



SISTEMA WEB PARA LA BÚSQUEDA DE CURSOS DE POSTGRADOS ESPAÑOLES

Memoria de Proyecto Final de Máster

Máster Universitario en Desarrollo de Sitios y Aplicaciones Web

Área de Informática, Multimedia y Comunicación

Autor: Santiago Andrés Bruno Pérez

Consultor: Carles Arnal Castello

Profesor: César Pablo Córcoles Briongos

Enero del 2022

Créditos/Copyright



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

Abstracto

El presente proyecto nace por la necesidad que tienen los profesionales de buscar cursos de especialización o máster a nivel universitario para crecer profesionalmente, debido a que no es tan fácil buscar esta información de una manera rápida y con las características importantes que el profesional necesita, tal sea como son créditos, duración, modalidad, idioma(s) de impartición, entre otros.

Es por ello que el objetivo de este proyecto es facilitar la búsqueda de cursos a los alumnos que están interesados en realizar un máster o postgrado en escuelas y/o universidades en España.

Esta aplicación contiene un diseño responsive, adaptable a diferentes tamaños de pantallas tales como ordenadores, tabletas o móviles.

Para el desarrollo de la aplicación se ha optado por el uso de Angular para el front-end, NodeJS y Express para el backend y MySQL como base de datos.

Palabras claves: Angular, Express, Node, Buscador, Postgrado, Máster, España.

Abstract (English)

This project was born from the need for professionals to search specialization or master courses at the university level to grow professionally, because they are not easy to find this information quickly with the important features that the professional needs, such as such as credits, duration, modality, language (s) of instruction, among others.

That is the reason why the objective of this project is to facilitate the search for courses for students who are interested in taking a master's or postgraduate degree in Spain.

This application contains a responsive design adaptable to different screen sizes such as computers, tablets or mobiles.

For the development of the application, we have chosen to use Angular for the front-end, NodeJS and Express for the backend and MySQL as a database.

Keywords: Angular, Express, Node, SearchEngine, Postgraduate, Master, Spain

Agradecimientos

A Dios, por darme un día más de vida, dándome la oportunidad de poder lograr un objetivo más.

A mis padres, por sus constantes apoyos en cada etapa profesional de mi vida.

A mis profesores del master, que me han ayudado a crecer profesionalmente, en especial a mi asesor Carles Arnal, porque sin su ayuda esto no hubiera sido posible.

Santiago Andrés Bruno Pérez

ÍNDICE

1.	Introducción	10
1.1	Contexto	10
1.2	Justificación.....	11
2.	Descripción.....	12
3.	Objetivos.....	14
3.1	Objetivos principales.....	14
3.2	Objetivos secundarios	14
4.	Marco teórico/Escenario	15
5.	Perfiles de usuario	17
6.	Contenidos	18
7.	Metodología.....	20
8.	Arquitectura de la aplicación/sistema/servicio	21
8.1	Front-end / Back-end.....	21
8.2	Diagrama de base de datos	24
9.	Plataforma de desarrollo.....	25
10.	Planificación	26
11.	Diagramas UML	28
11.1	Diagrama de clases	28
11.3	Diagramas de casos de uso.....	29
12.	Prototipos	35
13.	Análisis de mercado.....	58
13.1	Contexto social y de mercado.....	58
13.2	Productos similares	58
13.3	Estudio de mercado.....	58
14.	Viabilidad	60
14.1	Viabilidad técnica.....	60
14.2	Viabilidad legal:.....	60
14.3	Viabilidad económica:.....	60
14.4	Viabilidad de mercado del proyecto.....	60
15.	Presupuesto	62
16.	Usabilidad/UX.....	64
16.1	Patrones de Layout.....	64
16.2	Patrones de diseño de interfaces.....	65

17. Tests	66
17.1 Pruebas unitarias.....	66
18. Seguridad	67
19. Requisitos de instalación/implantación/uso	69
20. Instrucciones de instalación/implantación	70
20.1 Back-end.....	70
20.1.1 Local	70
20.1.2 Servidor - AWS.....	70
20.2 Front-end.....	72
20.2.1 Local	72
20.2.2 Servidor-AWS.....	72
21. Instrucciones de uso	74
21.1 Pantalla de inicio (Home)	74
21.2 Búsqueda y filtrado de cursos	74
21.3 Detalle del curso	75
21.4 Solicitar información del curso	75
21.5 Registrar Universidad	76
21.6 Inicio de sesión	76
21.7 Dashboard de la Universidad	76
21.8 Mis cursos – Universidad	76
21.9 Nuevo curso – Universidad	77
21.10. Nuevos alumnos interesados.....	77
22. Proyección a futuro	78
23. Conclusiones.....	79
Anexo 1. Entregables del proyecto	80
Anexo 2. Código fuente (extractos)	81
Anexo 3. Librerías/Código externo utilizado.....	88
Anexo 4. Bibliografía	90

ÍNDICE DE FIGURAS

Figura 1- Distribución del número de estudiantes matriculados en el Sistema Universitario.....	11
Figura 2 - Diagrama de la metodología encascada con retroalimentación.	20
Figura 3. Arquitectura de la aplicación.	23
Figura 4- Diagrama de base de datos	24
Figura 5- Planificación – Diagrama de Gantt.....	27
Figura 6- Diagrama de clases.....	28
Figura 7- Diagrama de caso de uso: Realizar búsqueda de curso	29
Figura 8- Diagrama de caso uso: Registrar universidad y Gestionar cursos	31
Figura 9 - Pantalla de inicio. Versión escritorio.	35
Figura 10 - Pantalla de inicio. Versión tablet.....	36
Figura 11 - Pantalla de inicio. Versión móvil.	36
Figura 12 – Búsqueda y filtrado de curso. Versión escritorio.	37
Figura 13 – Búsqueda y filtrado de curso. Versión tablet.	38
Figura 14 – Búsqueda y filtrado de curso. Filtro activado y no activado. Versión móvil.....	39
Figura 15 – Detalle del curso. Versión escritorio.....	40
Figura 16 – Detalle del curso. Versión tablet.....	41
Figura 17 – Detalle del curso. Versión móvil.....	41
Figura 18 – Solicitar información del curso. Versión escritorio.....	42
Figura 19 – Solicitar información del curso. Versión tablet.....	43
Figura 20 – Solicitar información del curso. Versión móvil.	43
Figura 21 – Registrar Universidad. Versión escritorio.....	44
Figura 22 – Registrar Universidad. Versión tablet.....	45
Figura 23 – Registrar Universidad.	45
Figura 24 – Inicio de sesión. Versión escritorio.....	46
Figura 25 – Inicio de sesión. Versión tablet.....	47
Figura 26 – Inicio de sesión.	47
Figura 27 – Dashboard. Versión escritorio.	48
Figura 28 – Dashboard. Versión tablet.....	49
Figura 29 – Dashboard. Versión escritorio.	49
Figura 30 – Mis cursos. Versión escritorio.	50

Figura 31 – Mis cursos. Versión tablet.	51
Figura 32 – Mis cursos. Versión móvil.....	51
Figura 33 – Nuevo y Editar curso. Versión tablet.....	52
Figura 34 – Nuevo y Editar curso. Versión tablet.....	53
Figura 35 – Nuevo y editar curso. Versión móvil.....	53
Figura 36 – Ver curso. Versión escritorio.	54
Figura 37 – Ver curso. Versión tablet.	55
Figura 38 – Ver curso. Versión móvil.....	55
Figura 39 –Alumnos interesados. Versión escritorio.....	56
Figura 40 –Alumnos interesados. Versión tablet.....	57
Figura 41 –Alumnos interesados.	57
Figura 42 – Prueba unitaria – Course test.....	66
Figura 43 –Token JWT - Registro, Login, Acceso Recurso.....	68
Figura 44 – Variables de entorno del backend	71
Figura 45 – Política bucket – AWS S3 (Frontend).....	72

ÍNDICE DE TABLAS

Tabla 1- Evolución del número de estudiantes matriculados en Máster por rama	10
Tabla 2- Número de estudiantes extranjeros en el Sistema Universitario Español.....	10
Tabla 3- Perfiles de usuario	17
Tabla 4- Software utilizado para el desarrollo	25
Tabla 5- Hardware utilizado para el desarrollo.....	25
Tabla 6 – Planificación.....	26
Tabla 7- Caso de uso: Realizar búsqueda de curso.....	31
Tabla 8- Caso de uso: Registrar universidad y Gestionar cursos.....	34
Tabla 9- Herramienta estratégica DAFO	61
Tabla 10- Presupuesto del equipo humano.	62
Tabla 11- Presupuesto del equipo técnico.	63
Tabla 12- Presupuesto de los recursos adicionales.....	63
Tabla 13- Presupuesto total.....	63
Tabla 14- Autorización de recursos para Universidad - JWT	67

1. Introducción

1.1 Contexto

Según el *Ministerio de Universidades, en el Datos y cifras del Sistema Universitario Español, publicación 2020-2021*, el número de estudiantes matriculados de Máster en el periodo 2019-20 es de 234.214, lo que representa un crecimiento del 64,8% respecto al número de alumnos 2014-15 y una tasa anual del 7,5%, representando un crecimiento sostenido en el tiempo.

	Curso académico					Tasa de variación	
	2014-15	2018-19	2019-20 ⁽¹⁾			Anual	2019-20 /2014-15
			Total	Univ. públicas	Univ. privadas		
Total	142.156	217.840	234.214	138.414	95.800	7,5%	64,8%
Rama de enseñanza							
Ciencias Sociales y Jurídicas	81.117	126.147	136.640	66.843	69.797	8,3%	68,4%
Ingeniería y Arquitectura	23.727	41.246	44.177	33.432	10.745	7,1%	86,2%
Artes y Humanidades	13.839	17.706	18.212	14.436	3.776	2,9%	31,6%
Ciencias de la Salud	15.803	23.386	25.358	14.370	10.988	8,4%	60,5%
Ciencias	7.670	9.355	9.827	9.333	494	5,0%	28,1%

Tabla 1- Evolución del número de estudiantes matriculados en Máster por rama de enseñanza y tipo de universidad.
(Ministerio de Universidades, 2020-2021)

Por otro lado, la distribución de estudiantes que se matriculan en un Máster tanto nacional como extranjero, indica que este nivel académico es muy demandado. Representando un total del 23,1% del periodo 2019-2020 en el Sistema Universitario Español.

	Matriculados ⁽¹⁾ (2019-20)				Egresados (2018-19)			
	Total	Extranjeros			Total	Extranjeros		
		Total	% total	UE-28		Total	% total	UE-28
Total	1.633.358	154.428	9,5%	47.304	313.098	34.229	10,9%	9.217
Estudiantes de Grado y 1º y 2º ciclo	1.309.791	75.925	5,8%	33.993	189.438	7.756	4,1%	4.226
Estudiantes en Máster	234.214	54.004	23,1%	7.720	114.320	24.140	21,1%	4.389
Estudiantes de Doctorado RD 99/2011	89.353	24.499	27,4%	5.591	9.340	2.333	25,0%	602

Tabla 2- Número de estudiantes extranjeros en el Sistema Universitario Español.
(Ministerio de Universidades, 2020-2021)

Entre ellos vemos que un gran mercado es “EEUU y Canadá” con “América latina y Caribe”.

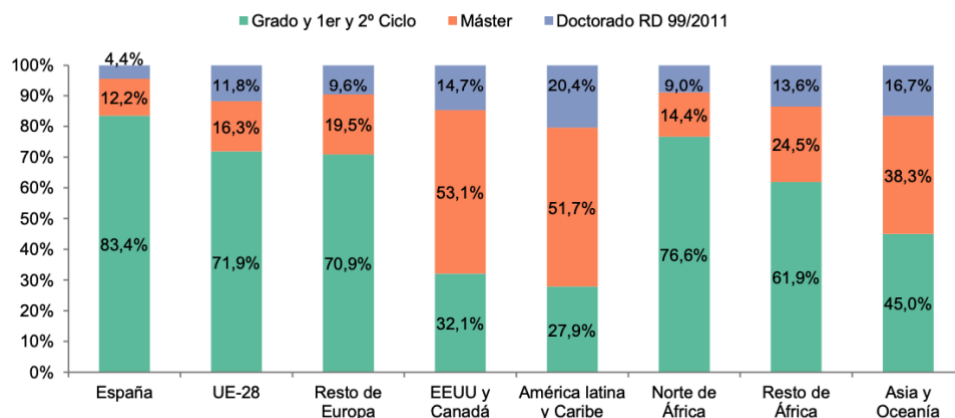


Figura 1- Distribución del número de estudiantes matriculados en el Sistema Universitario por nacionalidad y nivel de estudios. Curso 2019-2020. (Ministerio de Universidades, 2020-2021)

En conclusión, existe un gran mercado en el sector educativo de máster, por lo que es importante que las herramientas de búsqueda de cursos de postgrado tengan un proceso sencillo y eficaz.

1.2 Justificación

Debido a que la información de distintas escuelas está muy dispersa y cada uno tiene una navegación de usuario distinta, que algunas veces puede resultar muy complejo, en este proyecto se desarrollará una aplicación web que ayuda a los profesionales a buscar un máster o postgrado, según sus necesidades.

Basándose en dos enfoques, primero por el lado del alumno que es el profesional que tiene la necesidad de llevar un curso o especialización de postgrado. Para ellos la problemática es que la información es muy dispersa, es decir que se tiene que ingresar a la página web de cada escuela universitaria y mirar el curso en el que está interesado, generando esto largas horas de búsqueda.

Al momento de hacer una búsqueda, puede resultar ineficiente ya que en su mayoría no se muestran datos relevantes como duración del programa, créditos, modalidad, idiomas, entre otros.

Por otro lado, se encuentran las escuelas universitarias tanto públicas y privadas, que cada vez están buscando más canales para la captación de alumnos, para generar más oportunidades de ventas.

2. Descripción

Se propone como trabajo de fin de máster, de ahora en adelante TFM, una aplicación web para la búsqueda de máster o postgrados españoles. Dicha aplicación estará destinada principalmente a los estudiantes y/o profesionales que hayan terminado una carrera universitaria (técnicas, diplomaturas o grado) y estén en búsqueda de algún curso de postgrado o máster en alguna universidad tanto pública, como privada o cualquier escuela o institución de prestigio.

Esta aplicación les ayudará a encontrar un curso de acuerdo a su necesidad, en términos como rama de enseñanza, tipo de estudio, modalidad, provincia, duración, idiomas, precio. Así, evitando búsquedas prolongadas e ineficientes en diferentes páginas de universidad o escuelas. Es decir, todo se encontrará en una sola página, lo cual permitirá al estudiante realizar una comparación de los cursos bajo los criterios seleccionados por el estudiante.

Por otro lado, facilitará a las universidades o escuelas el poder registrarse y agregar cursos que se estén ofertando actualmente. Y así, poder contactar con el alumno por medio de correo o móvil.

Cabe destacar con respecto a la base de datos, se tomará como base el Registro de Universidades, Centros y Títulos (RUCT) el cual permite realizar una búsqueda por el nivel académico “master” y extraerlo en un Excel. Y posteriormente agregar manualmente todos los campos extras como créditos, duración, modalidad, idioma(s), entre otros. Además, para que dicha creación de la base de datos no sea tan exhaustiva, se acotará la “muestra” a solo las ramas de enseñanza de “Ingeniería y Arquitectura” y “Ciencias Sociales y Jurídicas” tales como: Master en Ingeniería Informática e Industrial y MBA (Máster en Administración y Dirección de Empresas). No obstante, esto no es ningún impedimento para que la base de datos vaya creciendo progresivamente.

Las funcionalidades principales del proyecto serán:

- Login de las universidades o escuelas.
- Creación del perfil visitante: Búsqueda de máster o cursos de postgrado según criterios del estudiante.
 - Buscador de curso
 - Filtrado de cursos

- Detalle del curso
- Solicitar información de un curso
- Creación del perfil universidades/escuelas
- Gestión de cursos y alumnos interesados para el perfil universidades/escuelas.

Para una descripción más detallada sobre las funcionalidades antes mencionadas, se debe acudir al documento anexo PAC_FINAL_funcionalidades_BrunoPerez_SantiagoAndres.pdf .

3. Objetivos

A continuación, se detallan los objetivos que pretende alcanzar la aplicación. Están conformados de objetivos principales, que describen la razón de ser del proyecto; y los objetivos secundarios, que son las características deseables de la aplicación, pero no determinan el objetivo principal.

3.1 Objetivos principales

Los objetivos clave del proyecto son los siguientes:

- Facilitar la búsqueda de cursos a los alumnos que están interesados en realizar un máster o postgrado en escuelas y/o universidades en España.
- Permitir que el alumno realice filtros de búsqueda relevantes tales como: Rama de enseñanza, tipo de estudio, modalidad, duración, idiomas, precio.
- Gestionar perfiles en la aplicación según sus necesidades:
 - Perfil visitante: permitir que cualquier persona pueda realizar una búsqueda de cursos.
 - Perfil de universidad y/o escuelas: ofrecer cursos de máster o postgrado para poder llegar a más alumnos.

3.2 Objetivos secundarios

Los objetivos adicionales que enriquecen el proyecto:

- Ofrecer una interfaz amigable basada en la usabilidad y experiencia de usuario.
- Facilitar el uso de la aplicación en distintos tipos de dispositivos como móvil, tableta y escritorio. Es decir, que sea igual de sencillo realizar la búsqueda de cursos, en un móvil como en un ordenador.

4. Marco teórico/Escenario

Actualmente, no es fácil encontrar una herramienta integral de búsqueda para los profesionales que están interesados en realizar un máster o postgrado en escuelas y/o universidades en España. Generalmente, cuando una escuela apertura un curso, realiza la publicación de la información y detalle del curso dentro de la misma página web de la entidad educativa; por lo tanto, si un alumno está interesado en estudiar un curso de postgrado tiene la necesidad de ingresar a cada una de las páginas web de las escuelas. Sin embargo, existe la posibilidad de que el alumno no se llegue a enterar de cursos nuevos que se encuentran en el mercado porque la búsqueda le resultó complicada o solo visitó la página web de una o dos escuelas; es decir, existe una oportunidad de mejora donde las escuelas puedan llegar a más alumnos interesados y los alumnos puedan tener más herramientas para poder elegir adecuadamente, el curso que más les interese en el mercado.

Los alumnos suelen buscar cursos de postgrado en las siguientes herramientas:

- **Páginas webs de Universidades y/o Escuelas:** Se encuentra información de cursos impartidos únicamente por esa escuela, información detallada y directa; sin embargo, cada página tiene una estructura de búsqueda distinta.
- **EMagister:** Es un portal web donde se puede encontrar cursos, carreras y postgrados de diferentes universidades, institutos o escuelas; sin embargo, no siempre se puede encontrar lo que el profesional necesita, es decir, este no te muestra el abanico real de cursos en todas las instituciones existentes sino que solo muestra la información de aquellas que han pagado para mostrarse como prioridad en la búsqueda, y así perdiendo el foco de la primera página donde debería mostrarse todo lo relacionado con la búsqueda.
- **Educaweb:** Es un portal web especializado en educación que ofrece un servicio de información y orientación académica dirigido a estudiantes y profesionales. Su foco principal es la asesoría educativa como charlas, ferias de empleo, entre otros; sin embargo, igual que EMagister, está orientado al marketing en línea para el sector educativo.

El problema general de estos buscadores, es que solo muestran las universidades, escuelas privadas, nuevas o simplemente aquellas que paguen más por sus anuncios, es decir, no necesariamente son las mejores escuelas o las que podría preferir el profesional si tuviera toda la información a mano, esto es por el fin comercial, sin importar la real necesidad del profesional en búsqueda de un curso de postgrado, que muchas veces prefiere estudiar en una buena escuela pública a un precio justo.

5. Perfiles de usuario

Para el desarrollo de los perfiles de usuario, se realizó una tabla resumen que se basa en los 5 pasos básicos de la investigación de usuarios que propone (Russ Unger, 2009).

1. Definir los grupos de usuario principales.
2. Planificación de la partición del usuario.
3. Realizar la investigación.
4. Validar las definiciones de los grupos de usuarios.
5. Generar requisitos de usuario.

PERFILES DE USUARIO			
Perfil	Función en el sistema	Personalidad	Usuario
Visitante	Es el usuario que se encuentra en la búsqueda de un curso de postgrado. Este perfil puede buscar, filtrar por criterios y solicitar más información de los cursos.	Es un profesional graduado de cualquier universidad publica/privada nacional o extranjera, que es perseverante, curioso, visionario, y con deseo de crecer profesionalmente.	Jordi Pérez, 26 años. Graduado en Ingeniería Informática. Soltero
Universidad	Es el usuario representante de una institución educativa, encargado de registrar la información de la universidad. Este perfil podrá gestionar los cursos de postgrado que se ofrecerán, así como el listado de alumnos interesados	Es una institución educativa de prestigio, que continuamente refuerza, actualiza sus programas de postgrados y busca la inscripción de nuevos profesionales.	Universidad Oberta de Catalunya 25 años Innovadora

Tabla 3- Perfiles de usuario

6. Contenidos

La aplicación web tendrá una estructura clásica es decir cabecera, cuerpo y pie de página:

- **Cabecera:** Encontraremos el logo de la aplicación web, el buscador de cursos y la zona de acceso a la aplicación.
- **Cuerpo:** Se mostrará en el *perfil visitante* (público), un filtro de máster o cursos de postgrado según criterios del estudiante tales como rama de enseñanza, tipo de estudio, modalidad, provincia, duración, idiomas, precio. En el *perfil de universidades y/o escuelas* se podrá registrar, listar, eliminar y actualizar un curso.
- **Pie de página:** Dispondrá de enlaces como quiénes somos, contáctanos, información legal y preguntas frecuentes.

Las vistas que necesitaremos implementar serán:

- **Home:**

Es la página inicial de la aplicación. Donde se podrá encontrar un buscador simple. Además, se mostrará una sección con los másteres más demandados, que serán aquellos en los que los alumnos piden más información.
- **Perfil visitante:**
 - Es la parte que se mostrará a todo el público, donde cualquiera podrá buscar un curso de postgrado o máster.
 - Adicionalmente se mostrará un Menú a la izquierda para filtrar detalladamente la búsqueda de los cursos, dando como resultado los cursos filtrados.
 - Una vez identificado el curso de interés, el visitante podrá seleccionarlo para visualizar más detalle del curso y también habrá una opción para solicitar información del mismo.

- **Perfil de universidad y/o escuelas**
 - Es donde se podrá gestionar los cursos que ofrece la escuela y/o universidad, es decir, listar, crear, editar, eliminar. La estructura de esta página estará conformada por las siguientes secciones: **“Mis cursos”**, **“Nuevo Curso”**, **“Alumnos interesados”**, donde la primera sección se podrá listar los campos principales de los cursos y realizar las funcionalidades como visualizar, editar, eliminar dicho curso; en la segunda sección, se visualizará un formulario para la creación de un nuevo curso. Además, en dicho formulario se podrá subir un archivo con formato permitido (.pdf, .doc, .docx) para el plan de estudios. Y, por último, en la tercera sección, se mostrará una lista con los alumnos interesados, que serán aquellos que están solicitando información del curso.

- **Formulario de Registro de escuelas/universidades:**
 - Aquí se podrá registrar a las escuelas/universidades con los datos necesarios como son nombre, año de creación, dirección, provincia, código postal, email, teléfono, username y password e inclusive subir una imagen de la institución.

- **Login de escuelas y/o universidades**
 - Se comprobará el acceso de las universidades registradas, haciendo una llamada a la base de datos para su respectiva autenticación.

- **Contáctanos**
 - Se podrá visualizar los datos para contactar con los administradores de la aplicación.

7. Metodología

La metodología que se ha decidido llevar a cabo es la metodología en cascada con retroalimentación, basado en la propuesta de (Royce, 1998), debido a que es un proyecto académico.

Esta metodología nos indica que no se puede avanzar a la siguiente fase hasta que se haya completado dicha fase, y en caso de error se puede volver a la anterior fase.

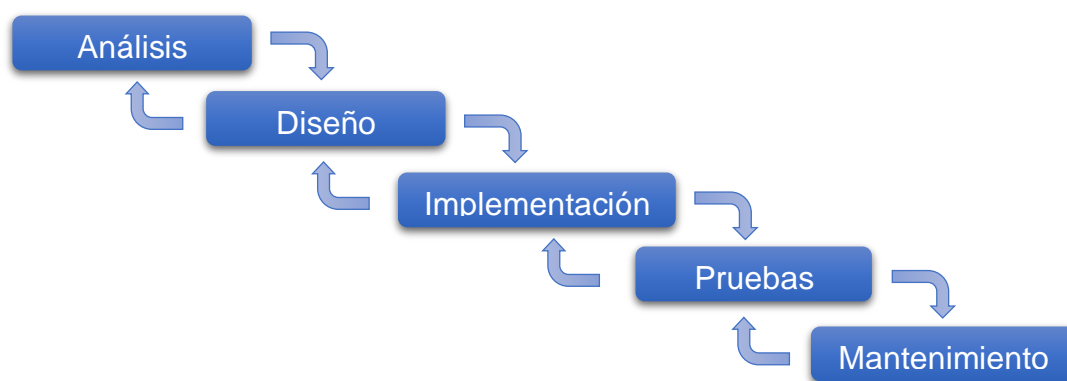


Figura 2 - Diagrama de la metodología encascada con retroalimentación.

Como podemos observar el diagrama consta de 5 fases:

- Análisis: En esta fase inicial, se realizan los requisitos de la aplicación para su respectivo análisis.
- Diseño: Fase en la que se crean especificaciones técnicas de diseño como son: diagrama de clases, casos de uso y los prototipos de las pantallas de la aplicación.
- Implementación: Fase en la que se desarrolla la aplicación en código.
- Pruebas: Fase en la que se realizan las pruebas para asegurar que la implementación funcione correctamente. Aquí se verá problemas con la funcionalidad y se verificará el responsive de los dispositivos.
- Mantenimiento: Es la fase que se encarga para corregir errores y hacer mejoras.

8. Arquitectura de la aplicación/sistema/servicio

8.1 Front-end / Back-end

La aplicación está compuesta por una Arquitectura “clásica” cliente-servidor, es decir, un front-end, un back-end y una base de datos:

a) Front-end

Está desarrollado con **Angular** con los lenguajes HTML, CSS y TYPESCRIPT. Con este framework conseguimos una aplicación SPA¹, es decir, todas las pantallas se muestran en la misma página sin recargar el navegador.

Cabe destacar que en el Front-end se usa el patrón MVC², debido a las ventajas que ofrece, como por ejemplo permite desarrollar la aplicación modularmente y de forma escalable. También se utiliza **Angular Material** para la creación de componentes modernos y amigables para el usuario, y se utiliza la librería **Tailwind** para personalizar el css mediante “Clases atómicas³” dichos componentes y convertirlos en responsive.

El cliente, como se muestra en el diagrama, está conformado por Componentes, Modelos, Vistas, Servicios, utilidades; y será el encargado de hacer peticiones http al Back-end mediante el Servicio.

b) Back-end

Se utiliza **Node**, el cual es un entorno que trabaja en tiempo de ejecución, multi-plataforma y código abierto. Además, se trabaja con **Express**, el cual es el framework web más popular de Node.

El back-end, es el encargado de recibir las peticiones que vienen desde el front-end, y dicha comunicación será mediante el API REST⁴. A continuación, actuará Express que manejará la petición por medio de “routes handler”, que son los encargados de manejar las peticiones HTTP. Estas últimas necesitan un controlador que son los encargados de realizar la consulta a la base de datos MySQL y devolver los datos a nuestro en formato JSON⁵ al front-end.

Cabe destacar que se tiene middlewares⁶, esto será necesario para la subida y descarga de logos, planes de estudios, los cuales serán guardados en un repositorio de Amazon AWS-S3, y registrados en la base de datos MySQL. Como también un middleware para la verificación del token, que se explicará más adelante en el punto 17. Seguridad.

-
1. SPA: Son las siglas de Single Page Application (Aplicación de una sola página).
 2. MVC: Son las siglas del patrón Modelo, Vista, Controlador.
 3. Clases atómicas: es una aproximación a escribir clases CSS pequeñas (abreviadas) con nombres basados en su función visual. Es decir, por ejemplo, pr-2 es padding-right: 0.5rem
 4. REST: Interfaz para conectar varios sistemas basados en el protocolo HTTP. Realiza operaciones Get, Post, Update, Delete (CRUD).
 5. JSON: Según (JSON, 2021), JavaScript Object Notation es un formato ligero de intercambio de datos. Está constituido por dos estructuras: una colección de pares de nombre/valor (objeto) y una lista ordenada de valores.
 6. Middleware: Según la página de (ExpressJS, 2021), son funciones que tienen acceso al objeto de solicitud (req), al objeto de respuesta (res) y a la siguiente función de middleware en el ciclo de solicitud/respuestas de la aplicación.

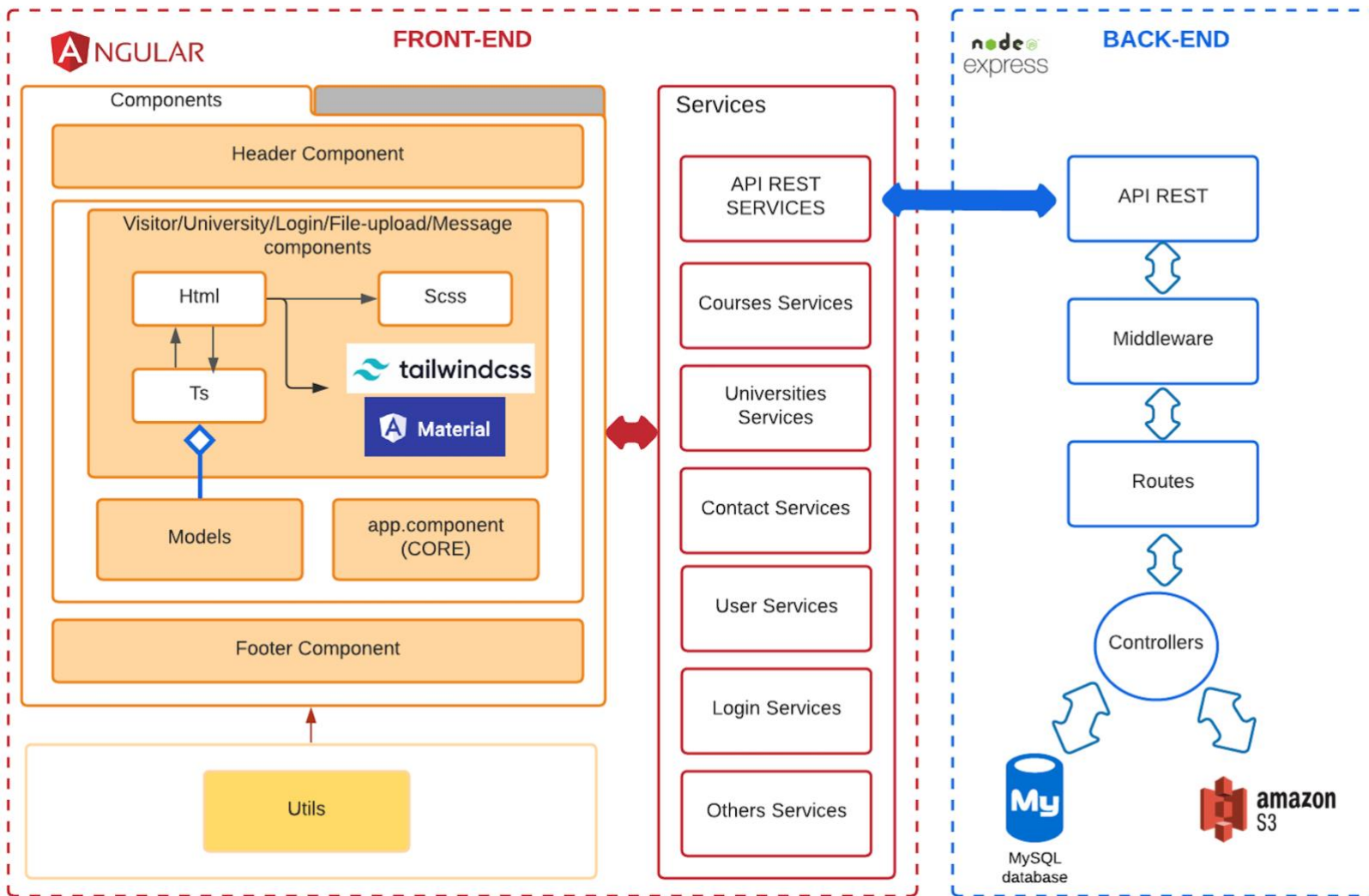


Figura 3. Arquitectura de la aplicación.

8.2 Diagrama de base de datos

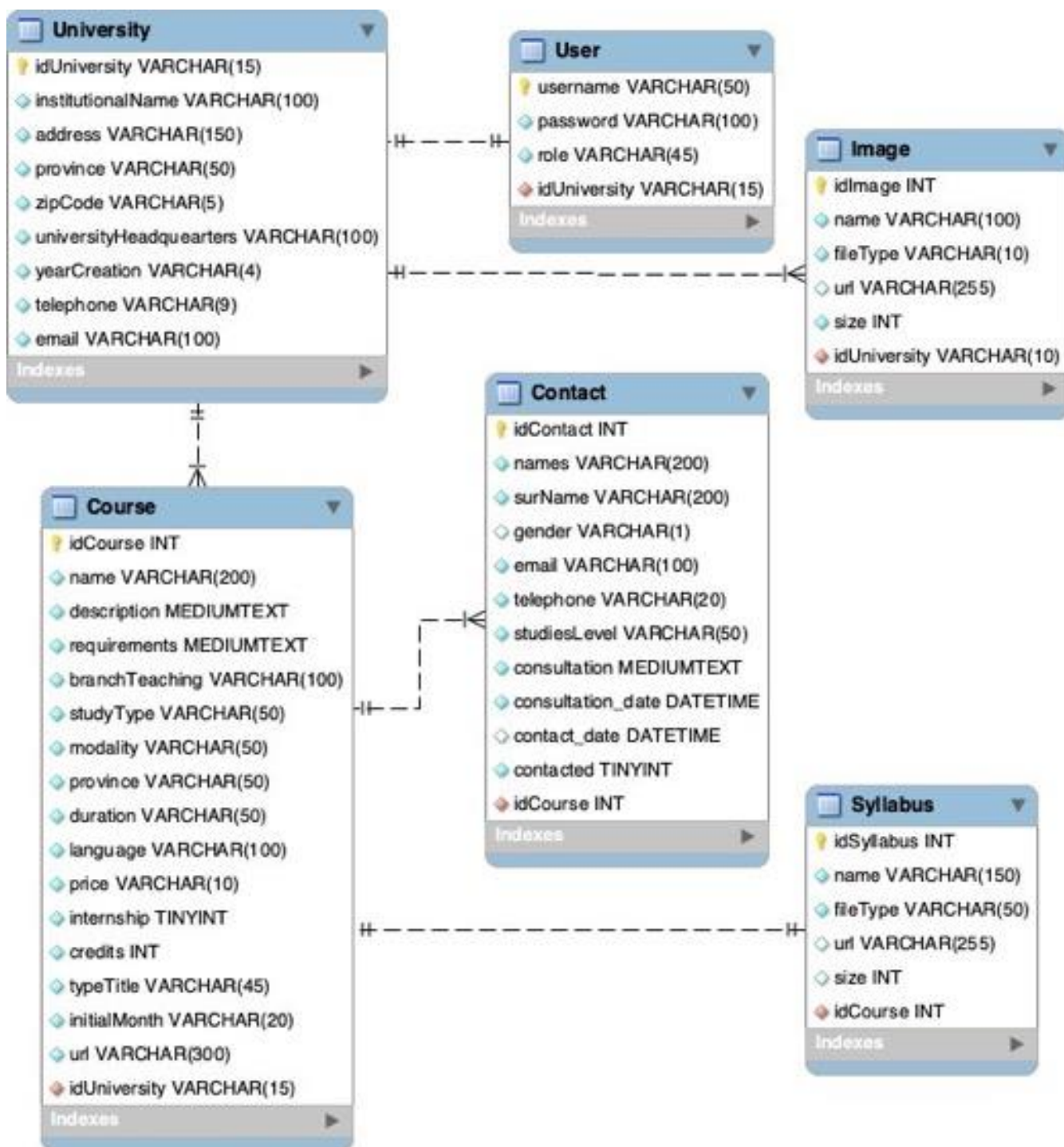


Figura 4- Diagrama de base de datos

9. Plataforma de desarrollo

Para el desarrollo del proyecto se utilizará lo siguiente:

Software	
Planificación	GanttProject
Diagramas UML	StarUML
Diagrama Arquitectura	LucidChart
IDE Desarrollo	Visual Studio Code
Creación de prototipos	JustInMind
Backend	NodeJS + Express
Frontend	Angular
Base de datos	MySQL
Control de versiones	GitHub/ SourceTree
Sistema Operativo	macOS BigSur
Test API REST	Postman
Navegadores	Google Chrome, Mozilla Firefox for Developers

Tabla 4- Software utilizado para el desarrollo

Hardware	
Ordenador	MacPro 13" 2,4 GHz i5 16gb 1536MB Gráfica

Tabla 5- Hardware utilizado para el desarrollo

10. Planificación

Tarea	Duración en días	Fecha de inicio	Fecha de fin	Horas
PEC1- Análisis/Requisitos	14 d	15/09/2021	28/09/2021	56 h
Planteamiento de la idea de Proyecto	4	15/09/2021	18/09/2021	16
Contexto y justificación	2	19/09/2021	20/09/2021	8
Descripción de las funcionalidades	2	21/09/2021	22/09/2021	8
Objetivos del proyecto	2	23/09/2021	24/09/2021	8
Planificación del proyecto	2	25/09/2021	26/09/2021	8
Documentación de la memoria	2	27/09/2021	28/09/2021	8
PEC2- Diseño	29 d	29/09/2021	27/10/2021	116 h
Diagrama de clases	3	29/09/2021	01/10/2021	12
Diagramas de caso de uso	4	02/10/2021	05/10/2021	16
Prototipos	10	06/10/2021	15/10/2021	40
Análisis del mercado	3	16/10/2021	18/10/2021	12
Estudio de la usabilidad	4	19/10/2021	22/10/2021	16
Documentación de la memoria	5	23/10/2021	27/10/2021	20
PEC3- Implementación y Pruebas	39 d	28/10/2021	05/12/2021	156 h
Configuración del entorno de desarrollo	1	28/10/2021	28/10/2021	4
Programación del backend	14	29/10/2021	11/11/2021	56
Programación del frontend	16	12/11/2021	27/12/2021	64
Testeo de la aplicación y corrección de errores	4	28/11/2021	01/12/2021	16
Creación del video de la primera versión de la aplicación	2	02/12/2021	03/12/2021	8
Documentación de la memoria	2	04/12/2021	05/12/2021	8
PEC4- Entrega final	29 d	06/12/2021	03/01/2022	116 h
Finalizar Backend	7	06/12/2021	12/12/2021	28
Finalizar Frontend	7	13/12/2021	19/12/2021	28
Corrección errores	2	20/12/2021	21/12/2021	8
Presentación proyecto para público general	3	22/12/2021	24/12/2021	12
Publicación web	3	25/12/2021	27/12/2021	12
Presentación en video la aplicación	2	28/12/2021	29/12/2021	8
Autoinforme de la valoración	1	30/12/2021	30/12/2021	4
Documentación de la memoria	4	31/01/2022	03/12/2022	16
Total	111 días			444 horas

El día laboral para el proyecto, es de 4 horas, incluyendo sábados y domingos.

Recursos: 1 persona.

Tabla 6 – Planificación

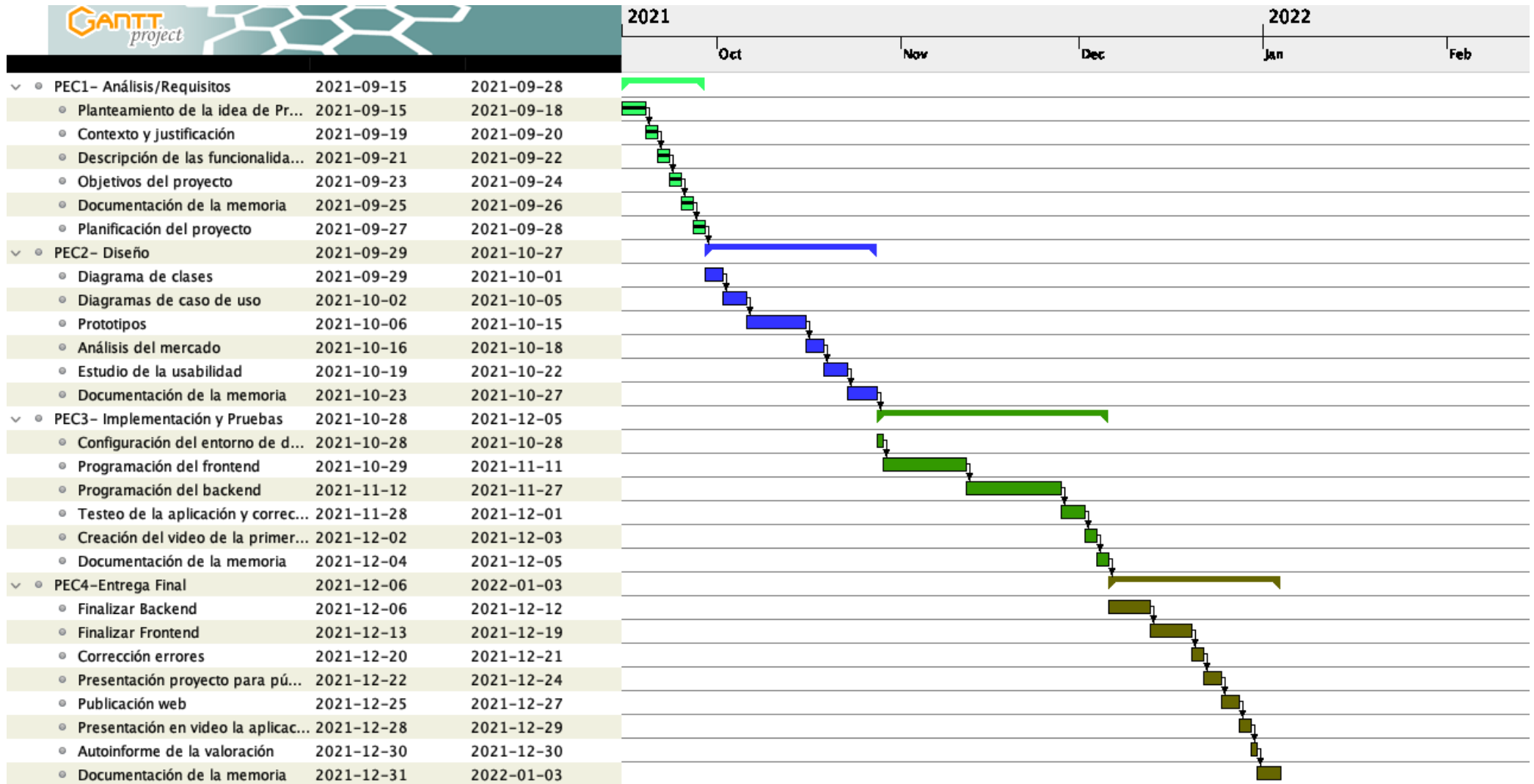


Figura 5- Planificación – Diagrama de Gantt

11. Diagramas UML

11.1 Diagrama de clases

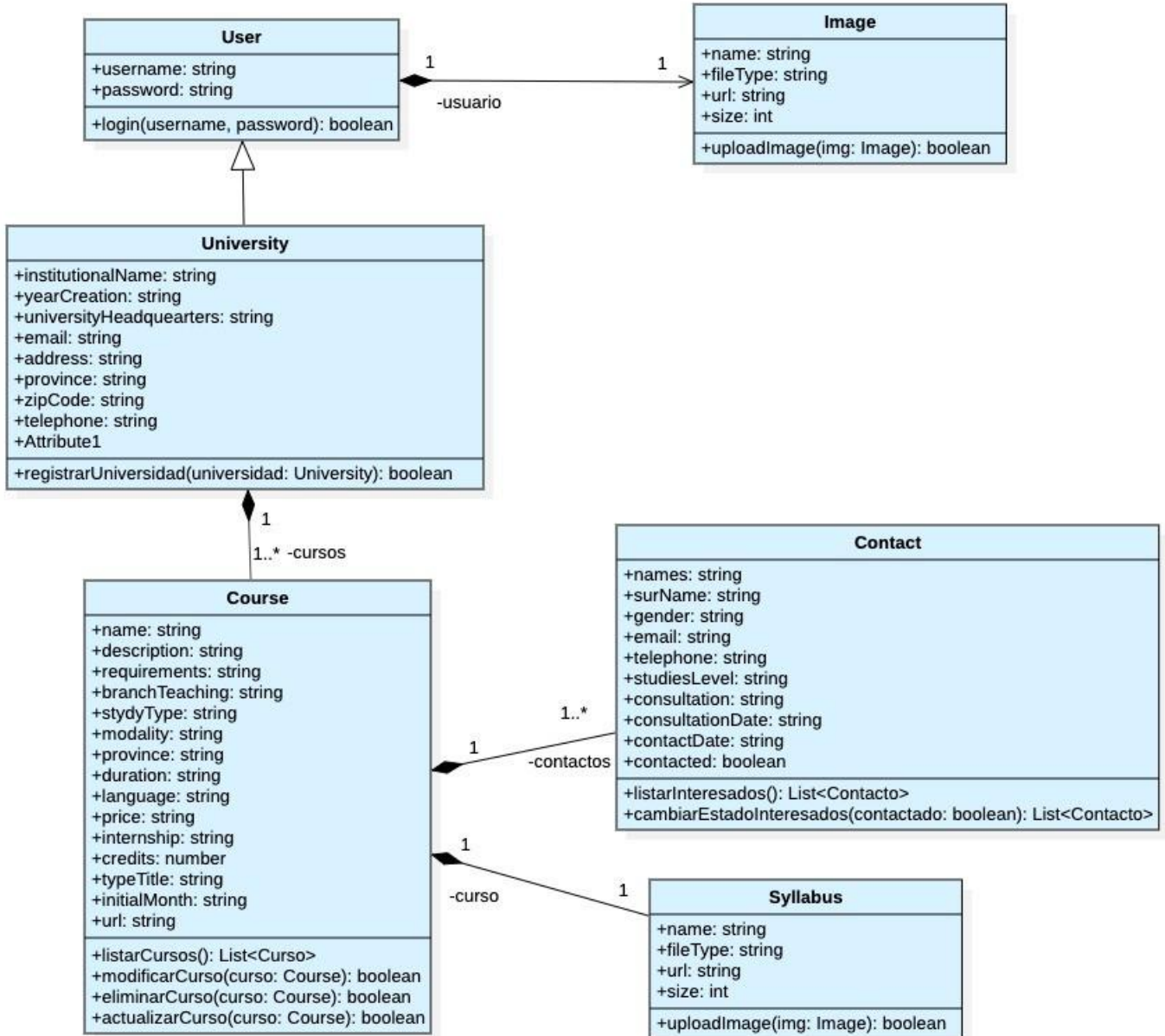


Figura 6- Diagrama de clases

11.3 Diagramas de casos de uso

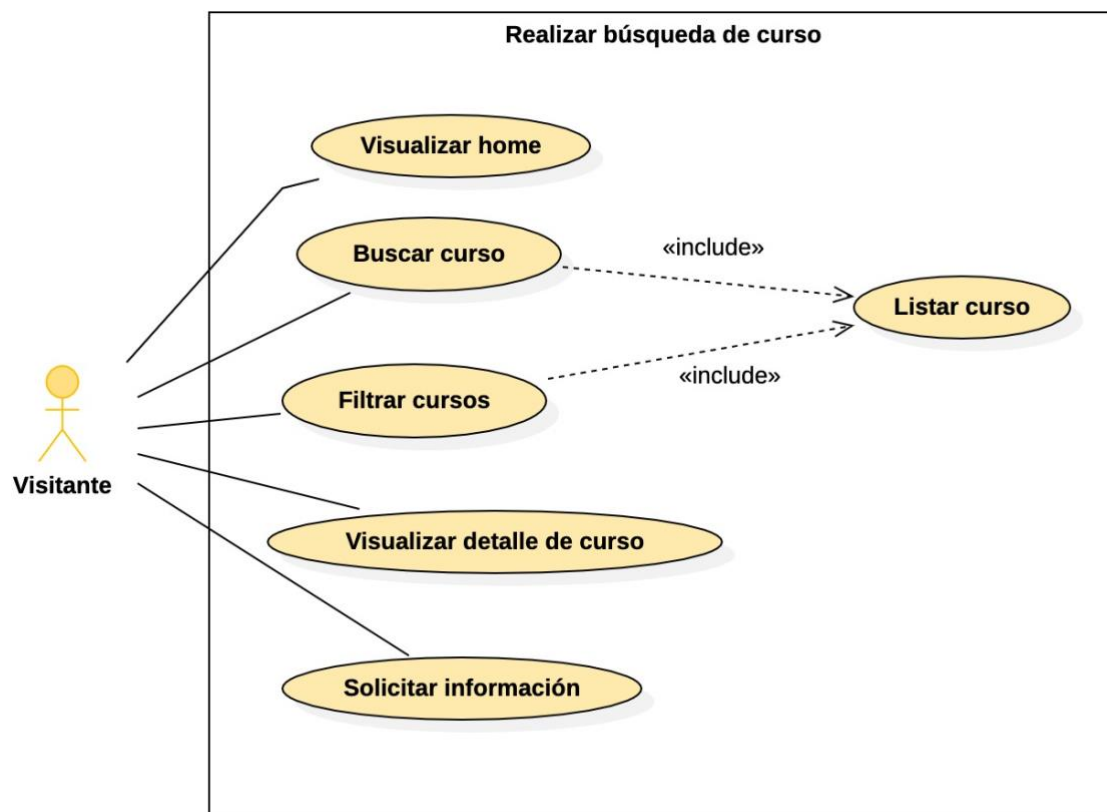


Figura 7- Diagrama de caso de uso: Realizar búsqueda de curso

Caso de uso: Realizar búsqueda de curso	
Actores:	Visitante
Descripción	Permite a cualquier visitante realizar una búsqueda de algún curso de interés.
Precondiciones:	1.Haber escrito la URL en el navegador y no estar logueado.
Flujo	<ol style="list-style-type: none"> 1. El visitante escribe el curso de interés en la caja de búsqueda en home. 2. El sistema sale de home, y muestra todos los cursos que coincidan con la búsqueda. 3. El visitante utiliza los filtros para acotar la búsqueda como son: rama de enseñanza, tipo de estudio, modalidad, provincia, duración, idiomas, precio y dará click en el botón “Filtrar”. 4. El sistema muestra los cursos según los filtros aplicados. 5. El visitante puede hacer click en el título de cada curso para ampliar la información. 6. El sistema mostrará toda la información respecto al curso de interés. Y mostrará detalle del curso como son: descripción, plan de estudios, URL de la universidad, requisitos, prácticas, créditos, tipo de título(oficial/propio), fecha de inicio, entre otros. 7. Sí mirando toda la información, el visitante tiene alguna duda, podrá dar click en “solicitar información”. 8. El sistema mostrará un pop-up con un formulario para completar datos del interesado como: nombre, apellidos, email, sexo, teléfono, nivel de estudios y consulta. 9. El visitante completa los datos y hace click en enviar. 10. Posteriormente, el sistema envía la información a la universidad. Y mostrará un mensaje indicando que se ha enviado correctamente.
Flujo alternativo:	<p>El visitante decide “solicitar información” directamente.</p> <ol style="list-style-type: none"> 2.1. Una vez que el sistema muestre los cursos (estén filtrados o no), el visitante puede dar click a “solicitar información” y seguirá a continuación los pasos del 8 al 10.

Postcondiciones	Enviar información a la universidad con los datos completados del formulario "Solicitar información".
------------------------	---

Tabla 7- Caso de uso: Realizar búsqueda de curso.

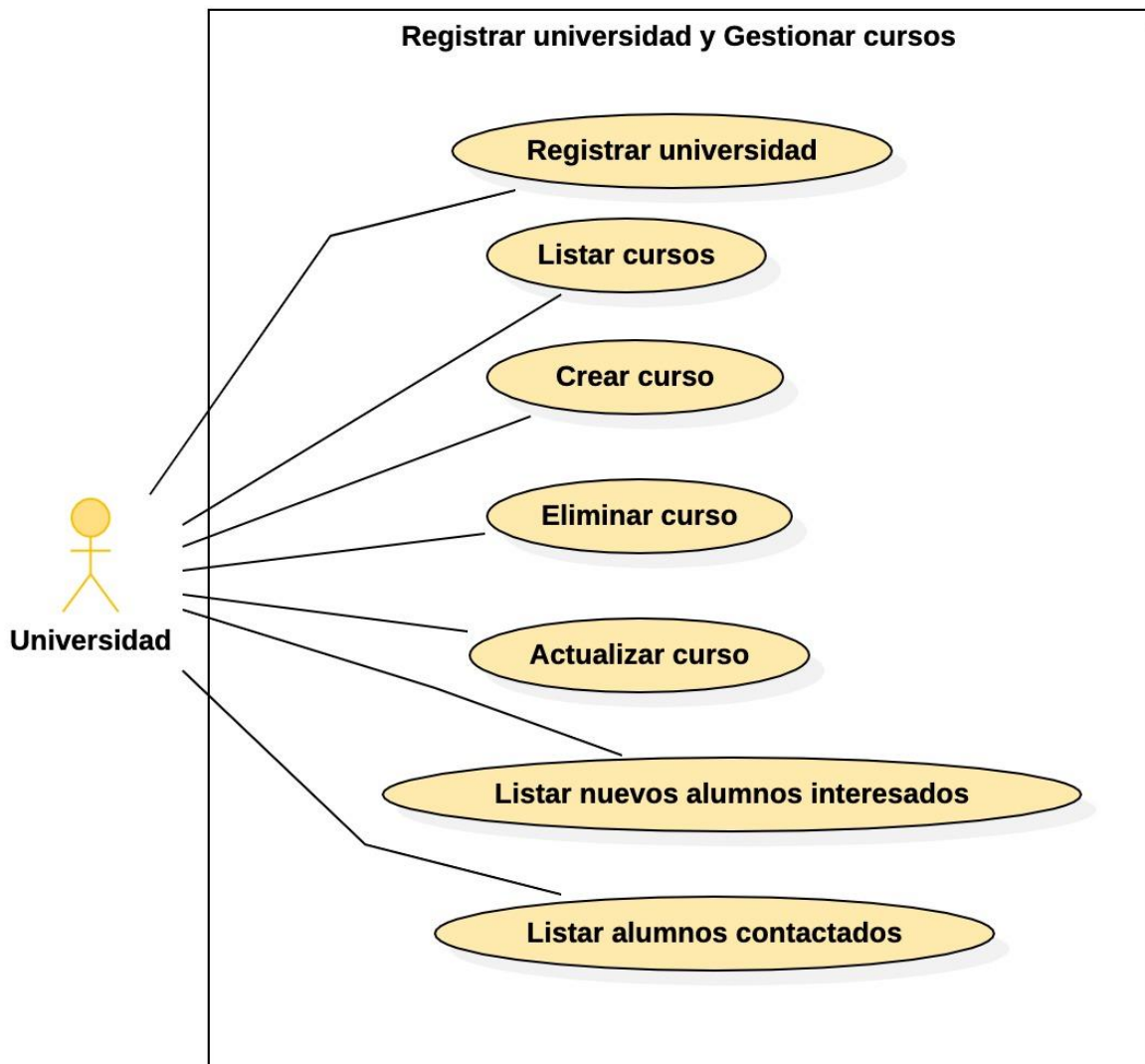


Figura 8- Diagrama de caso de uso: Registrar universidad y Gestionar cursos

Caso de uso: Registrar universidad y Gestionar cursos	
Actores:	Universidad
Descripción	Permite a la universidad registrarse por primera vez, realizar toda la gestión de cursos y listar "Alumnos interesados". Es decir, una vez ya registrado, en el perfil de universidad se mostrará una única pantalla con tres bloques: "Mis cursos" , "Nuevo curso" y "Nuevos Alumnos interesados" .
Precondiciones:	Para gestionar cursos: 1. La universidad debe estar registrada en la aplicación. 2. La universidad debe estar previamente logueado correctamente.
Flujo	1. La universidad selecciona "Regístrate" en la barra de navegación del menú principal. 2. El sistema muestra un formulario para completar con los siguientes campos: nombre, año de creación, dirección, provincia, código postal, email, teléfono, username y password. Adicionalmente, se pedirá que se suba un logo de la universidad. 3. La universidad completa todos los campos necesarios para crear su cuenta o perfil. 4. El sistema muestra un mensaje que su cuenta ha sido creada exitosamente. 5. La universidad puede loguearse, para ello tiene que hacer click en el botón de Acceso que se muestra en la cabecera. 6. El sistema dejará ingresar al perfil de universidad si las credenciales han sido correctas, en caso contrario mostrará una validación que los datos ingresados son incorrectos. 7. El usuario visualizará una pantalla con tres bloques: "Mis cursos" , "Nuevo curso" y "Nuevos Alumnos interesados" . 8. Si la universidad selecciona "Mis cursos" . El sistema muestra un listado separado por bloques de rama de enseñanza como son: "Ingeniería y Arquitectura",

	<p>“Ciencias Sociales y Jurídicas”. Y en cada lista muestra solo los campos más importantes.</p> <p>9. La universidad puede editar y eliminar algún curso del bloque de alguna rama enseñanza.</p> <p>10. Si el usuario elige eliminar, el sistema mostrará un mensaje que se ha eliminado correctamente o no. Si elige editar mostrará una vista con los datos precargados para realizar la edición, y posteriormente también mostrará un mensaje si ha ido todo correctamente.</p> <p>11. La universidad también puede seleccionar “Nuevo curso”.</p> <p>12. El sistema muestra un formulario para registrar dicho curso con los siguientes campos: Nombre, descripción, requisitos, subir plan de estudios, rama de enseñanza, tipo de estudio (master o postgrado), modalidad, provincia, duración, idioma, precio, prácticas, créditos, tipo de título (oficial/ propio), mes de inicio, URL.</p> <p>13. La universidad también puede seleccionar “Nuevos alumnos interesados”.</p> <p>14. El sistema muestra dos paneles. El primero llamado “Nuevos alumnos interesados” y el segundo “Alumnos Contactados”, en ambos se muestra información de los alumnos que completaron el formulario de “solicitar información” en el perfil visitante.</p> <p>15. En el primer panel, la universidad verá una lista con todos los alumnos interesados, donde para ver la consulta y fecha de la misma, tendrá que seleccionar “Ver”. Además, existe el campo Contactado, donde la universidad podrá actualizarlo cuando haya contactado con el futuro estudiante vía correo o telefónicamente.</p> <p>16. En caso que la universidad haya contactado con un alumno y seleccione Sí(contactado). El sistema actualizará los paneles. En el primer panel desaparecerá el registro, debido a que ya se contactó. En el segundo panel aparecerá el nuevo registro, debido a que ya se ha</p>
--	---

	<p>contactado y agregará el campo fecha de contacto, el cual se podrá observar si el usuario selecciona Ver.</p>
<p>Postcondiciones</p>	<p>Depende de la opción que haya seleccionado la universidad:</p> <ol style="list-style-type: none"> 1. Regístrate: Un nuevo registro de universidad con sus correspondientes campos es creado en la base de datos. 2. Mis cursos: Los cursos se verán actualizados o eliminados del sistema. 3. Nuevo curso: Los nuevos cursos creados quedarán registrados en la base de datos del sistema. 4. Nuevos Alumnos interesados: Los cambios de los alumnos interesados quedarán actualizados en la base de datos del sistema.

Tabla 8- Caso de uso: Registrar universidad y Gestionar cursos.

12.Prototipos

En esta sección incluiremos prototipos de Alta calidad diseñados con el software Justinmind. Se tiene en cuenta dispositivos:

- Desktop: A partir de 1024px.
- Tablet: A partir de 768px, tomando como referencia iPad 9.7”
- Mobile: A partir 374px, tomando como referencia iPhone X.

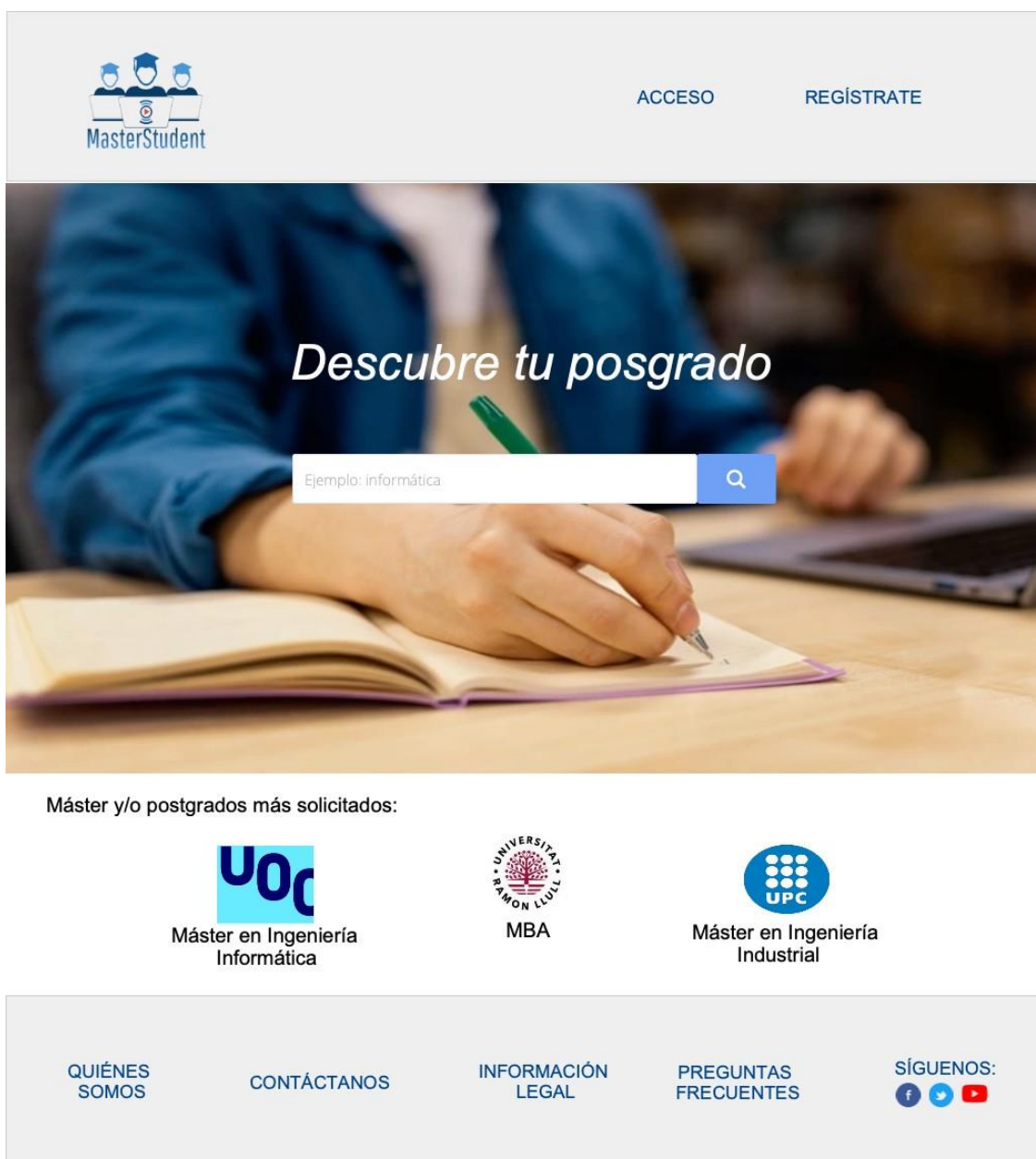


Figura 9 - Pantalla de inicio. Versión escritorio.

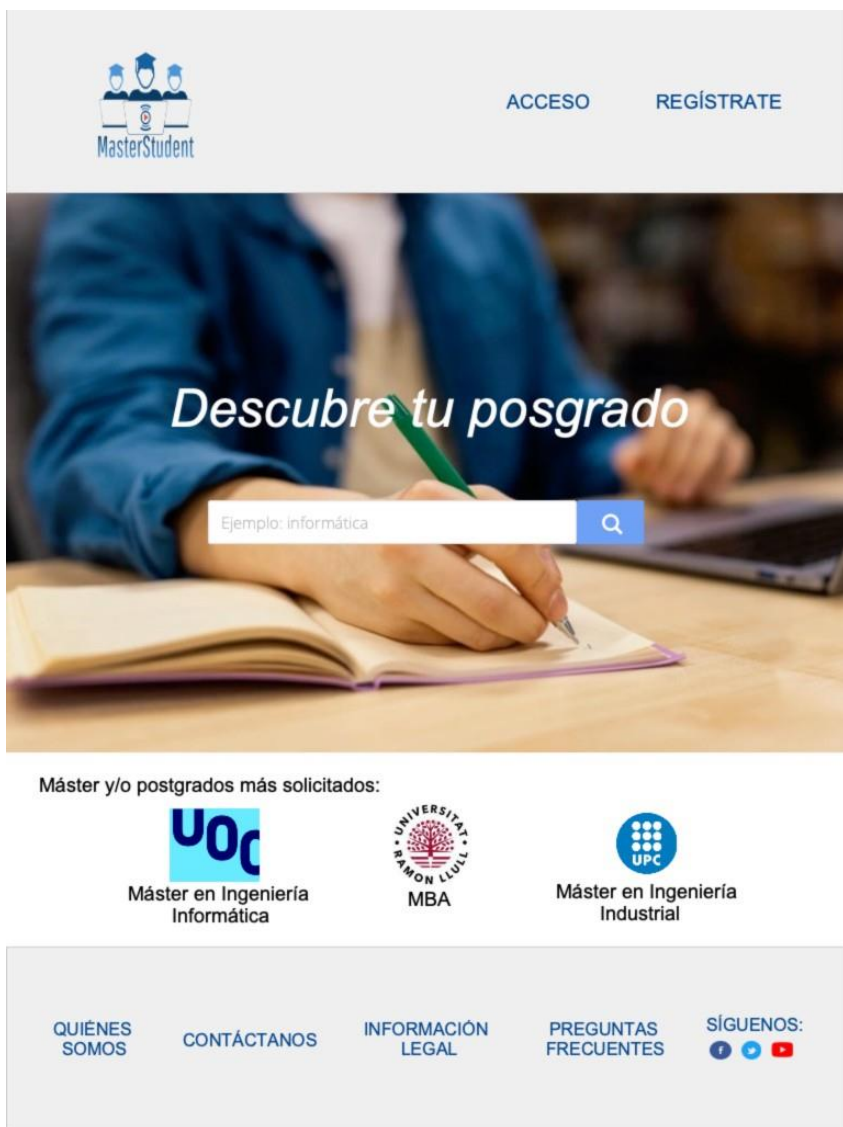



Figura 10 - Pantalla de inicio. Versión tablet.



Figura 11 - Pantalla de inicio. Versión móvil.



[ACCESO](#)
[REGÍSTRATE](#)

Home > Búsqueda de cursos

LIMPIAR
APLICAR

Rama de enseñanza

- Ingeniería y Arquitectura
- Ciencias Sociales y Jurídicas

Tipo de estudio

- Master
- Postgrado

Modalidad

- Presencial
- SemiPresencial
- Online

Provincia

- Barcelona

Duración

- Menor a un año
- Un año
- Mayor a un año


Idioma


- Castellano
- Catalán
- Inglés


Precio


- 0 - 1000€
- 1001 - 3000€
- 3001 - 5000€
- 5001 - 10000€
- + 10000€

Máster en informática
Universidad Politécnica de la Catalunya



 Official



 Presencial



 3800,00 €



 90 ECTS

Máster en informática
Universidad Oberta de Catalunya



 Official



 Presencial



 3600,00 €



 72 ECTS

Máster en informática
Universidad Politécnica de la Catalunya



 Official



 Presencial



 3400,00 €



 90 ECTS

Máster en informática
Universidad Oberta de Catalunya


 Official


 Presencial


 3100,00 €


 72 ECTS

<<
1
2
3
>>

[QUIÉNES SOMOS](#)

[CONTÁCTANOS](#)

[INFORMACIÓN LEGAL](#)

[PREGUNTAS FRECUENTES](#)

SÍGUENOS:








Figura 12 – Búsqueda y filtrado de curso. Versión escritorio.

MasterStudent

Ejemplo: Máster en informática ACCESO REGÍSTRATE

Home > Búsqueda de cursos

LIMPIAR APLICAR

Rama de enseñanza
Ingeniería y Arquitectura
Ciencias Sociales y Jurídicas

Tipo de estudio
Master
Postgrado

Provincia
Presencial
SemiPresencial
Online

Ciudad
Barcelona

Duración
Menor a un año
Un año
Mayor a un año

Idioma
Castellano
Catalán
Inglés

Precio
0 - 1000€
1001 - 3000€
3001 - 5000€
5001 - 10000€
+ 10000€

Máster en informática
Universidad Politécnica de la Catalunya
Oficial Presencial 3800,00 € 90 ECTS
SOLICITAR INFORMACIÓN

Máster en informática
Universidad Oberta de Catalunya
Oficial Presencial 3600,00 € 72 ECTS
SOLICITAR INFORMACIÓN

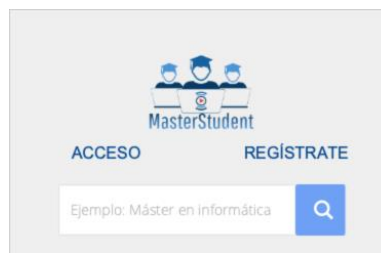
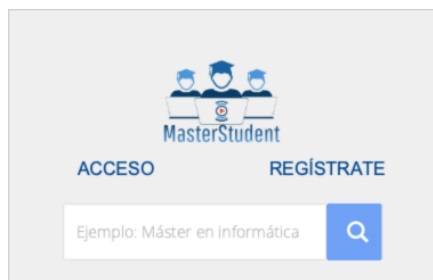
Máster en informática
Universidad Politécnica de la Catalunya
Oficial Presencial 3400,00 € 90 ECTS
SOLICITAR INFORMACIÓN

Máster en informática
Universidad Oberta de Catalunya
Oficial Presencial 3100,00 € 72 ECTS
SOLICITAR INFORMACIÓN

<< 1 2 3 >>

QUIÉNES SOMOS CONTÁCTANOS INFORMACIÓN LEGAL PREGUNTAS FRECUENTES SÍGUENOS:

Figura 13 – Búsqueda y filtrado de curso. Versión tablet.



Home > Búsqueda de cursos



<< 1 2 3 >>

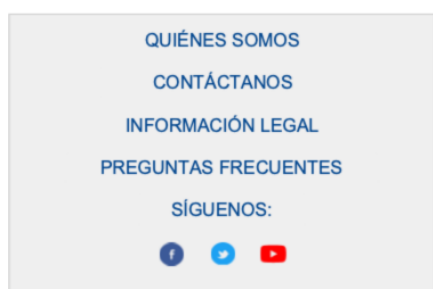


Figura 14 – Búsqueda y filtrado de curso. Filtro activado y no activado. Versión móvil.

The screenshot displays the 'MasterStudent' website interface. At the top, there is a search bar with the text 'Ejemplo: Máster en informática' and a search icon. To the right of the search bar are links for 'ACCESO' and 'REGÍSTRATE'. Below the search bar, a breadcrumb trail reads 'Home > Búsqueda de cursos > Máster en informática'. The main heading is 'Máster en informática' in blue, followed by 'Universidad Politécnica de la Catalunya'. Below this, there are three expandable sections: 'DESCRIPCIÓN', 'PLAN DE ESTUDIOS', and 'REQUISITOS'. The 'DATOS GENERALES' section is expanded, showing a two-column table of course details. At the bottom of the details section is a red button labeled 'SOLICITAR INFORMACIÓN'. The footer contains navigation links: 'QUIÉNES SOMOS', 'CONTÁCTANOS', 'INFORMACIÓN LEGAL', 'PREGUNTAS FRECUENTES', and 'SÍGUENOS:' with social media icons for Facebook, Twitter, and YouTube.

DATOS GENERALES	
RAMA DE ENSEÑANZA Ingeniería y Arquitectura	TIPO DE ESTUDIO Máster
MODALIDAD Presencial	PROVINCIA Barcelona
DURACIÓN 1,5 años	IDIOMA(S) Castellano, Catalán
PRECIO 3800,00 €	PRÁCTICAS Sí
CRÉDITOS 90 ECTS	TIPO DE TÍTULO Oficial
MES DE INICIO Septiembre	URL https://masters.fib.upc.edu/es/masters/master-en-ingenieria-informatica

Figura 15 – Detalle del curso. Versión escritorio.

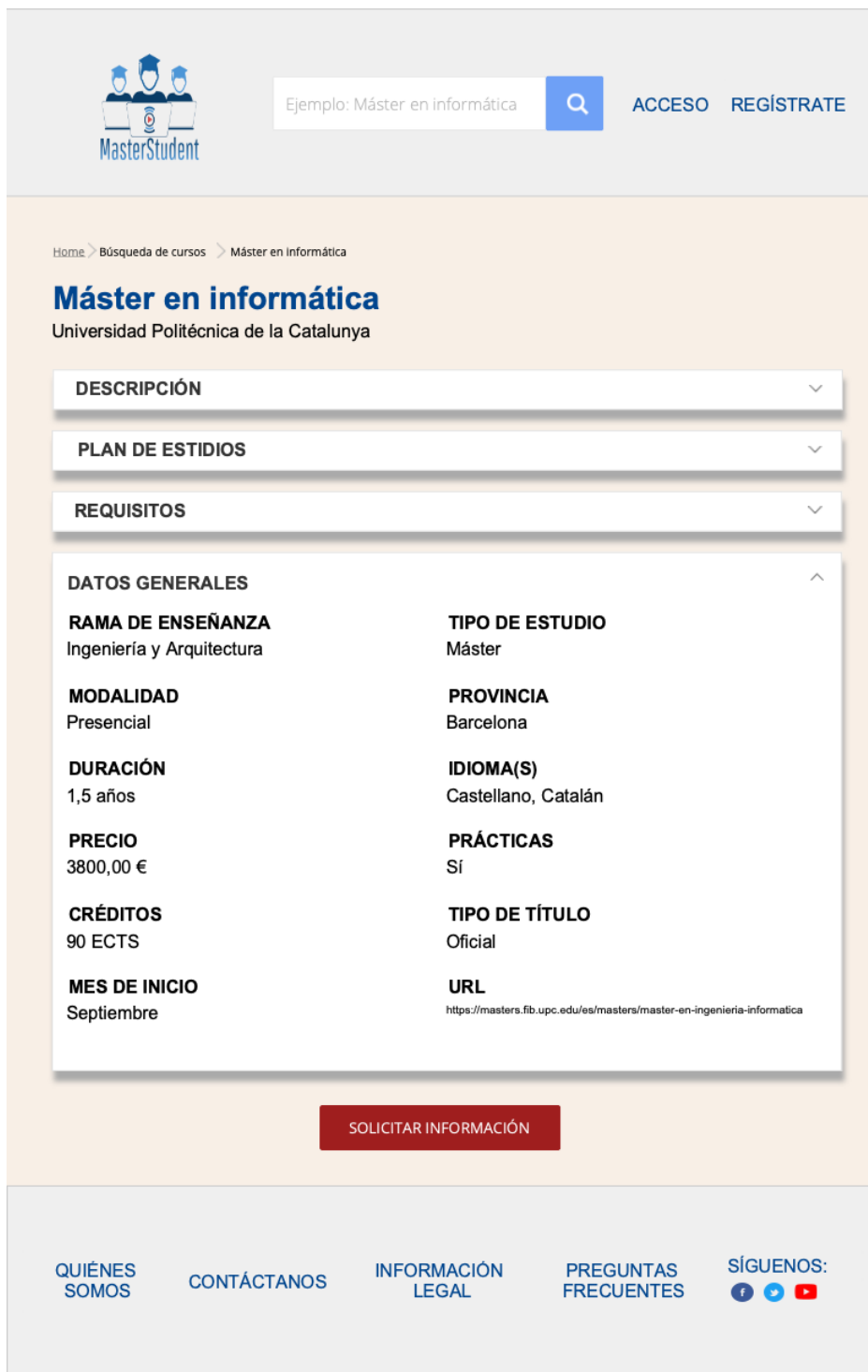


Figura 16 – Detalle del curso. Versión tablet.



Figura 17 – Detalle del curso. Versión móvil.

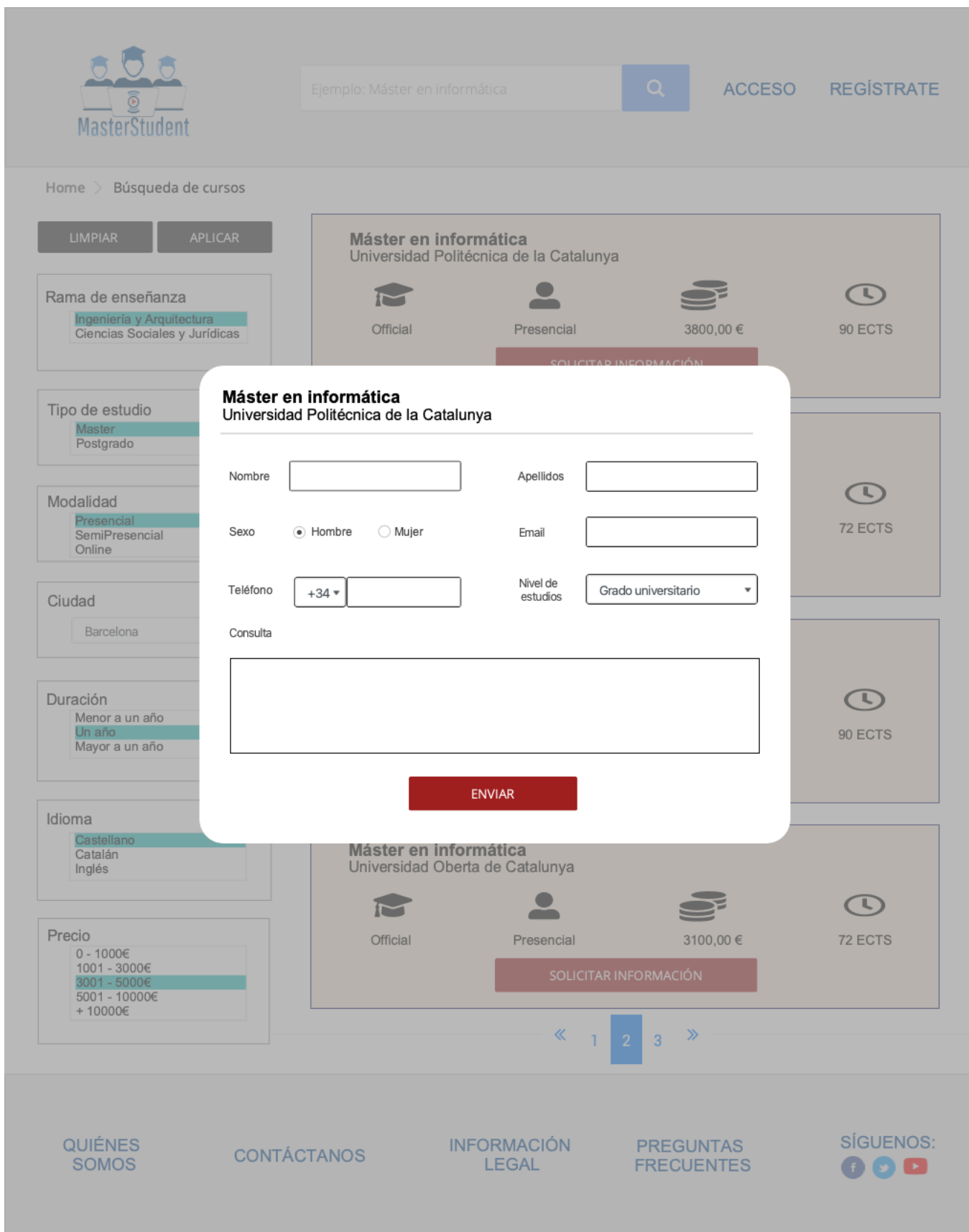


Figura 18 – Solicitar información del curso. Versión escritorio.

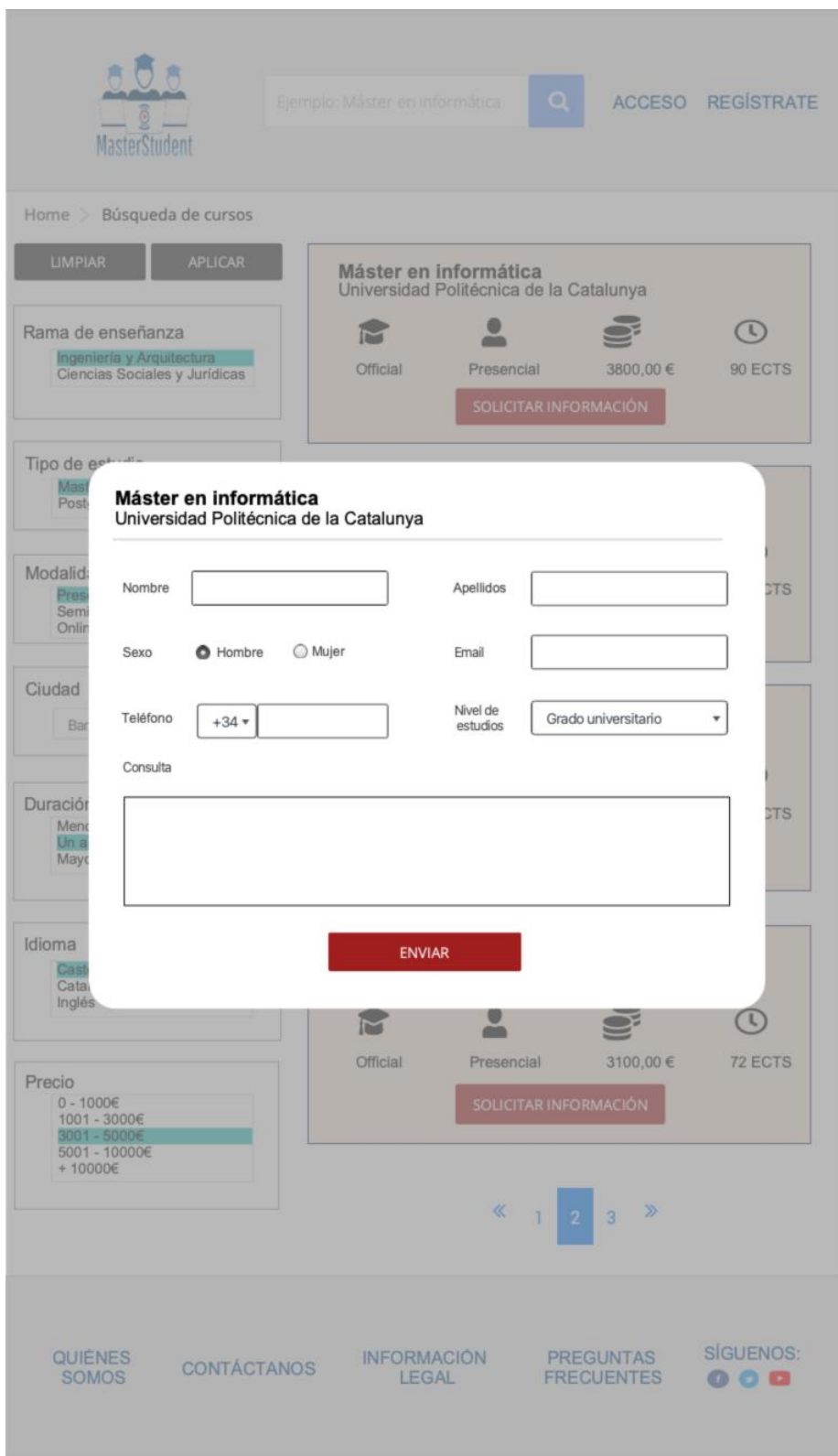


Figura 19 – Solicitar información del curso. Versión tablet.

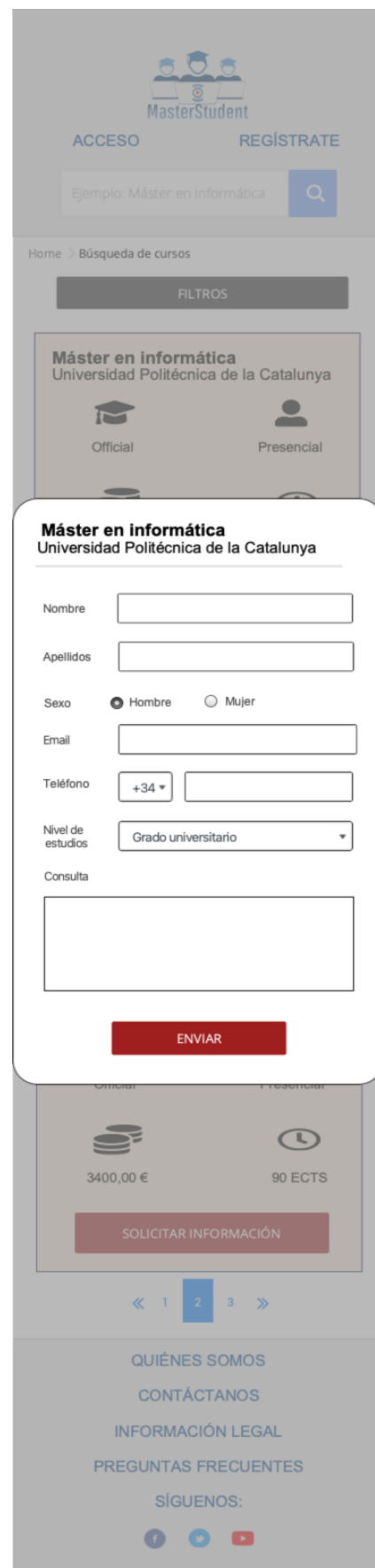



Figura 20 – Solicitar información del curso. Versión móvil.



Home > Registrar universidad

Registrar universidad

Nombre

Dirección


Provincia Código Postal

Sedes Año de creación

Teléfono Email

Username password

Logo de universidad

 Arrastrar y soltar o clickea aquí

REGISTRAR




QUIÉNES SOMOS CONTACTANOS INFORMACIÓN LEGAL PREGUNTAS FRECUENTES SÍGUENOS:   

Figura 21 – Registrar Universidad. Versión escritorio.

The tablet version of the 'Registrar universidad' form features a clean, wide layout. At the top left is the 'MasterStudent' logo. Below it, a breadcrumb trail reads 'Home > Registrar universidad'. The main heading 'Registrar universidad' is in a bold, blue font. The form fields are arranged in a grid: 'Nombre' (text input), 'Dirección' (text input), 'Provincia' (dropdown menu with 'Barcelona' selected), 'Código Postal' (text input), 'Sedes' (text input), 'Año de creación' (text input), 'Teléfono' (text input), 'Email' (text input), 'Username' (text input), and 'password' (text input). Below these is a 'Logo de universidad' section with a large rectangular area and the text 'Arrastrar y soltar o clickea aquí'. A red 'REGISTRAR' button is centered at the bottom of the form area. The footer contains navigation links: 'QUIÉNES SOMOS', 'CONTÁCTANOS', 'INFORMACIÓN LEGAL', 'PREGUNTAS FRECUENTES', and 'SÍGUENOS:' with social media icons for Facebook, Twitter, and YouTube.

Figura 22 – Registrar Universidad. Versión tablet.

The mobile version of the 'Registrar universidad' form is designed for vertical viewing. It starts with the 'MasterStudent' logo at the top. The breadcrumb trail 'Home > Registrar universidad' is followed by the heading 'Registrar universidad'. The form fields are stacked vertically: 'Nombre', 'Dirección', 'Provincia' (dropdown with 'Barcelona'), 'Código Postal', 'Sedes', 'Año de creación', 'Teléfono', 'Email', 'Username', and 'Password'. The 'Logo de universidad' section is at the bottom of the form area, with the text 'Arrastrar y soltar o clickea aquí'. A red 'REGISTRAR' button is positioned below the form. The footer contains the same navigation links as the tablet version: 'QUIÉNES SOMOS', 'CONTÁCTANOS', 'INFORMACIÓN LEGAL', 'PREGUNTAS FRECUENTES', and 'SÍGUENOS:' with social media icons.

Figura 23 – Registrar Universidad. Versión móvil.



Inicio de sesión

[¿Olvidaste tu contraseña?](#) [Crear cuenta](#)

QUIÉNES
SOMOS

CONTÁCTANOS

INFORMACIÓN
LEGAL

PREGUNTAS
FRECUENTES

SÍGUENOS:
  

Figura 24 – Inicio de sesión. Versión escritorio.



Figura 25 – Inicio de sesión. Versión tablet.

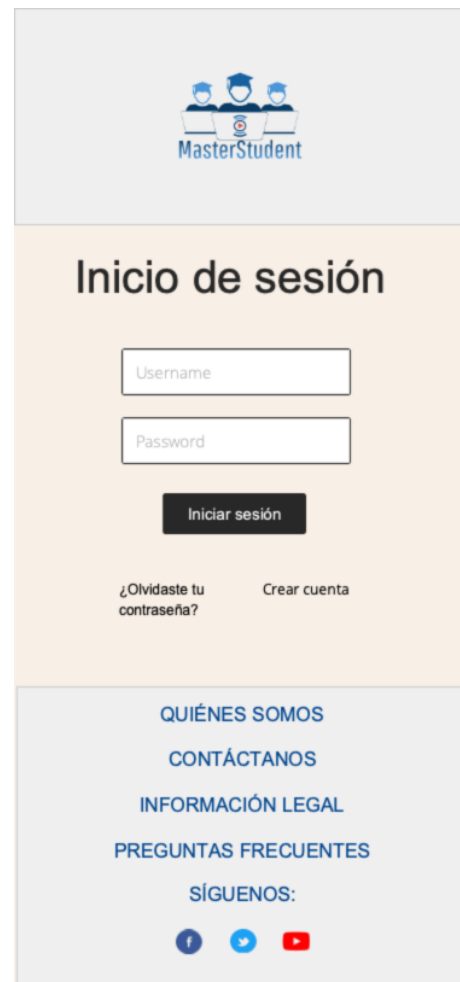


Figura 26 – Inicio de sesión. Versión móvil.

The dashboard features a top navigation bar with the MasterStudent logo on the left and 'MI CUENTA' and 'SALIR' links on the right. The main content area is a light beige rectangle containing the text 'BIENVENIDO(A)' in bold, the UPC logo (a blue circle with a grid of white dots and the letters 'UPC' below it), and the text 'UNIVERSIDAD POLITÉCNICA DE CATALUNYA' in bold. Below this are three blue rounded rectangular buttons labeled 'MIS CURSOS', 'NUEVO CURSO', and 'ALUMNOS INTERESADOS'. The bottom section is a light grey footer with five links: 'QUIÉNES SOMOS', 'CONTÁCTANOS', 'INFORMACIÓN LEGAL', 'PREGUNTAS FRECUENTES', and 'SÍGUENOS:'. The 'SÍGUENOS:' link is accompanied by small icons for Facebook, Twitter, and YouTube.

Figura 27 – Dashboard. Versión escritorio.



Figura 28 – Dashboard. Versión tablet.



Figura 29 – Dashboard. Versión escritorio.

Dashboard > Mis cursos

Mis cursos

INGENIERÍA Y ARQUITECTURA				
Nombre del curso	Modalidad	Créditos	Tipo de título	Acciones
Máster en Ingeniería Informática	Presencial	60 ECTS	Oficial	Ver Editar Eliminar
Máster en Ingeniería Industrial	Presencial	60 ECTS	Oficial	Ver Editar Eliminar

CIENCIAS SOCIALES Y JURÍDICAS				
Nombre del curso	Modalidad	Créditos	Tipo de título	Acciones
Máster en Administración y Dirección de Empresas	Presencial	60 ECTS	Oficial	Ver Editar Eliminar




QUIÉNES SOMOS CONTÁCTANOS INFORMACIÓN LEGAL PREGUNTAS FRECUENTES SÍGUENOS:   

Figura 30 – Mis cursos. Versión escritorio.

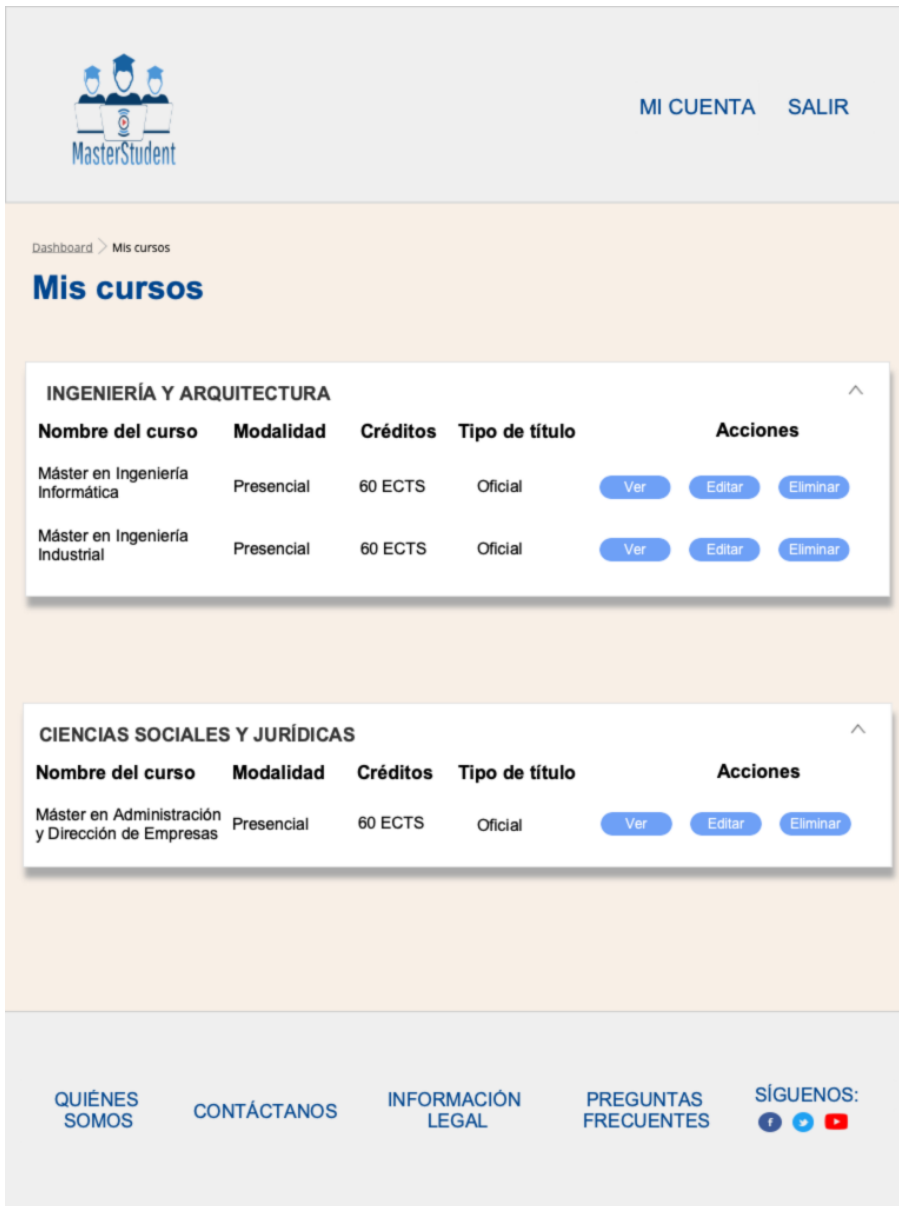


Figura 31 – Mis cursos. Versión tablet.

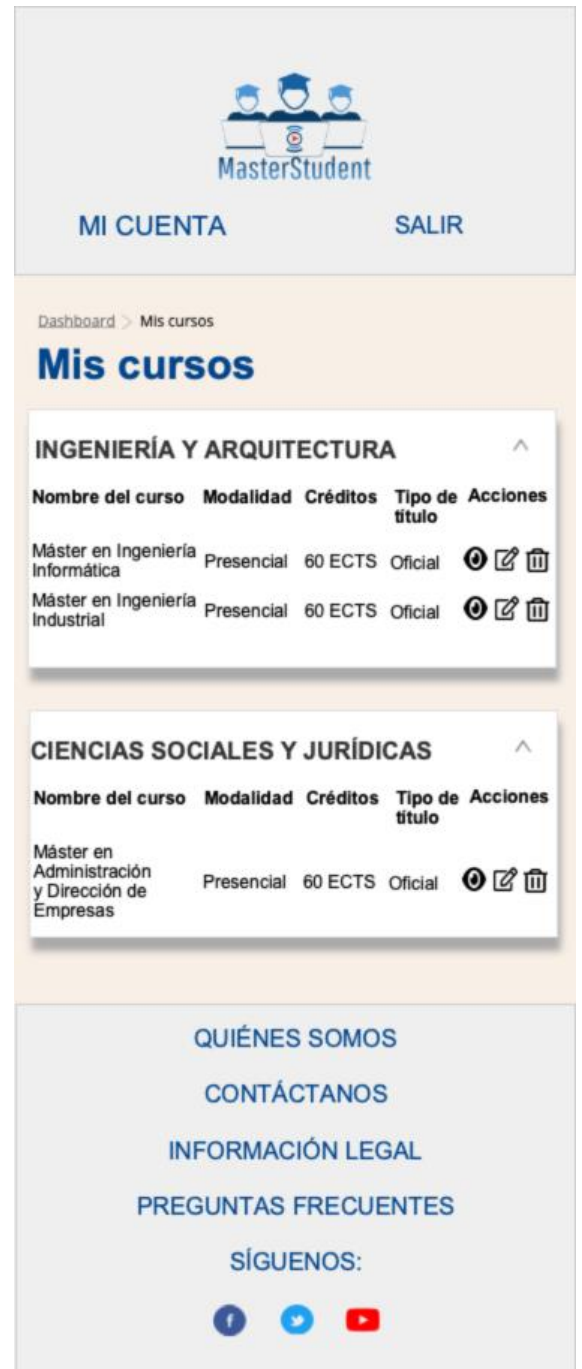



Figura 32 – Mis cursos. Versión móvil.

MI CUENTA SALIR

Dashboard > Mis cursos


Nuevo Curso

Universidad Politécnica de la Catalunya

Nombre

Descripción

Plan de Estudios

 Arrastrar y soltar o clickea aquí

Requisitos

Rama de enseñanza	<input data-bbox="544 1122 740 1158" type="text" value="Ingeniería y Arquitectura"/>	Tipo de estudio	<input data-bbox="1038 1122 1233 1158" type="text" value="Máster"/>
Modalidad	<input data-bbox="544 1182 740 1218" type="text" value="Presencial"/>	Provincia	<input data-bbox="1038 1182 1233 1218" type="text" value="Barcelona"/>
Duración	<input data-bbox="544 1243 632 1279" type="text"/> años <input data-bbox="651 1243 740 1279" type="text"/>	Idioma(s)	<input data-bbox="1038 1243 1233 1279" type="text"/>
Precio	<input data-bbox="544 1303 676 1339" type="text"/> Euros	Prácticas	<input data-bbox="1038 1303 1233 1339" type="text" value="Sí"/>
Créditos	<input data-bbox="544 1364 676 1400" type="text"/> ECTS	Tipo de título	<input data-bbox="1038 1364 1233 1400" type="text" value="Oficial"/>
Mes de inicio	<input data-bbox="544 1424 740 1460" type="text"/>	Url	<input data-bbox="1038 1424 1233 1460" type="text"/>

GUARDAR

QUIÉNES SOMOS

CONTÁCTANOS

INFORMACIÓN LEGAL

PREGUNTAS FRECUENTES

SÍGUENOS:




Figura 33 – Nuevo y Editar curso. Versión tablet.

MI CUENTA SALIR

Dashboard > Mis cursos

Nuevo Curso

Universidad Politécnica de la Catalunya

Nombre

Descripción

Plan de Estudios

Requisitos

Rama de enseñanza

Tipo de estudio

Modalidad

Provincia

Duración años

Idioma(s)

Precio Euros

Prácticas

Créditos ECTS

Tipo de título

Mes de inicio

Url

GUARDAR

QUIÉNES SOMOS CONTÁCTANOS INFORMACIÓN LEGAL PREGUNTAS FRECUENTES SÍGUENOS:

Figura 34 – Nuevo y Editar curso. Versión tablet.

MI CUENTA SALIR

Dashboard > Mis cursos

Nuevo Curso

Universidad Politécnica de la Catalunya

Nombre

Descripción

Plan de Estudios

Requisitos

Rama de enseñanza

Tipo de estudio

Modalidad

Provincia

Duración años

Idioma(s)

Precio Euros

Prácticas

Créditos ECTS

Tipo de título


Mes de inicio

Url

GUARDAR

QUIÉNES SOMOS CONTÁCTANOS INFORMACIÓN LEGAL PREGUNTAS FRECUENTES SÍGUENOS:

Figura 35 – Nuevo y editar curso. Versión móvil.

MI CUENTA SALIR

Dashboard > ver curso

Ver Curso

Universidad Politécnica de la Catalunya

Nombre

Descripción

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Sed vulputate odio ut enim blandit volutpat maecenas volutpat blandit. Urna molestie at elementum eu facilisis sed odio. Etiam erat velit scelerisque in dictum non consectetur a. Dictum non consectetur a erat nam at lectus urna. Sit amet mauris commodo quis imperdiet massa tincidunt nunc. Id cursus metus aliquam eleifend mi. Sed vulputate mi sit amet mauris commodo quis imperdiet. Senectus et netus et malesuada fames ac turpis. Molestie at elementum eu facilisis sed odio morbi quis commodo

Plan de Estudios

[Descargar](#)

Requisitos

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Morbi tempus iaculis urna id volutpat lacus. Id cursus metus aliquam eleifend mi. Odio euismod lacinia at quis risus sed. Commodo ullamcorper a lacus vestibulum sed. Mauris in aliquam sem fringilla ut morbi. Etiam erat velit scelerisque in dictum non consectetur a.

Rama de enseñanza	<input type="text" value="Ingeniería y Arquitectura"/>	Tipo de estudio	<input type="text" value="Máster"/>
Modalidad	<input type="text" value="Presencial"/>	Provincia	<input type="text" value="Barcelona"/>
Duración	<input type="text" value="1,5"/> años	Idioma(s)	<input type="text" value="Castellano, Catalán"/>
Precio	<input type="text" value="3800,00"/> Euros	Prácticas	<input type="text" value="Sí"/>
Créditos	<input type="text" value="90"/> ECTS	Tipo de título	<input type="text" value="Oficial"/>
Mes de inicio	<input type="text" value="Septiembre"/>	Url	<input type="text" value="https://masters.fib.upc.edu/es/r"/>

QUIÉNES SOMOS CONTÁCTANOS INFORMACIÓN LEGAL PREGUNTAS FRECUENTES SÍGUENOS:




Figura 36 – Ver curso. Versión escritorio.

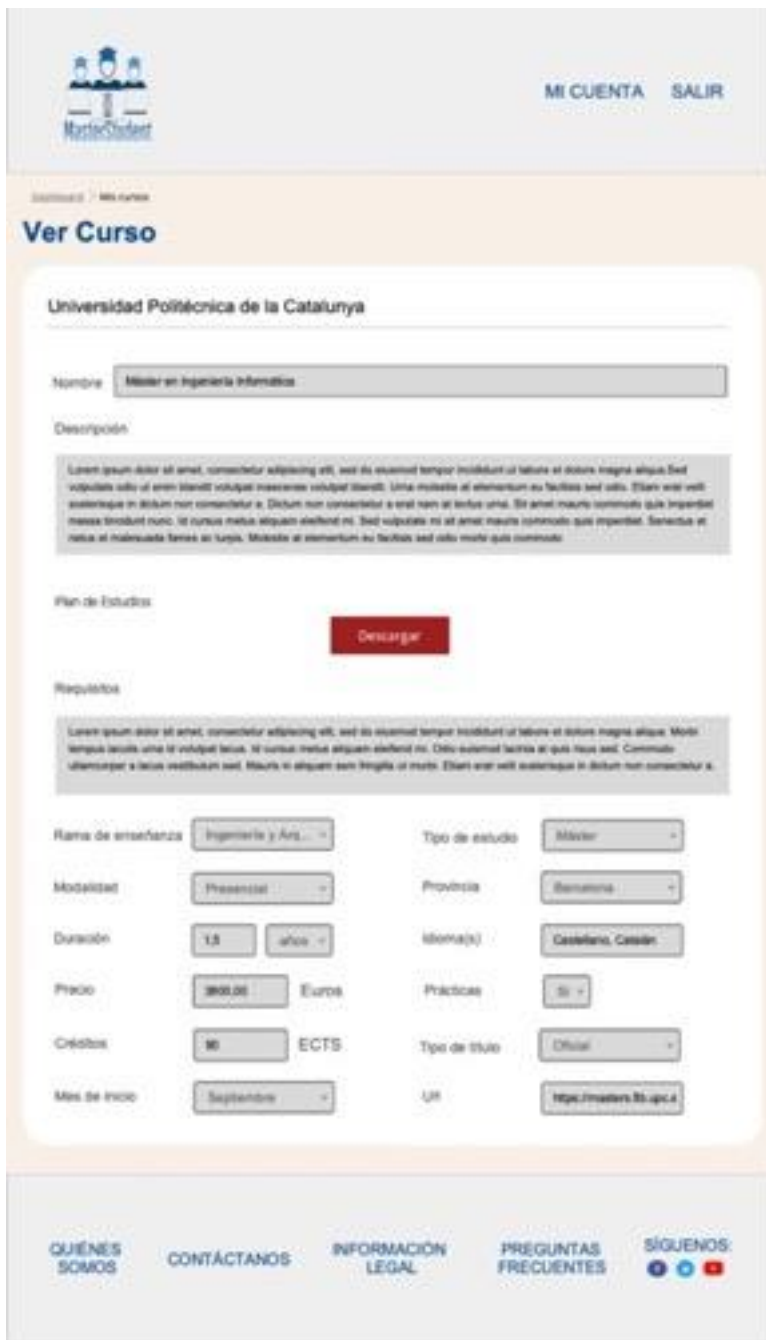


Figura 37 – Ver curso. Versión tablet.

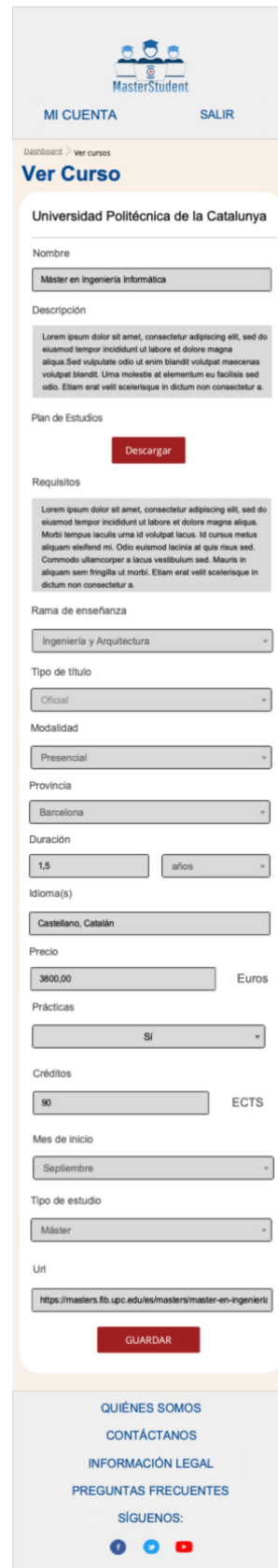


Figura 38 – Ver curso. Versión móvil.



[MI CUENTA](#)
[SALIR](#)

Dashboard > Nuevos alumnos interesados

Nuevos alumnos interesados- Universidad Politécnica de Catalunya

NUEVOS INTERESADOS

Nombre	Apellidos	Sexo	Email	Teléfono	Nivel de estudios	Nombre del curso	Consulta	F.consulta	Contactado
Santiago	Bruno	H	santiago31@gmail.com	689598854	E. universitarias	Master en Ing.Industrial	Ver	28/11/2021	No
Julio	Hernández	H	julioherna3@gmail.com	686635589	E. universitarias	Master en Ing.Informática	Ver	29/11/2021	No

ALUMNOS INTERESADOS

Nombre	Apellidos	Sexo	Email	Teléfono	Nivel de estudios	Nombre del curso	Consulta	F.consulta	F. contactado	Contactado
Juan	Bruno	H	juanb@gmail.com	689598854	E. universitarias	Master en Ing.Industrial	Ver	28/11/2021	29/11/2021	Sí
Elizabeth	Peréz	M	eper@gmail.com	689592551	E. universitarias	Master en Ing.Informática	Ver	28/11/2021	29/11/2021	Sí

[QUIÉNES SOMOS](#)

[CONTÁCTANOS](#)

[INFORMACIÓN LEGAL](#)

[PREGUNTAS FRECUENTES](#)

SÍGUENOS:



Figura 39 –Alumnos interesados. Versión escritorio.



Figura 40 –Alumnos interesados. Versión tablet.

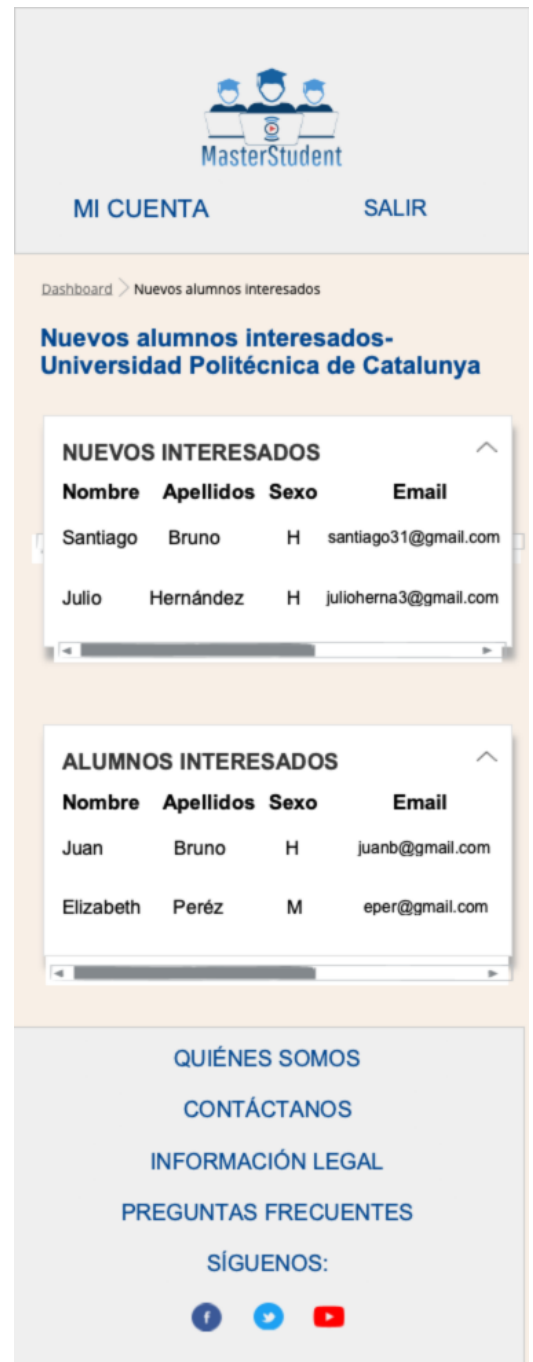


Figura 41 –Alumnos interesados. Versión móvil.

13. Análisis de mercado

13.1 Contexto social y de mercado

En el punto 1.1 Contexto y 1.2 Justificación se muestra el crecimiento del número de alumnos matriculados en un máster en los últimos años, concluyendo que este nivel académico es muy demandado en España y que existe un gran mercado para este sector educativo. Por lo tanto, para continuar con esta tendencia creciente, es importante que el proceso de búsqueda de cursos de postgrado sea un proceso sencillo y rápido.

Actualmente, existen dos enfoques de la problemática: Para los alumnos, la información se encuentra muy dispersa y es difícil encontrar los datos relevantes del curso de interés. Para las escuelas universitarias, que se encuentran en búsqueda constante de más canales de venta para la captación de alumnos.

13.2 Productos similares

En el punto 4 Marco teórico/Escenario, se lista los principales buscadores de curso de postgrado: Emagister y Educaweb; además, se realiza un análisis de cada uno de ellas, donde se concluye que la mayoría está orientado al fin comercial o marketing en línea, sin tener foco en la información de las escuelas públicas que también pueden ser de interés de los alumnos.

13.3 Estudio de mercado

El mercado de usuarios de la aplicación web se compone de dos tipos: Los alumnos, son aquellos profesionales que se encuentran en la búsqueda de un curso de postgrado para seguir creciendo profesionalmente. Las escuelas y/o universidades, son aquellas entidades educativas que buscan captar más alumnos para sus cursos de postgrado.

La diferencia que tiene la aplicación web de estudio con respecto a los competidores se basa en las siguientes razones:

- No solo se enfocará en universidades privadas sino también incluirá información de postgrado de universidades públicas.
- La principal prioridad de esta aplicación será que el alumno encuentre un curso según sus necesidades, en las escuelas y/o universidades de su preferencia sin

el sesgo comercial de las universidades privadas, las cuales en las aplicaciones de la competencia pagan más por ser primera prioridad dentro de la búsqueda.

- Para las universidades será fácil gestionar los cursos y los alumnos interesados debido a los patrones de usabilidad aplicados en el dashboard, donde se encontrarán las secciones “Mis cursos”, “Nuevo curso” y “Alumnos interesados”.

14. Viabilidad

Para realizar el análisis de la viabilidad del proyecto se tomó en consideración lo indicado en el blog académico del (Mba Executive - Cámara de Oviedo, 2020)

14.1 Viabilidad técnica

Para realizar este análisis se tendrá los requerimientos necesarios tanto de software, hardware, dominio y hosting que se visualizarán en la tabla 11 Presupuesto del equipo técnico y en la cual se puede encontrar más detalle.

Debido a que todo lo desarrollado es de software libre, excepto el hosting, dominio y hardware se considera que la creación y puesta en marcha de este proyecto es viable.

14.2 Viabilidad legal:

Para el desarrollo de la aplicación web no se incurrirá en ninguna infracción o responsabilidad legal porque la fuente de información de los cursos de postgrado es “Registro de Universidades, Centros y Títulos” (RUCT), que tiene carácter público y de registro administrativo con actualización constante.

14.3 Viabilidad económica:

Debido a que el presupuesto inicial del proyecto es bajo, tal como se verá más adelante en el punto de Presupuesto, se puede lanzar el nuevo producto al mercado.

Para que el proyecto sea rentable se espera recibir ingresos por publicidad de fundaciones y/o entidades educativas públicas y privadas tales como Fundación La Caixa, Fundación Banco Santander, Universidad Oberta de Catalunya entre otras.

14.4 Viabilidad de mercado del proyecto

Para analizar la viabilidad del proyecto se utilizará la herramienta estratégica DAFO para identificar Fortalezas, Oportunidades, Debilidades y Amenazas.

	Interno	Externo
Negativo	<p>Debilidades:</p> <ul style="list-style-type: none"> • No se tiene una API de las universidades públicas para que se retroalimenten. • Pocos recursos financieros propios. 	<p>Amenazas:</p> <ul style="list-style-type: none"> • Creciente número de competidores en el mercado con estrategias agresivas de marketing. • Ciberataques o actividad fraudulenta.
Positivo	<p>Fortalezas:</p> <ul style="list-style-type: none"> • Aplicación web con diseño enfocado en el alumno. • Contenido útil y relevante para la búsqueda de cursos. • Visualización de universidades públicas y privadas. • Navegación intuitiva para facilitar la gestión de cursos. 	<p>Oportunidades:</p> <ul style="list-style-type: none"> • Cada vez más alumnos prefieren utilizar buscadores de cursos que entrar a las páginas de escuelas. • Afán de las escuelas en buscar más medios de difusión de sus cursos.

Tabla 9- Herramienta estratégica DAFO

Dado que las fortalezas son mayores que las debilidades, la aplicación web se presentará en el mercado como un producto sólido, con ventajas competitivas respecto a la competencia y sobre todo con un elemento diferenciador que es el foco en el alumno.

Al analizar los factores externos, se tiene el mismo número de oportunidades y amenazas; sin embargo, se considera que la probabilidad de que ocurra dichas amenazas es baja. Por lo tanto, de acuerdo a lo mencionado anteriormente el proyecto resulta viable.

15.Presupuesto

El sistema web para la búsqueda de cursos de postgrados españoles es un proyecto académico, por lo que no se hará un estudio profundo acerca de los costes relacionados del mismo. Sin embargo, se hará una estimación lo más cercana a la realidad, por si alguna institución educativa y/o financiera estuviera interesada.

Dicho esto, a continuación, se verá el **presupuesto del equipo humano**. Aunque en un entorno empresarial, se debería contar con perfiles más especializados para calcular este presupuesto, en este proyecto se está considerando como recurso humano, una persona.

Además, como se puede observar en el punto 10. Planificación, se tuvo en cuenta las 4 partes de la planificación: Análisis/Requisitos, Diseño, Implementación/pruebas y entrega final, que se realizará en 111 días que equivalen a 444 horas y será realizado por una persona del perfil programador web de 30€/hora.

EQUIPO HUMANO			
Fase del proyecto	Tiempo	Horas	Importe (€)
Análisis/Requisitos	14 días	56 horas	1680€
Diseño	29 días	116 horas	3480€
Implementación	39 días	156 horas	4680€
Entrega final	29 días	116 horas	3480€
Total	111 días	444 horas	13320,00 €

Tabla 10- Presupuesto del equipo humano.

Para el **presupuesto del equipo técnico**, se analizó algunas de las alternativas de dominio y hosting, y se decidió contratar una “Plataforma como servicio” (PaaS) de Amazon AWS, el cual es un entorno de desarrollo e implementación completo en la Nube.

Debido a que es un proyecto académico, como recurso, solo se necesitará la capa gratuita de Amazon AWS. Estos servicios son del tipo “Elásticos”, lo que significa que a medida que vaya creciendo la aplicación se necesitarán más recursos y por lo tanto tendrá un mayor costo.

La capa gratuita de Amazon incluye, a grandes rasgos, Almacenamiento S3 de 5GB, Base de datos RDS 750horas al mes de uso, entre otros. (Amazon, 2022)

Finalmente, se verificó que para el dominio Masterstudent.es tiene el costo de 10,83€ en Amazon Route 53, y el hosting es gratuito por 12 meses con limitaciones.

EQUIPO TÉCNICO	
Concepto	Importe (€)
Dominio en AWS: MasterStudent.es	10,83 €
Hosting en AWS (Capa Gratuita- 12 meses)	0€
Ordenador Mac Pro (Hardware)	2000€
Software(s)	0€
Total	2010,83 €

Tabla 11- Presupuesto del equipo técnico.

El *presupuesto de recursos adicionales* se basa en lo siguiente:

RECURSOS ADICIONALES			
Concepto	Precio (€)	Meses	Importe Total (€)
Internet de fibra	70€	4	280 €
Servicio de luz	50€	4	200 €
Material de oficina	10€	4	40 €
Total			520 €

Tabla 12- Presupuesto de los recursos adicionales

Una vez que se realizó los presupuestos parciales, anteriormente mencionados, se hará la sumatoria de los mismos en la siguiente tabla:

PRESUPUESTO TOTAL	
Equipo Humano	13320,00 €
Equipo Técnico	2010,83 €
Recursos	520,00 €
Total	15850,83 €

Tabla 13- Presupuesto total

16. Usabilidad/UX

16.1 Patrones de Layout

Los patrones de Layout utilizados en este proyecto son la combinación de Mostly Fluid, Off Canvas y Tiny tweaks, clasificados de mayor a menor medida de uso en el proyecto.

Mostly Fluid

El patrón Mostly fluid consiste, principalmente, en una cuadrícula fluida. Por lo general, en las pantallas grandes o medianas se mantiene el mismo tamaño y simplemente se ajustan los márgenes en las más anchas. En las pantallas más pequeñas, la cuadrícula fluida genera el reprocesamiento del contenido principal, mientras que las columnas se apilan verticalmente. (Developers google, 2021).

Este patrón se utiliza en todas las pantallas de este proyecto, debido a su beneficio de no cambiar mucho el diseño, sino ajustarse al tipo de pantalla.

Off Canvas

En lugar de apilar contenido verticalmente, el patrón Off canvas coloca contenido menos usado (tal vez menús de navegación o de apps) fuera de la pantalla y solo lo muestra cuando el tamaño de la pantalla es suficientemente grande. (Developers google, 2021).

Este patrón se utiliza especialmente en la página de búsqueda de curso para ocultar los filtros en dispositivos móviles, es decir mostrándose cuando el usuario lo necesite y ocultándose cuando no.

Tiny tweaks

Este patrón se utiliza para realizar pequeños cambios. Un claro ejemplo es el tamaño de letra y el cambio de tamaño de imagen en diferentes dispositivos responsive de toda la aplicación web. Además, se utiliza en los paneles que tienen contenido en formato de tabla, debido a que son diseños de una sola columna y tienen que mantener su estructura en cualquier dispositivo.

16.2 Patrones de diseño de interfaces

Para la identificación de los patrones se utilizó el libro “Practical UI Patterns for Design Systems” de (MacDonald, 2019) y de la página web (Ui-patterns, 2021) :

Migas de pan

Este patrón nos permite saber la ubicación dentro de la estructura jerárquica del sitio web.

El beneficio lo tenemos en toda la aplicación web, un ejemplo de uso es cuando navegamos desde el home a la búsqueda de cursos y en la parte superior de la pantalla aparece una guía de ubicación en base a la estructura jerárquica de la aplicación web.

Caja de búsqueda

Este patrón te permite encontrar información específica. El beneficio en este proyecto es que nos permite realizar la búsqueda de un curso de interés en la página de inicio(home) y en la mayoría de páginas del perfil visitante.

Filtros múltiples

Este patrón te permite seleccionar varios filtros al mismo tiempo, es decir, se puede ir mostrando los resultados coincidentes, según se va seleccionando los filtros que desee el usuario.

Su beneficio en este proyecto es que nos permite hacer un filtrado más avanzado de los cursos ya buscados y lo hace de acuerdo a las preferencias del usuario ya que podrá seleccionarlo de manera directa y concisa.

Paginación

Este patrón nos permite separar grandes contenidos en diferentes páginas, en el cual se accede mediante un índice numérico.

Su beneficio en el proyecto lo tenemos cuando los resultados de búsqueda de cursos son extensos, y la creación de este patrón nos permite mostrar los resultados parcialmente a medida que se seleccione el índice.

17. Tests

En el proyecto se considero importante aplicar el test de funcionalidad a modo de pruebas unitarias.

17.1 Pruebas unitarias

Este tipo de pruebas aseguran que una determinada unidad de código funciona como se esperaba. En la aplicación se aplicaron test a los servicios implementados con Karma/Jasmine, entre ellos podemos mostrar un extracto del test de cursos.

```

it('Lista de cursos por universidad',(done)=>{
  const idUniversity= "UOC";
  coursesService.getCoursesListByUniversity(idUniversity).subscribe((listUni)=>{
    expect(listUni).toBe(mockCourses);
    done();
  });

  //Simulate asynchronous
  const req = httpTestingController.expectOne(`${api}/curso/universidad/UOC`);
  expect(req.request.method).toBe('GET');
  req.flush(mockCourses); // Mock the return value of http.get
});

it('Agregar curso ',(done)=>{
  const res = {
    text: 'Curso guardado',
    idCourse: 200
  }
  coursesService.saveCourse(newMockCourse).subscribe((msg)=>{
    expect(msg).toBe(res);
    done();
  });

  const req = httpTestingController.expectOne(`${api}/curso`);
  expect(req.request.method).toBe('POST');
  req.flush(res);
});

it('Eliminar curso',(done)=>{
  const id = 200;
  const res = { message: 'Curso y Plan de estudios eliminado ' };

  coursesService.deleteCourse(id).subscribe((msg)=>{
    expect(msg).toBe(res);
    done();
  });

  const req = httpTestingController.expectOne(`${api}/curso/${id}`);
  expect(req.request.method).toBe('DELETE');
  req.flush(res);
});

```

Figura 42 – Prueba unitaria – Course test

18.Seguridad

Por motivo de seguridad al realizar las llamadas a la API REST, se desarrolló JWT. Según (JWT, 2021) es “un método estándar abierto RFC 7519 para presentar peticiones de forma segura entre dos partes. La página oficial JWT le permite decodificar, verificar y generar JWT”. Esto nos servirá para generar (codificar) y verificar token cuando se desee acceder mediante el formulario de login, devolviendo un token al front-end en caso fuese exitoso el acceso mediante usuario y contraseña. Además, permite acceder a los recursos si el usuario está autorizado. Para más detalle, se puede observar la Figura 42- Token JWT - Registro, Login, Acceso Recurso.

Características del JWT en la aplicación:

- El token tiene validez de tres horas. Es decir, la universidad tendrá este tiempo con la sesión iniciada.
- El token es guardado en “window.sessionStorage” mediante el TokenStorageService. Es decir, cuando el cliente necesite utilizarlo puede recuperarlo.
- En el cliente, se tiene un interceptor de Angular para añadir el token en la cabecera de la solicitud (HTTP), para así poder validar el recurso. Además, se aseguraron las rutas del perfil universidad mediante guards de Angular.
- En el back-end, se tiene un middleware llamado check-token.ts, para autorizar los recursos solicitados. Los recursos que están protegidos son los del perfil universidad tales como:

CURSOS	ALUMNOS INTERESADOS
Consultar lista de cursos de una universidad logueada.	Consultar lista de alumnos interesados
Editar curso de una universidad	Consultar lista de alumnos contactados
Eliminar curso de una universidad	Actualizar lista de alumnos interesados/contactados
Crear nuevo curso para una universidad	

Tabla 14- Autorización de recursos para Universidad - JWT

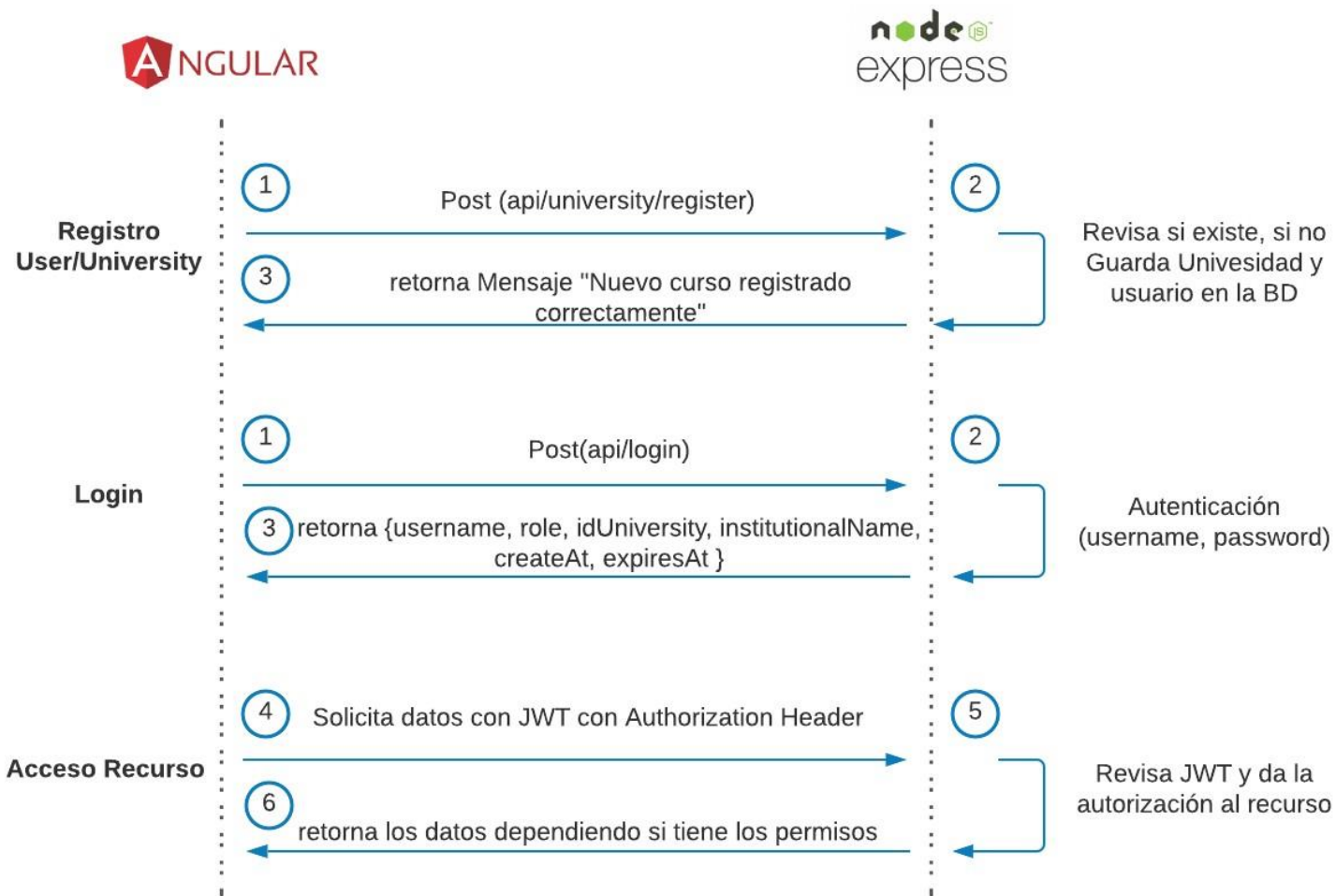


Figura 43 –Token JWT - Registro, Login, Acceso Recurso

19.Requisitos de instalación/implantación/uso

La información detallada es la siguiente:

- **Software**

Front-end. Los requisitos para ejecutar Angular son:

- Node 16.13.0 . Se puede descargar desde la página oficial <https://nodejs.org/en/>
- Npm 8.1.0. Viene instalado por defecto en Node, es decir, con instalar Node ya tendremos instalado el npm. Para mayor información: <https://www.npmjs.com/>
- Visual Studio Code. Se puede descargar directamente de su página oficial: <https://code.visualstudio.com/>
- Navegador moderno: Google Chrome, Mozilla Firefox, Safari, otro.

Back-end. Los requisitos para ejecutar Node/Express y BD.

- Base de datos MySQL 8.0.27
- MySQL Workbench 8.0.27, herramienta para visual para el diseño de base de datos. Se puede descargar desde su página oficial: <https://www.mysql.com/products/workbench/>
- Node y Npm, antes mencionadas, para instalar dependencias.

- **Hardware**

- Procesador mínimo i5 a partir de 2.4 Ghz.
- Memoria RAM de 16Gb.
- Almacenamiento de Disco como mínimo de 250GB.

- **Formación**

- Se necesita tener alguna formación mínima en el desarrollo y/o implementación de software. Es decir, cualquier grado académico relacionado con la Ingeniería Informática.

20. Instrucciones de instalación/implantación

Antes de comenzar verificamos los requisitos necesarios para el front-end y back-end detallados en el punto anterior y se procede a clonar el repositorio: <https://github.com/santtt3131/masterStudent.git>, aquí se encontrará todo el proyecto tanto back-end, base de datos, front-end.

20.1 Back-end

20.1.1 Local

- Se ejecuta el script de la base datos, que se encuentra en el directorio **mysql/masterstudent.sql** . Para ello se abre el MySQL Workbench, se abre una consulta, se copia el script y se ejecuta.

Luego, vemos que se crea la base de datos “masterstudent” con la data.

- A continuación, abrimos el Visual Studio Code con el proyecto clonado, y nos dirigimos al directorio server, el cual contiene todo el backend, seleccionamos click derecho y abrimos un terminal. Se explicará los pasos a seguir a continuación:
 - Instalamos las dependencias para ello escribimos: ***npm install***
 - Se construye el build con: ***npm run build***
 - Ejecutamos nodemon, el cual “es una herramienta que ayuda a desarrollar aplicaciones basadas en node.js al reiniciar automáticamente la aplicación de node cuando se detectan cambios de archivo en el directorio” (Nodemon, 2021). Para ello se ha configurado internamente y solo tenemos que ejecutar ***npm run dev***.

Con ello tendremos nuestro servidor levantado, escuchando en el puerto 3000. Para más información se puede mirar index.ts.

20.1.2 Servidor - AWS

En este punto primero se subirá la base de datos en:


<https://aws.amazon.com/es/rds/> (Amazon Web Services - Amazon Relation Database Service(RDS), 2022)

- Creamos la base de datos y realizamos las configuraciones pertinentes.

- Una vez creada copiamos el punto de enlace (endpoint), nos conectamos con MySQL Workbench y continuamos con los pasos indicados en el punto 19.1.1.

Para el backend se desplegará **AWS – EC2**, entre los pasos estarán:

- Se accede a la web <https://aws.amazon.com/es/ec2/> (Amazon Elastic Compute Cloud - Amazon EC2, 2022) y se crea una instancia según sea conveniente como por ejemplo Ubuntu, configurándolo a nuestra preferencia, entre ellas: grupos de seguridad como ssh, http, https, TCP. Antes de lanzar la instancia, Amazon nos facilitará una clave que tendremos que guardarla en formato .pem.
- Nos conectamos mediante nuestra consola de window/Linux/mac al ssh de nuestra instancia, con el siguiente comando : **ssh -i "backend_masterstudent.pem" ubuntu@ec2-63-33-62-208.eu-west-1.compute.amazonaws.com**
- Una vez conectados, se clona el repositorio del github.
- Se accede mediante comandos al directorio server e instala todas las dependencias.
- Un punto importante es crear dentro del directorio un fichero llamado .env, el cual contendrá todas las variables de entorno del backend.



```
.env
masterStudent > server > .env
1  PORT=3000
2  MYSQL_HOST=
3  MYSQL_USER=
4  MYSQL_PASS=
5  MYSQL_DATABASE=
6  TOKEN_JWT=
7  AWS_ACCESS_KEY_ID=
8  AWS_SECRET_ACCESS_KEY=
```

Figura 44 – Variables de entorno del backend

- Realizamos un build con el siguiente comando: **npm run build**
- Por último, se instala PM2. Según la página oficial “es un administrador de procesos demonio que lo ayudará a administrar y mantener su aplicación en

línea las 24 horas del día, los 7 días de la semana” (PM2, 2022) y se ejecuta el siguiente comando: **pm2 start build/index.js**

20.2 Front-end

20.2.1 Local

- Se procede a instalar Angular ejecutando desde la consola (cmd, terminal, consola de Visual Studio Code) el siguiente comando: **npm install -g @angular/cli**
- Con el Visual Studio Code abierto, se accede mediante consola a la carpeta client/masterStudent, la cual contiene todo nuestro código front-end y a continuación se instalan todas las dependencias, ejecutando el siguiente comando: **npm install**
- Luego para levantar el servidor se ejecuta el siguiente comando: **ng serve ó npm run start** y tendremos nuestra aplicación web totalmente funcional.

20.2.2 Servidor-AWS

- Se modifica la ruta con la cual se consumirá la api para producción, que básicamente es la ip pública del backend. Este archivo se llama apiPath.ts.
- Se compila el front para producción con el comando: **ng build --prod**
- Se crea un bucket para “Alojamiento de sitios web estaticos”, en <https://aws.amazon.com/es/s3/> (Amazon S3, 2022). A continuación, se sube todos los archivos generados en el directorio “**dist**” al bucket creado.
- Se crea una “política de bucket”:

Política de bucket

La política del bucket, escrita en JSON, proporciona acceso a los objetos almacenados en el bucket. Las políticas de bucket no se aplican a los objetos que pertenecen a otras cuentas. [Más información](#)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicRead",
      "Effect": "Allow",
      "Principal": "*",
      "Action": [
        "s3:GetObject",
        "s3:GetObjectVersion"
      ],
      "Resource": "arn:aws:s3:::master-student-university-frontend/*"
    }
  ]
}
```

Figura 45 – Política bucket – AWS S3 (Frontend)

- Una vez realizado todos estos pasos. Podemos acceder al sitio web:
 - <http://master-student-university-frontend.s3-website-eu-west-1.amazonaws.com>

21.Instrucciones de uso

21.1 Pantalla de inicio (Home)

Al entrar en la aplicación se encuentra la página de inicio que se divide entre 3 partes:

En la parte superior izquierda se encuentra el logo de aplicación y al lado derecho, dos enlaces públicos llamados “ACCESO” y “REGÍSTRATE”.

En la parte central se encuentra una caja de texto y un botón de búsqueda representado como una lupa, en donde se puede poner la palabra clave para comenzar dicha búsqueda. Además, hay una sección de los Master y/o postgrados más solicitados, que puede servir de referencia para la persona que realiza la búsqueda.

En la parte inferior se encuentra los enlaces “Quiénes somos”, “Contáctanos”, “Información legal”, “ Preguntas frecuentes” y redes sociales. Esta sección se repetirá en todas las pantallas.

21.2 Búsqueda y filtrado de cursos

Es una página de contenido público que se muestra luego de realizar la búsqueda en la página de inicio.

En la parte superior a diferencia del home, tiene una caja de texto y un botón de búsqueda, en caso se necesite buscar otro curso.

En la parte central izquierda, se muestra un panel con opciones para filtrar cursos en base a los siguientes criterios como: Rama de enseñanza (Ciencias Sociales y Jurídicas), tipo de estudio (Máster y Postgrado), Modalidad (combinadas entre presencial, semipresencial y online), Provincia (todas dentro de España), Idioma (Castellano, Catalán, Inglés), Precio (rangos a elegir). Además, tiene un botón de “Limpiar” y “Aplicar” filtro. Estas funcionalidades son similares para dispositivos móviles, con la diferencia que existe un botón llamado “Filtros” para que aparezca y desaparezca el mismo.

En la parte central derecha, se muestra los resultados de los filtros aplicados, representados por una “card” o caja donde se puede observar el nombre del curso, nombre de la universidad, el tipo de título (Oficial o propio), modalidad, precio y numero

de créditos ECTS, estos 4 últimos tienen un icono para una mejor visualización. Además, se tiene un botón llamado “Solicitar información” de color guinda para ponerse en contacto con la escuela.

Al final del “resultado de búsqueda”, se observa el componente de Paginación que nos permite separar el resultado por páginas y por bloques de “card”.

21.3 Detalle del curso

Es una página que se muestra luego de haber hecho un click en el nombre del curso de las cards de resultados.

En la parte central se muestra, el nombre del curso de color azul oscuro como cabecera principal, y cabecera secundaria el nombre de la universidad de color negro.

A continuación, se muestra los paneles con la información del detalle del curso:

- Descripción: explica con más detalle el curso.
- Plan de estudios: Se muestra un botón para descargar el plan de estudios.
- Requisitos: Son los necesarios para acceder a la titulación.
- Datos generales: Se muestra toda la información necesaria del curso. Entre ellos esta Rama de enseñanza, tipo de estudio, modalidad, provincia, duración, idioma, precio, prácticas, créditos, tipo de título, mes de inicio y url.

Estos paneles se pueden abrir y cerrar según la preferencia del usuario. Además, al final hay un botón de “Solicitar información”, si el futuro alumno necesita alguna información extra.

21.4 Solicitar información del curso

Se puede acceder a esta página de dos vías diferentes: La primera, luego de mostrar los resultados de la búsqueda, cada card tiene un botón “Solicitar Información”. La segunda, luego entrar al detalle de curso al final de los paneles se muestra el botón.

Esta página se caracteriza porque es una modal (pop-up) donde por medio de un formulario el usuario puede rellenar sus datos para que posteriormente se comuniquen con el. Entre sus campos están: nombre, apellidos, sexo, email, teléfono, nivel de

estudios, consulta. Y luego se puede dar click en el botón “Enviar”, para que la Universidad lo revise y puedan entrar en contacto con el alumno interesado.

21.5 Registrar Universidad

Si la universidad desea registrarse, se accederá desde la parte superior derecha de la cabecera de la página home al enlace “REGÍSTRATE”, el cual muestra un formulario con los datos necesarios para el registro. Adicionalmente también se pide completar el campo usuario y contraseña que desea crear, y subir el logo de universidad, que permitirá eliminar y volver a subirlo si es necesario. Una vez rellenado estos datos, se puede dar click en el botón “Registrar”.

21.6 Inicio de sesión

Si la universidad desea iniciar sesión, se accederá desde la parte superior derecha de la cabecera de la página home al enlace de “ACCESO”, el cual muestra el típico formulario para completar usuario y contraseña, con el botón “Iniciar sesión”. Si el usuario o contraseña son incorrectos aparece un mensaje de error en rojo. En caso que sean correctos, se redirige al Dashboard (tablero) de la Universidad.

21.7 Dashboard de la Universidad

Es la página que se muestra, una vez que la universidad inicie sesión. El dashboard contiene un mensaje de bienvenida, con el logo y nombre de la universidad. A continuación, se muestra 3 bloques representados por los botones: “Mis cursos”, “Nuevo curso”, “Alumnos interesados”, los cuales se explican a continuación.

21.8 Mis cursos – Universidad

En esta página se muestra como título “Mis cursos” y a continuación el nombre de la universidad que se ha logueado. Además, muestra el listado de cursos categorizados por la rama de enseñanza. Dentro de cada panel se muestra la información relevante de los cursos como Nombre, Modalidad, Créditos, Tipo de título y Acciones como:

- Ver: Muestra toda la información acerca del curso seleccionado y también permite realizar la descarga del plan de estudios.
- Editar: Permite realizar algún cambio en los diferentes campos del curso.
- Eliminar: Al seleccionar este botón, se muestra un pop up con un mensaje de confirmación para eliminar el curso.

21.9 Nuevo curso – Universidad

En esta página se muestra un formulario para completar la información de un nuevo curso, entre ellos también es necesario subir el plan de estudios en los formatos permitidos (.pdf, .doc, .docx). Una vez completada la información, se debe seleccionar Guardar para registrar la información y si todo ha sido satisfactorio, aparecerá un mensaje indicando que se ha registrado correctamente.

21.10 Nuevos alumnos interesados

En esta página se encuentra el listado de alumnos que han solicitado información previamente, el cual tiene como título “Nuevos alumnos interesados” y a continuación el nombre de la universidad que se ha logueado.

Además, muestra dos paneles: El primero llamado “Nuevos Interesados”, donde se muestra la información completada por los alumnos interesados adicionando la fecha de consulta. En este panel, la universidad puede ver la consulta y cambiar el estado de no contactado a contactado, por medio de un desglosable.

El segundo panel llamado “Alumnos Contactados”, además de los campos anteriormente descritos, se agregará un campo de fecha de contacto; es decir, se podrá visualizar tanto la fecha de consulta como la de contactado.

22. Proyección a futuro

MasterStudent, que es una aplicación web para la búsqueda de cursos de postgrados españoles, no deja de ser una aplicación que se ha desarrollado en el ámbito académico para la obtención del Máster. Es por ello que se tienen las siguientes mejoras para una proyección a futuro:

- A nivel empresarial, existe la necesidad de crear un **perfil Administrador** la cual te permite tener una mejor visión de los indicadores de gestión y control de la aplicación, donde se pueda **gestionar estadísticas generales** como número de alumnos que solicitan información y número de alumnos contactados por universidad; así como **gestionar las tablas paramétricas, modificar el contenido visual**, entre otros.
- En la página Home se podría desarrollar una funcionalidad que muestre los másteres más solicitados, considerando las solicitudes de los alumnos interesados y así se muestren los cursos más demandados en forma de “carrusel” con actualización en tiempo real, para que sirva de referencia al visitante.
- Afinar la búsqueda de los cursos mediante etiquetas con palabras claves, por ejemplo, si se busca por “MBA” y el título del curso no incluye esta palabra, no se buscará este curso. En cambio, si se agrega términos por etiquetas, esto ayudará a verificar que el curso está relacionado o es un MBA, así se fortalecerá y precisará dicha búsqueda.
- Implementación de distintos idiomas para la aplicación web. Esto facilitará la experiencia de los usuarios no hispanohablantes, permitiendo llegar a un público más amplio que desee llevar una experiencia enriquecedora en una escuela o universidad española.

23. Conclusiones

El desarrollo de la aplicación web **MasterStudent**, ha sido un gran reto académico y profesional para mí. La idea surgió de una inquietud personal sobre los másteres que ofrecía el mercado español en el área de Informática y no poder encontrar una página o aplicación donde se encuentre todos los másteres tanto de escuelas o universidades públicas y privadas; por lo tanto, sentí una gran motivación para alcanzar los objetivos de este proyecto. A nivel general, estoy muy contento y orgulloso con la aplicación desarrollada, ya que me ha permitido poner en práctica todos los conocimientos adquiridos en las asignaturas del máster.

Al haber realizado la aplicación completa con tecnologías basada en javascript como Angular, Node y Express, me ha permitido explorar, conocer y profundizar en diversos temas que desconocía o que necesitaba consolidar tales como seguridad en las apis con JWT, guards, Angular Material, entre otros. Adicionalmente, se utilizó el framework de CSS como Tailwind, base de datos MySQL y se aprendió a subir una aplicación web a producción como es Amazon Web Services, el cual aún es nuevo mundo tecnológico por explorar.

El resultado obtenido es muy gratificante, ya que se ha podido realizar todos los hitos estipulados en la planificación mediante la metodología en cascada de Winston Royce, cumpliendo así con los objetivos fijados.

Anexo 1. Entregables del proyecto

Los documentos que componen la entrega final del proyecto son los siguientes:

- Carpeta: **Documentación**
 - Archivo: PAC_FINAL_mem_BrunoPerez_SantiagoAndres.pdf
Contenido: La presente memoria completa del trabajo de fin de master.
 - Archivo: PAC_FINAL_funcionalidades_BrunoPerez_SantiagoAndres.pdf
Contenido: Detalle de funcionalidades principales.
 - Archivo: PAC_FINAL_autoevaluacion_BrunoPerez_SantiagoAndres.pdf
Contenido: Informe de autoevaluación.
 - Archivo: PAC_FINAL_prs_cliente_BrunoPerez_SantiagoAndres.pdf
Contenido: Presentación dirigida al público en general, no especializado.
 - Archivo: PAC_FINAL_prs_tribunal_BrunoPerez_SantiagoAndres.pdf
Contenido: Presentación dirigida al tribunal de evaluación.

- Carpeta: **Proyecto**
 - Archivo: PAC_FINAL_prj_BrunoPerez_SantiagoAndres.zip
Contenido: Código fuente del back-end, front-end, y script de la base de datos mysql.

 - Carpeta: Ejecutables
Contenido: Se encuentran todos los ejecutables como Justinmind, Gantt, diagrama de clases, casos de uso, entre otros.

- Carpeta: **Multimedia**
 - Archivo: PAC_FINAL_vid_BrunoPerez_SantiagoAndres.mp4
Contenido: Presentación en video destinada al tribunal de evaluación.

Anexo 2. Código fuente (extractos)

BACK-END

- Extracto del controlador de curso en Node/Express: courseController.ts

```
import { Request, Response } from 'express';

import pool from '../database';

class CourseController {

  public async getListCourseByUniversity(req: Request, res: Response): Promise<any> {
    const { id } = req.params;
    const course = await pool.query('SELECT * FROM Course C,University U'+
    ' WHERE C.idUniversity = U.idUniversity and C.idUniversity= ?', [id]);
    if (course.length > 0) {
      return res.json(course);
    }
    res.status(404).json({ text: "Curso no encontrado" });
  }

  public async getCourseById(req: Request, res: Response): Promise<any> {
    const { id } = req.params;
    const course = await pool.query('SELECT * FROM Course C WHERE C.idCourse= ?', [id]);
    if (course.length >= 0) {
      return res.json(course[0]);
    }
    res.status(404).json({ text: "Curso no encontrado" });
  }

  public async create(req: Request, res: Response): Promise<void> {
    const course = await pool.query('INSERT INTO Course set ?', [req.body]);
    res.json({
      text: 'Curso guardado',
      idCourse: course.insertId
    });
  }

  public async delete(req: Request, res: Response): Promise<void> {
    const { id } = req.params;
    await pool.query('DELETE FROM Syllabus WHERE idCourse = ?', [id]);
    await pool.query('DELETE FROM Course WHERE idCourse = ?', [id]);
    res.json({ message: 'Curso y Plan de estudios eliminado ' });
  }

  public async updateCourse(req: Request, res: Response): Promise<void> {
    const { id } = req.params;
    await pool.query('UPDATE Course set ? WHERE idCourse = ?', [req.body, id]);
    res.json({ message: 'La universidad fue actualizada' });
  }

  public async uploadSyllabus(req: Request, res: Response): Promise<void> {
    res.json({ text: 'Syllabus guardado en local' });
  }

  public async downloadSyllabus(req: Request, res: Response): Promise<void> {
    res.json({ text: 'Syllabus descargado' });
  }

  public async createSyllabus(req: Request, res: Response): Promise<void> {
    await pool.query('INSERT INTO Syllabus set ?', [req.body]);
    res.json({ text: 'Syllabus guardado' });
  }
}
```

- Fichero de rutas de cursos realizado en Node/Express. Se puede observar que algunas rutas están protegidas con el middleware checkToken.

```
import { Router } from 'express';
import { cursoController } from '../controllers/courseController';
import { syllabus } from '../middleware/upload-file';
import { downloadSyllabus } from '../middleware/download-file';
import { checkToken } from '../middleware/check-token';

class CourseRoutes{
  public router: Router = Router();

  constructor(){
    this.config();
  }

  config():void{
    this.router.get('/universidad/:id', checkToken , cursoController.getListCourseByUniversity );
    this.router.get('/:id', cursoController.getCourseById );
    this.router.post('/', checkToken, cursoController.create );
    this.router.post('/syllabus/', checkToken, cursoController.createSyllabus);
    this.router.post('/syllabus/upload/file', checkToken, syllabus.single('file'), cursoController.uploadSyllabus);
    this.router.post('/syllabus/download/file', downloadSyllabus , cursoController.downloadSyllabus);
    this.router.delete('/:id', checkToken, cursoController.delete);
    this.router.put('/:id', checkToken , cursoController.updateCourse);
    this.router.get('/download/syllabus/:id', cursoController.getSyllabus );
  }
}

const courseRoutes = new CourseRoutes();
export default courseRoutes.router;
```

- Fichero middleware check-token.ts.

```
import keys from "../keys";

const jwt = require('jwt-simple');
const moment = require('moment');

const checkToken = (req, res, next) => {
  if (!req.headers['authorization']) {
    return res.json({
      error: "You must include the header"
    });
  }
  let token = req.headers['authorization'];

  let payload = {
    username: '',
    idUniversity: '',
    createdAt: '',
    expiresAt: ''
  };

  try {
    token = token.replace("Bearer ", "");
    payload = jwt.decode( token, keys.database.tokenJWT);
  } catch (error) {
    return res.json({
      error: 'Invalid token'
    });
  }
  if (payload) {
    if (moment().unix() > payload.expiresAt) {
      return res.json({ error: 'Expired token' });
    }
  }

  req.username = payload.username;
  next();
};

export const checkTokenenn = checkToken;
```

FRONT-END

- Servicio de cursos: courses.services.ts . Vemos que son las funciones consumidoras de nuestra API del back-end.

```

import { Injectable } from '@angular/core';
import { HttpClient, HttpHeaders } from '@angular/common/http';
import { Observable } from 'rxjs';
import { Course } from '../models/Course';
import { Syllabus } from '../models/Syllabus';
import apiPath from './apiPath';

@Injectable({
  providedIn: 'root'
})
export class CoursesService {
  API_URI = apiPath.API_GLOBAL;

  constructor(private http: HttpClient) { }

  httpOptions = {
    headers: new HttpHeaders({ 'Content-Type': 'application/json' }),
  };

  getCoursesListByUniversity(idUniversity: string): Observable<Course[]> {
    return this.http.get<Course[]>(`${this.API_URI}/curso/universidad/${idUniversity}`);
  }

  getCourseById(idCourse: number): Observable<Course> {
    return this.http.get<Course>(`${this.API_URI}/curso/${idCourse}`, this.httpOptions);
  }

  updateCourseById(course: Course): Observable<Course> {
    return this.http.put<Course>(`${this.API_URI}/curso/${course.idCourse}`, course);
  }

  deleteCourse(idCourse: number): Observable<any> {
    return this.http.delete<Course>(`${this.API_URI}/curso/${idCourse}`, this.httpOptions);
  }

  saveCourse(course: Course) {
    return this.http.post(`${this.API_URI}/curso`, course, this.httpOptions);
  }

  saveSyllabus(syllabus: Syllabus) {
    return this.http.post(`${this.API_URI}/curso/syllabus`, syllabus, this.httpOptions);
  }

  saveFileSyllabus(syllabusFile: any) {
    return this.http.post(`${this.API_URI}/curso/syllabus/upload/file`, syllabusFile);
  }

  downloadFileSyllabus(nameSyllabus: string) {
    let body = {
      filename: nameSyllabus
    };
    return this.http.post(`${this.API_URI}/curso/syllabus/download/file`, body, {
      responseType: 'blob',
      headers: new HttpHeaders().append('Content-Type', 'application/json')
    });
  }

  getNameFileMySQL(idCourse:number){
    return this.http.get<Syllabus>(`${this.API_URI}/curso/download/syllabus/${idCourse}`);
  }
}

```

- Componente FileUploadMaster, es un componente para subida de archivos, que es utilizado para subir logos, como plan de estudios.

```
export class FileUploadMasterComponent implements OnInit {
  file_accept: string[] = [];
  textExtension: string = '';
  subscription: Subscription;
  @Output() onFileSelect: EventEmitter<any> = new EventEmitter();

  public fileUploadControl:any;

  constructor(
    private router: Router
  ) {
  }

  ngOnInit(): void {
    this.checkUploadComponent();
    this.subscription = this.fileUploadControl.valueChanges.subscribe(
      (file:any) => {
        this.onFileSelect.emit(file[0]);
      }
    );
  }

  checkUploadComponent() {
    let path = this.router.url;
    //upload logo or syllabus
    if (path.includes('registro')) {
      this.file_accept = ['.jpeg', '.jpg', '.png'];
      this.textExtension = '.jpeg, .jpg, .png';
    } else if (path.includes('nuevo-curso')) {
      this.file_accept = ['.pdf', '.doc', '.docx'];
      this.textExtension = '.pdf, .doc, .docx.';
    }
  }

  this.fileUploadControl = new FileUploadControl(
    {
      listVisible: true,
      discardInvalid: true,
      accept: [this.file_accept.join()],
    },
    [FileUploadValidators.accept(this.file_accept),
    FileUploadValidators.fileSize(1000000),
    FileUploadValidators.filesLimit(1)
  ]
  );
}
}
```

- En el typescript del componente UniversityCourseForm, se puede observar un ejemplo de las validaciones que se implementaron en el formulario nuevo/editar/ver curso, tales como: campos obligatorios, número de caracteres mínimos y máximos, caracteres solo numéricos y alfabéticos, entre otros.

```
initialCourse() {
  this.name = new FormControl({ value: null, disabled: !this.editScreen && !this.newScreen },
    [Validators.required, Validators.minLength(10), Validators.maxLength(200)]);
  this.description = new FormControl({ value: null, disabled: !this.editScreen && !this.newScreen },
    [Validators.maxLength(100000)]);
  this.requirements = new FormControl({ value: null, disabled: !this.editScreen && !this.newScreen },
    [Validators.required, Validators.minLength(50), Validators.maxLength(100000)]);
  this.branchTeaching = new FormControl({ value: null, disabled: !this.editScreen && !this.newScreen },
    [Validators.required]);
  this.studyType = new FormControl({ value: null, disabled: !this.editScreen && !this.newScreen },
    [Validators.required]);
  this.modality = new FormControl({ value: null, disabled: !this.editScreen && !this.newScreen },
    [Validators.required]);
  this.province = new FormControl({ value: null, disabled: !this.editScreen && !this.newScreen },
    [Validators.required]);
  this.duration1 = new FormControl({ value: null, disabled: !this.editScreen && !this.newScreen },
    [Validators.pattern(/^\d+$/)]);
  this.duration2 = new FormControl({ value: null, disabled: !this.editScreen && !this.newScreen }, []);
  this.language = new FormControl({ value: null, disabled: !this.editScreen && !this.newScreen },
    [Validators.required, Validators.pattern(/^[A-Za-zÑáéíóúÁÉÍÓÚ\,\s]*$/)]);
  this.price = new FormControl({ value: null, disabled: !this.editScreen && !this.newScreen },
    [Validators.required, Validators.pattern(/^\d+(\,\d{1,2})?$/)]);
  this.internship = new FormControl({ value: null, disabled: !this.editScreen && !this.newScreen }, []);
  this.credits = new FormControl({ value: null, disabled: !this.editScreen && !this.newScreen },
    [Validators.required]);
  this.typeTitle = new FormControl({ value: null, disabled: !this.editScreen && !this.newScreen },
    [Validators.required]);
  this.initialMonth = new FormControl({ value: null, disabled: !this.editScreen && !this.newScreen },
    [Validators.pattern(/^[A-Za-zÑáéíóúÁÉÍÓÚ\,]*$/)]);
  this.url = new FormControl({ value: null, disabled: !this.editScreen && !this.newScreen },
    [Validators.required]);
}
```

- En el HTML del componente UniversityCourseForm, se puede observar la utilización de la librería css Tailwind, que ofrece clases especializadas para la construcción de pantallas responsive en diferentes dispositivos.

```

  <div class="p-4 bg-beige-custom sm:text-sm">
    <xng-breadcrumb></xng-breadcrumb>
  </div>
<div class="bg-beige-custom sm:px-8 md:px-10 lg:px-12 md:min-h-screen/70 flex-col justify-center items-center">
  <h1 class="font-medium text-dark-blue-custom sm:text-lg sm:pb-4 md:text-xl md:pb-6 lg:text-2xl lg:pb-8">...
</h1>
  <div class="pb-4">
    <mat-card class="w-83vw flex justify-center rounded-full">
      <div>
        <h2 class="font-medium">{{institutionalName}}</h2>
        <hr />
        <br>
        <div>
          <form [formGroup]="universityCourseForm" class="grid items-center sm:grid-cols-1 gap-4 md:grid-cols-2"
            (ngSubmit)="save()">
            <mat-form-field appearance="outline" class="md:col-span-2">...
          </mat-form-field>
            <mat-form-field appearance="outline" class="md:col-span-2">...
          </mat-form-field>
            <div class="md:col-span-2">...
          </div>
            <div class="md:col-span-2 flex justify-center">...
          </div>
            <mat-form-field appearance="outline" class="md:col-span-2">...
          </mat-form-field>
        </div>
      </div>
    </mat-card>
  </div>
</div>

```

Anexo 3. Librerías/Código externo utilizado

A continuación, se detallan las librerías y código externo utilizados en la aplicación:

- Angular material

<https://material.angular.io/>

Angular material es un módulo construido por Angular que nos permite trabajar con componentes interactivos modernos con un estilo propio. Algunos de los componentes utilizados son expansión panel, sidenav, card, table, entre otros. Adicionalmente, se agregó spinner de material para mejorar la experiencia de usuario en cuanto a tiempos de respuesta de base de datos o AWS-S3. Un ejemplo claro es cuando se actualiza un curso.

- Tailwind

<https://tailwindcss.com/>

Tailwind es una librería para CSS, de bajo nivel, permitiendo el desarrollo web de una manera ágil y óptima. Se caracteriza por ser un framework utility-first, es decir, se basa en clases de utilidad como primera opción. Esto facilitará el diseño responsive.

- Fontawesome

<https://fontawesome.com/>

Fontawesome es una librería que proporciona multitud de iconos, y es por ello que se utilizaron los iconos de redes sociales para el footer. Tales son facebook, youtube y twitter.

- Bootstrap

<https://getbootstrap.com/>

Bootstrap es una librería css, la cual se usó en este proyecto puntualmente para el header, el footer, la caja de búsqueda y algunos links, haciéndolos un poco más dinámicos.

Algunas dependencias importantes, las cuales cuentan con licencia MIT de uso libre, son:

- Componente subida de archivos

<https://pivan.github.io/file-upload/>

Se utiliza el componente ngx-file-upload debido a su interfaz gráfica y facilidad de reutilización.

- JWT-simple

<https://www.npmjs.com/package/jwt-simple>

Se utiliza JWT-simple, que permite realizar encode y decode de manera sencilla, fue utilizado tanto para la autenticación y autorización.

- Promise-mysql

<https://www.npmjs.com/package/promise-mysql>

Es una dependencia de gran utilidad que básicamente lo que hace es envolver las llamadas de las funciones con promesas. Podemos ver su utilización en los controladores en el backend.

Anexo 4. Bibliografía

Amazon. (2022). AWS Amazon. Obtenido de AWS Amazon: <https://aws.amazon.com/es/free/>

Amazon Elastic Compute Cloud - Amazon EC2. (2022). Obtenido de Amazon Elastic Compute Cloud - Amazon EC2: <https://aws.amazon.com/es/ec2>

Amazon S3. (2022). Obtenido de Amazon S3: <https://aws.amazon.com/es/s3/>

Amazon Web Services - Amazon Relation Database Service(RDS). (2022). Obtenido de Amazon Web Services - Amazon Relation Database Service(RDS): <https://aws.amazon.com/es/rds/>

Developers google. (2021). Obtenido de <https://developers.google.com/web/fundamentals/design-and-ux/responsive/patterns?hl=es>

ExpressJS. (2021). Obtenido de <https://expressjs.com/es>

JSON. (2021). Obtenido de <https://www.json.org/json-es.html>.

JWT. (2021). Obtenido de <https://jwt.io/>

MacDonald, D. (2019). Practical UI Patterns for Design Systems: Fast-Track Interaction Design for a Seamless User Experience. New York: Apress.

Mba Executive - Cámara de Oviedo. (2020). Obtenido de <https://www.mba-asturias.com/empresas/viabilidad-economica-proyecto-empresarial/>

Ministerio de Universidades. (2020-2021). Datos y cifras del Sistema Universitario Español. Madrid.

Nodemon. (2021). Obtenido de <https://www.npmjs.com/package/nodemon>

PM2. (2022). Obtenido de <https://pm2.keymetrics.io/>: <https://pm2.keymetrics.io/>

Royce, W. (1998). Management, Software Project. New York: Addison-Wesley.

Russ Unger, C. C. (2009). A Project Guide to UX Design: For User Experience Designers in the Field or in the Making. USA: PEACHPIT.

Ui-patterns. (2021). Obtenido de <http://ui-patterns.com/>