANDRA

CURSO ACADÉMICO: 2021/2022

Agradecimientos

Me gustaría expresar mi más sincero agradecimiento a todas las personas que me han acompañado durante esta etapa académica.

A mi familia y pareja, que han sido un pilar fundamental; entendiéndome en todo momento, alentándome y confiando en mí para realizar el TFM.

A mi tutora del trabajo de fin de máster María Carmen Penadés Gramage y a mi directora Experimental Patricia Alexandra Quiroz Palma por guiarme durante el desarrollo del proyecto.

A mis amigos por siempre tener buenas palabras y apoyarme incluso en los momentos más difíciles, vuestra dedicación me ha ayudado a conseguir sobrellevar todo el estrés que ha supuesto el trabajo.

A los profesores del máster por compartir sus conocimientos y experiencia.

A la directora del máster Alicia Villanueva García por ofrecerme su ayuda y consejos durante esta etapa.

Resum

Un dels aspectes més importants en l'àmbit de la prevenció de riscos laborals en les organitzacions és la formació dels seus empleats per a poder anticipar-se als possibles riscos en els quals es puguen veure involucrats. Aquest procés ha anat evolucionant al llarg dels anys amb l'objectiu de dotar als empleats amb formació o eines més eficients per a millorar la seua resposta davant una possible situació d'emergència, no obstant això, no existeix una visió integrada de la formació i avaluació contínua durant la vida de l'empleat en l'organització. En nombroses ocasions, la formació es realitza al principi de la seua vida laboral en l'organització i no hi ha una continuïtat ni en la formació ni en l'avaluació d'aquesta.

El propòsit d'aquest TFM és abordar la millora de la capacitat dels empleats d'una organització en el domini concret de la prevenció de riscos laborals. S'utilitzarà el model CiET (Continuous improvement for Emergency Management Training) com a base per a abordar la definició i desenvolupament d'un marc de capacitat en riscos laborals perquè les organitzacions puguen gestionar la formació i entrenament dels seus empleats, així com l'avaluació contínua d'aquest aprenentatge. El marc proposat se suportarà per una eina que servisca tant per al reforç continu dels coneixements dels empleats, com per a l'extracció d'informació amb la intenció de millorar el procés de formació i entrenament d'aquests. Aquest reforç continu de coneixement es realitza per mitjà d'una aplicació mòbil, que mostrarà alarmes imitant possibles situacions d'emergència i sol·licita a l'usuari una resposta primerenca i correcta d'actuació i s'utilitzaran tècniques de ludificació. D'altra banda, es crearà una plataforma web destinada al responsable de riscos laborals de l'organització, amb la condició que pugua comptar amb una eina de visualització i extracció d'informació sobre la situació actual d'aptituds i capacitats apreses pels empleats, alhora que gestionarà els objectius d'aprenentatge a treballar.

Paraules clau: formació, PRL, capacitat, CiET, ludificació, aplicació web, aplicació mòbil, React, Spring

Resumen

Uno de los aspectos más importantes en el ámbito de la prevención de riesgos laborales en las organizaciones es la formación de sus empleados para poder anticiparse a los posibles riesgos en los que se puedan ver involucrados. Este proceso ha ido evolucionando a lo largo de los años con el objetivo de dotar a los empleados con formación u herramientas más eficientes para mejorar su respuesta ante una posible situación de emergencia, sin embargo, no existe una visión integrada de la formación y evaluación continua durante la vida del empleado en la organización. En numerosas ocasiones, la formación se realiza al principio de su vida laboral en la organización y no hay una continuidad ni en la formación ni en la evaluación de la misma.

El propósito de este TFM es abordar la mejora de la capacitación de los empleados de una organización en el dominio concreto de la prevención de riesgos laborales. Se utilizará el modelo CiET (Continuous improvement for Emergency Management Training) como base para abordar la definición y desarrollo de un marco de capacitación en riesgos laborales para que las organizaciones puedan gestionar la formación y entrenamiento de sus empleados, así como la evaluación continua de dicho aprendizaje. El marco propuesto se soportará por una herramienta que sirva tanto para el refuerzo continuo de los conocimientos de los empleados, como para la extracción de información con la intención de mejorar el proceso de formación y entrenamiento de los mismos. Este refuerzo continuo de conocimiento se realiza por medio de una aplicación móvil, que mostrará alarmas imitando posibles situaciones de emergencia y solicita al usuario una respuesta temprana y correcta de actuación, todo ello enriquecido con técnicas de gamificación. Por otro lado, se creará una plataforma web destinada al responsable de riesgos laborales de la organización, con tal de que pueda contar con una herramienta de visualización y extracción de información sobre la situación actual de aptitudes y capacidades aprendidas por los empleados, a la vez que gestionará los objetivos de aprendizaje a trabajar.

Palabras clave: formación, PRL, capacitación, CiET, gamificación, aplicación web, aplicación móvil, React, Spring

Abstract

One of the most important aspects in the field of occupational risk prevention in organizations is the training of their employees to be able to anticipate possible risks in which they may be involved. This process has evolved over the years with the aim of providing employees with more efficient training or tools to improve their response to a possible emergency situation, however, there is no integrated vision of continuous training and evaluation during the employee's life in the organization. In many situations, training is done at the beginning of their working life in the organization and there is no continuity in either the training or the evaluation of the training.

The purpose of this TFM is to address the improvement of the training of an organization's employees in the specific domain of occupational risk prevention. The CiET (Continuous improvement for Emergency Management Training) model will be used as a basis to address the definition and development of an occupational risk training framework for organizations to manage the education and training of their employees, as well as the continuous evaluation of such learning. The proposed framework will be supported by a tool that will serve both for the continuous reinforcement of employees' knowledge, as well as for the extraction of information with the intention of improving the training and coaching process. This continuous reinforcement of knowledge will be carried out by means of a mobile application, which will display alarms imitating possible emergency situations and will request the user an early and correct response and will use gamification techniques. On the other hand, a web platform will be created for the organization's occupational risk manager, so that he/she can have a tool to visualize and extract information on the current situation of skills and abilities learned by employees, while managing the learning objectives to be worked on.

Key words: training, PRL, training, CiET, gamification, web application, mobile application, React, Spring

Índice general

Índice general	VII
Índice de figuras	IX
Índice de tablas	XI
<hr/>	
1 Introducción	1
1.1 Motivación	1
1.2 Objetivos	2
1.3 Metodología	2
1.4 Estructura de la memoria	3
2 Estado del arte	5
2.1 Prevención de riesgos laborales	5
2.1.1 Normativa de prevención de riesgos laborales	6
2.1.2 Prevención riesgos laborales: personal docente	6
2.2 Aplicaciones PRL	7
2.2.1 Aplicaciones sobre gestión de PRL	8
2.2.2 Aplicaciones de aprendizaje	9
2.2.3 Comparación aplicaciones	13
2.3 CiET	14
2.3.1 CiET como plataforma de entrenamiento	15
3 ProSafe: Propuesta de la solución	19
3.1 Descripción general	19
3.1.1 Características diferenciadoras	19
3.1.2 Aplicación de CiET	20
3.1.3 Modelo inicial propuesto	20
3.1.4 Estadísticas	22
3.1.5 Uso del producto software	24
3.2 Especificación de requisitos	24
3.2.1 Requisitos funcionales	24
3.2.2 Requisitos no funcionales	26
3.3 Casos de uso	26
3.4 Arquitectura	40
4 Diseño de la solución	43
4.1 Diseño de API	43
4.2 Diseño de interfaces	51
4.2.1 Portal web	51
4.2.2 Aplicación móvil	59
5 Tecnologías utilizadas	63
5.1 Entorno	63
5.2 Herramientas	65
5.2.1 Herramientas de diseño	65
5.2.2 Herramientas de organización	67
5.3 Tecnologías y lenguajes de programación	69

5.3.1	Frontend	69
5.3.2	Backend	72
6	Desarrollo	75
6.1	Frontend	75
6.1.1	Preparación del entorno	75
6.1.2	Estructura general de los proyectos	76
6.1.3	Modulo internalización del proyecto	77
6.1.4	Estadísticas	77
6.1.5	Aplicación móvil	77
6.2	Backend	80
6.2.1	Integración del paquete ProSafe	80
6.2.2	Estructura de archivos	80
6.2.3	Patrón <i>message broker</i>	83
7	Pruebas	85
7.1	Pruebas de aceptación	85
7.1.1	Cambio de idioma	93
7.1.2	Visualización de información	95
7.1.3	Visualización de recursos	97
7.2	Otras validaciones	98
7.2.1	Validación de formularios	98
7.2.2	Cuadro de diálogo	100
8	Conclusiones y trabajos futuros	103
8.1	Conclusiones	103
8.2	Valoración personal	103
8.3	Trabajos futuros	104

Apéndice		
A	Casos de uso	107

Índice de figuras

1.1	Fases de la metodología en cascada	3
2.1	Pantalla principal de la pantalla web de Prevención10	8
2.2	Pantalla de la aplicación móvil de Ergonomic checkpoints	9
2.3	Pantalla de la aplicación móvil de tests	10
2.4	Pantalla de simulación de Segutractor	11
2.5	Pantalla de simulación de Segutractor	11
2.6	Pantalla principal de evaluación de trabajo repetitivo	12
2.7	Pantalla de contenidos de la aplicación Prevención de riesgos laborales	13
2.8	Diagrama de clases CiET (extraído de [7])	15
2.9	Diagrama de clases CiET plataforma (extraído de [8])	16
2.10	Pantalla de creación de planes de capacitación	17
2.11	Pantalla de creación de evaluación de contenidos	17
3.1	Clasificación: Capacidades de los interesados	20
3.2	Modelo inicial propuesto	21
3.3	Ejemplo de gráfico de columnas apiladas	23
3.4	Ejemplo de gráfico circular	23
3.5	Ejemplo de gráfico de barras	24
3.6	Diagrama de contexto de ProSafe	26
3.7	Diagrama de casos de uso para el actor learner	27
3.8	Diagrama de casos de uso para el actor trainer	28
3.9	Arquitectura ProSafe	40
4.1	Wireframe: Pantalla de inicio de sesión	51
4.2	Wireframe: Pantalla de inicio	52
4.3	Wireframe: Pantalla de gestión de planes de capacitación	53
4.4	Wireframe: Pantalla de gestión de temas	53
4.5	Wireframe: Pantalla de grupos de forma manual	54
4.6	Wireframe: Pantalla de grupos por criterios	55
4.7	Wireframe: Pantalla de documentos	55
4.8	Wireframe: Pantalla de métricas	56
4.9	Wireframe: Pantalla de información sobre ProSafe	56
4.10	Wireframe: Pantalla de información sobre datos del empleado	57
4.11	Wireframe: Diálogo de confirmación	57
4.12	Wireframe: Diálogo de información exitosa	58
4.13	Wireframe: Diálogo de información errónea	58
4.14	Wireframe: Pantalla de inicio de sesión (móvil)	59
4.15	Wireframe: Pantalla principal de usuario logueado (móvil)	60
4.16	Wireframe: Pantalla de recursos PRL (móvil)	60
4.17	Wireframe: Pantalla de información sobre la aplicación (móvil)	61
4.18	Wireframe: Pantalla de perfil de usuario (móvil)	61
5.1	IDE Webstorm	64

5.2	IDE IntelliJ IDEA	64
5.3	Organización ProSafe Github	65
5.4	Pantalla de edición de la aplicación móvil en Figma	66
5.5	Pantalla de edición de draw.io	66
5.6	Tablón de tareas para la organización	67
5.7	Tarea de diseño de interfaces desglosada en subtareas	68
5.8	Organización de la información en Notion	68
5.9	Gráfico de comparación (extraído de npsmtrends.com)	69
6.1	Estructura de carpetas "create-react-app"	75
6.2	Estructura de carpetas frontend	76
6.3	Documentos internalización: español e inglés	77
6.4	Configuración de capacitor	78
6.5	Scripts para ejecutar la aplicación	78
6.6	Aplicación en Android	79
6.7	Aplicación en IOS	79
6.8	Integración de ProSafe	80
6.9	Estructura de archivos:Servidor demo	80
6.10	Integración de ProSafe	81
6.11	Ejemplo de respuesta de una pregunta con broker	83
7.1	Página principal de <i>login</i>	85
7.2	Creación grupo: Añadir grupo vacío	86
7.3	Creación grupo: Añadir información	86
7.4	Creación grupo: Grupo creado	87
7.5	Creación de tema: Añadir tema vacío	87
7.6	Creación de tema: Añadir información	88
7.7	Creación de tema: Tema creado	88
7.8	Creación de plan: Añadir plan vacío	89
7.9	Creación de plan: Añadir información	89
7.10	Creación de plan: Plan creado	90
7.11	Inicio de sesión: Aplicación móvil	90
7.12	Notificación en dispositivo móvil Android	91
7.13	Pregunta asignada en dispositivo móvil Android	91
7.14	Pantalla de estadísticas (parte 1)	92
7.15	Pantalla de estadísticas (parte 2)	92
7.16	Menú de idiomas disponibles	93
7.17	Página de creación temas en español	93
7.18	Página de creación temas en inglés	94
7.19	Selector de idiomas en la aplicación móvil	94
7.20	Visualización información de la aplicación web	95
7.21	Visualización información aplicación móvil	95
7.22	Visualización información de usuario aplicación web	96
7.23	Visualización información aplicación móvil	96
7.24	Visualización recursos (aplicación web)	97
7.25	Visualización de recursos (aplicación móvil)	98
7.26	Campos requeridos: Creación plan de capacitación	99
7.27	Formato del campo: Correo electrónico y contraseña	99
7.28	Campos requeridos mínimos: Opciones en preguntas	100
7.29	Dialogo de información: Creación plan de capacitación con éxito	100
7.30	Dialogo de información: Creación plan de capacitación fracasado	101
7.31	Dialogo de confirmación: Eliminar pregunta	101

Índice de tablas

2.1	Tabla de comparación: Aplicaciones sobre gestión PRL	13
2.2	Tabla de comparación: Aplicaciones aprendizaje sobre PRL	14
3.1	Caso de uso: Crear grupo.	29
3.2	Caso de uso: Editar grupo.	29
3.3	Caso de uso: Visualizar grupo.	30
3.4	Caso de uso: Eliminar grupo.	30
3.5	Caso de uso: Crear plan de capacitación.	31
3.6	Caso de uso: Editar plan de capacitación.	32
3.7	Caso de uso: Visualizar plan de capacitación.	32
3.8	Caso de uso: Eliminar plan de capacitación.	33
3.9	Caso de uso: Crear tema.	34
3.10	Caso de uso: Editar tema.	35
3.11	Caso de uso: Visualizar tema.	35
3.12	Caso de uso: Eliminar tema.	36
3.13	Caso de uso: Añadir pregunta.	37
3.14	Caso de uso: Editar pregunta.	38
3.15	Caso de uso: Visualizar pregunta.	38
3.16	Caso de uso: Eliminar pregunta.	39
3.17	Caso de uso: Realizar pregunta.	39
4.1	Tabla endpoint <code>"/people/employees"</code>	43
4.2	Tabla de los endpoint del recurso employees	44
4.3	Tabla endpoint <code>"/people/employees/id"</code>	44
4.4	Tabla de los parámetros del endpoint: <code>"/people/employees/id"</code>	44
4.5	Tabla endpoint <code>"/people/departments"</code>	44
4.6	Tabla de los parámetros de la colección departments	44
4.7	Tabla endpoint <code>"/people/departments/id"</code>	45
4.8	Tabla de los parámetros del endpoint: <code>"/people/departments/id"</code>	45
4.9	Tabla endpoint <code>"/training/groups"</code>	45
4.10	Tabla de los parámetros del recurso groups	45
4.11	Tabla endpoint <code>"/training/groups/id"</code>	46
4.12	Tabla de los parámetros del endpoint: <code>"/training/groups/id"</code>	46
4.13	Tabla endpoint <code>"/training/plans"</code>	46
4.14	Tabla de los parámetros del recurso plans	46
4.15	Tabla endpoint <code>"/training/plans/id"</code>	47
4.16	Tabla de los parámetros del endpoint: <code>"/training/plans/id"</code>	47
4.17	Tabla endpoint <code>"/training/topics"</code>	47
4.18	Tabla de los parámetros del recurso plans	47
4.19	Tabla endpoint <code>"/training/topics/id"</code>	48
4.20	Tabla de los parámetros del endpoint: <code>"/training/topics/id"</code>	48
4.21	Tabla endpoint <code>"/stats/"</code>	48
4.22	Tabla endpoint <code>"/stats/:collection/count"</code>	49
4.23	Tabla endpoint <code>"/stats/employees/score"</code>	49

4.24	Tabla endpoint <code>"/training/tests/id/answer"</code>	49
4.25	Tabla de los parámetros del endpoint: <code>"/training/tests/id/answer"</code>	49
4.26	Tabla endpoint <code>"/training/posts"</code>	50
4.27	Tabla de los parámetros del recurso plans	50
4.28	Tabla endpoint <code>"/training/posts/id"</code>	50
4.29	Tabla de los parámetros del endpoint: <code>"/training/posts/id"</code>	50
4.30	Tabla endpoint <code>"/security/badge"</code>	50
5.1	Tabla de comparación de frameworks del lado del cliente	70
5.2	Tabla de comparación de frameworks del lado del servidor	73
A.1	Caso de uso: Iniciar sesión.	107
A.2	Caso de uso: Cerrar sesión.	107
A.3	Caso de uso: Ver información del usuario	108
A.4	Caso de uso: Ver información sobre la aplicación	108
A.5	Caso de uso: Visualizar recursos.	108
A.6	Caso de uso: Visualizar métricas.	109

CAPÍTULO 1

Introducción

La **prevención de riesgos laborales**, en adelante PRL, es uno de los pilares fundamentales con los que cuenta una organización y es **regulada por la Ley 31/1995, de 8 de noviembre [1]**. El objetivo de la PRL es **garantizar que los trabajadores cuenten con una protección ante la actividad que desempeñan en su lugar de trabajo o que deriven del mismo**, consiguiendo evitar o minimizar riesgos y accidentes. Para conseguir este objetivo se plantean una serie de normativas que deben de cumplir los empresarios, trabajadores y administraciones públicas, con tal de mejorar las condiciones de trabajo.

La **capacitación del personal ante riesgos laborales** es una de las medidas planteadas por esta normativa, requiriendo que los trabajadores cuenten con la formación necesaria para poder actuar ante los posibles riesgos que se puedan originar durante la actividad laboral a la que estén asignados. Sin embargo, aunque las formaciones son un punto clave para la capacitación de los empleados en las organizaciones, estas **suelen realizarse solo durante el primer contacto entre el trabajador y la empresa, no continuando durante la vida laboral del empleado en su organización**.

Este Trabajo Fin de Máster (TFM) **propone una aplicación que logre mejorar la capacitación sobre PRL** a través de la evaluación continua de los conocimientos a lo largo de la vida del trabajador en la empresa, y la extracción de información por parte del responsable de riesgos laborales para realizar un seguimiento de los empleados. Esta aplicación descrita se denomina ProSafe y consta de un panel de gestión destinada al responsable de riesgos laborales y una aplicación móvil en el que se realiza la evaluación de contenidos a los empleados de la organización por medio de alertas, simulando una emergencia real.

1.1 Motivación

En este apartado se va a proceder a detallar una serie de razones por las cuales se ha realizado este proyecto.

Por un lado, después de haber cursado la asignatura de Sistemas de Gestión de Emergencias, era notable observar que este **dominio no contaba con el desarrollo que cabría esperar de un sistema crítico** como el que nos ocupa. En cualquiera de las tres dimensiones de un sistema de gestión de emergencias tales como: la preparación, la respuesta y la investigación, existen aún procesos que cabría mejorar. Una de las opciones que existen para la mejora de estos procesos es la integración de la ingeniería de software en cualquiera de estas dimensiones, consiguiendo desarrollar sistemas más robustos que ayuden en la correcta gestión de emergencias en las que se puedan ver envuelto cualquier individuo. Tras la observación de los procesos de un gran número de empresas en cuanto

a riesgos laborales, se ha concluido que **existe un aprendizaje escaso al principio de la incorporación de un trabajador a su puesto de trabajo, no contando con ningún seguimiento ni refuerzo de los conocimientos adquiridos**. Por este motivo, se ha decidido diseñar una aplicación que consiga solventar estos problemas.

Asimismo, este proyecto **parte del modelo CiET (Continuous improvement for Emergency Management Training) [2]**, encargado de ayudar en la gestión de planes de capacitación de los empleados de una organización. Este modelo **funciona como base para el desarrollo del proyecto, sirviéndose de términos y conceptos que se definen en CiET**. También, el hecho de contar con la dirección de la profesora Patricia Alexandra Quiroz Palma, investigadora e ideadora del modelo CiET, forma parte de una de las motivaciones principales del proyecto, pudiendo consultar de primera mano las dudas que pudieran surgir con la integración del mismo.

Además, la creación de un producto software requiere visitar una serie de etapas y, en este caso en concreto se hará uso de diferentes tecnologías en auge. Todo ello, supone un gran aliciente para realizar el proyecto y así **conseguir aprender y preservar los conocimientos adquiridos en la realización del TFM**, que ayudarán a una salida profesional más exitosa.

1.2 Objetivos

El objetivo principal tras este TFM es la creación de una plataforma que sirva para la mejora de la capacitación de los empleados de una organización cuya actividad principal es la docencia. Para lograr este propósito se plantea la realización de una serie de puntos que se listarán a continuación:

- **Diseñar una aplicación web y móvil con interfaces amigables e intuitivas**, haciendo de esta forma que los usuarios que la utilicen puedan centrarse en el aprendizaje de los contenidos y no en el uso de la misma. Esta aplicación será **creada a partir de la extensión del modelo CiET**¹, especificándolo al ámbito de la prevención de riesgos laborales.
- Dotar de una **plataforma web que muestre gráficas y estadísticas** de diferentes datos sobre la organización para llevar un seguimiento **del proceso de capacitación de los usuarios** por parte del responsable de prevención de riesgos laborales de la empresa.
- **Aprender y profundizar diferentes tecnologías, lenguajes de programación y frameworks** en auge para la creación de aplicaciones web y móvil.
- **Poner en práctica conceptos aprendidos en el máster** para mejorar el proceso de creación de software.

1.3 Metodología

Como modelo para la elaboración del producto software se ha seguido una metodología en cascada [3]. Esta metodología se caracteriza por contar con un desarrollo marcado por su naturaleza secuencial, donde existen diferentes fases progresivas. A continuación, se listarán las etapas, explicando brevemente la función principal de cada una de ellas:

¹CiET: Marco para la mejora continua de la capacitación en el manejo de emergencias (Continuous improvement for Emergency Management Training)

- **Etapa de análisis:** Fase dedicada al análisis exhaustivo de los requisitos propios del sistema que se quiere desarrollar. Estos requisitos son documentados y especificados detalladamente al principio del proceso.
- **Etapa de diseño:** Fase dedicada al estudio de diferentes opciones y decisiones que marcarán el transcurso del software planteado.
- **Etapa de desarrollo:** Fase dedicada a la programación e implementación del producto software.
- **Etapa de pruebas:** Fase dedicada a la detección de errores o fallos cometidos en cualquiera de las etapas anteriores.
- **Etapa de despliegue o instalación:** Fase dedicada a poner en funcionamiento el sistema.
- **Etapa de mantenimiento:** Fase dedicada a la instalación, migración, soporte y mantenimiento del software.

En la figura 1.1 se puede observar un gráfico representativo de las etapas que conforman esta metodología.

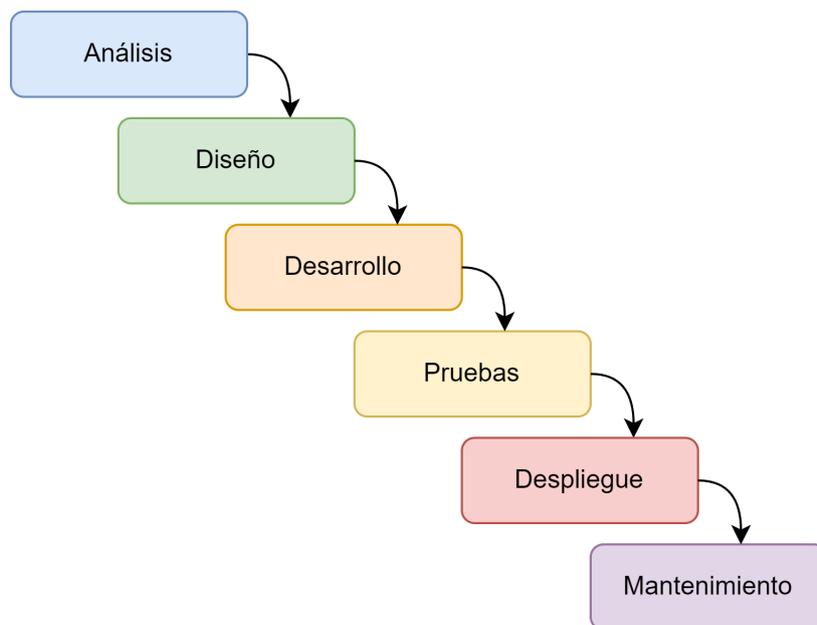


Figura 1.1: Fases de la metodología en cascada

La elección de esta metodología frente a otras recae en la idea de que el proyecto planteado cuenta con unos requisitos claros e invariables desde un primer momento. Además, en esta metodología no se requiere una disponibilidad del cliente total en gran parte del proyecto, solo en la primera fase de análisis y en los hitos que vayan surgiendo a lo largo del mismo, pudiendo desarrollar el software de manera más independiente.

1.4 Estructura de la memoria

La presente memoria del trabajo de fin de máster está distribuida en diferentes capítulos, cada uno de ellos dividido en apartados, donde se abordarán las diferentes fases que

se han llevado a cabo para la realización tanto del proyecto software como la extensión del modelo CiET. De esta manera, la memoria se estructura de la siguiente forma:

- **Capítulo 1:** Se introduce el tema del trabajo desarrollado, se exponen las motivaciones por las que se ha llevado a cabo, los objetivos y la metodología utilizada.
- **Capítulo 2:** Se muestra la situación actual de la normativa vigente sobre prevención de riesgos laborales, se exponen una serie de aplicaciones referentes a la capacitación de los individuos, comparándolas entre si, y se expone en última instancia el estado actual del modelo de referencia utilizado (CiET).
- **Capítulo 3:** Se realiza un análisis detallado de la propuesta planteada, describiendo de forma general sus características, relacionándolo con el modelo CiET y exponiendo los casos de uso y los requisitos propios de la aplicación.
- **Capítulo 4:** Se detalla el diseño de la aplicación en cuanto a su estructura arquitectónica y el diseño de bases de datos e interfaces a través de diferentes diagramas que lo clarifiquen.
- **Capítulo 5:** Se detallan las tecnologías y herramientas que se han utilizado a la hora de realizar el proyecto, explicando para ello el motivo de su elección frente a otras opciones.
- **Capítulo 6:** Se comenta la fase de desarrollo de software del proyecto y la estructura utilizada para cada parte del proyecto, haciendo hincapié en las soluciones y decisiones tomadas ante las diferentes dificultades que han ido surgiendo a lo largo de esta etapa.
- **Capítulo 7:** Se exponen las pruebas realizadas para validar el correcto funcionamiento de la aplicación y se muestra un ejemplo de uso de la aplicación.
- **Capítulo 8:** Se concluye la memoria con una serie de deducciones obtenidas del desarrollo del producto software, además se exponen diferentes propuestas de mejoras que se podrían realizar en un futuro pero que no se han podido elaborar por la limitación de tiempo determinado por el proyecto.

CAPÍTULO 2

Estado del arte

En el presente capítulo se expone la legislación y normativa vigente relacionada con la prevención de riesgos laborales, además, se documentan una serie de aplicaciones sobre PRL que buscan la mejora de capacitación en este ámbito. Por último, se describe el modelo CiET, y se especifica concretamente el uso de este modelo en una aplicación con las mismas siglas.

2.1 Prevención de riesgos laborales

La prevención de riesgos laborales es regulada por la **Ley 31/1995[1] del 8 noviembre de 1995**, aunque esta ley ha sido modificada a lo largo de los años, siendo actualizada por última vez el 29 de diciembre del año 2014. Dicha ley cuenta con las **bases legales principales que regulan la seguridad de los trabajadores en su puesto de trabajo**. Además, en el artículo catorce se establece expresamente el derecho de los trabajadores a la protección eficaz ante los riesgos laborales que puedan ocurrir, obligando al cumplimiento de una serie de normas que se establecen en ella. Los principales objetivos que esta ley pretende alcanzar son los siguientes:

- **Mejorar las condiciones de trabajo**, consiguiendo elevar el nivel de protección en términos de seguridad y salud de los trabajadores.
- **Promover la educación en materia preventiva**.
- **Fomentar actividades y/o investigaciones** que mejoren las condiciones de trabajo y la reducción de riesgos laborales.

Según la ley, el **encargado de establecer las medidas cautelares y abonar el coste que estas supongan es el empresario**. Este, deberá de **adoptar todas las medidas de prevención necesarias** para así conseguir evitar y combatir riesgos, evaluar aquellos que no sean evitables, adaptar el trabajo, instruir a los trabajadores, planificar la prevención etc. Este trabajo recae en **los delegados de prevención, que tienen como competencia colaborar con la dirección, promover y fomentar la cooperación de trabajadores, ser consultados por el empresario y ejercer una labor de vigilancia y control sobre el cumplimiento de la ley**.

En cuanto a la formación de los trabajadores, el empresario debe de **garantizar que reciban tanto formación teórica como práctica, adecuada al trabajo que desempeñan en la organización** y que se caracterice por: ser flexible, para así evolucionar ante la aparición de nuevos riesgos y que se imparta en las jornadas laborales, o en su defecto en

horas extras que deberán de descontarse. Por otro lado, esta formación deberá de repetirse de forma periódica, aplicando un seguimiento de los posibles riesgos en los que están o pueden estar sometidos.

En términos de la aplicación de la ley, **cumplir con la legislación vigente supone una reducción de los accidentes**, consiguiendo espacios de trabajo más saludables y seguros, e incluso aumentando la productividad y competitividad de la empresa.

2.1.1. Normativa de prevención de riesgos laborales

Existe legislación vigente en el ámbito de la prevención de riesgos laborales, tanto para España, Europa como de forma internacional. Dicha normativa aborda aspectos claves sobre la prevención de riesgos laborales y explica de manera detallada conceptos transversales tales como la integración de planes y auditorías, la correcta vigilancia de salud de los trabajadores o la normativa específica referente a cada colectivo de la sociedad.

Por otro lado, existen diferentes riesgos que la normativa aborda. Se entiende por **riesgo laboral**, según el artículo cuarto, *"la posibilidad de que un trabajador sufra un determinado daño derivado del trabajo"*. Algunos de los riesgos de los riesgos en los que se pueden ver afectados los trabajadores son los siguientes:

- **Riesgos mecánicos:** Riesgos producidos por elementos sólidos consistentes que pueden originar lesiones, traumatismos, amputaciones, heridas, fracturas e incluso puede acabar con la vida del trabajador.
- **Riesgos químicos:** Riesgos producidos por el almacenamiento, transporte y gestión de residuos tóxicos, cancerígenos y/o peligrosos.
- **Riesgos físicos:** Riesgos producidos por la exposición de los trabajadores ante ruidos, vibraciones, ambiente térmico y radiaciones.
- **Riesgos biológicos:** Relacionado por la exposición de los trabajadores ante microorganismos perjudiciales que puedan causar algún tipo de enfermedad.
- **Riesgos ergonómicos:** Relacionados con aquellos que pueden dar lugar a problemas de salud derivadas de posturas incorrectas, movimientos repetitivos, manipulación de cargas etc.
- **Riesgos psicosociales:** Riesgos producidos por los niveles de estrés, fatigas laborales y monotonía que se da en el puesto de trabajo por parte de los empleados.

2.1.2. Prevención riesgos laborales: personal docente

Una vez expuestos los riesgos laborales más generales que recoge la normativa, en este apartado se va a profundizar aquellos riesgos a los que están expuestos el personal docente de una organización, ya que el trabajo de fin de máster se centra en el caso de uso de la PRL de una organización cuya actividad principal es la docencia, aunque ProSafe pueda aplicarse para cualquier otro ámbito.

En este ámbito, se pueden encontrar documentos [4] que sirven como manual sobre prevención de riesgos laborales en el ámbito docente, donde se recoge información sobre los posibles peligros a los que se ven sometidos en este sector. De esta manera, existen dos tipos de riesgos más específicos en los que se puede ver envuelto el personal docente: los **riesgos derivados del trabajo** y los **riesgos derivados de la carga del trabajo**.

Riesgos derivados del trabajo

Este tipo de riesgos **proviene del espacio de trabajo de los empleados** y se pueden dar de varios tipos:

- **Riesgo eléctrico** debidos a accidentes que pueden derivar del contacto con la corriente eléctrica, ya sea por una mala instalación o algún tipo de aparato eléctrico deteriorado.
- **Caídas al mismo/distinto nivel**, debido a mal estado en el lugar de trabajo, mala disposición del mobiliario donde se encuentra el docente o riesgo de caída de altura.
- **Riesgos medio ambiente**, debidos a contaminación de algún tipo en el ambiente tales como: el ruido, temperatura, humedad, ventilación e iluminación.
- **Riesgos químicos**, debidos a sustancias toxicas, exposición a agentes químicos por ingestión o inhalación en laboratorios.
- **Riesgos biológicos**, debidos a la exposición a virus o bacterias que pueden provocar algún tipo de enfermedad al trabajador.

Riesgos derivados de la carga del trabajo

Este tipo de riesgos **proviene de la cantidad de trabajo con la que cuentan los docentes**, tanto físicos como psíquicos, afectando algunas veces a la salud del trabajador. Algunos de los riesgos que se pueden observar son los siguientes:

- **Riesgo de la voz:** Riesgo relacionado con los problemas de la voz que pueden darse a los docentes, tales como la disfonía, afonía e incluso falta de voz.
- **Riesgos ergonómicos:** Riesgo relacionado por los problemas musculoesqueléticos que se pueden dar por malas posturas, levantar mucho peso o incluso relacionados con las pantallas de ordenador, y provocando efectos de fatiga física, visual e incluso mental.
- **Riesgos psicosociales:** Riesgos debidos a factores laborales tales como el estrés laboral, *burnout*¹ o el *mobbing*².

2.2 Aplicaciones PRL

En el presente apartado se van a exponer algunas de las aplicaciones referentes a PRL que existen actualmente en el mercado. En primer lugar, se van a exponer los dos grandes grupos en los que se dividen este tipo de aplicaciones:

- **Aplicaciones de gestión:** Como su propio nombre indica, su principal cometido es gestionar los aspectos relacionados con la prevención de riesgos laborales de una organización.

¹Burnout: Síndrome del desgaste laboral debido al estrés crónico, lo que produce un agotamiento emocional y físico, conductas negativas hacia clientes o compañeros y desmotivación ante los proyectos o trabajos planteados, lo que supone un descenso en la productividad laboral del trabajador.

²Mobbing: Acoso laboral de manera sistemática que hace que la víctima se vea influida y perjudicada por el miedo o desánimo por parte del hostigador o hostigadores.

- **Aplicaciones de aprendizaje:** Aplicaciones destinadas al aprendizaje o refuerzo de contenidos relacionados con la prevención de riesgos laborales. Estos contenidos pueden ser específicos o generales sobre PRL.

Cabe destacar, que algunas de estas aplicaciones pueden contener las dos características, pero al final su cometido principal es aquel en el que están clasificados. A continuación se especifican las diferentes aplicaciones sobre PRL encontradas, aunque cabe remarcar que no se ha encontrado ninguna *app* concreta sobre PRL en el ámbito docente.

2.2.1. Aplicaciones sobre gestión de PRL

Prevencion10

Se trata de una aplicación gratuita creada por el Instituto Nacional de Seguridad y Salud en el trabajo (INSST) destinada al asesoramiento de pequeñas empresas o autónomos, además de ofrecer recursos y vídeos de aprendizaje sobre prevención de riesgos laborales con el objetivo de conseguir evaluar y formar a aquellos usuarios que utilicen la aplicación. En la figura 5.2 se puede observar una captura de la pantalla principal del portal web, donde se visualizan los apartados de recursos, vídeos, preguntas frecuentes y acceso de usuarios.



Figura 2.1: Pantalla principal de la pantalla web de Prevención10

Dicha aplicación puede ser descargada en la PlayStore o ser accedida desde su página web con el enlace siguiente: www.prevencion10.es/, aunque cabe destacar que la aplicación móvil muestra la misma aplicación web redimensionada a tamaño de un dispositivo móvil, lo que da indicios de que no cuenta con un desarrollo nativo propiamente dicho.

Ergonomic checkpoints

Aplicación móvil en inglés diseñada por International Labour Office para la creación de listas de control personalizado con el objetivo de mejorar la ergonomía en los espacios de trabajo de una organización. Además de las listas predeterminadas que ofrece la aplicación, el usuario puede crear nuevas que se ajusten más a sus espacios de trabajo. En la figura 2.2 se puede observar una captura de la primera pantalla de la aplicación,

donde se muestran las opciones que antes se has descrito, además de información de la aplicación, una guía de uso y una recomendación de las mejores prácticas en cuanto a ergonomía que se podrían llevar a cabo en la organización.

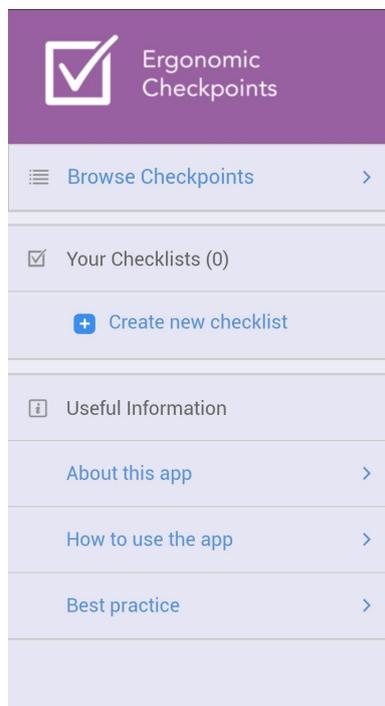


Figura 2.2: Pantalla de la aplicación móvil de Ergonomic checkpoints

Dicha aplicación puede ser descargada en la PlayStore a través del siguiente enlace: https://play.google.com/store/apps/details?id=com.aimermedia.ilo_checkpoints_ergonomics. Por otro lado, la misma compañía también ofrece una aplicación muy similar para la prevención del estrés en el trabajo, otro riesgo posible en el ámbito de los riesgos psicosociales que tiene una organización.

2.2.2. Aplicaciones de aprendizaje

Test Prevención Riesgo Laboral

Aplicación móvil diseñada para el aprendizaje y refuerzo de contenidos relacionados con los elementos de la ley que regula la prevención de riesgos laborales. Una vez el usuario se ha descargado la aplicación y accedido a la misma, se le muestran una serie de pruebas tipo test que puede elegir para realizar. Estos contienen un total de 20 preguntas y un máximo de 1 hora de tiempo para poder responderlas todas.



Figura 2.3: Pantalla de la aplicación móvil de tests

Dicha aplicación puede ser descargada en la PlayStore a través del siguiente enlace: <https://play.google.com/store/apps/details?id=testprevencionriesgoslaborales.com.oa>

APP y simulador Segutractor

Aplicación simulador sobre la conducción de un tractor, permitiendo a los usuarios detectar y actuar ante posibles riesgos en los que se vean envueltos, además de aprender hacer el uso correcto de los sistemas de seguridad con los que cuentan estos vehículos. Esta aplicación permite registrar al usuario para así anotar todos los avances que logre terminar. Una vez el usuario se ha registrado, puede practicar, realizar un tutorial, ver resultados, cambiar opciones del juego y observar los créditos, tal y como se puede observar en la figura 2.4.



Figura 2.4: Pantalla de simulación de Segtractor

Por otro lado, el usuario puede elegir entre una serie de modelos de tractores y escenarios diferentes para realizar la simulación. En la figura 2.5 se puede observar la interfaz dentro del juego.

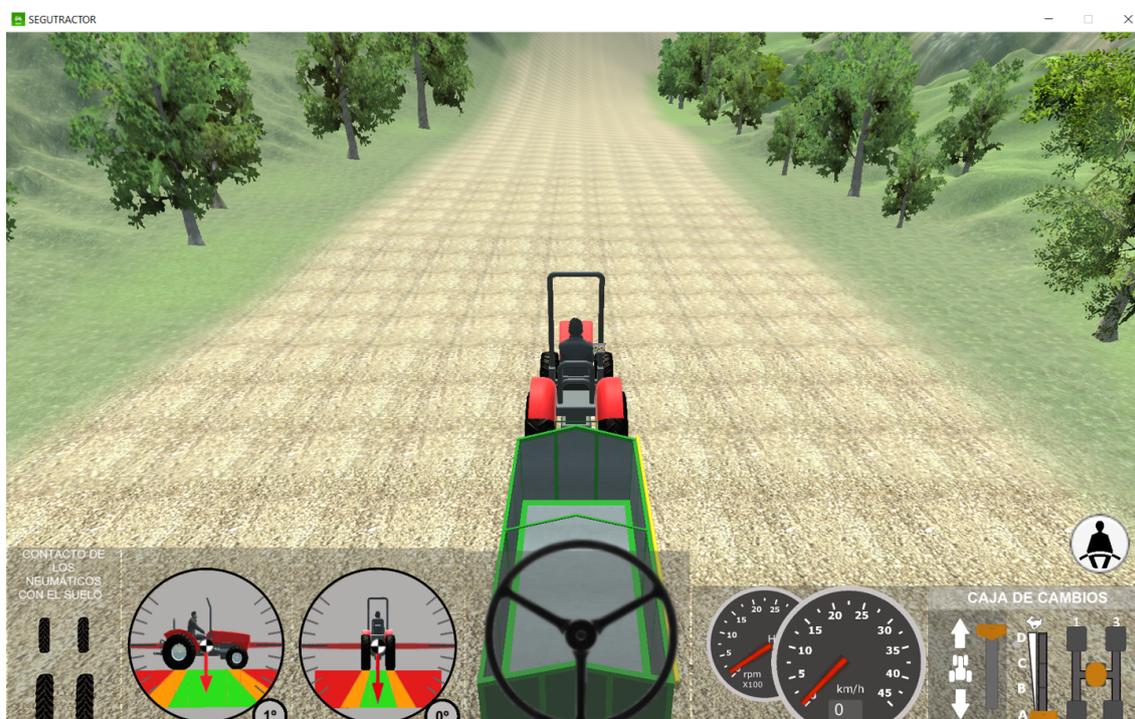


Figura 2.5: Pantalla de simulación de Segtractor

Dicha aplicación puede ser descargada en la página de INSST, concretamente en el siguiente enlace: www.insst.es/el-instituto-al-dia/app-segtractor, y ofrece soporte tanto para los sistemas operativos Windows y Mac, cómo los sistemas nativos de móviles IOS y Android.

APP y cuestionario Evaluación del trabajo repetitivo

Aplicación creada en 2021 que permite aprender sobre los riesgos y evaluar la carga física que se asocia a los trabajos repetitivos en un puesto de trabajo [5]. Esta aplicación es una actualización de una realizada en 2012 y permite realizar una serie de cuestionarios y acceder a recursos adicionales relacionados con el trabajo repetitivo. En la figura 2.6 se puede observar la pantalla principal de la aplicación, donde se muestra información sobre la aplicación, el acceso a cuestionarios y recursos además del aviso legal de INSST.

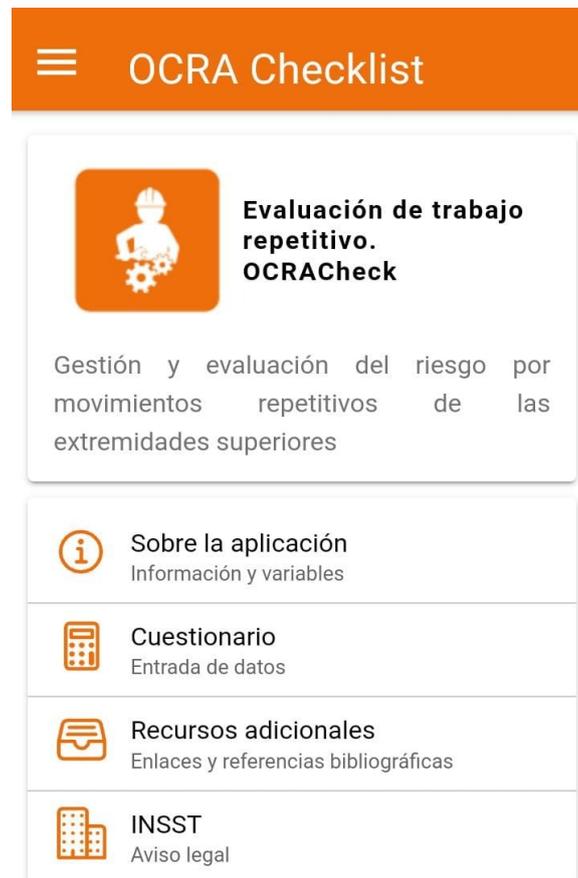


Figura 2.6: Pantalla principal de evaluación de trabajo repetitivo

Dicha aplicación puede ser descargada en la página de INSST, concretamente en el siguiente enlace: www.insst.es/documentacion/catalogo-de-publicaciones/evaluacion-de-trabajo-repetitivo-ocracheck

App: Prevención de riesgos laborales

Se trata de una aplicación [6] creada por Fasabe-Team que pretende dotar a los usuarios que la descarguen de nociones básicas sobre la prevención de riesgos laborales, tales como su legislación, derechos de los trabajadores, prevención de riesgos... Dicha aplicación proporciona recursos de texto para exponer estas ideas, ofreciendo a los usuarios un espacio donde puedan acceder de forma sencilla a estos contenidos. En la figura 2.7 se puede observar la pantalla referente a los contenidos de la aplicación, donde existe un buscador para encontrar el temario que se requiera.



Figura 2.7: Pantalla de contenidos de la aplicación Prevención de riesgos laborales

2.2.3. Comparación aplicaciones

En el siguiente apartado se van a exponer las aplicaciones en tablas comparativas con el fin de exponer sus características principales y analizar las funcionalidades que ofrecen al usuario.

Características	Prevencion10	Ergonomic checkpoints
Gestión de usuarios	Sí	No
Aplicación multiplataforma	Si, móvil y web	No, solo móvil
Riesgos PRL	Todos	Ergonómicos
Acceso	Gratuito	Gratuito
Idioma	Castellano, Euskera, Gallego y Catalán	Inglés
Recursos aprendizaje	Sí	No

Tabla 2.1: Tabla de comparación: Aplicaciones sobre gestión PRL

Características	Seguimiento	Trabajo repetitivo	Test prevención	APP PRL
Progreso guardado	Sí	Sí	Sí	Sí
Aplicación multiplataforma	Sí, app ordenador y móvil	No, app móvil	No, app móvil	No, app móvil.
Tipo de herramienta	Simulación y juego serio	HCI	HCI	HCI
Tipo de formación	Específica	Específica	General	General
Riesgos PRL	Físicos, ergonómicos, psicosociales	Todos	Todos	Todos
Aprendizaje continuo	No	No	No	No

Tabla 2.2: Tabla de comparación: Aplicaciones aprendizaje sobre PRL

Una vez expuestas las diferentes aplicaciones encontradas en el terreno de la prevención de riesgos laborales, se puede concluir que todas ellas cuentan con un desarrollo muy escaso para la importancia que recae en la seguridad de los trabajadores. Además, no se han encontrado aplicaciones concretas sobre PRL en el ámbito docente. Es por ello que se plantea la creación de una aplicación que ayude en el desarrollo de las competencias de los trabajadores a través de la expansión del modelo CiET.

2.3 CiET

CiET[2] es un modelo que se integra con otros marcos de evaluación tales como QuEP³ y que ayuda en la gestión de planes de capacitación con el objetivo de lograr la mejora continua y servir como apoyo al proceso de aprendizaje ante emergencias. De esta forma, CiET pretende contribuir en la formación y evaluación de los contenidos a través de diferentes recursos. Este modelo de capacitación está soportado por cuatro pilares fundamentales: *serious games*⁴, la gamificación, la resiliencia y la interacción humano computador, todo ello respaldado por la gestión de conocimiento.

El diagrama de clases de este modelo se puede observar en la figura 2.8, donde se ve reflejado que cada organización tiene que tener al menos un plan de capacitación, este contará con unos objetivos, unas técnicas, dimensiones y herramientas de aprendizaje y contenido, que corresponderán a una actividad o una fase de la emergencia. Además, cabe destacar que los interesados de una organización serán evaluados de estos contenidos.

³QuEP: Marco de identificación de riesgos en una organización con el objetivo de mejorar la gestión de emergencias (Quality of Emergency Plan Management).

⁴*Serious games*: Juegos diseñados con el objetivo de formación ante un tema determinado.

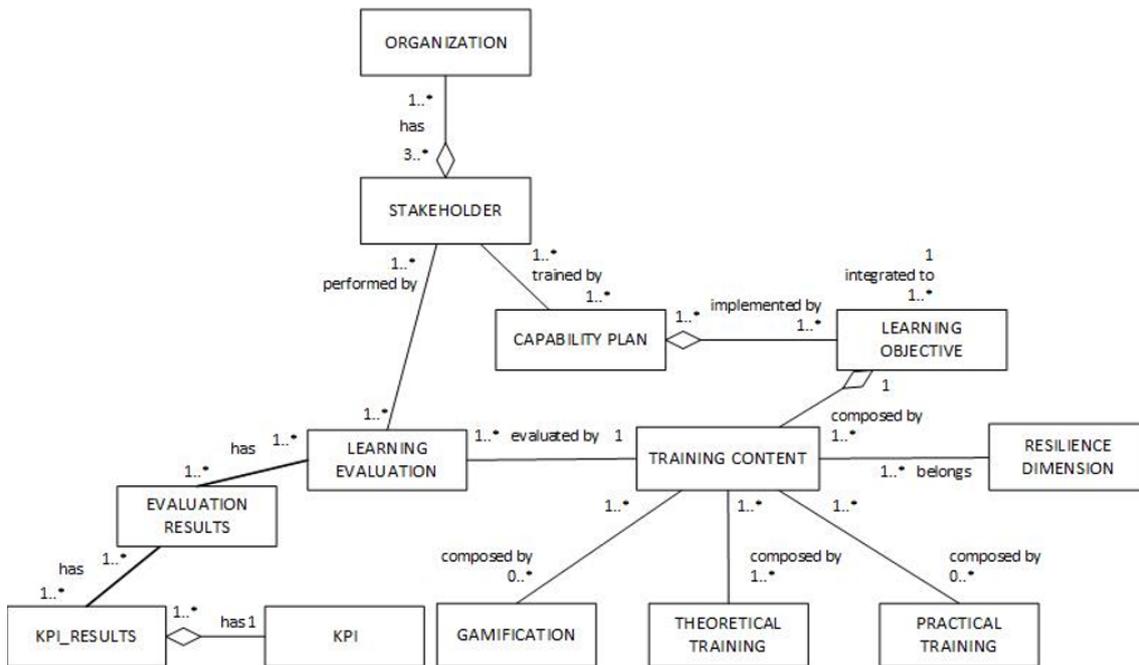


Figura 2.8: Diagrama de clases CiET (extraído de [7])

2.3.1. CiET como plataforma de entrenamiento

Una vez expuesto este modelo, se procedió a desarrollar un modelo específico (figura 2.9) para elaborar una herramienta que abordase los objetivos planteados. Las clases más relevantes que conforman el modelo son las siguientes:

- *Organization*: Referente a la organización que tiene que definir y proponer los planes de capacitación.
- *Stalkeholder*: Referente a los interesados de una organización. Existen los siguientes stalkeholders en un plan de emergencias:
 - Responsables de una organización
 - Planificadores técnicos de riesgos de emergencias
 - Trabajadores.
 - Ciudadanos externos a la organización.
 - Equipos de respuesta.
- *Exercise*: Referente a los ejercicios que realizan los *learners* de una organización. Estos ejercicios serán evaluados, tal y como se puede observar detallado en la clase "Evaluación".
- *Capability plan*: Referente al plan de capacitación que crea una organización. En dicho plan se definen una serie de objetivos, recursos y evaluaciones.
- *Objective*: Referente a los objetivos de un plan de emergencias.

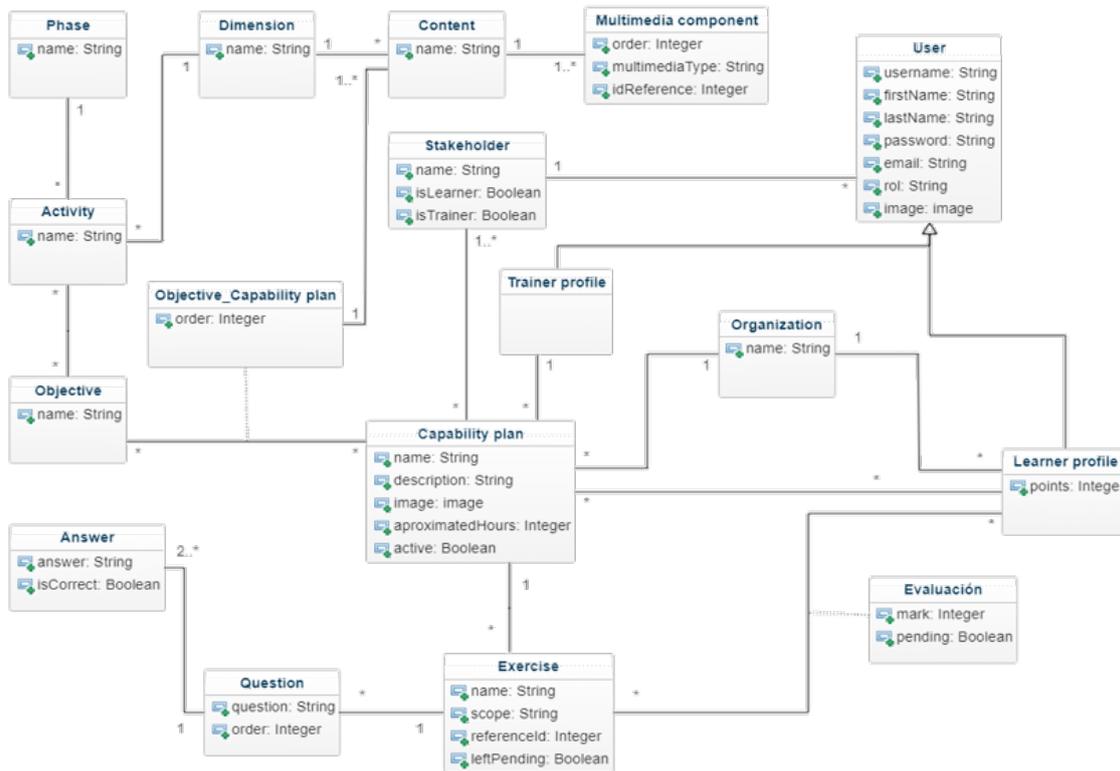


Figura 2.9: Diagrama de clases CiET plataforma (extraído de [8])

La herramienta de software web CiET está basada en el modelo de la figura 2.9, y ha sido desarrollada como parte del trabajo final de grado de Alberto Rausell Manchón, y dirigido por María Carmen Penadés Gramage y Patricia Alexandra Quiroz Palma [8]. Esta plataforma permite a las organizaciones gestionar los planes de capacitación de la empresa, además de servir como una plataforma de entrenamiento de los involucrados. En ella, se pueden distinguir tres roles bien diferenciados en cuanto a las acciones que pueden realizar en la plataforma: administrador, *trainer* y *learner*.

En términos generales, la herramienta permite la creación de planes de capacitación por parte de los *trainers* tal y como se puede observar en la figura 2.10. Aunque este tipo de usuarios también puede crear objetivos de capacitación o observar las estadísticas de los contenidos que realizan los *learners*.

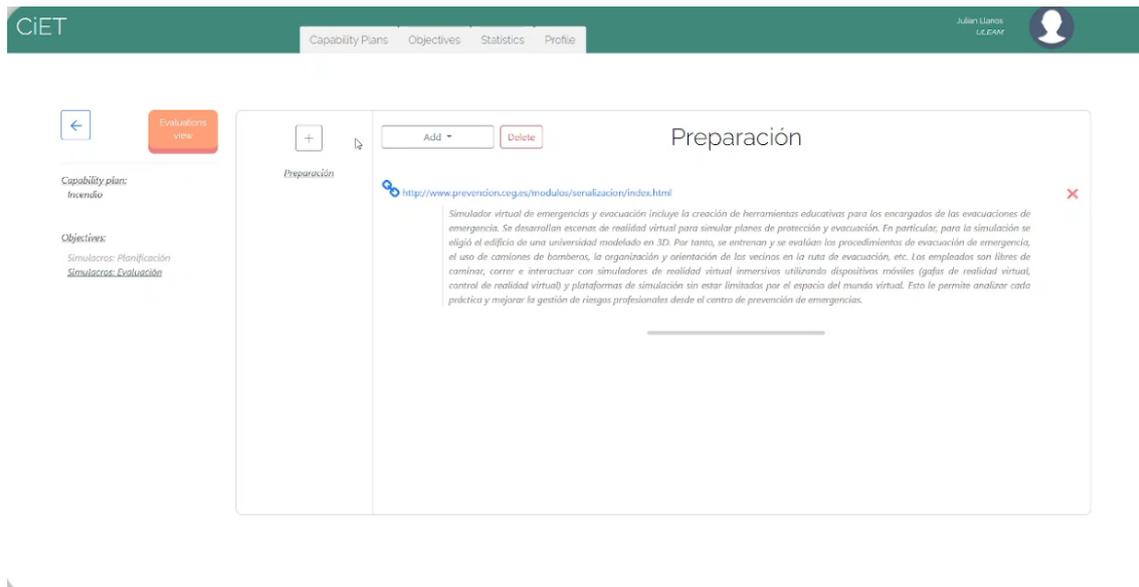


Figura 2.10: Pantalla de creación de planes de capacitación

Por otro lado, el usuario que se encarga de consumir y evaluarse de los contenidos creados por los *trainers* son los *learners*. En la figura 2.11 se puede observar la pantalla referente a la evaluación de los contenidos de un usuario.

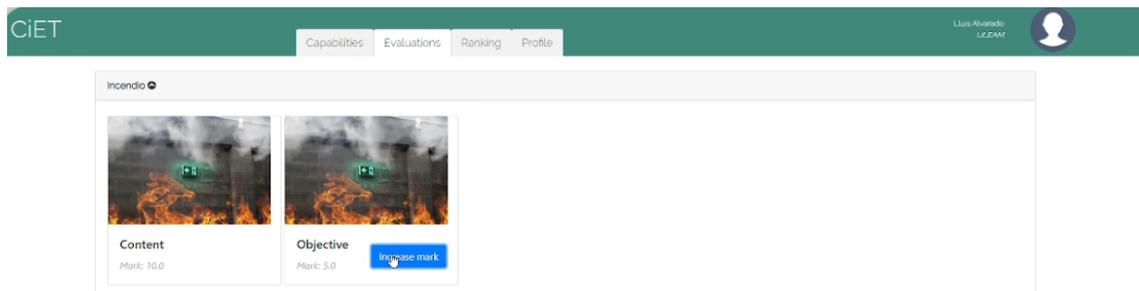


Figura 2.11: Pantalla de creación de evaluación de contenidos

CAPÍTULO 3

ProSafe: Propuesta de la solución

En el presente capítulo se expone la propuesta ProSafe, especificando sus características más relevantes y haciendo hincapié en la relación que existe con el modelo CiET.

3.1 Descripción general

ProSafe es una herramienta para la mejora de la capacitación de los empleados de una organización en términos de prevención de riesgos laborales. Dicha herramienta propone como característica principal la **integración de alertas simulando una emergencia real, para así conseguir evaluar la actuación o el conocimiento que poseen los usuarios ante dicha situación**, suponiendo una diferencia significativa para poder distinguirse de las aplicaciones existentes en el mercado. Estas alertas pueden darse en cualquier momento de la jornada laboral de un trabajador, considerándose un aprendizaje continuo y permitiendo al encargado de riesgos laborales de una empresa poder extraer información sobre el nivel de capacitación de los empleados que componen la misma. Esta plataforma de aprendizaje parte del modelo CiET expuesto en el apartado 2.3.

3.1.1. Características diferenciadoras

La propuesta Prosafe comparte características con las aplicaciones planteadas en el apartado 2.2, pero también tiene una serie de peculiaridades que hacen que se diferencie de las otras plataformas. Entre estas características se pueden encontrar:

- **Plataforma de aprendizaje y gestión sobre PRL para una organización**, abarcando todo el temario que el encargado de riesgos laborales de una empresa crea oportuno según la legislación. Cuenta con una formación general y específica.
- Plataforma que permite la **extracción de información sobre la capacitación de los empleados** por medio de estadísticas.
- **Aprendizaje continuo de los trabajadores** por medio de una evaluación adecuada según la información extraída.
- **Aplicación multiplataforma**: web y móvil.
- **Aplicación multilinguaje**: español e inglés. Soporta el aumento del número de idiomas de manera sencilla.

3.1.2. Aplicación de CiET

El modelo CiET ha sido utilizado como base para abordar la definición y desarrollo de la plataforma, pero cabe destacar que ProSafe difiere en varios aspectos. Tal y como se ha podido observar en el apartado 2.3, CiET propone una plataforma de aprendizaje y entrenamiento, no muy lejos de la idea principal de este trabajo de desarrollo, con la principal diferencia de que CiET está centrada en la capacitación de los stakeholders ante emergencias y ProSafe busca la capacitación ante riesgos laborales. Esa diferencia principal hace que los interesados que se definen en CiET difieran de los de la plataforma a desarrollar. En el caso de CiET, se habla de cinco tipos de stakeholders, que son propios de una situación de emergencia, pero en lo que respecta a la prevención de riesgos laborales solo existirían dos roles: planificador y empleados. Además, aunque CiET propone una aplicación web externa a las organizaciones, **ProSafe busca crear una aplicación web y móvil que se adapte a la organización en cuestión que requiera de estas funcionalidades.**

ProSafe se vale del concepto de rol definido por CiET. Como ya se ha visto en el apartado anterior, en esta plataforma se diferencia a los usuarios a través de los conceptos: *learner*, para designar un usuario que accede a la aplicación con el objetivo de aprender y *trainer*, para designar a un usuario que gestiona los contenidos y accede a las estadísticas.

Por otro lado, a través de la revisión sistemática que se realizó para la creación de CiET, se especificó una taxonomía con el objetivo de poder clasificar las capacidades de los interesados. En el caso de ProSafe, **la herramienta abordaría las capacidades referentes al entrenamiento individual, el entrenamiento por conocimiento y el entrenamiento basado en herramienta (HCI¹).** En la figura 3.1 se puede observar la taxonomía empleada a través de marcas.

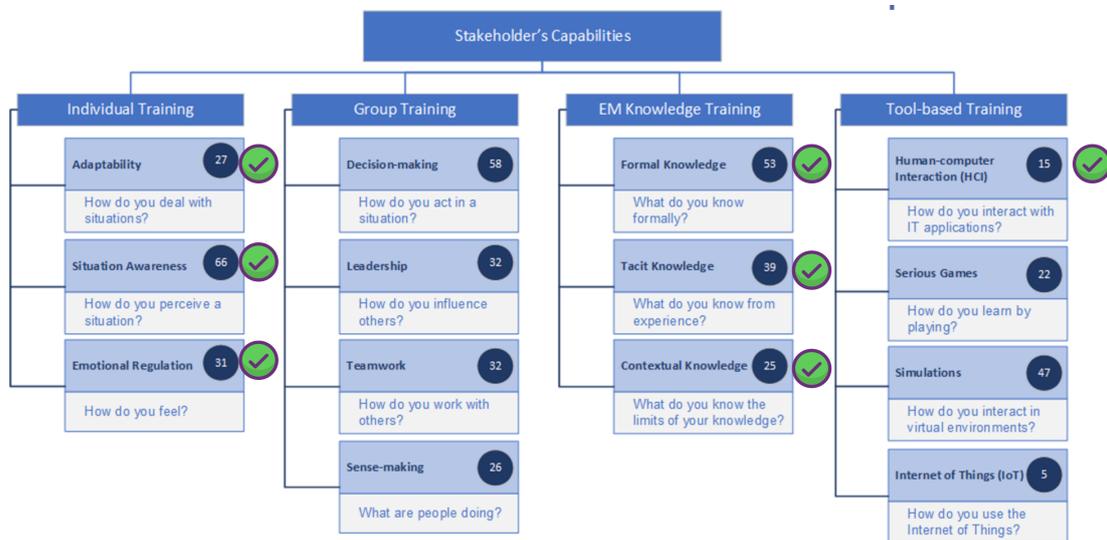


Figura 3.1: Clasificación: Capacidades de los interesados

3.1.3. Modelo inicial propuesto

Con el objetivo de mostrar un modelo inicial del sistema a través de los conceptos definidos por CiET, sin llegar a exponer el modelo completo y evitar de esta manera

¹HCI: Disciplina dedicada al diseño de sistemas informáticos que sean interactivos con el ser humano (Human-computer interaction)

una complejidad mayor, se ha hecho uso del diagrama de clases UML ². En la figura 3.2 se puede observar el diagrama de clases.

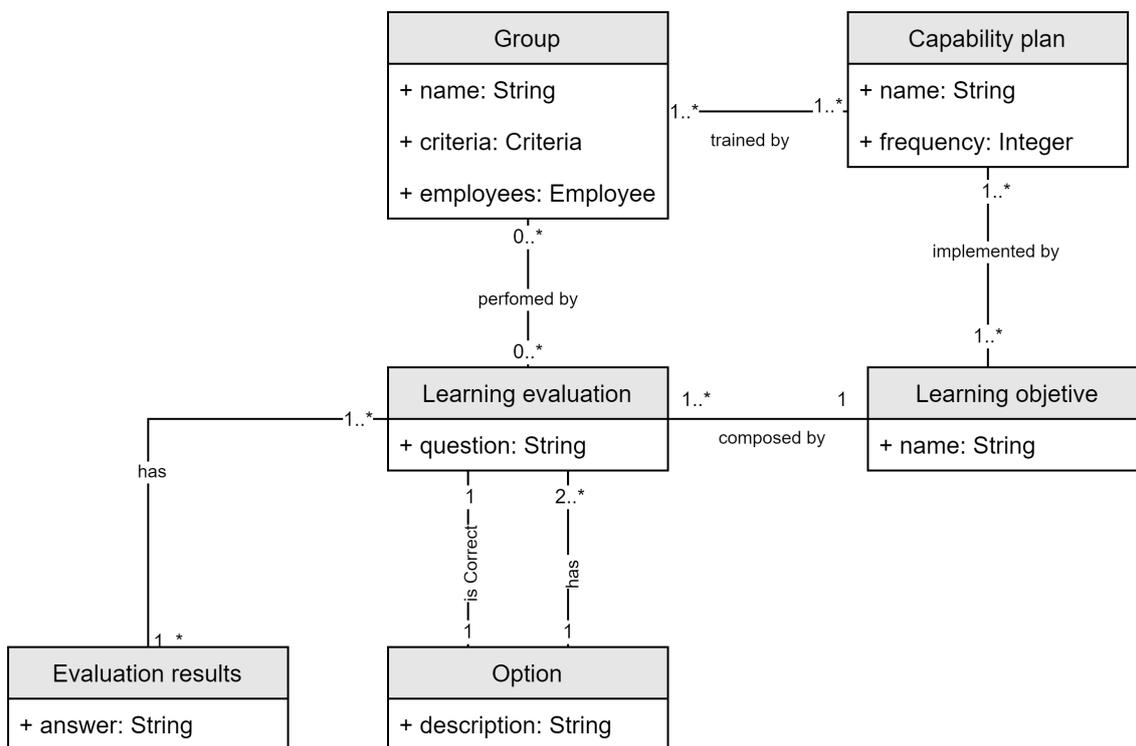


Figura 3.2: Modelo inicial propuesto

En dicho diagrama se pueden diferenciar las clases que se definen en los siguientes puntos:

- *Capability plan*: Hace referencia al plan de capacitación de una empresa. El plan de capacitación incluye los modos de evaluación creados para la aplicación, este plan de capacitación cuenta con un nombre, una frecuencia de envío, unos grupos a los que va dirigidos (*stakeholders*) y unos objetivos de aprendizaje (temas).
- *Group*: Grupos de empleados generados por parte del responsable de riesgos laborales. Existen dos tipos de grupos según la forma en la que son creados: por criterios o manuales. En primer lugar, los criterios posibles para la creación de grupos del primer tipo son: por departamento, sexo y/o edad. Por otro lado, el segundo tipo de grupo es creado por parte del responsable eligiendo uno a uno los empleados que quiere añadir. En relación con el modelo CiET, se podría hacer una analogía con la entidad Stakeholder, ya que cuenta con una serie de similitudes en las que se encuentran: en una primera instancia existirán unos grupos por defecto que serán generados a partir de una serie de empleados, correspondiendo estos a los los grupos de stakeholders para PRL.
- *Learning evaluation*: Referencia a la evaluación de los contenidos de PRL, por medio de preguntas catalogadas por un tema creado por el responsable de riesgos laborales.
- *Evaluation results*: Referencia a los resultados de la evaluación de los contenidos sobre PRL que realizan los empleados de una organización.

²UML: Lenguaje unificado de modelado (*Unified Modeling Language*)

- *Option*: Opciones que existen para cada test, para un test solo existirá una pregunta válida.

En términos generales, la herramienta Prosafe se puede definir a través de un paquete que será añadido en la organización pertinente, pero además de eso, se otorgarán dos aplicaciones con objetivos diferentes:

- **Aplicación web**: Panel de administración dirigido al encargado de riesgos laborales de la empresa. Dicho panel permite al *trainer* crear planes de capacitación, estos definen una serie de temas, que contienen preguntas, unos grupos a los que van dirigidos y una frecuencia de activación, definiendo en cuanto tiempo deben de ser mandadas las cuestiones. Además, también permite la gestión de los grupos y temas expuestos anteriormente, como también la visualización de estadísticas referentes al aprendizaje de los empleados de la organización.
- **Aplicación móvil**: Aplicación dirigida a los *learners* de la organización. El usuario puede observar información de tests anteriormente realizados y recursos genéricos de riesgos laborales. Además, al contar con la aplicación descargada en los dispositivos de los trabajadores, podrá recibir las alarmas de simulacros que deberán de contestar lo antes posible.

Diferencias con el modelo CiET

Como diferencias sustanciales al modelo CiET, esta plataforma será **ejecutada por cada organización, creando una instancia para cada una de ellas**, con lo que no es necesario crear una clase organización. El motivo de esta última decisión es que manejar el conjunto de datos de empleados, pedir permisos y gestionar la protección de estos datos supondría un coste mucho mayor del que se podría abordar en un proyecto de estas características. Es por este motivo, que se necesita una aplicación que esté compuesta por componentes débilmente acoplados, ya que se podría contar con información de diferentes bases de datos, o con otro tipo de estructura, por ello la aplicación debería de ser capaz de a través de un mínimo esfuerzo, y sin cambiar la lógica de negocio, poder utilizarse.

Por otro lado, se han descartado varias de las partes interesadas del modelo CiET, ya que en el caso de la prevención de riesgos laborales solo existirían los planificadores (técnicos de PRL), los trabajadores y los responsables de la organización. Además, se ha decidido eliminar la clase *Stakeholder*, ya que la entidad Group hace un papel parecido, debido a que Group crea grupos de empleados, estos grupos pueden creados de forma manual o por medio de criterios. En el caso de la forma por criterios, se puede especificar que se creen tres grupos conteniendo cada uno los grupos de los interesados anteriormente planteados: planificadores, trabajadores y responsables.

3.1.4. Estadísticas

Aunque en este apartado se podrían añadir un número elevado de gráficas que serían de utilidad para adquirir información sobre la capacitación de los empleados, se ha decidido resumirlas en tres tipos básicos, debido a las limitaciones de tiempo con las que se cuenta. En los siguientes puntos se detallan los tipos de gráficas utilizadas:

- **Gráfico de columnas apiladas** (*stacked column chart*): Este tipo de gráfico se va a emplear para dar información sobre las preguntas contestadas, siendo estas correctas o incorrectas. De este modo, se empleará tanto para los grupos creados, como

para los temas, dónde se mostrará el número de preguntas correctas, incorrectas y totales para cada uno de los recursos comentados. En la figura 3.3 se puede ilustrar el tipo de gráfico detallado.

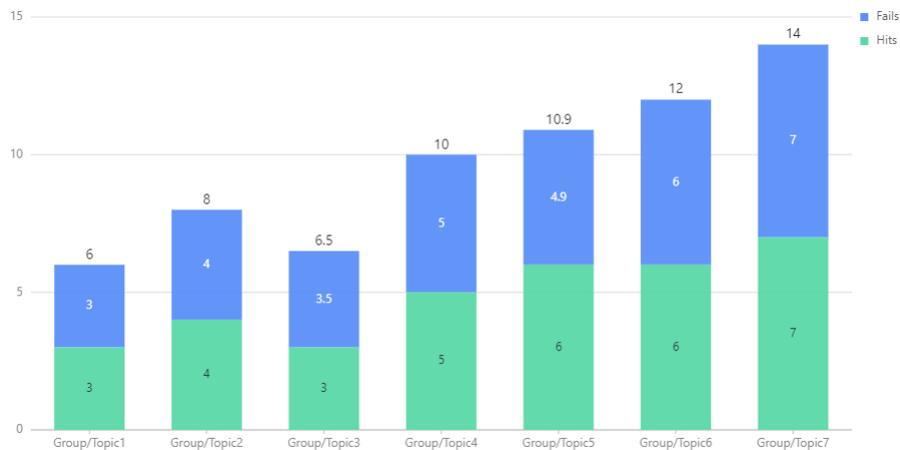


Figura 3.3: Ejemplo de gráfico de columnas apiladas

- **Gráfico circular o de tarta (pie chart):** Este tipo de gráfico se va a emplear para dar información sobre los grupos creados en la organización y el número de empleados que los componen. En la figura 3.4 se puede ilustrar el tipo de gráfico detallado.

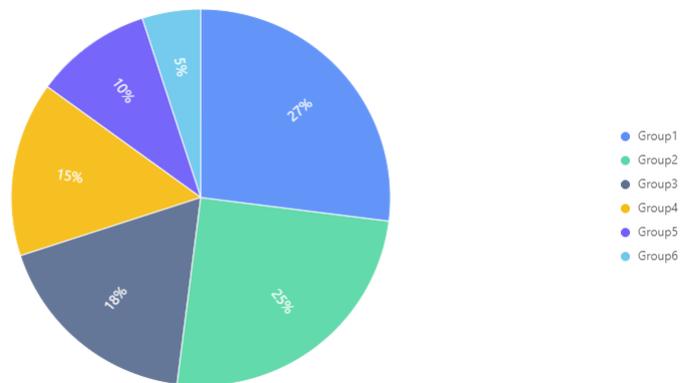


Figura 3.4: Ejemplo de gráfico circular

- **Gráfico de barras (bar plot):** Este tipo de gráfico se va a emplear para dar información sobre los puntos con los que cuenta cada grupo creado en la organización. En la figura se puede observar un ejemplo del gráfico empleado. En la figura 3.5 se puede ilustrar el tipo de gráfico detallado.

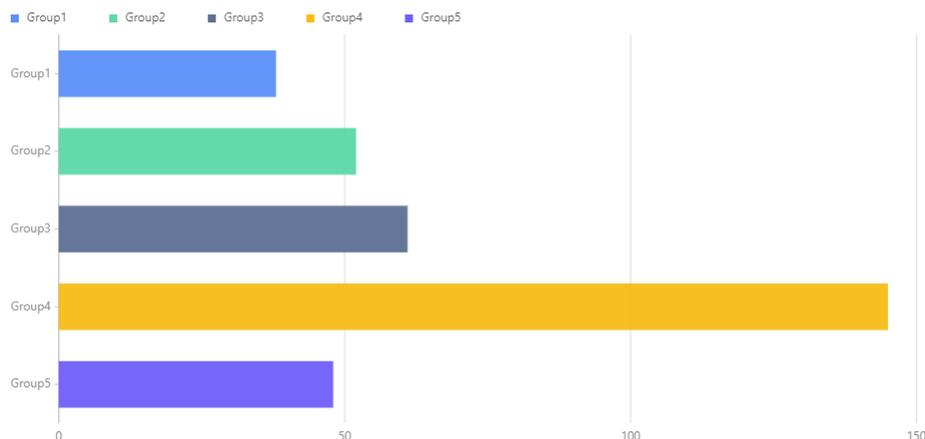


Figura 3.5: Ejemplo de gráfico de barras

3.1.5. Uso del producto software

ProSafe es una aplicación que debe de ser utilizada por la organización que requiera de este servicio. Para este propósito, se han valorado dos aproximaciones:

- **Desplegar como un servicio independiente**, siendo este servicio el encargado de exponer sus *endpoints* para que la empresa en cuestión pueda conectarse y acceder a las funcionalidades que ofrece ProSafe.
- **Integrar el servicio en la organización**, a través de la incorporación del servicio en el `pom.xml` de la organización en forma de paquete y, acceder de esta forma a las utilidades que ofrece.

A pesar de que la primera aproximación cuenta con la ventaja de ser un servicio que corre independiente a la organización y, que puede ser utilizado por organizaciones que no cuenten con un *backend* desarrollado en Java, se ha decidido descartar esta idea ya que el tiempo establecido para este proyecto es muy limitado, y esta propuesta requeriría un plazo más prolongado. Por lo tanto, la idea de **integrar el servicio en una organización resulta la idea más factible**, pudiendo utilizar todas las utilidades de ProSafe.

3.2 Especificación de requisitos

3.2.1. Requisitos funcionales

Los requisitos funcionales definen las acciones que debe de realizar el sistema software que se vaya a desarrollar. En los siguientes puntos se detallan los requisitos funcionales del sistema, en un primer lugar aquellos que son globales para los dos componentes, y a continuación los referentes al portal web y la aplicación móvil.

Generales

- El sistema permitirá el inicio de sesión en la aplicación por medio de las credenciales del usuario. Dependiendo del rol del usuario, podrá acceder a un componente o al otro. En este caso, el *learner* puede acceder a la aplicación móvil, mientras que el *trainer* puede acceder al portal web.

- El sistema permitirá el cierre de sesión por parte del usuario.
- El sistema mostrará diálogos de información y confirmación a la hora de la gestión de recursos en la aplicación.
- El sistema permitirá acceder a un apartado de ayuda que mostrará información sobre el uso y el propósito de la aplicación.
- El sistema validará los datos ofrecidos por los usuarios.

Portal web

- El sistema permitirá al usuario autorizado gestionar los planes de capacitación: creación, eliminación y edición.
- El sistema permitirá al usuario autorizado enviar preguntas a grupos a través de los planes de capacitación.
- El sistema permitirá listar todos los planes de capacitación creados.
- El sistema permitirá al usuario autorizado gestionar los temas: creación, eliminación y edición.
- El sistema permitirá listar todos los temas creados.
- El sistema permitirá al usuario autorizado gestionar las preguntas: creación, eliminación y edición.
- Al ingresar preguntas, estas deberán de siempre estar asociados a un tema.
- El sistema permitirá listar todas las preguntas creadas.
- El sistema permitirá al usuario autorizado gestionar los grupos: creación, eliminación y edición.
- El sistema permitirá listar todos los grupos creados: tanto los creados manualmente como los creados de forma automática según una serie de criterios.
- El sistema permitirá listar todas las estadísticas referente a la evaluación efectuada a los usuarios.
- El sistema validará que no se repita el nombre proporcionado por el responsable para los planes de capacitación, preguntas, grupos y temas.

Aplicación móvil

- El sistema permitirá al usuario autorizado listar la última información relacionada con los tests realizados.
- El sistema permitirá al usuario realizar test a través de alertas dirigidas a través del *trainer*.
- El sistema permitirá acceder a información referente a contenidos sobre PRL.

3.2.2. Requisitos no funcionales

Los requisitos no funcionales hacen referencias a los requisitos que especifican una serie de criterios referidos a las cualidades o características de un software. En este caso, se pueden especificar los siguientes requisitos no funcionales:

- **Disponibilidad:** Garantizar que el software sea accesible y utilizable por parte de los usuarios que lo utilicen.
- **Adaptabilidad:** Garantizar que el software sea capaz de ser adaptado a diferentes entornos.
- **Escalabilidad:** Garantizar que el software ser capaz de adaptarse y responder respecto a un aumento de datos o carga de trabajo.
- **Reusabilidad:** Garantizar que el software se puede reutilizar en diferentes organizaciones, o temas sobre PRL.

3.3 Casos de uso

En este apartado se detallan los casos de uso de la aplicación. En la tabla 3.6 se detallan los diferentes actores que existen en la aplicación: *learner* y *trainer*.

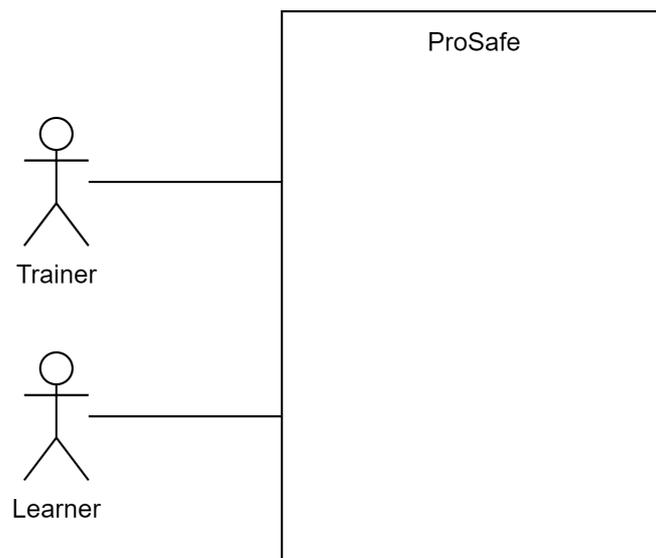


Figura 3.6: Diagrama de contexto de ProSafe

En la figura 3.7 se puede observar el diagrama de casos de usos para el actor *learner*. Cabe destacar, que el caso de uso “Visualizar información” hace referencia tanto a la información propia de la aplicación como la referente a los datos del empleado.

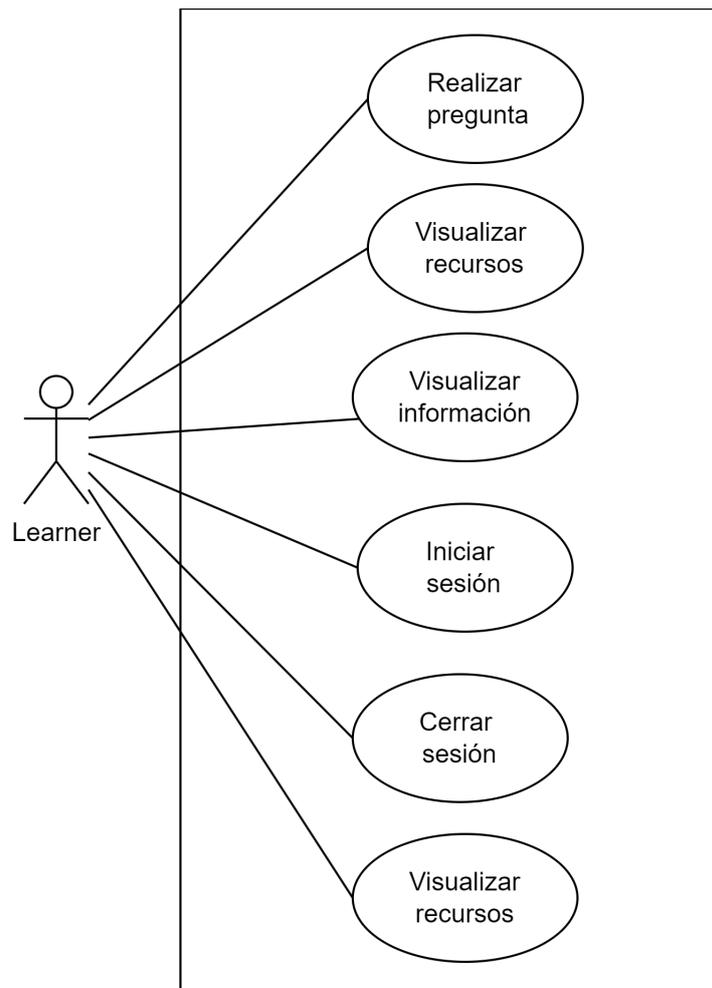


Figura 3.7: Diagrama de casos de uso para el actor learner

En la figura 3.8 se puede observar el diagrama de casos de usos para el actor *trainer*. Cabe destacar que las acciones de gestión engloban tanto la creación, la eliminación, la edición y visualización de los recursos mencionados.

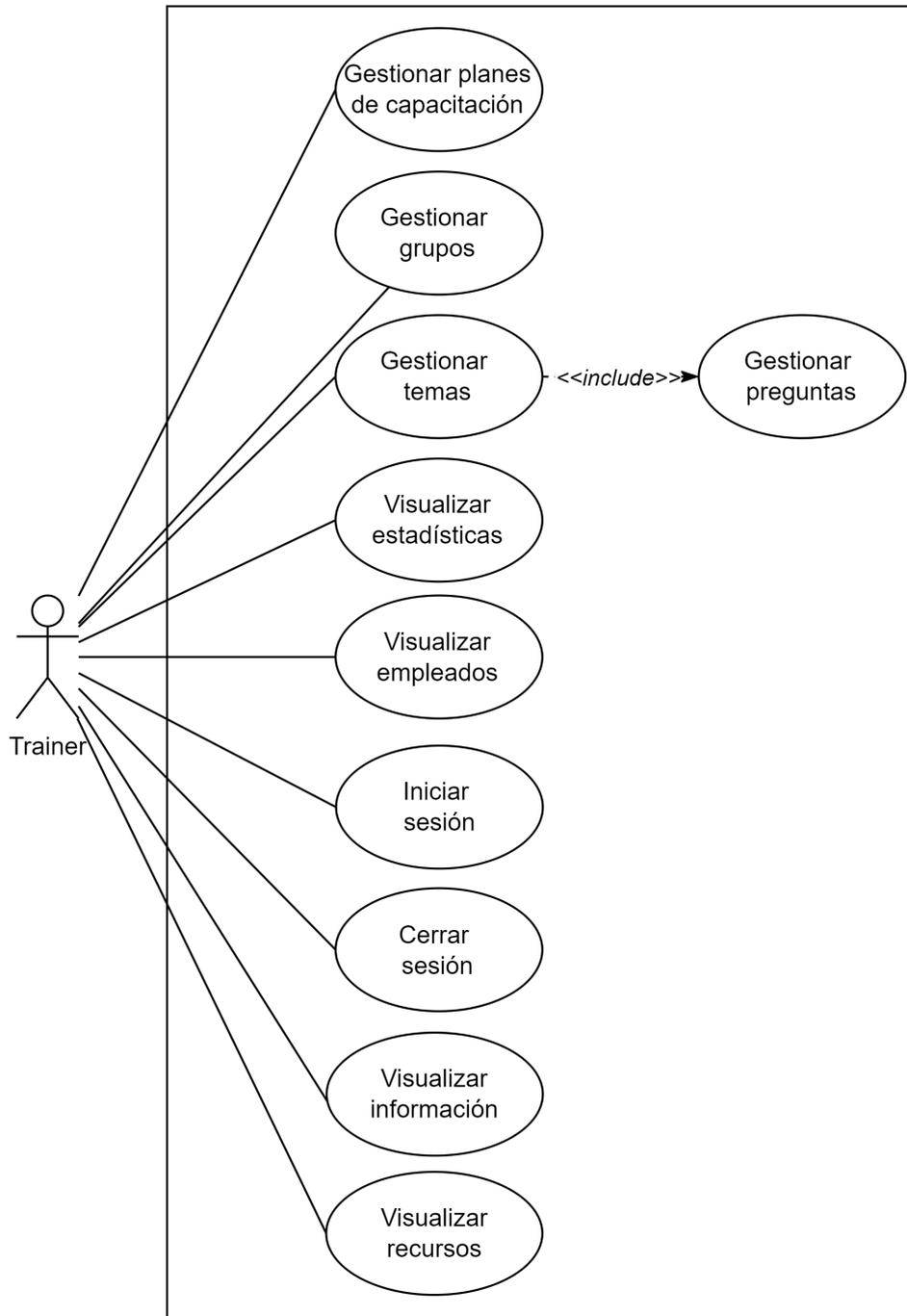


Figura 3.8: Diagrama de casos de uso para el actor trainer

A continuación se va a proceder a detallar los casos de uso más relevantes expuestos en los diagramas anteriores, con el fin de clarificar las acciones que pueden realizar los usuarios en la aplicación. En el anexo A se pueden observar los casos de uso restantes.

- **Crear grupo:** El usuario puede crear grupos manuales de empleados de la organización. En la tabla 3.1 se puede observar la definición más detallada del caso de uso.

Crear grupos	
Actores	Trainer
Resumen	El usuario crea grupos de empleados.
Precondición	El usuario debe de estar identificado.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clicka</i> en el enlace del menú referente a la administración, "Administración".	2. El sistema muestra una serie de paneles de administración
3. El usuario <i>clicka</i> en la pestaña referente a grupos.	4. El sistema muestra el apartado de grupos.
5. El usuario <i>clicka</i> en botón "Crear grupo".	
6. El usuario rellena la información solicitada.	
	7. El sistema valida la información
	8. El sistema da de alta el nuevo grupo.
	9. El sistema muestra en pantalla el nuevo grupo creado.

Tabla 3.1: Caso de uso: Crear grupo.

- **Editar grupo:** El usuario puede editar la información de aquellos grupos manuales que estén creados. En la tabla 3.2 se puede observar la definición más detallada del caso de uso.

Editar grupo	
Actores	Trainer
Resumen	El usuario edita información acerca del grupo.
Precondición	El grupo debe de estar creado.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clicka</i> en el enlace del menú referente a la administración, "Administración".	2. El sistema muestra una serie de paneles de administración.
3. El usuario <i>clicka</i> en la pestaña referente a grupos.	4. El sistema muestra el apartado de grupos.
5. El usuario <i>clicka</i> en el grupo que quiera cambiar su información.	
6. El usuario edita la información referente al grupo seleccionado.	7. El sistema valida la información.
	8. El sistema modifica la información referente al grupo.
	9. El sistema muestra el grupo con la nueva información.

Tabla 3.2: Caso de uso: Editar grupo.

- **Visualizar grupo:** El usuario puede visualizar la información de aquellos grupos manuales que estén creados. En la tabla 3.3 se puede observar la definición más detallada del caso de uso.

Visualizar grupo	
Actores	Trainer
Resumen	El usuario visualiza información acerca del grupo.
Precondición	El grupo debe de estar creado.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clicka</i> en el enlace del menú referente a la administración, "Administración".	2. El sistema muestra una serie de paneles de administración.
3. El usuario <i>clicka</i> en la pestaña referente a grupos.	4. El sistema muestra el apartado de grupos.
5. El usuario <i>clicka</i> en el grupo que quiera visualizar su información.	
	6. El sistema muestra el grupo su información.

Tabla 3.3: Caso de uso: Visualizar grupo.

- **Eliminar grupo:** El usuario puede eliminar los grupos manuales. En la tabla 3.4 se puede observar la definición más detallada del caso de uso.

Eliminar grupo	
Actores	Trainer
Resumen	El usuario elimina un grupo.
Precondición	El grupo debe de estar creado.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clicka</i> en el enlace del menú referente a la administración, "Administración".	2. El sistema muestra una serie de paneles de administración.
3. El usuario <i>clicka</i> en la pestaña referente a grupos.	
	4. El sistema muestra el apartado de grupos.
5. El usuario <i>clicka</i> en el grupo que quiera eliminar.	
	6. El sistema muestra el plan de capacitación seleccionado.
7. El usuario <i>clicka</i> en el botón "Eliminar grupo"	8. El sistema muestra un dialogo de confirmación.
9. El usuario <i>clicka</i> en el botón "Aceptar"	10. El sistema elimina el grupo de la base de datos.
	11. El sistema muestra una alerta sobre la eliminación satisfactoria del grupo.

Tabla 3.4: Caso de uso: Eliminar grupo.

- **Crear plan de capacitación:** El usuario puede crear planes de capacitación. En la tabla 3.5 se puede observar la definición más detallada del caso de uso.

Crear plan de capacitación	
Actores	Trainer
Resumen	El usuario crea un plan de capacitación.
Precondición	El usuario debe de estar identificado y deben de existir grupos y temas creados.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clicka</i> en el enlace del menú referente a la administración, "Administración".	2. El sistema muestra una serie de paneles de administración
3. El usuario <i>clicka</i> en la pestaña referente a planes de capacitación.	4. El sistema muestra el apartado de planes de capacitación.
5. El usuario <i>clicka</i> en botón "Crear plan de capacitación".	
6. El usuario rellena la información solicitada, eligiendo para ello también los grupos y temas que requiere el plan.	
	7. El sistema valida la información
	8. El sistema da de alta el plan de capacitación.
	9. El sistema muestra en pantalla el nuevo plan de capacitación creado.

Tabla 3.5: Caso de uso: Crear plan de capacitación.

- **Editar plan de capacitación:** El usuario puede editar la información de los planes de capacitación que seleccione. En la tabla 3.6 se puede observar la definición más detallada del caso de uso.

Editar plan de capacitación	
Actores	Trainer
Resumen	El usuario edita información acerca del plan de capacitación.
Precondición	El usuario debe de estar identificado y el plan de capacitación debe de estar creado.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clicka</i> en el enlace del menú referente a la administración, "Administración".	2. El sistema muestra una serie de paneles de administración.
3. El usuario <i>clicka</i> en la pestaña referente a grupos.	
	4. El sistema muestra el apartado de grupos.
5. El usuario <i>clicka</i> en el plan de capacitación que quiera cambiar su información.	
6. El usuario edita la información referente al plan de capacitación seleccionado.	7. El sistema valida la información.
	8. El sistema modifica la información referente al plan de capacitación.
	9. El sistema muestra el plan de capacitación con la nueva información.

Tabla 3.6: Caso de uso: Editar plan de capacitación.

- **Visualizar plan de capacitación:** El usuario puede visualizar la información de aquellos planes de capacitación que estén creados. En la tabla 3.7 se puede observar la definición más detallada del caso de uso.

Visualizar plan de capacitación	
Actores	Trainer
Resumen	El usuario visualiza información acerca del plan de capacitación.
Precondición	El usuario debe de estar identificado y el plan de capacitación debe de estar creado.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clicka</i> en el enlace del menú referente a la administración, "Administración".	2. El sistema muestra una serie de paneles de administración.
3. El usuario <i>clicka</i> en la pestaña referente a plan de capacitación.	4. El sistema muestra el apartado de plan de capacitación.
5. El usuario <i>clicka</i> en el plan de capacitación que quiera visualizar su información.	
	6. El sistema muestra el plan de capacitación su información.

Tabla 3.7: Caso de uso: Visualizar plan de capacitación.

- **Eliminar plan de capacitación:** El usuario puede eliminar los grupos manuales. En la tabla 3.8 se puede observar la definición más detallada del caso de uso.

Eliminar plan de capacitación	
Actores	Trainer
Resumen	El usuario elimina un plan de capacitación.
Precondición	El usuario debe de estar identificado y el plan de capacitación debe de estar creado.
Postcondición	
Flujo de eventos	
Actor	Sistema
1. El usuario <i>clicka</i> en el enlace del menú referente a la administración, "Administración".	2. El sistema muestra una serie de paneles de administración.
3. El usuario <i>clicka</i> en la pestaña referente a planes de capacitación.	
	4. El sistema muestra el apartado de planes de capacitación.
5. El usuario <i>clicka</i> en el plan de capacitación que quiere eliminar.	
5. El usuario <i>clicka</i> en "Eliminar plan".	
	6. El sistema muestra el plan de capacitación su información.
7. El usuario <i>clicka</i> en el botón "Eliminar plan"	8. El sistema muestra un dialogo de confirmación.
9. El usuario <i>clicka</i> en el botón "Aceptar"	10. El sistema elimina el plan de capacitación de la base de datos.
	11. El sistema muestra una alerta sobre la eliminación satisfactoria del plan de capacitación.

Tabla 3.8: Caso de uso: Eliminar plan de capacitación.

- **Crear tema:** El usuario puede crear planes de capacitación. En la tabla 3.9 se puede observar la definición más detallada del caso de uso.

Crear tema	
Actores	Trainer
Resumen	El usuario crea un tema.
Precondición	El usuario debe de estar identificado y deben de existir preguntas creadas.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clicka</i> en el enlace del menú referente a la administración, "Administración".	2. El sistema muestra una serie de paneles de administración
3. El usuario <i>clicka</i> en la pestaña referente a los temas.	4. El sistema muestra el apartado de temas.
5. El usuario <i>clicka</i> en botón "Crear tema".	
6. El usuario rellena la información solicitada, eligiendo para ello las preguntas que quiere añadir.	
	7. El sistema valida la información
	8. El sistema da de alta el tema.
	9. El sistema muestra en pantalla el nuevo tema creado.

Tabla 3.9: Caso de uso: Crear tema.

- **Editar tema:** El usuario puede editar la información de los temas. En la tabla 3.10 se puede observar la definición más detallada del caso de uso.

Editar tema	
Actores	Trainer
Resumen	El usuario edita información acerca del tema.
Precondición	El usuario debe de estar identificado y el tema debe de estar creado.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clica</i> en el enlace del menú referente a la administración, "Administración".	2. El sistema muestra una serie de paneles de administración.
3. El usuario <i>clica</i> en la pestaña referente a temas.	
	4. El sistema muestra el apartado de temas.
5. El usuario <i>clica</i> en el tema que quiera cambiar su información.	
6. El usuario edita la información referente al tema seleccionado.	7. El sistema valida la información.
	8. El sistema modifica la información referente al tema.
	9. El sistema muestra el tema con la nueva información.

Tabla 3.10: Caso de uso: Editar tema.

- **Visualizar tema:** El usuario puede visualizar la información de aquellos temas que estén creados. En la tabla 3.11 se puede observar la definición más detallada del caso de uso.

Visualizar temas	
Actores	Trainer
Resumen	El usuario visualiza información acerca tema.
Precondición	El usuario debe de estar identificado y el tema debe de estar creado.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clica</i> en el enlace del menú referente a la administración, "Administración".	2. El sistema muestra una serie de paneles de administración.
3. El usuario <i>clica</i> en la pestaña referente al tema.	4. El sistema muestra el apartado de tema.
5. El usuario <i>clica</i> en el tema que quiera visualizar su información.	
	6. El sistema muestra el tema su información.

Tabla 3.11: Caso de uso: Visualizar tema.

- **Eliminar tema:** El usuario puede eliminar los grupos manuales. En la tabla 3.12 se puede observar la definición más detallada del caso de uso.

Eliminar tema	
Actores	Trainer
Resumen	El usuario elimina un tema.
Precondición	El usuario debe de estar identificado y el tema debe de estar creado.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clicka</i> en el enlace del menú referente a la administración, "Administración".	2. El sistema muestra una serie de paneles de administración.
3. El usuario <i>clicka</i> en la pestaña referente a temas.	
	4. El sistema muestra el apartado de temas.
5. El usuario <i>clicka</i> en el tema que quiere eliminar.	
5. El usuario <i>clicka</i> en "Eliminar tema".	
	6. El sistema muestra la información del tema.
7. El usuario <i>clicka</i> en el botón "Eliminar tema"	8. El sistema muestra un dialogo de confirmación.
9. El usuario <i>clicka</i> en el botón "Aceptar"	10. El sistema elimina el tema de la base de datos.
	11. El sistema muestra una alerta sobre la eliminación satisfactoria del tema.

Tabla 3.12: Caso de uso: Eliminar tema.

- **Añadir pregunta:** El usuario puede crear preguntas. En la tabla 3.13 se puede observar la definición más detallada del caso de uso.

Añadir pregunta	
Actores	Trainer
Resumen	El usuario crea un pregunta.
Precondición	El usuario debe de estar identificado y deben de estar relacionado con un tema en concreto.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clica</i> en el enlace del menú referente a la administración, "Administración".	2. El sistema muestra una serie de paneles de administración
3. El usuario <i>clica</i> en la pestaña referente a los temas.	4. El sistema muestra el apartado de temas.
5. El usuario <i>clica</i> en botón "Crear pregunta".	
6. El usuario rellena la información solicitada, eligiendo para ello las preguntas y las opciones que quiere añadir.	
	7. El sistema muestra por pantalla las preguntas añadidas al tema
	8. El sistema da de alta la pregunta.
	9. El sistema muestra en pantalla el nuevo pregunta creado.

Tabla 3.13: Caso de uso: Añadir pregunta.

- **Editar pregunta:** El usuario puede editar la información de las preguntas. En la tabla 3.14 se puede observar la definición más detallada del caso de uso.

Editar pregunta	
Actores	Trainer
Resumen	El usuario edita información acerca del pregunta.
Precondición	El usuario debe de estar identificado y el pregunta debe de estar creado.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clicka</i> en el enlace del menú referente a la administración, "Administración".	2. El sistema muestra una serie de paneles de administración.
3. El usuario <i>clicka</i> en la pestaña referente a los temas.	
	4. El sistema muestra el apartado de temas.
5. El usuario <i>clicka</i> en el tema y la pregunta que quiera cambiar su información.	
6. El usuario edita la información referente a la pregunta seleccionada y <i>clicka</i> en el botón de guardo del tema.	7. El sistema valida la información.
	8. El sistema modifica la información referente a la pregunta.
	9. El sistema muestra la pregunta con la nueva información.

Tabla 3.14: Caso de uso: Editar pregunta.

- Visualizar preguntas de un tema:** El usuario puede visualizar la información de aquellas preguntas que estén creadas en un tema. En la tabla 3.15 se puede observar la definición más detallada del caso de uso.

Visualizar preguntas	
Actores	Trainer
Resumen	El usuario visualiza información acerca de la pregunta que ha seleccionado.
Precondición	El usuario debe de estar identificado y la pregunta debe de estar creada.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clicka</i> en el enlace del menú referente a la administración, "Administración".	2. El sistema muestra una serie de paneles de administración.
3. El usuario <i>clicka</i> en la pestaña referente a los temas.	4. El sistema muestra el apartado de temas.
5. El usuario <i>clicka</i> en el tema donde quiera visualizar la información de las pregunta.	
	6. El sistema muestra la información de la pregunta.

Tabla 3.15: Caso de uso: Visualizar pregunta.

- **Eliminar pregunta:** El usuario puede eliminar los grupos manuales. En la tabla 3.16 se puede observar la definición más detallada del caso de uso.

Eliminar pregunta	
Actores	Trainer
Resumen	El usuario elimina una pregunta.
Precondición	El usuario debe de estar identificado y la pregunta debe de estar creada.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clicka</i> en el enlace del menú referente a la administración, "Administración".	2. El sistema muestra una serie de paneles de administración.
3. El usuario <i>clicka</i> en la pestaña referente al tema.	
	4. El sistema muestra el apartado de temas.
5. El usuario <i>clicka</i> en la pregunta que quiere eliminar.	
5. El usuario <i>clicka</i> en "Eliminar pregunta".	6. El sistema muestra un dialogo de confirmación.
7. El usuario <i>clicka</i> en el botón "Aceptar"	8. El sistema elimina la pregunta de la base de datos, relacionado con el tema que se ha elegido.
	9. El sistema muestra una alerta sobre la eliminación satisfactoria de la pregunta.

Tabla 3.16: Caso de uso: Eliminar pregunta.

- **Realizar pregunta:** El usuario puede realizar la pregunta asignada a su grupo. En la tabla 3.17 se puede observar la definición más detallada del caso de uso.

Realizar pregunta	
Actores	Learner
Resumen	El usuario realiza una pregunta.
Precondición	El usuario debe de estar identificado, la pregunta debe de estar creada y asignada al usuario
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clicka</i> en la notificación que le ha aparecido en el dispositivo móvil.	2. El sistema muestra la pantalla de la pregunta.
3. El usuario elige la opción que crea conveniente.	
	4. El sistema guarda la información.
	5. El sistema muestra información sobre la opción correcta

Tabla 3.17: Caso de uso: Realizar pregunta.

3.4 Arquitectura

En el presente apartado se va proceder a definir la arquitectura que se va a seguir en el proceso de desarrollo de la aplicación ProSafe, con el objetivo de diseñar una base sólida para el proyecto. En la figura 3.9 se puede observar el diagrama que define la arquitectura de ProSafe. A continuación se explicará brevemente el funcionamiento de esta.

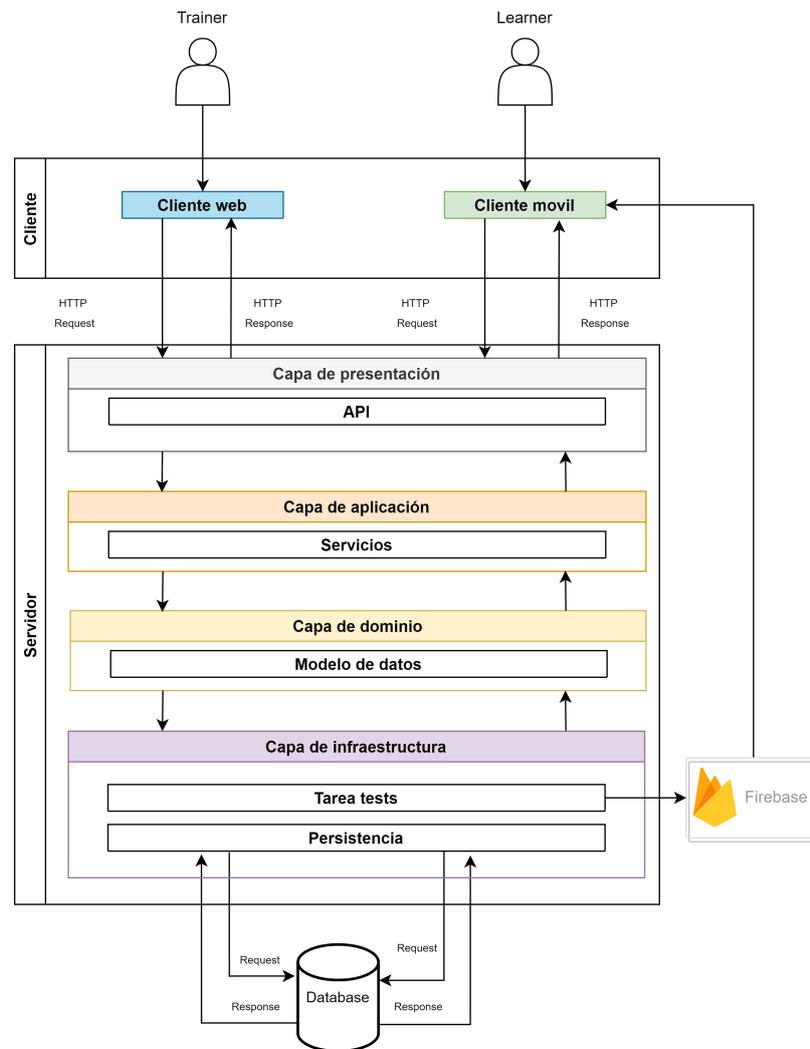


Figura 3.9: Arquitectura ProSafe

Como ya ha sido especificado anteriormente, la aplicación ProSafe cuenta con dos tipos de clientes: un cliente web, destinado a la gestión por parte del responsable en PRL y un cliente móvil, destinado a cualquier empleado de la organización.

Tanto el cliente web como el cliente móvil, se conectarán a través de peticiones HTTP a la capa de presentación, por medio de la API que ha proporcionado el servidor. La capa de presentación a su vez transformará esas peticiones y creará la representación de la acción que comunicará a la capa de aplicación, encargada de ejecutar las acciones y servicios que han sido solicitados. La capa de aplicación se vale de la capa de dominio, que define el modelo de datos utilizado en el negocio, que a su vez necesita persistir los datos, con lo que se comunica con la capa de infraestructura y esta con la base de datos definida. Además, en la capa de infraestructura se añade una tarea para gestionar el envío

de las notificaciones de la aplicación, esta función se conecta con Firebase, que enviará las notificaciones pertinentes que se hayan definido.

Por otro lado, y una vez que las acciones se hayan realizado en la capa de aplicación, se comunicará con la capa de presentación, encargada de *serializar* la respuesta para poder comunicar la respuesta con los clientes que solicitaron la petición.

CAPÍTULO 4

Diseño de la solución

En el presente capítulo se detalla el diseño que se ha especificado para la solución, desde el diseño de la API¹, hasta el de las interfaces tanto de la aplicación web como móvil.

4.1 Diseño de API

En el presente apartado se van a especificar los diferentes *endpoints* que se han diseñado para la API, donde se especifican los métodos HTTP² utilizados, la descripción y la especificación de parámetros (si cuenta con ellos).

Employees

El recurso *Employees* hace referencia a los empleados de la organización donde se haya instanciado la aplicación. En las tablas 4.1 y 4.3 se especifican los *endpoints* con los que cuenta la colección *employees*.

Endpoint	Método	Descripción
/employees	GET	Devuelve todos los empleados de la organización .

Tabla 4.1: Tabla endpoint "/people/employees"

En la tabla 4.2 se muestran los parámetros con los que cuenta el *endpoint* "/people/employees".

Parámetro	Tipo	Requerido?	Descripción
?limit	int	opcional	Limita el número máximo de entradas a retornar por la API.
?offset	int	opcional	Limita el número máximo de entradas a retornar por la API.

¹API: Interfaz de programación de aplicación

²HTTP: Protocolo de transferencia de hipertexto (Hypertext Transfer Protocol) que permite la transferencia de archivos en la web.

?group	string	opcional	Limita las entradas devueltas, filtradas por el nombre del grupo.
--------	--------	----------	---

Tabla 4.2: Tabla de los endpoint del recurso employees

Endpoint	Método	Descripción
/people/employees/{id}	GET	Devuelve un empleado de la organización identificado por un id .

Tabla 4.3: Tabla endpoint "/people/employees/id"

En la tabla 4.4 se muestran los parámetros con los que cuenta el *endpoint* "/people/employees/id".

Parámetro	Tipo	Requerido?	Descripción
:id	int	obligatorio	Identifica a un empleado de la organización.

Tabla 4.4: Tabla de los parámetros del endpoint: "/people/employees/id"

Departments

El recurso *Departments* hace referencia a los departamentos con los que cuenta la organización, estos servirán para la creación de grupos según criterios. En las tablas 4.5 y 4.7 se especifican los *endpoints* con los que cuenta la colección *departments*.

Endpoint	Método	Descripción
/people/departments	GET	Devuelve todos los departamentos de la organización .

Tabla 4.5: Tabla endpoint "/people/departments"

En la tabla 4.6 se muestran los parámetros con los que cuenta el *endpoint* "/people/departments".

Parámetro	Tipo	Requerido?	Descripción
?limit	int	opcional	Limita el número máximo de entradas a retornar por la API.
?offset	int	opcional	Limita el número máximo de entradas a retornar por la API.
?group	string	opcional	Limita las entradas devueltas, filtradas por el nombre del grupo.

Tabla 4.6: Tabla de los parámetros de la colección departments

Endpoint	Método	Descripción
/people/departments/{id}	GET	Devuelve un departamento de la organización identificado por un id.

Tabla 4.7: Tabla endpoint "/people/departments/id"

En la tabla 4.8 se muestran los parámetros con los que cuenta el *endpoint* "/people/departments/id".

Parámetro	Tipo	Requerido?	Descripción
:id	int	obligatorio	Identifica a un departamento de la organización.

Tabla 4.8: Tabla de los parámetros del endpoint: "/people/departments/id"

Groups

El recurso *Groups* hace referencia a los grupos creados, tanto de forma manual, como a través de criterios, de la organización donde se haya instanciado la aplicación.

En las tablas 4.9 y 4.11 se especifican los *endpoints* con los que cuenta la colección *groups*.

Endpoint	Método	Descripción
/training/groups	GET	Devuelve todos los grupos creados de la organización .
/training/groups	POST	Crea un grupo de empleados.

Tabla 4.9: Tabla endpoint "/training/groups"

En la tabla 4.10 se muestran los parámetros con los que cuenta el *endpoint* "/training/groups".

Parámetro	Tipo	Requerido?	Descripción
?limit	int	opcional	Limita el número máximo de entradas a retornar por la API.
?offset	int	opcional	Limita el número máximo de entradas a retornar por la API.
?embeds	boolean	opcional	Solicita la información completa de los empleados en la petición.

Tabla 4.10: Tabla de los parámetros del recurso groups

Endpoint	Método	Descripción
/training/groups/{id}	GET	Devuelve un empleado de la organización identificado por un <i>id</i> .

/training/groups/{id}	PUT	Actualiza el grupo identificado por el <i>id</i> .
/training/groups/{id}	PATCH	Actualiza de forma parcial el grupo identificado por el <i>id</i> .
/training/groups/{id}	DELETE	Elimina el grupo identificado por el <i>id</i> .

Tabla 4.11: Tabla endpoint "/training/groups/id"

En la tabla 4.12 se muestran los parámetros con los que cuenta el endpoint "/training/groups/id".

Parámetro	Tipo	Requerido?	Descripción
:id	int	obligatorio	Identifica a un empleado de la organización.

Tabla 4.12: Tabla de los parámetros del endpoint: "/training/groups/id"

Plans

El recurso *Plans* hace referencia a planes de capacitación creados en la organización donde se haya instanciado la aplicación.

En las tablas 4.13 y 4.15 se especifican los endpoints con los que cuenta la colección *plans*.

Endpoint	Método	Descripción
/training/plans	GET	Devuelve todos los planes de capacitación creados de la organización .
/training/plans	POST	Crea un plan de capacitación.

Tabla 4.13: Tabla endpoint "/training/plans"

En la tabla 4.14 se muestran los parámetros con los que cuenta el endpoint "/training/plans".

Parámetro	Tipo	Requerido?	Descripción
?limit	int	opcional	Limita el número máximo de entradas a retornar por la API.
?offset	int	opcional	Limita el número máximo de entradas a retornar por la API.
?embeds	boolean	opcional	Solicita la información completa de los grupos y temas en la petición.

Tabla 4.14: Tabla de los parámetros del recurso plans

Endpoint	Método	Descripción
/training/plans/{id}	GET	Devuelve un plan de capacitación de la organización identificado por un <i>id</i> .
/training/plans/{id}	PUT	Actualiza el plan de capacitación identificado por el <i>id</i> .
/training/plans/{id}	PATCH	Actualiza de forma parcial el plan de capacitación identificado por el <i>id</i> .
/training/plans/{id}	DELETE	Elimina el plan de capacitación identificado por el <i>id</i> .

Tabla 4.15: Tabla endpoint "/training/plans/id"

En la tabla 4.16 se muestran los parámetros con los que cuenta el endpoint "/training/plans/id".

Parámetro	Tipo	Requerido?	Descripción
:id	int	obligatorio	Identifica a un empleado de la organización.

Tabla 4.16: Tabla de los parámetros del endpoint: "/training/plans/id"

Topics

El recurso *Topics* hace referencia a los temas creados en la organización dónde se haya instanciado la aplicación.

En las tablas 4.17 y 4.19 se especifican los endpoints con los que cuenta la colección *topics*.

Endpoint	Método	Descripción
/training/topics	GET	Devuelve todos los temas creados de la organización .
/training/topics	POST	Crea un tema.

Tabla 4.17: Tabla endpoint "/training/topics"

En la tabla 4.18 se muestran los parámetros con los que cuenta el endpoint "/training/topics".

Parámetro	Tipo	Requerido?	Descripción
?limit	int	opcional	Limita el número máximo de entradas a retornar por la API.
?offset	int	opcional	Limita el número máximo de entradas a retornar por la API.

Tabla 4.18: Tabla de los parámetros del recurso plans

Endpoint	Método	Descripción
/training/topics/{id}	GET	Devuelve un tema de la organización identificado por un <i>id</i> .
/training/topics/{id}	PUT	Actualiza el tema identificado por el <i>id</i> .
/training/topics/{id}	PATCH	Actualiza de forma parcial el tema identificado por el <i>id</i> .
/training//topics/{id}	DELETE	Elimina el tema identificado por el <i>id</i> .

Tabla 4.19: Tabla endpoint "/training/topics/id"

En la tabla 4.20 se muestran los parámetros con los que cuenta el endpoint "/training/topics/id".

Parámetro	Tipo	Requerido?	Descripción
:id	int	obligatorio	Identifica a un empleado de la organización.

Tabla 4.20: Tabla de los parámetros del endpoint: "/training/topics/id"

Stats

El recurso *Stats* hace referencia a las estadísticas creadas en la organización dónde se haya instanciado la aplicación.

En las tablas 4.24, 4.22 y 4.23 se especifican los endpoints con los que cuenta la colección *stats*.

Endpoint	Método	Descripción
/stats/groups	GET	Devuelve las estadísticas referentes a los grupos (preguntas incorrectas y correctas por grupo, número de empleados por grupos y número de puntos ganado por grupos).
/stats/topic	GET	Devuelve la estadística referente a los temas (preguntas incorrectas y correctas por tema).

Tabla 4.21: Tabla endpoint "/stats/"

Endpoint	Método	Descripción
/stats/employees/count	GET	Devuelve el número total de empleados de la organización.
/stats/plans/count	GET	Devuelve el número total de planes de capacitación de la organización.

/stats/topics/count	GET	Devuelve el número total de temas de la organización.
/stats/groups/count	GET	Devuelve el número total de grupos de la organización.
/stats/tests/count	GET	Devuelve el número total de tests de la organización.

Tabla 4.22: Tabla endpoint "/stats/:collection/count"

Endpoint	Método	Descripción
/stats/employees/score	GET	Devuelve el número total de puntos ganados por los empleados en la organización.

Tabla 4.23: Tabla endpoint "/stats/employees/score"

Tests

El recurso *Tests* hace referencia a los tests creados y respondidos en la organización dónde se haya instanciado la aplicación. Estos tests contienen la información referente a las preguntas, pero también la respuesta de los usuarios.

En la tabla se especifica el endpoint con el que cuenta la colección *tests*.

Endpoint	Método	Descripción
/training/tests/:id/answer	POST	Actualiza el test con la información referente a la respuesta del empleado.

Tabla 4.24: Tabla endpoint "/training/tests/id/answer"

En la tabla 4.25 se muestran los parámetros con los que cuenta el endpoint "/training/tests/id/answer".

Parámetro	Tipo	Requerido?	Descripción
:id	int	obligatorio	Identifica una pregunta de la organización.

Tabla 4.25: Tabla de los parámetros del endpoint: "/training/tests/id/answer"

Posts

El recurso *Posts* hace referencia a los recursos sobre PRL almacenados en la organización. En las tablas 4.26 y 4.28 se especifican los endpoints con los que cuenta la colección *posts*.

Endpoint	Método	Descripción
/training/posts	GET	Devuelve todos las publicaciones creadas de la organización.

Tabla 4.26: Tabla endpoint `"/training/posts"`

En la tabla 4.27 se muestran los parámetros con los que cuenta el endpoint `"/training/posts"`.

Parámetro	Tipo	Requerido?	Descripción
?limit	int	opcional	Limita el número máximo de entradas a retornar por la API.
?offset	int	opcional	Limita el número máximo de entradas a retornar por la API.

Tabla 4.27: Tabla de los parámetros del recurso plans

Endpoint	Método	Descripción
/posts/{id}	GET	Devuelve una publicación de la organización identificado por un <i>id</i> .

Tabla 4.28: Tabla endpoint `"/training/posts/id"`

En la tabla 4.29 se muestran los parámetros con los que cuenta el endpoint `"/training/posts/id"`.

Parámetro	Tipo	Requerido?	Descripción
:id	int	obligatorio	Identifica a un empleado de la organización.

Tabla 4.29: Tabla de los parámetros del endpoint: `"/training/posts/id"`

Badges

El recurso *Badges* hace referencia al endpoint utilizado para el inicio de sesión a la aplicación por parte del empleado. En la tabla 4.30 se especifican los endpoints con los que cuenta la colección *posts*.

Endpoint	Método	Descripción	Códigos error
/security/badge/:auth	GET	Devuelve la autorización del usuario para poder acceder a la aplicación. Se añadirá en la cabecera "Authorization" [9] y en el parámetro :auth las credenciales del usuario.	<ul style="list-style-type: none"> ■ 201 OK ■ 401 Unauthorized

Tabla 4.30: Tabla endpoint `"/security/badge"`

4.2 Diseño de interfaces

Para la realización de un producto software se requiere que se invierta tiempo en diseñar una interfaz que sea clara, eficiente, flexible, coherente y simple. En la presente sección se expondrán los *wireframes*³ finales diseñados para la herramienta ProSafe. Cabe destacar que este proceso ha sido prolongado, desenchandose en el camino varias de las ideas propuestas hasta lograr una interfaz que cuente con las características anteriormente citadas.

4.2.1. Portal web

En este apartado se van a exponer las interfaces que se han diseñado para el apartado web de la aplicación. De esta forma, se pueden encontrar los siguientes wireframes:

Pantalla de inicio de sesión: En la figura 4.1 se pueden diferenciar los campos de credenciales respectivos al nombre de usuario y la contraseña. Además, se puede observar también los botones de inicio de sesión y de mantenimiento de las credenciales del usuario.

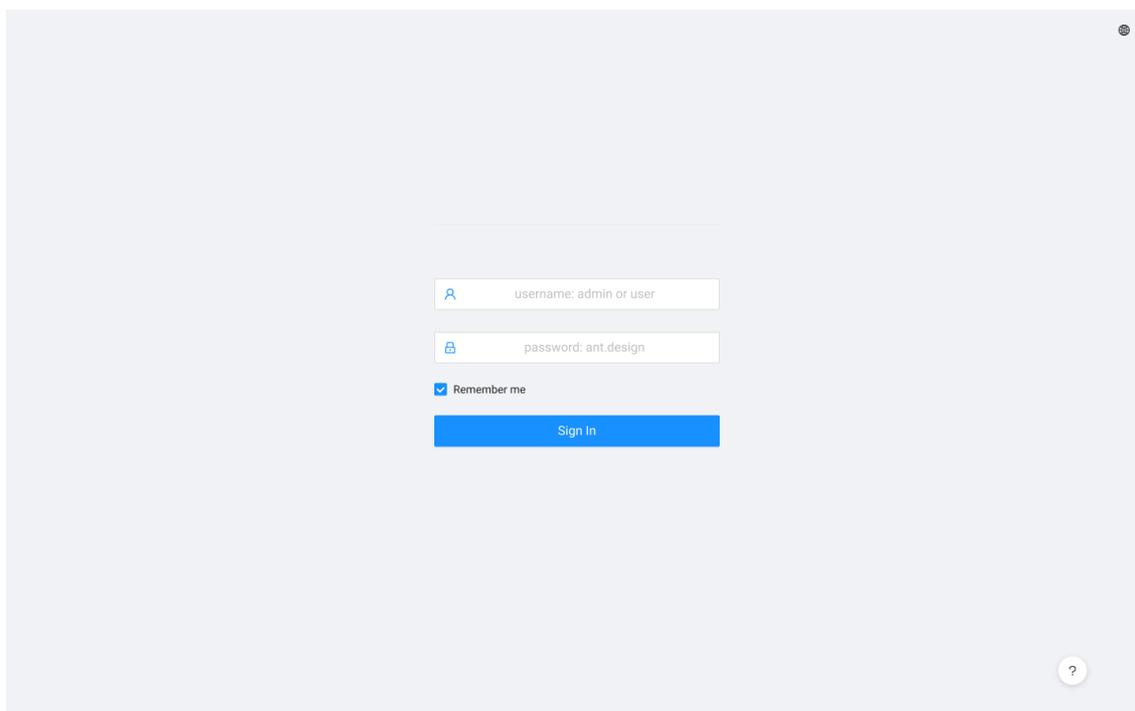


Figura 4.1: Wireframe: Pantalla de inicio de sesión

³Wireframes: Prototipos creados para el diseño de una página web, siendo estos una guía visual que representa el esqueleto esperado para un producto.

Pantalla de inicio: En la figura 4.2 se puede observar la pantalla de inicio que un usuario identificado podrá visualizar. Esta cuenta con una serie de estadísticas globales sobre datos generales de la organización además de contar con una tabla donde se muestra la información más relevante para la aplicación de todos los empleados de la empresa.

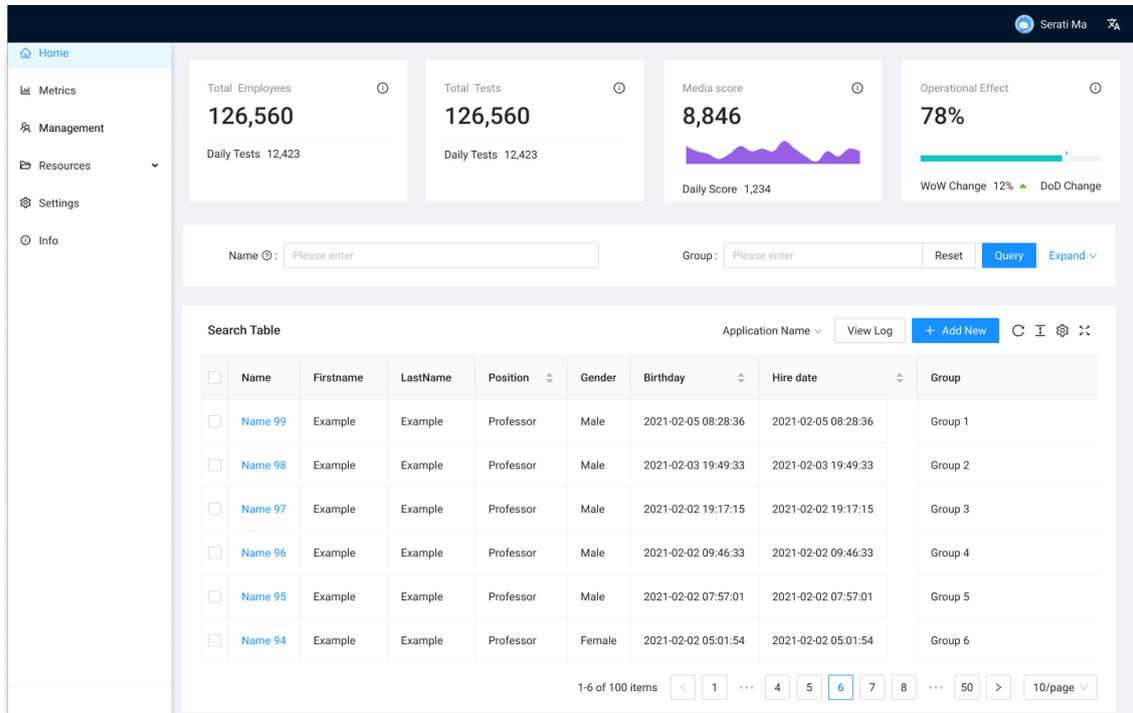


Figura 4.2: Wireframe: Pantalla de inicio

Pantalla de planes de capacitación: En la figura 4.3 se puede observar un menú lateral donde se encuentran todos los planes de capacitación creados en la aplicación, además de un botón para la creación de nuevos planes. Por otro lado, la información relacionada a estos planes se muestra junto a estos, contando con los campos: nombre, frecuencia, temas y grupos asociados. Asimismo, cuenta con un botón de guardado y de eliminación para cada uno de los planes.

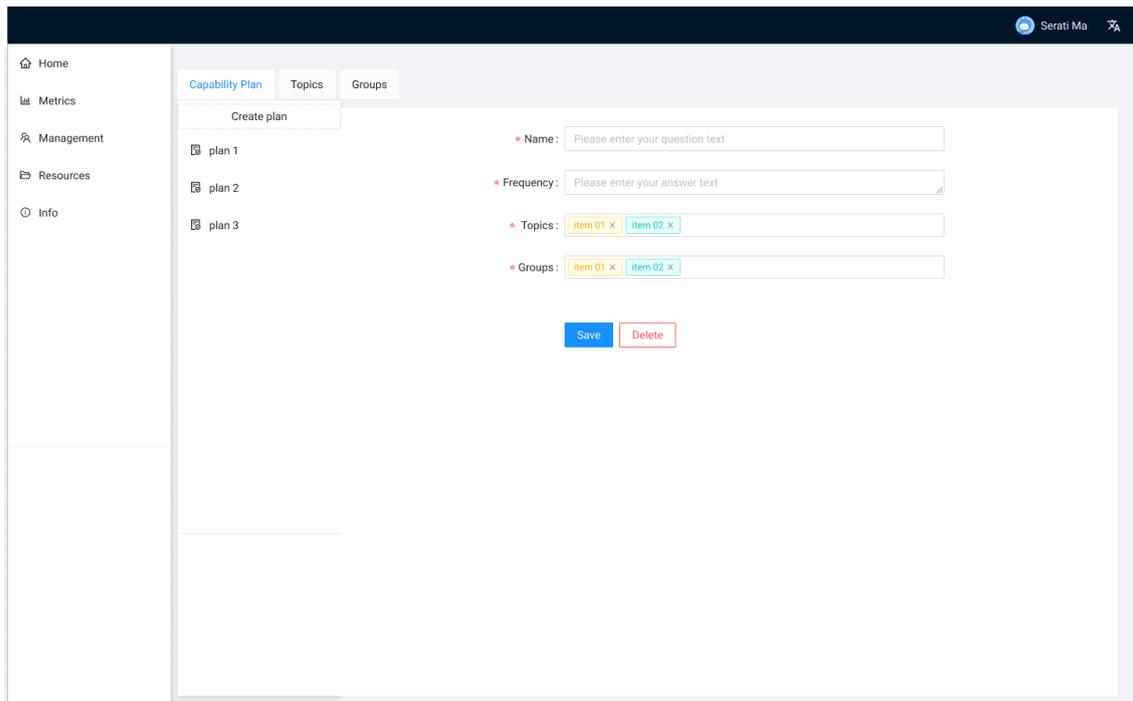


Figura 4.3: Wireframe: Pantalla de gestión de planes de capacitación

Pantalla de temas: En la figura 4.4 se puede observar un menú lateral dónde se encuentran todos los temas creados en la aplicación, además de un botón para la creación de nuevos temas. Por otro lado, la información relacionada a estos temas se muestra junto a estos, contando con los campos: nombre y una lista de cuestiones que pueden ir asociadas, estas tendrán un título de pregunta y una serie de respuestas, pudiendo el usuario solo marcar una como correcta. Asimismo, cuenta un botón de guardado y de eliminación para cada uno de los temas.

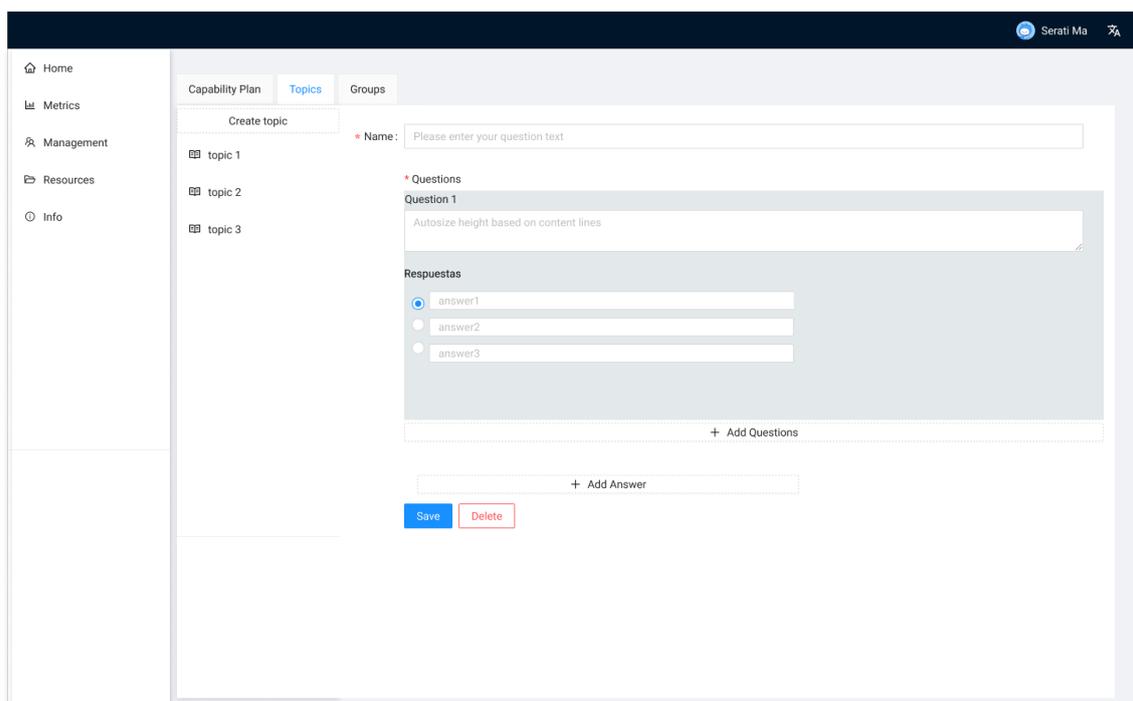


Figura 4.4: Wireframe: Pantalla de gestión de temas

Pantalla de grupos (forma manual): En la figura 4.5 se puede observar un menú lateral donde se encuentran todos los grupos creados en la aplicación, además de un botón para la creación grupos. Por otro lado, la información relacionada a estos grupos se muestra al lado, contando con los campos: nombre, tipo (en este caso está seleccionado el tipo manual), y un selector de los empleados. Asimismo, cuenta un un botón de guardado y de eliminación para cada uno de los grupos.

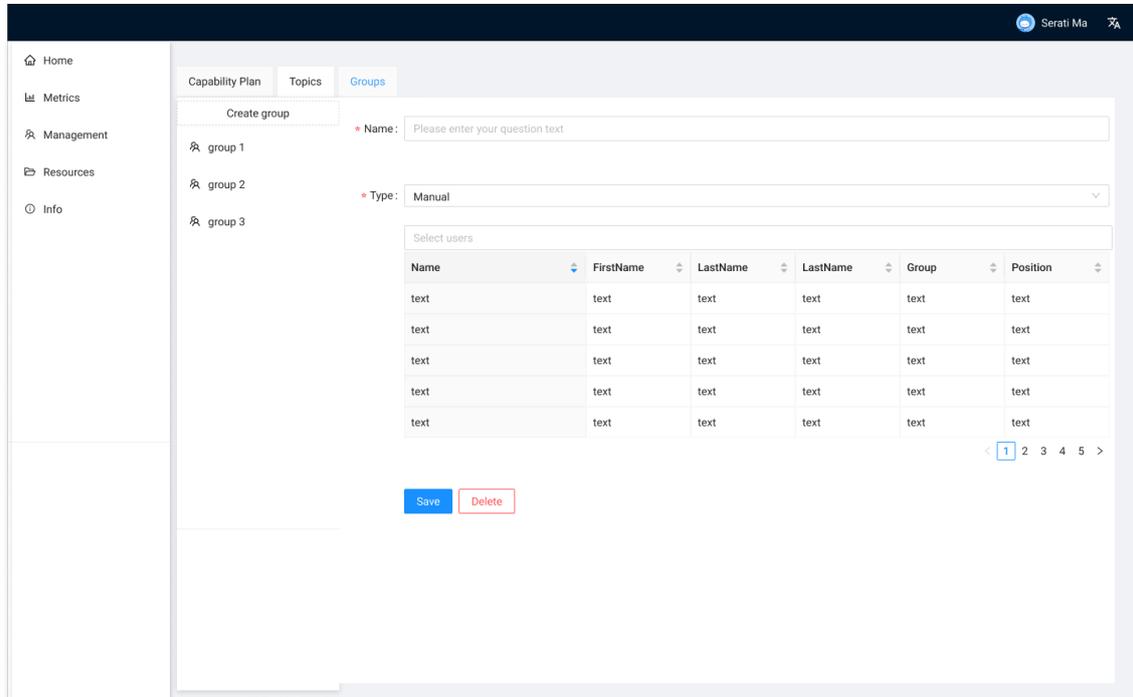


Figura 4.5: Wireframe: Pantalla de grupos de forma manual

Pantalla de grupos (por criterios): En la figura 4.6 se puede observar un menú lateral donde se encuentran todos los grupos creados en la aplicación, además de un botón para la creación grupos. Por otro lado, la información relacionada a estos grupos se muestra junto a estos, contando con los campos: nombre, tipo (en este caso está seleccionado el tipo por criterios), y un selector de los empleados. Asimismo, cuenta un botón de guardado y de eliminación para cada uno de los grupos.

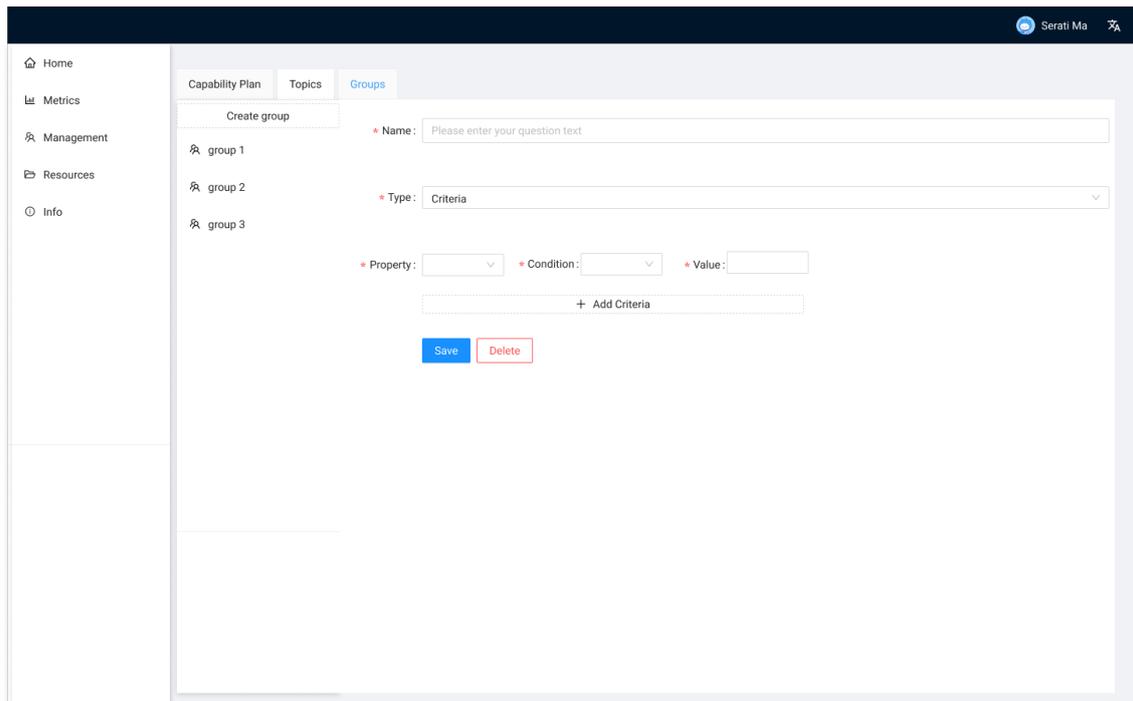


Figura 4.6: Wireframe: Pantalla de grupos por criterios

Pantalla de documentos: En la figura 4.7 se puede observar una serie de recursos sobre PRL con los que cuenta la aplicación.

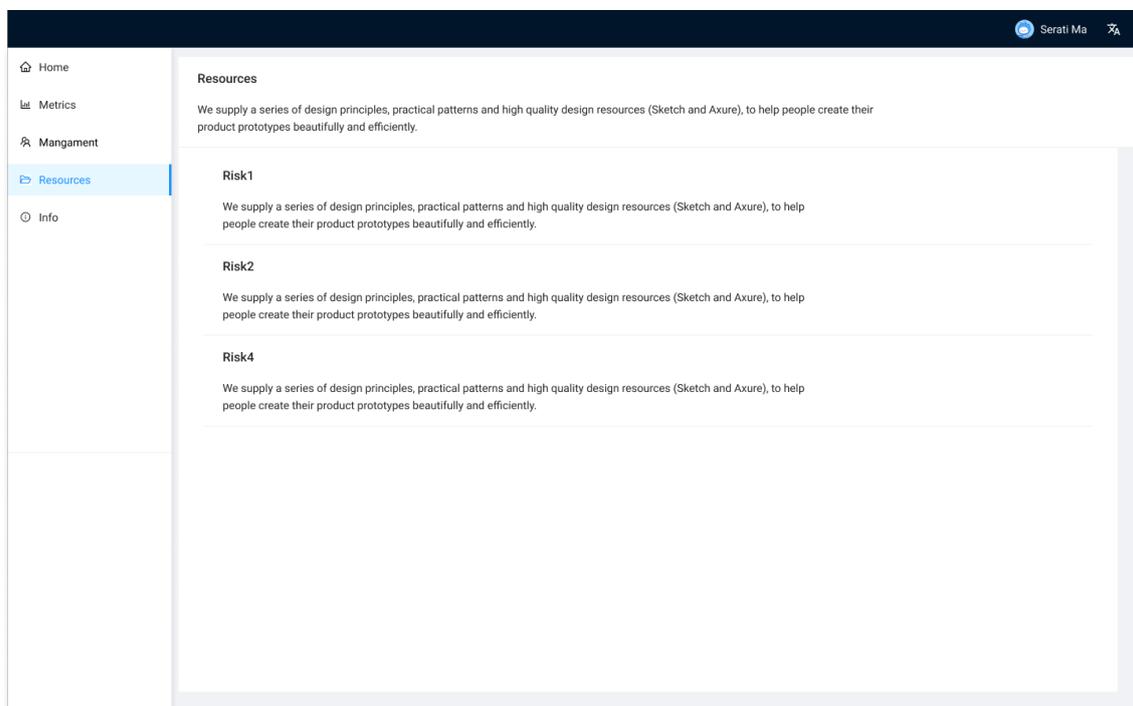


Figura 4.7: Wireframe: Pantalla de documentos

Pantalla de métricas: En la figura 4.8 se pueden observar una serie de gráficos sobre la realización de los tests o información propia del avance de la capacitación de los empleados.

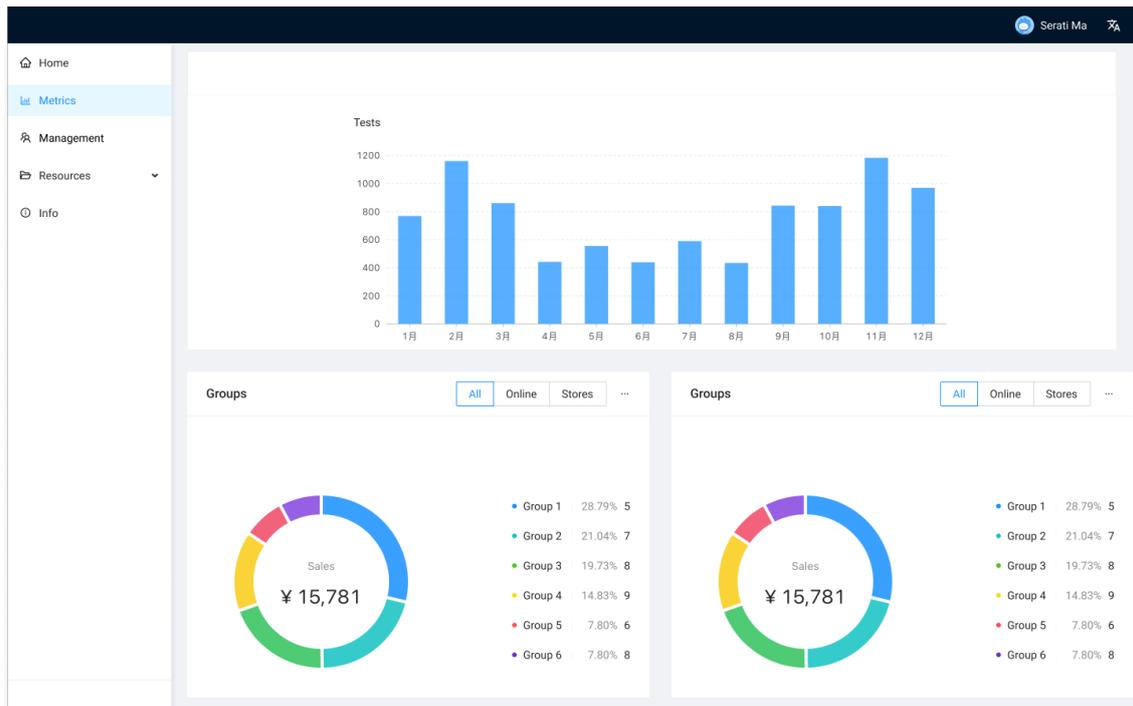


Figura 4.8: Wireframe: Pantalla de métricas

Pantalla de información sobre ProSafe: En la figura 4.6 se puede observar una serie de información sobre la aplicación y su uso.

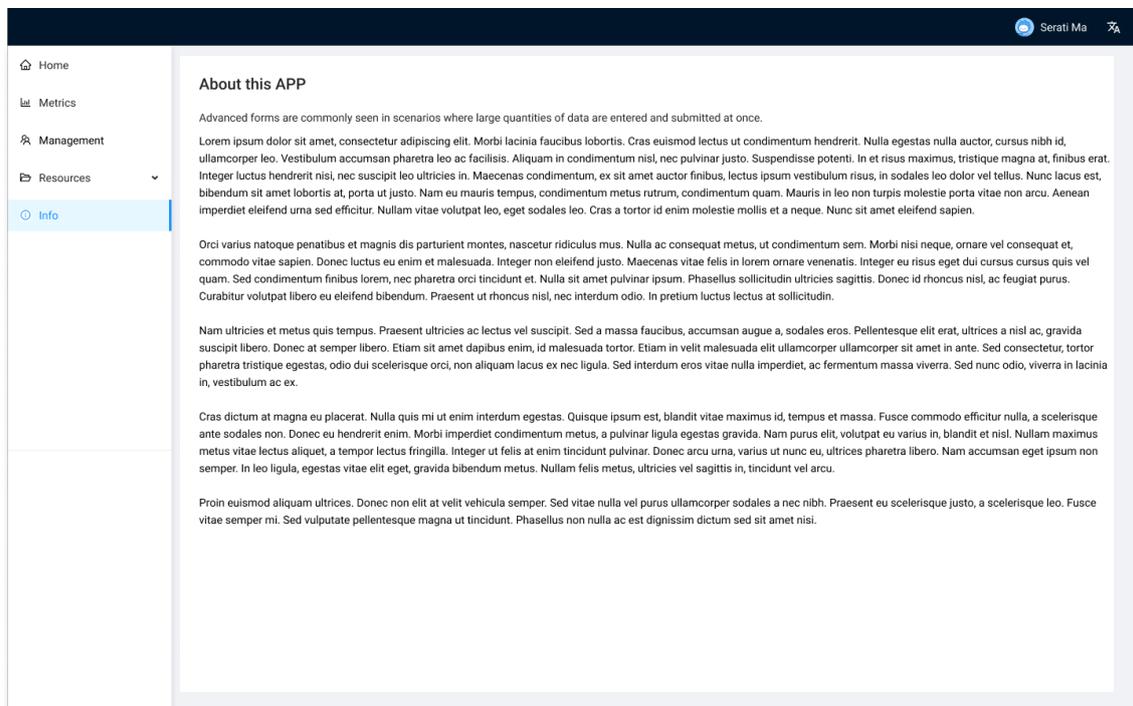


Figura 4.9: Wireframe: Pantalla de información sobre ProSafe

Pantalla de información sobre datos del empleado: En la figura 4.10 se puede observar un desplegable lateral que muestra la información sobre el empleado que ha iniciado sesión.

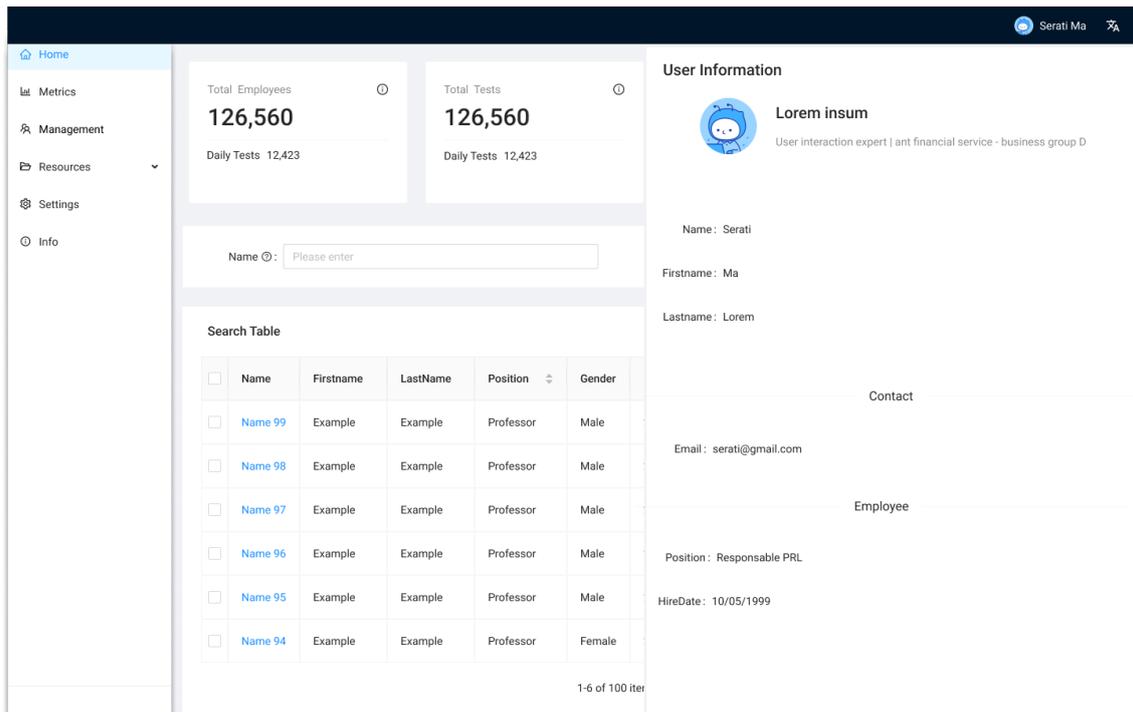


Figura 4.10: Wireframe: Pantalla de información sobre datos del empleado

Diálogo de confirmación: En la figura 4.11 se puede observar un diálogo de confirmación para ser usado en cada intento de eliminación de algún contenido, sea planes de capacitación, temas, grupos etc

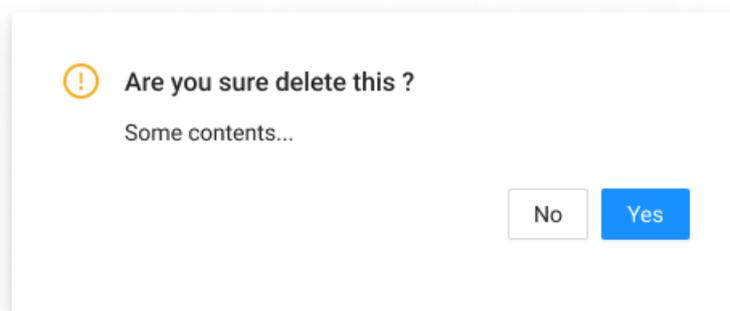


Figura 4.11: Wireframe: Diálogo de confirmación

Diálogo de información exitosa : En la figura 4.12 se puede observar un diálogo de confirmación para ser usado cada vez que el usuario cree algún tipo de contenido de manera correcta.

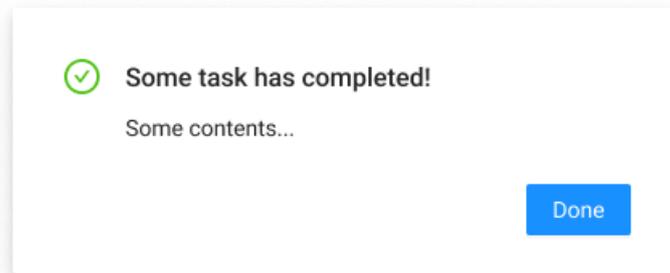


Figura 4.12: Wireframe: Diálogo de información exitosa

Diálogo de información errónea : En la figura 4.13 se puede observar un diálogo de confirmación para ser usado cada vez que el usuario cree algún tipo de contenido de manera incorrecta.

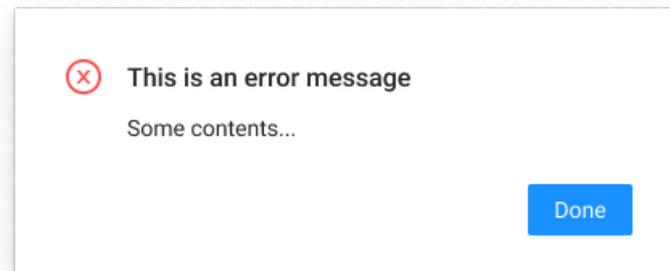


Figura 4.13: Wireframe: Diálogo de información errónea

4.2.2. Aplicación móvil

En este apartado se van a exponer las interfaces que se han diseñado para el apartado móvil de la aplicación. De esta forma, se pueden encontrar los siguientes wireframes:

Pantalla de inicio de sesión: En la figura 4.14 se pueden diferenciar los campos de credenciales respectivos al email del usuario y la contraseña. Además, se puede observar también los botones de inicio de sesión y de mantener las credenciales del usuario logueadas.

ProSafe

Username

Password

Sign in

Keep me logged in [Reset password](#)

© 2022 UPV

Figura 4.14: Wireframe: Pantalla de inicio de sesión (móvil)

Pantalla principal de usuario logueado: En la figura 4.15 se puede observar un mensaje de bienvenida con el nombre del usuario que se ha conectado a la aplicación.

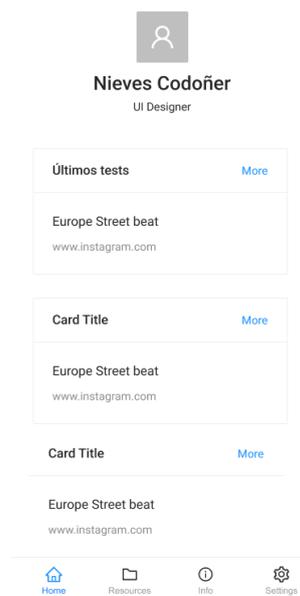


Figura 4.15: Wireframe: Pantalla principal de usuario logueado (móvil)

Pantalla de recursos de PRL: En la figura 4.16 se puede observar una serie de carpetas con información sobre los objetivos de aprendizaje de la aplicación. Además, en la parte inferior se encuentra un menú de navegación para la aplicación.

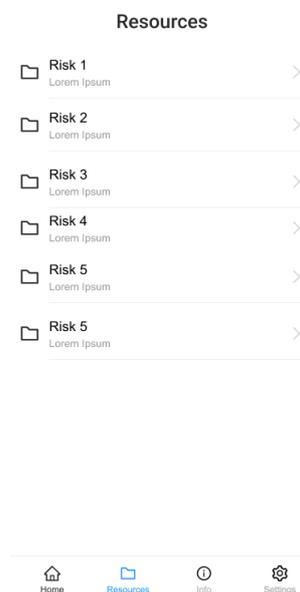


Figura 4.16: Wireframe: Pantalla de recursos PRL (móvil)

Pantalla de información: En la figura 4.17 se puede observar información referente a la aplicación, el desarrollador y una breve guía de uso.

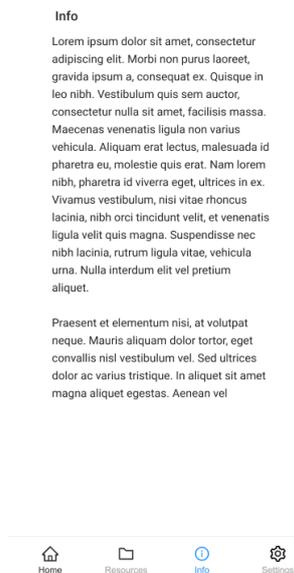


Figura 4.17: Wireframe: Pantalla de información sobre la aplicación (móvil)

Pantalla de perfil de usuario: En la figura 4.17 se puede observar una serie de campos del usuario, estos no podrán ser editables y se contará con un botón para pueda cerrar sesión con su cuenta.

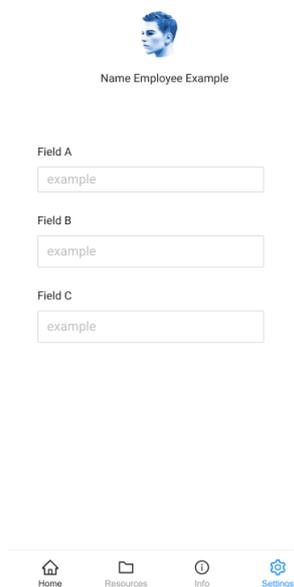


Figura 4.18: Wireframe: Pantalla de perfil de usuario (móvil)

CAPÍTULO 5

Tecnologías utilizadas

En el presente capítulo se van a presentar las diferentes tecnologías y herramientas utilizadas para el desarrollo del proyecto. Además, se van a exponer las alternativas que se han estudiado antes de elegir las herramientas finales de la propuesta. En dicho capítulo se van a detallar las tecnologías utilizadas en la realización del proyecto, como también las razones por las cuales se han elegido en comparación con otras.

5.1 Entorno

La elección de un IDE ¹ para el proceso de desarrollo de un proyecto software es un punto indispensable para facilitar tareas tales como la depuración, refactorización y validación de código entre otros. En este caso, se han elegido dos IDE creados por la compañía JetBrains. Cabe destacar que estos IDE tienen muchas similitudes, pero cada uno está centrado y se especializa en un lenguaje de programación en concreto.

En primer lugar se ha utilizado WebStorm [10], un IDE inteligente para JavaScript que permite una serie de funcionalidades atractivas para el desarrollo de un proyecto web, además de contar con una licencia de estudiantes para el aprendizaje de la herramienta. Esta se utilizará para el apartado frontend, donde se hará uso de algunas de las características que ofrece y que lograrán facilitar el proceso, entre ellas se encuentran [11]:

- Editor inteligente, facilitando tareas tales como: auto completar código, detectar y sugerir correcciones, refactorización navegación, búsquedas etc
- Acceso a documentación de forma rápida.
- Depuración y pruebas directamente donde se edita el código.
- Integración de control versiones con Git o con otros VCS² de forma visual.
- Editor personalizable: temas, mapas de teclas, complementos y accesibilidad

¹IDE: Entorno de desarrollo integrado (Integrated Development Environment)

²VCS: Sistemas de control de versiones (Version Control Systems)

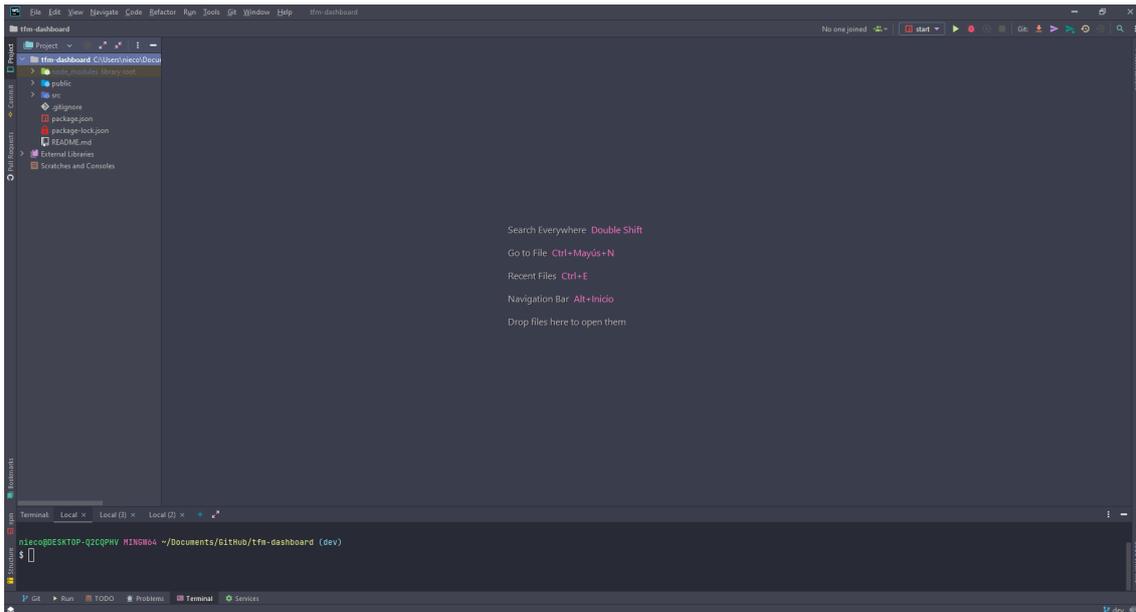


Figura 5.1: IDE Webstorm

Por otro lado, para el lado del servidor y el servicio de alarmas se ha utilizado IntelliJ IDEA [12], un IDE para Java que cuenta con características similares a las anteriormente citadas, pero centradas en proporcionar un entorno adecuado a la necesidades de los desarrolladores que utilizan el lenguaje de programación Java. Entre sus funcionalidades se puede encontrar: finalización de código inteligente, asistencia específica, ergonomía a la hora de programar, analizador de dependencias etc.

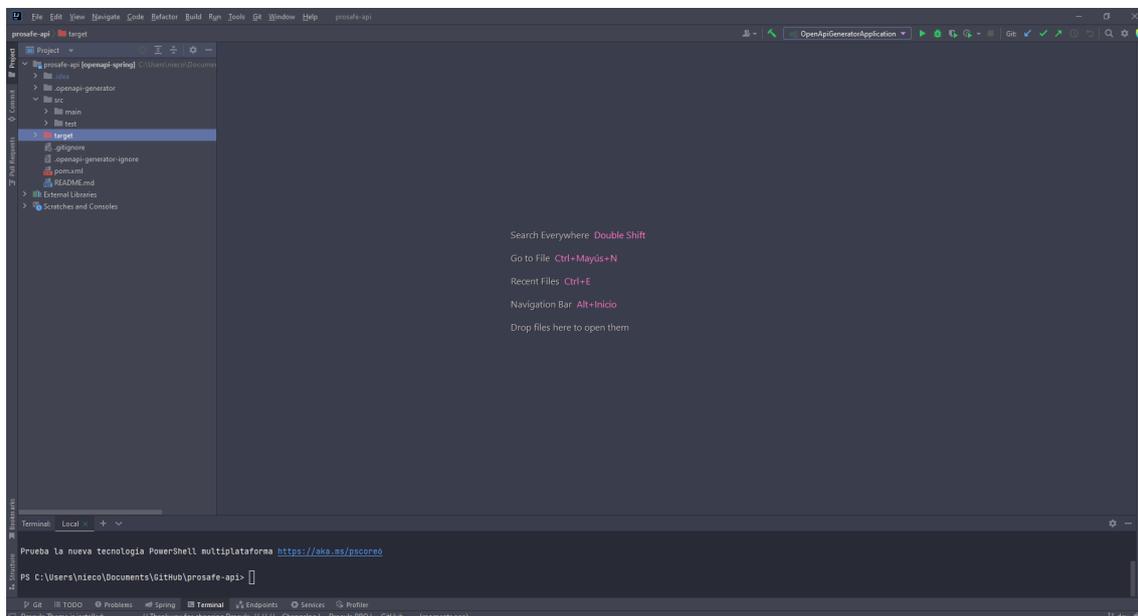


Figura 5.2: IDE IntelliJ IDEA

En otro orden de ideas, se ha utilizado Git [13] como sistema de control de versiones a través de la integración de este en los IDE de JetBrains comentados anteriormente. Además, GitHub ha sido la plataforma para alojar el proyecto por medio de la opción de organización, que te permite alojar diferentes repositorios en un mismo lugar. En la figura 5.3 se puede observar la organización con ProSafe, donde se puede ver los cuatro tipos

de repositorios creados: cliente web, cliente móvil, servidor(ejemplo) y funcionalidades ProSafe.

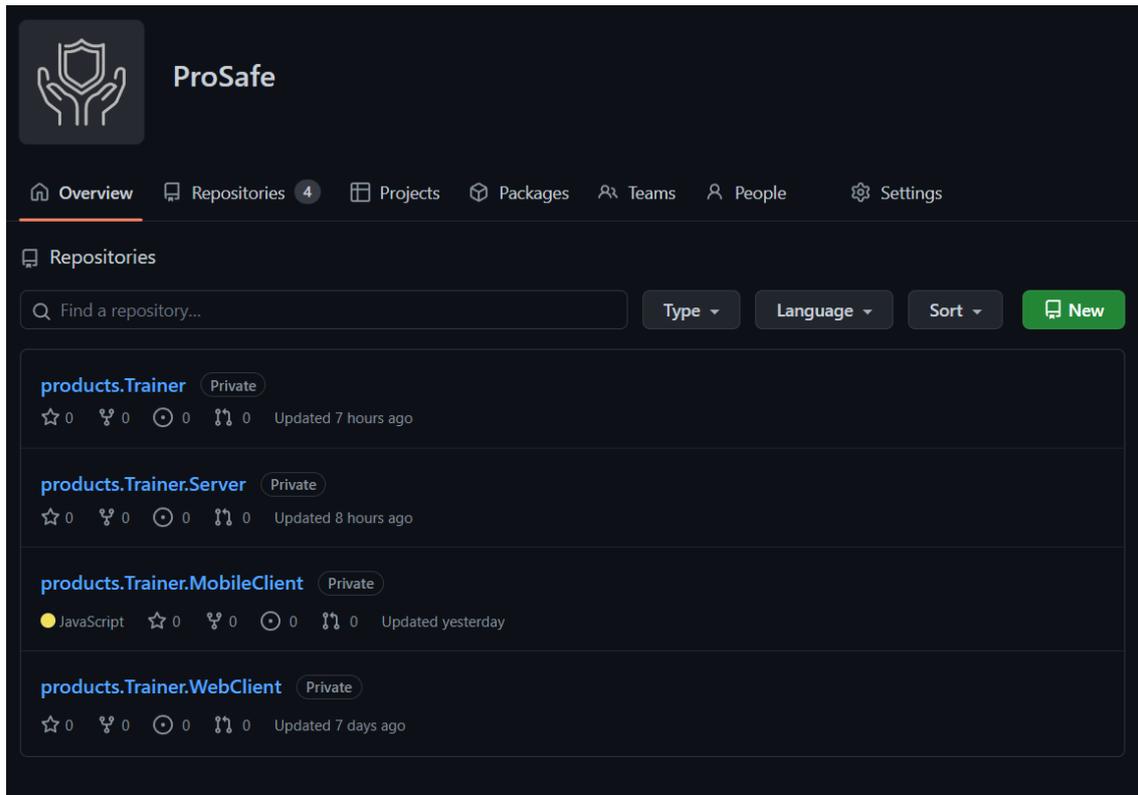


Figura 5.3: Organización ProSafe Github

5.2 Herramientas

5.2.1. Herramientas de diseño

La fase de diseño de un producto software es un punto indispensable para lograr una experiencia de usuario gratificante y un diseño de interfaz que consiga facilitar el uso de la aplicación. Para esta fase se ha hecho uso de herramientas que facilitan la creación de *mockups* y prototipos, acercando la vista de como será finalmente el producto final software. La herramienta elegida para este propósito es Figma [14], una plataforma que permite crear diseños consistentes y eficientes, con la posibilidad de añadir un gran número de *plugins* que faciliten en la automatización del proceso. En la figura 5.4 se puede observar una captura de pantalla en el proceso de creación de los *mockups* de la aplicación móvil.

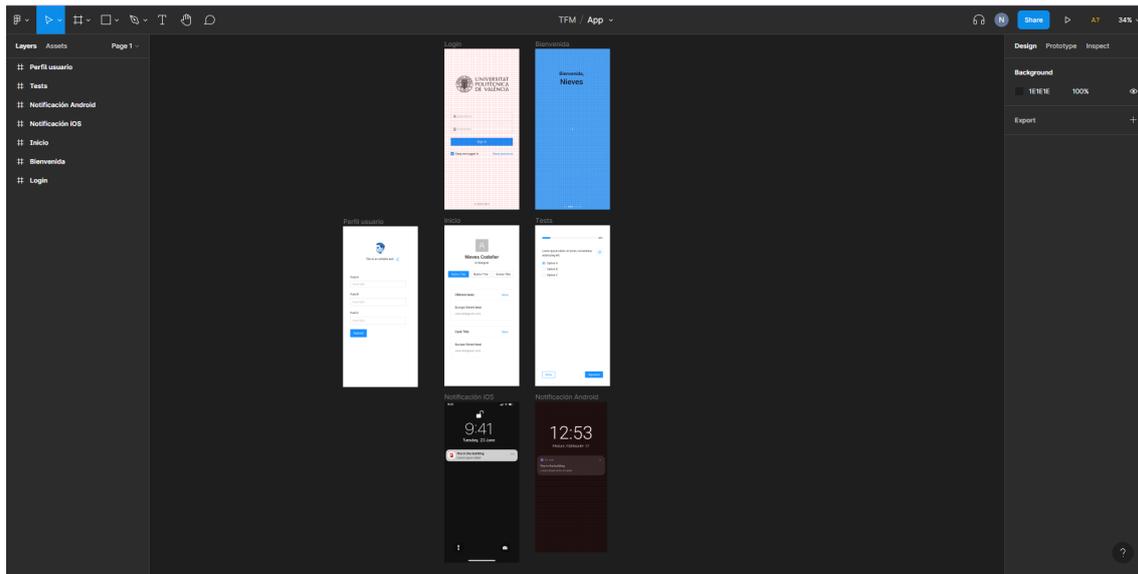


Figura 5.4: Pantalla de edición de la aplicación móvil en Figma

La razón principal por la que se ha elegido esta herramienta frente a otras tales como Balsamiq o Sketch es que Figma proporciona *plugins* relacionados con la librería de diseño que se va a utilizar en el lado del cliente, con lo que se pueden crear *mockups* más similares al producto final, utilizando para ello los componentes que ofrece la librería de diseño Ant Design. Además, permite realizar productos e incluso prototipos que las herramientas anteriormente citadas no proporcionan.

Por otro lado, se ha utilizado la herramienta draw.io para la realización de diagramas, tanto para el entendimiento personal de las arquitecturas, como para la exposición de diferentes diagramas en la memoria. En la figura 5.5 se puede observar la interfaz que ofrece draw.io.

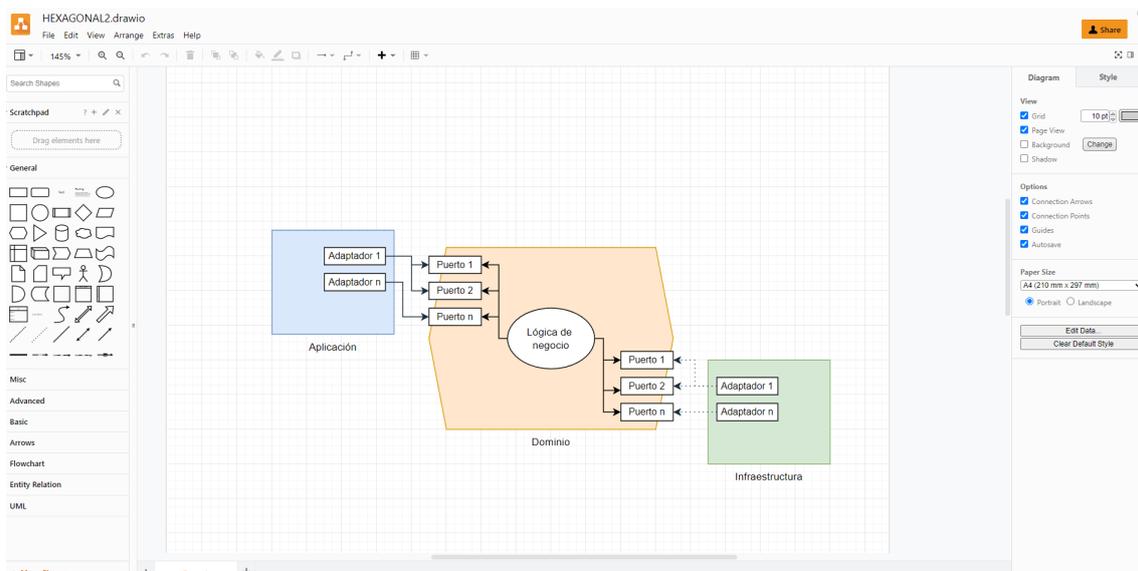


Figura 5.5: Pantalla de edición de draw.io

5.2.2. Herramientas de organización

Con el objetivo de organizar las tareas que hay que realizar en el proyecto, se ha utilizado la herramienta Notion [15]. Esta plataforma ofrece un espacio de trabajo que permite la creación de repositorios personalizables en los que se pueden añadir tableros de tareas. En la figura 5.6 se puede observar diferentes tareas que se han añadido en el tablero, donde se puede diferenciar la división de los componentes que conforman la aplicación (aplicación móvil, aplicación web y servidor), cada uno conteniendo las mismas columnas:

- *Not started*: Referencia a las tareas que hay que realizar en el proyecto, desde la investigación y diseño, hasta las tareas referentes a los casos de uso de la aplicación. Dichas tareas estarán etiquetadas por su prioridad y además, se detallarán y documentarán debidamente. Estas tareas se podrán subdividir en tareas más pequeñas para facilitar el trabajo de desarrollo (tal y como se puede observar en la figura 5.7, donde para el diseño de las interfaces de la aplicación móvil, se ha hecho una lista de las pantallas que hace falta diseñar).
- *In progress*: Referencia a las tareas que se están llevando a cabo actualmente por los desarrolladores.
- *Completed*: Referencia a las tareas que ya se han terminado.

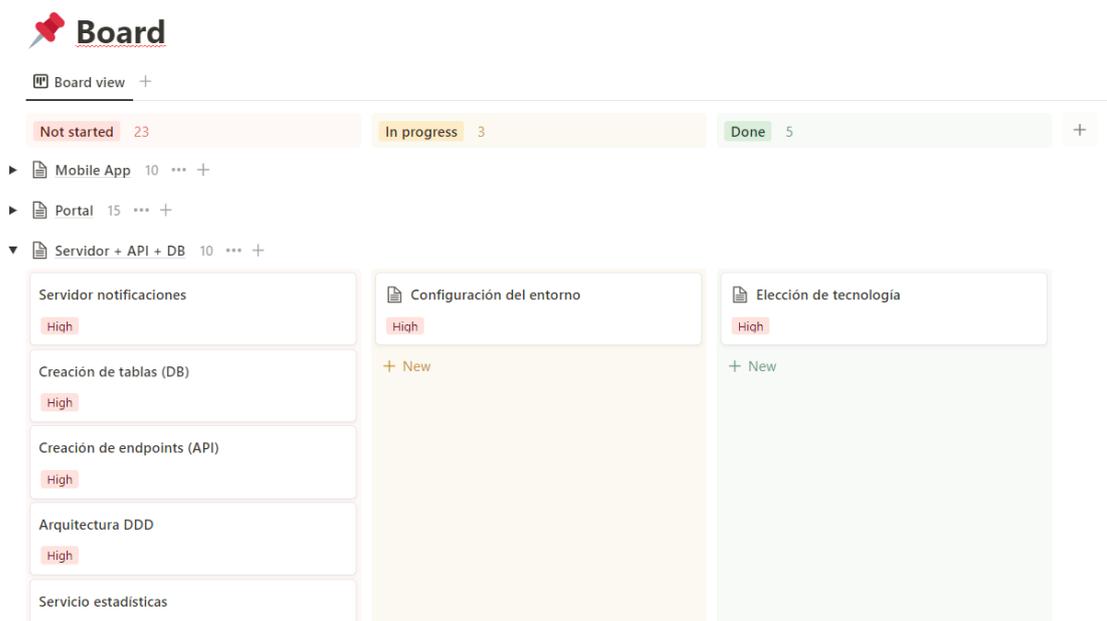
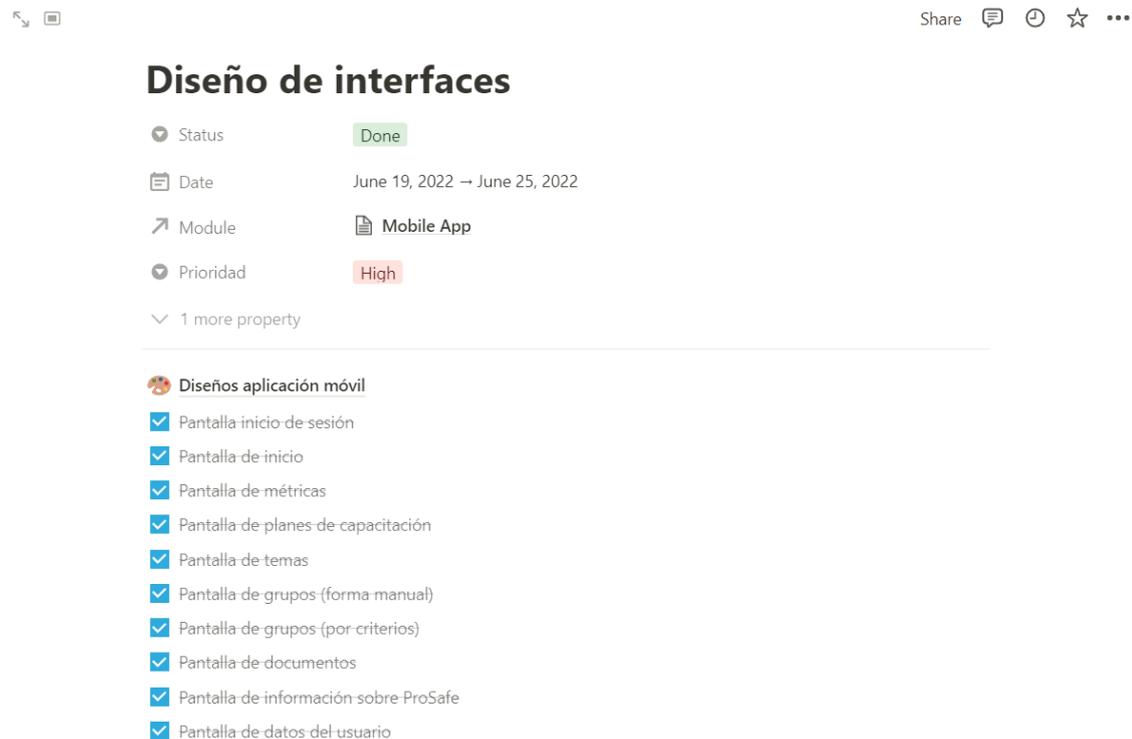


Figura 5.6: Tablero de tareas para la organización



The screenshot shows a task card in Notion titled "Diseño de interfaces". At the top right, there are icons for "Share", "Comments", "Clock", "Star", and "More". The task details are as follows:

- Status: Done (green box)
- Date: June 19, 2022 → June 25, 2022
- Module: Mobile App (with document icon)
- Prioridad: High (red box)
- 1 more property (collapsed)

Below the task details, there is a section titled "Diseños aplicación móvil" with a list of sub-tasks, each preceded by a blue checkmark:

- Pantalla inicio de sesión
- Pantalla de inicio
- Pantalla de métricas
- Pantalla de planes de capacitación
- Pantalla de temas
- Pantalla de grupos (forma manual)
- Pantalla de grupos (por criterios)
- Pantalla de documentos
- Pantalla de información sobre ProSafe
- Pantalla de datos del usuario

Figura 5.7: Tarea de diseño de interfaces desglosada en subtareas

Por otro lado, se ha creado un espacio en Notion donde además de recoger la información referente a las tareas, se almacena todo lo relacionado con el proceso de desarrollo del producto software: especificaciones, documentación, enlaces de interés, imágenes... Por ende, la información se encontrará centralizada en un solo lugar, lo que logrará agilizar el proceso. En la figura 5.8 se puede observar una captura de pantalla de las diferentes secciones (especificaciones, tablón y recursos) con las que cuenta el espacio de Notion creado para el proyecto.

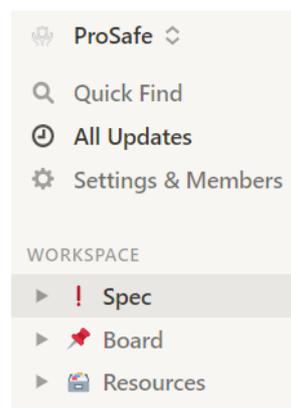


Figura 5.8: Organización de la información en Notion

En cuanto al motivo principal por el cual se ha elegido la herramienta Notion es que esta cuenta con la posibilidad de crear tableros, además de la creación de espacios de trabajo donde el usuario puede contar con la información ordenada y centralizada, en cambio, herramientas como Trello [16], aunque permiten una facilidad para la creación de tableros, no permiten crear espacios de trabajo.

5.3 Tecnologías y lenguajes de programación

5.3.1. Frontend

Librerías y *frameworks* comunes en web y móvil

En este apartado se muestran algunas de las librerías más importantes utilizadas en el proyecto, definiendo su propósito general y mostrando las razones por las cuales se ha elegido estas frente a otras opciones.

React [17] es una librería *opensource* de JavaScript que permite construir interfaces de usuario de manera sencilla. Una de las características principales de esta librería es que está basada en la creación de componentes que mantienen su estado. Además, estos componentes son reutilizables, permitiendo que gran parte de la comunidad que trabaja con esta librería comparta sus componentes, concediendo la posibilidad a cualquier programador poder utilizarlo en su aplicación. Por otro lado, existen una gran variedad de módulos y herramientas compatibles con React, además de contar con una gran comunidad que añade contenido relacionado con el aprendizaje de esta librería. Una de estas facilidades comentadas es el entorno *Create React App*, que es una forma de crear una aplicación de React *single-page* de una forma sencilla y sin necesitar que el programador invierta tiempo en crear una configuración exhaustiva para el proyecto.

La elección de esta librería recae en un conocimiento previo de la misma, además de las características anteriormente citadas, lo que permite que ya se cuente con conocimientos y el desarrollo será de forma más ágil. Por otro lado, React es una librería actualmente en auge, aumentando diariamente el número de descargas de la misma en comparación con otras librerías y frameworks destinados a un propósito similar. En la figura 5.9 se puede observar el número de descargas en npm de las librerías React y Vue, y los frameworks Angular y Angular Core.

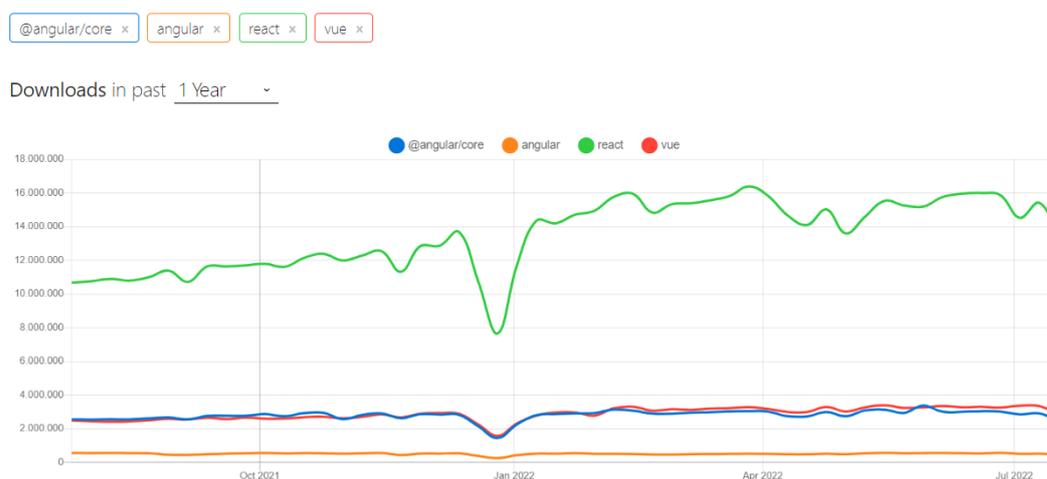


Figura 5.9: Gráfico de comparación (extraído de npmtrends.com)

En la tabla 5.1 se muestran de forma resumida las ventajas y desventajas de cada uno de los frameworks y librerías, algunas de las cuales se han definido en el párrafo anterior.

Framework	Ventajas	Desventajas
Angular	<ul style="list-style-type: none"> ▪ Mantenimiento realizado por Google. ▪ Comunidad grande de desarrolladores. ▪ Arquitectura MVC³. 	<ul style="list-style-type: none"> ▪ Verboso y complejo. ▪ Framework pesado. ▪ Curva de aprendizaje elevada. ▪ Disminución de su popularidad a lo largo de los años.
Vue	<ul style="list-style-type: none"> ▪ Curva de aprendizaje baja. ▪ Componentes reutilizables. ▪ Soporte de la comunidad. 	<ul style="list-style-type: none"> ▪ Complejidad en la reactividad. ▪ Falta de apoyo por parte de la comunidad para proyectos de grandes dimensiones. ▪ Recursos limitados en comparación con React y Angular.
React	<ul style="list-style-type: none"> ▪ El alumno ya cuenta con conocimientos previa con esta librería. ▪ Curva de aprendizaje baja. ▪ Facilidad de creación de aplicaciones web. ▪ Componentes reutilizables. ▪ Comunidad en auge (figura 5.9). 	<ul style="list-style-type: none"> ▪ Entorno muy cambiante. ▪ Dificultad en el entendimiento de JSX. ▪ Documentación no tan detallada debido al entorno cambiante.

Tabla 5.1: Tabla de comparación de frameworks del lado del cliente

Ant Design [18] y Ant Design Mobile [19] es una librería de diseño que cuenta con una base de guías sobre la implementación visual que deberían de ser aplicadas en las soluciones desarrolladas con la misma, aportando componentes utilizables que siguen estas pautas. En este caso, se ha elegido esta librería ya que cuenta con un diseño no ligado a un sistema operativo, y al tratarse de una aplicación que pretende tener una versión mó-

³MVC: Patrón de diseño que propone separar el código en tres componentes diferentes: modelo, vista y controlador (Model View Controller)

vil (Android y iOS) y una web. Además, posee una integración con la herramienta Figma, lo que permite crear mockups y prototipos más cercanos a la idea final de la aplicación.

i18next [20] es un *framework* con el propósito de realizar el apartado de la traducción de una aplicación. Dicho *framework* cuenta con una serie de opciones referentes a la traducción tales como soporte para variables (interpolación, anidadas, formas plurales e indefinidos), búsqueda de locales, cacheo de recursos en el navegador etc. El uso de esta herramienta permite poder definir estas variables y que la plataforma soporte, en este caso, tanto el inglés como el español.

Axios [21] es una librería que permite crear peticiones HTTP basadas en promesas y ofrece una serie de características tales como la interceptación y transformación de peticiones y respuestas, cancelación de peticiones, transformación automática de datos en formato JSON ⁴ y protección ante el ataque conocido con las siglas CSRF ⁵ por parte del cliente. Además, se hará uso de la propiedad "header", concretamente el parámetro "Authorization", para añadir la información referente a las credenciales del usuario, consiguiendo así pudiendo realizar peticiones para el inicio de sesión en la aplicación.

DOMPurify Librería que sirve para sanitizar del código HTML después de una petición HTTP y prevenir del ataque denominado XSS attack ⁶.

Web

En esta sección se van a listar algunos de las librerías utilizadas para el desarrollo del apartado web de la aplicación. De esta forma se pueden encontrar:

- *Lodash debounce*: Lodash es una librería que contiene una serie de funciones que pueden ser útiles para el proyecto. En concreto, *debounce* retrasa la invocación de una función que se le haya pasado, esta utilidad es utilizada en el proyecto para la búsqueda de empleados una vez el usuario deje de escribir.
- *Ant-design/charts*: Ant design cuenta con una extensa colección de componentes referidos a gráficos, que resultan útiles para el apartado de estadísticas de la aplicación.

Móvil

En cuanto al apartado móvil, existen una gran variedad de herramientas para el desarrollo de aplicaciones móviles, algunas de ellas se tratan de soluciones nativas, directamente desarrolladas en herramientas exclusivas tales como Android Studio o XCode. Aunque estas dos opciones podrían ser válidas, en este proyecto se ha elegido utilizar una herramienta que permite convertir aplicaciones web en aplicaciones móviles nativas. La elección de esta tecnología recae en la sencillez que se presenta a la hora de desarrollar desde un lenguaje de programación y tecnología como puede ser la web, con el que el programador ya cuenta con conocimiento y le permite centrarse en una sola tecnología y conseguir con un menor esfuerzo transformar estas páginas en aplicaciones nativas.

⁴Notación de objeto de JavaScript (*JavaScript Object Notation* para el intercambio de datos)

⁵CSRF: Falsificación de petición en sitios cruzados (Cross-site request forgery)

⁶XSS attack: Secuencia de comandos en sitios cruzados (Cross-site scripting)

Capacitor.js [22] es una librería gratuita que permite la conversión de aplicaciones web en aplicaciones móviles nativas. De esta forma, el desarrollador puede evitar programar en lenguajes de programación nativos tales como Kotlin o Swift, y programar con tecnologías web, lo que le permite que con una inversión menor de tiempo en el desarrollo puede crear aplicaciones para diferentes plataformas. Además, permite la creación de PWA ⁷, un nuevo concepto de páginas web que mediante el uso de “service Workers”, pueden ejecutarse en segundo plano sin tener que guardarse dentro del navegador. Aunque esta última aproximación puede ser interesante, se ha decidido optar por crear una aplicación web, envuelta por un contenedor nativo, tales como IOS y Android. Además, es compatible la librería React, con lo que puede ser añadido al proyecto de manera sencilla y sin tener que cambiar de tecnología.

5.3.2. Backend

Spring [23] es un framework de código abierto para Java que permite desarrollar aplicaciones backend. Aunque existen otras opciones que se podrían utilizar para el desarrollo del servidor, tales como Express.js o Django, se ha elegido esta opción por varios motivos que serán expuestos en esta sección. En primer lugar, aunque no se cuenta con experiencia previa con el framework, si se cuenta con experiencia desarrollando con el lenguaje de programación Java, ya que ha sido el lenguaje base visto en el grado. Por otro lado, cuenta con una serie de ventajas y características que son de utilidad a la hora de desarrollar el servidor del proyecto, entre ellas la inyección de dependencias, la comunidad ya consolidada, soporte para múltiples hilos y la facilidad de creación de código por medio de las anotaciones que proporciona tanto Java como Spring. Además, otorga una herramienta denominada Spring Boot [24] para la configuración del proyecto, lo que ahorra tiempo al desarrollador.

En la tabla 5.2 se muestran algunas de las ventajas y desventajas de los frameworks para el lado del servidor que se han investigado antes de elegir la opción planteada.

Framework	Ventajas	Desventajas
Express.js	<ul style="list-style-type: none"> ■ Desarrollo ágil. ■ Comunidad amplia: gran cantidad de ejemplos. ■ Curva de aprendizaje baja. ■ Fácil integración con otros servicios, librerías y middleware. 	<ul style="list-style-type: none"> ■ Dificultad para programadores que no estén familiarizados con programación orientada a eventos. (driven-event nature) ■ Minimalismo inherente en Express.js

⁷PWA: Aplicación web progresiva (Progressive Web Application)

Django	<ul style="list-style-type: none"> ▪ Desarrollo ágil (una vez se conoce) ▪ Comunidad amplia. ▪ Seguridad ante ataques tales como inyecciones SQL y 	<ul style="list-style-type: none"> ▪ Naturaleza monolítica. ▪ No encaja con proyectos pequeños ▪ Curva de aprendizaje elevada.
Spring	<ul style="list-style-type: none"> ▪ Flexible, poco acoplado, portable, rápido y seguro. ▪ Facilidad para hacer testing. ▪ Conocimiento muy básico del alumno sobre este framework. 	<ul style="list-style-type: none"> ▪ Curva de aprendizaje difícil. ▪ Abrumador número de opciones para los desarrolladores

Tabla 5.2: Tabla de comparación de frameworks del lado del servidor

Por otro lado, se ha hecho uso de la librería *lombok*, que permite reducir el código a través de diferentes anotaciones. Por ejemplo, se puede especificar la anotación “@AllArgsConstructor”, esta se encargará en tiempo de compilación de crear un constructor de la clase donde se haya especificado, a través de los parametros definidos en la misma.

Base de datos

MySQL [25] es un sistema de gestión de base de datos relacional gratuito. Su principal ventaja frente a otras es la sencillez de uso, además de contar con otras ventajas tales como su rapidez, seguridad y solicitar de pocos requerimientos y eficiencia de memoria. Una de las razones principales por las que se ha elegido esta tecnología es debido a que se cuenta con experiencia previa básica, sobretodo a la hora de la gestión de bases de datos relacionales que utilizan SQL⁸ para el manejo de la información.

⁸SQL: Lenguaje de Consulta Estructurada

CAPÍTULO 6

Desarrollo

En el presente capítulo se va a detallar el proceso de desarrollo del proyecto software, los problemas que han ido surgiendo y la solución a estos. En el presente capítulo se va a exponer el proceso seguido para el desarrollo de la aplicación software ProSafe.

6.1 Frontend

6.1.1. Preparación del entorno

Para la elaboración del proyecto se ha hecho uso del CLI¹ oficial de React elaborado por Facebook, `create-react-app` [26]. Dicho entorno permite crear aplicaciones de una forma sencilla sin necesidad de invertir tiempo en la configuración del proyecto. Sin embargo, si se han realizado una serie de cambios tales como cambios en la organización del proyecto o la eliminación de librerías relacionadas con las analíticas de la página. En la figura 6.1 se puede observar el entorno creado con este CLI.

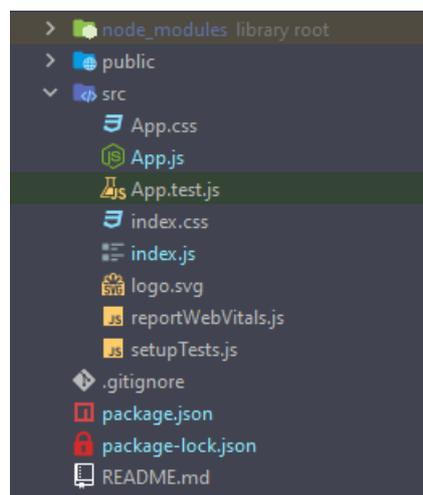


Figura 6.1: Estructura de carpetas “create-react-app”

Por otro lado, y tal como se ha expuesto en el apartado de tecnologías 5.1, se ha hecho uso de los IDE proporcionados por JetBrains, y se han realizado una serie de configuraciones e instalación de *plugins* con el objetivo de facilitar el proceso de desarrollo.

¹CLI: Intefaz de línea de comandos, permite comunicarse con un sistema mediante comandos.

6.1.2. Estructura general de los proyectos

En la siguiente sección se va a detallar la estructura general que siguen los dos proyectos frontend. Estos dos proyectos cuentan con una estructura de carpetas muy similar, esto es debido a que los dos han sido desarrollados a través de la librería React. De esta forma, en los dos proyectos se puede encontrar la distribución de carpetas muy similar representadas en la figura 6.2.

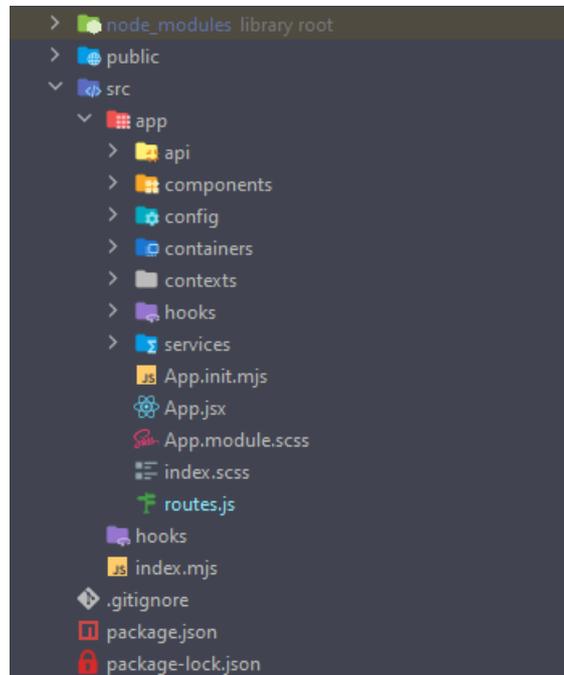


Figura 6.2: Estructura de carpetas frontend

A continuación se va a proceder a listar y explicar cada una de las carpetas y archivos que están contenidas en el directorio “src/app”, y hacen referencia al desarrollo del lado del cliente:

- Carpeta `api`: Contiene el código referente a la creación de una instancia de axios. Esta será utilizada para la petición de recursos a la API creada.
- Carpeta `components`: Contiene los componentes independientes y reutilizables que se han creado para la aplicación.
- Carpeta `containers`: Contiene los contenedores, que se encargan de envolver a los componentes y proporcionar un estado a los mismos.
- Carpeta `config`: Contiene todas las configuraciones del proyecto, desde las URL de las peticiones a la API, como configuraciones de las librerías utilizadas.
- Carpeta `context`: Contiene tanto el contexto global de la aplicación como los contextos diferenciados de los módulos.
- Carpeta `hooks`: Contiene todas las funciones que permiten enlazar el estado de React con el ciclo de vida con el que cuentan los componentes.
- Carpeta `services`: Contiene los servicios de la aplicación que se encargan de conectar la aplicación con la API del servidor.

- Archivo `App.init.mjs`: Contiene la inicialización de la aplicación de React.
- Archivo `App.jsx`: Contiene la estructura de rutas de la aplicación, además de inicializar la configuración de la aplicación.
- Archivo `App.module.scss`: Modulo que contiene los estilos globales creados por el alumno para la aplicación.
- Archivo `index.scss`: Contiene las importaciones referentes a las hojas de estilo proporcionadas por Ant Design.
- Archivo `routes.js`: Contiene la estructura de rutas de la aplicación.

6.1.3. Modulo internalización del proyecto

Para realizar una aplicación multilingüe se ha procedido a declarar un modulo que consigue dotar a la aplicación de la posibilidad de cambiar de idioma. En este caso se ha decidido añadir los idiomas: español e inglés, aunque con la implementación descrita es fácilmente ampliable a otros idiomas. El funcionamiento de este modulo se basa en la detección del idioma a través del navegador, aunque se ha añadido un selector para que el usuario pueda cambiar de forma sencilla de idioma. Tal y como se puede observar en la figura 6.3, los archivos donde se encuentran las definiciones de las traducciones se encuentran en la ruta `public/locales`.

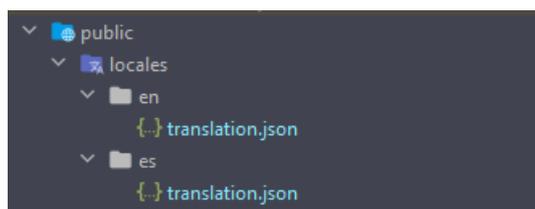


Figura 6.3: Documentos internalización: español e inglés

6.1.4. Estadísticas

Para la realización del apartado de estadísticas se ha utilizado la librería Ant Design Charts [27], que cuenta con una variedad de gráficos extensa, en los que poder añadir la información requerida. Además, se ha utilizado unas tarjetas para añadir datos interesantes sobre la organización: número total de empleados, de puntos, de grupos y planes creados etc.

Tanto los datos de la organización, como las gráficas, requerirán de una petición al servidor mediante la librería *axios* donde este contestará con los datos necesarios (si no ha habido ningún tipo de problema), y estos serán transformados al formato requerido por los componentes.

6.1.5. Aplicación móvil

Configuración de capacitor

Con el objetivo de permitir las notificaciones en la aplicación se ha creado un archivo llamado `capacitor.config.ts` (figura 6.4), donde se especifican los datos de configuración de la aplicación y la activación de los *plugins* referentes a las notificaciones.

```
import { CapacitorConfig }
  from '@capacitor/cli'

const config: CapacitorConfig = {
  appId: 'es.upv.niecogi.tfm.app',
  appName: 'ProSafe',
  webDir: 'build',
  bundledWebRuntime: false,
  plugins: {
    PushNotifications: {
      presentationOptions: ["badge", "sound", "alert"]
    }
  }
}

export default config
```

Figura 6.4: Configuración de capacitor

Exportar proyecto

Como ya se ha expuesto anteriormente, se ha utilizado la herramienta Capacitor.js, que permite convertir una aplicación web en una aplicación nativa de iOS o Android. Una vez la aplicación ha sido desarrollada, se procede a crear la aplicación a través del comando `npm run build`. En este caso, Capacitor ofrece una serie de comandos para poder exportar esta aplicación tanto a iOS como Android. En la figura 6.5 se pueden observar como se han añadido una serie de comandos para poder exportar y poner en marcha la aplicación en las diferentes plataformas descritas.

```
"scripts": {
  "start": "react-scripts start",
  "start:android": "npx cap run android",
  "start:ios": "npx cap run ios",
  "build": "react-scripts build && npx cap copy",
  "test": "react-scripts test"
},
```

Figura 6.5: Scripts para ejecutar la aplicación

Una vez los comandos referentes a la *build* del proyecto han sido ejecutados, se ha procedido a abrir desde terminal el programa Android Studio (programa que debe de ser descargado antes de realizar este paso) mediante el comando `npx cap open android`. A continuación, se ha procedido a emular la aplicación. En la figura 6.6 se puede observar la aplicación en el emulador que proporciona Android Studio.

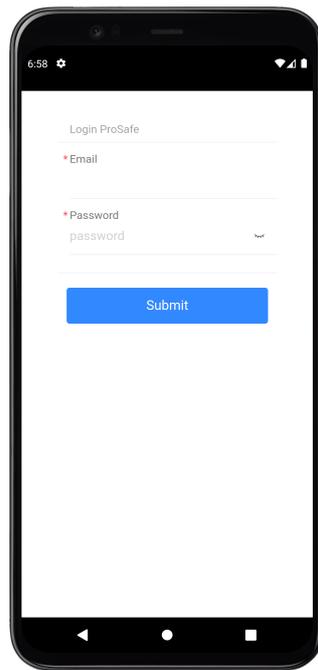


Figura 6.6: Aplicación en Android

De forma similar, se ha ejecutado la aplicación, pero esta vez para el sistema IOS. Cabe destacar que para realizar este paso se requiere un ordenador con un sistema operativo macOS, que tenga instalado XCode. Una vez se haya sincronizado la aplicación, se procede a ejecutar desde consola el siguiente comando: `npm run cap open ios`, el cual abre la aplicación XCode. En la figura 6.7 se puede observar la aplicación en el emulador de XCode.

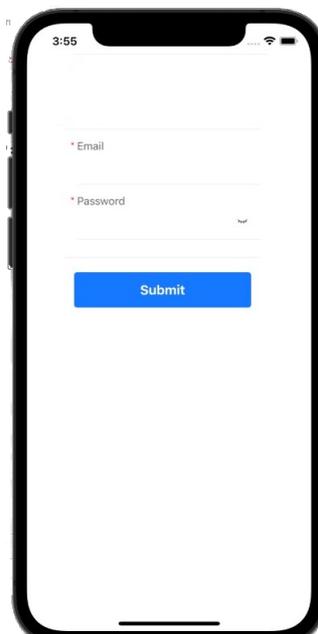


Figura 6.7: Aplicación en IOS

6.2 Backend

Con el objetivo de lograr mostrar la integración de la aplicación ProSafe al servidor de una organización, se ha procedido a desarrollar un servidor de ejemplo al cual se le integrará el servicio de ProSafe, tal y como ha sido explicado en el apartado 3.1.5.

6.2.1. Integración del paquete ProSafe

En este apartado se va a detallar la integración de ProSafe en una organización. Como no se cuenta con una organización real, se ha decidido crear un servidor de ejemplo al cual se le añadirá ProSafe. Cabe destacar, que este servidor es muy simple y por ello solo contendrá los datos referentes a los permisos, los empleados y los departamentos.

En este caso, para añadir en una organización ProSafe se necesitaría integrar en el repositorio referente al servidor de la organización una dependencia en el `pom.xml`, tal y como se muestra en la figura 6.8.

```
<dependency>
  <groupId>com.company</groupId>
  <artifactId>ProSafe</artifactId>
  <version>1.0-SNAPSHOT</version>
</dependency>
```

Figura 6.8: Integración de ProSafe

Una vez añadido, se debe de configurar los *endpoints* de la empresa para que llamen a las funcionalidades de ProSafe, o gestionar la funcionalidad que ofrece.

6.2.2. Estructura de archivos

Servidor de ejemplo

En la figura 6.9 se puede observar la estructura de carpetas utilizadas para ejemplificar la aplicación de la empresa a la que se añadirá ProSafe.

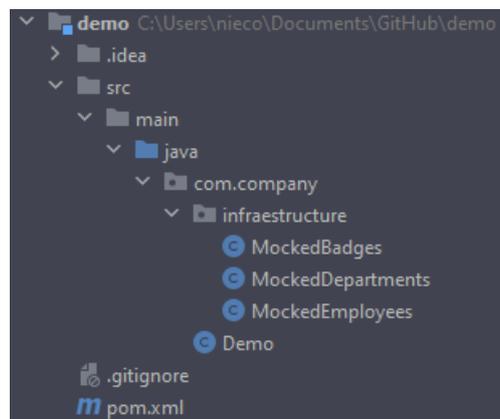


Figura 6.9: Estructura de archivos:Servidor demo

A continuación se expondrá el archivo y la carpeta que lo componen, estas irán almacenadas en un proyecto denominado `demo`, donde la funcionalidad principal se encontrará en el paquete representado por los módulos: `src.main.java.com.company`.

- **Carpeta** *infraestructure*: Contiene la definición de las diferentes clases referentes tanto a la lista de empleados y departamentos como a la búsqueda de permisos de un empleado (*badges*).
- **Clase** *Demo*: Contiene la definición de un objeto a través de un *patrón builder*², construyendo un objeto con las propiedades especificadas en el servidor ProSafe. De esta forma, se instancia un objeto de la clase “Trainer” y se le añaden las configuraciones referentes a los repositorios de datos especificados en la carpeta *infraestructure*.

Servidor ProSafe

En la figura 6.10 se puede observar la estructura de carpetas utilizada para el desarrollo del producto software ProSafe.

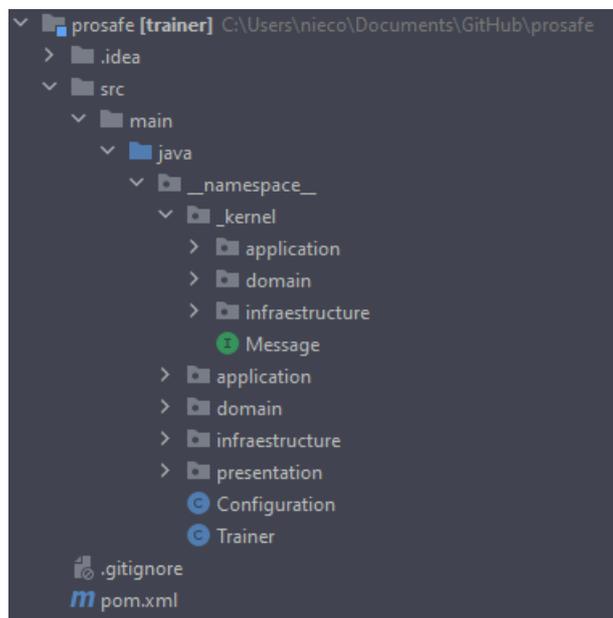


Figura 6.10: Integración de ProSafe

A continuación, se expondrán las diferentes carpetas que lo componen, todas ellas irán almacenadas en un proyecto denominado *prosafe*, donde la funcionalidad principal se encontrará en el paquete representado por los módulos: `src.main.java._namespace_`. Cabe destacar, que la carpeta “`_namespace_`” será modificada por el nombre dado en el proyecto de la empresa a la que se añadirá ProSafe, todo ello para respetar la coherencia y el *branding*³ de la organización.

- **Carpeta** `_kernel`: Contiene las interfaces bases que definen algunas de las clases utilizadas en la aplicación. La definición de interfaces es utilizada en este caso para añadir control y auto documentar el código. Esta carpeta cuenta con la siguiente estructura:
 - **Carpeta** `application`: Contiene las interfaces referentes a esta capa, tales como los comandos, *handlers*, *queries* y acciones.

²Patrón builder: Patrón de diseño útil cuando es necesario especificar un número elevado de configuraciones, proporcionando ventajas tales como el control en la construcción y la descripción

³Branding: Proceso por el cual una marca de una organización ha sido construido, contando con un conjunto de atributos y valores inherentes a esta marca creada.

- **Carpeta domain:** Contiene las interfaces que definirán el dominio, tales como eventos, repositorios, definición del CRUD⁴, entidades y propiedades, entre otras.
- **Carpeta infraestructure:** Contiene la interfaz referente al broker.
- **Carpeta presentation:** Se encarga de traducir la comunicación a través de la API, la cual se ocupa de mapear las rutas a través de un controlador y llamar a la capa de aplicación.
- **Carpeta application:** Contiene los servicios, excepciones y tareas que conectan el dominio de la aplicación con el mundo exterior. Esta capa se encarga de implementar la lógica y llevar a cabo las acciones que han sido solicitadas por parte de la capa de presentación.
- **Carpeta domain:** Define las entidades que participan en la lógica de negocio, y las reglas asociadas a cada una de ellas.
- **Carpeta infraestructure:** Encargada de implementar las dependencias que tiene la capa de dominio, permitiendo que estas puedan ser consumidas y que puedan persistir los objetos de negocio. Además, en esta carpeta se define la tarea programada para los tests, esta tarea será ejecutada en un lapso de tiempo determinado, definido por la frecuencia del plan de capacitación) y realiza las acciones asociados al lanzar los test de empleados.
- **Clase Configuration:** Clase encargada de definir las configuraciones generales del proyecto, ya sea el puerto, el nombre de la aplicación y la versión. Además, se definen los repositorios referentes a los empleados, la seguridad y los departamentos que conforman la organización.
- **Clase Trainer:** Punto de entrada de la aplicación, se encarga de escanear y ejecutar el proyecto a través de las anotaciones y clases definidas.

⁴CRUD: Acrónimo utilizado para definir las operaciones que se pueden realizar sobre la información requerida. Estas acciones se resumen en : “Crear, Leer, Actualizar y Borrar”

6.2.3. Patrón *message broker*

Para conseguir que los cambios en el módulo de entrenamiento tengan reacción en las estadísticas, se ha implementado un patrón *message broker* [28], donde el módulo referente a las estadísticas se suscribe a los eventos especificados por el módulo de entrenamiento. En la figura 6.11 se puede observar un ejemplo de este patrón, donde un usuario responde a una pregunta, el módulo *training* almacena esta respuesta y notifica que ha ocurrido un evento al *broker*. Posteriormente, el módulo de stats, que está suscrito a las publicaciones de *training*, recibe una notificación y realiza las operaciones oportunas.

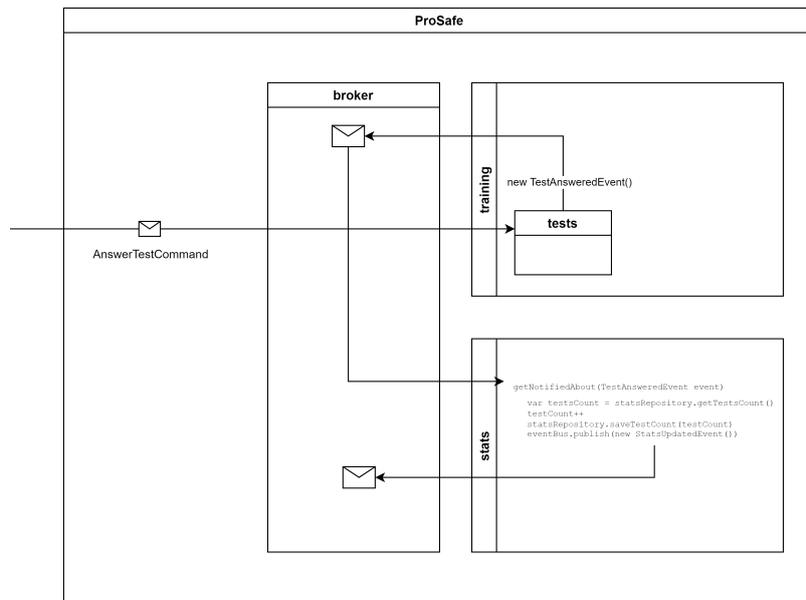


Figura 6.11: Ejemplo de respuesta de una pregunta con broker

CAPÍTULO 7

Pruebas

En el presente capítulo se van a presentar las diferentes pruebas de aceptación con el objetivo de validar el comportamiento esperado.

7.1 Pruebas de aceptación

Con el objetivo de mostrar el funcionamiento de la aplicación se ha decidido añadir este apartado, en el que se van a enumerar las distintas pruebas de aceptación que corresponden con distintos escenarios de ejecución. Cabe destacar que los ejemplos y los temas añadidos tienen que ver con contenido real sobre PRL en el ámbito docente.

Acceso panel de gestión ProSafe

Cómo ya se ha mencionado en la memoria, los únicos empleados que pueden acceder al panel de gestión son los responsables de PRL, que contarán con una cuenta proporcionada por la empresa. Si este usuario no se ha identificado con anterioridad, se le mostrará la página principal de acceso, donde deberá proporcionar las credenciales oportunas. En la figura 7.1 se puede observar la página de *login*.

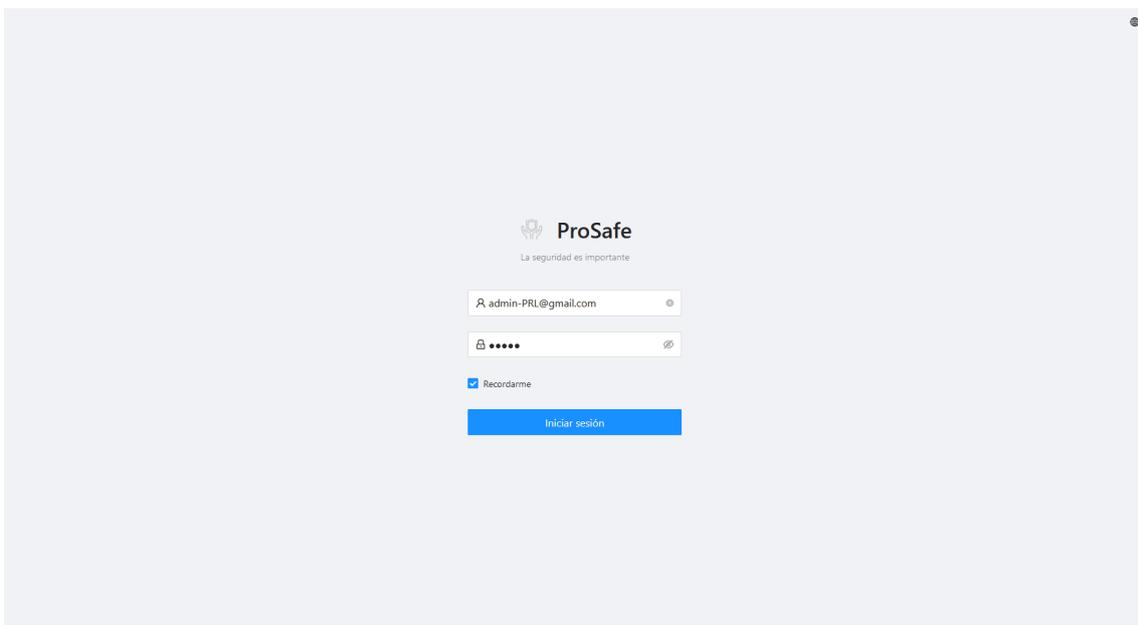


Figura 7.1: Página principal de *login*

Creación grupo

Para la creación de un tema el usuario debe de acceder al apartado de gestión, concretamente a la pestaña referida a los grupos. El usuario pulsa en la opción de crear un grupo, este se creará con un nombre genérico y los demás campos del formulario estarán vacíos, tal y como se puede comprobar en la figura 7.2.

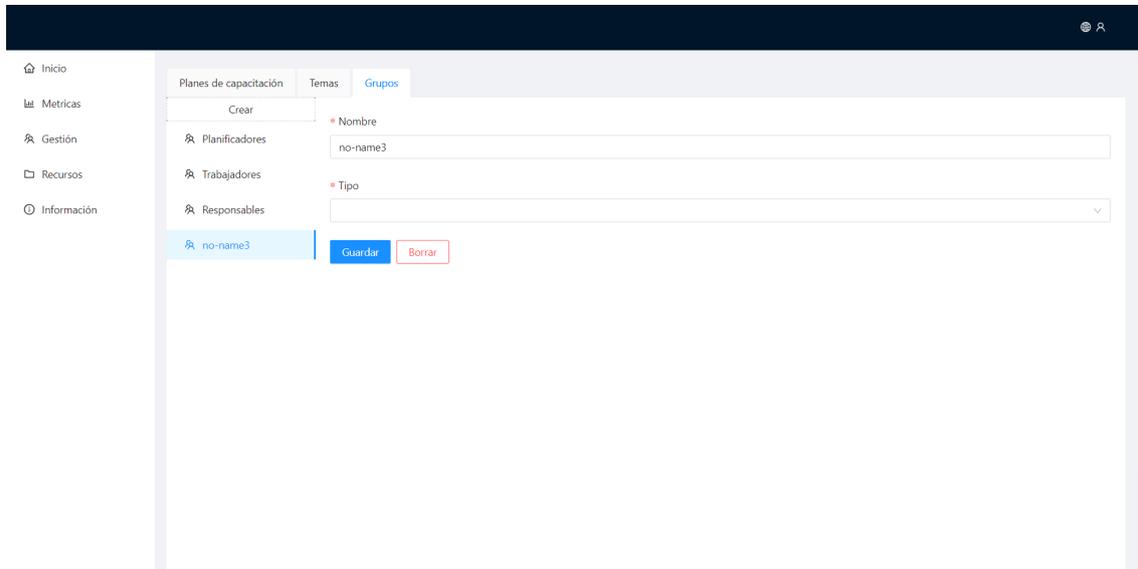


Figura 7.2: Creación grupo: Añadir grupo vacío

El usuario añade la información referente al grupo, en este caso se ha añadido el grupo "DSIC" y un criterio, tal y como se puede observar en la figura 7.3.

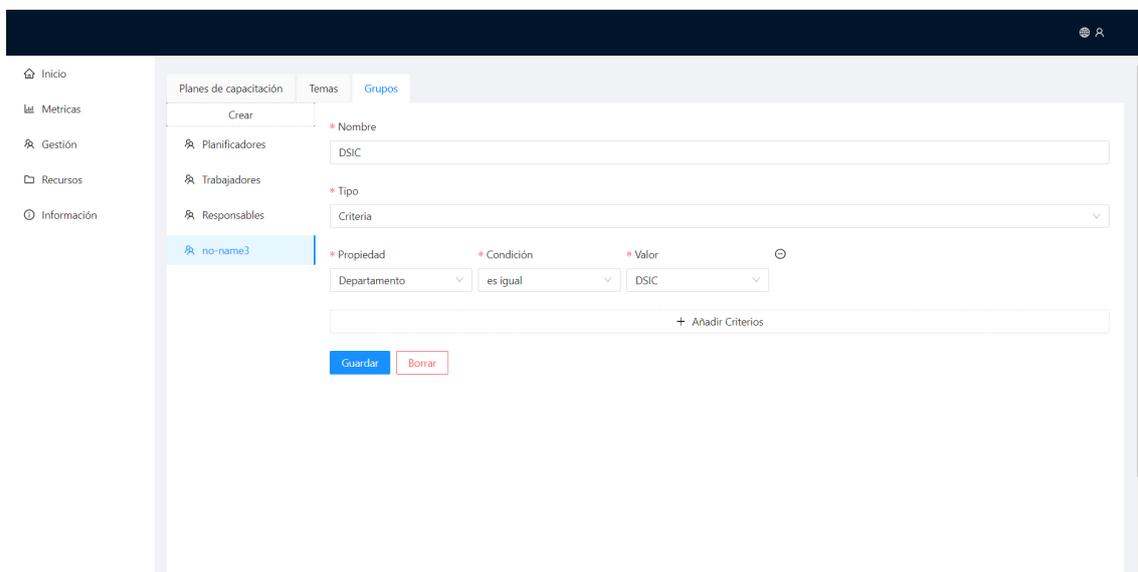


Figura 7.3: Creación grupo: Añadir información

Una vez la información ha sido introducida, y se hayan validado los campos, el sistema guarda la información en la base de datos y se añade el nombre elegido para el grupo en el menú lateral, tal y como se puede observar en la figura 7.4.

Planes de capacitación Temas Grupos

Crear

Planificadores

Trabajadores

Responsables

DSIC

* Nombre
DSIC

* Tipo
Criteria

* Propiedad * Condición * Valor
Departamento es igual DSIC

+ Añadir Criterios

Guardar Borrar

Figura 7.4: Creación grupo: Grupo creado

Creación tema

Para la creación de un tema el usuario debe de acceder al apartado de gestión, concretamente a la pestaña referida a los temas. El usuario pulsa en la opción de crear un tema, este se creará con un nombre genérico, y con ninguna pregunta añadida, tal y como se puede comprobar en la figura 7.5.

Planes de capacitación Temas Grupos

Crear

Riesgos biológicos

Riesgos ergonómicos

Riesgos de la voz

no-name3

* Nombre
no-name3

+ Añadir Pregunta

Guardar Borrar

Figura 7.5: Creación de tema: Añadir tema vacío

El usuario añade la información referente al tema sobre PRL, en este caso se ha añadido el tema “Riesgos psicosociales” y una pregunta, tal y como se puede observar en la figura 7.6.

Planes de capacitación Temas Grupos

Crear

* Nombre

Riesgos psicosociales

Pregunta 1

* Pregunta

Los factores de naturaleza psicosocial se definen como...

- La interacción entre determinadas condiciones de trabajo y aspectos relacionados con las características de los trabajadores más allá del entorno laboral
- Se relacionan principalmente con la organización, el contenido del trabajo y la realización de la tarea
- Si son desfavorables, los efectos negativos afectan tanto a la salud del trabajador como de la empresa
- Todas las afirmaciones son correctas

+ Añadir campo

Figura 7.6: Creación de tema: Añadir información

Una vez la información ha sido introducida, y se hayan validado los campos, el sistema guarda la información en la base de datos y se añade el nombre elegido para el tema en el menú lateral, tal y como se puede observar en la figura 7.4.

Planes de capacitación Temas Grupos

Crear

* Nombre

Riesgos psicosociales

Pregunta 1

* Pregunta

Los factores de naturaleza psicosocial se definen como...

- La interacción entre determinadas condiciones de trabajo y aspectos relacionados con las características de los trabajadores más allá del entorno laboral
- Se relacionan principalmente con la organización, el contenido del trabajo y la realización de la tarea
- Si son desfavorables, los efectos negativos afectan tanto a la salud del trabajador como de la empresa
- Todas las afirmaciones son correctas

Figura 7.7: Creación de tema: Tema creado

Creación plan de capacitación

Para la creación de un plan el usuario debe de acceder al apartado de gestión, concretamente a la pestaña referida a los planes de capacitación. El usuario pulsa en la opción de crear un plan, este se creará con un nombre genérico, y los demás campos vacíos, tal y como se puede comprobar en la figura 7.8.

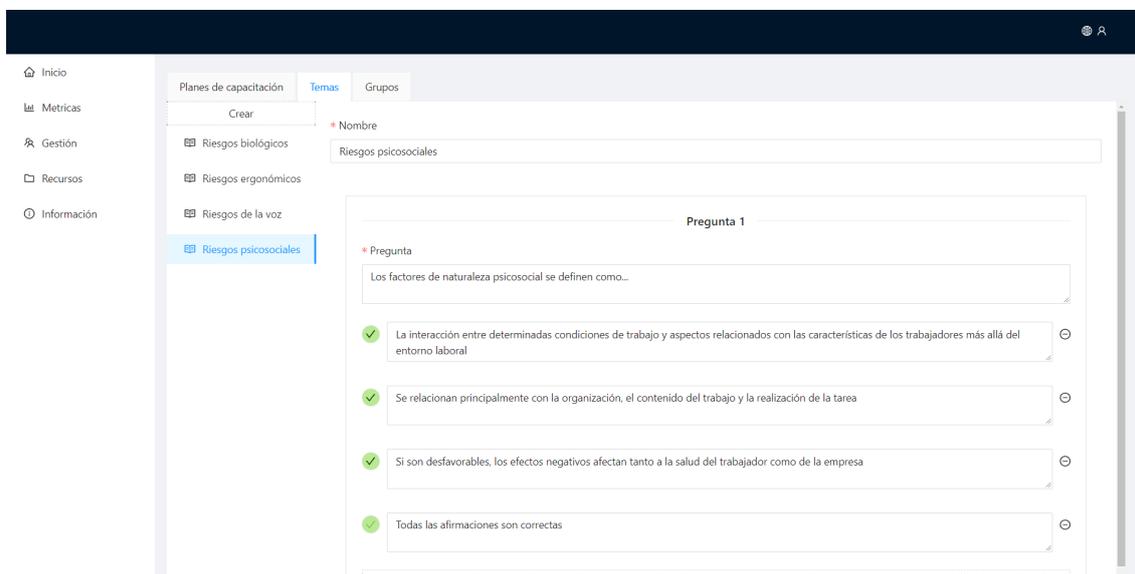


Figura 7.8: Creación de plan: Añadir plan vacío

El usuario añade la información referente al plan de capacitación, en este caso se ha añadido el plan “Plan capacitación 4”, añadiendo también el grupo creado anteriormente y dos temas como se puede observar en la figura 7.9 .

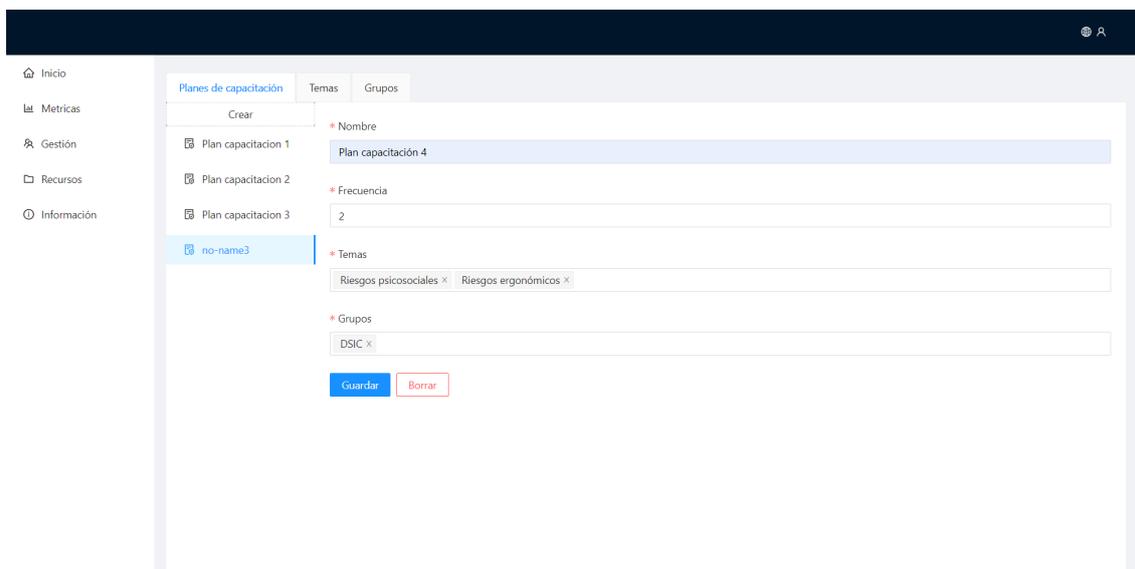


Figura 7.9: Creación de plan: Añadir información

Una vez la información ha sido introducida, y se hayan validado los campos, el sistema guarda la información en la base de datos y se añade el nombre elegido para el plan en el menú lateral, tal y como se puede observar en la figura 7.4.

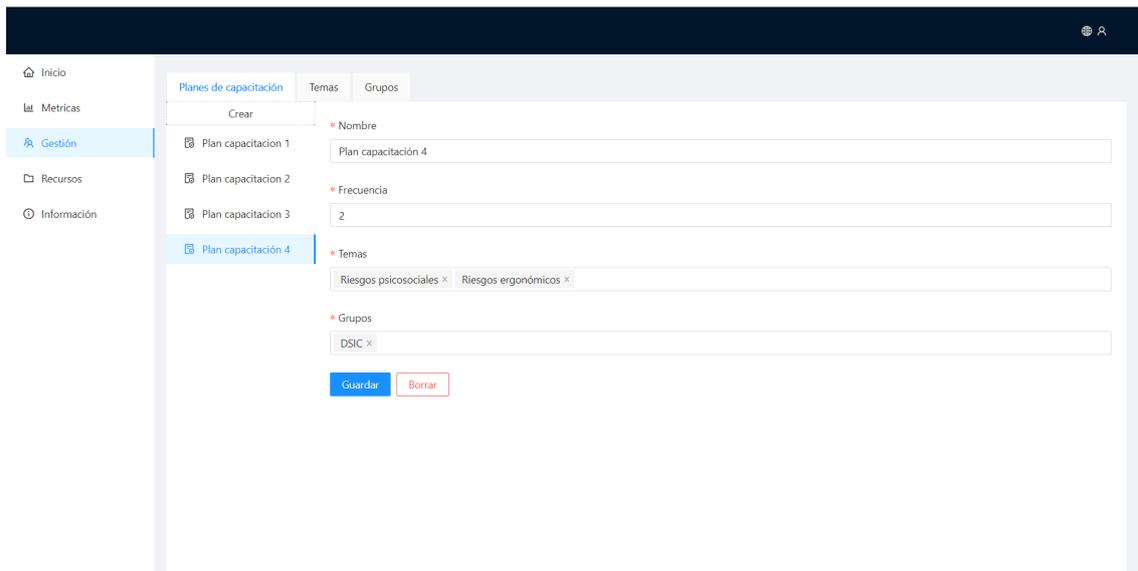


Figura 7.10: Creación de plan: Plan creado

Acceso aplicación móvil

Cómo ya se ha definido anteriormente en la memoria, los empleados con el rol *learner* serán aquellos que puedan acceder a la aplicación móvil, siendo la empresa la que les proporcione las credenciales para permitir su acceso, tal y como se puede observar en la figura 7.11.



Figura 7.11: Inicio de sesión: Aplicación móvil

Notificación y realización de test

Una vez el responsable de riesgos laborales cree el plan de capacitación, y cuando la frecuencia se dictamine, el empleado que esté asignado al grupo especificado en el

plan, recibirá una notificación para realizar el test. En la figura 7.12 se puede observar la notificación con información sobre la pregunta.

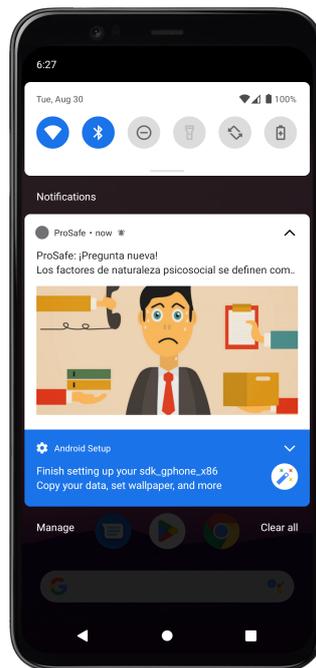


Figura 7.12: Notificación en dispositivo móvil Android

Justo a continuación, cuando el usuario pulse en la notificación, se abrirá la aplicación con la pregunta en cuestión, tal y como se puede observar en la figura 7.13.

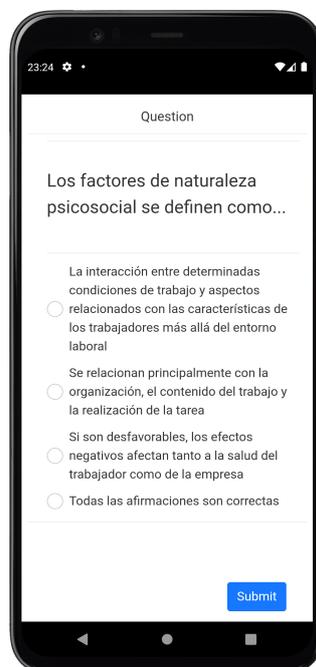


Figura 7.13: Pregunta asignada en dispositivo móvil Android

Visualización de estadísticas

El responsable de riesgos laborales puede acceder al apartado de estadísticas que ofrece el panel de Prosafe, mostrando las diferentes métricas halladas en base a las preguntas, los temas, los planes de capacitación y grupos creados en la organización. Siguiendo con el ejemplo, el usuario con el rol *learner* ha realizado el test sobre riesgos psicosociales, es por ello que en las figuras 7.14 y 7.15 se puede observar que el grupo “DSIC” cuenta solo con 100 puntos, ya que uno de los empleados que ha realizado la pregunta, la ha acertado. Además, este grupo solo se le han asignados los “riesgos psicosociales”, razón por la que solo existe un acierto y ningún fallo para este tema.

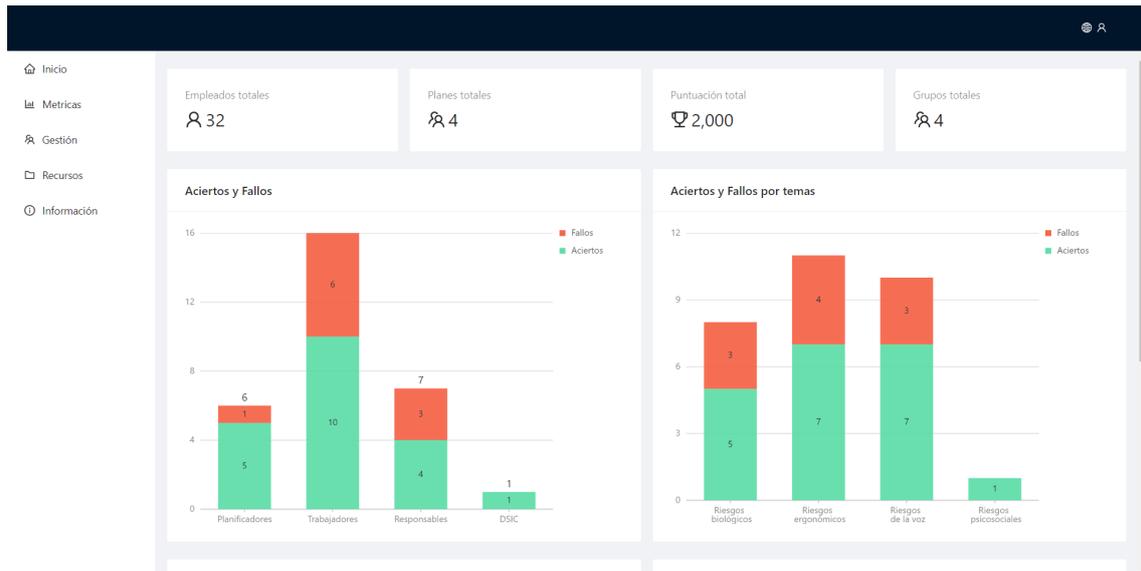


Figura 7.14: Pantalla de estadísticas (parte 1)

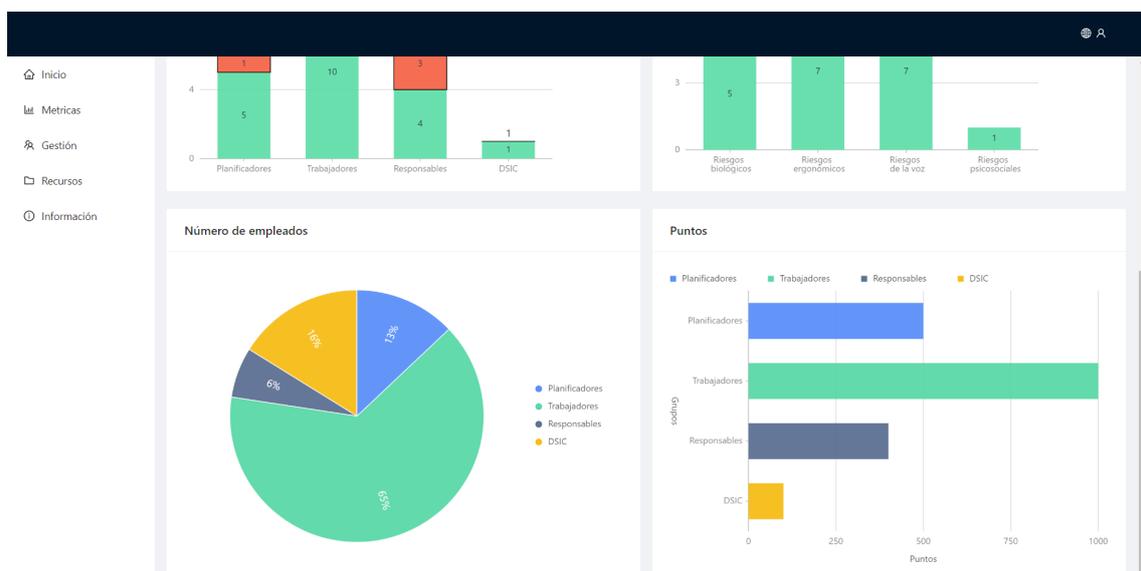


Figura 7.15: Pantalla de estadísticas (parte 2)

7.1.1. Cambio de idioma

El usuario tiene la posibilidad de cambiar de idioma por medio de un menú, tanto en la aplicación móvil como en la aplicación web. En la figura se puede observar el menú donde se encuentran los idiomas disponibles, en la figura 7.18 se puede observar cómo después de pulsar en la opción inglés, todos los campos de texto han cambiado al idioma establecido.

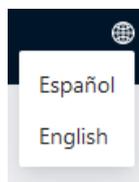


Figura 7.16: Menú de idiomas disponibles

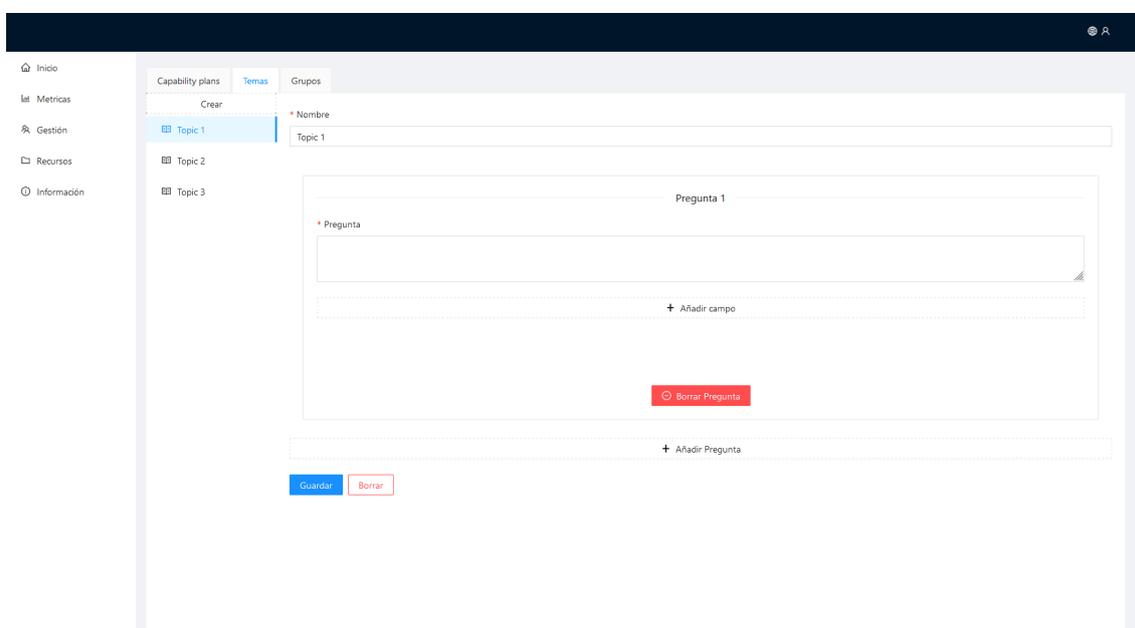


Figura 7.17: Página de creación temas en español

En la figura 7.18 se puede observar como los campos han cambiado después de pulsar la opción de idioma español.

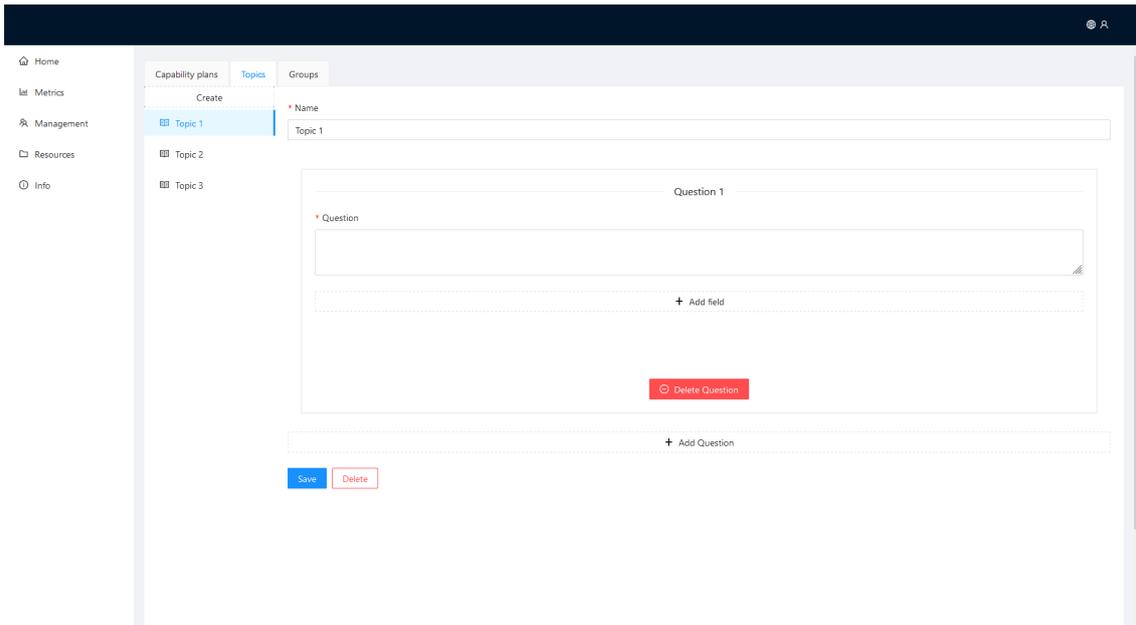


Figura 7.18: Página de creación temas en inglés

En cuanto a la aplicación móvil, el usuario puede cambiar esta información a través de los ajustes de la aplicación. En la figura 7.19 se puede observar un selector que permite alterar el idioma de la aplicación móvil.

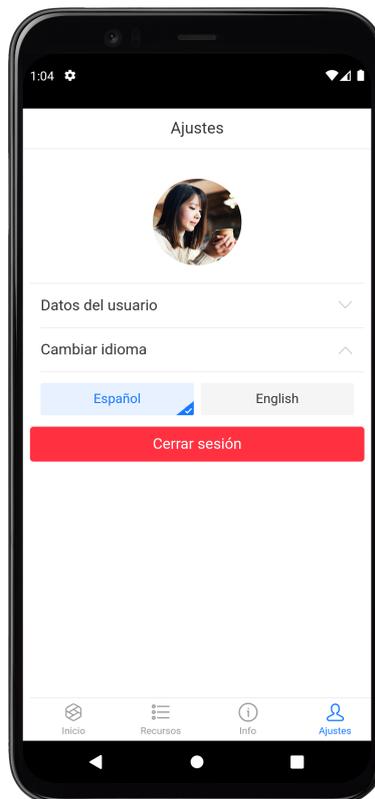


Figura 7.19: Selector de idiomas en la aplicación móvil

7.1.2. Visualización de información

Como ya se ha comentado en la especificación de requisitos, se necesita una página dedicada a la información referente a la aplicación ProSafe. En la figura 7.20 se puede observar la información para el portal web y la figura 7.21 para la aplicación móvil.

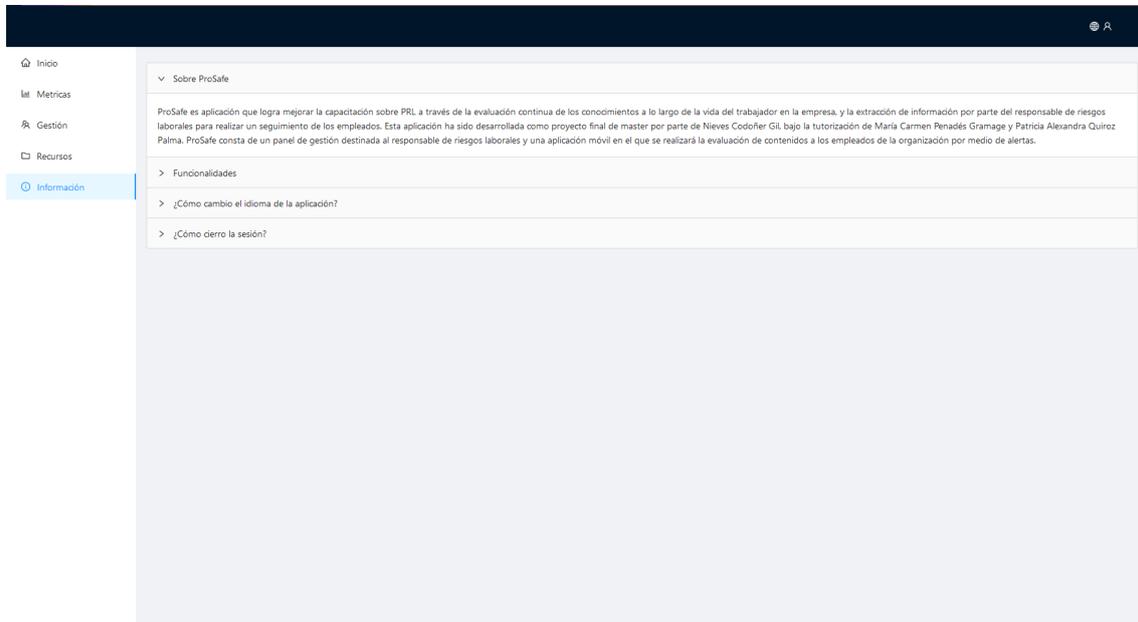


Figura 7.20: Visualización información de la aplicación web



Figura 7.21: Visualización información aplicación móvil

Por otro lado, se ha creado un panel desplegable lateral para mostrar la información referente a los datos personales del usuario que ha iniciado sesión en la plataforma, tal y como se puede observar en la figura 7.22.

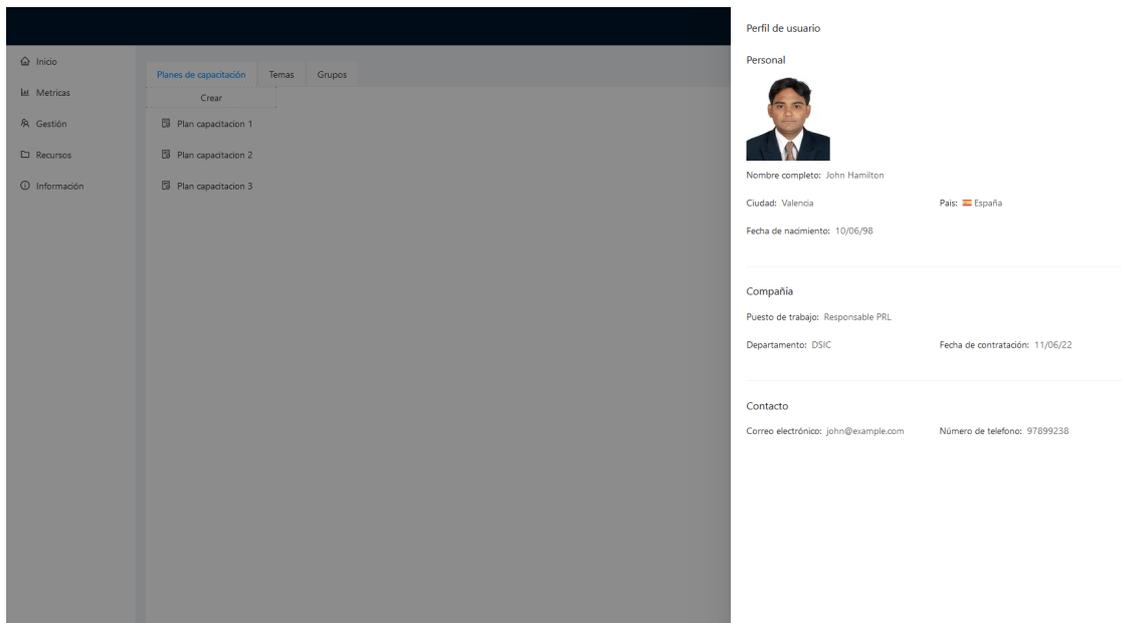


Figura 7.22: Visualización información de usuario aplicación web

Asimismo, en la aplicación móvil se puede observar como se muestra información referente a los datos del usuario en el apartado de "Ajustes", tal y como se puede observar en la figura 7.23.



Figura 7.23: Visualización información aplicación móvil

7.1.3. Visualización de recursos

En la aplicación web, concretamente en el apartado de recursos, se pueden observar una serie de documentos referidos tanto a recursos más generales de PRL, como más específicos, concretamente en el ámbito de una organización dónde la actividad predominante es la docente. En la figura 7.24 se puede observar la pantalla de recursos. Cabe destacar, que no se ha implementado la subida de recursos por parte del empleado con rol *trainer*, siendo esta tarea una posible extensión a futuro.

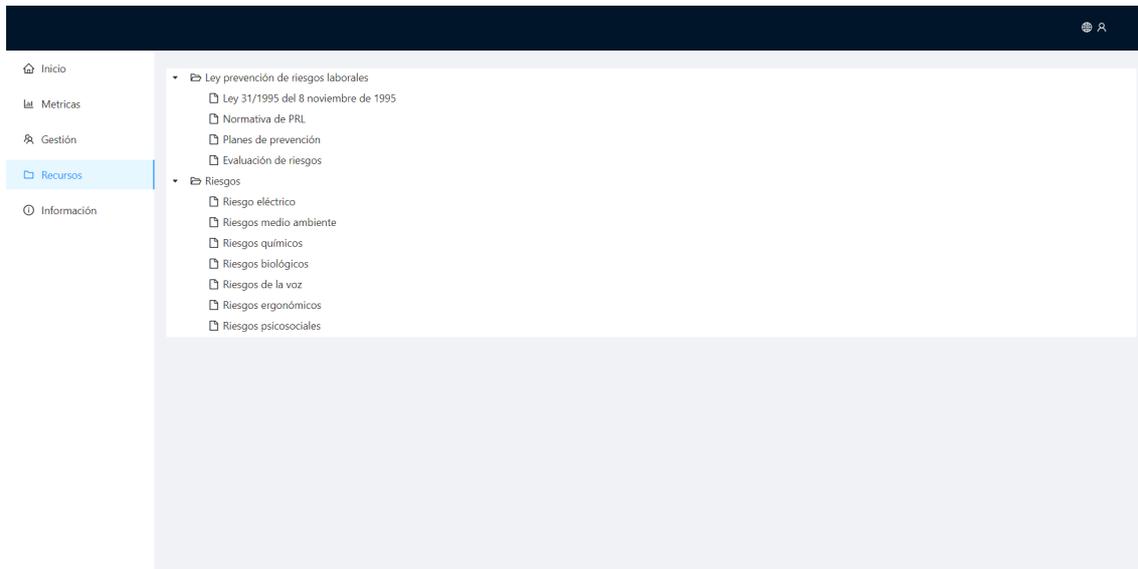


Figura 7.24: Visualización recursos (aplicación web)

Por otro lado, la aplicación móvil también cuenta con un apartado referente a los recursos. En la figura 7.24 se puede observar la pantalla de recursos sobre PRL en la aplicación móvil.



Figura 7.25: Visualización de recursos (aplicación móvil)

7.2 Otras validaciones

7.2.1. Validación de formularios

La validación de los formularios (lado del cliente) de una aplicación es de vital importancia, tanto para que los datos que coincidan con los requisitos establecidos, como para conseguir establecer más seguridad [29]. Esta validación se realizará en el lado del cliente, pero también contará con una validación en el lado del servidor. En los siguientes puntos se añaden algunas imágenes referentes a la validación de los formularios:

- **Campos vacíos:** Se comprueba que una vez el usuario intenta guardar el contenido de los planes, temas o grupos, se muestre por pantalla información con el objetivo de que el usuario añada la información que no ha establecido. En la figura 7.26 se puede observar este comportamiento para el formulario de planes de capacitación, pero esta validación se encuentra en todos los campos de los formularios de la aplicación.

Planes de capacitación | Temas | Grupos

Crear

- Plan capacitación 1
- Plan capacitación 2
- Plan capacitación 3
- no-name3

* Nombre
Este campo es obligatorio.

* Frecuencia
Este campo es obligatorio.

* Temas
Este campo es obligatorio.

* Grupos
Este campo es obligatorio.

Guardar Borrar

Figura 7.26: Campos requeridos: Creación plan de capacitación

- **Formato del campo:** Se comprueba que el usuario introduce los datos con el formato adecuado. En la figura 7.27 se puede observar la comprobación de que el usuario ha introducido un correo electrónico válido y de que el campo de contraseña requiere de 6 caracteres mínimo.

ProSafe
La seguridad es importante

example

Debe de proporcionar un email valido

La contraseña debe contener al menos 6 caracteres

Recordarme

Iniciar sesión

Figura 7.27: Formato del campo: Correo electrónico y contraseña

- **Campos requeridos mínimos:** Se comprueba que el usuario introduce o activa ciertas opciones. En la figura 7.28 se puede observar la comprobación de que el usuario ha añadido mínimo dos opciones para una pregunta y que haya seleccionado una como correcta.

Pregunta 1

* Pregunta

¿Esto es un ejemplo?

✓ option text

Este campo es obligatorio.

+ Añadir campo

Al menos 2 opciones

Es necesario especificar una opción correcta

Borrar Pregunta

Figura 7.28: Campos requeridos mínimos: Opciones en preguntas

En la figura se puede observar como para crear un grupo manual se necesitan al menos dos empleados para ser creado. Por otro lado, en la figura se muestra como para la creación de un grupo por criterios, al menos el usuario ha tenido que establecer un criterio mínimo.

7.2.2. Cuadro de diálogo

Para la gestión de los recursos de la aplicación se ha hecho uso de diálogos de confirmación para la ayuda del usuario que la maneje. En esta aplicación existen dos tipos de diálogos, en los siguientes puntos se muestran:

- **Dialogo de información:** Muestra información relevante para el usuario que está utilizando la aplicación. Este tipo de dialogo se utiliza para dar información sobre el estado de los formularios, cada vez que el usuario vaya a actualizar o crear algún tipo de recurso en la aplicación se le informará de si ha sido gestionado con éxito. En la figura 7.29 se puede observar un dialogo de éxito a la hora de la creación o actualización de un plan de capacitación, por el contrario, en la figura 7.30 se puede observar un cuadro que muestra el fracaso en la creación de dicho plan (los motivos de este aparecen como información de error en los campos del formulario).

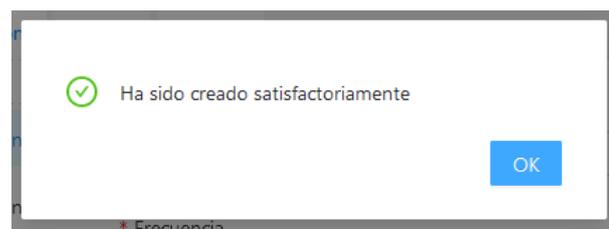


Figura 7.29: Dialogo de información: Creación plan de capacitación con éxito



Figura 7.30: Dialogo de información: Creación plan de capacitación fracasado

- **Dialogo de confirmación:** Muestra acciones que puede realizar el usuario, estos diálogos sirven para confirmar el propósito del usuario. En la aplicación desarrollada se utilizan para confirmar la eliminación de recursos y el cambio de actualización en la aplicación. En la figura 7.31 se puede observar un dialogo de confirmación para la eliminación de una pregunta.

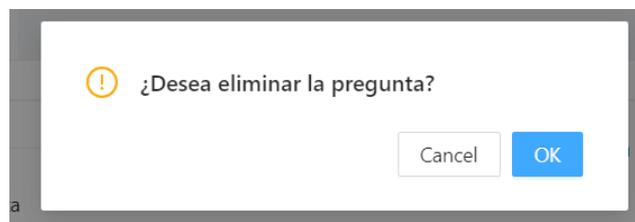


Figura 7.31: Dialogo de confirmación: Eliminar pregunta

CAPÍTULO 8

Conclusiones y trabajos futuros

En el presente capítulo se van a presentar las conclusiones obtenidas tras la realización del proyecto, indicando si se han cumplido con los objetivos propuestos y, dando una opinión sobre cómo ha sido el proceso de desarrollo. Por último, se detallarán una serie de funcionalidades que podrían resultar de utilidad para la mejora del proceso de capacitación que ofrece ProSafe.

8.1 Conclusiones

Cabe señalar que **los objetivos planteados en el capítulo 1.2 han sido cumplidos de manera satisfactoria**, pudiendo crear una aplicación tanto web como móvil que parta de los conceptos definidos por el modelo CiET y que sirva como base para ayudar en la mejora de la capacitación de los empleados de una organización. De la misma forma, **se ha profundizado en algunas tecnologías** con las que ya se contaba con algún tipo de conocimiento previo, pero algunas otras que se partía de cero, suponiendo un reto cumplir con todos los requisitos planteados.

En lo que se refiere a **los problemas que han surgido en el desarrollo del proyecto**, cabría destacar la **inexperiencia con algunas tecnologías y el tiempo invertido en el aprendizaje y elección de las mismas**. Actualmente, existen un número elevado de *frameworks* y herramientas, que a menudo pueden resultar beneficiosas para el desarrollo, sin embargo la elección de estas debe de ser de forma meticulosa, invirtiendo un periodo de tiempo en investigar y probar las diferentes posibilidades que ofrecen.

Además, al haber cursado en el grado la rama de ingeniería de computadores, **este máster y el presente proyecto me ha ayudado a asentar algunas de las bases de la ingeniería de software**, que si bien conocía algunos temas básicos, contaba con grandes lagunas, consiguiendo profundizar y tener conceptos más claros.

Por otro lado, **partir de un modelo ya establecido ha ayudado a asentar las bases del proyecto**, consiguiendo una aplicación que se basa en los conceptos definidos por el modelo ya validado por la directora experimental de este proyecto, Patricia Alexandra Quiroz Palma.

8.2 Valoración personal

En cuanto a mi valoración personal, creo que la idea del proyecto es muy acertada. Gracias a la asignatura Sistemas de gestión de emergencias y a contrastar opiniones con compañeros y conocidos, se ha podido concretar que existen muchas lagunas en términos

de prevención en riesgos laborales, no en cuanto a las medidas aplicadas, si no en la formación de los participantes. Es por ello, que surgió la idea de crear un software que pudiese ayudar en la capacitación continua de los empleados. Si bien es cierto que cabría añadir algunos conceptos a la hora de utilizarla en un ámbito profesional, además de incorporar alguna que otra funcionalidad para conseguir mejorar la capacitación, creo que ha sido desarrollada en un tiempo muy limitado, y esas características extras no tenían cabida en esa limitación de tiempo, dejándolas en un segundo plano y centrándose en las funcionalidades básicas que debería de tener el proyecto.

En lo que respecta al **crecimiento personal que ha supuesto este trabajo**, creo que cabría mencionar que el proyecto no solo me ha supuesto el **aprendizaje en diferentes tecnologías, estructuras y patrones**, si no que también me ha ayudado a **conseguir progresar en mis habilidades sobre la gestión eficiente del tiempo y el control de los niveles de estrés** que me ha podido suponer su realización. Del mismo modo, me ha ayudado a **conocer aspectos que no conocía sobre el tema de la PRL**, adentrándome en la legislación y proporcionándome conceptos e ideas carentes en mi.

8.3 Trabajos futuros

Adicionalmente, con el transcurso del desarrollo del proyecto han surgido nuevas ideas de implementación que podrían resultar interesantes como trabajos futuros, pero que no se encuentran entre los requisitos establecidos para el proyecto. En los siguientes puntos se detallan algunas de las funcionalidades propuestas:

- Creación de un **espacio específico en el portal web que permita al *trainer* crear contenidos/recursos** propios sobre un tema que elegido de la prevención de riesgos laborales de la organización.
- **Extensión en las estadísticas de la aplicación**, añadiendo gráficas flexibles que puedan ser manipuladas en tiempo real.
- **Extensión y mejora en el sistema de alertas de la aplicación móvil** para que permita notificar a los empleados de una posible emergencia que esté ocurriendo en la organización.
- **Extensión de la aplicación móvil** con el propósito de que sirva como **una vía de comunicación entre el encargado de los recursos laborales y el empleado**.
- **Extensión de la aplicación para añadir recompensas o beneficios** de la empresa a través de los puntos obtenidos por la contestación correcta de los tests.

Bibliografía

- [1] Jefatura del Estado. *Ley 31/1995, de 8 de noviembre, de prevención de Riesgos Laborales publicado en el «BOE» núm. 269, de 10/11/1995 y entrada en vigor el 10/02/1996*. URL: <https://www.boe.es/eli/es/l/1995/11/08/31/con> (visitado 19-06-2022).
- [2] Patricia Quiroz-Palma, M. Carmen Penadés y Ana-Gabriela Núñez. «Towards a Capability Model for Emergency Training Improvement». En: *Proceedings of the 16th International Conference on Information Systems for Crisis Response and Management, València, Spain, May 19-22, 2019*. Ed. por Zeno Franco, José J. González y José H. Canós. ISCRAM Association, 2019. URL: http://idl.iscram.org/files/patriciaquiroz-palma/2019/1945%5C_PatriciaQuiroz-Palma%5C_etal2019.pdf.
- [3] Mike McCormick. «Waterfall vs. Agile methodology». En: *MPCS, N/A 3* (2012).
- [4] Dirección provincial de Melilla. *Manual de información para personal docente*. URL: <https://www.educacionyfp.gob.es/dam/jcr:49a368e9-3779-4b05-b989-96802436a1a3/manualinformaciondocentes.pdf> (visitado 19-07-2022).
- [5] INSST. *App y cuestionario: Evaluación de trabajo repetitivo. OCRACheck - Año 2021*. URL: <https://www.insst.es/documentacion/catalogo-de-publicaciones/evaluacion-de-trabajo-repetitivo-ocracheck> (visitado 03-08-2022).
- [6] Fasabe-Team. *Prevención de Riesgos Laborales*. URL: <https://play.google.com/store/apps/details?id=com.fasabeteam.riesgoslaborales&hl=es&gl=US> (visitado 30-06-2022).
- [7] Patricia Alexandra Quiroz-Palma, Ma Carmen Penadés y Ana Gabriela Núñez. «Usabilidad de herramientas HCI que mejoran la capacitación en la gestión de emergencias». En: *Informática y Sistemas: Revista de Tecnologías de la Informática y las Comunicaciones 4.1* (2020), págs. 12-27.
- [8] Alberto Rausell Manchón. «Herramienta de soporte a los planes de capacitación en la gestión de emergencias». Tesis doct. Universitat Politècnica de València, 2021.
- [9] MDN web docs. *HTTP authentication*. URL: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Authentication> (visitado 02-09-2022).
- [10] Webstorm. *Funcionalidades*. URL: <https://www.jetbrains.com/es-es/webstorm/features/> (visitado 20-08-2022).
- [11] JetBrains. *WebStorm: Funcionalidades*. URL: <https://www.jetbrains.com/es-es/webstorm/features/> (visitado 23-06-2022).
- [12] IntelliJ IDEA. *¿Qué es IntelliJ IDEA?* URL: <https://www.jetbrains.com/es-es/idea/features/> (visitado 20-08-2022).
- [13] Git-scm.com. *About*. URL: <https://git-scm.com/about> (visitado 20-07-2022).
- [14] Figma. *Design features*. URL: <https://www.figma.com/design/> (visitado 23-06-2022).
- [15] Notion. *About*. URL: <https://www.notion.so/about> (visitado 30-06-2022).

-
- [16] Trello. *Tour*. URL: <https://trello.com/es/tour> (visitado 20-07-2022).
- [17] Facebook. *React*. URL: <https://es.reactjs.org/> (visitado 20-08-2022).
- [18] Alipay. *Ant Design*. URL: <https://ant.design/> (visitado 20-08-2022).
- [19] Alipay. *Ant Design Mobile*. URL: <https://mobile.ant.design/> (visitado 20-08-2022).
- [20] i18next. *Introduction*. URL: <https://www.i18next.com/> (visitado 20-08-2022).
- [21] Axios. *Getting Started*. URL: <https://axios-http.com/docs/intro> (visitado 20-08-2022).
- [22] Capacitor. *Introduction*. URL: <https://capacitorjs.com/docs> (visitado 05-07-2022).
- [23] Apache Software. *Why Spring?* URL: <https://spring.io/why-spring> (visitado 20-08-2022).
- [24] Apache Software. *Spring Boot*. URL: <https://spring.io/projects/spring-boot> (visitado 20-08-2022).
- [25] Wikipedia. *MySQL*. URL: <https://es.wikipedia.org/wiki/MySQL> (visitado 06-09-2022).
- [26] Facebook. *Create React App*. URL: <https://create-react-app.dev/> (visitado 19-08-2022).
- [27] Ant Design. *Ant Design Charts*. URL: <https://charts.ant.design/en> (visitado 10-09-2022).
- [28] Wikipedia. *Message broker*. URL: https://en.wikipedia.org/wiki/Message_broker (visitado 05-09-2022).
- [29] Mdn web docs. *Validación de formularios de datos*. URL: https://developer.mozilla.org/es/docs/Learn/Forms/Form_validation (visitado 24-08-2022).

APÉNDICE A

Casos de uso

- **Iniciar sesión:** Los empleados podrán iniciar sesión en la aplicación. En la tabla [A.1](#) se puede observar la definición más detallada del caso de uso.

Iniciar sesión	
Actores	Trainer y Learner
Resumen	El usuario puede acceder a la plataforma por medio de las credenciales ofrecidas por parte de la organización en la que trabaja.
Precondición	El usuario debe contar con una cuenta previamente.
Postcondición	El usuario iniciará sesión en la plataforma
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario introduce sus credenciales.	
2. El usuario <i>clica</i> en el botón de “Inicio de sesión”.	3. El sistema valida los datos.
	4. El sistema <i>loguea</i> al usuario.
	5. El sistema muestra la aplicación para el usuario.

Tabla A.1: Caso de uso: Iniciar sesión.

- **Cerrar sesión:** El usuario puede *deslogearse* de la aplicación. En la tabla [A.2](#) se puede observar la definición más detallada del caso de uso.

Cerrar sesión	
Actores	Trainer y Learner
Resumen	El usuario <i>clica</i> en el botón de cerrar sesión .
Precondición	El usuario debe de estar identificado.
Postcondición	El usuario no estará identificado.
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clica</i> en el botón de “Cerrar sesión”.	2. El sistema <i>desloguea</i> al usuario.

Tabla A.2: Caso de uso: Cerrar sesión.

- **Ver información del usuario:** Visualizar la información principal del usuario. En la tabla [A.3](#) se puede observar la definición más detallada del caso de uso.

Ver información del usuario	
Actores	Trainer y Learner.
Resumen	El usuario visualiza su información de usuario.
Precondición	El usuario debe estar registrado y identificado en la aplicación.
Postcondición	El usuario visualizará su información.
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clicka</i> en el icono que representa el perfil de cuenta del menu superior.	2. El sistema muestra una pantalla donde se muestran una serie de campos referentes a la información del usuario.

Tabla A.3: Caso de uso: Ver información del usuario

- **Ver información sobre la aplicación:** Visualizar la información principal de la aplicación. En la tabla A.4 se puede observar la definición más detallada del caso de uso.

Ver información sobre la aplicación	
Actores	Trainer y Learner.
Resumen	El usuario visualiza su información de usuario.
Precondición	El usuario debe estar registrado y identificado en la aplicación.
Postcondición	El usuario visualizará su información.
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clicka</i> en el icono de exclamación.	2. El sistema muestra una pantalla donde se muestra la información o guía que da la información.

Tabla A.4: Caso de uso: Ver información sobre la aplicación

- **Visualizar recursos :** Los empleados podrán visualizar temario de recursos sobre PRL. En la tabla A.5 se puede observar la definición más detallada del caso de uso.

Visualizar recursos	
Actores	Trainer y Learner
Resumen	El usuario puede visualizar el contenido referente a PRL en texto plano
Precondición	El usuario debe de estar logueado.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario accede al apartado de recursos de la aplicación.	
2. El usuario <i>clicka</i> en el recurso de su preferencia.	
	4. El sistema muestra al usuario los contenidos del recurso que ha pulsado.

Tabla A.5: Caso de uso: Visualizar recursos.

- **Visualizar métricas:** El usuario puede visualizar las gráficas referentes a los datos extraídos de la evaluación de los contenidos por medio de las preguntas por parte

del *learner*. En la tabla A.6 se puede observar la definición más detallada del caso de uso.

Visualizar métricas	
Actores	Trainer
Resumen	El usuario visualiza las métricas referentes a la evaluación realizada por los <i>learners</i> .
Precondición	El usuario debe de estar identificado.
Postcondición	
Flujo de eventos	
<i>Actor</i>	<i>Sistema</i>
1. El usuario <i>clica</i> en el enlace del menú referente a las métricas “Métricas”.	2. El sistema muestra las métricas sobre las preguntas y los grupos que las han realizado.

Tabla A.6: Caso de uso: Visualizar métricas.