



Universidad  
Politécnica  
de Cartagena

# UNIVERSIDAD POLITÉCNICA DE CARTAGENA



**Diseño y desarrollo de plataforma web  
open-source para la iniciativa Europea  
Alliance Centres of Vocational Excellence**

TRABAJO FIN DE MÁSTER

Máster Universitario en Ingeniería Telemática

**Autor: Ignacio Iglesias Castreño**

Directora: María Victoria Bueno Delgado

Codirectora: Cristina Albaladejo Pérez

Cartagena, septiembre 2021

## Resumen

Este trabajo consiste en el desarrollo una plataforma web de código abierto de evaluación de competencias y habilidades, creación de itinerarios de aprendizaje, búsqueda de empleo y cursos de formación centrada en el sector de la madera y mueble. Esta herramienta dispone de tres perfiles de usuario en un único sistema: usuario que busca empleo y/o cursos de formación, proveedor de empleo (reclutador o empresa), y proveedor de recursos de aprendizaje. Estos tres usuarios interactúan entre sí de manera que los usuarios buscan ofertas de trabajo de las empresas y cursos de formación de los proveedores, las empresas buscan candidatos para sus ofertas de empleo y cursos de formación para sus empleados, y los proveedores de formación buscan las habilidades más demandadas por las empresas para elaborar sus recursos.

La herramienta utiliza machine learning para ofrecer, mediante algoritmos de recomendación, resultados personalizados basados en usuarios similares.

Esta plataforma web ha sido financiada por el proyecto europeo ALLVIEW (Alliance of Centres of Vocational Excellence in the Furniture and Wood sector [1]), donde la UPCT es partícipe del proyecto.

## Palabras clave

Desarrollo web, arquitecturas web, ESCO, evaluación de habilidades, sistemas de recomendación.

## Índice

1. Introducción .....	5
1.1 La clasificación ESCO .....	5
1.2 Estado del arte de plataformas similares .....	6
1.3 Objetivo del trabajo .....	8
2. Requisitos funcionales de la plataforma .....	10
2.1 Usuario básico.....	10
2.1.1 Perfil de usuario .....	11
2.1.2 Buscador de cursos .....	11
2.1.3 Valoración de cursos .....	12
2.1.4 Buscador de ofertas de empleo .....	12
2.1.5 Itinerarios de aprendizaje.....	13
2.1.6 Favoritos.....	13
2.2 Empresa.....	14
2.2.1 Perfil de la empresa .....	14
2.2.2 Publicación de puestos de empleo.....	14
2.2.3 Búsqueda de candidatos.....	15
2.2.4 Buscador de cursos .....	15
2.2.5 Valoración de cursos .....	16
2.2.6 Favoritos.....	16
2.3 Proveedor de formación .....	17
2.3.1 Perfil de usuario .....	17
2.3.2 Publicación de cursos .....	18
2.3.3 Verificación de realización de cursos .....	18
2.4 Funcionalidades comunes .....	19
2.4.1 Acceso/registro .....	19
2.4.2 Ajustes .....	20
2.4.3 Ayuda .....	20
2.4.4 Cierre de sesión.....	20

3 Implementación técnica de la plataforma .....	21
3.1 Requisitos no-funcionales .....	21
3.1.1 Privacidad y seguridad de datos .....	21
3.1.2 Hosting .....	21
3.1.3 Copias de seguridad .....	22
3.2 Análisis de arquitecturas y tecnologías web .....	22
3.2.1 Arquitecturas web .....	22
3.2.2 Tecnologías de Frontend .....	24
3.2.3 Tecnologías de Backend.....	26
3.3 Arquitecturas y tecnologías escogidas .....	30
3.3.1 Arquitectura web .....	30
3.3.2 Frontend .....	30
3.3.3 Backend.....	31
4. Resultados.....	34
5. Conclusiones .....	42
6. Bibliografía.....	43

# 1. Introducción

## 1.1 La clasificación ESCO

Seguir un estándar es esencial para que una plataforma pueda ser integrada en otras usando el mismo formato de comunicación. Esto es básico para relacionar datos de empleados, empresas y centros educativos. Ejemplos de esto son: emparejar candidatos con empleos basados en sus habilidades, sugerir formación a personas que quieren mejorar sus habilidades y recomendar candidatos a empresas con las habilidades requeridas. Para ello, es fundamental desarrollar una plataforma que use una nomenclatura estándar para las habilidades, competencias, cualificaciones y ocupaciones.

La clasificación ESCO (European Skills, Competences, Qualifications, and Occupations) es una clasificación multidioma de las habilidades, competencias, cualificaciones y ocupaciones soportadas por la Comisión Europea, donde ESCO describe, identifica y clasifica los elementos relevantes [2].

ESCO proporciona las herramientas y recursos necesarios para que desarrolladores puedan implementar esta clasificación de manera gratuita en plataformas de búsqueda de empleo, cursos de formación y evaluación de habilidades. Los datos están organizados en tres pilares: habilidades/ competencias, ocupaciones y cualificaciones. Cada elemento está disponible en veintisiete idiomas.



Figura 1: clasificación ESCO de ocupaciones



Figura 2: clasificación ESCO de habilidades y competencias

Como se muestra en la Fig.1, ESCO proporciona actualmente una clasificación de 2942 ocupaciones y en la Fig.2, 12485 habilidades/competencias diferentes.

La plataforma desarrollada hace uso de esta clasificación para la elaboración de perfiles de usuario, seguimiento de itinerarios de aprendizaje, búsqueda de empleo, búsqueda de cursos de formación, selección de candidatos y análisis de habilidades demandadas. Aunque ESCO es una clasificación genérica para todos los sectores, este trabajo se centra en los elementos relacionados con el sector particular del mueble y la madera.













## 1.2 Estado del arte de plataformas similares

Antes de comenzar con el desarrollo del trabajo, se realizó una exploración de diversas plataformas de análisis de habilidades y búsqueda de empleo, tanto comerciales como de proyectos Europeos, con el fin de conocer el estado de arte. Las plataformas analizadas debían disponer de las siguiente características para ser consideradas:

- Ofertas de trabajo o de cursos de formación
- Sistema de recomendaciones
- Recursos de aprendizaje procedentes de proveedores de formación

Tras aplicar los criterios de filtrado, se realizó un análisis a las siguientes: Coursera, LinkedIn, Infojobs, Indeed, Realise, Woodwize, Skill Panorama, NADINE, SkillLab, OpenSKIMR, CareerOneStop, y Career Service.

En el análisis se establecieron dieciséis características para elaborar una tabla comparativa entre plataformas. En particular, si la plataforma es de código abierto o cerrado; la disponibilidad de aplicación móvil; su presencia en redes sociales; la disponibilidad de varios idiomas; si se tratad de una plataforma centrada en usuarios, empresas y proveedores de formación; si está focalizada en un sector concreto o genérica; disponibilidad de valoraciones de usuarios; si contiene ofertas de empleo y cursos de formación; si realiza recomendaciones personalizadas usando Machine Learning o recomendaciones genéricas; si dispone de rutas de aprendizaje y la implementación de la clasificación ESCO.

												
<b>Open Source</b>	x	x	x	x	x	x	✓	✓	x	✓	x	x
<b>App available</b>	✓	✓	✓	✓	x	x	x	x	✓	x	✓	x
<b>Social Networks</b>	✓	✓	✓	x	✓	✓	✓	✓	✓	✓	✓	✓
<b>Several idioms</b>	✓	✓	✓	x	x	✓	x	✓	✓	x	✓	✓
<b>For people</b>	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	x
<b>For companies</b>	✓	✓	✓	✓	✓	✓	✓	x	x	✓	✓	✓
<b>For Training providers</b>	✓	x	✓	x	x	✓	✓	x	x	x	x	x
<b>Nonspecific sector</b>	✓	✓	✓	✓	✓	x	✓	x	✓	✓	✓	x
<b>Ranking by users</b>	✓	x	x	✓	x	x	x	x	x	x	x	x
<b>Offers employment</b>	x	✓	✓	✓	x	✓	x	x	x	x	✓	✓
<b>Offers training</b>	✓	✓	✓	x	✓	✓	✓	✓	x	x	✓	✓

<b>ML</b>	✓	✓	x	x	x	x	x	x	✓	x	x	x
<b>Recommendations</b>	✓	✓	✓	✓	x	✓	✓	✓	✓	✓	✓	x
<b>Learning paths</b>	✓	x	x	x	x	x	✓	x	✓	✓	✓	x
<b>ESCO classification</b>	x	✓	x	x	x	✓	✓	✓	✓	✓	x	x

Tabla 1: comparativa de plataformas de evaluación de habilidades

La Tabla 1 muestra los resultados del estudio realizado sobre las trece plataformas de análisis de habilidades y búsqueda de empleo. En la comparativa se observó que carecían de las siguientes características:

- Código open-source
- Plataforma que disponga las partes de aprendizaje y empleo, involucrando los tres tipos de usuarios (personas, proveedores de aprendizaje y empresas)
- Recomendaciones mediante Machine Learning
- Valoraciones realizadas por usuarios
- Elaboración de itinerarios de formación
- Datos basados en la clasificación ESCO

### 1.3 Objetivo del trabajo

En base a los resultados de la comparativa anterior, y en vista de las carencias observadas, se definió el objetivo de este trabajo: desarrollar una plataforma web de evaluación de competencias y habilidades, creación de itinerarios de aprendizaje, búsqueda de empleo y cursos de formación centrada en el sector de la madera y mueble dentro del proyecto europeo ALLVIEW (referencia 621192-EPP-1-2020-1-ES-EPPKA3-VET-COVE). Este proyecto es una cooperación que conecta a socios del sector de la madera y el mueble a nivel regional, nacional y europeo.

Como característica básica, debe disponer de tres grupos de usuarios: personas, empresas y proveedores de aprendizaje. Estos tres usuarios deben interactuar dentro de la plataforma: un usuario puede buscar ofertas de empleo publicadas por empresas y cursos de formación para adquirir ciertas habilidades; una empresa puede buscar



cursos de formación para sus empleados y para puestos de trabajo; un proveedor de recursos de aprendizaje puede buscar las habilidades más demandadas y también validar que un usuario ha realizado un determinado curso de formación.

Además, la plataforma deberá implementar un sistema de recomendación mediante Machine Learning para indicar ofertas de empleo o cursos a usuarios clasificados en perfiles similares.

En los siguientes apartados se definen los requisitos funcionales y no funcionales que debe cumplir la plataforma y los detalles de la implementación técnica.

## 2. Requisitos funcionales de la plataforma

En esta sección se revisan las funcionalidades requeridas para el desarrollo de la plataforma web ALLVIEW. Se han separado por cada tipo de usuario, comenzando por el usuario básico que es el que busca formación y ofertas de trabajo. Posteriormente se revisan las funcionalidades para las empresas, interesadas en candidatos para sus ofertas de empleo y formación para sus empleados. A continuación se tratan los usuarios proveedores de formación, que tienen como objetivo principal detectar las habilidades que demandan las empresas y los usuarios básicos. Finalmente se describen las funcionalidades comunes para todos.

En la mayoría de casos, estas funcionalidades se acompañan de un diagrama de casos de uso, implementados mediante el lenguaje de modelado UML (Unified Modeling Language).

### 2.1 Usuario básico

Este tipo de usuarios pueden ser trabajadores, demandantes de empleo o estudiantes que buscan adquirir nuevas habilidades. Dispondrá de un formulario de acceso/registro, una página con su perfil de usuario, un buscador de cursos de formación, un buscador de ofertas de empleo, y un visualizador de itinerarios de aprendizaje. La Fig. 3 muestra el diagrama de casos de uso de este usuario.

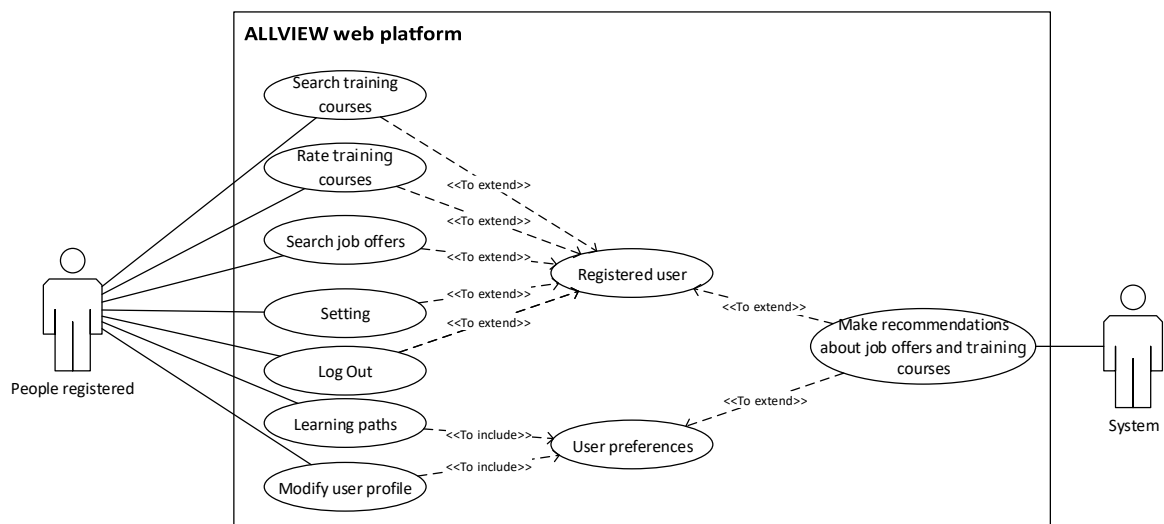


Figura 3: diagrama de casos de uso del usuario básico

### 2.1.1 Perfil de usuario

Como se ha indicado, la página de perfil del usuario básico deberá contener un formulario con datos personales tales como nombre, apellidos, número de teléfono, ciudad y foto de perfil. Estos datos son opcionales y se usarán para que las empresas puedan contactar con los solicitantes de empleo.

Además, el usuario debe poder seleccionar (siguiendo la clasificación ESCO) sus habilidades, competencias, experiencia laboral, estudios cursados y objetivos laborales. La elaboración de estos perfiles permitirá realizar recomendaciones basándose en usuarios similares y la elaboración de itinerarios de aprendizaje.

### 2.1.2 Buscador de cursos

Los usuarios podrán realizar búsquedas de los cursos disponibles en la plataforma, que hayan sido establecidos por los proveedores de formación o extraídos de manera automática. Estos cursos podrán ser filtrados mediante palabras clave, idiomas, habilidades y competencias. Si el usuario ha completado su perfil, recibirá recomendaciones de cursos de acuerdo con las recomendaciones establecidas para usuarios similares.

Los cursos resultantes de la búsqueda deben mostrar toda su información relevante, es decir, descripción, imagen, duración, valoraciones y sitio oficial. En la Fig. 4 se puede observar el diagrama de casos de uso de esta funcionalidad.

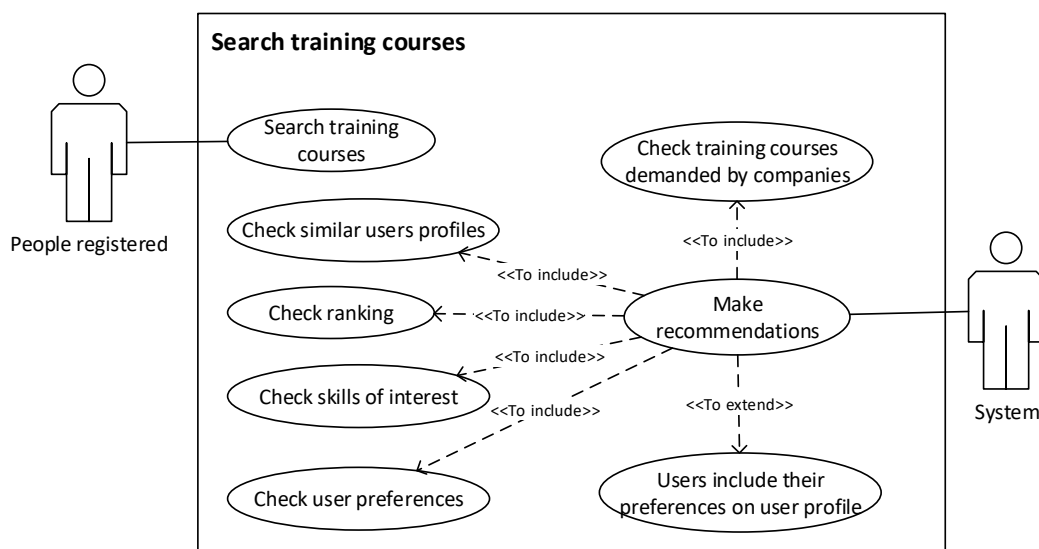


Figura 4: diagrama de casos de uso del buscador de cursos

### 2.1.3 Valoración de cursos

Los usuarios deben tener la posibilidad de valorar los cursos que han realizado con el objetivo de realizar un ranking de cursos dentro de la plataforma. Estas valoraciones consisten en un comentario personal y una puntuación entre una y cinco estrellas, siendo una estrella la peor puntuación y cinco estrellas la mayor. La Fig. 5 muestra los casos de uso de esta función de valoración.

Estas valoraciones no son publicadas directamente en la plataforma, sino que deberán ser revisadas por el proveedor de formación que publicó el curso. El objetivo de esta supervisión es evitar el spam y comprobar que el usuario realmente ha realizado ese curso de formación.

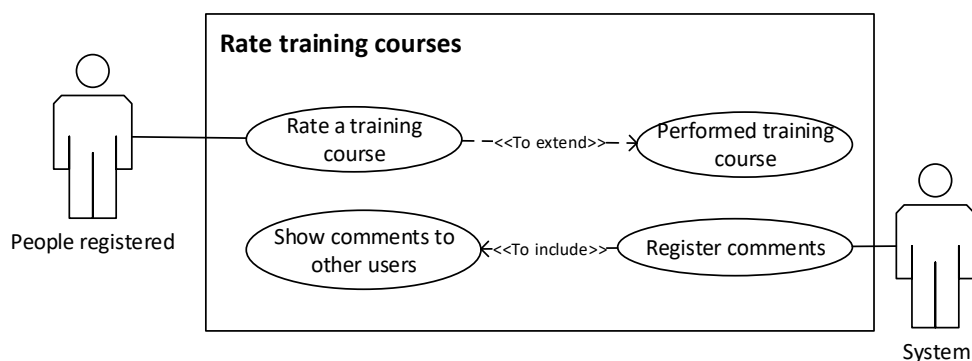


Figura 5: diagrama de casos de uso de la valoración de cursos

### 2.1.4 Buscador de ofertas de empleo

De manera similar al buscador de cursos de formación, los usuarios también deberán poder realizar búsquedas de las ofertas de empleo disponibles en la plataforma. Estas ofertas de empleo han sido determinadas por las empresas registradas en la plataforma o extraídas de manera automática de otras plataformas de socios del proyecto. Las ofertas de empleo podrán ser filtradas mediante palabras clave, habilidades, competencias y empresas. Si el usuario ha completado su perfil, recibirá recomendaciones de ofertas de empleo basadas en recomendaciones realizadas a usuarios similares. El diagrama de la Fig. 6 describe los casos de uso de este buscador.

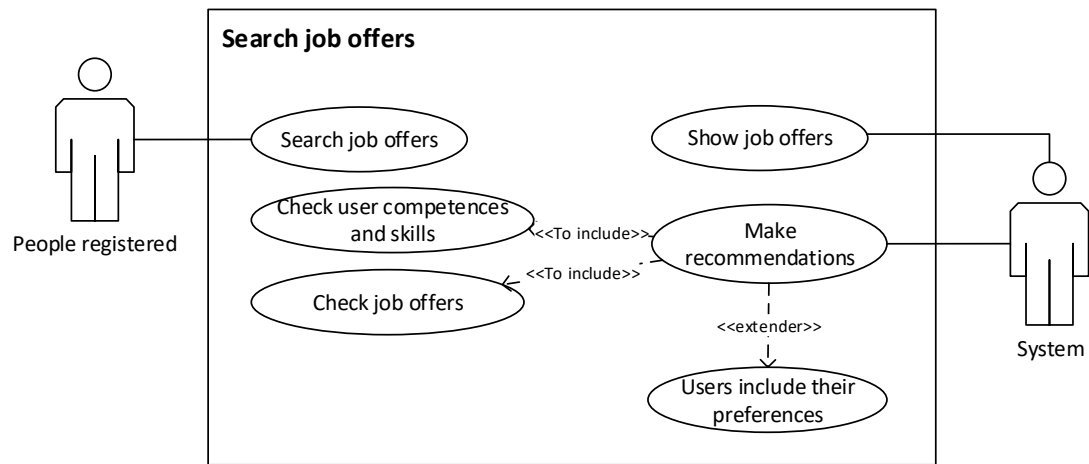


Figura 6: diagrama de casos de uso del buscador de ofertas de empleo

### 2.1.5 Itinerarios de aprendizaje

La plataforma deberá disponer de una funcionalidad para la elaboración de itinerarios de aprendizaje. Estos itinerarios muestran las habilidades y competencias que debe adquirir un usuario para alcanzar sus objetivos laborales. La plataforma deberá, a su vez, recomendar los cursos disponibles que permitan adquirir esas destrezas requeridas.

### 2.1.6 Favoritos

Los usuarios deben poder marcar cursos de formación y ofertas de empleo como favoritos. Estos cursos y ofertas laborales se guardarán en una página para este fin de manera que el usuario pueda acceder al listado directamente. El diagrama de la Fig. 7 muestra estos casos de uso.

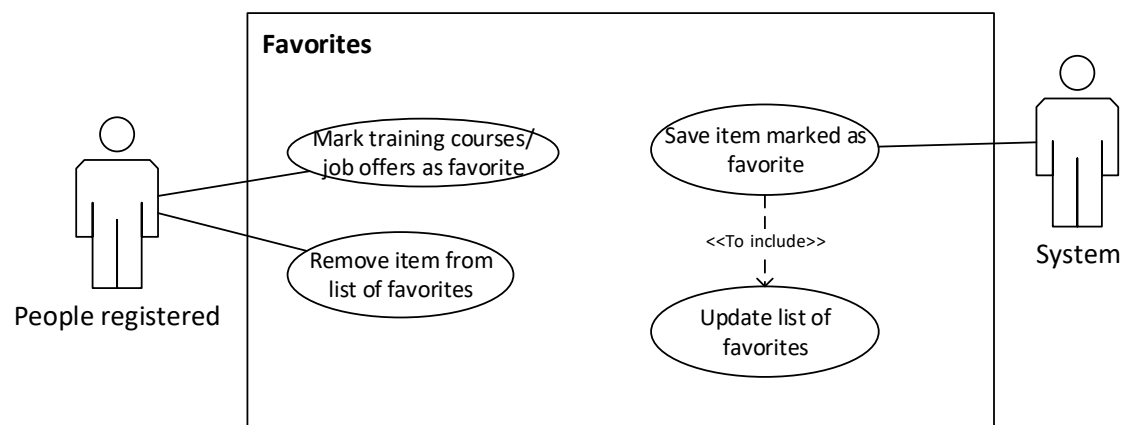


Figura 7: diagrama de casos de uso de favoritos

## 2.2 Empresa

Los usuarios clasificados como empresa son compañías que ofrecen puestos de trabajo y que buscan formar a sus trabajadores con cursos de formación.

Este tipo de usuario dispondrá de un formulario de acceso/registro, una página de perfil de la compañía, un formulario para publicar ofertas de empleo, un buscador de candidatos y un buscador de cursos de formación con valoraciones. En el diagrama de la Fig. 8 se muestra diagrama de casos de uso de este usuario

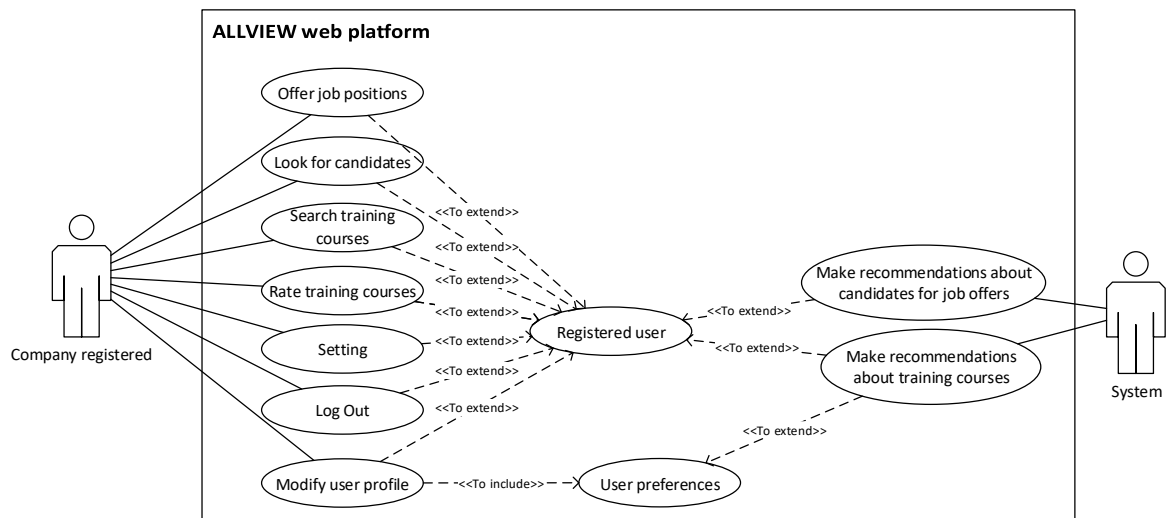


Figura 8: diagrama de casos de uso del usuario empresa

### 2.2.1 Perfil de la empresa

La página de perfil de la empresa contendrá información básica sobre la compañía, como el nombre, logo, identificación, página web, sector y ubicación. Estos datos podrán ser consultados por aquellos usuarios que busquen ofertas de empleo en la plataforma.

Estos datos podrán también ser utilizados para elaborar recomendaciones de cursos de formación entre empresas con características similares.

### 2.2.2 Publicación de puestos de empleo

Las empresas podrán crear y administrar publicaciones de puestos de trabajo en la plataforma. Estas publicaciones deberán incluir el título, descripción, habilidades y experiencia requeridas, sector, tipo de contrato, duración, salario y contacto. Las empresas deben poder ver una lista de sus ofertas de empleo, modificarlas y eliminarlas.

Además, podrán recibir sugerencias de candidatos para sus ofertas de empleo que estarán basadas en las características de las ofertas y en el perfil de los usuarios básicos. La Fig. 9 muestra los casos de uso de la publicación de ofertas de empleo.

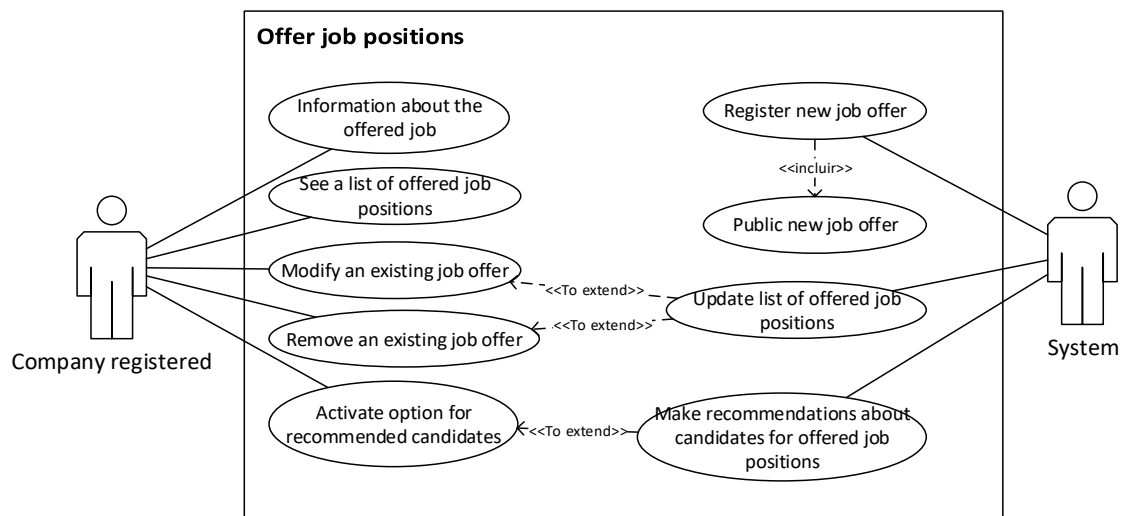


Figura 9: diagrama de casos de uso de la publicación de puestos de empleo

### 2.2.3 Búsqueda de candidatos

Las empresas deben poder realizar búsquedas en la base de datos de candidatos filtrando de acuerdo con determinadas habilidades, competencias y experiencia laboral. La plataforma devolverá una lista de usuarios que cumplen esos requisitos. Si estos usuarios tienen activada la opción de compartir información personal con las empresas, estas podrán ponerse en contacto directamente con los usuarios básicos.

### 2.2.4 Buscador de cursos

De manera similar a los usuarios básicos, las empresas podrán realizar búsquedas de los cursos disponibles en la plataforma para formar a sus empleados. Estos cursos podrán ser filtrados mediante palabras clave, idiomas, habilidades y competencias y se podrá ver información detallada sobre ellos.

La plataforma deberá además realizar recomendaciones sobre cursos ofrecidos por empresas con perfiles similares y en base a los cursos realizados anteriormente. En la Fig. 10 se detallan los casos de uso del buscador de cursos.

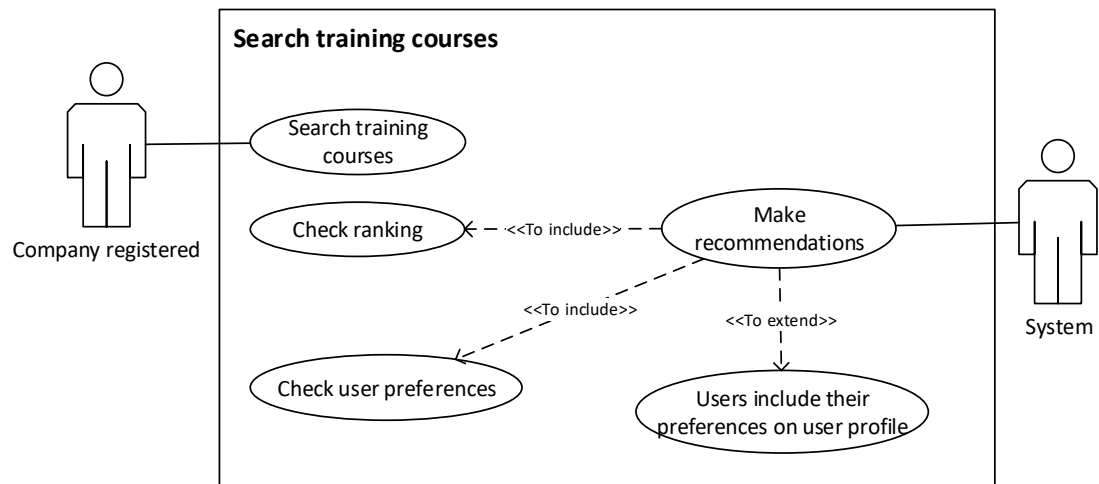


Figura 10: diagrama de casos de uso del buscador de cursos

### 2.2.5 Valoración de cursos

De igual manera que los usuarios básicos, las empresas deben tener la capacidad de valorar los cursos realizados con el objetivo de enriquecer la valoración de cursos en la plataforma. La Fig. 11 describe estos casos de uso. Al igual que con los usuarios básicos, estas valoraciones consisten en un comentario personal y una nota entre una y cinco estrellas, siendo una estrella la peor puntuación y cinco estrellas la mayor.

Estas valoraciones no son publicadas directamente en la plataforma, sino que deberán ser revisadas por el proveedor de formación que publicó el curso.

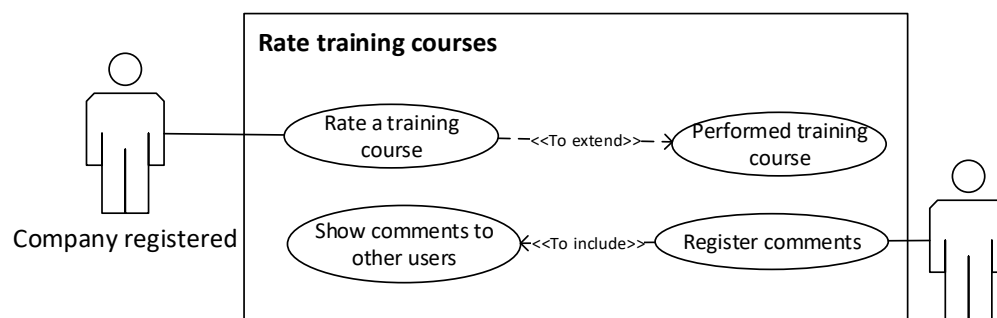


Figura 11: diagrama de casos de uso de la valoración de cursos

### 2.2.6 Favoritos

De la misma forma que con los usuarios básicos, las empresas deben poder marcar cursos de formación como favoritos que se guardarán en una página de favoritos de manera que el usuario pueda acceder a ellos directamente. En la Fig. 12 se muestran los casos de uso de esta funcionalidad.



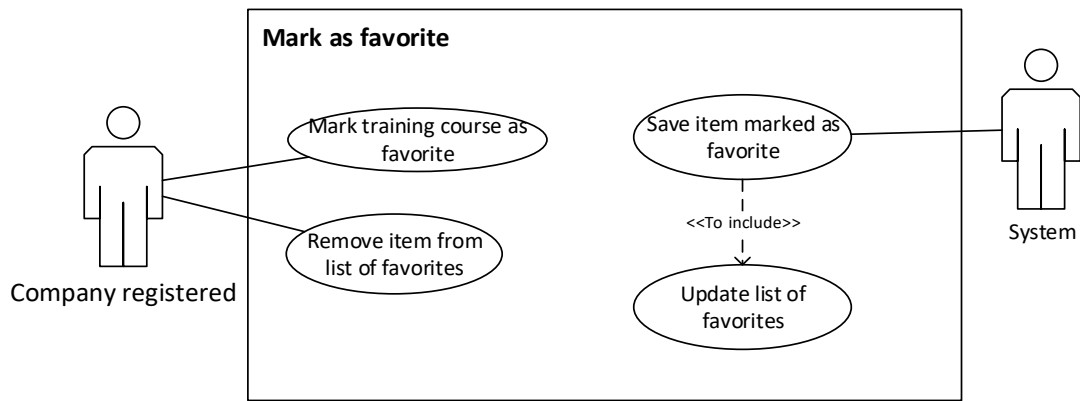


Figura 12: diagrama de casos de uso de favoritos

### 2.3 Proveedor de formación

Este tipo de usuarios está conformado por creadores de contenido educativo que buscan ofrecer sus recursos de formación e identificar las habilidades más demandadas para optimizar su material.

Los proveedores de formación dispondrán de un formulario de acceso/registro, una página de perfil, un formulario para publicar sus recursos educativos, una herramienta para revisar valoraciones y verificar la realización de cursos y un buscador de las habilidades más demandadas. El diagrama de la Fig. 13 representa estos casos de uso.

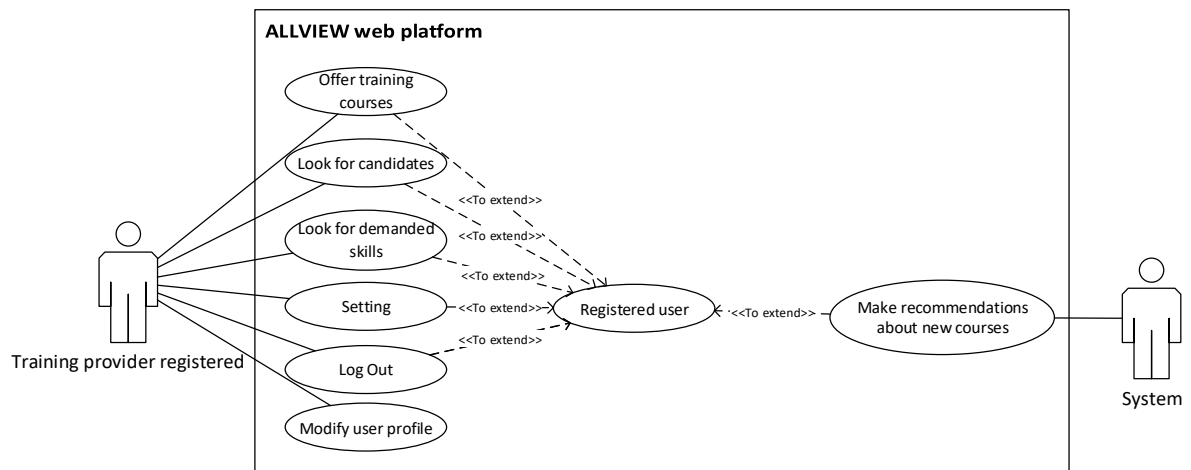


Figura 13: diagrama de casos de uso del usuario proveedor de formación

#### 2.3.1 Perfil de usuario

La página de perfil del proveedor de información contendrá información básica sobre la entidad proveedora de formación: nombre, logo, identificación, página web y sector.

### 2.3.2 Publicación de cursos

Los proveedores de formación podrán publicar sus recursos educativos a través de la plataforma, modificarlos y eliminarlos. La publicación de estos recursos se podrá realizar mediante la subida de un documento Excel estandarizado. Este documento deberá estar basado en una plantilla disponible en la plataforma. Alternativamente, se puede rellenar un formulario que contiene los mismos campos que el documento anterior directamente en la plataforma web.

El usuario debe poder ver todos los cursos publicados, además de modificarlos y tener la capacidad de eliminarlos de la plataforma. En el diagrama de la Fig. 14 se visualizan los casos de uso de esta funcionalidad.

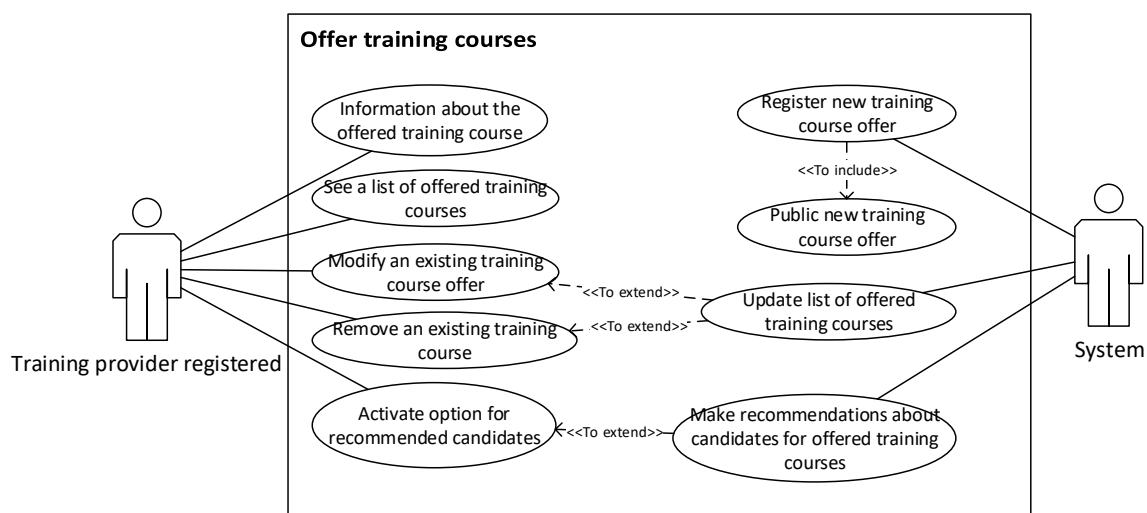


Figura 14. Diagrama de casos de uso de publicación de cursos

### 2.3.3 Verificación de realización de cursos

Una vez que los usuarios básicos o las empresas han finalizado un curso de aprendizaje, el proveedor de formación debe verificar que el curso ha sido realmente realizado. Para ello, los usuarios deberán valorar el curso y el proveedor de formación debe aceptar o rechazar las solicitudes de valoración. El diagrama de la Fig. 15 representa los casos de uso de esta función.

Si el proveedor de formación verifica la realización del curso, el alumno obtendrá esa característica en su perfil.

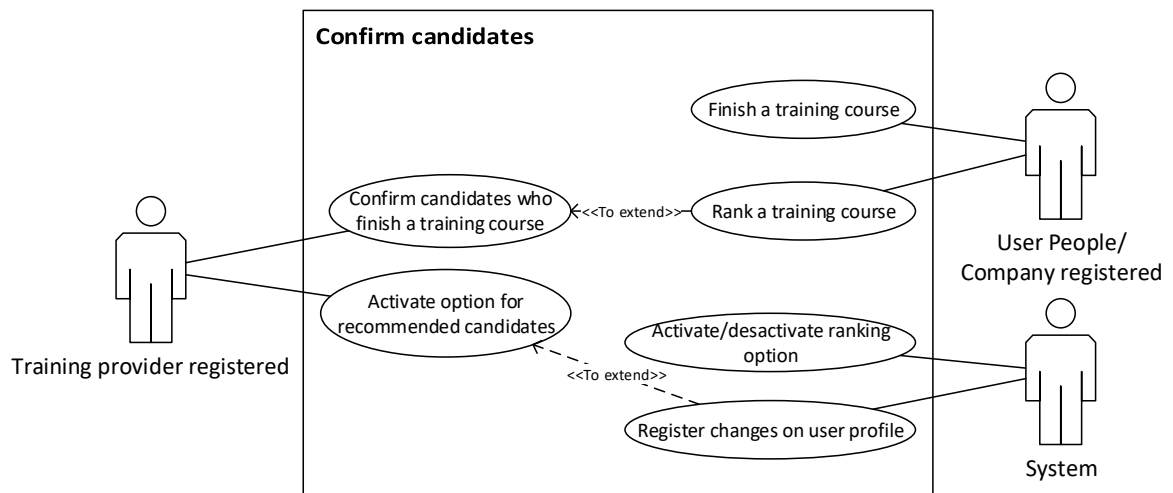


Figura 15: diagrama de casos de uso de la verificación de realización de cursos.

## 2.4 Funcionalidades comunes

### 2.4.1 Acceso/registro

Debe existir un formulario de creación de cuenta donde el usuario introduzca su correo electrónico, contraseña y tipo de usuario deseado (básico, proveedor de formación o empresa) para registrarse en la plataforma. Además, un usuario ya registrado debe poder acceder a la plataforma introduciendo correo electrónico y contraseña. Si el usuario ya estaba registrado pero no recuerda sus datos de acceso, podrá reestablecer su contraseña mediante un método de recuperación de cuenta mediante correo electrónico. La Fig. 16 describe estos casos de uso.

Para facilitar el proceso de acceso y registro, la plataforma deberá además permitir que los usuarios puedan usar su cuenta de Google para acceder a la plataforma.

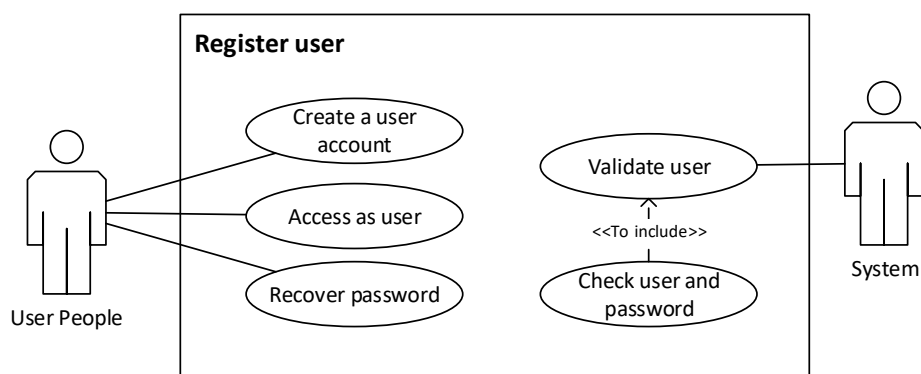


Figura 16: diagrama de casos de uso de registro en la plataforma

#### **2.4.2 Ajustes**

Todos los usuarios deberán disponer de una página de ajustes donde podrán cambiar su contraseña actual por una nueva y poder realizar un borrado de cuenta. Debido a que se tratan de operaciones delicadas, deberán ser validadas mediante un mensaje de correo electrónico de verificación.

#### **2.4.3 Ayuda**

La página de ayuda deberá contener una guía de uso de la plataforma por cada tipo de usuario, un listado de preguntas frecuentes y los datos de contacto.

#### **2.4.4 Cierre de sesión**

Mediante esta función, todos los usuarios podrán cerrar su sesión en la plataforma. Para volver a acceder a la aplicación, deberán volver a introducir sus datos de acceso.

## 3 Implementación técnica de la plataforma

En esta sección se definen los requisitos técnicos de la plataforma, se realiza un análisis de las arquitecturas y tecnologías disponibles hoy en día y finalmente se detallan las tecnologías usadas en el desarrollo de la plataforma y las razones de su elección.

### 3.1 Requisitos no-funcionales

Los requisitos no funcionales son aquellos centrados en cubrir lo que se espera del producto y sus propiedades.

#### 3.1.1 Privacidad y seguridad de datos

La plataforma deberá albergar datos confidenciales, como son los datos personales de los perfiles de usuarios, información de ofertas de trabajo y cursos. Por ello se debe prestar especial cuidado al manejo de estos datos. Esta información deberá ser guardada en bases de datos custodiadas por la UPCT, cumpliendo con las regulaciones de protección de datos europeas. La información almacenada en estas bases de datos solo podrá ser usada para los casos de uso explicados anteriormente.

La plataforma utiliza los algoritmos de cifrado más robustos disponibles actualmente. Las contraseñas se almacenan *hasheadas* mediante el algoritmo Bcrypt con un coste de dificultad de doce. La comunicación de las sesiones entre el *frontend* y *backend* se realiza mediante JWT usando el algoritmo SHA512 HMAC. Además, toda la información intercambiada en la plataforma se encuentra protegida mediante el uso de HTTPS.

#### 3.1.2 Hosting

La plataforma ALLVIEW debe garantizar un tiempo de respuesta correcto y un servicio ininterrumpido. También deberá estar en línea durante el desarrollo del proyecto y al finalizar. Actualmente la plataforma se encuentra desplegada en un servidor físico ubicado en el campus de la UPCT. En el futuro, cuando se disponga de una versión estable de la plataforma, se desplegará en un servidor en la nube acordado por todos los socios del proyecto para tener la máxima disponibilidad, siempre manteniendo altas medidas de seguridad.

### 3.1.3 Copias de seguridad

Los datos almacenados en la base de datos se deberán copiar todos los días en un servidor diferente. Otros datos menos importantes, como las imágenes, serán copiadas una vez por semana. El código de los componentes de la plataforma está organizado en diferentes repositorios y está almacenado en repositorios remotos de Git. Actualmente el código no es público pero será publicado en el lanzamiento de la plataforma en un repositorio de GitHub por determinar.

## 3.2 Análisis de arquitecturas y tecnologías web

Se realizó un estudio previo para elegir las tecnologías más adecuadas para el desarrollo de la plataforma ALLVIEW. Dentro de este estudio se prestó especial importancia a las tecnologías que permitieran un fácil mantenimiento de código, flexibilidad y modularidad y que fueran de código abierto. A continuación se detallan las diferentes arquitecturas web disponibles en la actualidad y las diferentes tecnologías de *frontend*, *backend* y *machine learning*.

### 3.2.1 Arquitecturas web

Las arquitecturas web consideradas relevantes para la plataforma ALLVIEW son las siguientes:

- Server-Side Rendering (SSR): el servidor genera el HTML completo de la página en cada navegación. El navegador recibe la página completa directamente y solo debe interpretarla. Por otro lado, toda la extracción de datos y lógica de la página se realiza en el servidor, de forma que la mayor carga de procesamiento recae en este. Esta arquitectura es idónea para obtener el mejor indexado en motores de búsqueda.

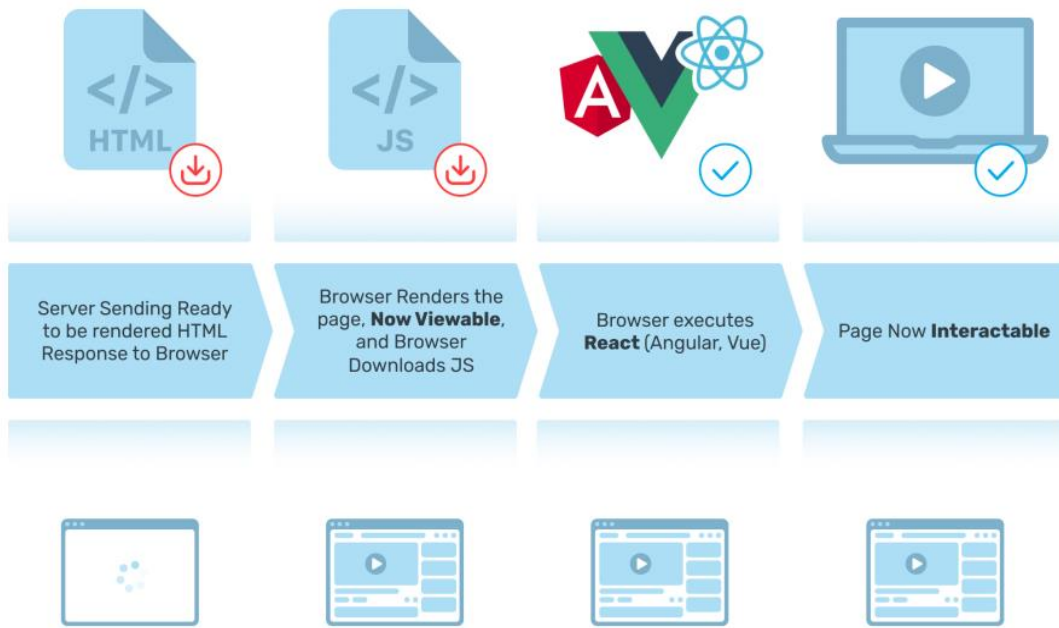


Figura 17: proceso de renderizado de una web SSR [3]

En la Fig. 17 se puede observar el proceso de renderizado de SSR. El servidor envía un HTML completo al cliente y este solo debe ejecutar JS para terminar de conformar los últimos detalles de la página.

- Client-Side Rendering (CSR): el renderizado de la página se realiza directamente en el navegador usando JavaScript. Toda la lógica de extracción de datos, funcionalidades de la web y enrutado se realiza en el cliente en lugar de en el servidor al contrario que en SSR. Las páginas Single Page Application (SPA) extraen los datos únicamente desde APIs REST. Este tipo de páginas tienen más problemas con la indexación en buscadores ya que los *crawlers* no obtienen la página completa directamente. Al recaer el renderizado de la página sobre los clientes, un único servidor puede servir a miles de usuarios simultáneos sin problema.

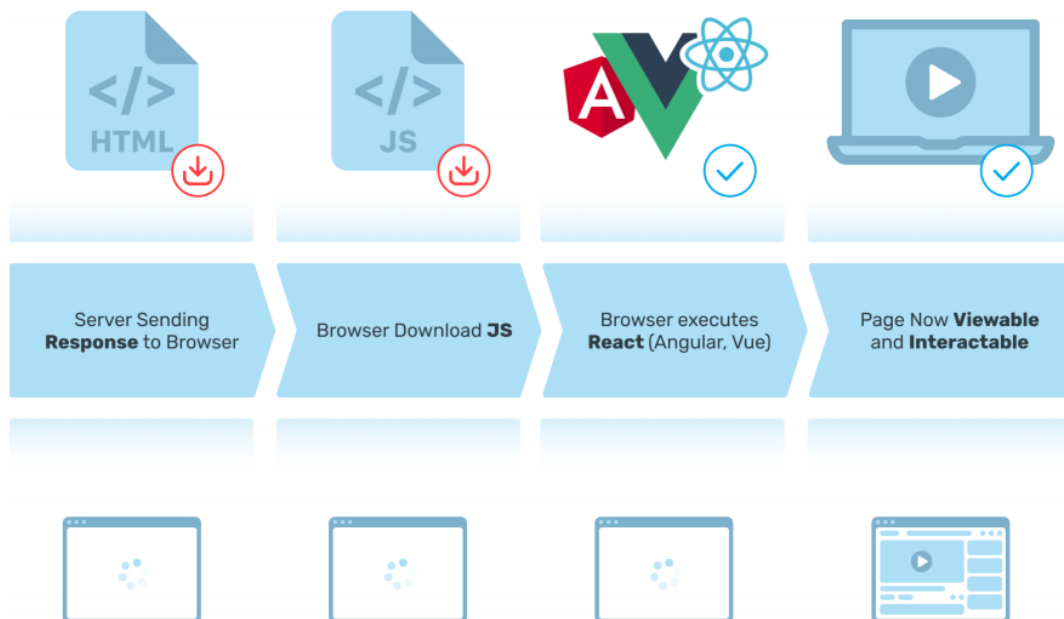


Figura 18: proceso de renderizado de una web CSR [3]

En la Fig. 18 se visualizan las fases de procesado de una web CSR. El servidor envía un código JS y un código HTML mínimo al cliente y este ejecuta el código JS que construye el HTML de la página. Si el navegador tiene desactivada la ejecución de JS, la página no se podrá procesar.

- Progressive Web App (PWA): pueden ser desarrolladas usando SSR, CSR o una combinación de las dos y tienen como principal ventaja que pueden ser instaladas en cualquier dispositivo como si fueran una aplicación nativa. Las PWA se encuentran a medio camino entre una página web y una aplicación tradicional [4].

### 3.2.2 Tecnologías de Frontend

El frontend es la interfaz gráfica donde los usuarios interactúan con la web. Consiste en tres elementos principales: el HTML que crea la estructura de la página, el CSS que le da formato al aspecto y estructura, y JavaScript que describe la funcionalidad y es responsable de los elementos dinámicos. Estos tres elementos se suelen desarrollar mediante un framework que hace el desarrollo más fácil.



Dentro de los numerosos frameworks de desarrollo de frontend, se han analizado los tres más importantes hoy en día [5] [6]:

- Angular (Fig. 19): la primera versión de este framework fue publicado en 2010 por Google. Se trata de un framework muy completo, con numerosas características disponibles por defecto. Los proyectos en Angular usan TypeScript en lugar de JavaScript. Debido a su alto número de funcionalidades y la complejidad añadida del uso de TypeScript, presenta una curva de aprendizaje pronunciada.



*Figura 19: logo del framework Angular*

- React (Fig. 20): desarrollado y mantenido por Facebook. Al contrario de Angular, es un framework muy simple que contiene pocas funcionalidades por defecto que son ampliables mediante módulos. Se desarrolla normalmente mediante JavaScript mezclado con HTML, lo que se conoce como JSX. Debido a esto, la curva de aprendizaje de React también es pronunciada, como la de Angular. Es un framework muy demandado por las empresas tecnológicas.



*Figura 20: logo del framework React*

- Vue.js (Fig. 21): a diferencia de Angular y React, fue fundado por un único desarrollador (Evan You) en lugar de una empresa tecnológica y actualmente está mantenida mediante donaciones por un equipo oficial de desarrolladores. Las aplicaciones hechas en Vue suelen usar JavaScript y el HTML está separado de la lógica de JS. Dispone de menos funcionalidades por defecto que Angular pero más que React. Es un framework más fácil de aprender que el resto de los

mencionados. Históricamente, la popularidad de Vue ha sido menor que Angular/React pero su importancia crece con fuerza.



Figura 21: logo del framework Vue.js

### 3.2.3 Tecnologías de Backend

El backend es el componente encargado de almacenar y manipular datos. Al contrario del frontend, en el backend los usuarios no realizan interacciones directamente. En la plataforma ALLVIEW, el backend está compuesto por tres componentes: una API REST que hace de intermediario entre el frontend y la base de datos, la base de datos y una aplicación que implementa los algoritmos de machine learning.

A continuación, se comparan las principales tecnologías de backend analizadas para la plataforma ALLVIEW [7]:

- Node.js (Fig. 22) [8]: es un entorno de ejecución que permite ejecutar JavaScript en el servidor. Actualmente JavaScript permite mucho más que hacer páginas webs interactivas, siendo igual de capaz que las demás tecnologías de backend. Las principales ventajas de Node.js son el menor coste de desarrollo al usar el mismo lenguaje para el backend que para el frontend, además de su gran velocidad de procesamiento y escalabilidad debido a su sistema de multitarea asíncrono. Sus puntos negativos son su bajo rendimiento en aplicaciones de cómputo pesadas y el modelo de programación asíncrono que dificulta el desarrollo [9].



Figura 22: logo del entorno Node.js

- Python (Fig. 23): es un lenguaje de alto nivel considerado de fácil aprendizaje, muy popular hoy en día para todo tipo de proyectos. Las ventajas de Python son el gran soporte de la comunidad, amplio número de librerías disponibles y una curva de aprendizaje baja. Por el contrario, Python tiene menor rendimiento comparado con otras tecnologías [10]. Hay que destacar además, que Python es el lenguaje por excelencia para el desarrollo de aplicaciones de Machine Learning [11].



*Figura 23: logo del lenguaje de programación Python*

- PHP (Fig. 24): lenguaje diseñado para backend muy popular, con gran soporte de la comunidad y muy estable. PHP tiene menor rendimiento que Node.js en aplicaciones típicas web con un gran número de usuarios simultáneos.



*Figura 24: logo del lenguaje de programación PHP*

- Java (Fig. 25): sigue siendo un lenguaje muy usado para el backend, con amplio soporte disponible y de alto rendimiento. No obstante, la curva de aprendizaje de Java es más inclinada comparado con el resto de tecnologías expuestas y es más complejo de mantener debido a la longitud de código [12].



Figura 25: logo del lenguaje de programación Java

Una vez analizadas las tecnologías de desarrollo de backend, es importante analizar los sistemas de bases de datos más relevantes para el almacenamiento, organización y extracción de los datos que este backend debe manejar.

- Bases de datos relacionales (Fig. 26): los datos se almacenan de forma estructurada con relaciones entre ellos. Estas bases de datos usan un modelo donde los datos se organizan en tablas. Dentro de estas tablas, cada fila dispone de un registro con un identificador y las columnas contienen los atributos. Gracias a esto se pueden establecer fácilmente relaciones entre los datos de diferentes tablas.



Figura 26: ejemplos de bases de datos relacionales

La Fig. 26 muestra tres ejemplos de bases de datos relacionales muy conocidas hoy en día [13]: PostgreSQL, MySQL y SQLite.

- Bases de datos no relacionales (Fig. 27): este tipo de bases de datos no implementan un esquema de filas y columnas que permitan relaciones entre los datos. En su lugar, dan más libertad al tipo de datos que se insertan y suelen ser más rápidas.



Figura 27: ejemplos de bases de datos no relacionales

En la Fig. 27 se pueden ver tres ejemplos representativos de bases de datos no relacionales [13]: Redis, MongoDB y Amazon DynamoDB.

- Bases de datos orientadas a objetos (Fig. 28): almacenan los datos de manera similar a los objetos de la programación orientada a objetos tradicional [14].



Figura 28: ejemplos de bases de datos orientadas a objetos

La Fig. 28 muestra tres ejemplos de bases de datos orientadas a objetos [13]: ObjectivityDB, GemStone y ObjectDB.

- Bases de datos basadas en grafos (Fig. 29): usan estructuras de grafos para organizar los datos mediante nodos, aristas y propiedades. El grafo relaciona los elementos en una colección de nodos y aristas, donde los nodos contienen el dato y las aristas las relaciones entre los nodos.



Figura 29: ejemplos de bases de datos basadas en grafos

En la Fig. 29 se pueden observar tres ejemplos de bases de datos basadas en grafos populares [13]: Neo4j JanusGraph y TigerGraph.

### 3.3 Arquitecturas y tecnologías escogidas

Tras realizar el análisis de las tecnologías más relevantes para el desarrollo de la plataforma ALLVIEW, en esta sección se explican las arquitecturas y tecnologías elegidas y el motivo de su elección. En resumen, el frontend se ha desarrollado como un Single Page Application (SPA) que usa la arquitectura web Client-Side Rendering (CSR), implementado mediante el framework Vue.js. En el backend se ha desarrollado una REST API mediante el entorno Node.js, un sistema de recomendación en Python y se ha usado una base de datos relacional MySQL.

#### 3.3.1 Arquitectura web

La arquitectura Client-Side Rendering ha sido elegida entre las arquitecturas web presentadas. La principal razón de esta elección es que la plataforma ALLVIEW no requiere de una indexación en navegadores excepcional, ya que se trata de una aplicación web que relaciona datos, al contrario que en un foro, una tienda online o una página de noticias.

Las aplicaciones CSR son normalmente más sencillas de desarrollar, mantener y escalar que las que realizan el renderizado en el servidor. Las Single Page Application (SPA) son un subgrupo dentro de CSR y suelen ser la elección para la gran mayoría de aplicaciones web actuales.

Otro punto positivo de las webs desarrolladas como CSR-SPA, es que pueden funcionar como Progressive Web Application (PWA) a la vez, y por tanto, es posible tener una página web, un programa de escritorio y una aplicación móvil con el mismo código.

#### 3.3.2 Frontend

Teniendo en cuenta los análisis que se han detallado en los apartados anteriores y los requisitos técnicos de la plataforma, el frontend de ALLVIEW ha sido desarrollado como una Single-Page Application (SPA) con el framework de desarrollo web Vue.js.

Se ha elegido el framework Vue.js frente a React y Angular por tener experiencia previa, ya que los tres frameworks son igualmente válidos para el desarrollo de esta plataforma. Ninguno de ellos es claramente superior al resto.

La principal función del frontend es mostrar datos a los usuarios, permitir una navegación por las diferentes partes de la web y modificación de datos de la plataforma.

El frontend está conectado mediante una REST API con el backend, donde se lleva a cabo la lógica interna de la plataforma y se realiza el intercambio de información.

En relación con el diseño, la interfaz de la plataforma está basada en el framework de diseño Vuetify (Fig. 30), que proporciona componentes responsivos basados en el estilo Material Design.



*Figura 30: logo del framework de diseño Vuetify*

### **3.3.3 Backend**

Como se ha explicado anteriormente, el backend de la plataforma ALLVIEW está conformado por tres componentes:

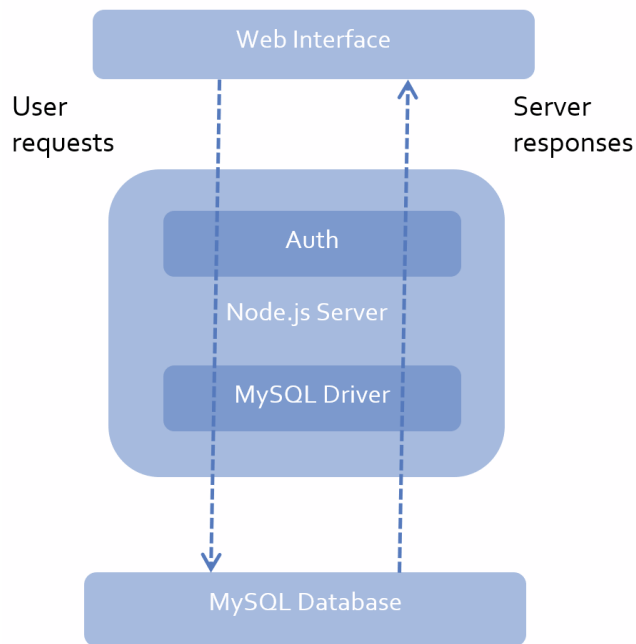
- Una base de datos relacional MySQL. Se ha escogido esta base de datos ya que los datos de la plataforma están muy relacionados entre sí y necesitan tener una estructura definida. Contiene los datos personales de los usuarios, roles, habilidades, competencias, ocupaciones, cursos de formación, ofertas de empleo, etc., y sus relaciones.
- Un servidor HTTP que implementa la lógica de la plataforma desarrollado con Node.js. Se ha escogido este entorno por las grandes ventajas de usar JavaScript en el backend y frontend en cuanto a la transferencia de información y al mantenimiento de código. Este servidor se comunica con el frontend mediante una API e interactúa con la base de datos MySQL con el driver MySQL para Node.js. El servidor ha sido desarrollado mediante el framework Express (Fig. 31) siguiendo una arquitectura REST, donde no se guarda el estado en las consultas, estas son independientes entre ellas. Dentro del backend, el servidor Node.js realiza las operaciones CRUD (create, read, update y delete), cifrado de datos y autenticación de usuarios.

# express

*Figura 31: logo del framework Express*

- Un programa escrito en Python será el encargado del sistema de recomendación usando Machine Learning. Se ha escogido Python como lenguaje para implementar el sistema de recomendación debido al gran soporte de librerías y comunidad en torno a esta temática. Estos algoritmos de machine learning evaluarán y actualizarán datos de la base de datos periódicamente. Gracias a esto, los usuarios no deben esperar a que los algoritmos se ejecuten mientras navegan por la plataforma. El desarrollo de este componente del backend no ha comenzado a desarrollarse y se propone como línea futura.





*Figura 32: flujo de datos de la plataforma web con el usuario*

La Fig. 32 muestra el intercambio de datos entre la interfaz web y la base de datos. Cuando un usuario interactúa con la interfaz web, el navegador realiza consultas hacia la API implementada en el servidor Node.js. Este servidor realiza comprobaciones de autenticación, descifra los datos e identifica el tipo de consulta. Dependiendo de la consulta, el servidor realizará una actualización, inserción, lectura o borrado de algún dato de la base de datos y devolverá una respuesta al navegador. Todo el intercambio de datos entre la interfaz web y el servidor se realiza mediante documentos JSON.

## 4. Resultados

Durante estos meses de trabajo se han desarrollado gran parte de componentes requeridos de la plataforma web que se han descrito en los apartados anteriores. En concreto, se ha completado una versión básica que dispone de una portada para usuarios no registrados, páginas de acceso y registro, y las funcionalidades básicas necesarias de los tres tipos de usuarios de la plataforma. El desarrollo de la plataforma continuará hasta su finalización prevista a final de año.

A continuación se incluyen algunas capturas de pantalla de la plataforma desarrollada para mostrar el nivel de desarrollo alcanzado. La Fig. 33 y 34 contienen la página de portada, la cual será mostrada a los usuarios no registrados. La Fig. 35 muestra el formulario de la página de registro. En la Fig. 36, se puede observar la página de la portada interna del usuario básico. La Fig. 37 muestra la interfaz de búsqueda de cursos y la Fig. 38, la página de detalles del curso. En la Fig. 39, se muestra la página de cursos guardados del usuario básico. Finalmente, la Fig. 40 contiene la interfaz que muestra los itinerarios de aprendizaje del usuario básico.

# Connecting people, training providers and companies for the European wood and furniture industry



People perform training courses to improve their skills and competences. Training providers research demanded skills by companies to offer training courses. Companies look for candidates for job offers and training courses for their employees. The Allview Platform connects these groups of people to improve the European wood and furniture industry.

PARTNERS



## The Allview Platform

The all-in-one platform that connects people and enterprises from the European wood and furniture industry



### Search for training courses

Users can look for training courses by applying filters such as keywords, skills, competences, types, topics, languages and Allview partners



### Search for job offers

Registered users can search for job offers by filtering keywords, skills, competences, companies, languages, and contract types



### Learning paths

Individuals can specify their desired job challenges and the platform will show the recommended skills and competences to learn



### Identify the most demanded skills

Training providers can identify the most demanded skills and competences requested by companies, and thus they can create the appropriate courses

Figura 33: parte superior de la portada de la plataforma ALLVIEW



### ESCO

ESCO is the multilingual classification of European Skills, Competences, Qualifications and Occupations. ESCO is part of the Europe 2020 strategy. The ESCO classification identifies and categorises skills, competences, qualifications and occupations relevant for the EU labour market and education and training. It systematically shows the relationships between the different concepts.

[Learn More](#)



### Recommendations

By using machine learning recommender systems, the platform performs recommendations of training courses, job offers, skills, and competences based on similar user profiles. These recommendations allow to improve the connection among people, training providers and companies for the European wood and furniture industry.

[Learn More](#)



### Centres of vocational experience

CoVEs aim to drive innovation and excellence in vocational education and training, by bringing together a wide range of local partners. VET providers, employers, chambers, research centres, development agencies, employment services, and social partners work together creating "skills ecosystems", that contribute to innovation, economic and social development at a regional and European level.

[Learn More](#)

#### Navigation

[Home](#)  
[Help](#)  
[Allview Website](#)

#### Contact Information

 **C/ Perales S/N 30510 Yecla, Murcia, Spain**  
 +34 968 752 040  
 [info@allview.eu](mailto:info@allview.eu)

#### Social Networks



© Allview 2021. All Rights Reserved

Figura 34: parte inferior de la portada de la plataforma ALLVIEW

av ALLVIEW PLATFORM Login Register

## Sign up today

Name

Email

Password

Repeat Password

Role

**Create Account**

OR SIGN UP WITH

BY SIGNING UP, YOU AGREE TO THE [TERMS AND CONDITIONS & PRIVACY POLICY](#)

Figura 35: página de registro de la plataforma



## Search for Courses & Jobs in the ALLVIEW Platform

### Top rated courses



Sample Course 1

Sample partner 1



Sample Course 2

Sample partner 2



Sample Course 3

Sample partner 3



Sample Course 4

Sample partner 4

Explore all the courses [→](#)

### Recommended courses for you



Sample Course 1

Sample partner 1



Sample Course 2

Sample partner 2



Sample Course 3

Sample partner 3



Sample Course 4

Sample partner 4

Explore all the courses [→](#)

### New Jobs



Sample Job 1

Sample Company 1



Sample Job 2

Sample Company 2



Sample Job 3

Sample Company 3



Sample Job 4

Sample Company 4

Explore all the jobs [→](#)

Figura 36: portada interna del usuario básico



## Search Training Courses from Allview Partners

Courses Keywords

Skills

Competences

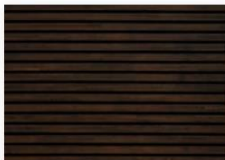
Types

Topics

Languages

Partners

Search



**Computer Literacy Of Older Adults**

WIC, ULUB, NIVET



**Digital UI - With Innovative Use Of Ict To Excellence**

WIC, ULUB, NIVET



**Smart Augmented And Virtual Reality Marketplace For Furniture Customisation**

WIC, ULUB, NIVET



**Tylko App.**

WIC, ULUB, NIVET



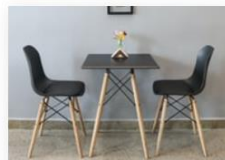
**Virtual Tour Of Showroom**

WIC, ULUB, NIVET



**Ikea Place**

WIC, ULUB, NIVET



**Summit Furniture**

WIC, ULUB, NIVET




**Interior Define**

WIC, ULUB, NIVET

Figura 37: buscador de cursos disponibles en la plataforma




**ALLVIEW PLATFORM**
Profile
Courses
Rate
Jobs
Learning Paths
Settings
Logout

## Training Course Details

**Computer literacy of older adults**

Teaching elderly about basics of internet and ICT

---

**Category:** Circular economy

**Topic:** E-literacy of elderly

**Keywords:** Elderly, ICT, Internet, Course

**Type of learning resource:** Webpage

**License Type:** Free resource

**Weblink:** <https://epale.ec.europa.eu/sl/blog/racunalnisko-opismenjevanje-starejsih-odraslih>

**Language:** Slovenian

**Date of creation:** 2014

**Relevance:** Contemporary

**EQF NQF level:** To be estimated

**Partner:** WIC, ULUB, NIVET

Add course to favorites
Remove from favorites






Figura 38: página detallada de cada curso



**ALLVIEW PLATFORM**
Profile
Courses
Rate
Jobs
Learning Paths
Settings
Logout

## My Favorite Courses




**Circular Economy: An Introduction**

WULS-SGGW



**Vr Eyeglasses And Equipment: Ultimate Guide About Vr World**

WIC, ULUB, NIVET




**Computer Literacy Of Older Adults**

WIC, ULUB, NIVET

Figura 39: página de cursos favoritos del usuario




[ALLVIEW PLATFORM](#)
[Profile](#)
[Courses](#)
[Rate](#)
[Jobs](#)
[Learning Paths](#)
[Settings](#)
[Logout](#)

## Learning Paths

**Skills and competences for: Wood products assembler**

Use technical documentation (essential)	✘
Develop assembly instructions (optional)	✘
Read engineering drawings (optional)	✘
Record production data for quality control (optional)	✘
Keep records of work progress (optional)	✘
Prepare production reports (optional)	✘
Prepare a comprehensive and unbiased report on wood technology production and progressive development of wood based materials.	

**Skills and competences for: Furniture assembler**

Follow written instructions (essential)	✘
Memorise assembly instructions (essential)	✘
Sell furniture (optional)	✘
Sell household goods (optional)	✘
Read standard blueprints (optional)	✘
Keep records of work progress (optional)	✘

Figura 40: página de elaboración de itinerarios de aprendizaje

## 5. Conclusiones

En esta memoria se ha descrito en primer lugar el objetivo de la plataforma ALLVIEW: desarrollar una plataforma web donde personas, empresas y proveedores de formación puedan poner en común sus recursos y objetivos con el fin de enriquecer el sector de la madera y mueble. A continuación, se han enumerado las funcionalidades básicas que debe implementar la plataforma, tanto la diferenciación entre los tres tipos de usuarios como las funcionalidades de cada uno. Finalmente, se ha detallado la implementación técnica, las arquitecturas y tecnologías usadas en el frontend y backend y la razón de su elección.

El resultado final del trabajo de Master es el desarrollo de una primera versión de la plataforma ALLVIEW, donde gran parte de los requisitos funcionales han sido completados.

El desarrollo completo de la plataforma está en curso. Se propone como trabajo futuro la implementación del sistema de recomendación, la extracción de datos de diferentes fuentes y perfeccionar la plataforma. Los próximos meses se dedicarán a completar estas tareas.

## 6. Bibliografía

- [1] ALLVIEW, «Allview.eu,» [En línea]. Available: <https://www.allview.eu/>.
- [2] Comisión Europea, «ec.europa.eu,» [En línea]. Available: <https://ec.europa.eu/esco/portal/home>.
- [3] Bacancytechnology, «<https://www.bacancytechnology.com/blog>,» [En línea]. Available: <https://www.bacancytechnology.com/blog/client-side-rendering-vs-server-side-rendering-vs-pre-rendering>.
- [4] Cloudinary, «<https://cloudinary.com/blog>,» [En línea]. Available: [https://cloudinary.com/blog/progressive\\_web\\_apps\\_architecture\\_and\\_examples#pwa](https://cloudinary.com/blog/progressive_web_apps_architecture_and_examples#pwa).
- [5] Academind, «[academind.com](https://academind.com),» [En línea]. Available: <https://academind.com/tutorials/angular-vs-react-vs-vue-my-thoughts/>.
- [6] Clockwise, «<https://clockwise.software/blog>,» [En línea]. Available: <https://clockwise.software/blog/angular-vs-react-vs-vue/>.
- [7] Brainvire, «<https://www.brainvire.com/blog>,» [En línea]. Available: <https://www.brainvire.com/blog/why-is-node-js-better-than-php-java/>.
- [8] Existek, «<https://existek.com/blog>,» [En línea]. Available: <https://existek.com/blog/node-js-backend-development/>.
- [9] Simform, «<https://www.simform.com/blog>,» [En línea]. Available: <https://www.simform.com/blog/nodejs-advantages-disadvantages>.
- [10] Techvidvan, «<https://techvidvan.com/tutorials>,» [En línea]. Available: <https://techvidvan.com/tutorials/python-advantages-and-disadvantages/>.
- [11] Technative, «<https://technative.io>,» [En línea]. Available: <https://technative.io/why-use-python-for-ai-and-machine-learning/>.
- [12] Dzone, «<https://dzone.com/articles>,» [En línea]. Available: <https://dzone.com/articles/the-good-and-the-bad-of-java-programming>.
- [13] db-engines, «[db-engines.com](https://db-engines.com),» [En línea]. Available: <https://db-engines.com/en/ranking>.

[14] Phoenixnap, «phoenixnap.com,» [En línea]. Available:  
<https://phoenixnap.com/kb/object-oriented-database>.