

---

Wishten: Aplicación web para el acercamiento  
con el mundo laboral y el aprendizaje a través  
de vídeos

Wishten: Web application for managing labor  
market and learning using videos

---



Trabajo de Fin de Grado  
Curso 2022–2023

**Autores**

Roberto Tejedor Moreno  
Enrique Martín Rodríguez

**Director**

Javier Bravo Agapito

Grado en Ingeniería Informática  
Facultad de Informática  
Universidad Complutense de Madrid



Wishten: Aplicación web para el  
acercamiento con el mundo laboral y el  
aprendizaje a través de vídeos  
Wishten: Web application for managing  
labor market and learning using videos

**Trabajo de Fin de Grado en Ingeniería Informática**

**Autores**

**Roberto Tejedor Moreno  
Enrique Martín Rodríguez**

**Director**

**Javier Bravo Agapito**

**Convocatoria:** *Septiembre 2023*

**Grado en Ingeniería Informática  
Facultad de Informática  
Universidad Complutense de Madrid**

**28 de septiembre de 2023**



# Agradecimientos

Agradecimientos al profesor y director del presente Trabajo de Fin de Grado, D. Javier Bravo Agapito, por su disposición a ser el director del trabajo, por su disponibilidad total a la hora de concretar tutorías y por la ayuda y ejemplos proporcionados a lo largo de todo el desarrollo del proyecto.

Agradecimientos también a todas aquellas personas que participaron y que invirtieron su tiempo para realizar las encuestas y pruebas llevadas a cabo durante el desarrollo del proyecto.



# Resumen

## Wishten: Aplicación web para el acercamiento con el mundo laboral y el aprendizaje a través de vídeos

El presente trabajo tiene como objetivo proporcionar a los estudiantes un complemento de estudio basado en la visualización de vídeos que podrán incluir cuestionarios. El uso de cuestionarios dentro de los vídeos tiene como finalidad ayudar a retener mejor la información aprendida durante la visualización del vídeo. Para este fin, se ha desarrollado una aplicación web llamada Wishten en la que los usuarios pueden subir vídeos y añadir cuestionarios dentro de los mismos.

Wishten también tiene como objetivo ofrecer una alternativa a la hora de buscar prácticas y ofertas de empleo a todos aquellos estudiantes que las están buscando. Para ello, se les ofrecerá un portal de búsqueda de ofertas intuitivo y fácil de utilizar, integrado en la propia aplicación.

Para poder desarrollar la aplicación, se ha utilizado el *framework* de Laravel y una arquitectura basada en el modelo cliente-servidor. Esta aplicación se ha implementado desde cero por no existir otras similares.

La aplicación se ha desarrollado buscando ser lo más intuitiva y fácil de utilizar para los usuarios. Para comprobarlo, se han realizado diversas pruebas a varios usuarios de las cuales se han obtenido resultados positivos.

## Palabras clave

Aplicación web, aprendizaje basado en vídeos interactivos, portal de empleo y prácticas, Laravel



# Abstract

## **Wishten: Web application for managing labor market and learning using videos**

The present work aims to provide students with a study supplement based on the viewing of videos that may include quizzes. The use of quizzes within the videos is intended to help better retain the information learned during video viewing. To this end, a web application called Wishten has been developed where users can upload videos and add quizzes within the videos.

Wishten also aims to offer an alternative when looking for internships and job offers to all those students who are looking for them. To this end, they will be offered an intuitive and easy-to-use job search portal, integrated into the application itself.

In order to develop the application, the Laravel framework and an architecture based on the client-server model have been used. This application has been implemented from scratch because there are no other similar ones.

The application has been developed looking to be as intuitive and easy to use for users. To prove it, several tests have been carried out to several users and positive results have been obtained.

## **Keywords**

Web application, interactive video-based learning, job and internship portal, Laravel



# Índice

<b>1. Introducción</b>	<b>1</b>
1.1. Motivación . . . . .	1
1.2. Objetivos . . . . .	2
1.3. Planteamiento del problema . . . . .	3
1.4. Plan de trabajo . . . . .	4
1.5. Organización del documento . . . . .	5
<b>Introduction</b>	<b>7</b>
<b>2. Trabajos relacionados</b>	<b>13</b>
2.1. Aprendizaje basado en vídeos . . . . .	13
2.2. Prácticas de empresa . . . . .	14
<b>3. Estudio previo</b>	<b>15</b>
3.1. Análisis de método de estudio . . . . .	15
3.1.1. Diseño del cuestionario . . . . .	16
3.1.2. Análisis descriptivo de la muestra . . . . .	16
3.1.3. Análisis de resultados . . . . .	22
3.2. Análisis de búsqueda de prácticas . . . . .	24
3.2.1. Diseño del cuestionario . . . . .	24
3.2.2. Análisis descriptivo de la muestra . . . . .	25
3.2.3. Análisis de resultados . . . . .	31
<b>4. Desarrollo</b>	<b>33</b>
4.1. Requisitos . . . . .	33
4.1.1. Requisitos funcionales . . . . .	33

4.1.2.	Requisitos no funcionales . . . . .	34
4.2.	Arquitectura . . . . .	35
4.3.	Base de datos . . . . .	36
4.3.1.	Diagrama entidad-relación . . . . .	36
4.3.2.	Tablas . . . . .	36
4.3.3.	Relaciones . . . . .	42
4.4.	Prototipo . . . . .	44
4.4.1.	Página principal . . . . .	44
4.4.2.	Subir un vídeo . . . . .	45
4.4.3.	Añadir un cuestionario . . . . .	46
4.4.4.	Vista de un vídeo . . . . .	46
4.4.5.	Perfil de un usuario . . . . .	47
4.4.6.	Vista de ofertas . . . . .	48
4.4.7.	Añadir una oferta . . . . .	48
4.4.8.	Ofertas de una empresa . . . . .	49
4.4.9.	Administración de usuarios . . . . .	50
4.4.10.	Administración de vídeos . . . . .	50
4.4.11.	Administración de ofertas . . . . .	51
4.4.12.	Administración de mensajes . . . . .	51
4.5.	Análisis de tecnologías . . . . .	52
4.5.1.	Framework . . . . .	52
4.5.2.	Gestión de la base de datos . . . . .	53
4.5.3.	Servidor . . . . .	54
4.5.4.	Sistema de control de versiones . . . . .	54
4.5.5.	Editor de código . . . . .	55
4.6.	Implementación . . . . .	56
4.6.1.	Diseño . . . . .	56
4.6.2.	Lógica . . . . .	63
4.6.3.	Funcionalidades . . . . .	65
4.7.	Casos de uso . . . . .	82
4.7.1.	Actores de la aplicación . . . . .	82
4.7.2.	Diagramas de casos de uso . . . . .	83
4.8.	Diagramas de flujo . . . . .	85
4.8.1.	Inicio de sesión y registro . . . . .	85

4.8.2.	Subir un vídeo . . . . .	85
4.8.3.	Editar un vídeo . . . . .	86
4.8.4.	Visualizar un vídeo . . . . .	86
4.8.5.	Subir una oferta . . . . .	86
4.8.6.	Editar una oferta . . . . .	87
4.8.7.	Visualizar una oferta . . . . .	87
4.9.	Pruebas . . . . .	88
<b>5.</b>	<b>Resultados</b>	<b>91</b>
<b>6.</b>	<b>Evaluación</b>	<b>93</b>
<b>7.</b>	<b>Conclusiones</b>	<b>95</b>
7.1.	Conclusiones . . . . .	95
7.2.	Trabajo futuro . . . . .	97
	<b>Conclusions</b>	<b>99</b>
	<b>Contribuciones Personales</b>	<b>103</b>
	<b>Bibliografía</b>	<b>107</b>
<b>A.</b>	<b>Cuestionarios de estudio previo</b>	<b>109</b>
A.1.	Cuestionario sobre método de estudio . . . . .	109
A.2.	Cuestionario sobre búsqueda de prácticas . . . . .	112



# Índice de figuras

1.1. Diagrama de Gantt . . . . .	5
1.2. Gantt chart . . . . .	11
4.1. Arquitectura de la aplicación . . . . .	35
4.2. Diagrama entidad-relación . . . . .	36
4.3. Relaciones entre las tablas . . . . .	43
4.4. Boceto de la vista <i>home</i> . . . . .	45
4.5. Boceto de la vista <i>uploadVideo</i> . . . . .	45
4.6. Boceto de la vista <i>addQuestion</i> . . . . .	46
4.7. Boceto de la vista <i>video</i> . . . . .	47
4.8. Boceto de la vista <i>profile</i> . . . . .	47
4.9. Boceto de la vista <i>offers</i> . . . . .	48
4.10. Boceto de la vista <i>uploadOffer</i> . . . . .	49
4.11. Boceto de la vista <i>yourOffers</i> . . . . .	49
4.12. Boceto de la vista <i>adminUsers</i> . . . . .	50
4.13. Boceto de la vista <i>adminVideos</i> . . . . .	50
4.14. Boceto de la vista <i>adminOffers</i> . . . . .	51
4.15. Boceto de la vista <i>adminMessages</i> . . . . .	51
4.16. Primer logotipo de la aplicación . . . . .	57
4.17. Logotipo definitivo de la aplicación . . . . .	57
4.18. Colores principales de la aplicación . . . . .	58
4.19. Carga del fichero de Font Awesome en el código de la aplicación . . . . .	59
4.20. Código HTML proporcionado por Font Awesome para incrustar el icono en las vistas . . . . .	60
4.21. Ejemplo de uso de <i>@media</i> en el CSS de la aplicación . . . . .	62
4.22. Arquitectura de una aplicación en Laravel . . . . .	63

4.23. Vista <i>login</i> . . . . .	66
4.24. Vista <i>registration</i> . . . . .	66
4.25. Vista <i>profile</i> . . . . .	67
4.26. Vista <i>follows</i> . . . . .	67
4.27. Vista <i>adminUsers</i> . . . . .	68
4.28. Vista <i>newVideo</i> . . . . .	69
4.29. Estructura de almacenamiento de los vídeos . . . . .	70
4.30. Vista <i>home</i> . . . . .	70
4.31. Vista <i>videoWatch</i> . . . . .	71
4.32. Vista <i>videoEdit</i> . . . . .	71
4.33. Vista <i>adminVideos</i> . . . . .	72
4.34. Vista <i>videoWatch</i> de un vídeo bloqueado . . . . .	72
4.35. Vista <i>addQuiz</i> . . . . .	73
4.36. Vista <i>videoWatch</i> - Respondiendo una pregunta . . . . .	74
4.37. Vista <i>videoStats</i> . . . . .	74
4.38. Vista <i>newOffer</i> . . . . .	75
4.39. Vista <i>adminOffers</i> . . . . .	75
4.40. Vista <i>offerEdit</i> . . . . .	76
4.41. Vista <i>offerHome</i> desde usuario con rol <i>company</i> o <i>admin</i> . . . . .	76
4.42. Vista <i>offerHome</i> desde usuario con rol <i>standard</i> . . . . .	77
4.43. Estructura de almacenamiento de los documentos asociados a las ofertas . . . . .	78
4.44. Vista <i>offerHome</i> cuando un usuario hace clic en <i>View</i> para ver la descripción de la oferta . . . . .	79
4.45. Vista <i>homeGuest</i> . . . . .	79
4.46. Vista <i>chat</i> - Conversación sin mensajes . . . . .	80
4.47. Vista <i>chat</i> - Conversación con mensajes . . . . .	80
4.48. Vista <i>myChats</i> . . . . .	81
4.49. Vista <i>chatList</i> . . . . .	82
4.50. Vista <i>adminMessages</i> . . . . .	82
4.51. Diagrama de casos de uso - Estudiante . . . . .	83
4.52. Diagrama de casos de uso - Empresa . . . . .	84
4.53. Diagrama de casos de uso - Administrador . . . . .	84
4.54. Diagrama de flujo - <i>Login</i> y registro . . . . .	85
4.55. Diagrama de flujo - Subir un vídeo . . . . .	85

4.56. Diagrama de flujo - Editar un vídeo . . . . .	86
4.57. Diagrama de flujo - Visualizar un vídeo . . . . .	86
4.58. Diagrama de flujo - Subir una oferta . . . . .	87
4.59. Diagrama de flujo - Editar una oferta . . . . .	87
4.60. Diagrama de flujo - Visualizar una oferta . . . . .	87
A.1. Cuestionario sobre método de estudio - Edad de los encuestados . . .	109
A.2. Cuestionario sobre método de estudio - Sexo de los encuestados . . .	109
A.3. Cuestionario sobre método de estudio - Estudios de los encuestados .	110
A.4. Cuestionario sobre método de estudio - Horas de estudio de los en-	
cuestados . . . . .	110
A.5. Cuestionario sobre método de estudio - Método de estudio de los	
encuestados . . . . .	111
A.6. Cuestionario sobre método de estudio - Reflexión sobre el método	
utilizado . . . . .	111
A.7. Cuestionario sobre método de estudio - Perjuicio del método utilizado	112
A.8. Cuestionario sobre método de estudio - Repetición del método utilizado	112
A.9. Cuestionario sobre búsqueda de prácticas - Realización o búsqueda	
de prácticas . . . . .	113
A.10. Cuestionario sobre búsqueda de prácticas - Pregunta sobre portal de	
búsqueda . . . . .	113
A.11. Cuestionario sobre búsqueda de prácticas - Respuestas sobre portal	
de búsqueda . . . . .	113
A.12. Cuestionario sobre búsqueda de prácticas - Portal de búsqueda pro-	
porcionado por el centro de estudios . . . . .	114
A.13. Cuestionario sobre búsqueda de prácticas - Utilidad del portal pro-	
porcionado por el centro de estudios . . . . .	114



# Índice de tablas

4.1. Tabla <i>users</i> . . . . .	37
4.2. Tabla <i>follows</i> . . . . .	38
4.3. Tabla <i>videos</i> . . . . .	38
4.4. Tabla <i>subjects</i> . . . . .	39
4.5. Tabla <i>visualized_videos</i> . . . . .	39
4.6. Tabla <i>questions</i> . . . . .	40
4.7. Tabla <i>answers</i> . . . . .	40
4.8. Tabla <i>user_answer</i> . . . . .	41
4.9. Tabla <i>offers</i> . . . . .	41
4.10. Tabla <i>conversations</i> . . . . .	42
4.11. Tabla <i>messages</i> . . . . .	42



# Introducción

## 1.1. Motivación

Hoy en día hay numerosas carreras en las que los estudiantes se centran en resolver ejercicios similares a los que previamente han aparecido en los exámenes. Esto ha fomentado la no retención de los conocimientos a largo plazo, lo que en ocasiones supone una mayor dificultad en aquellas asignaturas que dan por sentado dichos conocimientos. Esto puede deberse a que en diversas carreras universitarias, como Ingeniería Informática, el temario es muy denso para la cantidad de horas dedicadas a la asignatura, provocando, en numerosas ocasiones, que se estudie solo lo estrictamente necesario, ya que la cantidad de ejercicios prácticos que se pueden evaluar son limitados.

Diversos estudios (Rodríguez & Platas-García, 2022; Liao & Wu, 2023) muestran una mejora en el rendimiento académico de los estudiantes que han complementado su estudio con la visualización de vídeos. Sin embargo, para que los vídeos sean más eficaces, se ha comprobado que la utilización de elementos adicionales tales como anotaciones o preguntas mejoran aún más la retención de la información explicada en el vídeo (Moos & Bonde, 2016).

A fin de intentar reducir el número de estudiantes que no priorizan la asimilación de conocimientos, el presente trabajo busca ofrecerles una plataforma de ayuda para poder retener los conceptos clave. Para ello, esta plataforma utilizará cuestionarios incluidos en los vídeos que ayudarán al estudiante a priorizar la asimilación de los conocimientos.

Por otro lado, otra de las situaciones a la que están sometidos los estudiantes universitarios y que resulta interesante de estudiar es la de búsqueda de prácticas en empresas. Hoy en día muchos estudiantes universitarios no han sido capaces de encontrar un lugar adecuado para buscarlas o no se les ha proporcionado la ayuda suficiente cómo para encontrarlo a pesar de contar con numerosas plataformas o lugares adecuados para buscarlas.

Algunos estudios, como el de Alsolais (2022), desarrollan portales web orientados

a la búsqueda de prácticas de empresa, buscando facilitar la comunicación entre estudiantes, profesores y empresas y ayudando a que los estudiantes puedan contar con dichas experiencias previas a graduarse.

Bajo dichas circunstancias y con el fin de intentar disminuir el número de estudiantes que no son capaces de encontrar un lugar adecuado para solicitar prácticas de empresa, el presente trabajo busca también proporcionarles un portal interactivo que sirva cómo puente entre las empresas y los estudiantes.

## 1.2. Objetivos

El principal objetivo de este trabajo es proporcionar una ayuda a los estudiantes universitarios para priorizar la retención de conocimientos. Para ello se utilizarán vídeos creados por los propios usuarios.

Este objetivo principal está dividido en los siguientes objetivos específicos:

- Crear una aplicación web que sirva para facilitar la retención de conocimientos en los estudiantes universitarios.
  
- Evaluar la utilidad del uso de cuestionarios dentro de los vídeos.
  
- Proponer una alternativa más enfocada al aprendizaje de temas concretos. A pesar de la existencia de diversas plataformas orientadas a la subida y visualización de vídeos (como puede ser YouTube), no resulta fácil en dichas plataformas encontrar diversos temas de estudio concretos. Por ello, este trabajo busca amenizar y facilitar las búsquedas, con la finalidad de que no se pierda tiempo buscando y visualizando vídeos que no sean acordes al tema buscado.

Otro objetivo a destacar es facilitar la conexión empresa-estudiante a través de la publicación de ofertas, permitiendo al usuario contactar directamente con las empresas solicitantes.

Para conseguir este objetivo, se propone integrar en la mencionada aplicación web un portal que tenga como finalidad facilitar la interacción empresa-estudiante. La idea principal es que los estudiantes reciban confirmaciones o denegaciones lo antes posible para así evitar que tengan que aplicar a un gran número de ofertas diferentes a la vez. Dicha comunicación se realizará a través del chat de dicho portal, siendo esta una conversación privada entre la empresa y el estudiante.

También se busca, mediante este portal, que sea fácil e intuitivo encontrar prácticas acordes a lo que buscan los estudiantes. De esta manera, el portal estaría orientado sobre todo a aquellos estudiantes que desconocen o no se les ha facilitado un lugar donde poder buscar prácticas.

### 1.3. Planteamiento del problema

Se ha observado que la forma de estudio principal de un gran número de estudiantes universitarios no es la más efectiva para retener los conocimientos. Esto se debe a que en numerosas ocasiones, dicha forma de estudio se centra en la resolución de ejercicios de examen y no en los conceptos que hay detrás. Existen estudios como el de Gargallo et al. (2012) que analizan los diferentes enfoques de aprendizaje, resaltando que la mayoría de los estudiantes realizan un aprendizaje superficial. Dicho aprendizaje superficial se basa en una motivación extrínseca del estudiante que busca únicamente aprobar.

La finalidad de dicho método es aprobar el examen. Sin embargo, de esta manera, algunos de los conceptos clave no son asimilados correctamente, perjudicando a futuro al estudiante. En más de una ocasión se ha podido observar cómo existen a lo largo de la carrera y en el ámbito laboral, numerosos conceptos que requieren conocimientos previos.

Los autores de este trabajo han realizado cuestionarios dirigidos a diversos estudiantes universitarios y han podido comprobar la existencia de dicho problema. Los resultados de este estudio se muestran en la sección 3.1.

Para abordar el problema comentado, se realizó una investigación sobre diversos artículos reputados relacionados con dicho problema. Los artículos seleccionados son aquellos que deciden solucionar dicho problema mediante la utilización de vídeos educativos y, en ocasiones, interactivos.

Por ejemplo, se ha estudiado el uso de videotutoriales para la mejora del aprendizaje en estudiantes universitarios (Rodríguez & Platas-García, 2022). Los resultados obtenidos por Rodríguez & Platas-García (2022) en este estudio apuntan a que la utilización de tecnologías y, en concreto, de videotutoriales contribuye significativamente a la motivación de los estudiantes para aprender. Rodríguez & Platas-García (2022) indican que esto se debe al formato del contenido, más atractivo para los estudiantes, además de que los videotutoriales cubren un tema concreto y son fácilmente accesibles.

Por otra parte, se han analizado patrones de comportamiento durante la visualización de vídeos con el fin de comprobar su recepción entre los estudiantes. En este sentido, Liao & Wu (2023) analizaron la interacción de varios estudiantes universitarios con vídeos de aprendizaje, comprobando un aumento en su motivación, lo que a su vez incrementaba su rendimiento.

En adición a esto, se ha estudiado el uso de elementos adicionales durante la visualización del vídeo. Por ejemplo, se examinó el proceso de aprendizaje de varios estudiantes en un modelo de clase invertida que utilizaba vídeos (Moos & Bonde, 2016). De este modo, a una parte de los estudiantes se les presentó un vídeo que integraba indicaciones para el aprendizaje autorregulado, mientras que el resto de estudiantes visualizó el vídeo sin componentes adicionales. Moos & Bonde (2016) observaron que los estudiantes que recibieron indicaciones integradas en el vídeo tuvieron un mejor desempeño en el test que realizaron posteriormente.

Se ha detectado que los estudiantes universitarios suelen encontrar dificultad a la hora de comunicarse con empresas para la realización de prácticas durante su etapa de estudios. Esto ha sido comprobado por los autores de este trabajo a partir de cuestionarios realizados a varios estudiantes universitarios. Los resultados de dicho análisis se encuentran en la sección 3.2.

La realización de prácticas en empresas tiene un papel muy importante en el aprendizaje de un estudiante universitario. Binder et al. (2015) realizan un estudio para dar a conocer la importancia y beneficios de la realización de prácticas en empresas y cómo estas ayudan al desarrollo y aprendizaje del estudiante.

El presente trabajo tiene como objetivo solventar los problemas mencionados mediante el desarrollo de una aplicación web que permitirá a los usuarios crear y visualizar vídeos educativos, que podrán integrar cuestionarios para verificar el correcto aprendizaje del tema expuesto. Dichos vídeos estarán clasificados por temas concretos para facilitar su búsqueda y se mostrarán en primer lugar los que demuestren una mayor utilidad para los usuarios. Adicionalmente, se incluirá en la web un portal de ofertas que pondrá en contacto a empresas y estudiantes para negociar acuerdos de prácticas o trabajo.

## 1.4. Plan de trabajo

A fin de alcanzar los objetivos descritos previamente, se ha seguido la siguiente planificación:

- Realización de un estudio previo con el fin de comprobar el método de estudio de los estudiantes universitarios y su experiencia en la búsqueda de prácticas. Este estudio se llevó a cabo durante las dos últimas semanas de enero, tras la realización de los exámenes de la convocatoria ordinaria.
- Investigación sobre las posibles soluciones para el problema planteado, así como de trabajos que planteen un problema similar. Dicha investigación se llevó a cabo durante las dos últimas semanas de enero y la primera de febrero, ya que era necesario tener claro qué temas abordar con el desarrollo posterior.
- Definición de la base de datos, componiendo las tablas junto con sus atributos y todas las relaciones entre ellas. Esta tarea se llevó a cabo durante las dos últimas semanas de febrero.
- Diseño del prototipo de la aplicación a través de bocetos de las vistas. Esta actividad fue realizada durante la tercera semana de febrero.
- Búsqueda de herramientas tecnológicas que puedan ser de utilidad. Para ello se empleó las dos últimas semanas de febrero y la primera de marzo, ya que había que tener claro que herramientas utilizar para empezar a desarrollar la aplicación.

- Desarrollo de la aplicación web. El desarrollo se inició a mediados de marzo y se mantuvo hasta mediados de agosto. En los meses de mayo y junio se redujo el tiempo dedicado al desarrollo de la aplicación debido a los exámenes.
- Realización de pruebas con usuarios y análisis de los resultados obtenidos. Esta tarea fue realizada durante las dos últimas semanas de agosto. Para realizar esta tarea se necesitaba que la aplicación web estuviera finalizada para poder ofrecerles a los usuarios una versión completa.

En la figura 1.1 se muestra el diagrama de Gantt referente a las actividades realizadas durante el desarrollo del proyecto.

Tareas realizadas	Tiempo de duración																															
	enero				febrero				marzo				abril				mayo				junio				julio				agosto			
Semanas	1ª	2ª	3ª	4ª	1ª	2ª	3ª	4ª	1ª	2ª	3ª	4ª	1ª	2ª	3ª	4ª	1ª	2ª	3ª	4ª	1ª	2ª	3ª	4ª	1ª	2ª	3ª	4ª	1ª	2ª	3ª	4ª
Realización de estudio previo																																
Investigación de trabajos relacionados																																
Definición de la base de datos																																
Prototipo																																
Búsqueda de herramientas tecnológicas																																
Desarrollo de la aplicación																																
Pruebas con usuarios																																

Figura 1.1: Diagrama de Gantt

## 1.5. Organización del documento

En el capítulo 2 se exponen los trabajos previos relacionados con los objetivos propuestos. En este sentido, se llevó a cabo una investigación con el fin de obtener una mejor comprensión de los problemas que se pretenden abordar.

En el capítulo 3 se explica el estudio previo llevado a cabo mediante cuestionarios realizados a varios estudiantes, así como el análisis de sus resultados. Estos cuestionarios tenían como finalidad entender el método de estudio que siguen estos estudiantes y su impacto, así como su experiencia en la búsqueda de prácticas.

El capítulo 4 está dedicado a explicar en detalle todas las fases de desarrollo de la aplicación web, desde la especificación de los requisitos hasta la implementación y las pruebas, pasando por la estructuración de la base de datos, la elaboración de bocetos previos y el análisis de las tecnologías para desarrollarla.

En el capítulo 5 se muestran los resultados de las pruebas realizadas durante el desarrollo de la aplicación y se analizan dichos resultados para comprobar principalmente la usabilidad de la aplicación.

En el capítulo 6 se encuentra el diseño de la evaluación que se pretende realizar en un futuro para comprobar la utilidad de añadir cuestionarios dentro de los vídeos y sus beneficios sobre el proceso de aprendizaje de los estudiantes.

En el capítulo 7 se exponen las conclusiones a las que se ha llegado tras la realización de este proyecto. En él se indica en qué grado se han cumplido los objetivos propuestos y las dificultades encontradas durante el proceso de desarrollo y cómo éstas se han solventado. Además, en este capítulo también se explica el trabajo que se pretende desempeñar en el futuro, continuando con el desarrollo de la aplicación web y ampliando sus características y funcionalidades.

Después, se exponen las contribuciones personales de cada uno de los autores del proyecto, detallando la investigación y desarrollo que aportó cada uno al proyecto.

Por último, se muestra la bibliografía, en la que se aparecen todas las referencias consultadas para desarrollar el proyecto.

# Introduction

## Motivation

Nowadays, there are numerous university degrees in which students focus mainly in solving similar exercises to the ones that have previously appeared in exams. This has led to a lack of long-term knowledge retention, which sometimes leads to a greater difficulty in those subjects that take for granted such knowledge. This may be caused because in several degrees, such as Computer Engineering, the syllabus is very dense for the number of hours dedicated to the subject, often resulting in students only studying what is strictly necessary because the amount of practical exercises that can be evaluated is limited.

Various studies (Rodríguez & Platas-García, 2022; Liao & Wu, 2023) show an improvement in the academic performance of the students who have supplemented their study with the visualization of videos. However, in order to make videos more effective, it has been found that using additional elements, such as annotations or questions, improve even more the retention of the information explained in the video (Moos & Bonde, 2016).

With the purpose of reducing the number of students that do not prioritize the knowledge assimilation, this project aims to offer them a platform that helps them to retain the key concepts. In order to achieve this, this platform will use quizzes integrated in videos that will help the student to prioritize the knowledge assimilation.

On the other hand, another situation that university students face and is interesting to study, is the search for internship in companies. Nowadays, several university students have not been able to find a suitable place to search them or they have not been provided with enough help in order to find it, despite the existence of numerous appropriate platforms for searching.

Some studies, such as the one conducted by Alsolais (2022), develop web portals with the aim of searching internships in companies, attempting to facilitate the communication between students, teachers and companies, and helping students to gain such experience before concluding their degree.

Under these circumstances and attempting to reduce the number of students

that are not able to find a suitable place to apply for internships in companies, the present project aims to also provide them with an interactive portal that serves as a bridge between companies and students.

## Goals

The main goal of this project is to provide university students support to prioritize the knowledge retention. In order to achieve this, videos created by users themselves will be used.

This goal is divided in the following specific objectives:

- Creating a web application that serves to facilitate the knowledge retention to university students.
- Evaluating the usefulness of using quizzes embedded in videos.
- Offering an alternative more focused on learning specific topics. Despite of the existence of various platforms oriented to the uploading and visualization of videos (such as YouTube), it is not an easy task to find some specific study content. Because of this, the present work aims to make more pleasant and facilitate the searching, with the purpose of avoiding wasting time searching and viewing videos that are not in accordance with the desired topic.

Another goal to point out is to facilitate the connection between companies and students through offers publishing, allowing the user to contact directly with the companies.

With the aim of achieving this goal, this project purposes integrating in the aforementioned web application a portal to manage the interaction between companies and students. The main idea is to give students confirmations or denials as soon as possible, in order to avoid having to apply to a large number of different offers at the same time. Such communication will be performed through a chat integrated in the mentioned portal, being a private conversation between the company and the student.

This portal also aims to make it easier and intuitive finding the appropriate internships for what students are searching. Therefore, this portal would be oriented especially to those students who do not know or are not provided with a place where they can search for internships.

## Problem statement

It has been observed that the primary study method of a large number of university students is not the most effective for knowledge retention. This is due to the fact that in numerous occasions, such study method is centered in solving exam exercises

and not in the concepts behind them. There are studies, such as the one by Gargallo et al. (2012), that analyse different learning approaches, pointing out the fact that the majority of the students perform a surface learning approach. Such surface learning is based in the external motivation of the student that aims exclusively to pass the subject.

The purpose of this method is to pass the exams. However, in this way, some of the key concepts are not absorbed correctly, which is detrimental to the student in the long term. On several occasions, there exist numerous concepts that require previous knowledge throughout degrees and work environment.

The authors of this work have conducted questionnaires addressed to various university students and they have verified the existence of the aforementioned problem. The results of this study are shown in section 3.1.

In order to approach the mentioned problem, research was carried out on various renowned studies related to such problem. The selected articles are the ones that decide solve this problem through the usage of instructive and eventually interactive videos.

For instance, it has been studied the use of tutorial videos to improve learning in university students (Rodríguez & Platas-García, 2022). The results obtained by Rodríguez & Platas-García (2022) in this study point to the fact that the usage of technologies and tutorial videos in particular, contribute significantly in students' motivation to learn. Rodríguez & Platas-García (2022) indicate that this is due to the content's format, which is more attractive for students, besides the fact that tutorial videos cover a specific subject and are easily accessible.

Moreover, some behavioral patterns have been analysed during video visualization, with the attempt of verifying the video impact on students. On this matter, Liao & Wu (2023) analysed the interaction of various university students with learning videos, confirming an increase in their motivation, which also incremented their performance.

In addition to this, it has been studied the use of additional elements during a video visualization. For instance, the learning process of numerous students in a flipped classroom model using videos was examined (Moos & Bonde, 2016). In this way, part of the students were provided with a video that embedded self-regulated learning prompts, whereas the rest of the students viewed the video without additional components. Moos & Bonde (2016) observed that the students who were provided with embedded prompts in videos developed a better performance in the test taken afterwards.

It has been found that university students tend to find it difficult to get in touch with companies for internship during their studies. This has been tested by the authors of this project by some questions addressed to various university students. The results of such analysis are shown in section 3.2.

The fulfillment of internships in companies plays an important role in a university student's learning. Binder et al. (2015) conducted an study to publish the importance and benefits of internships in companies and how they support the student's

development and learning.

The purpose of the present work is to solve the aforementioned problems through the development of a web application that will allow users to create and visualize educative videos which will be able to integrate interactive quizzes in order to verify the correct learning of the explained topic. Such videos will be classified by specific subjects to facilitate their searching, and the ones that prove to be more useful will be shown in first place to users. In addition, an offers portal will be included in the web and it will put companies and students in contact with each other in order to negotiate internships and job agreements.

## Work plan

With the purpose of reaching the goals described previously, the following planning has been followed:

- A preliminary study was carried out in order to check the study method of university students and their experience in the search for internships. This study was carried out during the last two weeks of January, after the exams of the ordinary exams.
- Research on possible solutions to the problem posed, as well as on works that pose a similar problem. This research was carried out during the last two weeks of January and the first week of February, as it was necessary to be clear about the issues to be addressed in the further development.
- Definition of the database, composing the tables together with their attributes and all the relationships between them. This task was carried out during the last two weeks of February.
- Design of the prototype of the application through sketches of the views. This activity was carried out during the third week of February.
- Search for technological tools that could be useful. The last two weeks of February and the first week of March were used for this, as it was necessary to be clear which tools to use in order to start developing the application.
- Development of the web application. Development started in mid-March and continued until mid-August. In May and June the time devoted to the development of the application was reduced due to exams.
- User testing and analysis of the results obtained. This task was carried out during the last two weeks of August. In order to carry out this task, the web application needed to be finalised in order to be able to offer users a complete version.

Figure 1.2 shows the Gantt chart of the activities carried out during the development of the project.

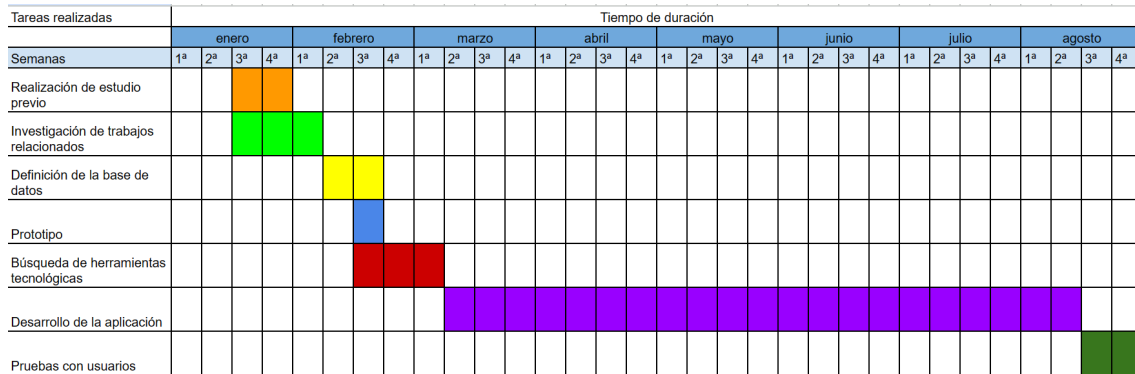


Figure 1.2: Gantt chart

## Document organization

Chapter 2 sets out the previous work related to the proposed objectives. Therefore, research was carried out in order to gain a better understanding of the problems to be addressed.

Chapter 3 explains the previous study that was carried out through questionnaires addressed to several students, as well as the analysis of their results. This questionnaires aimed to understand the study method followed by these students and its impact, as well as their experience in searching for internship.

Chapter 4 is dedicated to explaining in detail all the phases of the web application development, from the requirement specification to the implementation and testing, including the database structuring, the previous sketches elaboration and the analysis of the technologies to implement it.

Chapter 5 shows the results of the tests carried out during the application development and discusses this results in order to check mainly the application's usability.

Chapter 6 contains the assessment design that is intended to be carried out in the future to verify the usefulness of adding quizzes within videos and their benefits over students learning process.

Chapter 7 sets out the conclusions reached after this project has been carried out. It indicates the degree to which the purposed objectives have been met and the difficulties that were faces and how were they resolved. In addition, this chapter explains the future work that is intended to be carried out, continuing with the development of the web application and extending its features and functionalities.

The personal contributions of each of the project's authors are then presented, detailing the research and development that each contributed to the project.

Finally, the bibliography is shown, in which all the references consulted for the development of the project are listed.



## Trabajos relacionados

Para la realización de este trabajo se ha realizado una investigación sobre diversos estudios y proyectos relacionados con los temas planteados previamente.

Para comprender el método de estudio de los estudiantes, Gargallo et al. (2012) analizaron los diversos enfoques de aprendizaje que adoptan los estudiantes, llegando a la conclusión de que la mayoría adopta un enfoque muy superficial que prioriza los resultados sobre los conocimientos obtenidos. Por otro lado, se pudo observar que los estudiantes más destacados adoptaban un enfoque más profundo, entendiendo por profundo que la motivación se basa en el interés del estudiante y no en el resultado del examen.

Dada la existencia de este problema, se decidió buscar un método de aprendizaje que pudiera ayudarlos a tomar ese enfoque más profundo comentado anteriormente.

### 2.1. Aprendizaje basado en vídeos

En primer lugar se investigó sobre la utilidad de los vídeos en la enseñanza. Rodríguez & Platas-García (2022) realizaron un estudio con la finalidad de analizar la experiencia de estudiantes universitarios con el uso de videotutoriales. En dicho estudio se encuentra evidencia sobre la percepción positiva de los estudiantes debido a su facilidad en el manejo de las tecnologías, lo que hace bastante común este tipo de aprendizaje hoy en día.

Siguiendo con el uso de vídeos en la enseñanza, Liao & Wu (2023) estudiaron la huella digital dejada por los estudiantes en entornos de aprendizaje basado en vídeos, entendiendo por huella digital acciones como pausar el vídeo o rebobinarlo. Tras obtener los resultados, observaron que se produjo una mejora tanto en la motivación como en el proceso de aprendizaje de los estudiantes.

Sin embargo, otros estudios como el de Moos & Bonde (2016) apuntan a que la utilización de vídeos por sí solos no siempre asegura una mejora en el aprendizaje. Por tanto, Moos & Bonde usaron avisos incrustados en los vídeos con la finalidad de comprobar si estos producían una mejora en el aprendizaje. Tras realizar dicho

estudio, se pudo observar que aquellos estudiantes que habían contado con dichos vídeos obtuvieron mejores resultados.

Siguiendo esta línea de estudio Mirriahi et al. (2021) realizaron un estudio dividiendo a los estudiantes en tres grupos, un grupo que visualizaba un vídeo con preguntas emergentes, otro grupo que lo visualizaba sin dichas preguntas y un último grupo que utilizaba una herramienta para realizar anotaciones durante el vídeo. Al estudiar los resultados obtenidos sobre los tres grupos, se pudo observar que los que contaron con preguntas emergentes y los que realizaron anotaciones, obtuvieron mejores resultados que el otro grupo.

Haagsman et al. (2020) realizaron un estudio similar que demostró que incluir preguntas emergentes en vídeos educativos mejora el rendimiento de los estudiantes con respecto a la retención de los conocimientos.

Una vez estudiados los trabajos relacionados con el aprendizaje a través del uso de vídeos, se investigó sobre la importancia y necesidad de las prácticas de empresa, así como de las facilidades y dificultades que sus búsquedas pueden suponer en los estudiantes universitarios.

## 2.2. Prácticas de empresa

Binder et al. (2015) realizaron un estudio sobre el valor académico y personal que tienen las prácticas en el desarrollo y aprendizaje de los estudiantes, así como de la importancia de que las instituciones animen a los estudiantes a tomarlas teniendo en cuenta siempre las capacidades académicas y sociales de cada uno.

Alsolais (2022) desarrolló un portal web para facilitar la comunicación entre profesores, estudiantes y empresas relacionadas con el mundo de la medicina tomando como base la importancia de estas en el desarrollo de las habilidades y la experiencia aplicada al ámbito de la medicina. Tras realizar el estudio se pudo observar cómo el portal sirvió de puente entre los estudiantes y las empresas proporcionándoles oportunidades de cara al futuro y cómo los estudiantes se vieron beneficiados al adquirir los conocimientos aprendidos durante la realización de las prácticas.

Tras lo comentado anteriormente y teniendo en cuenta los problemas de aprendizaje de los estudiantes comentados por Gargallo et al. (2012), el presente trabajo busca ofrecer a los estudiantes una alternativa para cambiar su enfoque de aprendizaje, tomando como base las conclusiones obtenidas por los estudios relacionados con la mejora en el aprendizaje mediante la utilización de vídeos interactivos para el desarrollo de una aplicación web basada en la visualización de vídeos interactivos creados por otros usuarios de dicha aplicación. Además de contar con dicha funcionalidad, también se incluirá un portal web que permita a los estudiantes comunicarse directamente con las empresas, siendo esta idea diferente a la propuesta por Alsolais (2022) en la que el estudiante, una vez seleccionado el lugar de prácticas que le interesa, debe comunicárselo al coordinador de la universidad y este ponerse en contacto con la empresa.

## Estudio previo

El propósito del presente trabajo es mejorar el aprendizaje de los estudiantes y ayudarles en la búsqueda de prácticas en empresas, por lo que se ha realizado un estudio con el fin de comprobar estas cuestiones. Para ello, se han elaborado dos cuestionarios, que serán respondidos en su mayoría por estudiantes, ya que son el grupo de la población que tienen relación directa con el problema planteado por el presente proyecto.

Con el fin de recopilar información relevante y comprobar si los estudiantes se centraban solo en la realización de ejercicios similares a los de examen, prescindiendo el resto del temario, se decidió realizar un cuestionario para conocer la opinión de diversos estudiantes.

Por otro lado, también se decidió realizar otro cuestionario relacionado con la búsqueda de prácticas en empresas buscando que los estudiantes encuestados proporcionaran sus experiencias.

A partir de los resultados de ambos cuestionarios, se pretende realizar un análisis adoptando un enfoque cuantitativo que pueda ayudar a esclarecer la existencia de dicho problema.

En el apéndice A se muestran las preguntas que se hicieron en ambos cuestionarios, así como las respuestas proporcionadas a cada una.

### 3.1. Análisis de método de estudio

Este primer cuestionario se centra en recopilar información sobre los métodos de estudio de los estudiantes encuestados. Mediante este cuestionario se trata verificar si estos, en el momento de preparar un examen, estudian todo el temario, o si, por el contrario, solo repiten ejercicios similares a los de años anteriores, excluyendo el resto del temario.

### 3.1.1. Diseño del cuestionario

Teniendo en cuenta lo explicado anteriormente, el cuestionario consta de un total de ocho preguntas. Sin embargo, los encuestados no responden a las ocho preguntas, ya que, en función de algunas respuestas, aparecerán unas u otras preguntas.

Se debe tener en cuenta que el usuario debe responder a todas las preguntas que le aparezcan, es decir, no puede dejar ninguna de las preguntas que le aparezcan sin contestar.

Las primeras cinco preguntas del cuestionario son comunes a todos los usuarios, independientemente de las respuestas que ofrezcan, ya que son preguntas que buscan conocer un poco a los encuestados. Todas ellas son preguntas cerradas, a excepción de la tercera, que busca conocer la carrera o el grado que estudia el encuestado. Esta pregunta es abierta debido a que lo interesante es comprobar si los resultados son muy variables dependiendo de lo que cursa cada encuestado.

De estas cinco preguntas, las cuatro primeras son demográficas, ya que se considera interesante ver si la segmentación influye o no en los resultados, mientras que la última no lo es.

La última pregunta, actúa como un *screenner*, ya que, si bien todas las formas de estudio son interesantes para la investigación, el presente trabajo busca enfocarse en una en concreto para determinar si existe un problema o no con ella. Aun así, las preguntas que aparecen en función de las respuestas proporcionadas a dicha pregunta, buscan obtener información similar.

Si el usuario afirma usar dicho método de estudio, se busca obtener información mediante preguntas cerradas con la finalidad de saber si les ha resultado útil dicho método de estudio y si lo volverían a realizar.

Si por el contrario no ha usado dicho método de estudio, se buscará obtener información sobre si por lo menos se lo ha planteado o no.

Una vez el usuario haya respondido a todas las preguntas, se almacenarán sus respuestas y se podrá realizar posteriormente un estudio cuantitativo sobre los datos obtenidos.

### 3.1.2. Análisis descriptivo de la muestra

Para los dos cuestionarios realizados se llevó a cabo el mismo tipo de muestreo, por lo que lo que se explica a continuación es aplicable a ambos cuestionarios ya que es una explicación común a ambos.

El proceso de muestreo realizado ha sido intencionado, lo que significa que se han seleccionado aquellos individuos que son afectados directamente por los problemas que se quieren abordar mediante la realización de este proyecto. En este caso, dicho grupo de individuos relevantes han sido estudiantes universitarios de diferentes facultades.

Los estudiantes universitarios seleccionados para participar en el estudio han

sido identificados en función de su relación previa con los encuestados y su afinidad con el tema en cuestión (método de estudio y búsqueda de prácticas en empresas). Además, a dichos estudiantes se les pidió que compartieran la encuesta con dos o tres estudiantes de su mismo grado para así poder lograr una comprensión más completa de la situación.

Hay que tener en cuenta, que estas encuestas han sido realizadas usando la herramienta de Google Forms<sup>1</sup>, la cual es gratuita y ofrece la posibilidad de crear cuestionarios de cero y poder posteriormente ver las respuestas. Como se quería que las respuestas de los usuarios fueran anónimas, se decidió no registrar el correo. Tampoco se puede seguir las respuestas que una persona en específico ha dado a las preguntas, sino que solo se puede ver el resultado global.

Esto no supone un problema ya que lo que se busca es obtener una visión general de una muestra intencionada pero variada.

La muestra utilizada en ambos cuestionarios fue idéntica, ya que, durante el proceso de envío, se distribuyeron ambos cuestionarios conjuntamente y se solicitó a los participantes que completaran ambos. Debido a esto, las respuestas a las tres primeras preguntas de ambos cuestionarios coinciden, dado que estas preguntas corresponden a aspectos demográficos y se repiten en ambos cuestionarios.

Aunque sean las mismas preguntas, con la finalidad de mantener la estructura del documento, dichas preguntas se volverán a colocar en el análisis descriptivo y de resultados del cuestionario relacionado con las prácticas en empresas.

Antes de comenzar con el análisis descriptivo de la muestra hay que tener en cuenta que las encuestas han sido respondidas por 24 estudiantes universitarios. En el apéndice A.1 se pueden consultar los diagramas de barras y de sectores.

De la primera pregunta demográfica relacionada con la edad se observa la siguiente distribución:

1. Distribución de respuestas:

- Menor de 18: 0 personas.
- Entre 18 y 24 años: 23 personas.
- Entre 25 y 34 años: 1 persona.
- Entre 35 y 44 años: 0 personas.
- Más de 45 años: 0 personas.

2. Porcentaje de respuestas:

- Menor de 18: 0 %.
- Entre 18 y 24 años: 95.8 %.
- Entre 25 y 34 años: 4.2 %.
- Entre 35 y 44 años: 0 %.

---

<sup>1</sup><https://www.google.com/forms/about>

- Más de 45 años: 0 %

### 3. Promedio de edad:

Para calcular el promedio se utilizarán las marcas de clase y se asignarán valores numéricos a cada opción. Para la primera opción se tomará el 16 (Para emplear un número cercano a 18 y menor que sea representativo), para la segunda el 21, para la tercera el 29.5, para la cuarta 39.5 y para la quinta 50 (Para utilizar un número mayor que 45, pero cercano).

Gracias a la asignación de estos valores numéricos se puede calcular el promedio obteniendo un valor de 21.354 años, es decir aproximadamente 21 años (El promedio se ha calculado como  $((0*16) + (23*21) + (1*29.5) + (0*39.5) + (0*50)) / (0+23+1+0+0)$ ).

El valor promedio era el esperado, ya que la mayoría de las respuestas estaban entre 18 y 24 años.

### 4. Análisis de distribución:

Se observa que la mayoría de los encuestados (95.8 %) tienen entre 18 y 24 años y el encuestado restante (4.2 %) entre 25 y 34 por lo que la encuesta está predominada por gente joven.

De la segunda pregunta demográfica relacionada con el sexo se obtiene la siguiente distribución:

#### 1. Distribución de respuestas:

- Hombre: 20 personas.
- Mujer: 4 personas.

#### 2. Porcentaje de respuestas:

- Hombre: 83.3 %.
- Mujer: 16.7 %.

#### 3. Análisis de la distribución:

- Respuesta “Hombre” (83.3 %): Aquí se refleja el número de estudiantes universitarios encuestados.
- Respuesta “Mujer” (16.7 %): Representa la cantidad de estudiantes universitarias encuestadas.

La pregunta relacionada con la carrera que se estudia parece tener numerosas respuestas diferentes, sin embargo, si se observa detenidamente, se puede ver cómo en su mayoría son la misma carrera escrita de diferente forma, ya sea porque faltan tildes, porque unos la escriben con mayúsculas otros con minúscula o porque otros usan directamente la abreviatura (ADE de Administración y dirección de empresas).

Una vez se tiene esto en cuenta, se puede observar la distribución de la muestra por áreas de estudio en la pregunta relacionada con la carrera que se estudia:

## 1. Distribución de respuestas:

- Ingeniería informática: 11 personas.
- Administración y Dirección de empresas: 3 personas.
- Ingeniería Química: 3 personas.
- Doble Grado en ADE e Informática: 2 personas.
- Ingeniería de Computadores: 2 personas.
- Grado de Videojuegos: 1 persona.
- Ingeniería del Software: 1 persona.
- Fabricación de Productos Farmacéuticos: 1 persona.

## 2. Porcentaje de respuestas:

- Ingeniería informática: 45.83 %.
- Administración y Dirección de empresas: 12.50 %.
- Ingeniería Química: 12.50 %.
- Doble Grado en ADE e Informática: 8.34 %.
- Ingeniería de Computadores: 8.34 %
- Grado de Videojuegos: 4.17 %.
- Ingeniería del Software: 4.17 %.
- Fabricación de Productos Farmacéuticos: 4.17 %.

## 3. Análisis de la distribución:

Al tratarse de una pregunta abierta, sacar valores como el promedio o la mediana no va a resultar útil de cara a realizar un análisis y, aun sacando la moda y observando que Ingeniería Informática es la carrera más cursada entre nuestros encuestados tampoco se obtiene un buen análisis de la muestra.

Lo que sí se puede observar y que junto al valor de la moda es interesante que el 70.83 % de los encuestados están cursando carreras relacionadas con la informática. A su vez que el 79.16 % están cursando una ingeniería por lo que el porcentaje de estudiantes que a priori, pueden ser objetivo de nuestro proyecto es bastante elevado.

En relación con la primera pregunta ya exclusiva de este cuestionario y relacionada con distribución de las horas de estudio se obtuvieron los siguientes resultados:

## 1. Distribución de respuestas:

- Entre 1 y 3 horas al día: 12 personas.
- Entre 3 y 5 horas al día: 7 personas.
- Más de 5 horas al día: 5 personas.

## 2. Porcentaje de respuestas:

- Entre 1 y 3 horas al día: 50 %.
- Entre 3 y 5 horas al día: 29.2 %.
- Más de 5 horas al día: 20.8 %.

## 3. Promedio de horas estudiadas:

Para calcular el promedio se van a asignar valores numéricos a cada opción. Para la primera opción se va a tomar el 2 (punto medio entre 1 y 3), para la segunda el 4 (punto medio entre 3 y 5) y para la tercera el 6 (Para no utilizar el propio 5).

Gracias a la asignación de estos valores numéricos se puede calcular el promedio obteniendo un valor de 3.41 horas (El promedio se ha calculado como  $((12*2) + (7*4) + (5*6)) / (12+7+5)$ ).

## 4. Análisis de la distribución:

Se observa que la mitad (50 %) de los encuestados estudian entre 1 y 3 horas al día. Un porcentaje significativo (29.2 %) dedica entre 3 y 5 horas al día al estudio. El porcentaje restante (20.8 %) invierte más de 5 horas al día. Este análisis muestra una variedad de enfoques de estudio, pero gracias al promedio se puede observar cómo el tiempo medio está entre el rango de 1 a 3 y de 3 a 5 ya que es donde están concentradas el mayor número de respuestas.

La siguiente pregunta actúa como *screener* por lo que en función de la respuesta se mostraran unas preguntas u otras. De ella se ha obtenido la siguiente distribución:

## 1. Distribución de respuestas:

- Estudio ejercicios similares: 18 personas.
- Estudio el temario entero: 6 personas.

## 2. Porcentaje de respuestas:

- Estudio ejercicios similares: 75 %.
- Estudio el temario entero: 25 %.

## 3. Análisis de la distribución:

- Respuesta “Estudio ejercicios similares” (75 %): Aquí se refleja el número de estudiantes universitarios encuestados que se enfocan en la realización de ejercicios similares o parecidos a los que han salido en exámenes anteriores. Estos son los estudiantes objetivos de nuestra encuesta ya que representan a aquellos estudiantes que se busca ayudar mediante la realización de la aplicación de vídeos de nuestro proyecto.

- Respuesta “Estudio el temario entero” (25 %): Representa a los estudiantes universitarios encuestados que prefieren estudiar de manera general el temario. Si bien estos no son los estudiantes objetivo de nuestra encuesta, es interesante conocer si aun realizando este método de estudio, se han planteado alguna vez utilizar el otro, por ello se les realizará una pregunta más antes de finalizar con su cuestionario.

Primero se va a analizar la pregunta realizada a los encuestados que estudian el temario entero. Hay que tener en cuenta que esta pregunta ha sido respondida solo por 6 personas (25 % de los encuestados), ya que son esas 6 personas que han dicho que estudian el temario entero en la pregunta anterior.

Teniendo esto en cuenta se ha podido observar lo siguiente referente a si alguna vez se han planteado estudiar solo ejercicios similares a los que han caído en exámenes anteriores:

1. Distribución de respuestas:

- Sí: 5 personas.
- No: 1 personas.

2. Porcentaje de respuestas:

- Sí: 83.3 %.
- No: 16.7 %.

3. Análisis de la distribución:

- Respuesta “Sí” (83.3 %): Aquí se refleja el número de estudiantes universitarios encuestados que se han planteado cambiar su método de estudio y adquirir el de estudiar solo preguntas relacionadas con las que caen en los exámenes. Es decir, la mayoría podría haber pertenecido al otro grupo de encuestados.
- Respuesta “No” (16.7 %): Representa a los estudiantes universitarios (en este caso 1) encuestados que no se han planteado dejar de estudiar el temario entero.

Una vez concluido el estudio de las respuestas del primer grupo de encuestados, se procede a analizar el grupo objetivo. De este grupo lo que se busca saber a continuación es si en alguna ocasión se han visto perjudicados por haber aplicado dicho método de estudio. A partir de aquí se ha podido observar lo siguiente:

1. Distribución de respuestas:

- Sí: 12 personas.
- No: 6 personas.

2. Porcentaje de respuestas:

- Sí: 66.7 %.
- No: 33.3 %.

3. Análisis de la distribución:

- Respuesta “Sí” (66.7 %): Aquí se refleja el número de estudiantes universitarios encuestados que se han visto perjudicados por la utilización de dicho método. Dos tercios de dichos estudiantes sí se han visto perjudicados, lo que representa una gran cantidad.
- Respuesta “No” (33.3 %): Representa a los estudiantes universitarios encuestados que no se han visto perjudicados por dicho método. Un tercio de dichos estudiantes aseguran que no se han visto perjudicados por dicho método de estudio, sin embargo, esta opinión puede cambiar en el futuro.

Ahora que se sabe la cantidad de universitarios que utilizan dicho método que se han visto perjudicados, se quiere saber si a pesar de ello volverían a utilizarlo.

Para ello se les plantea dicha pregunta de la cual se han obtenido los siguientes resultados:

1. Distribución de respuestas:

- Sí: 16 personas.
- No: 2 personas.

2. Porcentaje de respuestas:

- Sí: 88.9 %.
- No: 11.1 %.

3. Análisis de la distribución:

- Respuesta “Sí” (88.9 %): Aquí se refleja el número de estudiantes universitarios encuestados que sí volverían a utilizar dicho método de estudio. Un porcentaje muy elevado teniendo en cuenta la gran cantidad de encuestados que sí se habían visto perjudicados por su utilización.
- Respuesta “No” (11.1 %): Representa a los estudiantes universitarios encuestados que volverían a utilizar dicho método de estudio. Si se tiene en cuenta la gran cantidad de encuestados que se habían visto perjudicados por dicho método de estudio, este porcentaje es bastante bajo.

### 3.1.3. Análisis de resultados

De las primeras 2 preguntas demográficas, se puede observar que la gran mayoría de los encuestados tiene entre 18 y 24 años, habiendo solo uno de ellos que tiene entre 25 y 34. También se observa que, en su mayoría son hombres, habiendo sólo 4 mujeres.

De la pregunta relacionada con la carrera que se estudia se puede ver cómo el 79.16 % están cursando una ingeniería. Dichos estudiantes están más que acostumbrados a la realización de exámenes con un gran número de preguntas tanto teóricas como prácticas, por lo que son muy interesantes para evaluar su método de estudio y por tanto confirmar si estudian o no solo las preguntas que pueden entrar en un examen o por el contrario estudian el temario entero.

Las respuestas a estas tres primeras preguntas comunes a ambos cuestionarios eran esperables, esto se debe a la relación que presentan los encuestados con el origen de la muestra.

Esta relación se hace más que evidente en la pregunta relacionada con la carrera que se estudia ya que en la Facultad de Informática de la Universidad Complutense de Madrid (UCM) coinciden las carreras de Ingeniería Informática, Ingeniería de Computadores, Ingeniería del Software, Doble grado de ADE e Informática y el Grado de Videojuegos, además de otras carreras. Por ello, el porcentaje de estudiantes encuestados que cursen una carrera relacionada con la informática sea del 70.83 % ya que, si bien en ese porcentaje habrá también estudiantes de otras universidades, muchos de ellos serán de la UCM por su relación con el origen de la muestra.

Con relación a la cantidad de horas que estudian al día se puede observar cómo las respuestas están bastante repartidas, sin embargo, la mayoría le suele dedicar solo entre 1 y 3 horas al día. Esto puede deberse a que en las carreras de ingeniería se suele ir avanzando a lo largo del curso mediante la realización de proyectos o prácticas, mientras que en el resto de las que han proporcionado, los encuestados se suele tener que estudiar un grueso de temario mayor y, por tanto, dedicarle más tiempo en momentos puntuales. Mientras una es menos tiempo al día, pero más días, otra es menos días, pero mucho más tiempo al día.

Al observar el análisis descriptivo referente a la pregunta del método de estudio utilizado, se puede ver cómo la gran mayoría estudia realizando ejercicios similares, enfocándose en realizar una preparación más específica y dejando de lado la asimilación del temario completo. Por otro lado, el resto de los estudiantes, que representan a la minoría, estudian el temario entero sugiriendo la búsqueda de una comprensión más profunda sobre los conceptos impartidos. Observando las carreras estudiadas, lo más probable es que esta minoría represente aquellas carreras que no son ingenierías en las que no hay tantas preguntas prácticas que se repitan año tras año. Esto también se relaciona con la cantidad de horas estudiadas, si los estudiantes deciden centrarse en ciertas partes del temario realizando solo ejercicios similares a los de exámenes anteriores, están disminuyendo su tiempo de estudio con respecto a aquellos que se estudian todo el temario.

Tomando lo respondido por aquellos estudiantes que estudian el temario entero, se puede concluir que la práctica de estudio que consiste en estudiar solo ejercicios similares es muy común entre los estudiantes ya que, 23 de los 24 encuestados han utilizado o se han planteado utilizar dicho método de estudio.

Ahora que se sabe que la gran mayoría ha usado o se ha planteado usar dicho método, lo que se busca saber es si la idea que tiene el proyecto para realizarse basada en los problemas que acarrea la utilización de dicho método de estudio a

futuro es o no correcta. Para ello es interesante analizar los resultados obtenidos en las dos últimas preguntas y estudiarlos en conjunto.

Se ha podido observar que, a pesar de que hay una gran mayoría que se ha visto perjudicada por la utilización de este método (ya sea por la no asimilación de conceptos importantes que se siguen utilizando o por otros motivos), solo 2 personas decidirían no volver a utilizar este método, mientras que las otras 16 no tendrían ningún problema.

Esto sumado a los resultados obtenidos en la pregunta realizada aquellas personas que se estudiaban el temario entero sirven de ayuda para ver cómo, efectivamente, existe un problema con el método de estudio de los estudiantes universitarios ya que, a pesar de estar viéndose en ocasiones perjudicados, estos no ven la necesidad de cambiar. Por ello, el presente proyecto busca proporcionarles una alternativa orientada para ellos y que les ayuda a intentar interiorizar mejor los conceptos generales aun utilizando su propio método de estudio.

Se busca que no tengan que dedicarle una gran cantidad de tiempo con respecto al que le dedicaban antes y que les sea cómoda la asimilación de dichos conceptos para que se den cuenta del problema que puede surgir por la utilización de dicho método.

## 3.2. Análisis de búsqueda de prácticas

En este segundo cuestionario, se busca recopilar información sobre la opinión de los estudiantes que buscan prácticas en empresas. Mediante este cuestionario, se pretende verificar si a los estudiantes les resultó fácil dicha búsqueda, o si, por el contrario, tuvieron dificultades tanto para ponerse en contacto con empresas, como para encontrar un portal acondicionado para ello.

### 3.2.1. Diseño del cuestionario

La estructura de este cuestionario sigue la misma que se plantea en el cuestionario anterior. En esta ocasión, el cuestionario consta de ocho preguntas. Las cuatro primeras son preguntas comunes a todos los usuarios, mientras que las cuatro restantes solo aparecerán en función de la respuesta proporcionada a las preguntas que actúan como *screeener*.

Al igual que en el anterior cuestionario, todas las preguntas que se le proponen al usuario son de respuesta obligatoria.

Todas las preguntas, a excepción de la relacionada con el grado o carrera estudiados, son cerradas ya que buscan obtener una respuesta del tipo sí o no, o un grado de satisfacción para luego poder realizar un estudio cuantitativo de los resultados.

Las tres primeras preguntas son iguales a las del cuestionario anterior y buscan obtener información general sobre los encuestados mediante la realización de preguntas demográficas.

La cuarta busca actuar como *screeener*, ya que en este cuestionario se pretende conocer la opinión que tienen los estudiantes que sí han cursado prácticas en empresa o que han intentado buscarlas, por lo que si no las ha cursado, no va a poder proporcionar datos para el estudio.

Si ha realizado o buscado prácticas en empresa, entonces se pretende investigar el grado de dificultad que le ha supuesto encontrarlas y si las ha encontrado a través de un portal facilitado por el centro en el que estudia.

Si el usuario las encontró a través de dicho portal, se busca obtener el grado de satisfacción que éste ha tenido encontrando dicho portal y utilizándolo. Si, por el contrario, no utilizó ningún portal para encontrar las prácticas, el cuestionario finaliza, puesto que ya ha proporcionado la información que se necesitaba, por lo que dicha pregunta actúa también como un *screeener*.

Una vez el usuario haya respondido a todas las preguntas, se almacenarán sus respuestas y se podrá realizar posteriormente un estudio cuantitativo sobre los datos obtenidos.

### 3.2.2. Análisis descriptivo de la muestra

Como se ha explicado anteriormente se van a volver a exponer los datos relacionados con las tres primeras preguntas del cuestionario, comunes a ambos. Sin embargo, en esta ocasión se omitirán sus respectivas explicaciones en la parte de resultados, ya que son análogas a las expuestas previamente. Lo único que cambia es la explicación de los estudiantes objetivos de la encuesta proporcionada en la pregunta de la carrera que se estudia.

De la primera pregunta demográfica relacionada con la edad se observa la siguiente distribución:

#### 1. Distribución de respuestas:

- Menor de 18: 0 personas.
- Entre 18 y 24 años: 23 personas.
- Entre 25 y 34 años: 1 persona.
- Entre 35 y 44 años: 0 personas.
- Más de 45 años: 0 personas.

#### 2. Porcentaje de respuestas:

- Menor de 18: 0 %.
- Entre 18 y 24 años: 95.8 %.
- Entre 25 y 34 años: 4.2 %.
- Entre 35 y 44 años: 0 %.
- Más de 45 años: 0 %

### 3. Promedio de edad:

Para calcular el promedio se asignarán valores numéricos a cada opción. Para la primera opción se tomará el 16 (Para utilizar un número cercano a 18 y menor que sea representativo); para la segunda, el 21; para la tercera, el 29.5; para la cuarta, 39.5; y para la quinta, 50 (Para seleccionar un número mayor que 45 y que se separe un poco de él).

Gracias a la asignación de estos valores numéricos se puede calcular el promedio obteniendo un valor de 21.354 años, es decir aproximadamente 21 años. (El promedio se ha calculado como  $((0*16) + (23*21) + (1*29.5) + (0*39.5) + (0*50)) / (0+23+1+0+0)$ ).

El valor promedio era el esperado, ya que la mayoría de las respuestas estaban entre 18 y 24 años.

### 4. Análisis de distribución:

Se observa que la mayoría de los encuestados (95.8 %) tienen entre 18 y 24 años y el encuestado restante (4.2 %) entre 25 y 34 por lo que la encuesta está predominada por gente joven.

De la segunda pregunta demográfica relacionada con el sexo se obtiene la siguiente distribución:

#### 1. Distribución de respuestas:

- Hombre: 20 personas.
- Mujer: 4 personas.

#### 2. Porcentaje de respuestas:

- Hombre: 83.3 %.
- Mujer: 16.7 %.

#### 3. Análisis de la distribución:

- Respuesta “Hombre” (83.3 %): Aquí se refleja el número de estudiantes universitarios encuestados.
- Respuesta “Mujer” (16.7 %): Representa la cantidad de estudiantes universitarias encuestadas.

De la tercera y última pregunta común a ambos cuestionarios relacionada con la carrera que se estudia se han obtenido los siguientes resultados:

#### 1. Distribución de respuestas:

- Ingeniería informática: 11 personas.
- Administración y Dirección de empresas: 3 personas.

- Ingeniería Química: 3 personas.
- Doble Grado en ADE e Informática: 2 personas.
- Ingeniería de Computadores: 2 personas.
- Grado de Videojuegos: 1 persona.
- Ingeniería del Software: 1 persona.
- Fabricación de Productos Farmacéuticos: 1 persona.

2. Porcentaje de respuestas:

- Ingeniería informática: 45.83
- Administración y Dirección de empresas: 12.50 %.
- Ingeniería Química: 12.50 %.
- Doble Grado en ADE e Informática: 8.34 %.
- Ingeniería de Computadores: 8.34 %
- Grado de Videojuegos: 4.17 %.
- Ingeniería del Software: 4.17 %.
- Fabricación de Productos Farmacéuticos: 4.17 %.

3. Análisis de la distribución:

Al tratarse de una pregunta abierta, sacar valores como el promedio o la mediana no va a resultar útil de cara a realizar un análisis y, aun sacando la moda y viendo que Ingeniería Informática es la carrera más cursada entre nuestros encuestados tampoco se obtiene un buen análisis de la muestra.

Para esta encuesta los estudiantes objetivo son aquellos que sí han realizado prácticas en empresas o que al menos las han buscado. Aquellos que no han hecho ninguna de las dos cosas no pueden proporcionar información relevante para nuestro estudio. Por este motivo, no se puede obtener una idea inicial de que porcentaje de estudiantes objetivos hay.

Para la primera pregunta exclusiva de este cuestionario y con la finalidad de actuar como *screeener* para saber cuántos de los encuestados han realizado o, al menos, han buscado prácticas en empresas, se ha obtenido la siguiente distribución:

1. Distribución de respuestas:

- Sí: 19 personas.
- No: 5 personas.

2. Porcentaje de respuestas:

- Sí: 79.2 %.
- No: 20.8 %.

### 3. Análisis de la distribución:

- Respuesta “Sí” (79.2 %.): Aquí se refleja el número de estudiantes universitarios encuestados que sí han realizado o al menos buscado prácticas en empresas. Estos son los estudiantes que son objetivo de la encuesta, ya que lo que se quiere saber es su experiencia con dichas prácticas o búsqueda.
- Respuesta “No” (16.7 %): Representa a los estudiantes universitarios encuestados que ni han realizado prácticas en empresas ni las han buscado. Dado que no han realizado ninguna de las dos no pueden proporcionar información relevante para el estudio relacionado con la accesibilidad y utilidad de los portales web de búsqueda de prácticas.

Antes de seguir con las siguientes preguntas hay que tener en cuenta que ya no se cuenta con 24 estudiantes, si no que ahora solo con 19 ya que 5 no habían realizado o buscado prácticas en empresas.

A continuación, se preguntó por la facilidad que han tenido los encuestados para encontrar un portal web para la búsqueda de prácticas se ha observado la siguiente distribución (La escala va de muy difícil a muy fácil siendo el 1 muy difícil y el 5 muy fácil):

#### 1. Distribución de respuestas:

- 1 (Muy difícil): 0 personas.
- 2 (Difícil): 8 personas.
- 3 (Neutro): 6 personas.
- 4 (Fácil): 4 personas.
- 5 (Muy fácil): 1 persona.

#### 2. Porcentaje de respuestas:

- 1 (Muy difícil): 0 %.
- 2 (Difícil): 42.1 %.
- 3 (Neutro): 31.6 %.
- 4 (Fácil): 21.1 %.
- 5 (Muy fácil): 5.3 %.

#### 3. Análisis de la distribución:

Para este tipo de preguntas en las que los valores son categóricos, resulta más apropiado sacar valores como la mediana y la moda antes que un promedio.

La mediana es la opción 3 (Neutro), ya que si se ordenan los datos de las 19 respuestas ( $0 + 8 + 6 + 4 + 1$ ) el décimo valor pertenece al grupo 3.

La moda es la opción 2 (Difícil), ya que es el valor que aparece con más frecuencia en esa colección de datos.

A continuación, se busca saber si la Universidad en la que estudias te ha proporcionado un portal para dicha búsqueda de prácticas o no. Hay que tener en cuenta que si no te han facilitado dicho portal el cuestionario finaliza.

A partir de las respuestas proporcionadas, se obtiene la siguiente distribución:

1. Distribución de respuestas:

- Sí: 17 personas.
- No: 2 personas.

2. Porcentaje de respuestas:

- Sí: 89.5 %.
- No: 10.5 %.

3. Análisis de la distribución:

- Respuesta “Sí” (89.5 %): Aquí se refleja el número de estudiantes universitarios encuestados a los que su universidad les ha proporcionado un portal para la búsqueda de prácticas. Estos son los estudiantes objetivo ya que van a poder proporcionar su experiencia con el uso de dichos portales (son los que tienen más a mano), lo cual va a poder ayudar a la creación de un nuevo portal que es otro de los motivos por los que surge el presente proyecto.
- Respuesta “No” (10.5 %): Representa a los estudiantes universitarios encuestados a los que su universidad no les ha proporcionado un portal para la búsqueda de prácticas y por tanto han tenido ellos que buscar uno o tomar otra alternativa.

Las preguntas que vienen a continuación buscan conocer la experiencia que han tenido dichos estudiantes utilizando los portales de búsqueda de prácticas que les han sido proporcionados.

Llegados a este punto la cantidad de estudiantes que responden a las dos últimas preguntas planteadas en la encuesta es de 17, ya que mediante la pregunta anterior se han eliminado aquellos estudiantes que no eran objetivo de la encuesta.

La primera pregunta tiene el fin de observar que tan útil les ha resultado dicho portal. Se ha obtenido la siguiente distribución:

1. Distribución de respuestas:

- 1 (Muy poco útil): 1 persona.
- 2 (Poco útil): 6 personas.
- 3 (Neutro): 6 personas.
- 4 (Útil): 1 persona.

- 5 (Muy útil): 3 personas.

2. Porcentaje de respuestas:

- 1 (Muy poco útil): 5.9 %.
- 2 (Poco útil): 35.3 %.
- 3 (Neutro): 35.3 %.
- 4 (Útil): 5.9 %.
- 5 (Muy útil): 17.6 %.

3. Análisis de la distribución: Como se ha explicado anteriormente, para este tipo de preguntas en las que los valores son categóricos, resulta más apropiado sacar valores como la mediana y la moda antes que un promedio.

La mediana es la opción 3 neutro ya que el valor que se encuentra en el medio cuando los datos se ordenan. Hay 17 respuestas (1 + 6 + 6 + 1 + 3) por lo que el valor que se va a tomar es el que se situará en la novena posición, lo que hace que la mediana en este caso corresponda a la opción 3 (Neutro).

La moda es bimodal ya puede ser tanto la opción 2 (Poco útil) como la opción 3 (Neutro).

De estos datos se puede observar que dicho portal ofrece opiniones divididas, aunque se observa una tendencia hacia la sensación de ser poco útil.

La segunda y última pregunta busca saber lo fácil que les ha sido encontrar y utilizar a los estudiantes universitario dicho portal. De esta pregunta se ha obtenido la siguiente distribución:

1. Distribución de respuestas:

- 1 (Muy difícil): 4 personas.
- 2 (Difícil): 6 personas.
- 3 (Neutro): 4 personas.
- 4 (Fácil): 0 personas.
- 5 (Muy fácil): 3 personas.

2. Porcentaje de respuestas:

- 1 (Muy difícil): 23.5 %.
- 2 (Difícil): 35.3 %.
- 3 (Neutro): 23.5 %.
- 4 (Fácil): 0 %.
- 5 (Muy fácil): 17.6 %.

3. Análisis de la distribución: Como ya se ha explicado anteriormente, para este tipo de preguntas en las que los valores son categóricos, resulta más apropiado sacar valores como la mediana y la moda antes que un promedio.

La mediana es la opción 2 (Difícil), ya que si se ordenan los datos de las 17 respuestas ( $4 + 6 + 4 + 0 + 3$ ) el noveno valor pertenece al grupo 2.

La moda es la opción 2 (Difícil), ya que es el valor que aparece con más frecuencia en esa colección de datos.

Se observa entonces que les ha resultado difícil tanto encontrar como utilizar el portal web que les proporcione su centro.

### 3.2.3. Análisis de resultados

De las tres primeras preguntas se pueden obtener las mismas conclusiones que en el análisis de método de estudio, salvo en el caso de la carrera que se estudia ya que esta no es tan relevante con respecto a la opinión sobre el uso o la utilización de un portal web para buscar prácticas.

Como se ha expuesto anteriormente, solo 19 de los 24 encuestados habían realizado o buscado prácticas en empresas.

Observando las respuestas, el diagrama de barras asociado (véase apéndice A.2) y observando los valores de la mediana y la moda se puede observar cómo a la mayoría de gente le ha resultado difícil encontrar un lugar para buscar prácticas. Si bien también hay encuestados que les ha resultado fácil, si se analiza en conjunto, se puede concluir que no ha sido una tarea fácil. Esto no significa que haya tenido que ser difícil, pero sí que al menos dicha tarea se puede mejorar y hacer más fácil. Por esta razón, se propuso la idea de desarrollar un portal web para proporcionar una alternativa a los ya existentes.

Si bien se busca una alternativa a los ya existentes y, por tanto, debería ser indiferente si a los estudiantes se les ha proporcionado o no un portal web, se considera que, de habérselo proporcionado, éste sería el más accesible para ellos. Por ello se considera a dichos estudiantes el objetivo de la encuesta. Tras esto los 19 encuestados se reducen a 17.

Para poder analizar mejor los resultados obtenidos se observarán las dos últimas preguntas en conjunto.

Tras analizar los diagramas de barras, así como las medianas y las modas de ambas preguntas, se puede observar cómo, efectivamente, si bien no es que los portales no sean útiles, o muy difíciles de encontrar o utilizar, sí tiene bastante margen de mejora por lo que si los estudiantes contaran con un portal que fuera más útil, accesible y fácil de utilizar podrían comenzar a utilizarlo. Por tanto, que el portal que se busca crear este proyecto deberá lo más simple posible, intentado ser lo más intuitivo posible y ofrecer una alternativa atractiva a los ya existentes.

El objetivo principal es ayudar a aquellos estudiantes que hayan visto su experiencia de búsqueda de prácticas perjudicada por los aspectos mejorables que presentan

los portales web de búsqueda de prácticas ya existentes. Para ello se debía saber si dichos portales tenían fallos y asegurarse de si les eran o no accesibles.

# Capítulo 4

## Desarrollo

### 4.1. Requisitos

En esta sección se definen los requisitos funcionales y no funcionales que deberá tener la aplicación web para cumplir los objetivos definidos.

#### 4.1.1. Requisitos funcionales

Los requisitos funcionales definen cómo se debe comportar la aplicación y lo que hará para satisfacer las necesidades de los usuarios. Para la aplicación web desarrollada en el presente trabajo se establecieron los siguientes requisitos:

- Registro de usuarios. Los usuarios podrán crear una cuenta para acceder a la aplicación web.
- Inicio de sesión. Los usuarios accederán a la aplicación web utilizando sus credenciales.
- Modificación de perfil. Los usuarios tendrán la posibilidad de modificar su perfil en la aplicación, cambiando su nombre, foto de perfil, etc.
- Subida de vídeos. Los usuarios podrán subir vídeos de contenido educativo a la aplicación.
- Inserción de cuestionarios: Los usuarios podrán añadir cuestionarios a sus vídeos para hacer el contenido más útil para los espectadores.
- Edición de vídeos. Los usuarios podrán modificar aspectos de sus vídeos una vez subidos.
- Búsqueda de vídeos. Los usuarios tendrán herramientas para encontrar vídeos sobre los temas que les interesen.

- Visualización de vídeos. Los usuarios podrán ver los vídeos creados por otros usuarios y responder a los cuestionarios que estos incluyan.
- Visualización de resultados. Los usuarios podrán comprobar los conocimientos adquiridos a partir de los resultados que obtengan en los cuestionarios respondidos.
- Estadísticas de vídeos. Los usuarios que creen vídeos en la aplicación podrán saber la recepción que éstos tienen y su utilidad en base al número de visualizaciones y las respuestas que proporcionen los usuarios en los cuestionarios.
- Subida de ofertas. Existirán usuarios que se registren como empresa y podrán subir ofertas de empleo indicando el perfil que buscan, el salario y las plazas disponibles.
- Búsqueda de ofertas. Los usuarios tendrán herramientas para encontrar ofertas que les interesen.
- Descarga de ofertas. Los usuarios podrán descargar los documentos asociados a la oferta de trabajo subida por las empresas.
- Edición de ofertas. Los usuarios registrados como empresa podrán editar la información de una oferta ya subida por ellos.
- Envío y recepción de mensajes. Los usuarios podrán iniciar un chat en la oferta que les interese, enviar mensajes a dicha empresa, así como recibirlos.
- Visualización de chat. Los usuarios podrán ver los mensajes enviados y recibidos con aquellas empresas que hayan hablado previamente.

#### 4.1.2. Requisitos no funcionales

Los requisitos no funcionales definen los atributos que deberá tener la aplicación para garantizar la calidad del servicio que ofrece. Los requisitos no funcionales definidos para la presente aplicación son:

- Seguridad. La aplicación debe garantizar la seguridad de los usuarios, restringiendo el acceso a ciertas acciones y permitiendo modificar la información únicamente a los usuarios autorizados para ello.
- Compatibilidad. La aplicación web debe ser compatible con todos los navegadores.
- Usabilidad. La interfaz de la aplicación debe ser fácil de utilizar e intuitiva para los usuarios nuevos y contará con indicaciones para ayudar al usuario a utilizarla.
- Mantenibilidad. La aplicación debe ser fácil de actualizar y reparar los errores que surjan.

- Escalabilidad. La aplicación debe ser capaz de crecer conforme aumente el número de usuarios y contenido.
- Rendimiento. La aplicación debe manejar las peticiones de los usuarios de manera rápida y proporcionará una rápida respuesta.
- Fiabilidad. Se debe minimizar las caídas del sistema para garantizar un funcionamiento estable de la aplicación.
- Cumplimiento legal. Se debe cumplir las leyes y regulaciones correspondientes para garantizar la protección de datos y los derechos de autor.

## 4.2. Arquitectura

La aplicación que se pretende desarrollar utilizará la arquitectura Cliente-Servidor, que permite a los usuarios acceder al servicio a través de un navegador web y utilizando una interfaz de usuario. El servidor al que se conecten los clientes se compone de dos elementos:

- La aplicación. Contiene toda la lógica que gestiona las peticiones HTTP de los clientes y les muestra la interfaz con la que interactúan desde el navegador.
- La base de datos. Almacena los datos persistentes de la aplicación, como los contenidos y la información de los usuarios.

Se ha elaborado un diagrama que muestra los tres elementos que compondrán la arquitectura de la aplicación y cómo se relacionan entre ellos. Este diagrama se muestra en la figura 4.1.

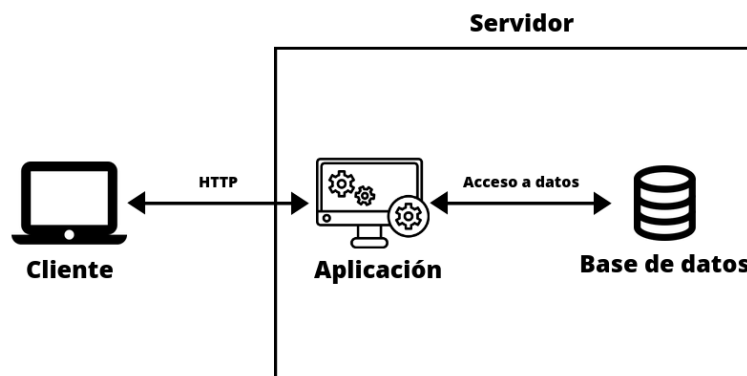


Figura 4.1: Arquitectura de la aplicación

### 4.3. Base de datos

A continuación, se define el modelo de base de datos utilizado para almacenar los datos persistentes del sistema. Se ha utilizado un modelo relacional en el que las tablas que contienen la información se relacionan unas con otras a través de identificadores.

#### 4.3.1. Diagrama entidad-relación

En la figura 4.2 se muestra el diagrama entidad-relación sobre el que se estructurará la base de datos. En este diagrama se observan dos elementos: las entidades, que se representan con un rectángulo y las relaciones, representadas con un rombo. Las entidades se relacionan entre ellas con una cardinalidad, por ejemplo, un único usuario puede crear  $N$  vídeos.

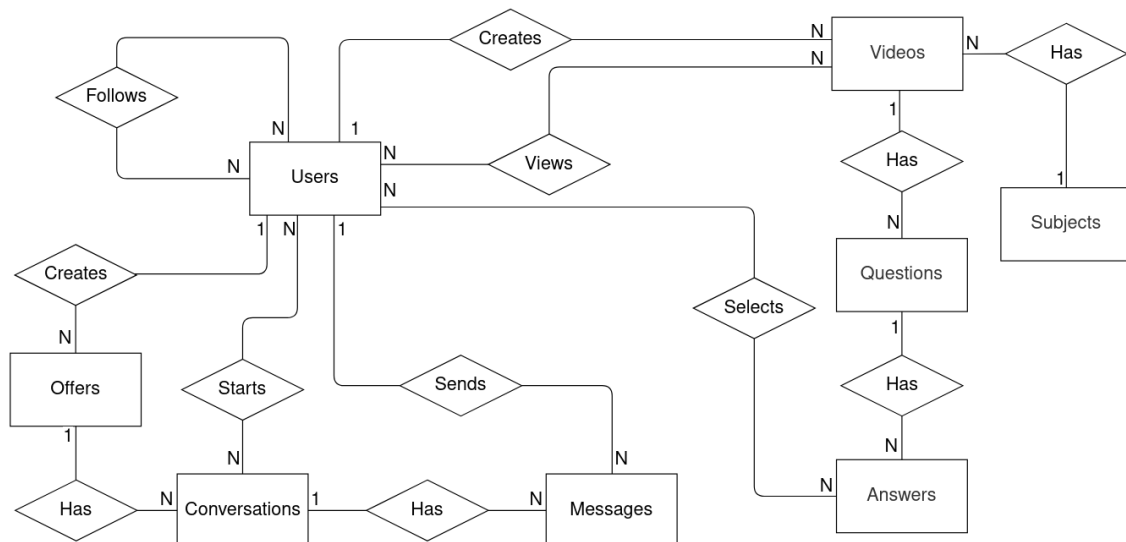


Figura 4.2: Diagrama entidad-relación

Cada entidad corresponde a una tabla en la base de datos, donde se almacenarán sus atributos, que se definirán en el siguiente apartado. Para implementar las relaciones 1-N se utilizarán identificadores con los que se relacionarán las tablas entre ellas.

Por otro lado, las relaciones N-N requerirán de tablas adicionales que implementen dicha relación. Estas tablas intermedias utilizarán las claves primarias de las tablas que relacionan y algunos atributos adicionales que sean de utilidad.

#### 4.3.2. Tablas

Seguidamente, se muestran todas las tablas de la base de datos y se explican los campos que las componen. En las tablas, aparece el nombre del campo, el tipo de

dato que es y las restricciones que se han definido para dicho campo.

#### 4.3.2.1. Users

En la tabla 4.1 se muestran los campos de la tabla *users*, donde *id* es clave primaria. El *id* es un identificador único que se asigna a cada usuario cuando se crea una cuenta. Para relacionarse con otras tablas, resulta más sencillo que el campo *id* actúe como clave foránea a que actúe como tal un campo nombre con numerosos caracteres.

Los campos *name* e *email* son únicos debido a que se busca que no haya dos usuarios que se llamen igual, ni dos cuentas diferentes que tengan un mismo correo.

El campo *ban* es un *tinyint* con valor por defecto a no baneado (valor por defecto es 0). Este campo se utiliza para saber si un usuario está o no baneado.

El campo *role* es un enumerado que almacena el tipo de cuenta que tiene el usuario. Por tanto, un usuario con el rol de *company*, se diferencia de otro con rol *standard* en que el primero puede subir ofertas de trabajo. Por su parte, un usuario con rol *admin*, tendrá los permisos para administrar todo el contenido de la página.

Nombre del campo	Tipo	Restricciones
id	BIGINT UNSIGNED	PRIMARY KEY
name	VARCHAR(255)	UNIQUE
email	VARCHAR(255)	UNIQUE
password	VARCHAR(255)	NOT NULL
role	ENUM('standard', 'admin', 'company')	DEFAULT('standard')
profile_pic	VARCHAR(255)	DEFAULT('None')
ban	TINYINT(1)	DEFAULT(0)
created_at	TIMESTAMP	
updated_at	TIMESTAMP	

Tabla 4.1: Tabla *users*

#### 4.3.2.2. Follows

La tabla *follows*, tal como se muestra en la tabla 4.2, cuenta con una clave primaria compuesta, formada por los campos *follower\_id* y *followed\_id*. Ambos campos almacenan un *id* de usuario, el usuario con *id* almacenado en *follower\_id* ha comenzado a seguir al usuario con *id* almacenado en *followed\_id*. Es clave compuesta, ya que un usuario puede tener muchos seguidores diferentes y éstos a su vez, seguir a otros usuarios, al igual que se puede dar el caso de que un usuario siga a alguien y que ese alguien no lo siga.

El campo *followed\_at* almacena la última fecha en la que han comenzado a seguirse lo usuarios, ya que pueden haberse dejado de seguir y luego volverse a seguir.

Nombre del campo	Tipo	Restricciones
follower_id	BIGINT UNSIGNED	FOREIGN KEY('users')
followed_id	BIGINT UNSIGNED	FOREIGN KEY('users')
followed_at	DATETIME	

Tabla 4.2: Tabla *follows*

#### 4.3.2.3. Videos

En la tabla 4.3 se observan los atributos de la tabla *videos*, que tiene como clave primaria el campo *id*. Los vídeos pueden llamarse igual, por eso, para distinguirlos, se les asocia un *id* nada más se crean.

El campo *subject\_id* guarda el *id* del tema que trata el vídeo. En lugar de guardar el nombre del tema al que va asociado el vídeo, se almacena solamente el *id*.

El campo *video\_path* y *thumb\_path* almacenan las rutas del fichero de vídeo y de la imagen de miniatura, respectivamente. Si bien el usuario es el que debe subir dichos archivos, en la tabla se guardan las rutas donde se almacenarán dichos archivos en el servidor, las cuales se definirán en el código.

El campo *status* almacena el estado en el que se encuentra el vídeo, siendo *valid* por defecto. Los estados *pending* y *blocked* son asignados por los administradores tras revisar los vídeos, y afectan a la visibilidad de éstos.

Nombre del campo	Tipo	Restricciones
id	BIGINT UNSIGNED	PRIMARY KEY
owner_id	BIGINT UNSIGNED	FOREIGN KEY('users')
subject_id	BIGINT UNSIGNED	FOREIGN KEY('subjects')
		DEFAULT(1)
title	VARCHAR(255)	NOT NULL
description	VARCHAR(255)	NOT NULL
video_path	VARCHAR(255)	NOT NULL
thumb_path	VARCHAR(255)	NOT NULL
status	ENUM('valid', 'pending', 'blocked')	DEFAULT('valid')
created_at	TIMESTAMP	
updated_at	TIMESTAMP	

Tabla 4.3: Tabla *videos*

#### 4.3.2.4. Subjects

La tabla *subjects*, representada en la tabla 4.4, tiene el campo *id* como clave primaria. El *id* se le asigna al tema en el momento en el que éste se crea (los usuarios elegirán si subir el vídeo a un tema ya existente o crear ellos uno).

El campo *name* es único para poder agrupar todos los vídeos de un mismo tema juntos y que no haya dos temas con nombres idénticos.

Nombre del campo	Tipo	Restricciones
id	BIGINT UNSIGNED	PRIMARY KEY
name	VARCHAR(255)	UNIQUE

Tabla 4.4: Tabla *subjects*

#### 4.3.2.5. Visualized\_videos

La tabla 4.5 muestra los campos de la tabla *visualized\_videos*, que tiene una clave primaria compuesta formada por los campos *user\_id* y *video\_id*. El motivo de que sea compuesta se debe a que la tabla representa los vídeos vistos por un usuario, entendiendo como tal que un mismo usuario puede ver más de un vídeo, al igual que un vídeo puede haber sido visto por varios usuarios.

El campo *fav* es un *tinyint* que representa si el usuario eligió dicho vídeo como favorito o no. Por defecto el usuario no ha seleccionado el vídeo como favorito (por defecto es 0).

El campo *score* guarda la cantidad de preguntas acertadas por el usuario tras haberlas contestado durante el vídeo. Se guarda la puntuación que el usuario obtuvo la última vez, ya que puede haber visto y respondido a las preguntas más de una vez con anterioridad, pero solo se le mostrará la puntuación que obtuvo en la última vista, ya que es la más reciente.

El campo *date* almacena la última fecha en la que se visualizó el vídeo, ya que éste ha podido ser visualizado más de una vez por el mismo usuario.

Nombre del campo	Tipo	Restricciones
user_id	BIGINT UNSIGNED	FOREIGN KEY('users')
video_id	BIGINT UNSIGNED	FOREIGN KEY('videos')
fav	TINYINT(1)	DEFAULT(0)
score	INT	DEFAULT(0)
date	DATETIME	

Tabla 4.5: Tabla *visualized\_videos*

#### 4.3.2.6. Questions

La tabla *questions*, cuyos campos se observan en la tabla 4.6, tiene como clave primaria el campo *id*. Pueden existir preguntas iguales en diferentes vídeos, por lo que al crearse una, ésta obtiene su *id* único e identificativo. El vídeo al que pertenece la pregunta se referencia mediante la clave externa *video\_id*, que guarda el *id* de dicho vídeo.

El campo *text* es todo el cuerpo de la pregunta, excluyendo las posibles respuestas asociadas a esta.

El campo *question\_time* almacena el minuto del vídeo seleccionado por el usuario para que aparezca la pregunta que se ha creado.

Nombre del campo	Tipo	Restricciones
id	BIGINT UNSIGNED	PRIMARY KEY
video_id	BIGINT UNSIGNED	FOREIGN KEY('videos')
text	VARCHAR(255)	NOT NULL
question_time	INT	NOT NULL

Tabla 4.6: Tabla *questions*

#### 4.3.2.7. Answers

La tabla *answers*, representada en la tabla 4.7, tiene como clave primaria el campo *id*. Esto debido a que la tabla va a almacenar respuestas diferentes asociadas a una misma pregunta, por lo que se necesita un campo que haga distinción entre ellas.

El campo *question\_id* es la clave externa que referencia el *id* de la pregunta asociada a esta respuesta. Es decir, si una pregunta tuviera cuatro posibles respuestas, existirían cuatro entradas en esta tabla con un mismo *question\_id*. Por su parte, en el campo *text* se almacena el cuerpo de la respuesta.

El campo *is\_correct* actúa como booleano indicando si la respuesta es correcta o no (las preguntas son tipo test por lo que se puede elegir una y verificar si es o no correcta). Toma como valor inicial *false* (valor por defecto de 0).

Nombre del campo	Tipo	Restricciones
id	BIGINT UNSIGNED	PRIMARY KEY
question_id	BIGINT UNSIGNED	FOREIGN KEY('questions')
text	VARCHAR(255)	NOT NULL
is_correct	TINYINT(1)	DEFAULT(0)

Tabla 4.7: Tabla *answers*

#### 4.3.2.8. User\_answer

La tabla 4.8 representa los campos de la tabla *user\_answer*, cuenta con una clave primaria compuesta, formada por los campos *user\_id* y *answer\_id*. Es compuesta ya que indica lo que un usuario ha respondido, es decir, guarda el *id* del usuario que ha respondido y el *id* de la respuesta que ha seleccionado (en la tabla respuestas es donde se encuentra qué respuesta es, a qué pregunta va asociada y si es correcta o no).

El campo *date* sirve para almacenar la fecha en la que se ha seleccionado dicha respuesta, ya que se quiere guardar solo la última vez que se respondió a dicha pregunta.

Nombre del campo	Tipo	Restricciones
user_id	BIGINT UNSIGNED	FOREIGN KEY('users')
answer_id	BIGINT UNSIGNED	FOREIGN KEY('answers')
date	DATETIME	

Tabla 4.8: Tabla *user\_answer*

#### 4.3.2.9. Offers

La tabla *offers*, mostrada en la tabla 4.9, cuenta con un campo *id* que actúa como clave primaria. Las ofertas pueden llamarse igual o pedir las mismas características, por ello, para distinguir una de otras, se les asocia el campo *id* nada más se crean.

El campo *owner\_id* actúa como clave foránea ya que está relacionado con el campo *id* de la tabla *users*, de los cuales únicamente los usuarios con el rol *company* pueden subir ofertas.

El campo *vacants* representa las plazas libres que tiene la empresa para dicha oferta de trabajo.

El campo *document\_path* almacena la ruta del fichero que quiere subir la compañía (las ofertas cuentan con un archivo de texto que se puede descargar). La compañía es la que debe subir dicho archivo y en este campo se guarda la ruta donde se almacenará dicho archivo en el servidor (dicha ruta se definirá en el código).

Nombre del campo	Tipo	Restricciones
id	BIGINT UNSIGNED	PRIMARY KEY
owner_id	BIGINT UNSIGNED	FOREIGN KEY('users')
title	VARCHAR(255)	NOT NULL
description	VARCHAR(255)	NOT NULL
salary	DOUBLE	NOT NULL
vacants	INT	NOT NULL
document_path	VARCHAR(255)	NOT NULL
created_at	TIMESTAMP	
updated_at	TIMESTAMP	

Tabla 4.9: Tabla *offers*

#### 4.3.2.10. Conversations

La tabla *conversations*, representada en la tabla 4.10, tiene un campo *id* que actúa como clave primaria. Cada conversación entre dos usuarios es única y cuando se crea se le asocia dicho *id*.

Las conversaciones siempre las compone un usuario *standard*, cuyo *id* se almacena en el campo *id\_user*, y un usuario con rol *company*, cuyo *id* estará almacenado en el campo *id\_company*.

Además, cada conversación pertenece a una oferta y, a su vez, una oferta puede contener múltiples conversaciones. Por ello, cada conversación almacena el *id* de la oferta a la que va asociada en el campo *id\_offer*.

Nombre del campo	Tipo	Restricciones
id	BIGINT UNSIGNED	PRIMARY KEY
id_user	BIGINT UNSIGNED	FOREIGN KEY('users')
id_company	BIGINT UNSIGNED	FOREIGN KEY('users')
id_offer	BIGINT UNSIGNED	FOREIGN KEY('offers')
created_at	TIMESTAMP	
updated_at	TIMESTAMP	

Tabla 4.10: Tabla *conversations*

#### 4.3.2.11. Messages

En la tabla *messages*, mostrada en la tabla 4.11, el campo *id* actúa como clave primaria. Un mensaje con el mismo contenido puede ser enviado en numerosas conversaciones por lo que cada mensaje en el momento en el que alguien lo envía se almacena y se le asigna un *id* para poder identificarlo.

El mensaje va asociado a una conversación (*id\_conversation*) y ha sido enviado por un usuario (*id\_sender*).

Nombre del campo	Tipo	Restricciones
id	BIGINT UNSIGNED	PRIMARY KEY
id_conversation	BIGINT UNSIGNED	FOREIGN KEY('conversations')
id_sender	BIGINT UNSIGNED	FOREIGN KEY('users')
content	VARCHAR(255)	NOT NULL
date	TIMESTAMP	

Tabla 4.11: Tabla *messages*

### 4.3.3. Relaciones

Para entender cómo se relacionan las tablas de la base de datos, descritas en el apartado anterior, se ha elaborado un diagrama en el que se muestran dichas relaciones a través de las claves primarias de cada tabla.

Dicho diagrama ha sido creado utilizando la herramienta [dbdiagram.io](https://dbdiagram.io)<sup>1</sup>, que permite crear diagramas de bases de datos utilizando un lenguaje de marcado. Esta herramienta es muy fácil de utilizar además de que representa de manera clara la estructura de la base de datos. Además, permite exportar en diferentes formatos el diagrama creado, siendo en este caso exportado en formato PNG para incrustarlo en el documento.

<sup>1</sup><https://dbdiagram.io/home>

En la figura 4.3 se muestra el diagrama en el que aparecen todas las tablas de la base de datos y cómo se relacionan entre ellas a través de las claves primarias.

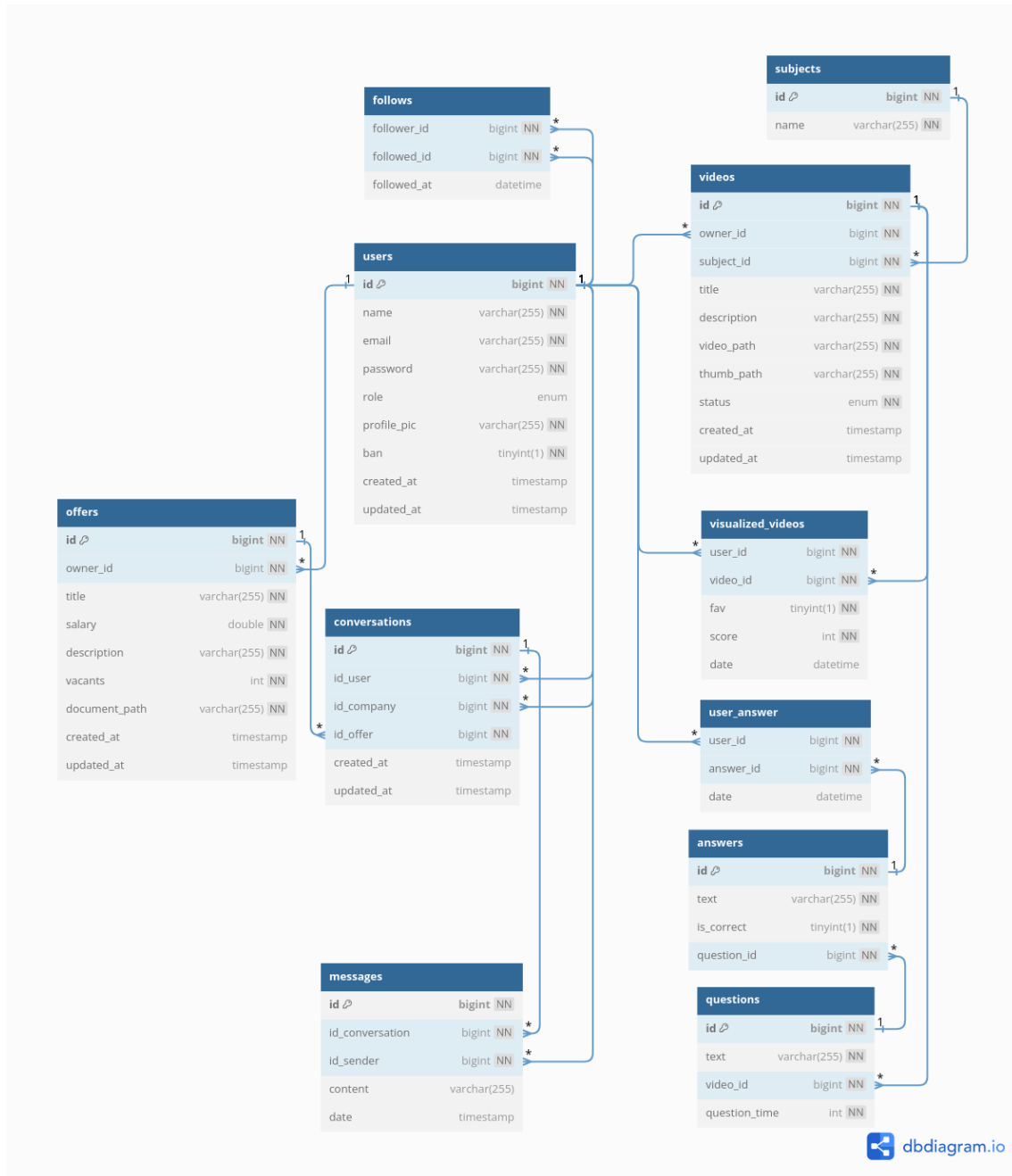


Figura 4.3: Relaciones entre las tablas

En este diagrama se puede apreciar de manera directa toda la estructura que presentará la base de datos. También se puede apreciar la cardinalidad de las relaciones, es decir, el número de veces que se pueden relacionar las tablas entre ellas.

Asimismo, se pueden distinguir las tablas intermedias, que no se ven de forma tan clara en el diagrama entidad-relación.

## 4.4. Prototipo

Para la realización de los bocetos de este proyecto se ha decidido usar la herramienta Balsamiq Wireframes<sup>2</sup>, que es una herramienta de pago, pero ofrece la posibilidad de obtener una prueba en la nube durante un mes. Esta herramienta permite realizar *mockups*, *wireframes* y prototipos de aplicaciones, entre otras funcionalidades, ofreciendo una fidelidad media. Se decidió usar esta herramienta debido a que se tenían conocimientos previos sobre su funcionamiento y a que la forma en que permite representar las diferentes vistas de una aplicación web es suficiente para pasar a digital los bocetos a papel sin tener que realizar modificaciones a los prototipos.

Existen numerosas herramientas de prototipado además de Balsamiq Wireframes, como por ejemplo Figma<sup>3</sup>, una de las más utilizadas a la hora de representar interfaces gráficas. Sin embargo, para representar los bocetos no era necesario plasmar el flujo de acciones, por lo que se consideró que sería mejor no utilizarla, ya que Balsamiq sería más adecuada para ello.

Otro ejemplo podría ser Just in Mind<sup>4</sup>, pero es una herramienta que ofrece una gran fidelidad y está pensada para utilizarse en la representación de prototipos más avanzados.

También existen algunas herramientas muy similares a Balsamiq Wireframes, como puede ser Pencil<sup>5</sup>, pero no se utilizaron debido a que se consideró que no aportaban ninguna mejora con respecto a Balsamiq.

### 4.4.1. Página principal

El boceto de la página principal de la aplicación, que aparece en la figura 4.4, incluye listas de vídeos, ordenados según diferentes parámetros: el número de visitas, el número de usuarios que los han guardado en favoritos o la fecha en que se hayan subido dichos vídeos. Para cada vídeo se muestra la miniatura del mismo así como su título para que el usuario pueda entender el contenido que va a encontrar cuando acceda al vídeo. Al final de cada lista, se incluye un botón en forma de flecha para cargar más vídeos, ya que en un primer momento sólo se cargan los más relevantes en cada categoría.

Además, a la derecha de los vídeos se incluye una barra de desplazamiento, que indica al usuario que puede desplazarse hacia abajo en la página y encontrará más contenido. Asimismo, se incorpora una barra de búsqueda para encontrar los vídeos que se deseen a partir de las palabras que introduzcan los usuarios.

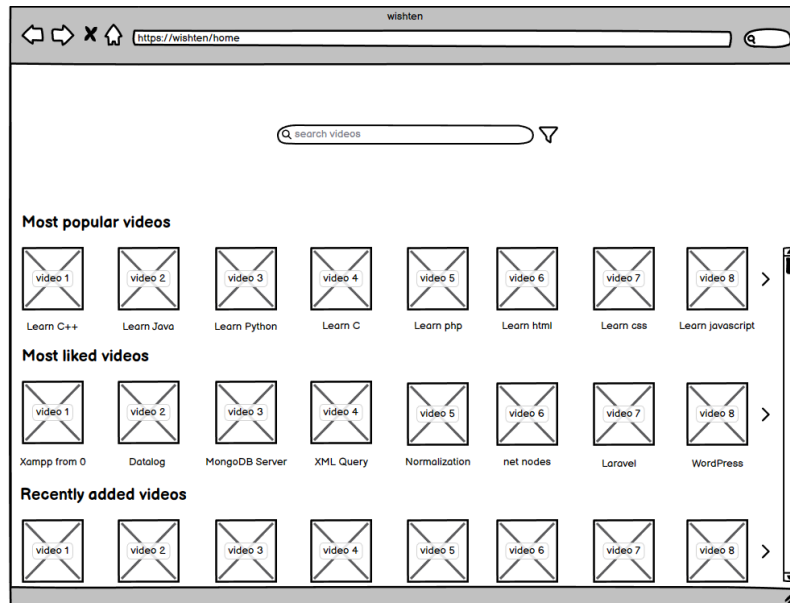
---

<sup>2</sup><https://balsamiq.com/wireframes>

<sup>3</sup><https://www.figma.com>

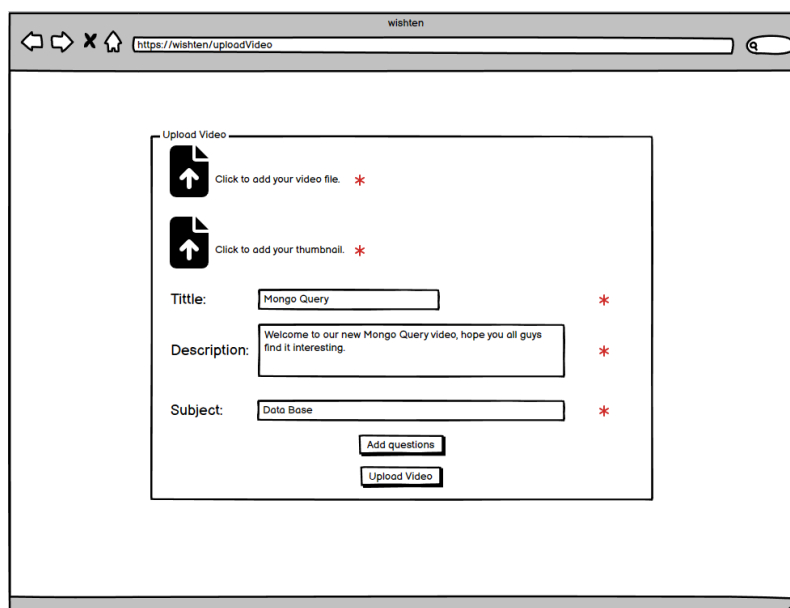
<sup>4</sup><https://www.justinmind.com>

<sup>5</sup><https://pencil.evolus.vn>

Figura 4.4: Boceto de la vista *home*

#### 4.4.2. Subir un vídeo

En la figura 4.5 se muestra el boceto de la vista para subir un vídeo. En ésta se incluyen iconos sobre los que el usuario pulsará para subir el archivo del vídeo y de la miniatura. Por otro lado, añadirá la información del vídeo: título, descripción, tema y preguntas. Todo ello siguiendo un flujo descendente, que termina en el botón para confirmar la subida del vídeo.

Figura 4.5: Boceto de la vista *uploadVideo*

### 4.4.3. Añadir un cuestionario

Tal como se observa en la figura 4.6, el usuario seleccionará en la barra de progreso del vídeo el minuto en el que desea añadir un cuestionario. A su vez, se le muestran los campos que debe rellenar, cerrando la acción con un botón para añadir el cuestionario.

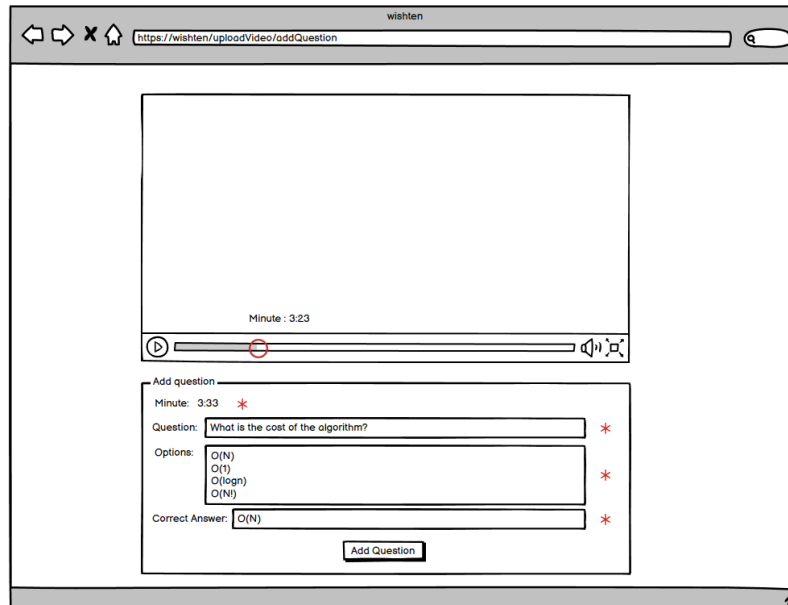


Figura 4.6: Boceto de la vista *addQuestion*

### 4.4.4. Vista de un vídeo

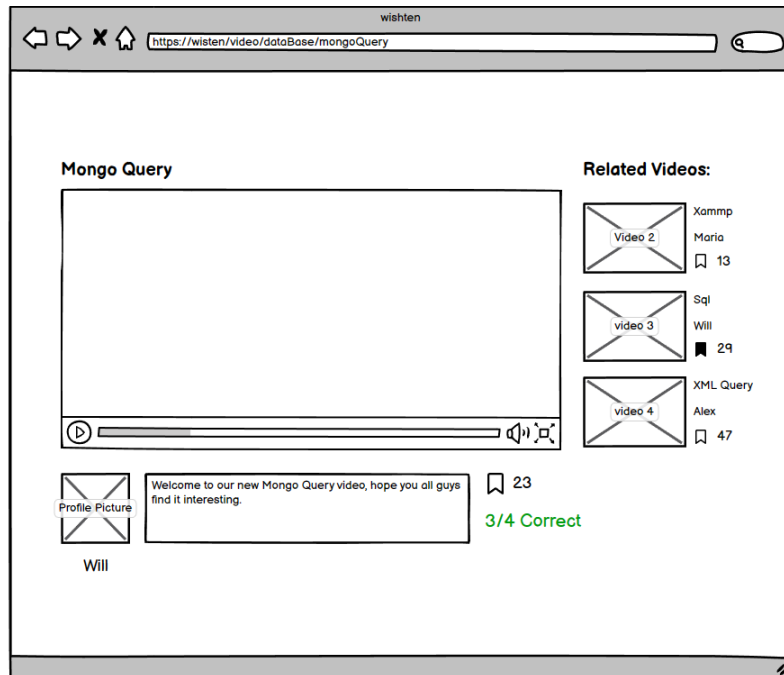
Cuando el usuario acceda a un vídeo para visualizarlo, se le mostrará la vista que aparece en la figura 4.7. En esta vista, el usuario observará en la parte central de la vista el contenido del vídeo, donde además aparecerán los cuestionarios, en caso de haber sido añadidos. En la parte inferior, el usuario tiene los controles del vídeo para pausar, subir el volumen o ponerlo en pantalla completa.

En la parte superior del vídeo se muestra el título y en la parte inferior, el resto de información relativa al vídeo: el autor, la descripción, el número de favoritos y los resultados del usuario en los cuestionarios de ese vídeo.

A la derecha del vídeo, se observa una lista con los vídeos relacionados con el que se está visualizando, es decir, los vídeos que traten el mismo tema. En cada uno de ellos se muestra la miniatura, el título, el autor y el número de favoritos.

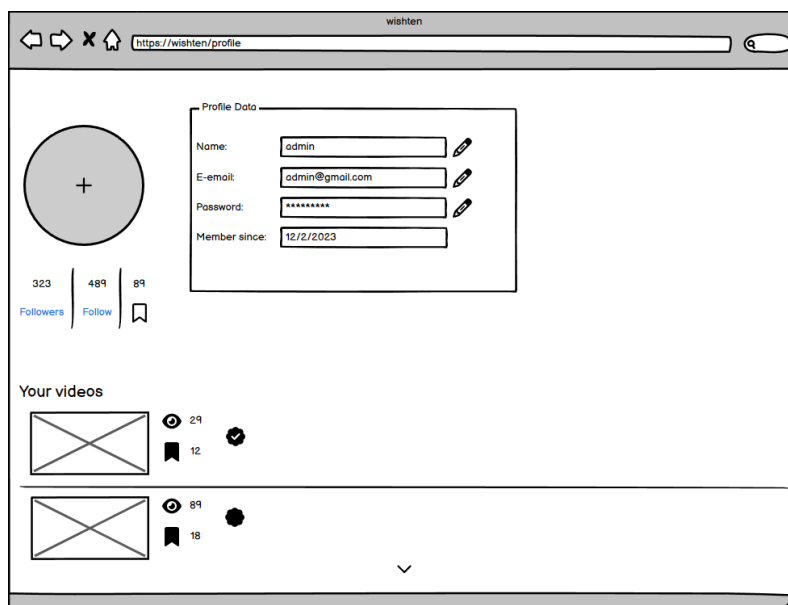
La disposición de esta vista se ha basado en la que utilizan algunas plataformas importantes de visualización de vídeos, como YouTube<sup>6</sup>.

<sup>6</sup><https://www.youtube.com>

Figura 4.7: Boceto de la vista *video*

#### 4.4.5. Perfil de un usuario

En la figura 4.8 se muestra el boceto para la vista que tendrá un usuario en su perfil. En esta vista se incluye a la izquierda el perfil público del usuario. En la parte central, se muestran los datos del usuario, con un botón en forma de lápiz que indica que puede modificar dicha información. En la parte inferior, se muestra una lista con todos los vídeos que ha subido ese usuario.

Figura 4.8: Boceto de la vista *profile*

### 4.4.6. Vista de ofertas

El usuario dispondrá de una vista con las ofertas publicadas, como se muestra en la figura 4.9. En esta vista, podrá observar una lista con la información general de cada oferta, y dos botones a la derecha: uno para descargar el documento asociado a la oferta y otro para acceder al chat y comunicarse con la empresa en cuestión. En la parte superior, también se incluye una barra de búsqueda con la que encontrar las ofertas que sean de interés para el usuario.

En caso de ser una empresa, se mostrarán botones para añadir una nueva oferta, editar o para ver las ofertas publicadas por dicha empresa.

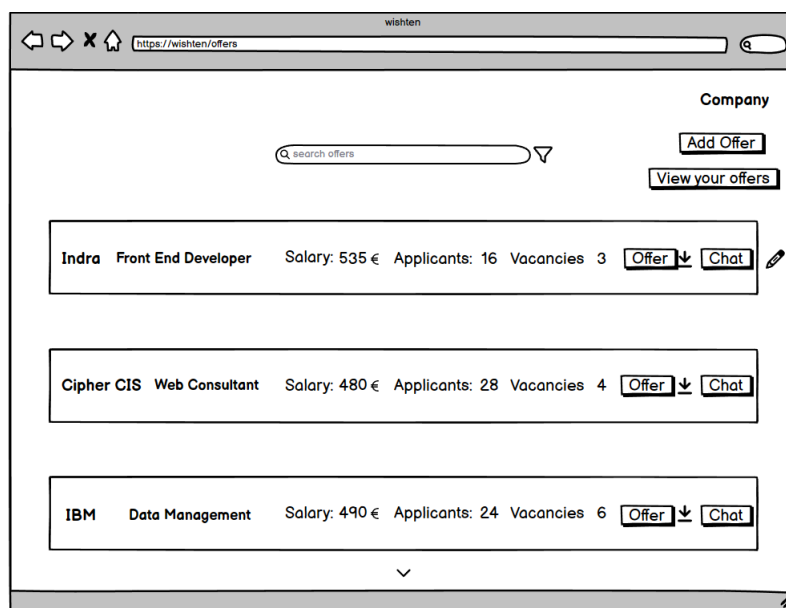


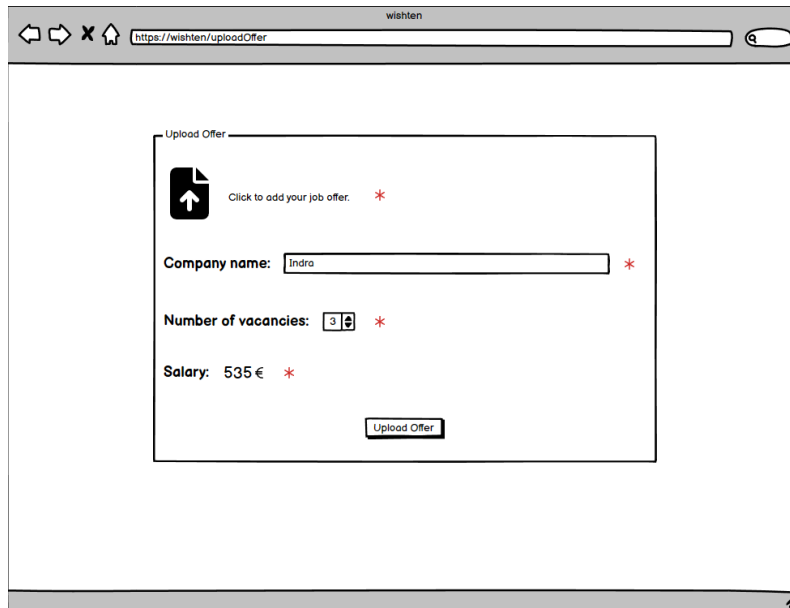
Figura 4.9: Boceto de la vista *offers*

### 4.4.7. Añadir una oferta

En la figura 4.10 se muestra la vista para añadir una nueva oferta. En esta vista, se le presentan al usuario los campos que debe rellenar para crear una oferta, siguiendo un flujo descendente para que resulte más intuitivo.

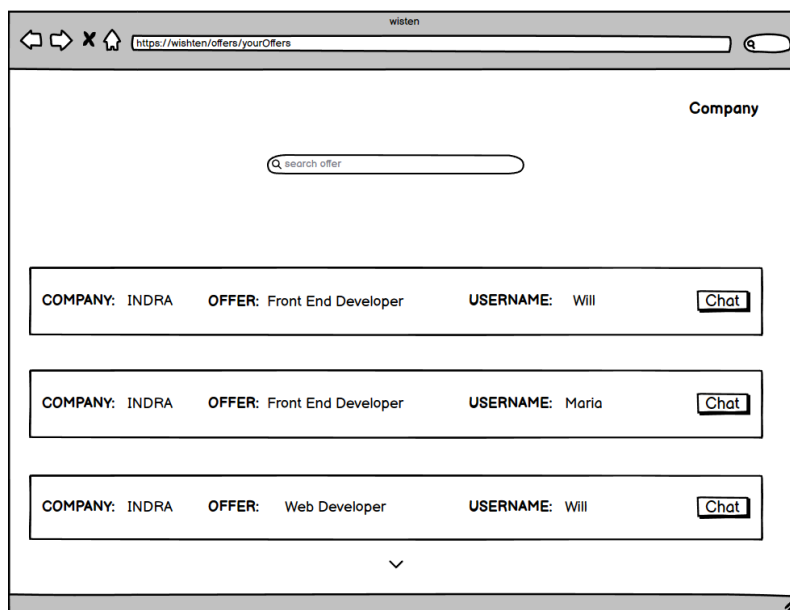
En primer lugar, un icono sobre el que pulsará para añadir el documento con toda la información de la oferta. A continuación, un campo de formulario donde añadir el nombre de la empresa. Finalmente, un selector para indicar el número de plazas ofertadas y otro campo en el que se elige el salario que se ofrece.

Toda la secuencia de acciones se cierra con el botón en la parte inferior de la vista que guardará toda la información cumplimentada.

Figura 4.10: Boceto de la vista *uploadOffer*

#### 4.4.8. Ofertas de una empresa

El boceto que se observa en la figura 4.11 le muestra al usuario registrado como empresa todas las ofertas que éste ha subido. En cada una, se muestra la información general y qué usuario se ha registrado en cada una de ellas, así como un botón para acceder al chat con cada uno de los usuarios.

Figura 4.11: Boceto de la vista *yourOffers*

### 4.4.9. Administración de usuarios

En la figura 4.12 se muestra el boceto de la vista que tendrán los administradores de la página para gestionar los usuarios. En esta vista se observa una lista que contiene la información de cada usuario y a la derecha los botones de acción referente a dicho usuario.

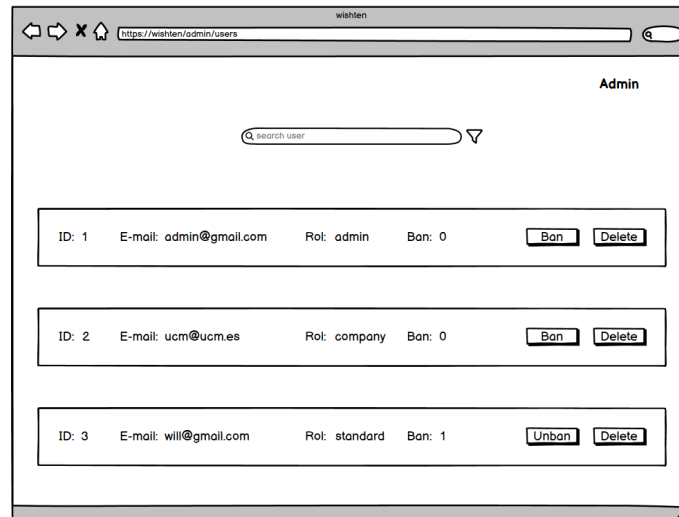


Figura 4.12: Boceto de la vista *adminUsers*

### 4.4.10. Administración de vídeos

El boceto de la página de administración de vídeos, presente en la figura 4.13, sigue la misma estructura que la de administración de usuarios, lo que facilita la tarea a los administradores.

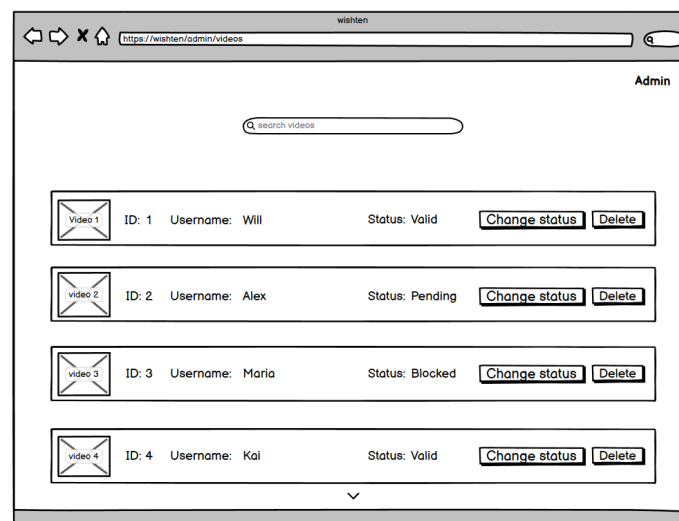


Figura 4.13: Boceto de la vista *adminVideos*

### 4.4.11. Administración de ofertas

En la figura 4.14 se muestra el boceto de la vista que tendrán los administradores de la página para gestionar las ofertas. Ésta tiene la misma estructura que las anteriores, es decir, una lista con las ofertas y botones de acción para cada una.

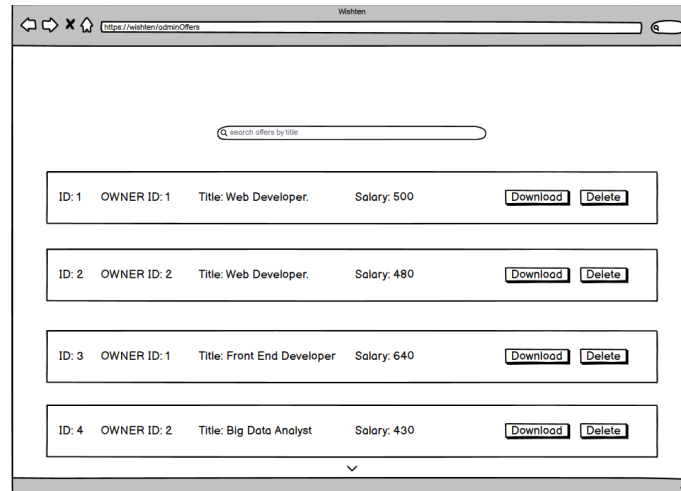


Figura 4.14: Boceto de la vista *adminOffers*

### 4.4.12. Administración de mensajes

El boceto de la página de administración de mensajes se observa en la figura 4.15 y sigue la misma estructura que el resto de páginas de administración.

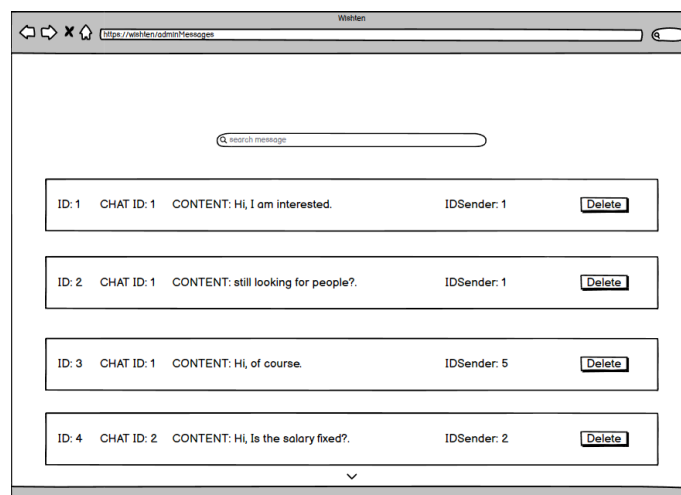


Figura 4.15: Boceto de la vista *adminMessages*

## 4.5. Análisis de tecnologías

En esta sección se analizan las distintas tecnologías que se pueden emplear en el desarrollo de la aplicación web y las que finalmente se decidió utilizar.

### 4.5.1. Framework

Un *framework* es un conjunto de herramientas que proporciona una estructura base sobre la que desarrollar *software*. De esta manera se facilita la programación de aplicaciones evitando tener que desarrollarlas desde cero. Un *framework* puede ser general o específico para un tipo de *software*.

En el presente trabajo, se analizaron los *frameworks* más utilizados en el desarrollo de aplicaciones web para determinar cuál sería el idóneo para implementar la aplicación que se pretende desarrollar.

#### 4.5.1.1. Ruby on Rails

Ruby on Rails<sup>7</sup> es un *framework* que utiliza Ruby como lenguaje de programación para desarrollar aplicaciones web. Para ello, utiliza el patrón modelo-vista-controlador, el cual separa la lógica y los datos de su representación. A pesar de ser un *framework* bastante potente debido a su facilidad de uso y buen rendimiento, el lenguaje Ruby se tendría que aprender desde cero, por lo que se decidió optar por otras propuestas.

#### 4.5.1.2. Django

En cuanto a Django<sup>8</sup>, éste ofrece una gran cantidad de documentación, además de un tutorial detallado para aprender a desarrollar aplicaciones en dicho entorno (Django, 2023). Utiliza Python como lenguaje de programación y está muy centrado en los aspectos de seguridad. Éstas características lo convertían en una buena opción para el desarrollo de la presente aplicación, sin embargo se consideró que la curva de aprendizaje sería demasiado grande como para comenzar a desarrollar la aplicación cómodamente.

#### 4.5.1.3. Node.js

Por su parte, Node.js<sup>9</sup> utiliza el lenguaje JavaScript y es un *framework* muy utilizado en aplicaciones grandes, como Netflix o LinkedIn. Además realiza operaciones de entrada-salida asíncronas, lo que evita bloqueos y ofrece un mejor rendimiento. Sin embargo, no se consideró el *framework* más adecuado para el desarrollo de

---

<sup>7</sup><https://rubyonrails.org>

<sup>8</sup><https://www.djangoproject.com>

<sup>9</sup><https://nodejs.org/es>

la aplicación, debido a la falta de estándares de desarrollo, lo que dificultaría el desarrollo y el mantenimiento del código.

#### 4.5.1.4. Laravel

Laravel<sup>10</sup> es un *framework* que utiliza PHP como lenguaje de programación, además del patrón modelo-vista-controlador para desarrollar aplicaciones web. Incluye una extensa página de documentación, que contiene estándares, recomendaciones y buenas prácticas que ayudan mucho a los desarrolladores y facilitan su aprendizaje (Laravel, 2023d). Por otra parte, el lenguaje PHP resultaba familiar ya que ha sido utilizado durante la carrera para la programación de aplicaciones web. Asimismo, la estructura de los proyectos es robusta y el código, legible y conciso, además de que cuenta con múltiples paquetes y complementos que ayudan a simplificar el desarrollo.

Por todo ello, se decidió utilizar Laravel para el desarrollo de la aplicación web, ya que es un *framework* muy completo, versátil y con una estructura robusta en la que implementar las funcionalidades que se pretende.

## 4.5.2. Gestión de la base de datos

Los datos persistentes de la aplicación deben ser gestionados a través de un sistema de gestión de bases de datos (SGDB o DBMS, en inglés, es el sistema que se encarga de manejar toda la información que se almacena en una base de datos). En esta sección se analizan los diferentes sistemas que pueden utilizarse para este propósito.

### 4.5.2.1. MongoDB

MongoDB<sup>11</sup> es un SGDB no relacional, lo que significa que no utiliza tablas que se relacionan entre sí para almacenar la información. Este sistema es útil cuando se tienen que almacenar datos no estructurados o semi estructurados, sin embargo en la base de datos previamente definida se pretende almacenar la información en tablas, de manera estructurada, ya que garantiza la integridad de los datos, lo cual es un aspecto muy importante en relación a la seguridad de la aplicación.

Por tanto, los sistemas de gestión de bases de datos no relacionales no se contemplaron para la gestión de los datos persistentes.

### 4.5.2.2. SQLite

SQLite<sup>12</sup> es un sistema de gestión muy sencillo, orientado a bases de datos pequeñas para aplicaciones a pequeña escala. Por esta razón, SQLite no es una buena

---

<sup>10</sup><https://laravel.com>

<sup>11</sup><https://www.mongodb.com>

<sup>12</sup><https://www.sqlite.org/index.html>

opción para la aplicación que se pretende desarrollar en este trabajo, ya que se requiere un sistema más potente para almacenar una gran cantidad de información de usuarios, vídeos y ofertas.

#### 4.5.2.3. PostgreSQL

PostgreSQL<sup>13</sup>, por su parte, ofrece un gran rendimiento para aplicaciones web que requieran bases de datos muy voluminosas sobre las que realizar consultas complejas. No obstante, la aplicación que se pretende desarrollar en el presente trabajo, no requiere de estas características, sino que se trata de simplificar la base de datos para lograr un desarrollo más sencillo.

#### 4.5.2.4. MySQL

Para lograr un sistema con un buen balance entre facilidad de uso y rendimiento, se optó por utilizar MySQL<sup>14</sup>, un SGDB bien conocido, ampliamente utilizado y con el que se han realizado varios proyectos a lo largo de la carrera. Este sistema es propiedad de Oracle<sup>15</sup> y cuenta con una amplia comunidad que lo respalda y gran cantidad de soporte, lo que lo hacen muy sencillo de utilizar, además de ser compatible con Laravel.

### 4.5.3. Servidor

Para ejecutar la aplicación de forma remota y hacer pruebas se eligió el servicio DigitalOcean<sup>16</sup>. Se desplegó un prototipo<sup>17</sup> en una máquina Linux y se realizó toda la instalación necesaria, tanto del *framework*, como del software de servidor (Apache<sup>18</sup>, ya que se había trabajado previamente con él), como del sistema de gestión de la base de datos.

### 4.5.4. Sistema de control de versiones

Para llevar a cabo el desarrollo de cualquier tipo de *software* es importante llevar un control sobre los avances en dicho desarrollo y los cambios que se van realizando en el código. Para esta tarea se utilizan los sistemas de control de versiones, que permiten llevar un registro de todos los cambios realizados en el código, al igual que deshacerlos o aplicarlos en paralelo al proyecto principal, para evitar conflictos.

En este sentido, se decidió utilizar Git<sup>19</sup>, un sistema distribuido de control de versiones, que permite coordinar el trabajo de varios desarrolladores, soportando

---

<sup>13</sup><https://www.postgresql.org>

<sup>14</sup><https://www.mysql.com>

<sup>15</sup><https://www.oracle.com>

<sup>16</sup><https://www.digitalocean.com>

<sup>17</sup>El prototipo es accesible a través de <http://178.128.164.243>

<sup>18</sup><https://apache.org>

<sup>19</sup><https://git-scm.com>

un flujo de trabajo no lineal, es decir, puede haber varias ramas paralelas en cada ordenador, que luego pueden unirse.

#### 4.5.4.1. GitHub

GitHub<sup>20</sup> es una plataforma de desarrollo colaborativo que utiliza el sistema Git para el control de versiones. Esta plataforma permite almacenar el código en un repositorio, accesible para todo el mundo en caso de ser público.

Además de esto, GitHub ha sido ampliamente utilizada a lo largo de la carrera para gestionar multitud de proyectos, por lo que ya se tenían los conocimientos necesarios para utilizarla eficientemente a la hora de desarrollar la presente aplicación.

Por todo ello se decidió utilizar esta plataforma para el desarrollo de la presente aplicación, cuyo código se encuentra disponible en un repositorio público<sup>21</sup>.

### 4.5.5. Editor de código

Cuando se programa cualquier tipo de *software* es importante elegir un buen editor de código que resalte la sintaxis, complete sentencias e indique los errores, entre otras cosas. Esto ayuda a que el proceso de desarrollo sea más eficiente y por tanto se analizaron las diferentes alternativas.

#### 4.5.5.1. Notepad++

Notepad++<sup>22</sup> es un editor muy utilizado para desarrollo web, sin embargo es demasiado sencillo para las dimensiones de la aplicación que se pretende desarrollar, ya que no cuenta con demasiadas características adicionales para manipular el código, más allá de resaltar la sintaxis y completar algunas palabras.

#### 4.5.5.2. Sublime Text

Sublime Text<sup>23</sup> por su parte, es más completo que Notepad++, ya que incluye más herramientas para autocompletar código y navegar sobre él. Sin embargo, al igual que Notepad++, sigue sin ser una buena opción para proyectos grandes, que requieren gestionar una gran cantidad de ficheros y de código.

#### 4.5.5.3. Visual Studio Code

Visual Studio Code<sup>24</sup> (VS Code) va un paso más allá del resto de editores de código, ya que tiene muchas funcionalidades adicionales, como *IntelliSense*, que permite

---

<sup>20</sup><https://github.com>

<sup>21</sup><https://github.com/PoloDeNaranja/Wishten>

<sup>22</sup><https://notepad-plus-plus.org>

<sup>23</sup><https://www.sublimetext.com>

<sup>24</sup><https://code.visualstudio.com>

autocompletar el código basándose en el código ya escrito previamente. También integra las funcionalidades de Git, por lo que se puede trabajar sobre el repositorio directamente desde VS Code de manera sencilla.

En adición a esto, VS Code cuenta con extensiones específicas para el código que se está desarrollando, como en este caso: PHP, HTML, JavaScript e incluso existen extensiones para Laravel.

Por todas estas razones se decidió utilizar Visual Studio Code para escribir y gestionar el código fuente de la aplicación web.

## 4.6. Implementación

En esta sección se explica en detalle cómo se ha implementado la aplicación, utilizando el *framework* de Laravel, hojas de estilos y código JavaScript para programar el *backend* (lógica) y el *frontend* (interfaz) de la aplicación.

La aplicación web se ha implementado separando la interfaz de la lógica. Ambas partes se relacionan entre ellas mediante peticiones de tipo *GET* y *POST* enviadas al servidor desde el cliente.

En el momento de comenzar con el desarrollo de la aplicación web, la última versión disponible de Laravel era Laravel 9. Se decidió continuar el desarrollo en esta versión, a pesar de que se lanzó la versión 10 en febrero del 2023 (Laravel, 2023f), ya que la versión 9 continuaría recibiendo actualizaciones de seguridad y no se requería ninguna funcionalidad de las que se añadieron con la actualización. Por todo ello, se determinó posponer la actualización a la nueva versión del *framework* para más adelante.

### 4.6.1. Diseño

#### 4.6.1.1. Nombre

El nombre de la aplicación web, Wishten, surge de la unión de dos palabras: *wish*, que en inglés significa *deseo* o *desear*; y *ten*, que es el número diez en inglés. La idea que subyace tras este nombre radica en el problema que pretende solucionar la aplicación, es decir, reducir el número de estudiantes que priorizan la calificación de la asignatura frente a la asimilación de los conocimientos. Por tanto, su traducción literal (*desear el diez*), en este caso representa la voluntad del estudiante por aprender todos esos conceptos importantes para lograr extraer su máximo potencial.

#### 4.6.1.2. Logotipo

El logotipo de la aplicación fue creado con la herramienta Canva<sup>25</sup>, una aplicación muy útil para realizar diseño gráfico. El diseño partió del nombre de la aplicación,

---

<sup>25</sup><https://www.canva.com>

Wishten, y se pretendía representar la letra *w* de *wish* y el número diez (*ten*). Se ha buscado un diseño simple y fácilmente reconocible, con un color azul verdoso que no es muy utilizado en otras plataformas similares, lo que hace que destaque más. Este color, además, es bastante versátil ya que puede utilizarse tanto en fondos oscuros, como claros, por lo que facilita su colocación dentro de la aplicación.

En una primera aproximación, se creó un primer logotipo, que se puede ver en la figura 4.16. En este primer logo se utilizó un gradiente de colores azul oscuro y el azul verdoso mencionado anteriormente. Asimismo, se utilizaron colores más oscuros en algunas partes de la letra para dar un efecto tridimensional.



Figura 4.16: Primer logotipo de la aplicación

Sin embargo este primer diseño tenía bastantes problemas, por lo que se propuso realizar un rediseño, cuyo resultado final se presenta en la figura 4.17. En el primer logotipo, la letra *w* se diseñó buscando que en ésta se apreciara también el número diez, sin embargo no se logró el efecto deseado, por lo que se decidió añadir el número a continuación. No obstante, se mantuvo un diseño similar de la letra *w* ya que le aportaba una identidad a la aplicación y se optó por utilizar un color uniforme, ya que es más compatible con distintos tipos de fondo.



Figura 4.17: Logotipo definitivo de la aplicación

### 4.6.1.3. Colores

Para la selección de los colores utilizados en la aplicación web se escogió una paleta de colores proporcionada por un sitio web llamado Color Hunt<sup>26</sup>. Este sitio web ofrece una gran variedad de paletas de colores, por lo que resulta de gran ayuda para desarrollar el diseño gráfico de la aplicación web.

Color Hunt se destaca además de por tener una gran variedad de paletas, por ser muy sencilla de utilizar. No obliga a registrarse y solo se necesita buscar entre el gran número de paletas que tiene aquella que más se ajuste a lo que se necesite.

Todo esto, sumado al conocimiento previo que se tenía sobre Color Hunt hizo que se decidiera utilizar sus paletas para el desarrollo del presente proyecto.

A la hora de buscar la paleta se tenía claro que se querían utilizar colores que no fueran muy claros. La utilización de colores claros en entornos poco iluminados provoca una mayor fatiga y desgaste al observar una pantalla. Por ello, y dado que no se quiere que los usuarios puedan llegar a sentirse molestos por esto, se decidió utilizar una paleta que contara con diferentes tonos de azul y verde.

Estos colores, combinan a la perfección y permiten utilizar tonos claros sin que estos lleguen a ser molestos o perjudiciales para los usuarios de la aplicación. La paleta seleccionada con sus respectivos valores hexadecimales se puede observar en la figura 4.18.

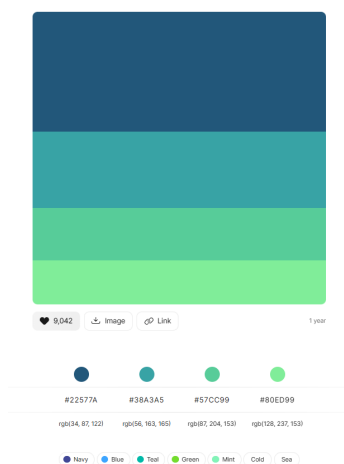


Figura 4.18: Colores principales de la aplicación

De los colores seleccionados, se decidieron usar aquellos con tonos más claros para botones o pequeños iconos de la aplicación y aquellos con tonos más oscuros para el resto de las vistas. Además de los colores presentes en la paleta se decidió usar el rojo para la representación de acciones que consisten principalmente en eliminar, ya que dicho color resulta muy útil a la hora de alertar al usuario del peligro que conlleva realizar dicha acción.

Para poder aplicar dichos colores lo único que se tiene que hacer es copiar el valor hexadecimal de cada color en los archivos css asociados a las vistas.

<sup>26</sup><https://colorhunt.co>

#### 4.6.1.4. Iconos

La correcta selección de los iconos que se van a utilizar en un sitio web es muy importante. Los iconos ayudan a que los usuarios tengan claro que representa cada cosa sin la necesidad de que se les tenga que poner un texto, por ello, es importante que puedan entender qué tipo de icono es, qué representa y a qué va asociado de la manera más rápida y sencilla posible.

Para la selección de los iconos se decidió utilizar Font Awesome<sup>27</sup>, que cuenta con una versión de pago y otra gratuita. Al principio se iban a utilizar imágenes para la representación de dichos iconos, sin embargo, tras investigar un poco y ver diversos proyectos y páginas web, se pudo observar cómo en su mayoría utilizaban los iconos creados proporcionados Font Awesome.

Font Awesome no presenta dependencias con ninguna otra librería, solo necesita que los navegadores cuenten con el soporte *@font-face*, lo que lo hace compatible con la mayoría de los navegadores. Cuenta con un gran número de iconos diferentes disponibles e intuitivos y gracias a la utilización de dichos iconos, el número de imágenes que se utilizan a la hora de desarrollar la aplicación se ven disminuido, lo que afecta positivamente a la velocidad de carga del sitio web (utiliza para crear los iconos una sustitución llevada a cabo por vectores).

Todo lo comentado anteriormente sumado a su licencia libre y a su fácil uso, hizo que se decidiera utilizar sus iconos a la hora de representar las vistas de la aplicación web Wishten.

Para poder utilizar sus iconos primero hay que registrarse y a continuación usar un kit. Un kit es un paquete donde van a estar guardados todos los iconos que se utilizan en la aplicación web. Para usar el kit en proyecto, solo se debe copiar y pegar el enlace al fichero JavaScript que te proporciona el sitio web en la sección correspondiente. Esto se puede ver en la figura 4.19.

```
{{-- Scripts --}}
<script async src="{{ url('/js/dropdown.js') }}"></script>
<script async src="https://kit.fontawesome.com/0d34bde1b9.js" crossorigin="anonymous"></script>
@yield('js')
```

Figura 4.19: Carga del fichero de Font Awesome en el código de la aplicación

Una vez se ha obtenido el kit, lo único que queda por hacer es buscar el icono que se desee, copiarlo (Font Awesome proporciona la línea de código) y pegarlo en la vista que se desee utilizar. Hay que tener claro cuál es el nombre que tiene dicho icono para poder modificar la forma en la que este se ve o coloca desde el correspondiente css asociado a la vista en la que se encuentre. La línea de código que te proporciona para representar cada icono se puede ver en la figura 4.20.

Si bien dicho icono se podría utilizar junto al elemento `<span>` en HTML5, cómo la línea de código se te proporciona directamente, se decidió dejar el uso del icono junto con el elemento `<i>`. Se puede cambiar dicha línea de código en función del entorno en el que se vaya a utilizar el icono, ya que Font Awesome ofrece también

<sup>27</sup><https://fontawesome.com>

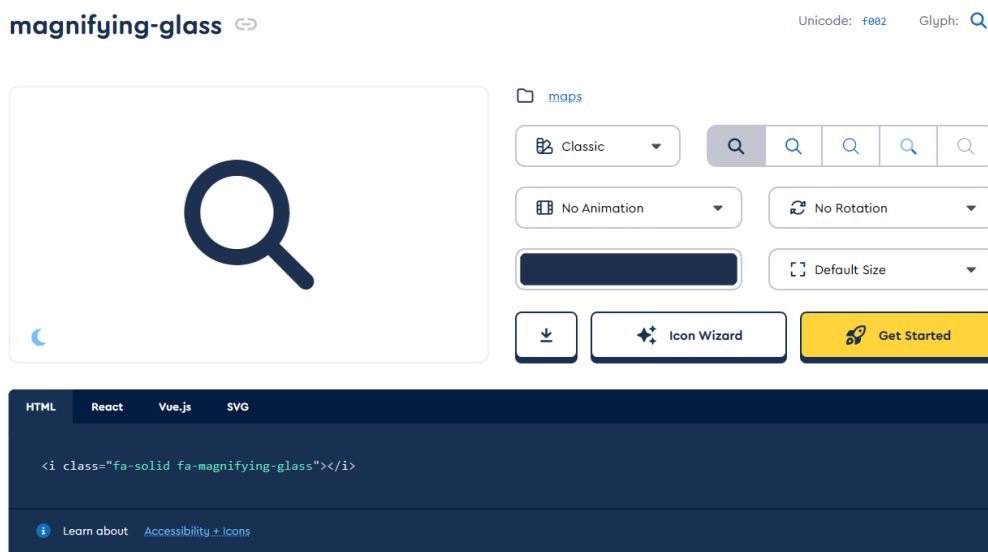


Figura 4.20: Código HTML proporcionado por Font Awesome para incrustar el icono en las vistas

las opciones para *react* o *vue.js*. Sin embargo, como para el desarrollo de Wishten no se utilizan estas herramientas para representar las vistas, no se usaron dichas opciones.

#### 4.6.1.5. Estilo

Para realizar las vistas de la página web se decidió desde un principio utilizar HTML5 y CSS puros. En primera instancia, se valoró utilizar Bootstrap<sup>28</sup> para agilizar el proceso de creación de las vistas y poder centrarse en las diversas funcionalidades, sin embargo al final no se utilizó por diversos motivos.

Uno de los principales motivos por los que no se utilizó Bootstrap es debido a una de sus mejores características: los componentes y estilos predefinidos. Bootstrap es una *toolkit* muy utilizada que ofrece una gran cantidad de componentes, y estilos listos para usar, los cuales hubieran agilizado el proceso de desarrollo de Wishten. No obstante, el hecho de que dichas plantillas estuvieran presentes en numerosas páginas web hizo que se optara por no utilizar Bootstrap y buscar un diseño diferenciador.

Tras investigar el diseño de algunas aplicaciones como TikTok<sup>29</sup>, Instagram<sup>30</sup> u otras redes sociales conocidas, se pudo observar cómo todas ellas usaban vistas muy simples con pocos botones que se encargan de mostrar, mediante ventanas modales, la diferente información asociada a ellos (acciones de editar, eliminar, crear, etc.).

Todas las plantillas de Bootstrap que se examinaron estaban orientadas a mostrar toda la información al usuario y, si bien visualmente era muy atractivo, no era exactamente lo que mejor se adaptaba a este proyecto.

<sup>28</sup><https://getbootstrap.com>

<sup>29</sup><https://www.tiktok.com>

<sup>30</sup><https://www.instagram.com>

La aplicación que se pretende desarrollar, busca que los usuarios puedan encontrar vídeos que les sean útiles y ofertas de empleo o de prácticas en empresas. Por ello, lo que se busca es que la información que se le muestre al usuario sea mínima y lo más informativa posible. Con esto se pretende que la mayor parte del tiempo que los usuarios pasen en la aplicación sea viendo el vídeo o la oferta que les interese.

Esta forma de mantener a los usuarios en la aplicación es muy utilizada en las redes sociales y por ello se han usado éstas como ejemplo a seguir de cara a estructurar y definir las vistas de la aplicación.

Otro de los motivos por los que Bootstrap es tan utilizado es porque facilita mucho la adaptación de las vistas en diferentes tamaños de pantalla. Esto resulta muy útil ya que una aplicación web puede ser usada en un dispositivo móvil, ordenador, *tablet*, etc. Sin embargo, la presente aplicación no busca estar orientada a otro dispositivo que no sea un ordenador. Eso no quiere decir que no se hayan adaptado las vistas a los demás dispositivos, si no que, al no ser un motivo de tanto peso, no se consideró una ventaja dicha utilidad de Bootstrap para la representación de las vistas de este proyecto.

Si bien es cierto que todo lo que te ofrece Bootstrap es modificable, éste no tenía las características que se buscaban, además ya se tenían conocimientos previos sobre el manejo de HTML5 y CSS. Por ello, al no utilizar plantillas predefinidas, se contaba con total libertad para crear las vistas. Por todo ello y por los motivos descritos anteriormente se decidió al final no utilizar Bootstrap ni ningún otro *framework* de *frontend*.

La representación de las vistas usando HTML5 y CSS ha sido muy variada en Wishten, a pesar de que existen algunos componentes y estilos comunes a todas ellas y que son importantes de explicar para entender su estructura y representación.

Para estructurar los elementos de la aplicación, se han creado ficheros CSS (en */public/css*) desde los que se define el estilo de cada página. Como cada página muestra elementos diferentes y tiene una estructura diferenciada, se ha creado prácticamente un fichero CSS por cada vista. De esta manera se puede ajustar más el estilo al contenido específico de cada página.

Dentro de cada vista, existen elementos que comparten estilo, como botones, *inputs* de formularios o selectores. Por tanto, para aplicarles el mismo estilo a todos, se han utilizado selectores de clase, en los que se establece el color, el tamaño y el resto de características. Por otro lado, para los elementos únicos, se utilizan identificadores (ID), que son únicos en todo el código HTML de la vista.

Estos componentes se agrupan dentro de contenedores, para los que se utiliza la etiqueta *<div>* y se define su disposición en las hojas de estilo. Principalmente se ha utilizado el diseño *flex* (flexible), que dispone los elementos uno detrás de otro, de forma vertical u horizontal. Esto se ha utilizado tanto para alinear los campos de los formularios, como para mostrar las listas de vídeos y ofertas en las diferentes vistas.

También se ha utilizado un diseño *grid*(cuadrícula) para las vistas en las que se muestra una gran cantidad de elementos, ya que este tipo de diseño dispone los

elementos en una cuadrícula, es decir, el grupo crece tanto vertical como horizontalmente. Dicha disposición se utiliza, por ejemplo, para mostrar los resultados de búsqueda de la sección de vídeos de la aplicación.

Además, se han utilizado las hojas de estilo para lograr que la aplicación sea responsiva, es decir, que el contenido que se muestra se modifique en función del tamaño de la pantalla. Para ello, se ha usado la regla *@media*, en la que se definen condiciones, como el ancho o el alto de la pantalla, que modifican la disposición o el tamaño del contenido. Por ejemplo, tal como se ve en la figura 4.21, se establece la anchura mínima necesaria para pasar de una disposición *flex* vertical (*column*) a una horizontal (*row*).

```
@media (min-width: 100px) {
  .video-view {
    flex-direction: column;
  }
  .edit-info {
    align-items: center;
  }
}

@media (min-width: 1200px) {
  .video-view {
    flex-direction: row;
    align-items: center;
  }
  .edit-info {
    align-items: flex-start;
  }
}
```

Figura 4.21: Ejemplo de uso de *@media* en el CSS de la aplicación

#### 4.6.1.6. JavaScript

Para implementar diversos aspectos visuales y generar código HTML de manera dinámica dentro de la aplicación, se ha hecho uso de ficheros JavaScript (en */public/js*). A continuación se explican los principales *scripts* que se han desarrollado para este proyecto.

En primer lugar, para implementar la lógica para mostrar los cuestionarios dentro de los vídeos, se crearon dos scripts: *showQuiz.js* y *answerQuiz.js*. El primero hace visible las respuestas por encima del vídeo y lo pausa, además de reanudar el vídeo y esconder la pregunta cuando esta ya ha sido respondida. El segundo se encarga de crear *inputs* escondidos que contienen el *id* de las respuestas seleccionadas y que se enviarán cuando el usuario pulse en “guardar resultados” mediante una petición HTTP.

Por otro lado, para mostrar al usuario las estadísticas de cada pregunta del vídeo se utilizó la librería *Chart.js*<sup>31</sup>, que permite crear gráficos en JavaScript de manera sencilla.

<sup>31</sup><https://www.chartjs.org>

También se ha empleado código JavaScript para algunos detalles más simples: un botón para mostrar la contraseña y que el usuario compruebe si la ha escrito correctamente, un filtro de tablas HTML, ventanas modales que se superponen y algunos efectos visuales.

#### 4.6.1.7. Licencia utilizada

El código de este proyecto ha sido desarrollado bajo la licencia GNU Affero General Public License v3.0 (AGPL), la cual permite su modificación y distribución por terceras personas, ajenas al desarrollo del presente proyecto. Todo ello con la condición de que se distribuya bajo la misma licencia. Se ha seleccionado la AGPL frente a la GPL, ya que ésta es más utilizada para desarrollos software que se van a ejecutar en un servidor web.

### 4.6.2. Lógica

El *framework* que se usa en esta propuesta, Laravel, está basado en el patrón modelo-vista-controlador (MVC, un patrón de arquitectura software que separa los datos y la lógica de su visualización) para gestionar la aplicación web. En la figura 4.22 se puede observar un diagrama que se ha elaborado para explicar cómo se conectan estos elementos dentro de una aplicación desarrollada en Laravel.

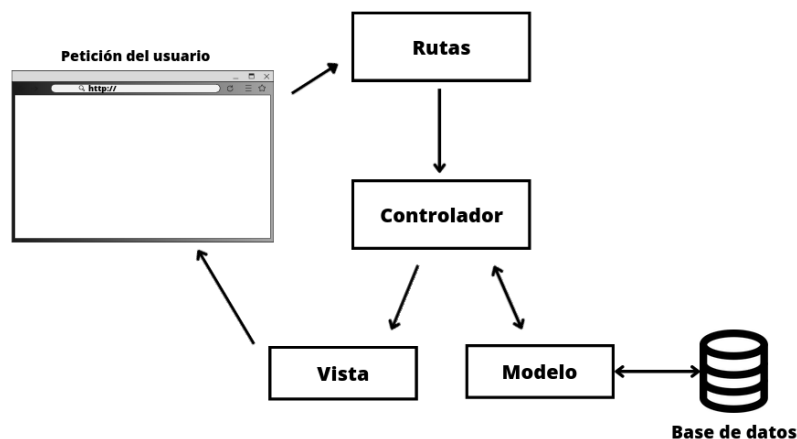


Figura 4.22: Arquitectura de una aplicación en Laravel

#### 4.6.2.1. Gestión de la base de datos

La base de datos se gestiona desde el propio framework, utilizando un *object-relational mapper* (ORM, un tipo de herramienta que permite realizar consultas a la base de datos usando programación orientada a objetos) llamado Eloquent, que permite realizar todas las operaciones sobre la base de datos utilizando únicamente clases de PHP. Las tablas de la base de datos se crean a partir de migraciones

(*/database/migrations*), que son ficheros PHP que especifican los atributos de la tabla y el ORM se encarga de traducirlo a SQL (Laravel, 2023b).

Las relaciones entre tablas se gestionan, por tanto, utilizando los modelos y creando funciones de diferentes clases según el tipo de relación. De esta manera, una vez se crean las migraciones y se establecen dichas relaciones entre los modelos, Eloquent se encarga de actualizar la base de datos a medida que se introducen cambios desde la aplicación.

Además las tablas que relacionan los modelos principales (usuarios, vídeos, preguntas. . .) se gestionan en Laravel mediante relaciones entre los modelos. Por tanto no es necesario crear un modelo ni un controlador para gestionar dichas tablas intermedias (*visualized\_videos*, *follows*, *user\_answer*...), ya que estas se manejan a través de los modelos principales.

#### 4.6.2.2. Modelo Vista Controlador

En cuanto a la estructura de MVC, los modelos (*/app/Http/Models*) son clases de PHP que se corresponden con las tablas principales de la base de datos. Por tanto, utilizando estas clases, se consultan y modifican los datos de cada tabla.

En segundo lugar están las vistas (*/resources/views*), que utilizan un sistema de plantillas llamado Blade (Laravel, 2023a). Este sistema permite crear secciones en el código HTML, dejar huecos que se rellenan con llamadas a funciones y añadir bucles y condicionales para gestionar lo que se muestra en una página.

En tercer lugar, están los controladores (*/app/Http/Controllers*), que son los que gestionan todas las peticiones mediante funciones que actualizan la base de datos, haciendo uso de los modelos, y redireccionan a las vistas correspondientes.

#### 4.6.2.3. Rutas

Para que el cliente envíe peticiones al servidor, existen las rutas, que se gestionan desde el fichero */routes/web.php*. En él se definen los nombres de las rutas a las que acceden los usuarios y las acciones que realizan (GET, POST, etc.). Las rutas se pueden agrupar en torno a un controlador que gestiona todas ellas y de esta manera se declara una ruta por cada función del controlador.

Además, para restringir ciertas acciones, se utilizó un *middleware* (Laravel, 2023e), que es un mecanismo para filtrar las peticiones HTTP. Se utilizó el *middleware Authenticate*, proporcionado por Laravel para controlar las rutas que requieren la autenticación previa de los usuarios. Se creó también un *middleware* propio (*UserRoles*) para controlar el acceso a las rutas que requieren un rol específico.

#### 4.6.2.4. Políticas de acceso

Para restringir el acceso de los usuarios a la realización de determinadas acciones sobre los modelos, se utilizaron también las políticas de autorización de Laravel.

Para ello se creó una clase para cada modelo donde se definen las acciones que puede realizar un usuario como funciones (*create*, *update*, *delete*...).

En general, los usuarios con el rol de administrador son los únicos autorizados para crear, actualizar y eliminar cualquier modelo. Por su parte, al resto de usuarios se les permite únicamente realizar cambios sobre los modelos que les pertenezcan, por ejemplo, su propio perfil o los vídeos que hayan subido a la aplicación.

Las clases que representan las políticas de cada modelo se almacenan en el directorio `/app/Polícies` y dentro de los métodos de los controladores se comprueba si el usuario que realiza la petición puede o no realizar la acción solicitada.

#### 4.6.2.5. Validación

En cuanto a la validación de los campos que introducen los usuarios a través de los formularios, se utilizan clases personalizadas (en `/app/Http/Requests`), en las que se especifican los campos requeridos y lo que tienen que cumplir (tipo de campo, longitud, formato de ficheros...).

Además, para proteger la aplicación de ataques de tipo Cross-Site Request Forgery (CSRF, un ataque que se produce cuando se ejecutan acciones no autorizadas desde un usuario en el que confía la aplicación), en todos los formularios de Laravel se incluye además un token especial (`@csrf` en el código Blade) que confirma que es el usuario el que ha hecho la petición en cuestión desde la aplicación.

#### 4.6.2.6. Almacenamiento

En relación a la gestión de los ficheros que se suben al servidor, éstos se almacenan en el directorio `/storage/app`, dentro del cual pueden ser públicos o privados, teniendo una carpeta *public* o *private* según el caso. Los ficheros que se quieran visualizar en la página deben estar en la carpeta `/public`, por tanto se debe crear un enlace simbólico desde esta carpeta a la carpeta `/storage/app/public` para poder crear una ruta a ese fichero dentro de la aplicación (Laravel, 2023c).

### 4.6.3. Funcionalidades

Para cumplir con las necesidades de los usuarios, se han implementado diversas funcionalidades que gestionan cada aspecto de la aplicación web.

#### 4.6.3.1. Gestión de usuarios

El proceso de *login* y registro de usuarios se gestiona mediante el controlador *AuthController*, que contiene las funciones para validar los datos de entrada, reportar los errores correspondientes y crear y eliminar la sesión cuando se hace *logout*. Para estas acciones se utilizan de las clases *Auth* y *Sessions* de Laravel, que implementan la autenticación y la gestión de sesiones, respectivamente. Las vistas utilizadas son

*login* (figura 4.23) y *registration* (figura 4.24), que muestran los formularios de inicio de sesión y registro.

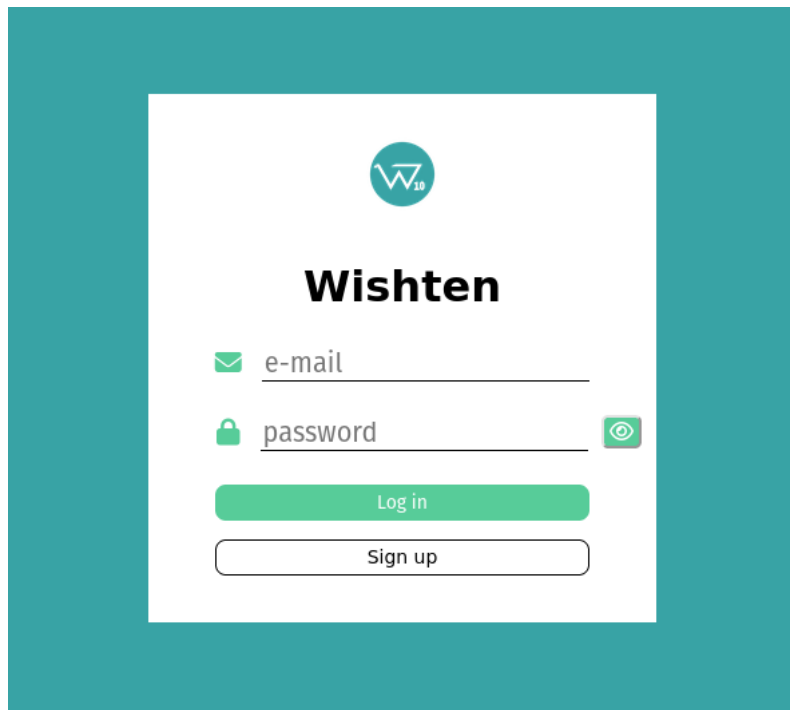
The image shows a login form for 'Wishten' centered on a teal background. At the top is the Wishten logo, a teal circle with a white 'W' and '10'. Below the logo is the title 'Wishten' in bold black text. There are two input fields: 'e-mail' with a green checkmark icon on the left, and 'password' with a green lock icon on the left and a green eye icon on the right. Below these fields are two buttons: a green 'Log in' button and a white 'Sign up' button with a black border.

Figura 4.23: Vista *login*

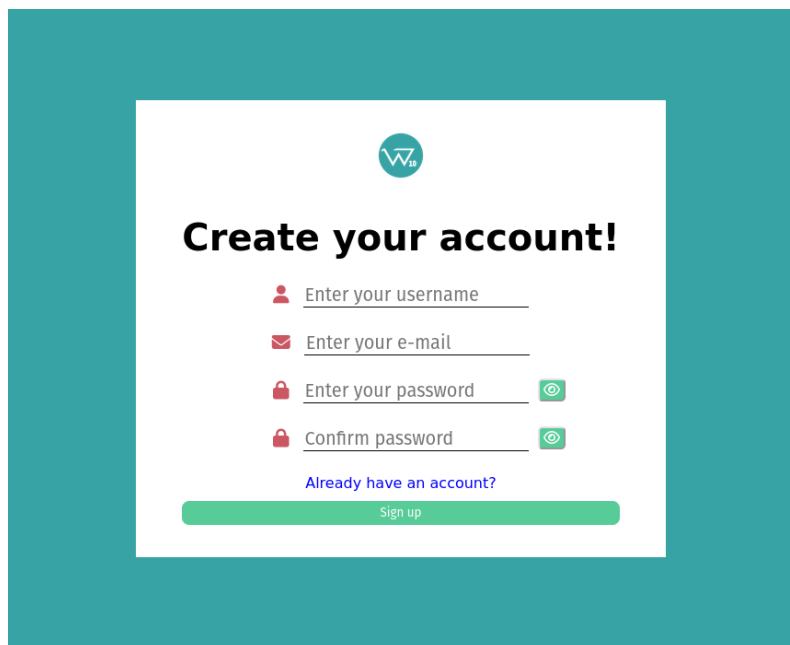
The image shows a registration form for 'Wishten' centered on a teal background. At the top is the Wishten logo, a teal circle with a white 'W' and '10'. Below the logo is the title 'Create your account!' in bold black text. There are four input fields: 'Enter your username' with a red person icon on the left, 'Enter your e-mail' with a red envelope icon on the left, 'Enter your password' with a red lock icon on the left and a green eye icon on the right, and 'Confirm password' with a red lock icon on the left and a green eye icon on the right. Below these fields is a blue link 'Already have an account?' and a green 'Sign up' button.

Figura 4.24: Vista *registration*

Para gestionar la información de los usuarios se utiliza el controlador *UserController*, que contiene las funciones para actualizar los atributos del usuario. Los usuarios pueden acceder a la vista *profile*, tal como se ve en la figura 4.25, para modificar su

nombre, dirección *email* y foto de perfil. Las fotos de perfil se almacenan en */storage/app/public/profile\_pics* con un nombre aleatorio asignado por la aplicación y la ruta se almacena en un campo de la tabla *users*.

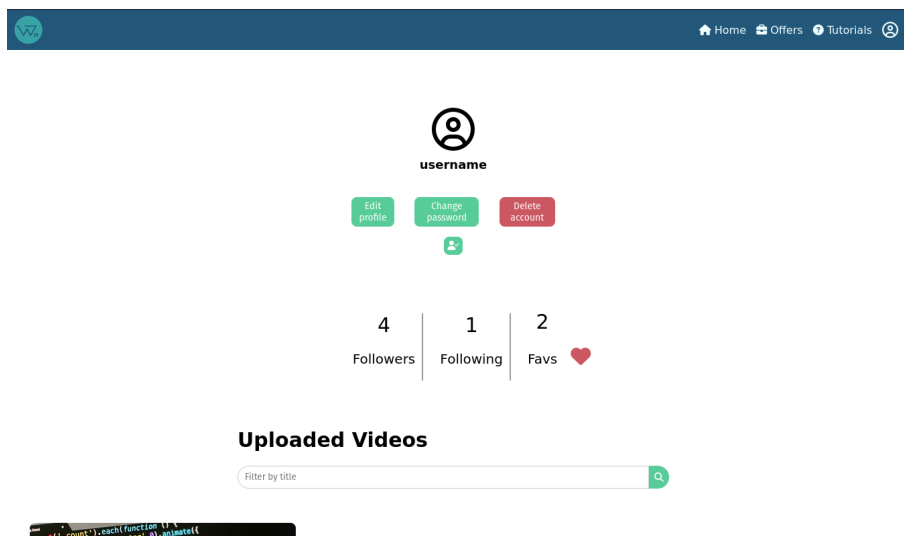


Figura 4.25: Vista *profile*

Por otro lado, un usuario puede también ver los perfiles de otros usuarios desde la vista *profile*, accediendo a través de los vídeos que éstos hayan subido. En esta vista, además de la información del usuario consultado, aparece un botón para seguir al usuario.

En la vista del perfil se muestran también el número de seguidores, seguidos y el número total de favoritos, sumando todos los vídeos del usuario. Adicionalmente, se puede consultar las listas de seguidores y seguidos, que se mostrarán en la vista *follows*, que se puede ver en la figura 4.26.

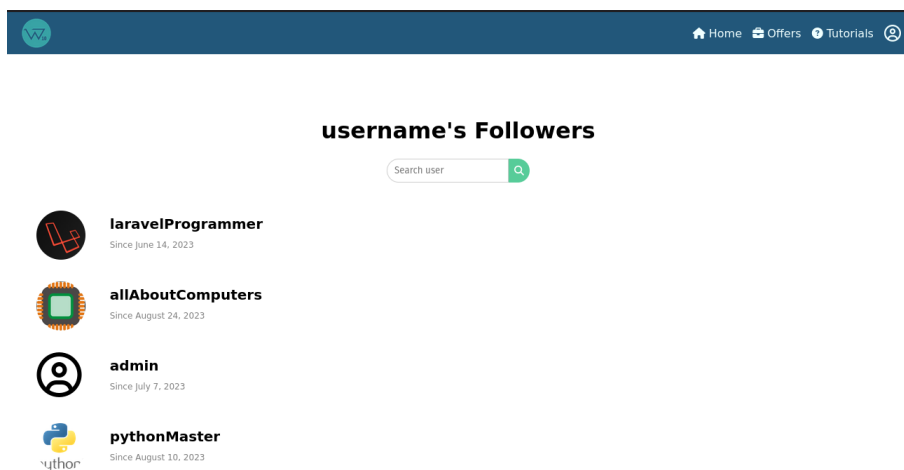


Figura 4.26: Vista *follows*

Los administradores gestionarán los usuarios de la aplicación desde la vista *adminUsers*, accesible a través de la página de administración. Desde ahí podrán añadir

nuevos usuarios, eliminar cuentas o modificar la información de éstos, como se ve en la figura 4.27.

Los administradores no pueden modificarse a sí mismos desde la página de administración, puesto que para ello ya utilizarán la vista de perfil. Sin embargo sí aparecen en la tabla de usuarios, ya que en ésta deben figurar todos los usuarios de la aplicación, salvo que en este caso, los botones de acción están deshabilitados para su propia cuenta de usuario.

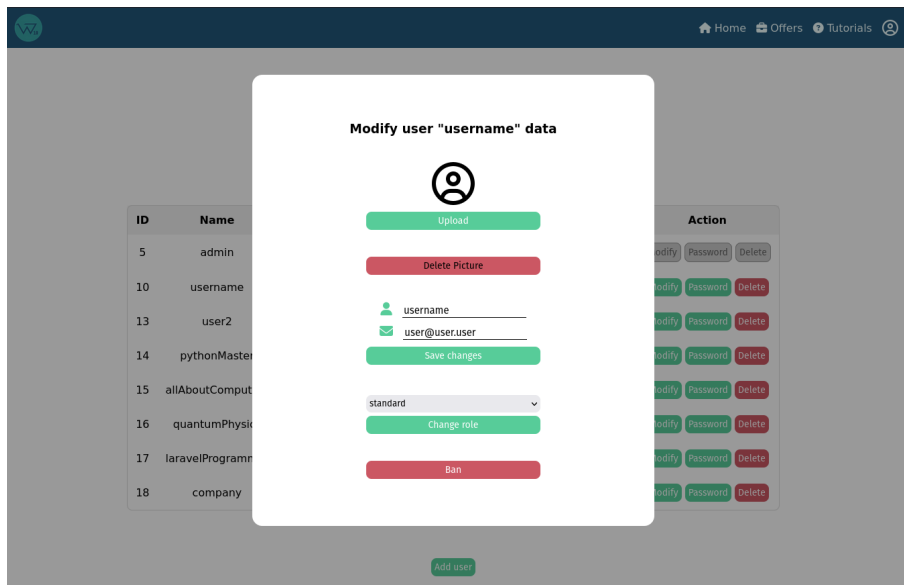


Figura 4.27: Vista *adminUsers*

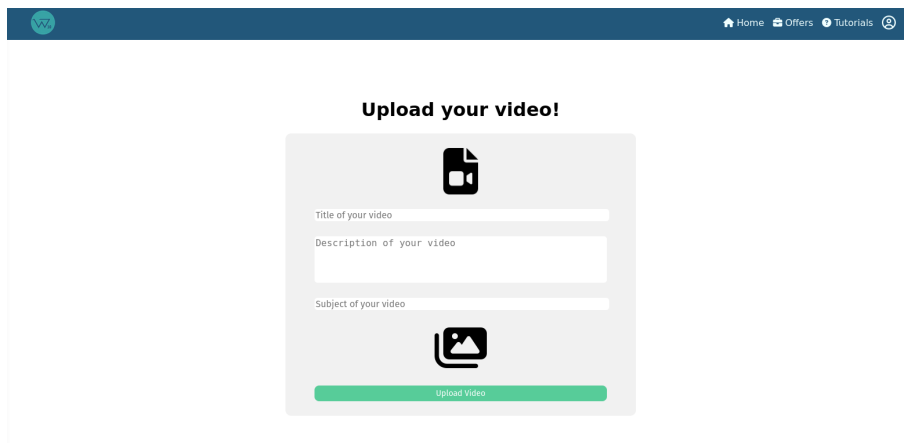
#### 4.6.3.2. Gestión de vídeos

Todas las funcionalidades para crear, editar y eliminar los vídeos se gestionan desde el controlador *VideoController*.

Los vídeos pueden ser subidos por cualquier usuario registrado desde la vista *newVideo*, tal como se muestra en la figura 4.28. Para crear un nuevo vídeo, debe incluirse el propio archivo de vídeo, un título, una descripción, un tema y un archivo de imagen que sirve como miniatura. Por tanto, los usuarios no podrán incluir contenidos de otras plataformas, sino que tendrán que subir el archivo desde su equipo. Esto se debe a que se pretende que los contenidos subidos a esta aplicación web sean originales, además de evitar los problemas de propiedad intelectual que conlleva permitir a los usuarios incrustar vídeos de otras plataformas.

En cuanto a los temas de los vídeos, aquellos que ya se hayan utilizado, se almacenan en la base de datos y aparecen como sugerencia, pero el usuario siempre podrá añadir el tema que quiera y éste se guardará de la misma manera. De esta forma, los propios usuarios propondrán los temas de los vídeos, siendo más útiles para los usuarios con los mismos intereses.

Una vez subido el vídeo, el usuario puede decidir añadir preguntas interactivas, lo que mejorará la experiencia de aprendizaje de los usuarios.

Figura 4.28: Vista *newVideo*

La información de los vídeos se almacena en la base de datos, donde además se guardan las rutas a los ficheros de vídeo e imagen. Por su parte, el archivo de vídeo y la imagen para la miniatura se guardan en la carpeta `/storage/app/public/videos`.

Se utiliza una estructura de directorios para almacenar ordenadamente los ficheros que consiste en agrupar los vídeos del mismo tema en la misma carpeta, de modo que se tiene subcarpetas con el nombre de cada vídeo más un identificador formado a partir de la fecha exacta de subida (para el caso de que hubiese varios vídeos con el mismo título). Dentro de cada subdirectorio habría dos archivos, uno de vídeo y otro de imagen, correspondiente a la miniatura. Los nombres de estos archivos se generan de la siguiente forma:

- vídeo: `wishten-{identificador fecha}-{id del vídeo}.{extensión}`
- miniatura: `wishten-{identificador fecha}-{id del vídeo}-thumbnail.{extensión}`

Para generar el nombre de los ficheros se utiliza un identificador generado con la fecha para corroborar su correspondencia con el directorio en el que se encuentra, por lo que se puede saber si los archivos han sido modificados o reubicados donde no corresponde. También se utiliza el *id* del vídeo, que es único y que, al buscarlo en la base de datos, debe coincidir el título del vídeo con el que aparece en el nombre del directorio. De esta manera se puede verificar que el contenido almacenado se corresponde con la información almacenada en la base de datos.

De esta manera, los ficheros se estructurarían automáticamente tal y como se muestra en la figura 4.29.

Mediante esta forma de organizar los ficheros, éstos son más fáciles de acceder por parte de los administradores, en caso de existir algún problema con ellos que no se pudiera gestionar desde la página de administración de la propia aplicación.

En cuanto a la búsqueda de vídeos, los usuarios registrados contarán con una vista general desde la página principal que se puede ver en la figura 4.30. En esta vista encontrarán los vídeos organizados en varias listas, ordenados según el número de visitas, el número de usuarios que lo guardaron en favoritos y la fecha de subida.

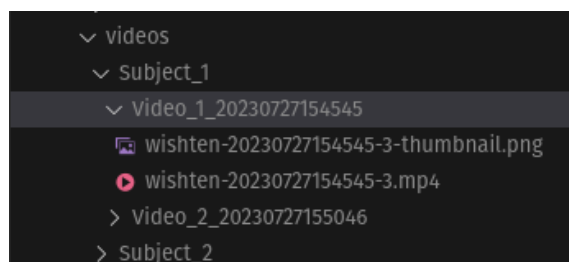


Figura 4.29: Estructura de almacenamiento de los vídeos

Además, se incluye una lista con los vídeos que contienen cuestionarios y otra lista por cada usuario seguido en la que se muestran los últimos vídeos de cada uno.

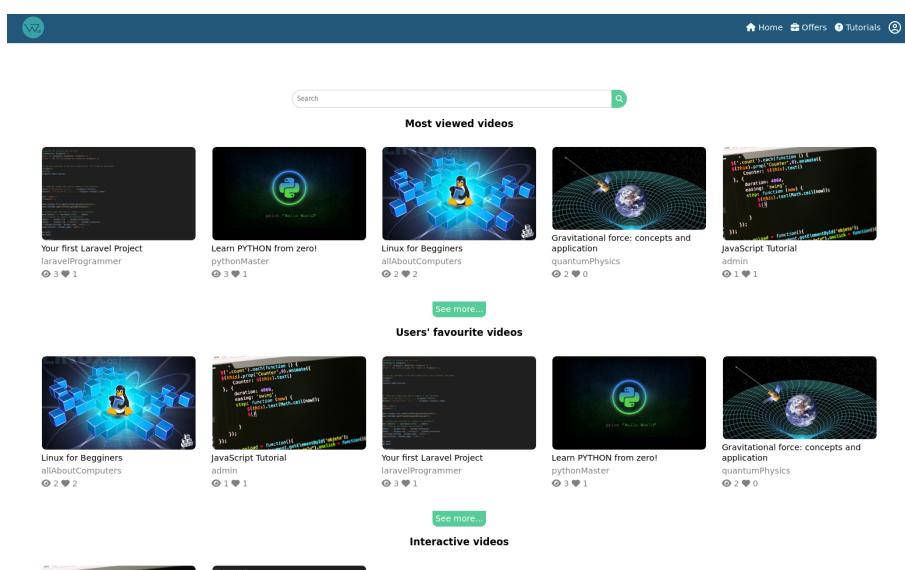


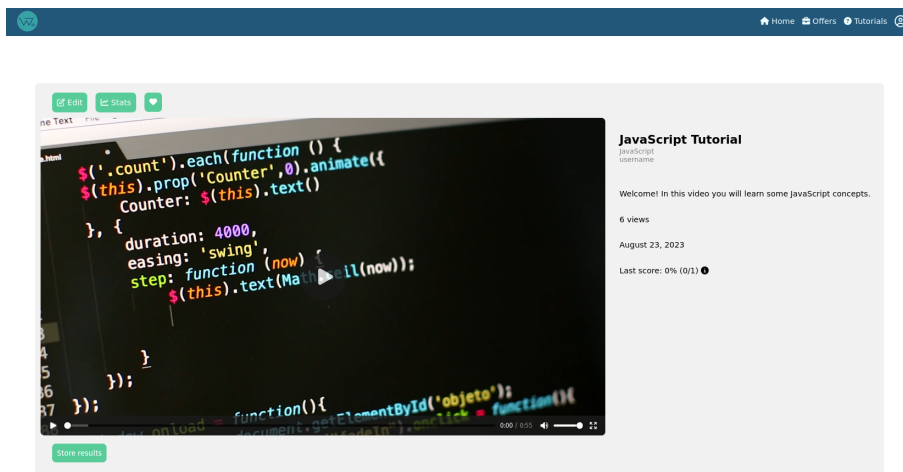
Figura 4.30: Vista *home*

En esta vista, los usuarios también disponen de una barra de búsqueda mediante la que pueden buscar los vídeos por temas.

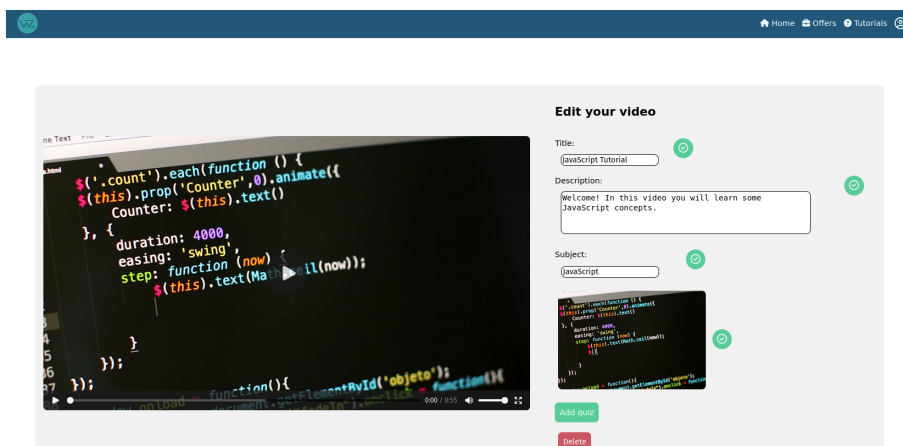
Cuando un usuario selecciona un vídeo, éste se muestra en la vista *videoWatch* (figura 4.31), donde aparece el vídeo, junto con la información del mismo (título, descripción, fecha de subida...). Desde esta vista puede también añadir el vídeo a favoritos, y, si es el propietario del vídeo, tendrá acceso a editar el vídeo o a visualizar las estadísticas de éste.

Si el vídeo contiene preguntas incrustadas, el usuario podrá ver la última puntuación que consiguió, es decir, el número de preguntas que acertó. Además debajo del vídeo tiene un botón para almacenar las respuestas que haya dado a las preguntas, una vez finalice de ver el vídeo. En caso de que el usuario no desee guardar sus resultados, simplemente podrá salir del vídeo y continuar navegando por la página.

Si el usuario es el propietario del vídeo o un administrador, podrá acceder a la vista *videoEdit* que, como se muestra en la figura 4.32, contiene formularios para editar toda la información del vídeo, a excepción del propio fichero de vídeo. Esto se debe a que el fichero de vídeo es el elemento principal, al que van asociados tanto

Figura 4.31: Vista *video Watch*

el título, descripción y tema, como las preguntas embebidas. Por ello si un usuario quiere modificar el fichero de vídeo, lo que tendrá que hacer es subir ese fichero como un vídeo nuevo, lo que garantiza la coherencia de todos esos elementos.

Figura 4.32: Vista *video Edit*

Desde esta vista, el usuario también puede eliminar el vídeo y acceder a la vista para añadir cuestionarios, que se explicará en el siguiente apartado.

Los usuarios con el rol de *admin* pueden gestionar todos los vídeos que se suben desde la página de administración de vídeos (figura 4.33). Desde esta vista, los administradores tienen acceso a todos los vídeos subidos en la aplicación que aparecen en forma de tabla, al igual que el resto de páginas de administración.

Aquí pueden modificar todos los aspectos del vídeo y, en caso de haber alguna irregularidad, pueden poner el vídeo en estado de revisión o bloquearlos para que no aparezcan en las búsquedas y, si fuera necesario, eliminarlo.

Los vídeos que no tengan estado válido, es decir, que estén pendientes de revisión o bloqueados, no se mostrarán al público general. Por tanto estos vídeos sólo serán

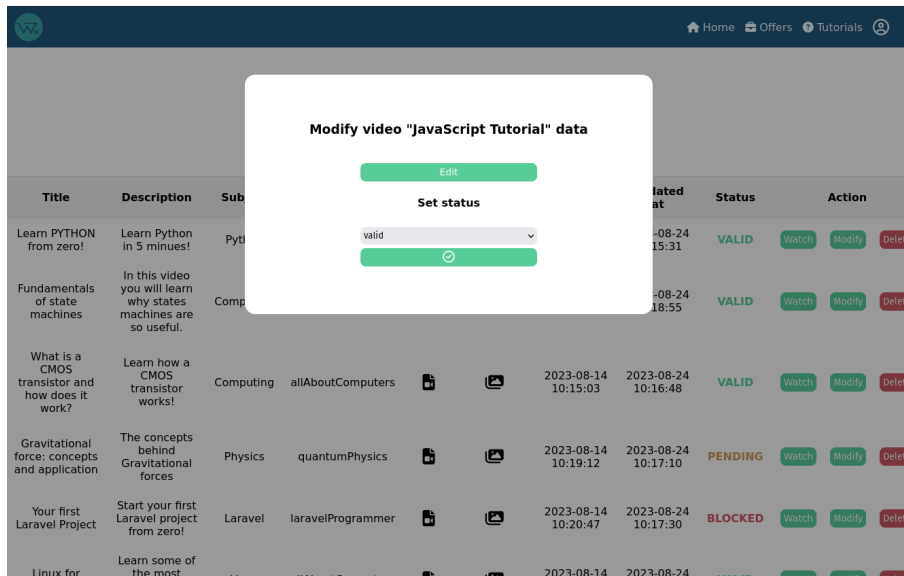


Figura 4.33: Vista *adminVideos*

accesibles para el dueño del vídeo o los administradores, mostrando un aviso en la parte superior, como se observa en la figura 4.34.

De esta manera, se previene que existan vídeos en la página que no sean adecuados o que no tengan nada que ver con los objetivos que se proponen con el desarrollo de esta aplicación.

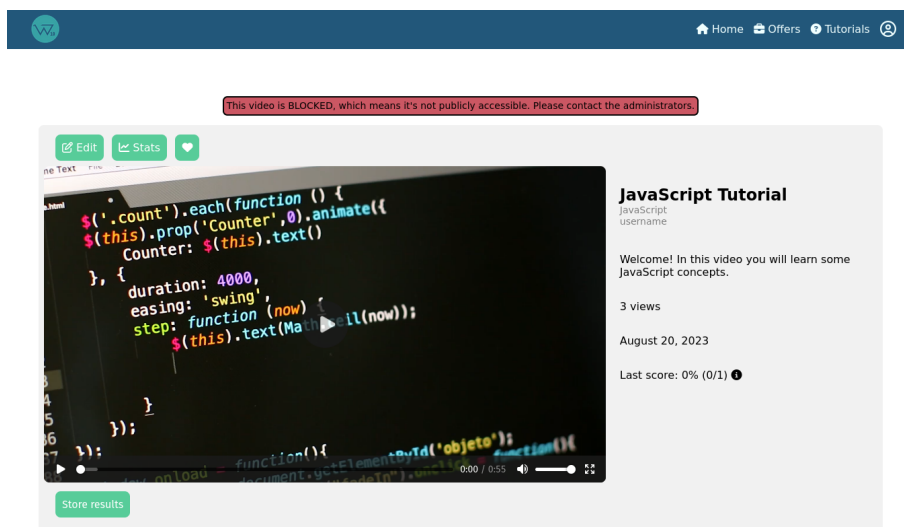


Figura 4.34: Vista *videoWatch* de un vídeo bloqueado

#### 4.6.3.3. Gestión de cuestionarios

Para gestionar todas las funcionalidades relacionadas con los cuestionarios, se utiliza el controlador *QuizController*, que contiene las funciones para crear, modificar y eliminar preguntas y respuestas para los vídeos.

Cuando un usuario crea un vídeo, tiene la opción de añadir cuestionarios dentro de él para mejorar el aprendizaje del espectador. Los cuestionarios se almacenan en forma de preguntas con respuestas cerradas, definidas por el usuario e incrustadas en un instante del vídeo. De esta manera, es el creador del vídeo quien decide el momento en el que aparecerán las preguntas y las opciones para responderlas.

Si el usuario lo desea, puede utilizar también las preguntas a modo de anotaciones, lo único que tiene que hacer es no añadir ninguna respuesta a la pregunta. Por tanto, el usuario también tiene la posibilidad de complementar la información del vídeo o de resumir algunas partes mediante dichas anotaciones.

Para añadir y modificar las preguntas dentro de un vídeo, se utiliza la vista *addQuiz*, en la que, como se ve en la figura 4.35, se muestra el vídeo en cuestión y un formulario para añadir las preguntas. El usuario seleccionará el instante en el que quiere que aparezca la nueva pregunta, añadirá el texto y las opciones posibles, y establecerá las respuestas correctas a la pregunta.

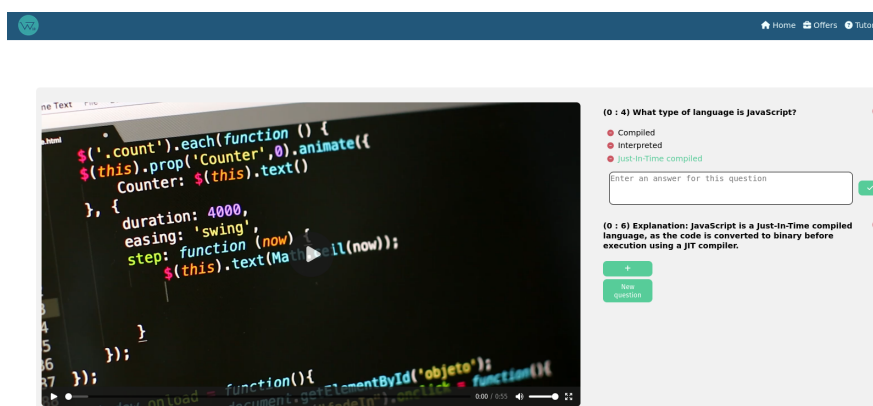


Figura 4.35: Vista *addQuiz*

En cada pregunta se muestra el minuto en el que aparece y además están ordenadas por orden de aparición. Esto facilita al usuario la tarea de añadir preguntas, ya que puede observar en el vídeo en qué momento aparece cada pregunta y el orden de las mismas.

Por otro lado, para añadir las respuestas se le muestra un botón con el signo más y éstas se acumulan debajo del texto de la pregunta correspondiente. Para establecer las respuestas correctas, el usuario sólo tiene que hacer clic sobre ellas y éstas se marcarán en verde, indicando que son correctas. Si hace clic sobre una respuesta correcta, ésta se marcará como incorrecta, a no ser que sea la única, ya que debe haber al menos una respuesta correcta.

Para eliminar tanto preguntas como respuestas, se le muestra un botón rojo con el signo menos (en contraposición con el signo más para añadir) junto a la pregunta o respuesta que desee eliminar.

Cuando un usuario responde una pregunta, obtiene *feedback* inmediato, tal como se muestra en la figura 4.36 y puede saber si sus respuestas son correctas mediante un código de colores: se marca en verde las respuestas correctas y en rojo las respuestas

incorrectas que haya dado el usuario. Además el usuario puede almacenar los resultados obtenidos en ese vídeo, así como ver su último resultado y volver a contestar el cuestionario para afianzar los conocimientos obtenidos en la visualización del vídeo.

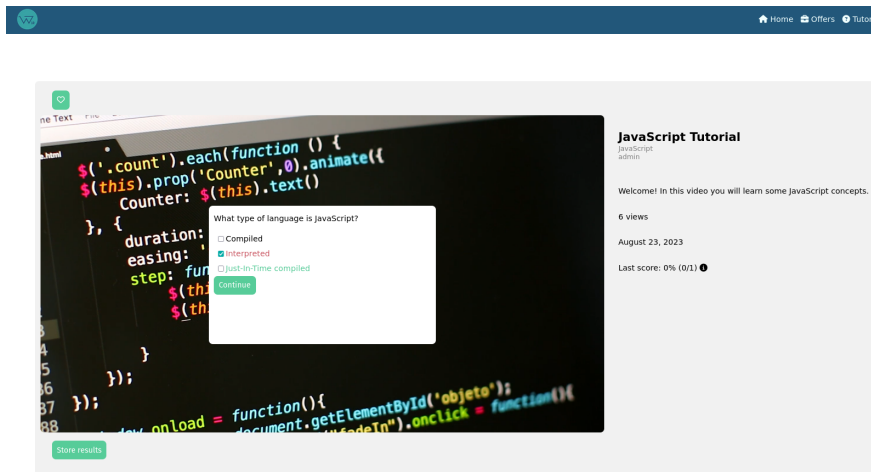


Figura 4.36: Vista *videoWatch* - Respondiendo una pregunta

De la misma manera, el usuario que crea un vídeo puede ver la recepción que éste ha tenido, analizando los datos recopilados en las visualizaciones del mismo, presentados en la vista *videoStats* (figura 4.37). El usuario puede visualizar cuánta gente ha visto el vídeo, cuántos lo han guardado en favoritos y, si contiene preguntas embebidas, puede ver todas las respuestas que han dado los usuarios y así determinar si la explicación que da es lo suficientemente clara.

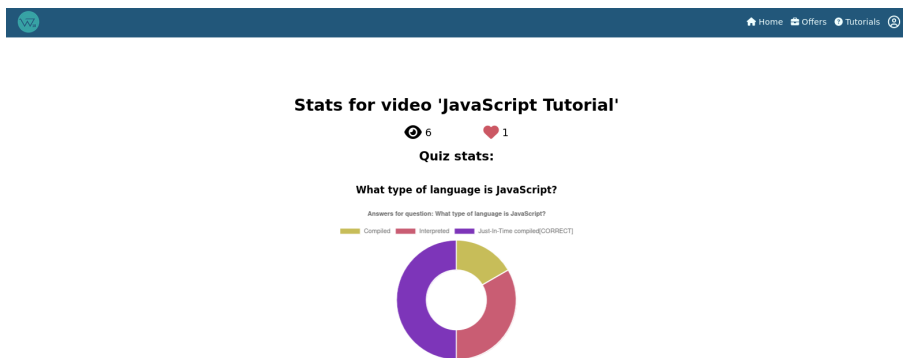


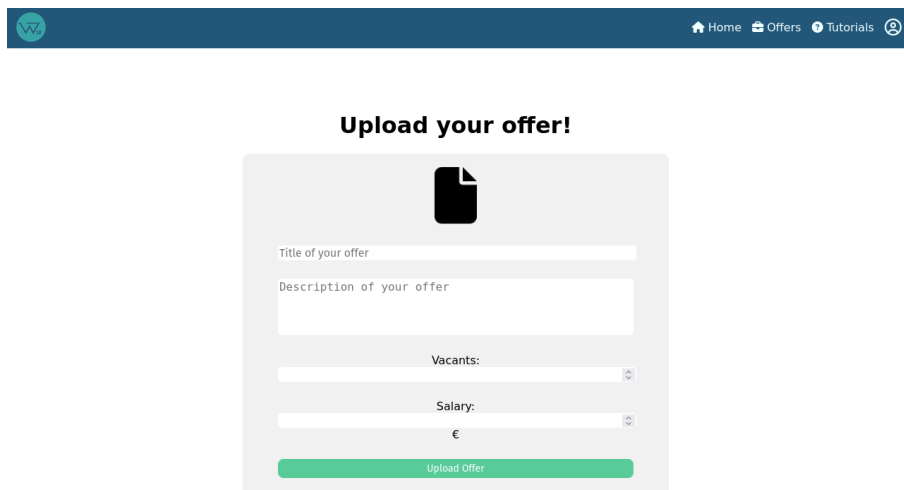
Figura 4.37: Vista *videoStats*

#### 4.6.3.4. Gestión de ofertas

Todas las funcionalidades para crear, editar y eliminar las ofertas se gestionan desde el controlador *OfferController*.

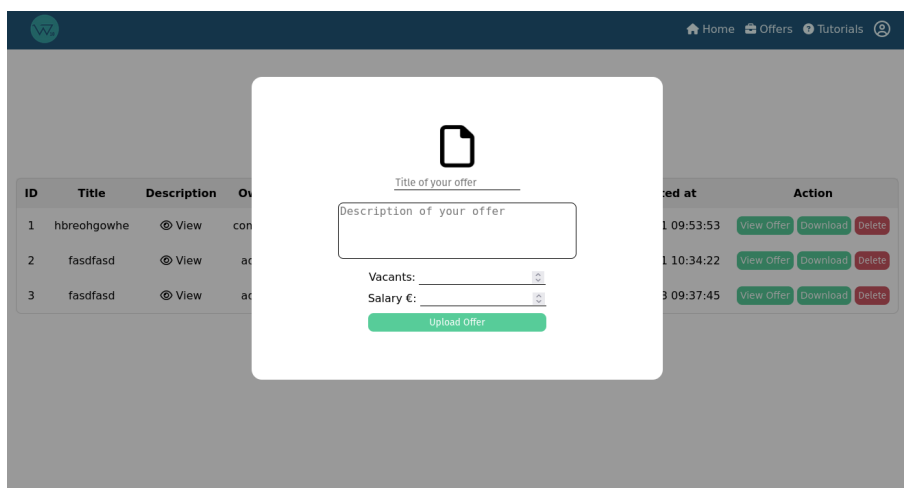
Las ofertas solo pueden ser subidas por aquellos usuarios que cuenten con el rol de *company* o administrador. Los administradores pueden subirlas también, ya que son los que se encargan de revisarlas y comprobar si son o no adecuadas.

Las ofertas se pueden subir desde la vista *newOffer* como se puede ver en la figura 4.38. Para acceder a ella se debe pulsar el botón de *upload offer* que aparece en la vista *homeOffer* si el usuario tiene el rol de *company* o *admin*. Si el usuario es un administrador, podrá además subir una oferta desde la vista *adminOffers* a la cual accederá desde la vista *admin* tal y como se muestra en la figura 4.39.



The screenshot shows a web interface with a dark blue header containing a logo and navigation links for Home, Offers, and Tutorials. The main content area is titled "Upload your offer!" and features a form with the following fields: "Title of your offer" (text input), "Description of your offer" (text area), "Vacants:" (dropdown menu), and "Salary:" (dropdown menu with a Euro symbol). A green "Upload Offer" button is positioned at the bottom of the form.

Figura 4.38: Vista *newOffer*



The screenshot shows the same "Upload your offer!" form as in Figure 4.38, but it is displayed as a modal window over a background view. The background view includes a table with columns for ID, Title, Description, and Action. The table contains three rows of data. The "Action" column for each row contains three buttons: "View Offer", "Download", and "Delete".

ID	Title	Description	Action
1	hbreohgowhe	© View con	View Offer Download Delete
2	fasdfasd	© View ac	View Offer Download Delete
3	fasdfasd	© View ac	View Offer Download Delete

Figura 4.39: Vista *adminOffers*

Cabe destacar que, si el usuario no tiene el rol de *company* o *admin*, en la vista de *homeOffer* no aparecerá el botón para añadir una nueva oferta o ver las ofertas subidas. La vista para ver las ofertas subidas se llama *myOffers* y solo muestra las ofertas subidas por el usuario pero se ve cómo la vista de *homeOffer*.

Si el usuario es el propietario de la oferta o bien administrador, podrá editar la oferta y acceder a la vista *offerEdit* cómo se puede ver en la figura 4.40. Esta vista

permite al propietario cambiar los campos de la oferta, pero no puede cambiar el documento asociado a ella. Esto se debe a que el fichero de la oferta es el elemento principal al que van asociados el resto de los campos, es decir sin documento, dichos campos no irían asociados a nada, por ello, si se desea cambiar el fichero de la oferta, esta debe crearse desde cero y subir una nueva oferta. La vista cuenta además con un botón para eliminar la oferta.

Figura 4.40: Vista *offerEdit*

Las diferencias en cuanto a la aparición o la no aparición de los botones para realizar las acciones comentados anteriormente se puede apreciar observando las figuras 4.41 y 4.42.

Figura 4.41: Vista *offerHome* desde usuario con rol *company* o *admin*

Para poder crear y subir una oferta se debe incluir, un archivo que tenga alguna de las extensiones que se han especificado en el fichero *OfferRequest*. También debe

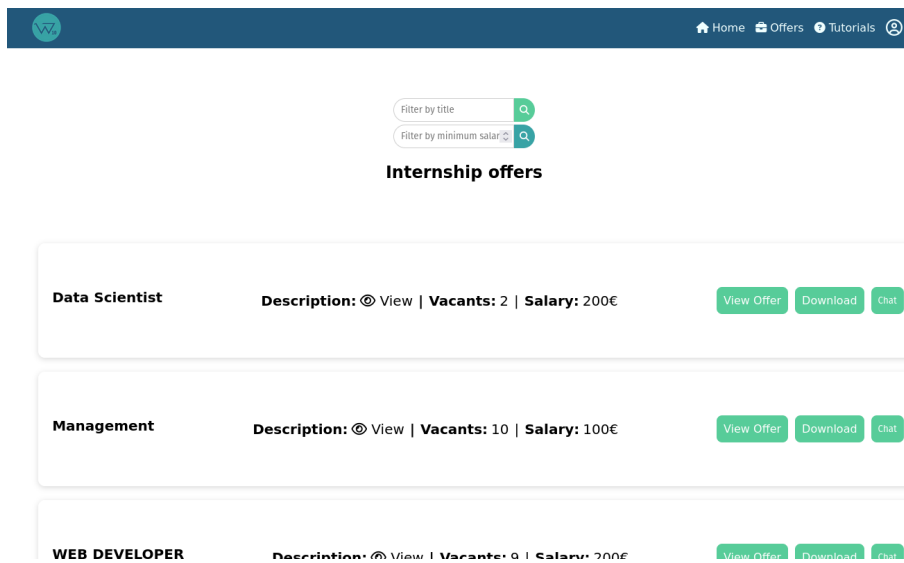


Figura 4.42: Vista *offerHome* desde usuario con rol *standard*

incluir un título para la oferta, una descripción, un número de plazas libres y un salario. El fichero que se suba debe ser propio ya que, al igual que con los vídeos, se pretende que todo aquello subido a la aplicación web sea original y evitar así los posibles problemas relacionados con la propiedad intelectual de los propietarios de dichos ficheros.

Todos los campos mencionados son obligatorios a la hora de subir una oferta a la aplicación, ya que todos ellos contienen información de interés para los estudiantes.

Las ofertas son almacenadas en la base de datos, el documento que decida subir la empresa a la hora de crear la oferta se guarda en una ruta en específico en la carpeta `/storage/app/public/offers`. Esto es importante debido a que los usuarios van a poder descargar dichos documentos con el fin de evaluar de mejor forma la oferta propuesta por la compañía, por ello es importante tener claro dónde van a estar estos documentos almacenados.

Con el fin de intentar seguir manteniendo un orden en el almacenamiento de los documentos, estos se almacenan en la ruta comentada anteriormente utilizando el título que tenga la oferta que contiene dicho documento y la fecha en la que se ha subido, ya que si solo fuera el título habría problemas al existir más de dos ofertas que contaran con el mismo título.

A la diferencia de lo que ocurre con los vídeos donde se guardan dos ficheros, aquí solo se guarda un fichero, por ello no hace falta crear subcarpetas para cada uno dentro de la carpeta `offers`. Para guardar el fichero se utiliza el siguiente formato dentro de la carpeta `offers`:

- `wishten-{identificador de la fecha}-{título de la oferta}.{extensión}`

Se utiliza este formato ya que es el que resulta más cómodo a la hora de tener que buscar una oferta en concreto. El uso de la fecha se debe a que se pueden modificar las ofertas y por tanto cambiar la fecha de estas a la fecha en la que se modificó.

Esta forma de almacenar los documentos hace que la labor de los administradores a la hora de gestionar las ofertas sea más rápida en caso de no poder realizarse dicha labor desde la página de administración de ofertas de la propia aplicación.

Gracias a lo comentado anteriormente los ficheros asociados a cada oferta se estructurarán automáticamente cómo se muestra en la figura 4.43.

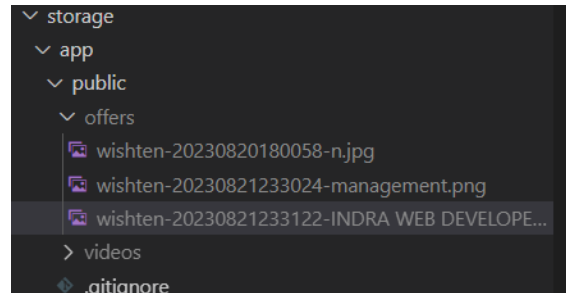


Figura 4.43: Estructura de almacenamiento de los documentos asociados a las ofertas

Respecto a la búsqueda de ofertas, los usuarios tendrán una vista general desde la vista *homeOffer*, a la cual se accede utilizando el segundo icono de la cabecera de la propia aplicación.

En esta vista encuentran todas las ofertas unas debajo de otras y se podrán realizar búsquedas por título gracias a la barra de búsqueda, la cual permite que los usuarios puedan centrarse solo en las que les interesan.

También existe otra barra de búsqueda que permite buscar las ofertas según un valor mínimo de salario, es decir, todas aquellas ofertas que tengan ese valor o uno mayor para el campo salario serán mostradas al usuario. De esta forma, los usuarios podrán buscar las ofertas por su título o por su salario, siendo ambas búsquedas compatibles, ya que se pueden buscar ofertas de un determinado título y dentro de ese título filtrar para ver aquellas que ofrecen las condiciones de salario que decida buscar el usuario.

Desde su vista el usuario podrá ver todos los campos de la oferta, es decir, el título, la descripción, la cantidad de plazas libres y el salario. Para el documento este deberá pulsar el botón llamado *download* y se le descargará el documento que haya subido la compañía a la oferta. También contará con un botón para previsualizar el documento antes de descargarlo.

Hay que destacar que, dado que el campo descripción puede ser muy largo, se ha decidido usar una ventana modal para que este se muestre en cada oferta y no interfiera con la estructura de la vista. Esto puede verse en la figura 4.44.

Finalmente, cabe destacar que se ha creado una página principal para los usuarios no registrados (figura 4.45) en la que se explican las funcionalidades que ofrece la aplicación. Asimismo, se muestran tres vídeos en los que se explican algunas de las funcionalidades básicas: subir un vídeo, subir una oferta y añadir preguntas a un vídeo.

Además, los usuarios registrados que deseen consultar esta información podrán seguir accediendo a esta vista a través de la ruta *tutorials*.

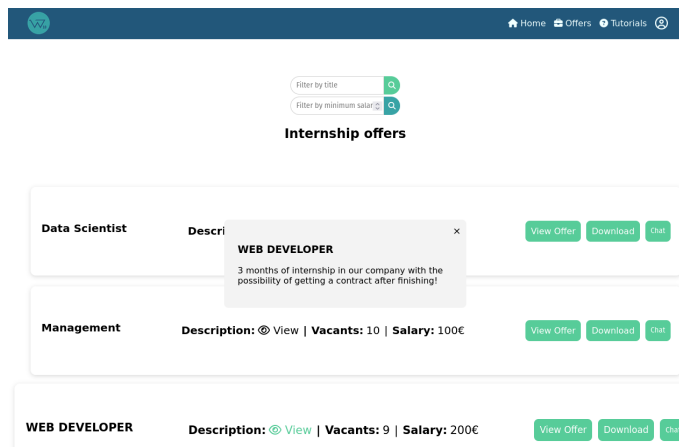


Figura 4.44: Vista *offerHome* cuando un usuario hace clic en *View* para ver la descripción de la oferta

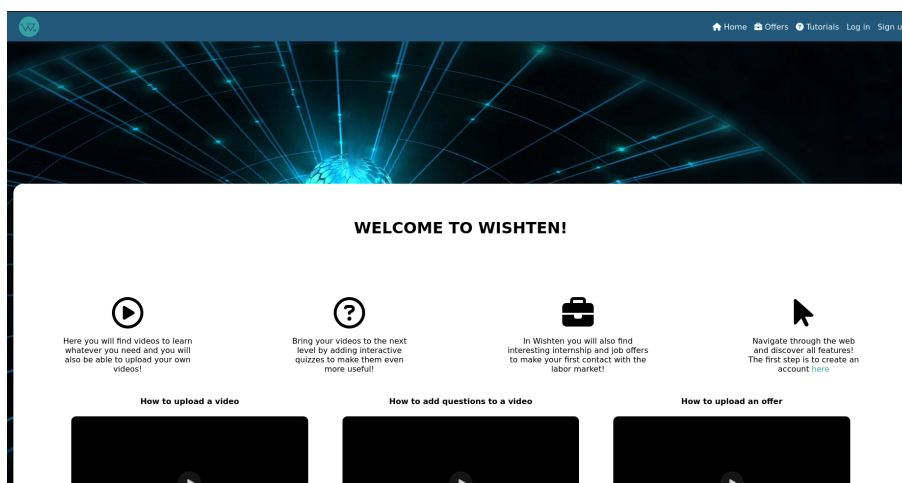


Figura 4.45: Vista *homeGuest*

#### 4.6.3.5. Gestión de chat

Todas las funcionalidades para crear un chat, enviar y recibir mensajes se gestionan desde el controlador *ChatController*.

Para entender cómo se envían y reciben mensajes, primero hay que explicar cómo se crea un chat. Los usuarios que tengan el rol *standard*, en cada oferta tendrán un botón llamado chat y cuando éstos hagan clic sobre él, se creará una conversación entre el usuario que ha subido la oferta, es decir, la empresa, y el usuario que pulsó el botón. En el momento que se pulsa dicho botón se crea la conversación entre ambos usuarios y dicha conversación seguirá creada, aunque no se envíe ningún mensaje.

Los usuarios con rol *standard* son los únicos que pueden iniciar una conversación, por lo que son ellos siempre los que la van a crear al pulsar el botón de chat.

La vista *chat* es la encargada de mostrar la conversación entre dos usuarios. A la hora de crear un chat, éste estará vacío y, por tanto, la vista mostrará sólo un chat en blanco como se puede observar en la figura 4.46. Sin embargo, cuando el usuario

comience a enviar mensajes, éstos irán apareciendo en pantalla tal y cómo se puede ver en la figura 4.47.

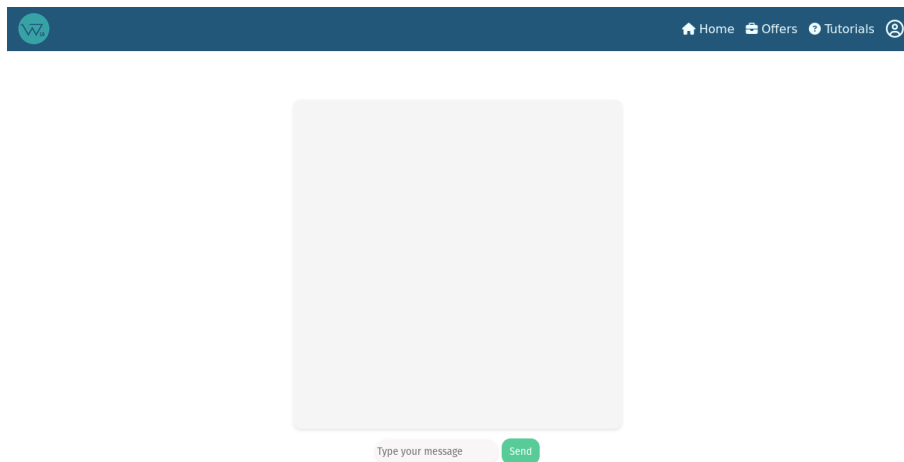


Figura 4.46: Vista *chat* - Conversación sin mensajes



Figura 4.47: Vista *chat* - Conversación con mensajes

Cada mensaje que se envía muestra el nombre del usuario que ha enviado dicho mensaje, en la parte superior, y la fecha en que dicho mensaje ha sido enviado, en la parte inferior derecha. Para enviar mensajes únicamente se debe escribir el cuerpo del mensaje que se quiera enviar en el input que aparece en la parte inferior y pulsar el botón *send* para enviarlo. Todos los mensajes enviados por ambas partes se mostrarán en orden cronológico para seguir el flujo de la conversación.

Para que un usuario pueda ver todos los chats que tiene activos, es decir, todos aquellos chats en los que ha enviado algún mensaje, se ha creado la vista *myChats*. A esta vista pueden acceder los usuarios con rol *standard* desde la página principal de ofertas, a través de un botón situado en la parte superior, encima de la barra de búsqueda de ofertas.

En la vista *myChats* aparecerán todos los chats en los que el usuario ha enviado algún mensaje, mostrando por cada uno el nombre de la empresa, el título de la oferta a la que va asociada la conversación y un botón para acceder al chat correspondiente, como se puede observar en la figura 4.48.

Sólo se muestran los chats que tengan algún mensaje, ya que puede que el usuario tenga varias conversaciones abiertas con varias empresas, pero no haber enviado ningún mensaje en ellas. También puede darse el caso en el que no tenga ningún chat activo, por lo que, en dicho caso, no aparecerá ningún chat.

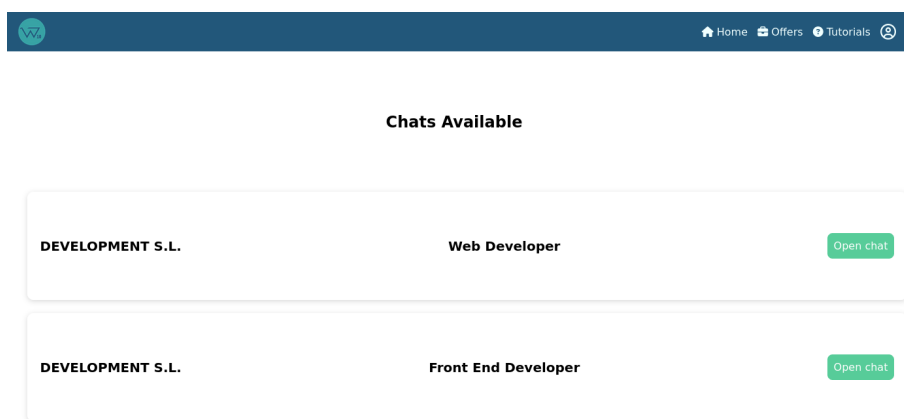


Figura 4.48: Vista *myChats*

Los usuarios con rol *company* o *admin* no contarán con la vista comentada anteriormente, pero sí con otra llamada *chatList*.

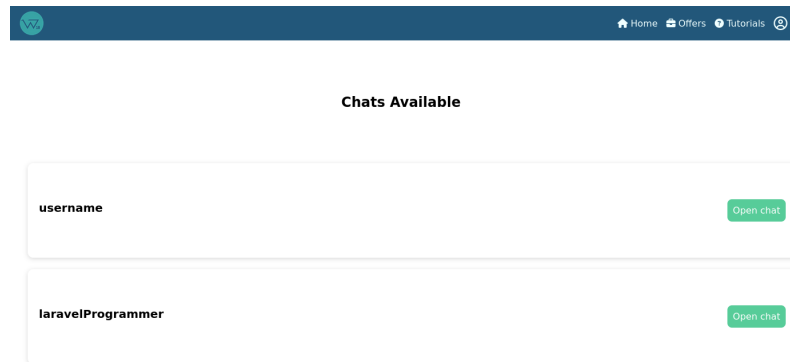
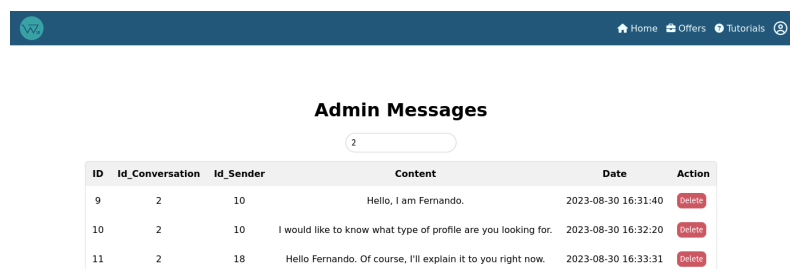
La función de esta vista es proporcionar a las empresas un listado de todos los chats que tienen activos por cada oferta. El hecho de que estén activos significa que la conversación contiene algún mensaje.

Para poder acceder a la vista *chatList* sólo hay que pulsar el botón llamado *View Chats* que se encuentra asociado a cada oferta en la vista *myOffers*.

Gracias a esto, las empresas pueden ver cada cierto tiempo qué ofertas son las que tienen chats activos, ya que la vista *chatList* muestra todos ellos, como se puede ver en la figura 4.49. Puede darse el caso de que una oferta no tenga chats activos, por lo que en dicho caso no aparecerá ningún usuario.

Por último, el administrador podrá acceder desde la vista *adminMessages* a todos los mensajes, pudiendo buscar todos aquellos mensajes de una determinada conversación, introduciendo su id correspondiente tal y como se ve en la figura 4.50.

En esta vista el administrador podrá eliminar aquellos mensajes que crea necesario pulsando el botón de *delete*, al igual que en el resto de las vistas de administración. Como éste es el único usuario que puede eliminar un mensaje, no ha sido necesario en esta ocasión crear un fichero *policy* para restringir las acciones sobre los mensajes.

Figura 4.49: Vista *chatList*Figura 4.50: Vista *adminMessages*

## 4.7. Casos de uso

A continuación, se definen los actores de la aplicación web, es decir, cada tipo de usuario que interactúa con el sistema, así como las propias interacciones que éstos realizan con la aplicación. Estas interacciones se denominan casos de uso y difieren según el tipo de usuario.

### 4.7.1. Actores de la aplicación

Dentro de la aplicación se diferencian tres actores, que se corresponden con los tres roles de usuario definidos anteriormente:

- **Estudiante:** Son los usuarios con rol *standard* y se presupone que en su mayoría son estudiantes universitarios. Éstos usuarios pueden acceder a todo el contenido de la aplicación, tanto vídeos como ofertas. Además, tienen la capacidad de crear y publicar los vídeos que otros usuarios visualizan, así como introducir cuestionarios dentro de ellos.
- **Empresa:** Son los usuarios con rol *company* y únicamente pueden acceder al contenido relativo a las ofertas. Éstos usuarios pueden crear y gestionar las ofertas publicadas, así como comunicarse con los solicitantes.

- Administrador: Son los usuarios con rol *admin* y tienen acceso y control total sobre todo el contenido de la aplicación. Pueden crear, modificar y eliminar vídeos, ofertas y usuarios, así como alterar características de éstos, como la visibilidad de un vídeo o el rol de un usuario.

### 4.7.2. Diagramas de casos de uso

Para comprender cómo interactúan los usuarios con la aplicación, se ha elaborado un diagrama por cada actor en el que se muestran los casos de uso, es decir, las funcionalidades del sistema con las que interactúa dicho actor. Los diagramas han sido creados siguiendo el Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, es un lenguaje de modelado visual que se utiliza para diseñar sistemas software). En estos diagramas, los casos de uso se conectan entre ellos mediante dos tipos de relaciones: *«include»*, que implica que un caso de uso utiliza siempre a otro, por lo que depende de él; y *«extend»*, que implica que un caso de uso extiende la funcionalidad de otro, pero de manera opcional.

#### 4.7.2.1. Estudiante

En la figura 4.51 se muestra el diagrama de casos de uso de los estudiantes. Todas las acciones requieren haber iniciado sesión y, por tanto, haberse registrado en primer lugar en la aplicación. De esta forma, se pueden almacenar métricas, como las respuestas que den los usuarios a los cuestionarios, el número de usuarios que ve cada vídeo o el número de usuarios que utiliza la aplicación. Si los usuarios no se registraran previamente, se perdería esta información, que es valiosa para conocer el uso que se le da a la aplicación.

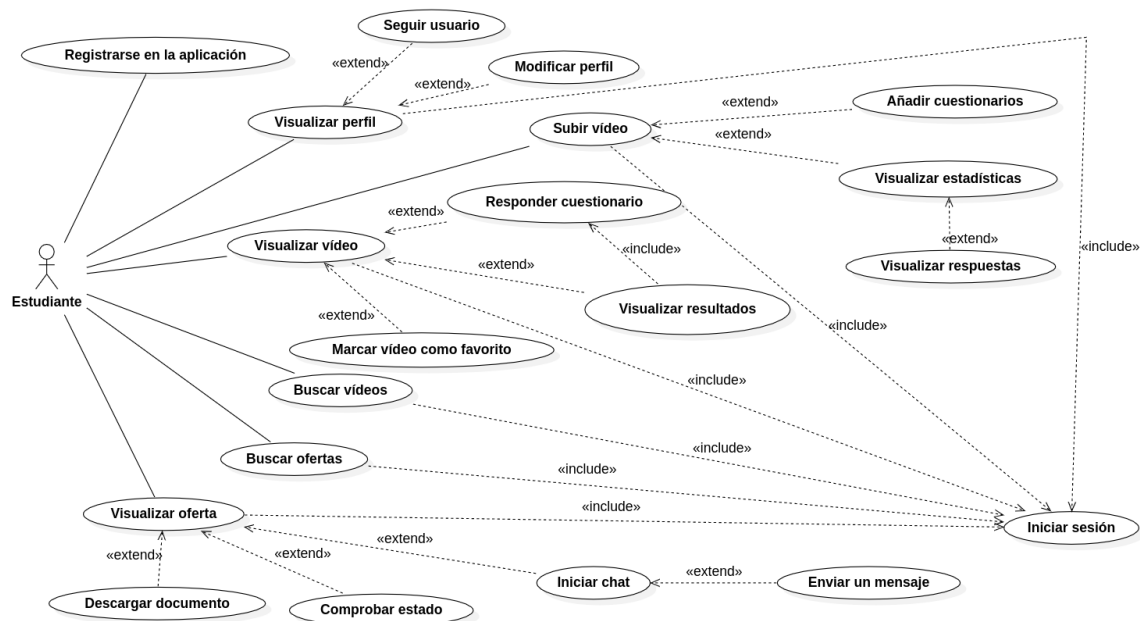


Figura 4.51: Diagrama de casos de uso - Estudiante

### 4.7.2.2. Empresa

En el diagrama que se muestra en la figura 4.52 se observan los casos de uso específicos para los usuarios con rol empresa. Estos usuarios pueden realizar las mismas acciones que los estudiantes, más las funcionalidades mostradas en el diagrama, relativas a la gestión de ofertas.

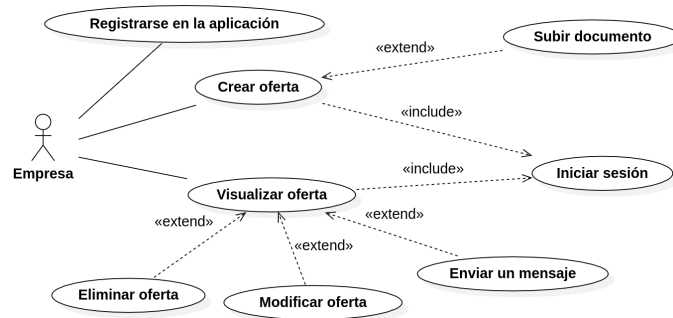


Figura 4.52: Diagrama de casos de uso - Empresa

### 4.7.2.3. Administrador

El diagrama de casos de uso del usuario administrador se muestra en la figura 4.53. Los administradores tienen control sobre todo el contenido de la aplicación: cuentas de usuario, vídeos, ofertas y mensajes.

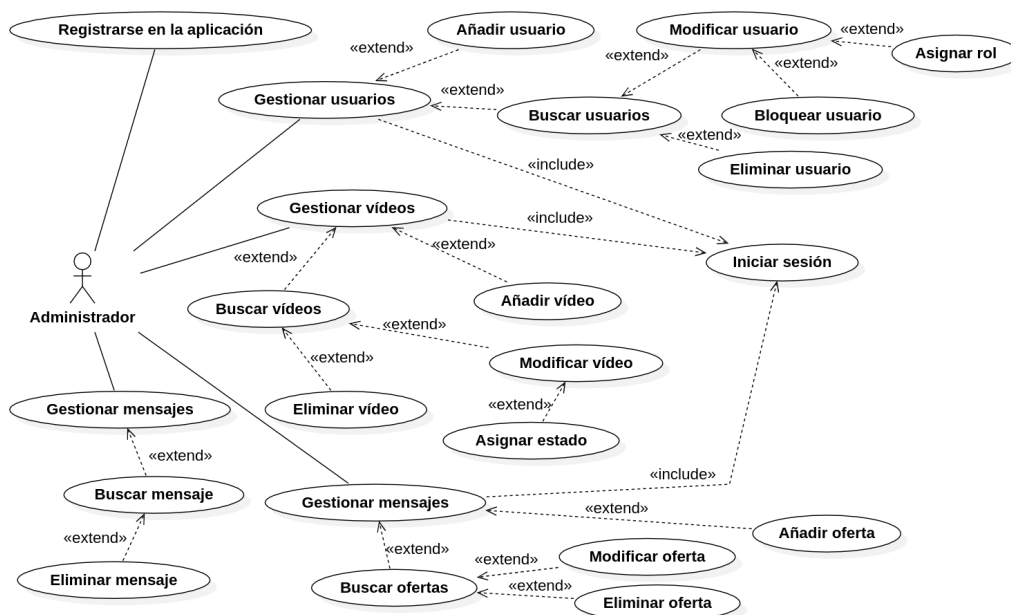


Figura 4.53: Diagrama de casos de uso - Administrador

## 4.8. Diagramas de flujo

Seguidamente se muestran los diagramas de flujo elaborados para explicar de manera gráfica la interacción de los usuarios con la aplicación. En ellos se muestra el flujo de acciones que se llevan a cabo en las principales funcionalidades.

### 4.8.1. Inicio de sesión y registro

En la figura 4.54 se muestra el proceso que se sigue para iniciar sesión en la aplicación. Para ello, el usuario debe registrarse previamente en la aplicación e introducir correctamente su *email* y contraseña.

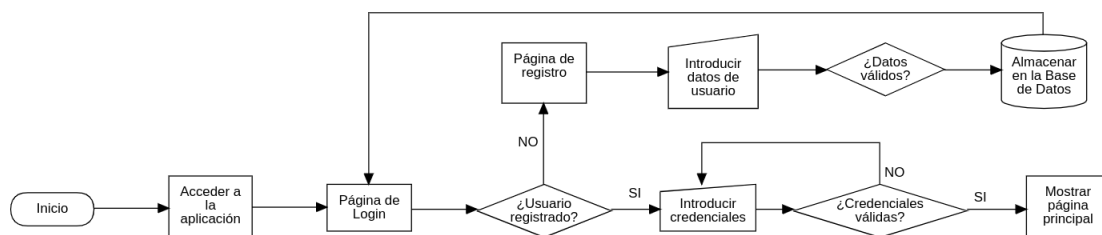


Figura 4.54: Diagrama de flujo - *Login* y registro

### 4.8.2. Subir un vídeo

Para subir un nuevo vídeo, el usuario debe introducir la información de éste y añadir los ficheros correspondientes al vídeo y la miniatura. En la figura 4.55 se muestra dicho proceso en un diagrama de flujo.

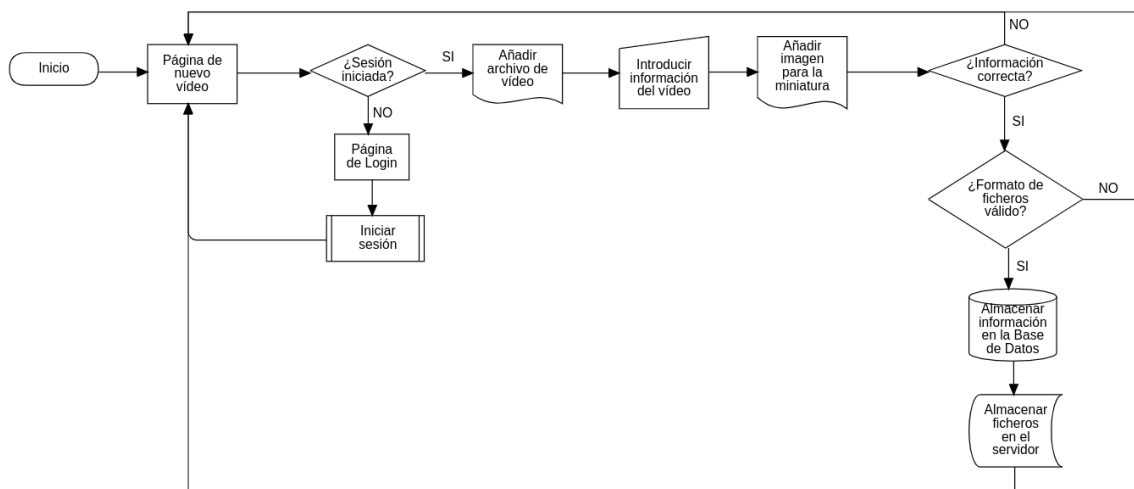


Figura 4.55: Diagrama de flujo - Subir un vídeo

### 4.8.3. Editar un vídeo

En la figura 4.56 se observa el proceso para editar un vídeo. Desde la página de edición de vídeo se pueden efectuar cambios sobre el vídeo, así como acceder a la página *add-quiz* para añadir preguntas al vídeo.

A los usuarios que no sean administradores ni propietarios del vídeo, se les restringe el acceso, reportándoles el código HTTP 403, que indica que tienen el acceso prohibido a esa página.

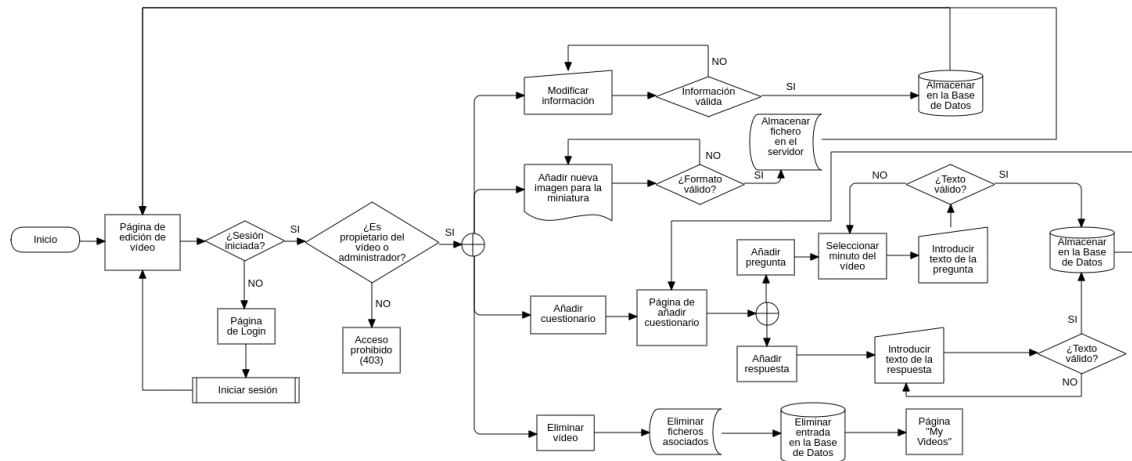


Figura 4.56: Diagrama de flujo - Editar un vídeo

### 4.8.4. Visualizar un vídeo

El proceso que sigue un usuario para encontrar y visualizar un vídeo se muestra en la figura 4.57. El usuario primero busca el nombre del tema que le interesa y se le mostrarán todos los vídeos subidos etiquetados con ese tema.

En caso de un vídeo que contenga preguntas interactivas, el usuario podrá contestarlas mientras visualiza el vídeo y, finalmente, almacenar su puntuación.



Figura 4.57: Diagrama de flujo - Visualizar un vídeo

### 4.8.5. Subir una oferta

En la figura 4.58 se muestra el proceso para subir una nueva oferta. Si el usuario no tiene el rol *company* o *admin*, es redirigido a la página de la que viniera.

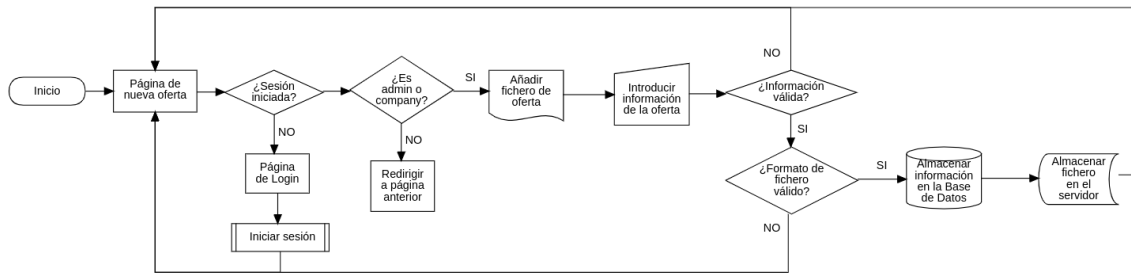


Figura 4.58: Diagrama de flujo - Subir una oferta

### 4.8.6. Editar una oferta

Para editar una oferta, se sigue el proceso descrito en el diagrama que se muestra en la figura 4.56. Para poder editar la información de una oferta, el usuario debe ser el propietario de la oferta o un administrador.

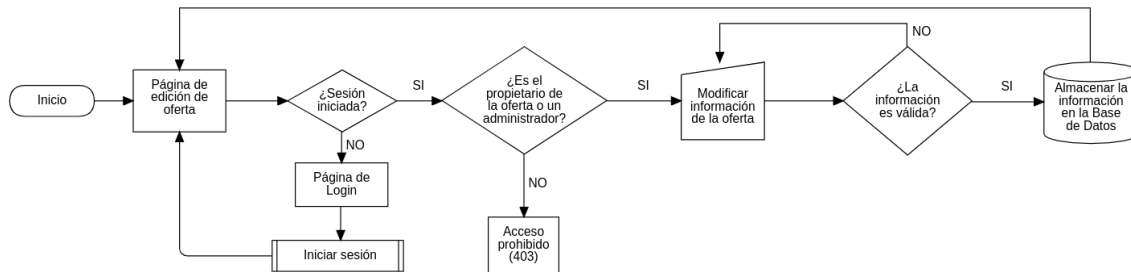


Figura 4.59: Diagrama de flujo - Editar una oferta

### 4.8.7. Visualizar una oferta

El proceso que sigue un usuario para encontrar e interactuar con una oferta se observa en la figura 4.60. Los usuarios con rol *standard* pueden iniciar un chat en cada oferta para hablar con la empresa que las publicó. Por su parte, los usuarios con rol *company* podrán acceder a la lista de chats de cada oferta para hablar con los usuarios solicitantes.

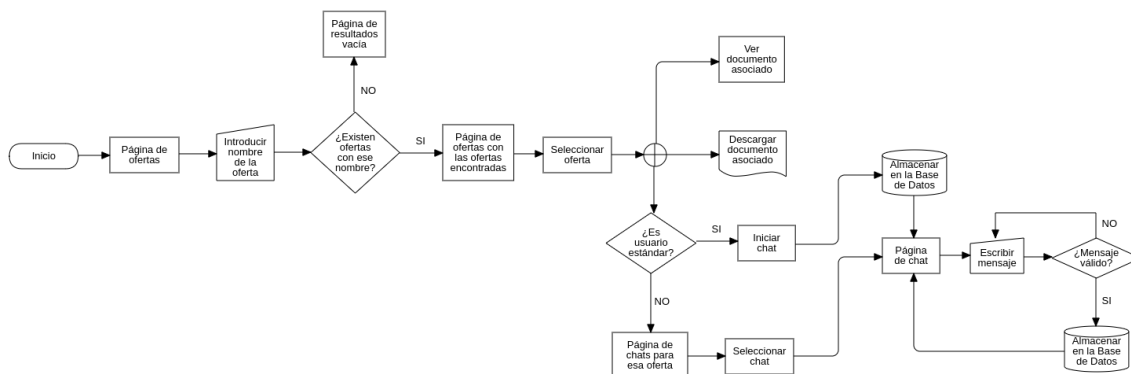


Figura 4.60: Diagrama de flujo - Visualizar una oferta

## 4.9. Pruebas

Una vez implementado el código de la aplicación y tras haber definido los casos de uso y los diagramas de flujo, se realizaron pruebas para garantizar el correcto funcionamiento del programa desarrollado. Para ello se elaboró un guion con una serie de tareas a realizar por los tres tipos de usuarios definidos anteriormente.

Para el usuario con rol *standard* se definieron las siguientes tareas:

1. Registrarse en la aplicación e iniciar sesión.
2. Visualizar el perfil.
3. Cambiar la foto de perfil.
4. Navegar por la página principal y buscar vídeos por temas.
5. Visualizar un vídeo y responder a las preguntas incrustadas en éste.
6. Subir un vídeo a la aplicación.
7. Editar la información del vídeo.
8. Añadir preguntas y anotaciones al vídeo.
9. Eliminar un vídeo.
10. Navegar por el portal de ofertas y realizar búsquedas.
11. Visualizar y descargar un documento asociado a una oferta.
12. Iniciar un chat con una empresa desde una oferta.
13. Enviar un mensaje desde el chat de una oferta.
14. Visualizar todos los chats activos.

Para el usuario con rol *company* se propuso otra lista de tareas:

1. Acceder al portal de ofertas.
2. Subir una nueva oferta.
3. Editar la información de la oferta.
4. Reemplazar el documento asociado a la oferta.
5. Visualizar todos los chats activos en una oferta.
6. Enviar un mensaje a un usuario desde el chat de una oferta.
7. Eliminar una oferta.

Finalmente, para el usuario con rol *admin* se elaboró otra lista de tareas:

1. Acceder a la página de administración de usuarios.
2. Añadir un usuario nuevo.
3. Modificar la información de un usuario.
4. Bloquear la cuenta de un usuario.
5. Cambiar el rol de un usuario.
6. Eliminar un usuario.
7. Acceder a la página de administración de vídeos.
8. Añadir un vídeo nuevo.
9. Modificar la información de un vídeo.
10. Cambiar el estado de un vídeo.
11. Eliminar un vídeo.
12. Acceder a la página de administración de ofertas.
13. Añadir una nueva oferta.
14. Editar una oferta.
15. Eliminar una oferta.
16. Acceder a la página de administración de mensajes.
17. Eliminar un mensaje.

Todas las pruebas se realizaron sobre la aplicación desplegada en un servidor Apache, montado sobre una máquina Linux y utilizando el servicio MySQL para gestionar la base de datos.

En primer lugar, para comprobar el correcto funcionamiento de la aplicación, se realizaron las tareas descritas anteriormente. Todas ellas fueron cumplidas con éxito, por lo que se determinó que las funcionalidades principales de la aplicación fueron implementadas correctamente.

En segundo lugar, a fin de examinar la usabilidad de la aplicación, se realizaron pruebas con siete usuarios ajenos al desarrollo de la misma. A uno de ellos se le asignó el rol de *admin*, a otro, el de *company* y a los cinco restantes, el rol *standard*.

Para la realización de estas pruebas se utilizó la técnica de *think aloud*, en la que los usuarios verbalizan los pensamientos, los sentimientos y las acciones que realizan mientras realizan una tarea. De esta manera, se pueden comprobar los fallos de usabilidad que encuentran los usuarios y que no se detectaron en el proceso de desarrollo.

Las pruebas de los usuarios con rol *admin* y *company* fueron realizadas de forma presencial, mientras que las de los cinco usuarios con rol *standard* lo fueron en remoto. Para la realización de las pruebas en remoto se utilizó la aplicación Discord<sup>32</sup>, una aplicación de mensajería que cuenta con la opción de compartir pantalla durante una llamada.

Para que los usuarios que realizaron las pruebas en remoto pudieran utilizar la aplicación, se les proporcionó acceso al servidor para que fueran ellos los que realizaran las acciones mientras compartían la pantalla y decían en voz alta lo que iban haciendo.

Se les pidió a los usuarios que comentaran todo aquello que considerasen mejorable o que no entendieran. Además, se les instó a visualizar previamente los vídeos explicativos que se encuentran en la sección *tutorials* para que tuvieran una referencia sobre el funcionamiento y la distribución de la aplicación.

Gracias a estas pruebas, se pudo obtener retroalimentación en cuanto a la usabilidad de la aplicación para mejorarla.

---

<sup>32</sup><https://discord.com>

## Resultados

Una vez realizadas las pruebas se procedió a analizar los resultados obtenidos a partir de ellas.

Como las pruebas fueron realizadas por usuarios que adoptaron tres roles diferentes, se va a proceder a obtener los resultados de dichas pruebas en función de cada rol.

En primer lugar se van a examinar los comentarios proporcionados por el usuario que adoptó el rol de administrador.

Este usuario realizó la prueba presencialmente por lo que no tuvo que realizar el paso previo de conectarse al servidor. A este usuario se le proporcionó un guion con las acciones que debía llevar a cabo.

El usuario, tras realizar las tareas asociadas a la gestión de usuarios, comentó que no tuvo ningún problema en encontrar la página a la que tenía que ir para realizar todas las acciones (la página de administración de usuarios). Una vez dentro de ella encontró rápidamente los botones necesarios para cumplir las tareas.

De cara a la realización del resto de acciones, como el usuario tuvo que pasar por la página de administración para acceder a la página de administración de usuarios, ya sabía dónde encontrar el resto de páginas de administración para las ofertas, los vídeos, y los mensajes. El usuario afirmó que el hecho de que todas esas páginas sigan una misma estructura y se encuentren agrupadas facilitó la realización de las tareas.

En segundo lugar se van a observar los comentarios dados por el usuario que adoptó el rol de empresa.

Desde el principio se pudo observar cómo el usuario no tuvo ningún problema a la hora de realizar ninguna de las acciones que se le pidieron. Sin embargo, éste presentó una duda referente al motivo de la existencia de un botón para ver la oferta y otro para descargarla, ya que no entendía la diferencia entre ambos. No obstante, como se considera que eso no supone un problema de usabilidad, se ha determinado que no es necesario realizar ningún cambio al respecto.

Si bien no tuvo problemas para editar una oferta, sí que preguntó por la imposi-

bilidad de cambiar el documento asociado a la oferta. A pesar de que se le explicaron los motivos asociados a la dicha circunstancia, el comentario aportado se tendrá en cuenta para realizar una mejora en el futuro.

Por último, se van a examinar los comentarios proporcionados por los usuarios restantes que adoptaron el rol *standard*, es decir, el usuario promedio de la aplicación.

El primer usuario no tuvo ningún problema a la hora de realizar las acciones que se le pidieron ni para acceder a las páginas dónde realizarlas. Sin embargo, el usuario expresó que hubiera preferido que los tutoriales estuvieran disponibles en español, lo cual ya se tenía planeado añadir en un futuro.

El segundo usuario tuvo problemas para guardar los resultados de los cuestionarios, ya que pensó que se actualizaban automáticamente, a pesar de que para ello es necesario pulsar un botón para almacenar las respuestas proporcionadas en un vídeo. Este usuario indicó que le hubiera resultado más cómodo si dicho botón no existiera o, de existir, que éste se encontrara más visible.

El tercer usuario no tuvo ningún problema en relación a la realización de las acciones pedidas ni realizó ningún comentario sobre aspectos que le resultaran incómodos o mejorables.

El cuarto usuario tampoco tuvo problemas a la hora de realizar las acciones pero sí hizo varios comentarios. El primero fue relativo a la existencia de las dos barras de búsqueda en la vista de las ofertas. Preguntó si no se podría realizar la búsqueda únicamente con una barra ya que, a pesar de ser intuitivo y de entender el funcionamiento de ambas, indicó que no suele ser común contar con dos barras de búsqueda en otras aplicaciones.

El segundo comentario que hizo fue relativo al idioma de los tutoriales, puesto que hubiera preferido que estuvieran en español.

El quinto y último usuario expresó que no le resultaba muy cómodo el tener que crear un vídeo de cero en lugar de, simplemente, cambiar el archivo de vídeo. También realizó el mismo comentario referente a las ofertas. Sin embargo, al explicarle los motivos detrás de esta implementación, el usuario lo entendió perfectamente. No obstante, el comentario será tenido en cuenta de cara a una futura mejora.

Todos los comentarios proporcionados por los usuarios de prueba serán tomados en cuenta para mejorar la aplicación en un futuro. En primer lugar se centrarán los esfuerzos en aquellas mejoras relacionadas con el idioma de los tutoriales y de la página, así como añadir la posibilidad de cambiar el fichero de vídeo en los vídeos y el de texto en las ofertas.

## Evaluación

Puesto que se pretende realizar un estudio exhaustivo sobre la utilidad en el uso de cuestionarios dentro de los vídeos, se ha realizado un diseño de evaluación para el efecto. Esta evaluación no se ha podido llevar a cabo durante el desarrollo del presente trabajo, ya que se pretendía que ésta se realizara a lo largo de un curso académico completo.

Para la realización de dicha evaluación se buscarán 20 estudiantes universitarios de Ingeniería Informática y se seleccionarán 4 asignaturas que vayan a cursar dichos estudiantes en ese año: 2 en el primer cuatrimestre y 2 en el segundo.

En primer lugar, se creará una serie de vídeos aplicados a cada asignatura bajo la supervisión de los profesores que imparten dichas asignaturas, con el fin de garantizar que los vídeos abarquen todos los contenidos necesarios.

De esta manera, los vídeos seleccionados serán utilizados por los estudiantes como complemento a su estudio habitual y se prepararán cuestionarios y anotaciones para incluir en los vídeos que se suban a Wishten.

Una vez se hayan seleccionado a los 20 estudiantes, se les dividirá en dos grupos. Un primer grupo de 10 estudiantes que actuará como grupo de control durante la evaluación y un segundo grupo con los estudiantes restantes que actuará como grupo experimental.

El grupo de experimentación utilizará exclusivamente la aplicación Wishten como complemento a su estudio, no permitiéndoles el uso de otras plataformas de vídeos como Youtube, Vimeo, etc. Por otra parte, al grupo de control se le pedirá que sí utilice dichas plataformas pero no se le concederá acceso a Wishten.

Una vez organizados los grupos, al grupo de experimentación se le proporcionará acceso a través de Wishten a los vídeos comentados anteriormente junto a sus respectivos cuestionarios y anotaciones. Por su parte, al grupo de control se le proporcionará los vídeos a través de otros medios y sin los cuestionarios ni las anotaciones.

Durante el periodo de evaluación se realizará periódicamente una serie de encuestas para comprobar la opinión que tienen sobre el método de estudio que están

utilizando.

Finalmente, al término de cada cuatrimestre y tras la realización de los exámenes y la obtención de las calificaciones finales se realizará una evaluación final. En dicha evaluación se consultará la opinión de los estudiantes sobre los respectivos métodos de estudio y se analizará el rendimiento académico obtenido en ambos grupos. Adicionalmente, se le pedirá al grupo de experimentación que realice una evaluación y una valoración de la aplicación.

Una vez obtenidos todos los resultados, se procederá realizar un análisis de los mismos.

Primeramente, se estudiarán los resultados de las encuestas periódicas relativas al método de estudio empleado por cada grupo, e individualmente por cada estudiante. Se relacionará el impacto de dicho método de estudio sobre los resultados académicos, así como la percepción de los estudiantes respecto al método empleado.

Por último, se analizarán las opiniones finales de los estudiantes en relación a todo el proceso de aprendizaje seguido en cada cuatrimestre y se evaluará la percepción que tengan los estudiantes pertenecientes al grupo de experimentación sobre la aplicación utilizada.

Con estos resultados se pretende evaluar la utilidad de incluir cuestionarios dentro de los vídeos, así como comprobar si su implementación dentro de la aplicación web les resulta útil a los estudiantes. Se espera que los resultados de esta evaluación vayan en consonancia con los obtenidos por los trabajos relacionados analizados en el capítulo 2.

En relación a la evaluación del portal de ofertas, como no se tiene contacto con ninguna empresa, no se puede realizar una evaluación íntegra. Sin embargo, se propone publicar algunas ofertas de prueba y realizar una encuesta de satisfacción a los estudiantes del grupo de experimentación para que valoren también este aspecto de la aplicación.

## Conclusiones

### 7.1. Conclusiones

En general, la mayor parte de los objetivos que se plantearon previamente al desarrollo de la aplicación web se han cumplido. Los dos objetivos principales que buscaba abordar la aplicación son, por un lado, proporcionar una ayuda a los estudiantes a través de la visualización de vídeos interactivos, y por otro, facilitar la conexión empresa-estudiante proporcionando un portal donde poder ver las ofertas y hablar con las empresas.

Con el fin de poder cumplir los objetivos planteados inicialmente, se desarrolló con éxito una aplicación web que incorporara los elementos y las funcionalidades necesarias.

En este sentido, el objetivo relacionado con la visualización de vídeos interactivos se ha conseguido cumplir de manera satisfactoria, puesto que en la aplicación desarrollada los usuarios pueden subir vídeos, añadir cuestionarios dentro de ellos y buscar aquellos vídeos que les interesen. Mediante la visualización de dichos vídeos, se pretende que los estudiantes logren alcanzar una mayor retención de la información gracias a los cuestionarios realizados durante los vídeos.

Gracias a las pruebas realizadas y a los resultados obtenidos, se ha podido comprobar que la aplicación presenta una buena usabilidad en este aspecto, ya que los usuarios que la utilizaron la encontraron cómoda e intuitiva.

Sin embargo, no se pudo realizar una evaluación exhaustiva y prolongada sobre la utilidad del uso de cuestionarios dentro de los vídeos. No obstante, dicha evaluación ha sido diseñada para realizarse en un futuro.

El otro objetivo principal, relacionado con la conexión empresa-estudiante, también se ha podido alcanzar con éxito. La aplicación permite a las empresas subir ofertas, las cuales pueden ser vistas o descargadas por el resto de usuarios de la aplicación. Además de esto, el portal de ofertas permite a los usuarios comunicarse con las empresas a través de un chat asociado a cada una de las ofertas subidas por cada empresa.

Gracias a la existencia de los chats, que ofrecen una comunicación más directa, y a la utilización de los filtros existentes, los usuarios pueden agilizar el proceso de búsqueda de ofertas.

Al igual que con los vídeos, las pruebas realizadas han generado resultados positivos en relación a la usabilidad y comodidad.

Si bien al final se han podido cumplir los objetivos principales de manera satisfactoria, durante el desarrollo de estos se han presentado diversas dificultades que se han ido resolviendo de forma progresiva.

Al principio, se tuvieron muchas dudas sobre las tecnologías que se iban a emplear para el desarrollo de la aplicación. Se dedicó mucho tiempo a investigar qué *framework* sería el más adecuado para alcanzar los objetivos previamente planteados. De esta manera, finalmente, se llegó a la conclusión de que Laravel sería el *framework* más indicado para la implementación del código de la aplicación.

Una vez seleccionado el *framework*, a pesar de los conocimientos previos que se tenían sobre PHP, HTML, CSS y JavaScript, se tuvo que investigar en profundidad para comprender la estructura de ficheros de Laravel, así como los comandos y la sintaxis que éste utiliza.

Cuando se empezó a desarrollar la aplicación, se planteó la idea de implementar una funcionalidad que permitiera a los usuarios grabar sus propios vídeos desde la propia aplicación, dándoles la posibilidad de grabar su pantalla y su audio. Para ello, se utilizaron dos *scripts* en Python: uno para grabar el audio y otro para grabar la pantalla. Sin embargo, si bien el de grabación de audio llegó a funcionar, el *script* encargado de grabar la pantalla no funcionaba como se esperaba. Como todavía quedaba el resto de funcionalidades por implementar, se decidió posponer esta funcionalidad para un trabajo futuro. Por ello, se conservaron estos *scripts* en el repositorio del proyecto, para continuar con su implementación posteriormente.

En relación a la funcionalidad de gestión de vídeos, se tuvieron dificultades para implementar los cuestionarios dentro de los vídeos. Si bien se valoró la posibilidad de utilizar *frameworks* de JavaScript que ya implementan dicha funcionalidad, como H5P<sup>1</sup>, éstos no eran compatibles con la versión de Laravel en la que se estaba desarrollando la aplicación. Por tanto, al final, se decidió implementar esta funcionalidad desde cero utilizando JavaScript.

Otra de las dificultades encontradas fue la creación del chat, ya que al principio se planteó una estructura de las tablas en la base de datos, pero como no se conseguía alcanzar lo que se pretendía, se tuvo que rediseñar esta estructura de nuevo. Se valoró la opción de utilizar Pusher<sup>2</sup>, el cual facilita el desarrollo de aplicaciones en tiempo real, sin embargo, para utilizarlo se requería realizar muchas modificaciones en la estructura del proyecto. Por esta razón se decidió seguir el camino que se planteó en un inicio pero realizando las modificaciones necesarias en la estructura de las tablas utilizadas para los chats.

Una de las dificultades que requirió más tiempo para solventarse fue la forma

---

<sup>1</sup><https://h5p.org>

<sup>2</sup><https://pusher.com>

de almacenar los ficheros correspondientes a los vídeos y las ofertas en Laravel. Al principio se desconocía qué carpetas de Laravel se encargaban de dicha funcionalidad, así cómo las instrucciones necesarias para implementarla. Sin embargo, tras investigar y realizar numerosas pruebas se pudo encontrar una manera sencilla de almacenar los ficheros, basada en una jerarquía de directorios.

Si bien la utilización de Laravel en un principio supuso una serie de dificultades, una vez se entendió su funcionamiento, el uso de este *framework* agilizó mucho el proceso de desarrollo de la aplicación, gracias a su estructura de ficheros y a la sintaxis que éste utiliza.

A pesar de todas las dificultades encontradas a lo largo del desarrollo del proyecto, el resultado obtenido se considera satisfactorio, ya que cumple con éxito los objetivos que se habían planteado inicialmente.

## 7.2. Trabajo futuro

A pesar de que la mayoría de los objetivos iniciales planteados se han logrado cumplir de manera satisfactoria, hay algunos marcos y funcionalidades de la aplicación que se hubieran querido mejorar o ampliar. Sin embargo, debido a la planificación del tiempo en relación con el desarrollo, no se pudieron llevar a cabo.

Una de las funcionalidades que se empezó a desarrollar, pero que al final se tuvo que posponer, es la ofrecer a los usuarios un espacio para poder crear sus propios vídeos. Lo que se pretendía hacer desde un principio era ofrecerles a los usuarios la posibilidad de grabar su pantalla y audio desde la propia aplicación. De esta manera, no tendrían que recurrir a aplicaciones externas para grabar el vídeo que deseen subir.

A la hora de desarrollar dicha funcionalidad, se elaboraron dos *scripts* en Python que servían para grabar el audio y la pantalla del usuario. La grabación de audio no presentaba ningún inconveniente, pero a la hora de grabar la pantalla éste *script* no funcionaba como se esperaba. Como esta funcionalidad no formaba parte de los objetivos principales, se decidió pausar su desarrollo para continuarlo en un futuro.

Otra funcionalidad que se hubiera querido añadir es la de permitir editar un fichero de vídeo, es decir, realizar cortes, editar las pistas de audio, etc. Esta funcionalidad, por el contrario, no se empezó a desarrollar, ya que abarcaría una gran cantidad de tiempo que se requería para desarrollar funcionalidades necesarias para cumplir los objetivos principales de la aplicación. Sin embargo, se considera que sería interesante poder realizar dichas acciones desde Wishten sin necesidad de tener que usar otras aplicaciones.

Por otra parte, en base a las pruebas realizadas con los usuarios, se ha llegado a la conclusión de que hay dos funcionalidades que se deberían implementar en un futuro cercano.

La primera de ellas está relacionada con la posibilidad de reemplazar los ficheros de vídeo en los vídeos y los de texto en las ofertas. Tal y como está actualmente

desarrollada la aplicación, si se desea cambiar dichos ficheros se tiene que crear el vídeo o la oferta de cero. Dado que han sido varios los usuarios que han destacado esta circunstancia, se estudiará la posibilidad de añadir dicho cambio y la forma de hacerlo.

La segunda de ellas es la relativa al idioma de los tutoriales. El idioma, tanto de la aplicación como de los tutoriales, es el inglés, ya que se considera que, de esta forma, la aplicación podría ser utilizada por un gran número de personas. No obstante, sí que se quiere dar la posibilidad a los usuarios de seleccionar el idioma que deseen de tal forma que tanto la aplicación como los tutoriales estén en el idioma seleccionado. Dado que la cuestión del idioma ha sido bastante resaltada por parte de los usuarios, se le dará prioridad en futuras mejoras de la aplicación.

Por último, se realizará la evaluación diseñada para comprobar la utilidad del uso de cuestionarios dentro de los vídeos en el proceso de aprendizaje de los estudiantes. También se evaluará la utilidad del portal de ofertas, para lo que se buscará contar con la ayuda de algunas empresas que puedan subir diversas ofertas al portal. Finalmente, se les realizará tanto a las empresas como a los usuarios una encuesta de satisfacción para poder ver los puntos a mejorar en el portal.

Una vez realizada la evaluación y añadidas las mejoras mencionadas, se probará a poner la aplicación en producción en un servidor público para comprobar la recepción de los usuarios. No obstante, antes de publicar la aplicación, se realizará una revisión general aplicada al ámbito de la seguridad y del tráfico de usuarios. Para ello se investigarán las herramientas que se pueden utilizar para mejorar dichos aspectos de la aplicación.

# Conclusions

## Conclusions

Generally speaking, most of the goals that were posed before the web application's development were accomplished. The two main goals to be approached by the application are, on the one hand, to provide support to the students through interactive videos visualization and, on the other hand, to facilitate the connection between companies and students by providing a portal to view offers and talk to companies.

In order to achieve the goals set out initially, a web application was successfully developed to incorporate the necessary elements and functionalities.

In this respect, the goal related with the interactive videos visualization was successfully achieved, as the application developed allows users to upload videos, include quizzes in them and search the ones that interest them. The purpose of these videos is that students can reach a better information retention, thanks to the quizzes included on them.

The tests that were carried out and the results obtained have proven that the application has a great usability in this respect, as users found it comfortable and intuitive.

However, it was not possible to carry out a thorough and extended assessment about the usefulness of using quizzes within videos. Nevertheless, this assessment has been designed in order to be carried out in the future.

The other main goal, related with the connection between students and companies, was also successfully achieved. The application allows companies to upload offers which can be viewed or downloaded by the rest of the users. In addition, the offers portal allows users to communicate with companies through a chat linked to each of the offers uploaded by each company.

Thanks to the existence of these chats, that provide more direct communication, and to the existing filters, users can speed up the process of searching offers.

As well as with the videos, the testing carried out have generated positive results in relation to usability and comfort.

Although in the end the main goals were achieved satisfactorily, various difficulties have been encountered in the course of the development, which have been progressively solved.

At first, there were a lot of doubts about the technologies that would be used in the application development. A lot of time was dedicated to investigate which framework best fitted to achieve the previously stated goals. In this way, finally, it was concluded that Laravel would be the most suitable framework to implement the code.

Once the framework was selected, despite of the previous knowledge in PHP, HTML, CSS and JavaScript, in-depth research had to be carried out in order to understand the file structure of Laravel, as well as the commands and syntax used.

When the development had already started, it was considered the idea of implementing a functionality to allow users to record their own videos from the application, giving them the possibility of recording their screen and audio. In order to do that, two Python scripts were used: one for recording audio and one to record the screen. However, despite the fact that the one that recorded audio worked correctly, the one used to record the screen did not work as expected. As the rest of the functionalities were still to be implemented, it was decided to put off this functionality for future work. Therefore, the scripts were kept in the repository for further development in the future.

In relation to video management functionality, some difficulties were encountered when implementing quizzes within videos. Although it was considered to use a JavaScript framework, such as H5P<sup>3</sup>, to achieve this, these frameworks were not compatible with the Laravel version in which the application was being developed. Therefore, at the end, it was decided to implement this functionality from scratch on JavaScript.

Another difficulty encountered was in the creation of the chat, since at the beginning a table structure was proposed in the database, but as it was not possible to achieve what was intended, the table structure had to be redesigned from scratch. It was considered to use Pusher<sup>4</sup>, which facilitates the development of real time applications, however, in order to use it, a lot of modifications on the project structure had to be carried out. For this reason, it was decided to continue with the initial path, but making the necessary modifications on the table structure used for the chats.

One of the difficulties that took more time to resolve it was how to store the files corresponding to videos and offers in Laravel. At first it was not known which Laravel folders were responsible for this functionality, and the instructions needed to implement it. However, after some research and testing, a simple way of storing the files, based on a directory hierarchy, was found.

Although the usage of Laravel did involve a number of difficulties at first, once it was understood how it works, the use of this framework speeded up the development

---

<sup>3</sup><https://h5p.org>

<sup>4</sup><https://pusher.com>

process significantly, thanks to its file structure and its syntax.

Despite all the difficulties encountered throughout the development of the project, the result obtained is considered to be satisfactory, as it successfully fulfils the objectives that were initially set.

## Future work

Even though most of the goals that were set up initially have been satisfactorily achieved, there are some issues and functionalities with the application that would have liked to be improved or extended. However, due to the time planning for the development, they could not be carried out.

One of the functionalities that was started to be developed, but had to be put off in the end, is the one that provided users a space to create their own videos. What was intended to do initially was to offer the users the possibility of recording their screen and audio right from the application itself. In this way, they would not have to depend on external application to record the video they want to upload.

In order to develop such functionality, two Python scripts were made to record the audio and the user's screen. The audio record did not involve any drawbacks, but the script that managed the screen recording did not work as expected. As this was not part of the main goals, it was decided to pause its development in order to continue it in the future.

Another functionality that would have been desirable to add is the one that allows to edit a video file, i.e. to make cuts, edit audio tracks, etc. This functionality, by contrast, was not started to be developed, as it would have involved a large amount of time that was needed to develop the necessary functionalities in order to achieve the main goals of the application. However, it is considered to be interesting to be able to perform such actions from Wishten without having to use other applications.

Moreover, based on user testing, it has been concluded that there are two more functionalities that should be implemented in the near future.

The first of them is the possibility of replacing the video files attached to the videos and the text files attached to the offers. The way the application is developed currently, if a user wish to change such files they need to create the video or the offer from the beginning. Since several users have pointed out this circumstance, the possibility of adding this functionality and the way to do it will be studied.

The second of them is the one related to the tutorials' language. The language of both the application and tutorials is English, as it is considered that, in this way, the application could be used by a large amount of people. However, it is intended to give users the possibility to select the language of their choice so that both the application and the tutorials are in the selected language. Since the language issue has been highlighted by users, it will be prioritized in future improvements of the application.

Finally, the designed evaluation will be carried out in order to verify the useful-

ness of the use of in-video quizzes in the student learning process. The usefulness of the offers portal will also be evaluated, for which it will be sought to have the help of some companies that can upload different offers to the portal. Finally, a satisfaction survey will be conducted with both companies and users to identify areas for improvement in the portal.

Once the evaluation is completed and the aforementioned improvements have been added, the application will be deployed to production on a public server in order to test the reception of the users. However, before publishing the application, a general review applied to the security and user traffic fields will be carried out. For this purpose, tools that can be used to improve such features will be investigated.

# Contribuciones personales

## Roberto Tejedor Moreno

En primera instancia, se tuvo que llevar a cabo una investigación para encontrar los estudios y trabajos previos relacionados con lo que se proponía solucionar mediante el presente proyecto. Para ello, se tuvo que investigar primero la parte relacionada con el aprendizaje a través de vídeos, descubriendo la utilidad que tenía el hecho de añadir cuestionarios y anotaciones en el proceso de aprendizaje de los estudiantes. En segundo lugar, se investigó la utilidad de realizar prácticas en empresa, así como los portales de búsqueda de las mismas.

Para comprobar los problemas que se querían abordar con el desarrollo del presente proyecto, se realizó un estudio previo al desarrollo en el que se crearon dos cuestionarios.

Estas tareas de investigación y estudio previo fueron llevadas a cabo por ambos autores del proyecto.

Siguiendo con el proceso de desarrollo, se definieron en primer lugar los requisitos relativos a las funcionalidades que tendría que tener la aplicación, en segundo lugar, se investigó y seleccionó Laravel como *framework* de desarrollo, y, por último, se creó el proyecto con las funciones básicas, incluyendo la previa estructuración de la base de datos. Todo ello fue realizado por ambos integrantes del proyecto.

Respecto a la redacción de la memoria, la mayoría de esta ha sido redactada en Google Drive<sup>5</sup> estando ambos conectados en remoto. Aquello que no se ha realizado en Google Drive ha sido repartido equitativamente, por lo que la memoria ha sido redactada en partes iguales por ambos integrantes.

Las funcionalidades de la aplicación que se han realizado han sido la de gestión de usuarios, vídeos y cuestionarios dentro de los mismos.

Se implementó toda la lógica relacionada con la gestión de usuarios, planteando tres roles de usuario con diferentes permisos, siendo estos: *standard*, *admin*, y *company*. Para ello, se implementó el registro e inicio de sesión de los usuarios, la modificación de los perfiles y la lógica para que los usuarios se puedan seguir entre

---

<sup>5</sup><https://drive.google.com>

ellos.

Además, creó la página de administración de usuarios para gestionar toda la información relativa a estos desde una cuenta de administrador.

Por otro lado, la funcionalidad relativa a la gestión de vídeos se desarrolló de tal forma que los usuarios pudieran subir vídeos a la aplicación, así como buscarlos y visualizarlos. Los vídeos contienen una serie de campos informativos que ayudan al usuario a decidir qué vídeos le pueden resultar más interesantes.

En este sentido, con el fin de facilitarle al usuario la búsqueda de vídeos, se implementó la manera de filtrar los vídeos en base a diversos parámetros mediante consultas a la base de datos.

Además, se desarrolló el código necesario para almacenar los ficheros de vídeo en el servidor de manera ordenada, mediante un sistema de directorios. Por último, se implementó la página de administración de vídeos, siguiendo la misma estructura que la de usuarios.

En relación a la gestión de los cuestionarios, se definió una estructura para almacenar las preguntas y respuestas de los cuestionarios, así como las respuestas que los usuarios seleccionan.

Una de las dificultades encontradas en este apartado fue el desarrollo de la lógica para mostrar los cuestionarios, así como almacenar las respuestas que daban los usuarios. Para ello, se crearon ficheros JavaScript que muestran las preguntas (teniendo en cuenta que las que no tienen respuesta, se cuentan como anotaciones) y almacenan las respuestas que va proporcionando el usuario durante la visualización de un vídeo.

Además, se implementó un algoritmo para que, cuando se envíen dichas respuestas, se calcule el número de preguntas acertadas y se almacene en la tabla de visualizaciones. De la misma manera, se desarrolló todas las vistas necesarias para poder visualizar las respuestas que dan los usuarios, así como para interactuar con los cuestionarios, es decir, para crearlos y modificarlos. Para crear los cuestionarios, también empleó JavaScript, para poder recoger el instante del vídeo en el que se quiere añadir la pregunta, así como algunos efectos visuales que hacen más intuitiva dicha tarea.

En cuanto a la redacción del presente documento, se le dio formato utilizando para ello LaTeX, un software para formatear documentos. Para ello, se empleó la plantilla proporcionada por la Universidad Complutense de Madrid. La principal dificultad en este sentido fue colocar las figuras y tablas de modo que siguieran el flujo del texto, ya que dependiendo del tamaño de las mismas, éstas se colocan automáticamente, por lo que fue necesario adaptar el texto a las figuras y viceversa.

Seguidamente, ambos autores diseñaron las pruebas a realizar y recogieron y analizaron los resultados obtenidos en las mismas. Ambos se encargaron de supervisar a los usuarios de prueba y recoger los comentarios y retroalimentación que éstos aportaron durante la realización de las acciones requeridas. De la misma manera, se elaboró el diseño de la evaluación, detallando cómo se pretendía llevar a cabo.

## Enrique Martín Rodríguez

En primera instancia, se tuvo que llevar a cabo una investigación para encontrar los estudios y trabajos previos relacionados con lo que se proponía solucionar mediante el presente proyecto. Para ello, se tuvo que investigar primero la parte relacionada con el aprendizaje a través de vídeos, descubriendo la utilidad que tenía el hecho de añadir cuestionarios y anotaciones en el proceso de aprendizaje de los estudiantes. En segundo lugar, se investigó la utilidad de realizar prácticas en empresa, así como los portales de búsqueda de las mismas.

Para comprobar los problemas que se querían abordar con el desarrollo del presente proyecto, se realizó un estudio previo al desarrollo en el que se crearon dos cuestionarios.

Estas tareas de investigación y estudio previo fueron llevadas a cabo por ambos autores del proyecto.

Siguiendo con el proceso de desarrollo, se definieron en primer lugar los requisitos relativos a las funcionalidades que tendría que tener la aplicación, en segundo lugar, se investigó y seleccionó Laravel como *framework* de desarrollo, y, por último, se creó el proyecto con las funciones básicas, incluyendo la previa estructuración de la base de datos. Todo ello fue realizado por ambos integrantes del proyecto.

Respecto a la redacción de la memoria, la mayoría de esta ha sido redactada en Google Drive estando ambos conectados en remoto. Aquello que no se ha realizado en Google Drive ha sido repartido equitativamente, por lo que la memoria ha sido redactada en partes iguales por ambos integrantes.

Las funcionalidades de la aplicación que se han realizado han sido la de gestión de ofertas, de chat y la realización de la mayor parte de las vistas de la página web.

En relación con la gestión de ofertas, esta se realizó de tal forma que los usuarios que fueran empresa pudieran subir ofertas a la aplicación y que el resto de los usuarios pudieran ver y descargar dichas ofertas. Las ofertas cuentan una serie de campos informativos que ayudan al usuario a tener una idea previa sobre la oferta antes de abrir si quiera el documento de esta.

Con el fin de agilizar el proceso de búsqueda de ofertas, al usuario se le ofrece la opción de buscar éstas por título o por salario. La principal idea detrás de ofrecer estos dos tipos de búsqueda es combinar ambas búsquedas de tal forma que el usuario busque aquel ámbito en el que le gustaría filtrar ofertas y dentro de todas las ofertas de ese ámbito ver cuáles son las que más o menos salario ofrecen.

La mayor dificultad que se tuvo a la hora de desarrollar la funcionalidad de las ofertas fue entender cómo y dónde se guardaban los archivos, ya que para poder los usuarios ver los archivos de las ofertas o descargarlos, se necesitaba saber el lugar exacto en el que éstos se almacenaban. También había que tener claro qué tipo de archivos de texto se iba a permitir que subieran las empresas, así como el nombre que se le iba a dar a dicho archivo a la hora de almacenarlo.

Al añadir esta funcionalidad se tuvo que modificar algunos aspectos de la gestión

de usuarios así cómo añadir las acciones pertinentes a los administradores y empresas sobre la gestión de ofertas.

La gestión de chat está muy relacionada con las ofertas, ya que cada oferta tiene una interacción usuario-empresa que es única. Cada oferta puede tener varios chats de diferentes personas y una sola persona puede tener chats con varias empresas distintas.

Hay que destacar de esta funcionalidad que el chat se crea en el momento en el que el usuario le da al botón de chat asociado a cada oferta. De esta manera el usuario y la empresa van a tener chats vacíos, pero sólo se mostrarán aquellos que estén activos, es decir, aquellos que cuentan con al menos un mensaje.

El principal problema que se encontró a la hora de desarrollar el chat fue el crear el chat entre la empresa y el usuario, ya que había que tener claro quién lo creaba y qué mensaje era enviado por qué persona. Para evitar los problemas que fueron surgiendo durante el desarrollo de esta funcionalidad, se decidió que los usuarios con el rol *standard* fueran los únicos capaces de crear el chat, ya que se considera que los usuarios con el rol *standard* van a ser siempre los que inicien la conversación, al ser estos los interesados en hablar con las empresas que han ofertado las prácticas.

Al desarrollar la funcionalidad de chat se tuvo que modificar un poco la funcionalidad relacionada con la gestión de usuarios y añadir las acciones relacionados con la gestión de chat a los usuarios con el rol de administración y empresa.

Durante el desarrollo de las vistas, se buscó conseguir que estuvieran bien estructuradas y fueran lo más intuitivas y atractivas posible para los usuarios. Para conseguir esto, se tuvo muy en cuenta el diseño web de la aplicación, todo ello con el fin de poder causar así una impresión positiva en los usuarios que visiten y utilicen la aplicación web. Por ello, se organizaron los elementos de la forma que se explica en el apartado de las vistas.

En un principio, no se tenía muy claro cómo y qué utilizar para representar los iconos en la página web. Sin embargo, tras investigar sobre que se solía utilizar a la hora de añadir iconos en páginas web y ver los beneficios que el uso de los iconos procedentes de Font Awesome aportaban al desarrollo de una aplicación web, se decidió utilizar dichos iconos para el desarrollo de las vistas.

Cabe destacar que, si bien hay numerosas vistas, muchas de ellas son parecidas entre sí, contacto con pequeños ajustes que se les realizaban a medida que cada uno iba realizando sus respectivas funcionalidades.

Los vídeos que se pueden encontrar en la página de tutoriales fueron editados utilizando el software de edición gratuito Shotcut<sup>6</sup>.

Seguidamente, ambos autores diseñaron las pruebas a realizar y recogieron y analizaron los resultados obtenidos en las mismas. Ambos se encargaron de supervisar a los usuarios de prueba y recoger los comentarios y retroalimentación que éstos aportaron durante la realización de las acciones requeridas. De la misma manera, se elaboró el diseño de la evaluación, detallando cómo se pretendía llevar a cabo.

---

<sup>6</sup><https://www.shotcut.org>

# Bibliografía

- Alsolais, A. M. (2022). Development of a web-based student internship portal for students of health colleges. *Fortune Journal of Health Sciences*, 5(3): 451–460. <https://doi.org/10.26502/fjhs.072>.
- Binder, J. F., Baguley, T., Crook, C., & Miller, F. (2015). The academic value of internships: Benefits across disciplines and student backgrounds. *Contemporary Educational Psychology*, 41: 73–82. <https://doi.org/10.1016/j.cedpsych.2014.12.001>.
- Django (2023). Writing your first Django app, part 1 | Django documentation | Django. Consultado el 21 de julio de 2023. <https://docs.djangoproject.com/en/4.2/intro/tutorial01/>.
- Gargallo, B., Rodríguez, J. M. S., García, E., Pérez, C., & Sahuquillo, P. M. (2012). Enfoques de aprendizaje en estudiantes universitarios excelentes y en estudiantes medios. *Revista Española De Pedagogía*, 70(252): 185–200.
- Haagsman, M. E., Scager, K., Boonstra, J., & Koster, M. C. (2020). Pop-up questions within educational videos: Effects on students' learning. *J Sci Educ Technol*, 29(6): 713–724. <https://doi.org/10.1007/s10956-020-09847-3>.
- Laravel (2023a). Blade templates - Laravel - The PHP Framework For Web Artisans. Consultado el 13 de julio de 2023. <https://laravel.com/docs/9.x/blade>.
- Laravel (2023b). Eloquent: Getting started - Laravel - The PHP Framework For Web Artisans. Consultado el 13 de julio de 2023. <https://laravel.com/docs/9.x/eloquent>.
- Laravel (2023c). File storage - Laravel - The PHP Framework For Web Artisans. Consultado el 14 de julio de 2023. <https://laravel.com/docs/9.x/filesystem>.
- Laravel (2023d). Installation - Laravel - The PHP Framework For Web Artisans. Consultado el 21 de julio de 2023. <https://laravel.com/docs>.
- Laravel (2023e). Middleware - Laravel - The PHP Framework For Web Artisans. Consultado el 15 de julio de 2023. <https://laravel.com/docs/9.x/middleware>.

- Laravel (2023f). Release notes - Laravel - The PHP Framework For Web Artisans. Consultado el 30 de julio de 2023. <https://laravel.com/docs/10.x/releases>.
- Liao, C.-H. & Wu, J.-Y. (2023). Learning analytics on video-viewing engagement in a flipped statistics course: Relating external video-viewing patterns to internal motivational dynamics and performance. *Computers & Education*, 197: 104754. <https://doi.org/10.1016/j.compedu.2023.104754>.
- Mirriahi, N., Jovanović, J., Lim, L.-A., & Lodge, J. M. (2021). Two sides of the same coin: video annotations and in-video questions for active learning. *Education Tech Research Dev*, 69(5): 2571–2588. <https://doi.org/10.1007/s11423-021-10041-4>.
- Moos, D. C. & Bonde, C. (2016). Flipping the classroom: Embedding self-regulated learning prompts in videos. *Tech Know Learn*, 21(2): 225–242. <https://doi.org/10.1007/s10758-015-9269-1>.
- Rodríguez, M. S. & Platas-García, A. (2022). Uso de videos tutoriales en el proceso de aprendizaje de estudiantes universitarios. *Revista Electrónica de Investigación Educativa*, 24. <https://doi.org/10.24320/redie.2022.24.e21.4176>.

## Cuestionarios de estudio previo

### A.1. Cuestionario sobre método de estudio

En primer lugar, en las figuras A.1 y A.2 se observan las preguntas referentes a la edad y el sexo de los encuestados. Se puede apreciar que en su mayoría son hombres de entre dieciocho y veinticuatro años.

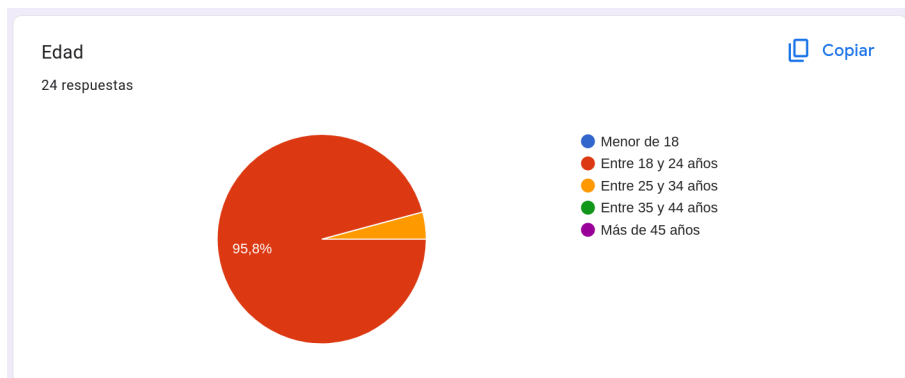


Figura A.1: Cuestionario sobre método de estudio - Edad de los encuestados

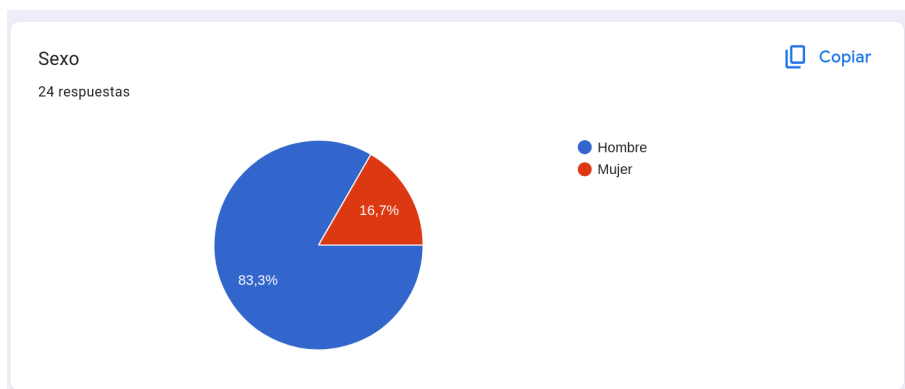


Figura A.2: Cuestionario sobre método de estudio - Sexo de los encuestados

Seguidamente, en las figuras A.3 y A.4 se muestran las preguntas referentes a los estudios que realizan los encuestados y cuántas horas le dedican. Se puede observar que la mayoría cursan estudios relacionados con la informática, ya que muchos de los encuestados son estudiantes de la Facultad de Informática de la Universidad Complutense de Madrid.

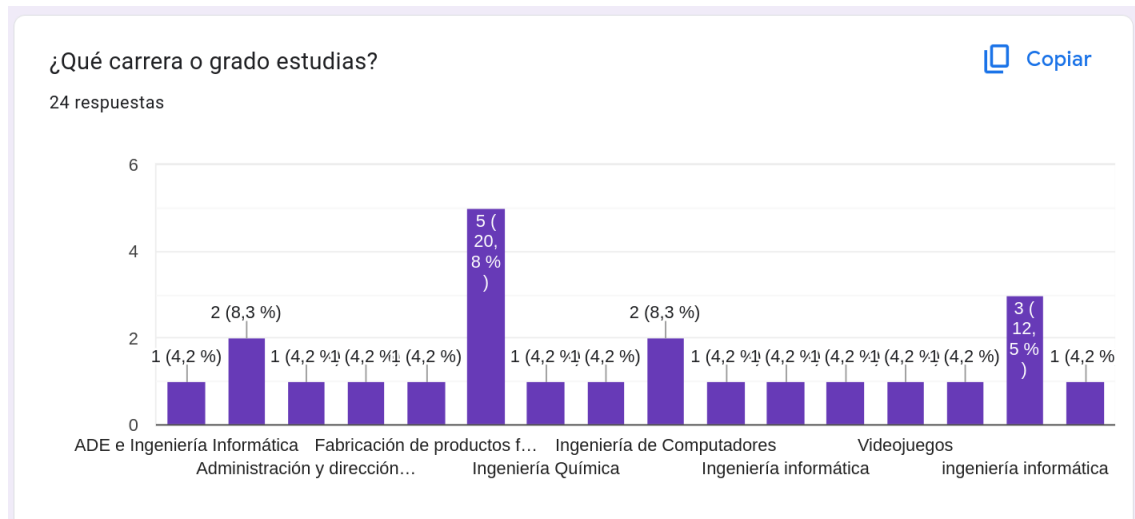


Figura A.3: Cuestionario sobre método de estudio - Estudios de los encuestados

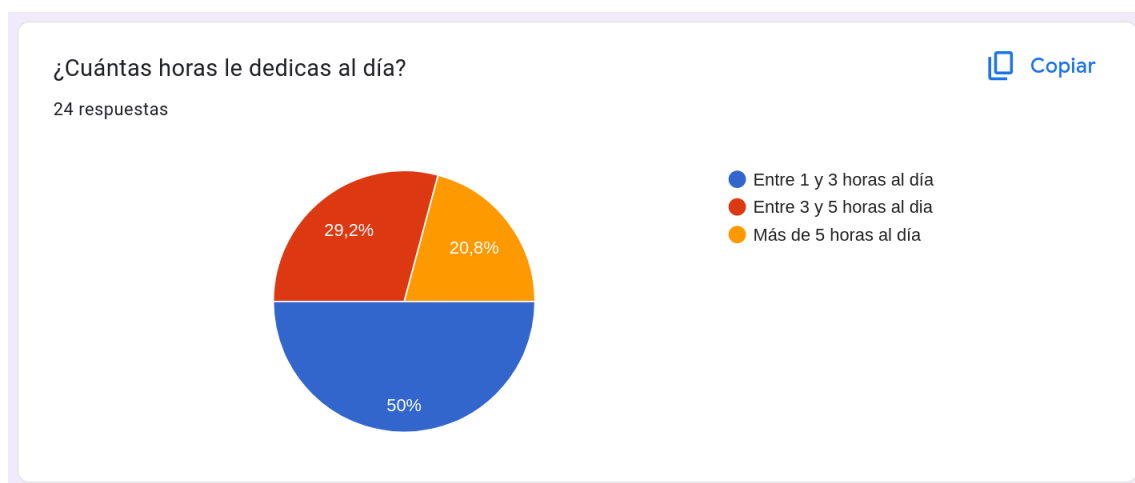


Figura A.4: Cuestionario sobre método de estudio - Horas de estudio de los encuestados

En la figura A.5 se muestra la pregunta principal del cuestionario, relativa al método de estudio. En su mayoría, los participantes contestaron que estudian centrándose en los ejercicios de examen, en lugar de estudiar el temario completo.

A los participantes que contestaron que estudian el temario completo, se les preguntó si habían pensado alguna vez en cambiar de método y estudiar únicamente los ejercicios de examen. Tal como se ve en la figura A.6, la mayor parte de ellos confirmaron dicho pensamiento.

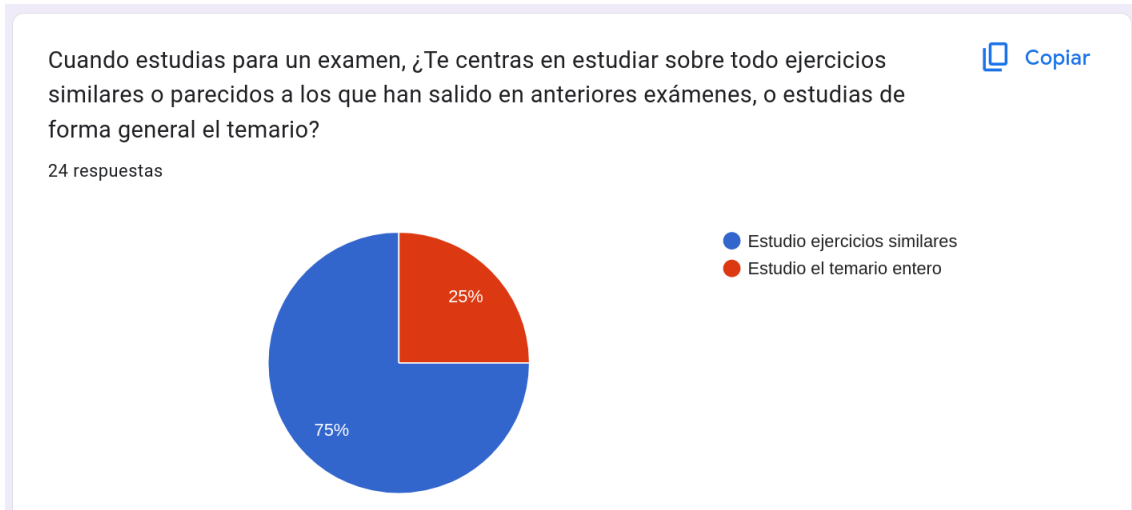


Figura A.5: Cuestionario sobre método de estudio - Método de estudio de los encuestados

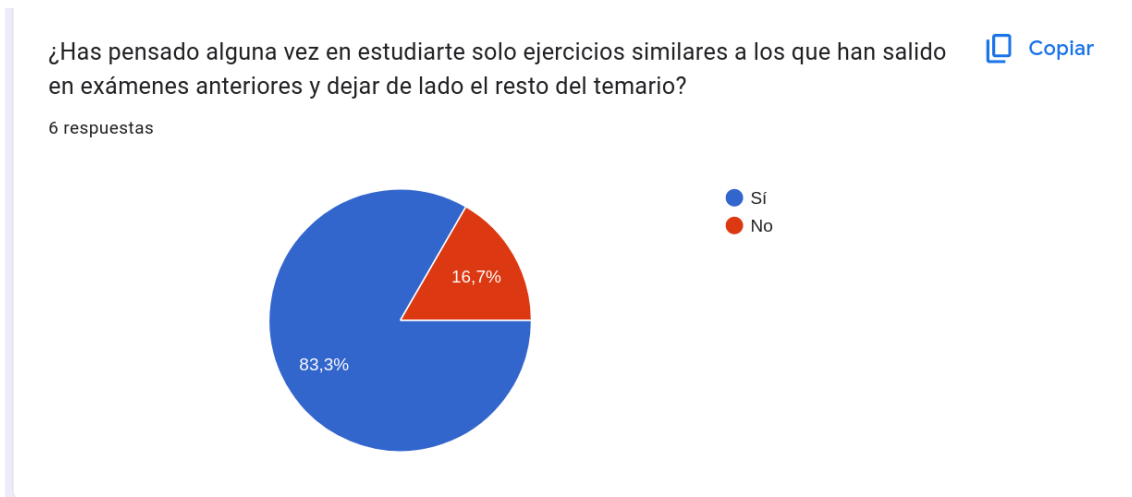


Figura A.6: Cuestionario sobre método de estudio - Reflexión sobre el método utilizado

Por su parte, a los encuestados cuyo método de estudio se basaba en estudiar ejercicios similares a los de examen, se les preguntó si este método les había perjudicado en alguna ocasión. Como se muestra en la figura A.7, la mayoría coinciden en que sí les ha perjudicado estudiar de esta forma.

Finalmente, a estos participantes también se les preguntó si volverían a estudiar únicamente los ejercicios de examen, a lo que la mayoría contestaron afirmativamente, tal como se refleja en la figura A.8.

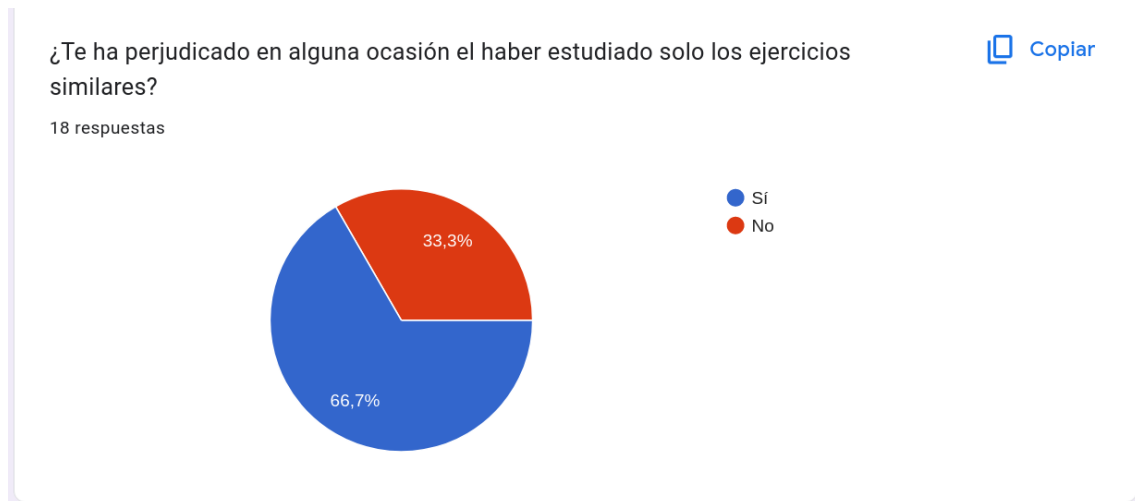


Figura A.7: Cuestionario sobre método de estudio - Perjuicio del método utilizado

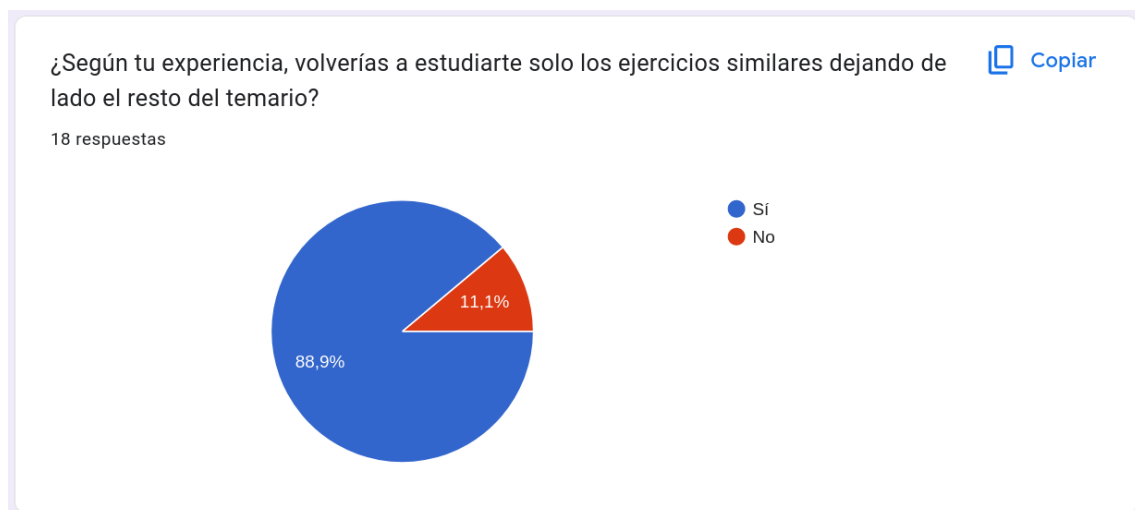


Figura A.8: Cuestionario sobre método de estudio - Repetición del método utilizado

## A.2. Cuestionario sobre búsqueda de prácticas

Las tres primeras preguntas referentes a la edad, sexo y estudios de los encuestados obtuvieron los mismos resultados, puesto que se realizó el estudio sobre los mismos estudiantes.

A continuación, se les preguntó a los participantes si habían realizado o buscado prácticas en empresas, a lo que la mayoría contestó afirmativamente, tal como se ve en la figura A.9.

Seguidamente, a los encuestados que sí habían realizado prácticas o las habían buscado se les preguntó por la facilidad para encontrar un portal de búsqueda de prácticas. Las respuestas a esta pregunta se midieron con una escala de valoración, de “Muy difícil” a “Muy fácil”, tal como se muestra en la figura A.10

Como se observa en la figura , la mayor parte de los encuestados respondieron a

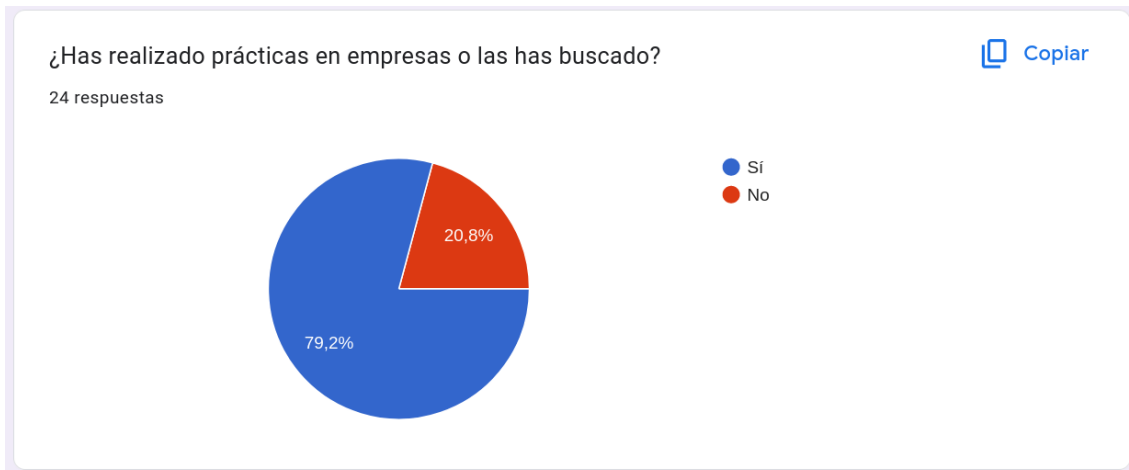


Figura A.9: Cuestionario sobre búsqueda de prácticas - Realización o búsqueda de prácticas

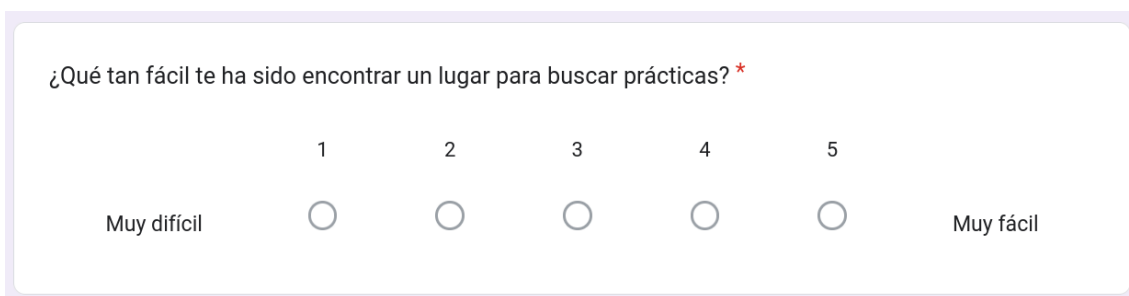


Figura A.10: Cuestionario sobre búsqueda de prácticas - Pregunta sobre portal de búsqueda

esta pregunta con el valor 2, lo que se interpreta como “difícil” dentro de la escala de valoración.



Figura A.11: Cuestionario sobre búsqueda de prácticas - Respuestas sobre portal de búsqueda

Finalmente, se le preguntó a los participantes si su centro les había proporcionado un portal para realizar la búsqueda de prácticas, tal como se ve en la figura A.12. La mayoría de ellos respondió afirmativamente y se les preguntó por la utilidad de dicho portal, utilizando la misma escala de valoración usada anteriormente. Tal como se muestra en la figura A.13, la mayoría de ellos se mostraron neutrales o descontentos con el portal proporcionado.



Figura A.12: Cuestionario sobre búsqueda de prácticas - Portal de búsqueda proporcionado por el centro de estudios

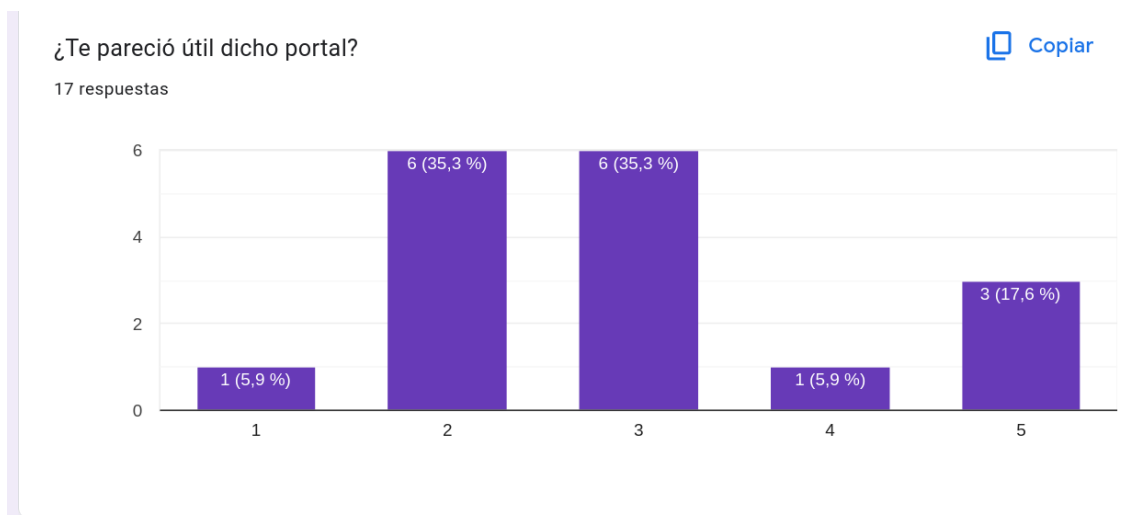


Figura A.13: Cuestionario sobre búsqueda de prácticas - Utilidad del portal proporcionado por el centro de estudios