UAB
Universitat Autònoma de Barcelona

Dipòsit digital de documents de la UAB

This is the **published version** of the bachelor thesis:

Argente Guirao, Meritxell. *Aplicación web para la gestión de ofertas y asignación de prácticas curriculares*. Treball de Final de Grau (Universitat Autònoma de Barcelona), 2026 (Enginyeria Informàtica)

This version is available at https://ddd.uab.cat/record/334429

under the terms of the  license.

Aplicación Web para la Gestión de Ofertas y Asignación de Prácticas Curriculares

Meritxell Argente Guirao

Resumen— Las prácticas externas constituyen un elemento fundamental en la formación universitaria, ya que permiten a los estudiantes aplicar los conocimientos adquiridos facilitando así su transición al mercado laboral. En el grado de Ingeniería Informática de la *Escola d'Enginyeria* de la *Universitat Autònoma de Barcelona*, la gestión de ofertas y acuerdos de prácticas curriculares implica la participación de diversos actores, como la coordinación de prácticas, gestión académica, las entidades colaboradoras y los propios estudiantes. Sin embargo, el procedimiento seguido actualmente presenta limitaciones derivadas de tareas administrativas manuales, así como de la falta de trazabilidad y transparencia en la gestión de las ofertas. Con el objetivo de mejorar este proceso, se ha diseñado e implementado una aplicación web que centraliza la publicación e inscripción de ofertas y la gestión de acuerdos. La solución desarrollada optimiza las tareas administrativas, mejora la coordinación entre las partes implicadas y proporciona a los estudiantes una mayor visibilidad sobre el estado de sus candidaturas y la evolución de los procesos de selección asociados a cada oferta.

Paraules clau— Prácticas externas curriculares, Scrum, Aplicación Web, Clean Architecture, .NET, Blazor, PostgreSQL, Swagger, DBeaver, Entity Framework Core.

Abstract— Curricular internships are a fundamental component of university education, as they allow students to apply the knowledge acquired during their studies and facilitate their transition into the professional world. In the Computer Engineering degree at the *Escola d'Enginyeria* of the *Universitat Autònoma de Barcelona*, the management of internship offers and agreements involves multiple stakeholders, including the internship coordination office, academic management, collaborating organizations, and students themselves. However, the current process presents significant limitations stemming from manual administrative tasks and a lack of traceability and transparency in offer management. To address these issues, a web application has been designed and implemented to centralize the publication and registration of internship offers and the management of internship agreements. The developed solution streamlines administrative tasks, improves coordination among all parties involved, and provides students with greater visibility into the status of their applications and the progress of the selection processes associated with each offer.

Index Terms— Curricular internships, Scrum, Web platform, Clean Architecture, .NET, Blazor, PostgreSQL, Swagger, DBeaver, Entity Framework Core.

1 Introducción - Contexto del trabajo

En el sistema universitario español, las prácticas externas curriculares forman parte de la mayoría de los planes de estudio de los diferentes grados. Permiten a los estudiantes complementar su formación académica mediante la aplicación de conocimientos adquiridos en un entorno profesional y favorecen el desarrollo de competencias transversales esenciales para su incorporación en el mercado laboral.

Desde la *Escola d'Enginyeria* (en adelante, EE) de la *Universitat Autònoma de Barcelona* (en adelante, UAB) se impulsan diversas actividades y procesos para dar soporte a los estudiantes en la búsqueda de una entidad colaboradora (en adelante, entidad) donde realizar las prácticas. Sin embargo, la gestión de las ofertas, su publicación y la formalización del convenio asociado, presentan limitaciones para las partes implicadas en el proceso: coordinación de prácticas, gestión académica, estudiantes y entidades.

En el grado de Ingeniería Informática, alcance inicial de este proyecto, las ofertas se publican por la coordinación de prácticas en la herramienta Campus Virtual. Esta publicación se realiza tras la recepción de la oferta por correo electrónico y requiere de una gestión manual para mantenerlas actualizadas. Por otro lado, la formalización del convenio asociado se gestiona a través del correo electrónico institucional e interviene la coordinación de prácticas, gestión académica, estudiante y entidad para definir las condiciones de la estancia y formalizar el acuerdo.

El procedimiento seguido con tareas administrativas mayoritariamente manuales y repetitivas, conlleva ciertos riesgos inherentes como el riesgo de pérdida de información y la dificultad en la trazabilidad de las ofertas, entre otros.

- *E-mail de contacto: Meritxell.argente@autonoma.cat*
- *Mención realizada: Ingeniería del Software*
- *Trabajo tutorizado por: Jordi Pons (DEIC)*
- *Curso 2025/26*

En este contexto, el proyecto propone una plataforma web institucional con el objetivo de centralizar todo el proceso en una sola herramienta: ConnectUs. En ella participan todas las partes implicadas mediante roles definidos, permitiendo gestionar la publicación de ofertas, la inscripción de estudiantes y seguimiento del estado de sus candidaturas, y la gestión de acuerdos de prácticas.

El objetivo principal del proyecto es mejorar la eficiencia del proceso actual reduciendo las tareas manuales, aumentando la trazabilidad de la información y proporcionando a los estudiantes mayor visibilidad sobre las ofertas disponibles y el estado de sus candidaturas.

En el presente informe se describe el proceso de diseño y desarrollo de la plataforma. En primer lugar, se presenta el estado del arte, donde se analizan soluciones existentes relacionadas con la gestión de ofertas para estudiantes. A continuación, se define el alcance del proyecto, así como sus posibles líneas de evolución futura, y se detallan los objetivos y metodología seguida para su consecución. Posteriormente, se tratan las cuestiones técnicas, incluyendo arquitectura y tecnologías utilizadas, junto con la interfaz de usuario y las pruebas realizadas. Finalmente, se exponen las funcionalidades implementadas, los resultados obtenidos y las conclusiones del trabajo.

2 Estado del arte

Actualmente no existe una plataforma específica en la EE que centralice la búsqueda de ofertas de prácticas curriculares, su inscripción y la gestión de acuerdos entre los estudiantes y las entidades para formalizar el convenio asociado. Sin embargo, existen distintas plataformas generalistas que cubren parcialmente esta limitación y que los estudiantes utilizan para la búsqueda de ofertas.

2.1 Plataformas de empleo generalistas

Existen numerosas plataformas digitales orientadas a un mercado laboral amplio que incluye perfiles de distintos niveles formativos, como estudiantes universitarios, de ciclos formativos o los recientes *bootcamps* en el sector tecnológico. Entre las más comunes destacan **LinkedIn** [1], **Indeed**[2] e **Infojobs** [3]. La primera de ellas es actualmente la mayor red profesional a nivel global y su principal valor reside en la creación de una red profesional. Indeed destaca por su gran volumen de ofertas y sistema de búsqueda personalizado, mientras que Infojobs es la plataforma líder en España.

A pesar de su relevancia y amplio alcance, estas plataformas presentan ciertas limitaciones en el contexto de las prácticas universitarias. El elevado número de ofertas y candidatos puede dificultar la identificación de oportunidades adecuadas para estudiantes que buscan su primera experiencia en el mercado laboral. Además, la información proporcionada en muchas de las ofertas no siempre se adapta al contexto académico, omitiendo información como el horario necesario para valorar la compatibilidad con los estudios o las condiciones específicas del entorno formativo.

2.2 Plataformas enfocadas a estudiantes

Como alternativa a las plataformas generalistas, existen plataformas orientadas a estudiantes de distintos niveles. Entre ellas destaca **Talentoteca** [4], una plataforma lanzada por la Fundación Universidad-Empresa (FUE). Entre sus funcionalidades destaca la publicación e inscripción de ofertas, así como la colaboración con universidades a través de programas de seguimiento y tutorización.

Aunque este tipo de solución parece estar más alineado con el ámbito académico, no proporciona una trazabilidad completa del proceso de selección de los estudiantes ni integra la gestión académica asociada a las prácticas curriculares como la asignación de tutores académicos o la formalización del convenio.

2.3 Plataformas institucionales de la UAB

En el ámbito de la universidad, la gestión de las prácticas se realiza mediante diferentes herramientas según la modalidad de estas. Para las prácticas extracurriculares existe la plataforma ampliamente conocida por los estudiantes, **Nexus** [5]. Es una plataforma utilizada por todos los centros de la UAB donde las entidades pueden registrar sus ofertas. Sin embargo, son ofertas destinadas a estancias de prácticas extracurriculares, que no son el objeto de este proyecto.

Las plataformas analizadas han servido como referencia para identificar funcionalidades habituales en la gestión de ofertas y aspectos relacionados con la experiencia de usuario y el diseño de este tipo de herramientas. Sin embargo, ninguna de las soluciones se adapta completamente a las necesidades específicas para gestionar las ofertas de prácticas curriculares, ya que se orientan principalmente a la búsqueda de empleo o prácticas extracurriculares. Asimismo, las herramientas institucionales existentes no contemplan actualmente la adaptación necesaria para gestionar las prácticas curriculares.

Por ello, ConnectUs se plantea como una solución específica que centraliza las necesidades identificadas y adapta el proceso a las características propias de las prácticas curriculares.

3 Alcance del proyecto

El alcance inicial de la plataforma ConnectUs se centra en la gestión de ofertas y acuerdos de prácticas curriculares en el grado de Ingeniería Informática de la EE de la UAB. La aplicación permite centralizar en una sola herramienta la publicación de las ofertas, la inscripción de estudiantes, el seguimiento de las candidaturas y la gestión de los acuerdos de prácticas entre las partes implicadas.

No obstante, el diseño de la aplicación se ha realizado con el objetivo de ofrecer la posibilidad de extender la plataforma a otros estudios de la EE, a medio plazo, y a otros centros de la UAB a largo plazo. La definición de roles, la gestión independiente de entidades como centros, departamentos y estudios, permiten adaptar la aplicación a nuevos escenarios sin modificar la lógica del sistema.

Asimismo, ConnectUs establece una base sobre la que pueden incorporarse nuevas funcionalidades futuras,

como la generación de estadísticas sobre ofertas que faciliten la detección de variaciones en la demanda o el traspaso de datos seguro a las herramientas utilizadas actualmente por la coordinación de prácticas y tutores académicos para el seguimiento de las prácticas.

4 Objetivos

El objetivo principal de este proyecto es diseñar y desarrollar una aplicación web que permita gestionar de forma centralizada y eficiente la publicación de ofertas de prácticas curriculares y su asignación a estudiantes de la EE, con un enfoque inicial en el grado de Ingeniería Informática.

Para lograr este objetivo, se definen los siguientes objetivos específicos:

- **Eficiencia administrativa**. Reducir la carga administrativa asociada a la gestión de ofertas de prácticas, centralizando la información en una única plataforma y eliminando tareas manuales.
- **Soporte a los estudiantes**. Dar soporte a los estudiantes y mejorar su experiencia en la búsqueda de una entidad donde realizar las prácticas.
- **Mejora de la colaboración entre la EE y las entidades**. Impulsar la colaboración de las entidades con la EE ofreciendo un formulario guiado e intuitivo para facilitar la publicación de sus ofertas.
- **Agilización del proceso de asignación de prácticas**. Reducir el tiempo entre la publicación de la oferta y su asignación a uno de los estudiantes.
- **Mejora de la trazabilidad de las ofertas**. Facilitar el seguimiento de su ciclo de vida, así como el registro de los cambios de estado y de las acciones realizadas sobre ellas.
- **Disponibilidad de información actualizada de las ofertas disponibles**. Ofrecer tanto a los estudiantes como a la coordinación de prácticas de la EE información sobre las ofertas actualizadas en todo momento.
- **Impacto universitario**. Ofrecer la posibilidad de que otros estudios y centros de la UAB puedan hacer uso de la aplicación.

5 Metodología

5.1 Modelo de desarrollo y planificación

El proyecto se ha realizado siguiendo la metodología Scrum [6], marco de trabajo ampliamente conocido y utilizado en proyectos complejos. Esta metodología ha permitido mejorar la eficiencia durante las fases de desarrollo y facilitar la detección de desviaciones respecto a los objetivos y requerimientos del sistema en una fase temprana.

Durante el desarrollo del proyecto se han realizado un total de seis sprints:

- **Sprint 0: Definición del proyecto y diseño de la solución**. En este sprint se realizaron todas las tareas relacionadas con la definición del proyecto (análisis de la situación actual, alcance y requisitos del sistema) y el diseño de la solución, incluyendo la arquitectura, tecnologías, sistema gestor de la base de datos, y diseños iniciales.
- **Sprint 1: Autenticación y gestión de usuarios internos**. Primer sprint de desarrollo en el que se implementó el servicio de usuarios internos que incluye el registro, la autenticación y gestión de su información básica.
- **Sprint 2**: **Gestión de las entidades**. Sprint centrado en las operaciones iniciales que puede realizar una empresa en la plataforma, incluyendo su registro, edición de su perfil y validación por parte de la administración de la plataforma.
- **Sprint 3**: **Gestión de ofertas**. Este sprint se centró en la gestión de las ofertas, incluyendo el registro y edición por parte de las entidades y la validación por parte de la coordinación de prácticas de los estudios a los que se dirige la oferta.
- **Sprint 4**: **Perfil del estudiante y gestión de las inscripciones de ofertas**. Sprint dedicado al servició de gestión del estudiante que incluye la creación y edición de un perfil con datos de formación académica, idiomas, redes sociales de interés para las entidades y currículums, entre otros. Asimismo, se implementó el servicio de inscripciones para gestionar las inscripciones, incluyendo la gestión de estas por parte de la entidad.
- **Sprint 5: Gestión de acuerdos**. Último sprint de desarrollo en el que se incorporaron las funcionalidades necesarias para que la empresa pueda registrar un acuerdo para un estudiante concreto, la validación por parte de la coordinación de prácticas y la descarga de los datos en formato Excel para la tramitación del convenio.

En cuanto a las reuniones previstas para este marco de trabajo, se han realizado las reuniones *Sprint Planning*, *Sprint Review* y *Sprint Retrospective* en un único día con el cliente cada dos sprints.

5.2 Organización del trabajo

Para la organización del trabajo se ha utilizado la herramienta Notion. En el inicio del proyecto se creó un espacio de trabajo estructurado mediante distintas bases de datos interrelacionadas que han permitido mantener la trazabilidad entre sprints, tareas y reuniones.

El uso de la herramienta ha permitido organizar el trabajo y hacer un seguimiento y control efectivo del proyecto durante su desarrollo.

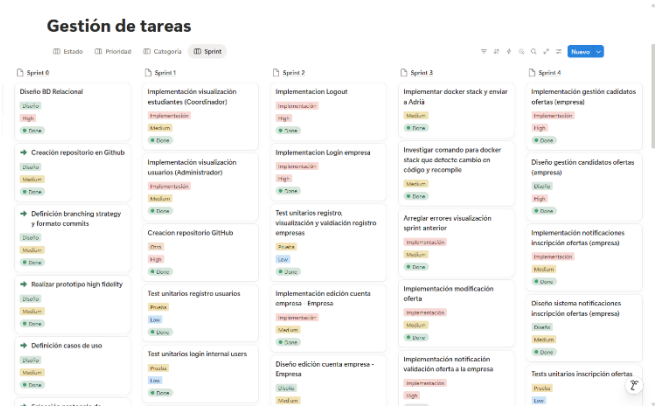

Figura 1. Gestión de tareas en Notion

6 Diseño de la solución

6.1 Requisitos del sistema

Para la definición de los requisitos del sistema se identificaron en primer lugar los actores que intervienen en el proceso de la gestión de ofertas y acuerdos de prácticas curriculares: administración de la plataforma, coordinación de prácticas, gestión académica, estudiante y entidad. Cada uno de ellos tiene asignado un rol con permisos diferenciados que determinan las operaciones que puede realizar sobre el sistema.

A partir de esta identificación, se definieron los requisitos funcionales y no funcionales del sistema siguiendo el método MoSCoW, que clasifica cada requisito según su prioridad. El listado completo de requisitos funcionales puede consultarse en el Anexo A1.

6.2 Arquitectura de la solución

La solución se estructura en una arquitectura cliente-servidor compuesta por dos procesos: backend y frontend, que se comunican mediante una API REST sobre HTTP.

Adicionalmente, se ha desarrollado una biblioteca compartida (*Shared*) que contiene contratos y entidades compartidas por ambos procesos. Esta biblioteca permite garantizar la consistencia del modelo de datos y evitar duplicidad de código.

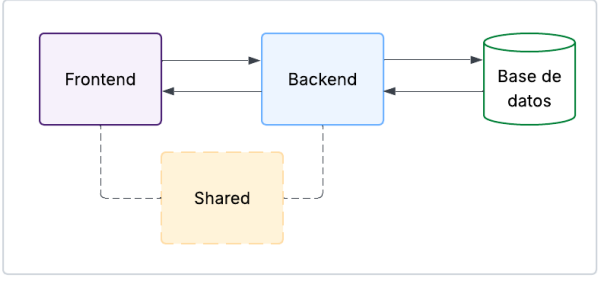

Figura 2. Diagrama arquitectura de la solución

6.3 Tecnologías utilizadas y arquitectura del backend

El backend de la plataforma se ha desarrollado utilizando C# sobre la plataforma .NET, utilizando ASP.NET Core como *framework*. Al ser C# un lenguaje fuertemente tipado, facilita la definición de contratos entre los procesos de la plataforma. Asimismo, .NET incorpora un contenedor de dependencias nativo, lo que permite aplicar el principio de inyección de dependencias.

En cuanto a la arquitectura, se ha utilizado el patrón *Clean Architecture* [7], estructurando el proceso en diferentes capas donde las dependencias se dirigen hacia el dominio. Estas capas son:

- ***App***. Capa encargada del arranque y configuración inicial de la aplicación. Pese a no formar parte explícita de las capas definidas por el patrón, permite aislar la lógica de inicialización del resto del sistema.
- ***Presentation***. Capa encargada de exponer los diferentes puntos de entrada de la aplicación mediante *endpoints* HTTP.
- ***Application***. Capa responsable de implementar los casos de uso de la aplicación y coordinar la lógica para ejecutar las operaciones solicitadas.
- ***Infrastructure***. Capa responsable del acceso a los datos. En ella se define el contexto de la base de datos mediante *Entity Framework Core*, utilizando *DbContext* como intermediario entre la aplicación y la base de datos.
- ***Domain***. Capa más interna de la arquitectura responsable de los modelos de dominio y donde se definen las entidades propias del backend.

La elección de las tecnologías y de la arquitectura se ha realizado considerando la necesidad de desarrollar un sistema modular, mantenible y fácilmente extensible. Además, estas decisiones debían adaptarse a los objetivos y requerimientos del sistema. Asimismo, se ha valorado la experiencia previa con estas tecnologías en proyectos desarrollados en un entorno profesional para garantizar un ritmo de desarrollo óptimo centrando los esfuerzos en el desarrollo de la solución.

6.4 Tecnologías utilizadas y arquitectura del frontend

El frontend de la aplicación se ha desarrollado con Blazor, una tecnología basada en .NET que permite crear páginas web interactivas utilizando C# junto con componentes que combinan HTML y CSS.

Se ha seleccionado el modelo de hospedaje Blazor Server [8], donde la lógica de la aplicación se ejecuta en el servidor y la interfaz de usuario se actualiza dinámicamente mediante una conexión persistente con el cliente utilizando *SignalR*.

La estructura inicial proporcionada por Blazor Server se ha ampliado incorporando una organización propia orientada a mejorar la modularidad y mantenibilidad del código. Por ello, se han definido directorios específicos para agrupar las páginas, componentes reutilizables, servicios y entidades del frontend.

Entre sus elementos principales cabe destacar *DependencyInjection*, responsable de centralizar el registro de los servicios utilizados por la aplicación mediante el

contenedor de dependencias .NET. Estos servicios incluyen funcionalidades propias del frontend y aquellos responsables de la comunicación con el backend.

Asimismo, se han desarrollado componentes reutilizables como *PageBase* y *OrganizationPageBase*, de los que heredan las páginas correspondientes. En estos componentes se centraliza la lógica común relacionada con la autenticación del usuario, validación de tokens JWT y control de permisos según el rol o roles asignados. De esta forma, se ha evitado la duplicación de código y se ha favorecido la consistencia de la aplicación.

6.5 Biblioteca Shared

El uso del mismo lenguaje de programación tanto en el frontend como en el backend ha permitido desarrollar una biblioteca compartida denominada *Shared*, con el objetivo de centralizar los contratos de comunicación y los tipos compartidos entre ambos procesos.

La biblioteca se divide en dos directorios principales:

- ***Communications***: encargado de abstraer la comunicación entre frontend y backend. Se definen las interfaces de los servicios, clientes HTTP específicos por dominio y módulos encargados de gestionar la creación de peticiones, serialización de datos y autenticación mediante tokens.
- ***StandardTypes***: encargado de contener los modelos compartidos como entidades, objetos de transferencia (DTOs), objetos de valor y extensiones para realizar el mapeo entre modelos de dominio y modelos expuestos mediante la API.

El desarrollo y uso de esta biblioteca evita que frontend y backend dependan directamente el uno del otro, reduciendo la duplicación de código y garantizando la consistencia del modelo de datos entre ambos procesos.

6.6 Modelado de datos

Para la persistencia de los datos se ha escogido PostgreSQL como sistema gestor de base de datos relacional por su velocidad, flexibilidad y compatibilidad con *Entity Framework Core* [9].

Para el acceso y la gestión de la base de datos se ha utilizado la clase *DbContext* en la que se definen las entidades del sistema que representan tablas en la base de datos. Además, se han realizado configuraciones adicionales sobre el modelo para las claves compuestas, relaciones entre las entidades y comportamientos de eliminación de registros. El diagrama entidad – relación puede consultarse en el Anexo A2.

Adicionalmente, se han utilizado migraciones [10] de *Entity Framework Core* que permiten mantener sincronizado el modelo de la aplicación con la base de datos sin pérdida de registros. Su aplicación se ha automatizado en el arranque de la plataforma, garantizando así que la base de datos esté siempre actualizada antes de la ejecución del sistema.

6.7 Arquitectura de despliegue

La aplicación se ha desplegado en un servidor del grado de Ingeniería Informática de la EE. Ha sido necesaria la creación de contenedores Docker personalizados, ya que los previstos actualmente en el servidor cuentan con tecnologías que no se adaptan a las seleccionadas por el proyecto.

Se han lanzado tres contenedores: un contenedor para backend y otro para frontend, ambos construidos sobre la imagen oficial del SDK de .NET, y un contenedor para la base de datos construido con la imagen oficial de PostgreSQL.

Asimismo, dado que la autenticación de los usuarios internos de la UAB se realiza a través del Servicio de Control de Acceso Centralizado de la universidad (CAS), se ha utilizado el contenedor de Apache ya existente en el servidor.

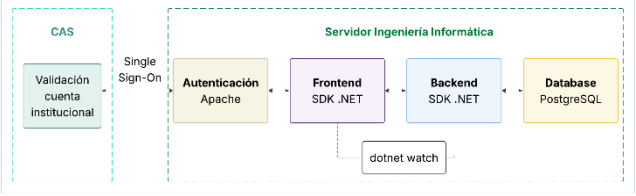

Figura 3. Diagrama arquitectura de despliegue

Con el objetivo de agilizar el proceso de despliegue, se configuró el comando *dotnet watch* [11] en los contenedores de backend y frontend. Este comando monitoriza todos los archivos del proyecto y, al detectar cambios, recompila y reinicia la aplicación de forma automática. De esta forma, se evita tener que reiniciar los contenedores tras cada actualización del código.

Asimismo, se definieron variables de entorno específicas para cada contenedor, incluyendo parámetros de conexión a la base de datos y configuración de JWT en el backend, y la URL del backend en el frontend. Esto permite desacoplar los parámetros sensibles del código fuente y facilitar su gestión en el entorno de despliegue.

6.8 Herramientas y entorno de desarrollo

Para el diseño de la solución y de las funcionalidades de la aplicación se han utilizado las herramientas Draw.io (generación de diagramas) y Figma (construcción del prototipo de alta fidelidad).

Para el desarrollo se ha utilizado Docker Desktop para contenerizar la base de datos, Swagger para documentar y probar los *endpoints* del backend y DBeaver como cliente de base de datos para visualizar y verificar los datos almacenados.

Para el control de versiones se ha utilizado GitHub siguiendo una estrategia basada en ramas (*main*, *develop* y *features*) junto con un estándar de nomenclatura de *commits* (*feat*, *fix*, *style*, *refactor*, *test*). Esto ha permitido facilitar la lectura del historial de cambios durante el desarrollo.

El entorno de desarrollo utilizado ha sido Rider de JetBrains, uno de los principales IDEs multiplataforma para el desarrollo de aplicaciones .NET.

6.9 Roles y actores del sistema

La plataforma define cinco actores con acceso diferenciado a las funcionalidades de la aplicación. La tabla 1 expone las principales funcionalidades de cada uno de ellos y los servicios a los que tienen acceso para ello.

Tabla 1. Roles y principales funcionalidades

Administración
Registro y visualización de usuarios (*Gestión de usuarios internos*).
Validación del registro de entidades (*Gestión de entidades*).
Coordinación de prácticas
Registro, visualización y edición de departamentos de su centro (*Gestión de departamentos*).
Registro, visualización y edición de tutores académicos de su centro (*Gestión de tutores académicos*).
Registro y visualización de estudiantes (*Gestión de usuarios internos*).
Validación y visualización de ofertas dirigidas a su estudio (*Gestión de ofertas*).
Visualización de inscripciones a ofertas y su estado (*Gestión de inscripciones*).
Validación, visualización y edición de acuerdos dirigidos a un estudiante de su estudio (*Gestión de acuerdos*).
Exportación de los datos de un acuerdo a fichero Excel (*Gestión de acuerdos*).
Estudiante
Creación y edición de un perfil (*Gestión de estudiantes*).
Visualización e inscripción a ofertas (*Gestión de ofertas e inscripciones*).
Consulta del historial de sus inscripciones y posibilidad de cancelación (*Gestión de inscripciones*).
Gestión académica
Consulta de ofertas y acuerdos de su centro (*Gestión de ofertas y acuerdos*).
Exportación de los datos de un acuerdo a fichero Excel (*Gestión de acuerdos*).
Entidad
Registro en la plataforma (*Gestión de entidades*)
Creación y edición de un perfil (*Gestión de entidades*).
Registro y consulta de sus ofertas (*Gestión de ofertas*).
Gestión de las candidaturas (*Gestión de inscripciones*).
Registro de acuerdos de prácticas (*Gestión de acuerdos*).

7 Interfaz de usuario

El diseño de la interfaz de usuario se ha definido a partir del análisis de diferentes plataformas de la universidad, así como portales de empleo y herramientas digitales del sector tecnológico.

Los objetivos principales en el diseño de la interfaz han sido garantizar una experiencia de usuario accesible y amigable que, a la vez, estuviese alineada con la identidad visual de la universidad.

Para la identidad visual de la aplicación se ha escogido la figura de un zorro. Esta elección se ha basado en la asociación del animal con ciertas características como la astucia, inteligencia y adaptabilidad. Estas cualidades se alinean con uno de los objetivos de la plataforma de dar soporte a los estudiantes y poner en valor su capacidad de adaptación en distintos contextos profesionales. Asimismo, el uso de un animal como marca puede conllevar beneficios como la generación de empatía del usuario hacia la aplicación, entre otros.

En cuanto al nombre de la solución se ha escogido ConnectUs dado que refleja el objetivo de lograr una conexión entre estudiantes, entidades y la UAB. Asimismo, el componente US refuerza la idea de comunidad, uno de los pilares de la universidad.

Por lo que refiere a la identidad corporativa, se han seguido las directrices establecidas en la guía de estilo web publicada por la UAB [12]. Esta guía incluye la paleta de colores oficial, la tipografía y la disposición del logo institucional, entre otros.

8 Testing

Durante el desarrollo y tras la implementación de cada servicio, se han realizado test unitarios en los puntos críticos de la arquitectura. Estos incluyen los objetos de valor para la validación de las reglas de negocio, los mapeos de las entidades para la verificación de la persistencia y transformación de tipos, y los servicios con lógica de negocio a través de *Mocks* de los diferentes repositorios.

9 Funcionalidades

Las distintas funcionalidades de la aplicación se han estructurado siguiendo un enfoque orientado a servicios, con el objetivo de construir un sistema modular. Cada servicio agrupa un conjunto de operaciones relacionadas con un dominio funcional de la aplicación. Esta estructuración permite aplicar el principio de responsabilidad única, facilitando así el mantenimiento y evolución de cada módulo.

9.1 Gestión de usuarios internos

La gestión de usuarios internos centraliza el proceso de autenticación y registro de usuarios internos de la universidad. En ella intervienen todos los usuarios internos de la UAB – administración de la plataforma, coordinación de prácticas, gestión académica y estudiantes – para su autenticación en la plataforma. Además, la administración puede registrar usuarios de cualquier rol, pudiendo asignar más de un rol a un mismo usuario; y la coordinación de prácticas puede registrar estudiantes de su estudio de forma individual o masiva.

Pese a que la autenticación inicial en la plataforma se

realiza mediante el Servicio de Control de Acceso Centralizado de la universidad, es necesario que los usuarios estén registrados en la aplicación para acceder a sus funcionalidades. El servicio implementa el inicio de sesión mediante la generación de JWT, utilizado posteriormente para la autenticación de peticiones.

Las acciones principales del servicio son:

- Autenticación de usuarios internos.
- Registro de estudiantes, coordinación de prácticas y gestión académica por parte de la administración de la plataforma.
- Registro de estudiantes de forma individual o masiva a través de un fichero Excel por parte de la coordinación de prácticas.
- Consulta de los usuarios registrados por parte de la administración y la coordinación de prácticas.

La interfaz de la gestión de usuarios internos se muestra en el Anexo A3 (figura 10).

9.2 Gestión de departamentos

El servicio de gestión de departamentos centraliza el registro y mantenimiento de los departamentos de un centro de la universidad. En él interviene únicamente la coordinación de prácticas, responsable de registrar y mantener actualizados los departamentos de su centro.

Las operaciones principales del servicio son:

- Registro de un departamento.
- Edición de los departamentos registrados.
- Consulta de los departamentos, incluyendo los tutores académicos asociados a cada uno.

La interfaz de la gestión de departamentos se muestra en el Anexo A3 (figura 11).

9.3 Gestión de tutores académicos

La gestión de tutores académicos centraliza el proceso de registro y mantenimiento de los profesores que tutorizan prácticas curriculares. Al igual que en la gestión de departamentos, en ella interviene únicamente la coordinación de prácticas, responsable de registrar y mantener actualizada la información de cada tutor en cuanto a su información básica y el listado de estudios que tutoriza.

Las operaciones principales son:

- Registro de un tutor académico.
- Edición de un tutor académico. En caso de dejar un tutor sin estudios, este pasará automáticamente a inactivo.
- Consulta de los tutores académicos.

La interfaz de la gestión de tutores se muestra en el Anexo A3 (figura 12).

9.4 Gestión de entidades

Servicio responsable de todas las operaciones relacionadas con las entidades. En él intervienen dos actores: la entidad para su registro, autenticación y creación de su perfil; y la administración de la plataforma para la validación de su registro.

El registro de una entidad debe ser validado por la administración para que esta pueda acceder a la plataforma. En caso de que su registro esté pendiente de validación o requerir modificaciones en sus datos podrá acceder a la plataforma únicamente para editar sus datos.

A diferencia de los usuarios internos, las entidades se autentican mediante usuario y contraseña, dado que no forman parte de la comunidad universitaria y no tienen acceso al Servicio de Control de Acceso Centralizado de la UAB.

Las operaciones principales son:

- Registro de una cuenta por parte de una entidad.
- Validación del registro de una entidad por parte de la administración.
- Autenticación por parte de la entidad.
- Edición del perfil por parte de la entidad.

La interfaz de la gestión de usuarios internos se muestra en el Anexo A3 (figura 13).

9.5 Gestión de estudiantes

La gestión de estudiantes centraliza el proceso de creación y edición del perfil del estudiante.

En el perfil el estudiante puede incluir: imagen, resumen, datos personales, formación académica, idiomas, links a redes sociales y currículum, permitiendo la subida de varios currículums.

Las operaciones principales del servicio son:

- Crear y editar el perfil por bloques.
- Consultar el perfil.

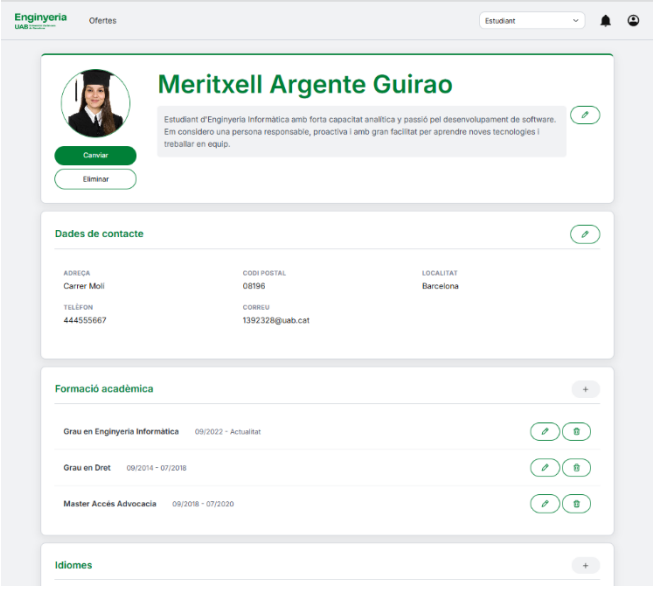

Figura 4. Perfil de estudiante

9.6 Gestión de ofertas

El servicio de gestión de ofertas centraliza el proceso de publicación y validación de las ofertas de prácticas curriculares. En él intervienen tres actores: la entidad, que registra y gestiona sus ofertas; la coordinación de prácticas, que las valida; y el estudiante, que las consulta y puede inscribirse en ellas.

Una oferta puede dirigirse a más de un estudio de forma simultánea. Para garantizar la independencia entre coordinaciones de prácticas, el estado de la oferta se gestiona de forma individual por estudio, por lo que la validación de una coordinación no queda condicionada por el trabajo de otra.

Cada oferta lleva asociado un estado: pendiente de validar, validada, requiere modificaciones, pausada o cerrada (consultar diagrama de estados en el Anexo A4, figura 14). El estado de la oferta será el que condicione las operaciones que se pueden realizar sobre ella. Sobre ello cabe destacar que cualquier edición de una oferta, independientemente del estado actual, restablece su estado a pendiente de validar para todos los estudios para garantizar su control y validación ante cualquier cambio que pudiera afectar a su validez.

Las acciones principales del servicio son:

- Registro y edición de ofertas por parte de la entidad.
- Validación y solicitud de cambios por parte de la coordinación de prácticas.
- Consulta de ofertas validadas, pausadas y cerradas por parte del estudiante.
- Consulta global de ofertas por parte de coordinación de prácticas y gestión académica.

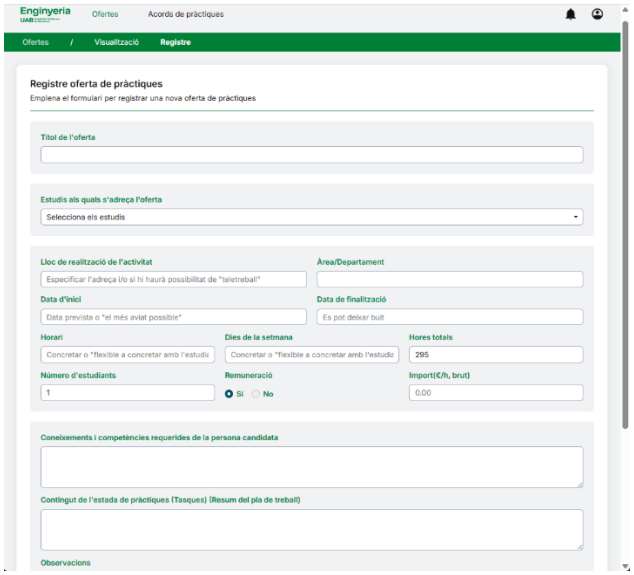

Figura 5. Registro oferta de prácticas

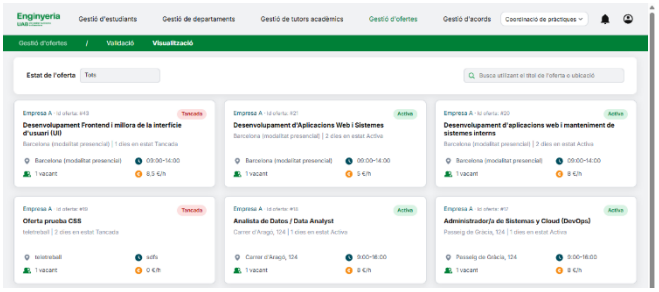

Figura 6. Visualización de ofertas

9.7 Gestión de inscripciones

La gestión de inscripciones recoge todas las operaciones relacionadas con las inscripciones a ofertas de prácticas por parte de los estudiantes. En ella intervienen dos actores: el estudiante que realiza la inscripción y consulta su estado; y la entidad, que las consulta y gestiona.

Al igual que las ofertas, las inscripciones llevan asociado un estado: enviada, en revisión, entrevista programada, entrevista realizada, seleccionada, rechazada, cancelada (consultar diagrama de estados en el Anexo A3, figura 15).

La entidad puede consultar las inscripciones realizadas para cada una de sus ofertas. Para ello se listan los estudiantes inscritos con la opción de visualizar el perfil de cada uno de ellos junto con el currículum y carta de motivación (esta última opcional). Asimismo, puede gestionar el estado de cada inscripción, con la opción de establecer notas públicas para el estudiante o privadas, para la propia entidad, en cada cambio de estado.

Por otro lado, el estudiante puede consultar el listado de ofertas en las que se ha inscrito, junto con el historial de su candidatura y la opción de cancelarla.

Las acciones principales del servicio son:

- Inscripción en una oferta por parte del estudiante.
- Consulta de las inscripciones en sus ofertas por parte de la entidad.
- Gestión del estado de las inscripciones por parte de la entidad.
- Consulta del historial de la candidatura por parte de la entidad y el estudiante.
- Cancelación de la inscripción por parte del estudiante.

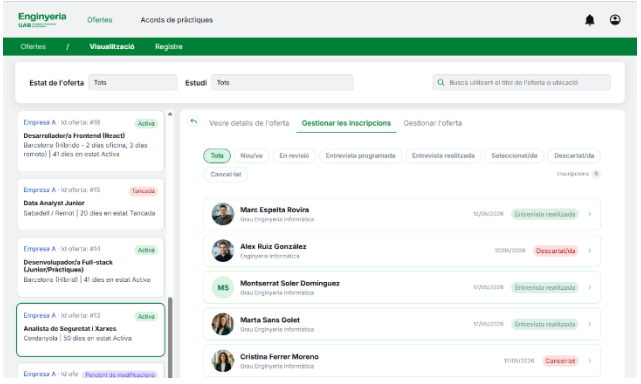

Figura 7. Visualización de las inscripciones en una oferta

9.8 Gestión de acuerdos

El servicio de gestión de acuerdos centraliza el proceso de registro y validación de acuerdos de prácticas curriculares. En él intervienen: la entidad para registrar el acuerdo; la coordinación de prácticas para validarlo, editarlo para añadir el DNI del estudiante o su mención y exportar los datos a un fichero Excel; y gestión académica para su consulta y la misma funcionalidad de exportación que la coordinación.

La entidad puede registrar un acuerdo que provenga de una oferta publicada previamente en la plataforma o no. En el primer caso, se da la opción a la entidad de indicar el identificador de la oferta para incluir los datos de ésta en el formulario facilitando así su registro. Asimismo, en el caso de introducir un NIU de un estudiante registrado en la plataforma se introducirán de forma automática sus datos de interés para el acuerdo como nombre, apellidos y estudio.

Los estados del acuerdo son: pendiente de validar, validado, requiere modificaciones, listo para formalizar y cerrado (consultar diagrama de estados en el Anexo A4, figura 16). Estos serán gestionados únicamente por la coordinación de prácticas.

Las operaciones principales del servicio son:

- Registro de acuerdos por parte de la entidad.
- Validación, edición y solicitud de cambios por parte de la coordinación de prácticas.
- Exportación de los datos del acuerdo a fichero Excel por parte de la coordinación de prácticas y gestión académica.
- Consulta de acuerdos por parte de gestión académica.

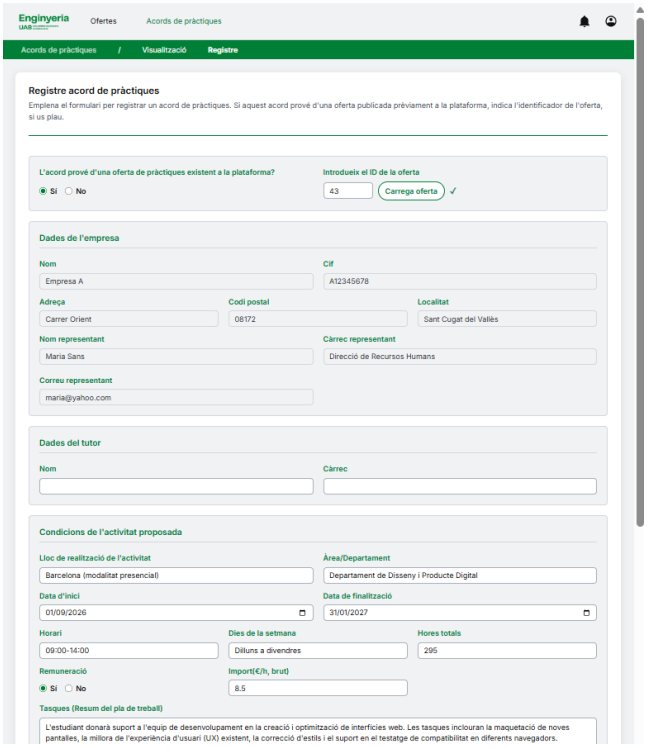

Figura 8. Registro acuerdo de prácticas

10 Resultados obtenidos

Tal como se ha visto en las funcionalidades implementadas, el desarrollo de ConnectUs ha permitido alcanzar el objetivo principal del proyecto: disponer de una herramienta institucional que centraliza la gestión de ofertas y acuerdos de prácticas curriculares en el grado de Ingeniería Informática de la EE, eliminando limitaciones derivadas del proceso manual actual.

A continuación, se valora el grado de consecución de cada uno de los objetivos específicos.

Eficiencia administrativa. La plataforma elimina tareas manuales repetitivas como la gestión de correos, publicación manual de ofertas y su mantenimiento en el Campus Virtual. La coordinación de prácticas dispone ahora de una herramienta centralizada para validar ofertas y acuerdos, visualizar sus cambios de estado, inscripciones y exportar los datos para la formalización del convenio.

Soporte a los estudiantes. Los estudiantes disponen de un espacio único donde consultar ofertas disponibles, inscribirse en ellas y hacer un seguimiento del estado de sus candidaturas.

Mejora de la colaboración entre la EE y las entidades. La plataforma facilita la comunicación entre la EE, estudiantes y entidades, permitiendo el registro de ofertas, gestión de las candidaturas y registro de acuerdos mediante formularios con validaciones automáticas.

Agilización del proceso. La eliminación de tareas manuales reduce el tiempo entre la publicación de una oferta y su asignación al estudiante.

Trazabilidad e información actualizada. Las ofertas, inscripciones y acuerdos cuentan con estados y transiciones controladas, lo que permite conocer su evolución. Además, la información se actualiza en tiempo real, evitando inconsistencias o acceso a ofertas ya cubiertas.

Escalabilidad. El diseño de la solución permite incorporar nuevos estudios y centros de la universidad sin modificar la lógica del sistema, facilitando así su evolución futura.

11 Conclusiones

En el apartado anterior se han expuesto los resultados obtenidos tras el desarrollo del proyecto, cumpliendo estos con los objetivos que se definieron en su inicio.

La metodología de trabajo escogida, Scrum, ha permitido organizar el proyecto en funcionalidades principales, iniciando el desarrollo con aquellas más básicas e incorporando progresivamente mayor complejidad. Las reuniones previstas en este marco de trabajo han sido clave para detectar desviaciones en los requisitos y ajustar el desarrollo de forma temprana. Asimismo, el uso de Notion ha permitido llevar un control efectivo del proyecto a través del seguimiento de los sprints junto a las tareas y reuniones.

En cuanto a las tecnologías utilizadas, .NET, Blazor y PostgreSQL, han sido adecuadas para el desarrollo de la plataforma. La experiencia previa en estas tecnologías ha permitido seguir un ritmo óptimo de trabajo centrando los esfuerzos en el desarrollo, pero también con aprendizajes

como la automatización de la evolución de la base de datos a través de migraciones. Asimismo, la adopción de *Clean Architecture* ha permitido construir un software modular y fácilmente extensible a través de la incorporación de nuevos servicios.

No obstante, el proyecto ha presentado dificultades en la fase de despliegue derivadas de la inexperiencia en este ámbito y las limitaciones del servidor. La necesidad de crear contenedores Docker personalizados y las restricciones de permisos en el servidor supuso una inversión de tiempo no prevista inicialmente. Debido a estas limitaciones, se automatizó el despliegue para evitar la dependencia hacia el responsable del servidor durante el desarrollo, así como en su evolución futura.

El alcance del proyecto ha estado condicionado por el tiempo disponible, lo que ha llevado a priorizar las funcionalidades principales y dejando fuera algunas previstas inicialmente, como el sistema de notificaciones o la generación automática del convenio. Asimismo, para una futura puesta en marcha será necesario incorporar pruebas de integración y validación por perfiles de usuario.

En el futuro, ConnectUs puede evolucionar incorporando nuevas funcionalidades que permitan ampliar sus capacidades. Entre estas mejoras se plantea la posibilidad de que las entidades puedan configurar cambios de estado de sus ofertas mediante reglas configurables, notificaciones por correo, configuraciones específicas para cada estudio, o visualización por parte de la coordinación de prácticas de los perfiles de los estudiantes junto a las inscripciones para ofrecer soporte a quien lo necesite. Aun así, la solución actual establece una base que puede utilizarse mejorando el proceso actual.

Uso de la IA Generativa

Para la redacción de este informe se ha utilizado la herramienta ChatGPT para la revisión de texto, incluyendo la corrección de errores tipográficos y mejora del estilo gramatical. En cuanto al desarrollo del proyecto, se han utilizado herramientas como Claude y Gemini como soporte en la resolución de errores puntuales y en la implementación de estilos CSS en el frontend, así como en la configuración y resolución de incidencias en el despliegue de la aplicación en el servidor. El diseño de la solución, la selección de tecnologías y la implementación del sistema han sido realizadas por la estudiante.

Agradecimientos

Quiero dar las gracias a mi tutor, Jordi Pons, por su implicación en el proyecto, su disponibilidad y por ayudarme a dar forma a un proyecto que se ha desarrollado con mucha ilusión. También a Adrià Sánchez, responsable del servidor de Ingeniería Informática, por su ayuda en las tareas relacionadas con el despliegue en el servidor.

Por último, quiero dar las gracias a mi entorno más cercano. A mi pareja, David Sánchez, por acompañarme desde hace ya más de una década y por su apoyo en todas las decisiones que he tomado, personales y profesionales. Y a mi madre, Mercedes Guirao, por ser mi pilar y por recordarme siempre la importancia de buscar una profesión que me hiciera feliz y de la que pudiera sentirme orgullosa. No se me ocurre una mejor forma de cerrar esta etapa que agradeciéndole todo lo que me ha enseñado y todo lo que me ha dado.

Bibliografía

[1] LinkedIn, *Acerca de LinkedIn*. [En línea]. Disponible en: https://about.linkedin.com/es-es. Último acceso: 27/02/2026.

[2] Indeed, *Acerca de Indeed*. [En línea]. Disponible en: https://es.indeed.com/about. Último acceso: 27/02/2026.

[3] Infojobs, *Infojobs, siempre a mejor*. [En línea]. Disponible en: https://nosotros.infojobs.net/sobre-infojobs/. Último acceso: 27/02/2026

[4] Talentoteca, *Talentoteca by FUE: prácticas remuneradas, becas de postgrado y primer empleo para jóvenes*. [En línea]. Disponible en: https://www.talentoteca.es/sobre-talentoteca/. Último acceso: 27/02/2026.

[5] Nexus. [En línea]. Disponible en: https://nexus.uab.es/UAB/logon.do. Último acceso: 27/02/2026.

[6] Ken Schwaber y Jeff Sutherland, *La guía Scrum.* [En línea]. Disponible en: https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-European.pdf. Último acceso: 14/02/2026

[7] Robert C.Martin, *Clean Architecture: A Craftsman's Guide to Software Structure and Design*. Addison-Westley, 2017, Foreword.

[8] Microsoft, *Modelos de hospedaje Blazor en ASP.NET Core*. [En línea]. Disponible en: https://learn.microsoft.com/es-es/aspnet/core/blazor/hosting-models?view=aspnetcore-10.0. Último acceso: 07/03/2026

[9] Microsoft, *Entity Framework Core*. [En línea]. Disponible en: https://learn.microsoft.com/es-es/ef/core/. Último acceso: 11/03/2026.

[10] Microsoft, *Información general sobre migraciones*. [En línea]. Disponible en: https://learn.microsoft.com/es-es/ef/core/managing-schemas/migrations/?tabs=dotnet-core-cli. Último acceso: 20/03/2026

[11] Microsoft, *Dotnet watch*. [En línea]. Disponible en: https://learn.microsoft.com/es-es/dotnet/core/tools/dotnet-watch#hot-reload. Ultimo acceso: 15/03/2026.

[12] Universitat Autònoma de Barcelona, *Guia d'estil web*. [En línea]. Disponible en: https://www.uab.cat/doc/guia-estil-web. Último acceso: 05/04/2026.

ANEXO

A1. REQUISITOS FUNCIONALES DEL SISTEMA

Los requisitos que se exponen a continuación corresponden a la definición inicial del proyecto. Durante el desarrollo, algunos fueron revisados y ajustados como resultado de las reuniones con el cliente y la evolución natural del proyecto.

Tabla 2. Requisitos funcionales del sistema

Código	Descripción	Cumplimiento
RF-1	El sistema debe permitir iniciar sesión mediante el CAS de la UAB	Sí
RF-2	El sistema debe permitir escoger el rol al iniciar sesión en caso de que el usuario tenga asignado más de uno	Sí
RF-3	El sistema debe permitir visualizar todas las ofertas en estado "Validada".	Sí
RF-4	El sistema debe permitir visualizar todos los acuerdos en estado "Validado"	Sí
RF-5	El sistema debe permitir generar un archivo en formato Excel en el que consten los datos del Estudiante, entidad y prácticas registrados en un acuerdo ya validado	Sí
RF-6	El sistema debe permitir dar acceso a un Estudiante no matriculado en la asignatura "Prácticas externas", en cuyo caso únicamente tendrá permisos de visualización	No (*could*)
RF-7	El sistema debe permitir pausar una oferta en estado "Validada", indicando el motivo de la pausa.	Sí (sin indicación del motivo)
RF-8	El sistema debe permitir cerrar una oferta en estado "Pendiente de validación", "Validada" o "Pausada", indicando el motivo del cierre	Sí (sin indicación del motivo)
RF-9	El sistema debe permitir subir una plantilla de "Conveni específic de cooperació educativa per a la realització de practiques acadèmiques externes (curriculars) en entitats col·laboradores".	No (*could*)
RF-10	El sistema debe permitir generar un "Conveni específic de cooperació educativa per a la realització de practiques acadèmiques externes (curriculars) en entitats col·laboradores" con los datos del estudiante, entidad y las prácticas introducidas en la plataforma.	No (*could*)
RF-11	El sistema debe permitir al Administrador validar el registro de una entidad.	Sí
RF-12	El sistema debe permitir al Administrador el registro de un programa de estudios (grado o máster).	No (*should*)
RF-13	El sistema debe permitir al Administrador el registro de un usuario con rol de Coordinación de Prácticas, que en todo caso deberá estar asociado a un grado o máster.	Sí
RF-14	El sistema debe permitir a la Coordinación de Prácticas registrar un tutor académico	Sí
RF-15	El sistema debe permitir a la Coordinación de Prácticas visualizar las ofertas de su grado cualquiera que sea su estado	Sí
RF-16	El sistema debe permitir a la Coordinación de Prácticas visualizar los acuerdos de su grado o máster, cualquiera que sea su estado	Sí
RF-17	El sistema debe permitir a la Coordinación de Prácticas validar o rechazar las ofertas en estado "Pendiente de validación".	Sí
RF-18	El sistema debe permitir a la Coordinación de Prácticas validar o rechazar los acuerdos en estado "Pendiente de validación".	Sí
RF-19	El sistema debe permitir a la Coordinación de Prácticas filtrar las ofertas por: estado, ámbito, ubicación y texto	Sí
RF-20	El sistema debe permitir a la Coordinación de Prácticas filtrar los acuerdos por estado.	Sí
RF-21	El sistema debe permitir a la Coordinación de Prácticas visualizar los perfiles de los Estudiantes del grado o máster que coordina. En cada perfil se indicarán las vacantes a los que el Estudiante se ha inscrito.	No (*could*)
RF-22	El sistema debe permitir al Estudiante editar su perfil donde indicará nombre completo, grado o máster y currículums, además de aquella información adicional que considere oportuna.	Sí

RF-23	El sistema debe permitir al Estudiante almacenar ofertas en estado "Validada" a favoritos	No (*should*)
RF-24	El sistema debe permitir al Estudiante filtrar las ofertas en estado "Validada" por favoritos, inscritas, ámbito, ubicación y texto.	Sí (salvo favoritos)
RF-25	El sistema debe permitir al Estudiante inscribirse en una vacante en estado "Disponible" debiendo indicar qué currículum de los que constan en su perfil desea enviar y permitiendo la subida de un archivo en formato pdf como carta de presentación o motivación	Sí (la carta de motivación se incorpora en texto, no como archivo)
RF-26	El sistema debe permitir el envío de una notificación a las Entidades cuando un Estudiante se inscriba en una de sus vacantes, siempre y cuando la Entidad tenga activada la opción de recibir notificaciones.	No (*must*)
RF-27	El sistema debe permitir al Estudiante eliminar la inscripción a una vacante en estado "Disponible" o "Pendiente de aceptación".	Sí
RF-28	El sistema debe permitir el envío de una notificación a las Entidades cuando un Estudiante elimine su inscripción en una de sus vacantes, siempre y cuando la Entidad tenga activada la opción de recibir notificaciones.	No (*should*)
RF-29	El sistema debe permitir al Estudiante aceptar o rechazar la asignación de una vacante en estado "Pendiente de aceptación".	No (*must*)
RF-30	El sistema debe permitir el registro para la creación de una cuenta de Entidad.	Sí
RF-31	El sistema debe permitir a la Entidad editar los datos de la cuenta.	Sí
RF-32	El sistema debe permitir a la Entidad iniciar sesión con una cuenta ya creada.	Sí
RF-33	El sistema debe permitir a la Entidad recuperar la contraseña de una cuenta ya creada	Sí
RF-34	El sistema debe permitir a la Entidad escoger el sistema de notificaciones deseado.	No (*should*)
RF-35	El sistema debe permitir a la Entidad registrar sus datos de contacto, indicando en todo caso su página web	Sí
RF-36	El sistema debe permitir a la Entidad registrar una oferta de prácticas.	Sí
RF-37	El sistema debe permitir a la Entidad editar una oferta en estado "Pendiente de validación", "Validada", "Requiere modificaciones" o "Pausada".	Sí
RF-38	El sistema debe permitir a la Entidad asignar una vacante en estado "Disponible" asociada a una oferta en estado "Validada" a un Estudiante.	No (*must*)
RF-39	El sistema debe permitir a la Entidad visualizar únicamente sus propias ofertas y filtrarlas por estado	Sí
RF-40	El sistema debe permitir a la Entidad visualizar un listado de los Estudiantes inscritos en una vacante	Sí
RF-41	El sistema debe permitir a la Entidad visualizar el perfil de los Estudiantes inscritos en una vacante, así como el currículum seleccionado y la carta de presentación o motivación, si el Estudiante la ha adjuntado en el momento de la inscripción	Sí
RF-42	El sistema debe permitir a la Entidad rechazar un Estudiante inscrito en una vacante, indicando el motivo del rechazo.	Sí
RF-43	El sistema debe permitir a la Entidad registrar un acuerdo de prácticas	Sí
RF-44	El sistema debe permitir a la Entidad aceptar o rechazar las condiciones correspondientes para poder registrar un acuerdo. En caso de rechazarlas no se aceptará el registro del acuerdo.	No (*must*)
RF-45	El sistema debe permitir a la Entidad editar un acuerdo de prácticas en estado "Pendiente de validación" o "Requiere modificaciones".	Sí

A2. DIAGRAMA BASE DE DATOS

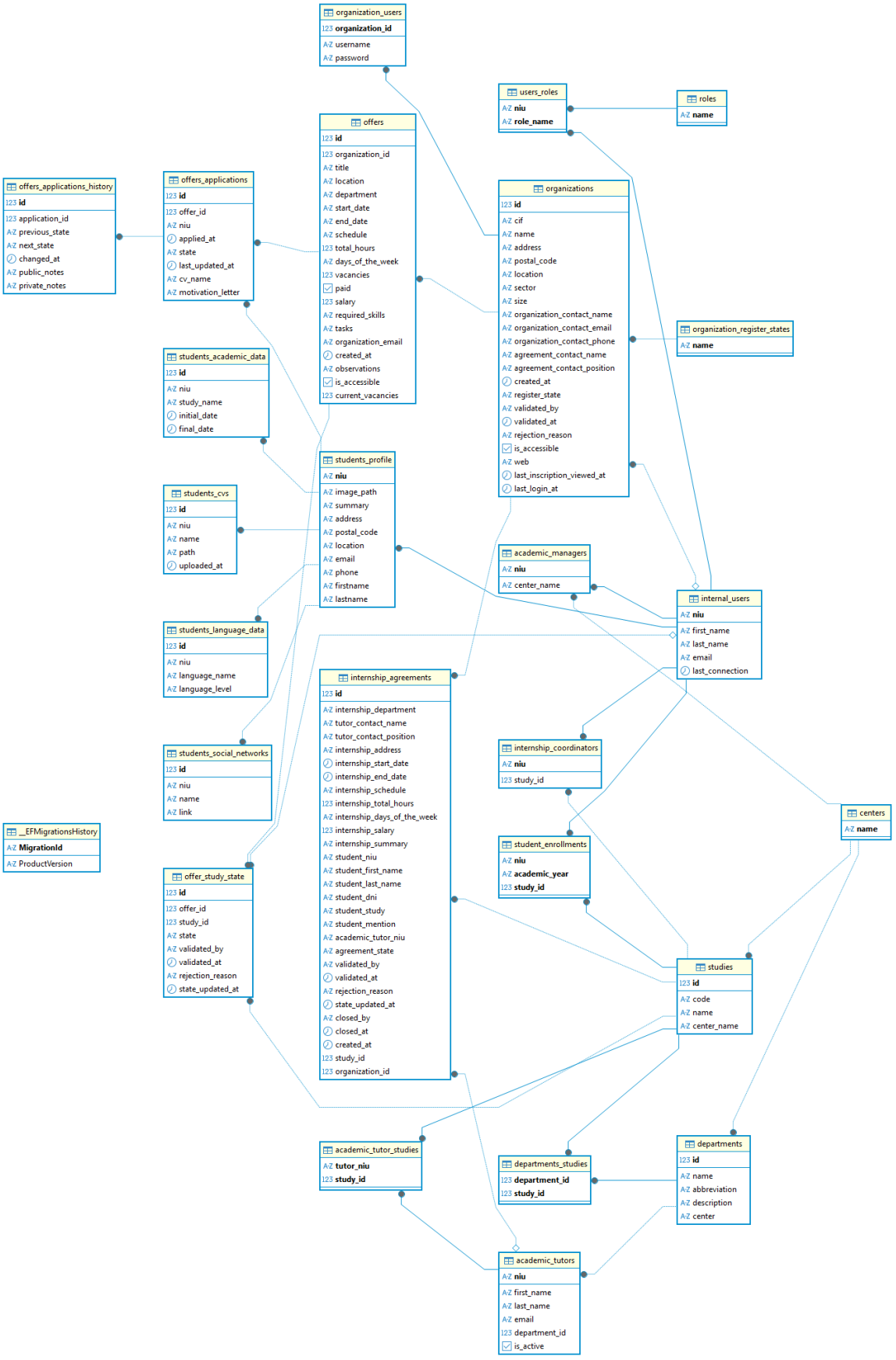

Figura 9. Diagrama base de datos

A3. APLICACIÓN WEB

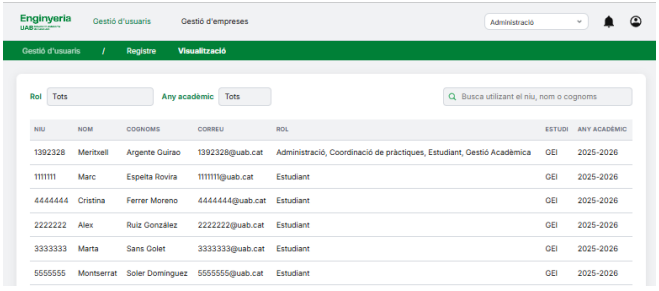

Figura 10. Visualización de usuarios internos

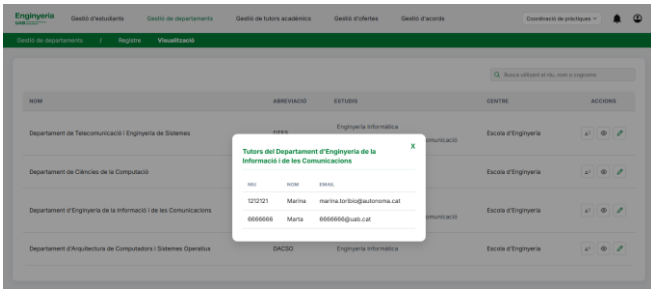

Figura 11. Visualización de tutores de un departamento

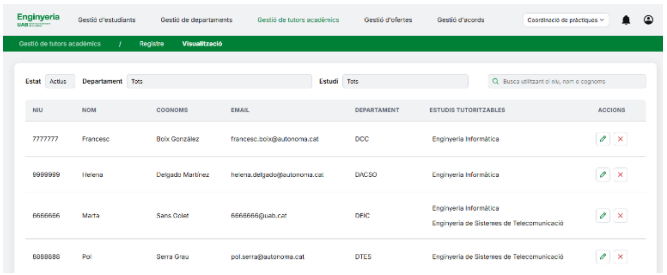

Figura 12. Visualización de tutores

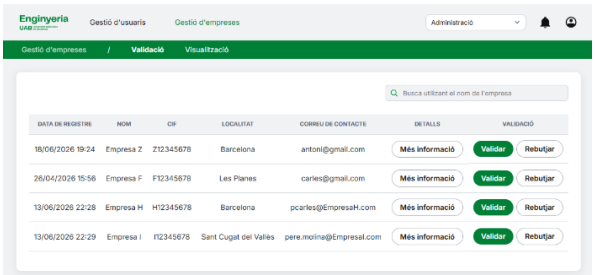

Figura 13. Visualización de entidades (validación)

A4. DIAGRAMA ESTADOS DE OFERTA, INSCRIPCIÓN Y ACUERDO

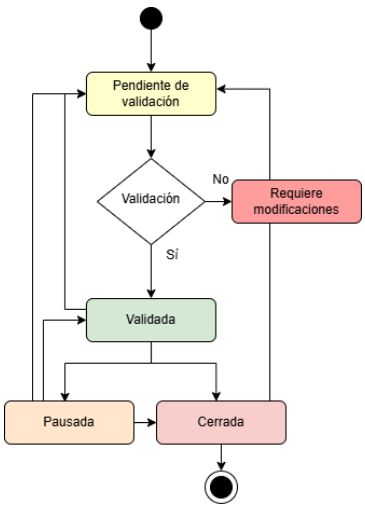

Figura 14. Diagrama de estados de Oferta

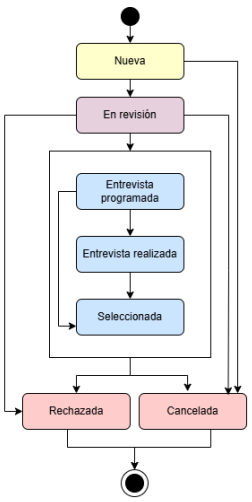

Figura 15. Diagrama de estados de Inscripción

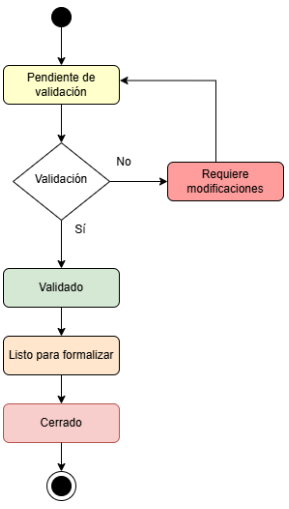

Figura 16. Diagrama de estados de Acuerdo