

# PROYECTO DE PROYECTOS

Lucía Hurtado Cardoso

## DOCUMENTACIÓN DEL PROYECTO

Desarrollo de Aplicaciones Web  
2019/20

# ÍNDICE

1. Introducción	.....2
2. Planteamiento del problema	.....2
3. Solución del problema	.....3
4. Análisis de la aplicación	.....4
5. Diseño de la aplicación	.....13
6. Manuales	.....17
7. Gestión del proyecto	.....17
8. Conclusiones	.....17
9. Bibliografía	.....18

## 1. Introducción

Durante una conversación con la que ahora es mi tutora de proyecto, me comentó que llevaba algún tiempo queriendo llevar a cabo un proyecto de estas características, me pareció un reto muy interesante, ya que vi la posibilidad realizarlo con tecnologías bastante actuales y que llevaba tiempo queriendo aprender.

La finalidad del 'Proyecto de proyectos' es mejorar la comunicación entre profesor y alumno durante la realización del proyecto de fin de grado.

Para la elaboración del mismo me he tenido que enfrentar a muchos problemas, ya que no estaba familiarizada a las tecnologías utilizadas. Pero esto ha hecho que aumentaran mis ganas de superarme y de llevar el proyecto siempre un poco más allá de lo que me planteé en un principio.

## 2. Planteamiento del problema

Cuando comencé a plantearme cómo iba a llevar a cabo el proyecto y que tecnologías pensaba usar, tenía muy claro que algo fundamental era aprender lo máximo posible mientras desarrollaba el mismo. Lo cual conllevaba usar tecnologías que no había usado con anterioridad y debía enfrentarme a infinidad de problemas, los cuales no estaba acostumbrada a solventar.

A pesar de todos los miedos que me abordaban dadas las circunstancias, tenía claro las tecnologías que iba a usar para mi proyecto: **Spring5** para el backEnd y **Angular** para el frontEnd.

¿Por qué Spring5? Pues bien, esta tecnología ofrece una conexión con la base de datos fácil y eficiente, con una seguridad máxima y, además, es una de las tecnologías más potentes y más usadas hoy en día.

¿Por qué Angular? La estructura de Angular está basada en componentes, lo que hace que sean reutilizables y ayuda a que el mantenimiento y la actualización eficiente de código sean posible. Incluso las aplicaciones de Google han optado por utilizar este framework.

### 3. Solución del problema

En un proyecto de tal envergadura, es fundamental extraer del usuario y del cliente toda la información relevante para el mismo. A su vez, es importante tener en cuenta todos aquellos análisis previamente realizados.

El momento de plasmar de forma efectiva lo que previamente hemos planteado en nuestra mente, es algo fundamental, ya que, si no somos capaces de llevarlo a cabo, no conseguiremos nunca el resultado esperado.

Para la **gestión de los datos** se ha usado una base de datos relacional, la cual ha sido gestionada por MySQL. Para el desarrollo de la misma me he apoyado en MySQL Workbench, una herramienta visual de diseño de base de datos que integra desarrollo de software, administración de base de datos, diseño de la misma, gestión y almacenamiento...

El **backEnd** (servicio REST) de la aplicación ha sido realizado con Spring5, ya que es uno de los frameworks más potentes de Java y ofrece infinidad de facilidades. El entorno de desarrollo utilizado ha sido Eclipse con el plugin STL (Spring Tool Suite) que hace que la programación de Spring en Eclipse sea mucho más fluida.

Para la conexión entre la base de datos y el mismo, he utilizado un módulo de Spring llamado 'Spring Data'. La misión de este se basa en proporcionar un modelo de programación familiar y consistente y facilita el uso de tecnologías de acceso a datos, por ello conecta a la perfección con el tipo de base de datos utilizada.

Spring Data entre otras cosas, nos proporciona un potente repositorio para el mapeo de objetos, 'JpaRepository', el cual ha sido de gran utilidad a la hora de realizar operaciones con la base de datos.

Para la salida de los datos al frontEnd, he utilizado un módulo de Spring llamado 'Spring Rest', que facilita la devolución de los datos al frontEnd en forma de JSON.

He implementado, también, 'Spring Security' con JWT, para la gestión del acceso a los recursos, organizando a los usuarios con roles.

Con todo esto y más detalles que se explicarán en otros puntos a continuación, se puede decir, que el backEnd de mi aplicación es, en definitiva, una API REST.

El **frontEnd** de la aplicación ha sido realizado con Angular, ya que es uno de los frameworks más potentes en la actualidad para el desarrollo de aplicaciones frontEnd. Angular trabaja con TypeScript, lo que convierte a JavaScript en un lenguaje orientado a objeto, esto potencia el uso del mismo. También, como mencione anteriormente, la organización del código en componentes ayuda a optimizar el código. El entorno de desarrollo usado ha sido Visual Studio Code.

## 4. Análisis de la aplicación

Los requisitos iniciales para la aplicación son:

- Permitir la comunicación entre alumno y profesor.
- Que el usuario pueda registrarse como 'profesor' o como 'alumno'.
- En el caso de que el usuario se registre como 'profesor' deberá introducir un código para verificar que este es profesor y que no cualquier persona pueda registrarse con este rol.
- Que la cuenta de email sea verificada.
- Que el usuario pueda editar su perfil.
- Que el rol 'alumno' pueda crear un proyecto, editar el mismo, crear una nueva publicación, etc...
- Facilitar el seguimiento del proyecto al profesor, con comentarios y que este pueda calificar el proyecto.
- Una vez calificado el proyecto por el profesor, si la nota de este es igual o mayor al 5, este se publicará para que alumnos de años posteriores puedan ayudarse y coger ideas de él.
- Los proyectos publicados estarán categorizados por lenguajes, grados...
- Debe existir un usuario con rol 'admin' que pueda acceder a todos los apartados de la aplicación y mostrar/ocultar proyectos del apartado de 'proyectos finalizados con éxito'.

Teniendo en cuenta estos requisitos iniciales, resultan los siguientes casos de uso:

- En la **página inicial/login** el usuario puede:
  - \*Puede acceder todo el mundo.*
  - Autenticarse con su email y contraseña y acceder a la aplicación.
  - Hacer clic en '¿No estás registrado?' y acceder a la página de registro.
  - Hacer clic en 'Olvidé mi contraseña' para restablecer la misma.
  - Hacer clic en 'Proyecto de proyectos' y acceder a los proyectos públicos.
  - Mostrar/ocultar la contraseña.

- Cometer algún error al introducir sus datos, en este caso la aplicación le informará de que el email o la contraseña no son válidos.
  - Clicar en el botón 'LOGIN' sin haber rellenado anteriormente los campos 'email' y 'contraseña', en ese caso el programa le informará que no puede dejar esos campos vacíos.
- En la **página de registro** el usuario puede:

*\*Puede acceder todo el mundo.*

- Hacer clic en 'Proyecto de proyectos' y acceder a los proyectos públicos.
  - Hacer clic en 'Iniciar sesión' y acceder a la página de login.
  - Seleccionar un rol, 'alumno' o 'profesor'.
  - En el caso de que el usuario deje vacío algún campo, se le avisará que dicho campo no puede estar vacío y el botón registro estará deshabilitado.
  - Si el usuario introduce un email ya existente en la base de datos, se informará al usuario que ya existe una cuenta con dicho email y no se procederá al registro.
  - Mostrar/ocultar contraseña.
  - Si la contraseña introducida y la confirmación de la misma no coincide, se informará al usuario y no se procederá al registro.
  - Una vez introducidos bien todos los datos, se procede al registro del usuario con la cuenta 'deshabilitada' temporalmente hasta que el usuario verifique el email.
- En la **página de verificación** el usuario puede:

*\*Solo podrá acceder la persona a la que le haya llegado el correo de verificación.*

- Clicar en el botón verificar email y proceder a iniciar sesión.
- En la **página de restablecer contraseña** el usuario puede:
- \*Solo podrá acceder la persona a la que le haya llegado el correo de verificación.*
- Introducir una nueva contraseña y proceder a iniciar sesión.

- En el **perfil del usuario con rol 'alumno'** este puede:

*\*Puede acceder cualquier usuario autenticado con rol 'profesor' aunque no tendrá acceso a algunas opciones, el usuario con rol 'admin' y solo el usuario con rol 'alumno' al que pertenezca el perfil.*

- Hacer clic en 'Proyecto de proyectos' y acceder a los proyectos públicos.
- Hacer clic en su nombre y clicar en 'Mi perfil' que lo llevaría a su perfil de nuevo o en 'Cerrar sesión' que cerraría la sesión del mismo.
- Hacer clic en el icono 'Editar perfil'.
- Hacer clic en el icono 'Cambiar contraseña'.
- En caso de que el usuario no tenga proyectos creados aun o que los tenga creados, pero estos estén cerrados, el usuario podrá clicar en 'Crear nuevo proyecto' para crear un nuevo proyecto.
- Hacer clic en 'Editar proyecto' y editar el mismo, siempre que este esté abierto.
- Hacer clic en la URL de GitHub que aparece en cada proyecto y acceder a la página en cuestión.
- Clicar en 'Ver más' para acceder a la página del proyecto en cuestión.

- En el **modal que se abre tras clicar en 'Editar perfil'** el usuario puede:

*\*Cada usuario autenticado accede a la edición de su propio perfil.*

- Cambiar nombre o apellidos.
- Subir una foto de perfil clicando encima de la misma.
- Guardar los cambios haciendo clic en el botón 'Editar la información del perfil'.
- Cerrar el modal sin guardar los cambios clicando en la cruz que aparece en la parte superior derecha.

- En el **modal que se abre tras clicar en 'Cambiar contraseña'** el usuario puede:

*\*Cada usuario autenticado accede a la edición de su propia contraseña.*

- Cambiar su contraseña.
- Cerrar el modal sin guardar los cambios clicando en la cruz que aparece en la parte superior derecha.

- En el **modal que se abre tras clicar en ‘Editar proyecto’** el usuario puede:

*\*Solo los usuarios autenticados con rol ‘alumno’ tendrán esta opción y solo para los proyectos propios.*

- Modificar el título del proyecto.
- Modificar la descripción del proyecto.
- Modificar la URL de GitHub.
- Eliminar tecnologías.
- Añadir tecnologías ya existentes en la base de datos.
- Añadir tecnologías nuevas que no existen en la base de datos (se añadirán a la misma y se creará la relación entre esta nueva tecnología y el proyecto en cuestión).
  - Si la tecnología que el usuario a introducido ya existe en la base de datos, no se creará una nueva, sino que se creará la relación entre esta y el proyecto en cuestión con la ya existente en la base de datos.
  - Si la tecnología que el usuario introduce ya existe en la base de datos y además ya está relacionada con el proyecto en cuestión el programa no añadirá una nueva tecnología a la base de datos ni al proyecto.
- Seleccionar un ciclo.
- Seleccionar una convocatoria.
- Guardar los cambios haciendo clic en el botón ‘Editar la información del proyecto’.
- Cerrar el modal sin guardar los cambios clicando en la cruz que aparece en la parte superior derecha.

- En el **modal que se abre tras clicar en ‘Crear nuevo proyecto’** el usuario puede:

*\*Solo los usuarios autenticados con rol ‘alumno’ que o bien no tengan ningún proyecto aun o que los demás estén cerrados. El nuevo proyecto se le asignará al alumno que esté logueado en este momento.*

- Rellenar toda la información del proyecto (título, descripción, url GitHub, seleccionar tecnologías de la base de datos, añadir tecnologías nuevas, seleccionar un ciclo y seleccionar una convocatoria).



- En caso de que el usuario deje algún campo sin rellenar se le avisará de que dicho campo no puede estar vacío y no se procederá a la creación del nuevo proyecto.
  - Crear el proyecto haciendo clic en el botón 'Crear nuevo proyecto'.
  - Cerrar el modal sin crear el proyecto clicando en la cruz que aparece en la parte superior derecha.
- En el **perfil del usuario con rol 'profesor'** este puede:

*\*Solo puede acceder el usuario autenticado con rol 'profesor' dueño de la cuenta y el usuario con rol 'admin' con algunas restricciones.*

- Hacer clic en 'Proyecto de proyectos' y acceder a los proyectos públicos.
  - Hacer clic en su nombre y clicar en 'Mi perfil' que lo llevaría a su perfil de nuevo o en 'Cerrar sesión' que cerraría la sesión del mismo.
  - Hacer clic en el icono 'Editar perfil'.
  - Hacer clic en el icono 'Cambiar contraseña'.
  - Clicar en el botón 'Asignar nuevo proyecto' y acceder a la página donde le aparecerán todos los proyectos sin asignar aun a un tutor.
  - Hacer clic en la URL de GitHub que aparece en cada proyecto y acceder a la página en cuestión.
  - Hacer clic en el nombre del usuario y acceder al perfil del mismo (Al profesor no le saldrán las opciones de 'editar perfil', 'editar proyecto' o 'crear nuevo proyecto' cuando acceda al perfil del usuario).
  - Clicar en 'Ver más' para acceder a la página del proyecto en cuestión.
  - Hacer clic en 'Calificar proyecto' y calificar el mismo.
- En la **página de proyectos públicos** el usuario puede:
- \*Puede acceder todo el mundo.*
- Ver todos los proyectos publicados de otros años que han sido calificados con una nota igual o mayor al 5.
  - Acceder a la URL de GitHub especificada.

- Descargar el archivo adjunto de un proyecto en cuestión, este archivo es un .zip que contiene los archivos que el tutor de proyecto ha creído interesantes.
- Hacer clic en su nombre y clicar en 'Mi perfil' que lo llevaría a su perfil o en 'Cerrar sesión' que cerraría la sesión del mismo.
- Hacer clic en 'Proyecto de proyectos' que lo llevaría a la página actual de nuevo.
- Si el usuario que accede tiene rol 'admin', tendrá un botón en la parte superior derecha de cada proyecto para ocultarlo si lo cree necesario.
- Filtrar los proyectos por **ciclo, convocatoria, año o tecnologías**.

- En la **página asignar proyectos** el usuario puede:

*\*Solo puede acceder el usuario con rol 'profesor' y rol 'admin'.*

- Clicar en la opción 'asignar proyecto' y asignarse dicho proyecto como tutor.
- Acceder a la URL de GitHub especificada.
- Acceder al perfil del alumno en cuestión (Al profesor no le saldrán las opciones de 'editar perfil', 'editar proyecto' o 'crear nuevo proyecto' cuando acceda al perfil del usuario).
- Hacer clic en su nombre y clicar en 'Mi perfil' que lo llevaría a su perfil o en 'Cerrar sesión' que cerraría la sesión del mismo.
- Hacer clic en 'Proyecto de proyectos' que lo llevaría a la página de proyectos publicados.
- Filtrar los proyectos por **ciclo, convocatoria o año**.

- En la **página del proyecto** el usuario puede:

*\*Solo pueden acceder el usuario autenticado con rol 'alumno' al que pertenece el proyecto, el usuario autenticado con rol 'profesor' tutor de dicho proyecto y el usuario con rol 'admin'.*

- Ver la información referente al proyecto (título, descripción, url GitHub, alumno, convocatoria, ciclo, tecnologías y tutor).
- Ver las publicaciones que se han hecho referentes a dicho proyecto.
- Descargar los archivos adjuntos que contienen las publicaciones, en caso de que los haya.
- Ver comentarios que se han hecho referentes a las publicaciones.

- Hacer un nuevo comentario.
  - Si existen más de dos comentarios en una publicación, se podrán mostrar/ocultar los demás comentarios.
  - Si el usuario autenticado tiene rol 'alumno', podrá crear una nueva publicación.
  - Hacer clic en su nombre y clicar en 'Mi perfil' que lo llevaría a su perfil o en 'Cerrar sesión' que cerraría la sesión del mismo.
  - Hacer clic en 'Proyecto de proyectos' que lo llevaría a la página de proyectos publicados.
- En el **modal que se abre tras clicar en 'Crear nueva publicación'** el usuario puede:

*\*Solo puede acceder el usuario con rol 'alumno' al que pertenezca el proyecto.*

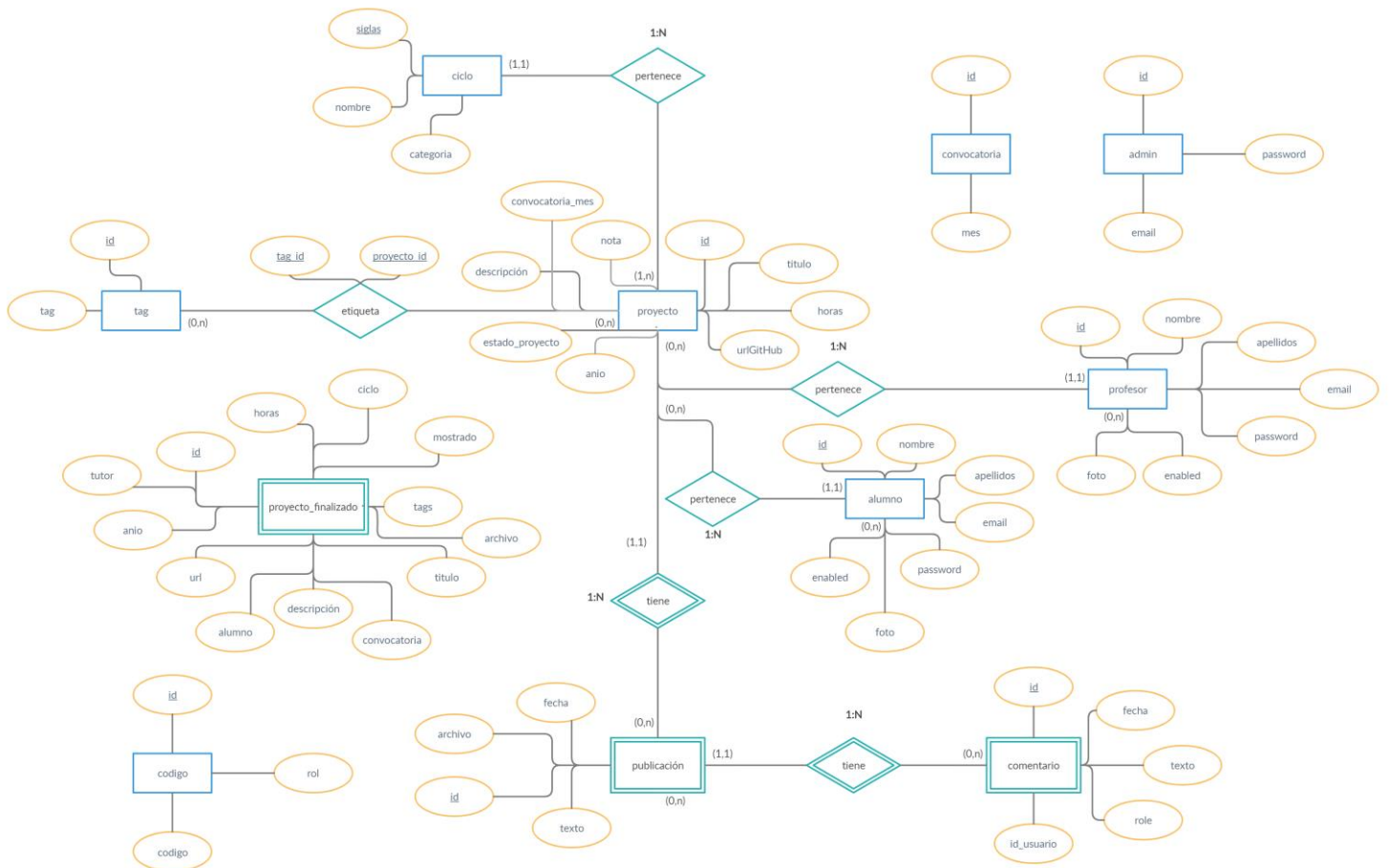
- Rellenar el campo: breve descripción del motivo de la publicación (obligatorio) y seleccionar un archivo para subir asociado a la publicación (opcional)
  - En caso de que el usuario deje el campo obligatorio sin rellenar se le avisará de que dicho campo no puede estar vacío y no se procederá a la creación de la nueva publicación.
  - Crear la publicación haciendo clic en el botón 'Crear nueva publicación'.
  - Cerrar el modal sin crear la publicación clicando en la cruz que aparece en la parte superior derecha.
- En el **modal que se abre tras clicar en 'Calificar proyecto'** el usuario puede:

*\*Solo puede acceder el usuario autenticado con rol 'profesor' que sea tutor del proyecto en cuestión.*

- Ver toda la información referente al proyecto en cuestión.
- Acceder a la URL de GitHub especificada.
- Rellenar los campos 'horas' y 'nota'.
- Si el usuario no rellena alguno de los dos campos anteriormente especificados, se informará al usuario con un mensaje y se deshabilitará el botón 'Calificar proyecto'.

- Seleccionar los archivos relacionados con el proyecto que el usuario con rol 'profesor' crea conveniente que sean publicados junto al proyecto.
  - Calificar el proyecto y publicarlo en caso de que la nota sea igual o superior al 5 haciendo clic en el botón 'Calificar proyecto'.
  - Cerrar el modal sin calificar el proyecto clicando en la cruz que aparece en la parte superior derecha.
  - En caso de que la nota del proyecto sea igual o mayor a 5, el proyecto se publicará en la página 'proyectos publicados' y se generará un .zip con los archivos que el usuario con rol 'profesor' ha seleccionado, este se asignará al proyecto publicado en cuestión.
- En el **perfil del usuario con rol 'admin'** este puede:
    - \*Solo puede acceder el usuario autenticado con rol 'admin' dueño de la cuenta.*
    - Ver todos los proyectos que han sido ocultados de 'proyectos finalizados con éxito'.
    - Volver a mostrar el proyecto que desee clicando en el botón que se encuentra en la parte superior derecha de cada proyecto.
    - Filtrar los proyectos ocultados.
    - Hacer clic en 'Proyecto de proyectos' y acceder a los proyectos públicos.
    - Hacer clic en su nombre y clicar en 'Mi perfil' que lo llevaría a su perfil de nuevo o en 'Cerrar sesión' que cerraría la sesión del mismo.
    - Hacer clic en 'Cambiar contraseña'.
    - Hacer clic en 'Cambiar código de acceso'.
  - En el **modal que se abre tras clicar en 'Cambiar código de acceso'** el usuario puede:
    - \*Solo podrá acceder el usuario autenticado con rol 'admin'.*
    - Elegir un rol (profesor o alumno).
    - Cambiar el código de acceso para ese rol clicando en 'Cambiar código de acceso'.
    - Cerrar el modal sin guardar los cambios clicando en la cruz que aparece en la parte superior derecha.

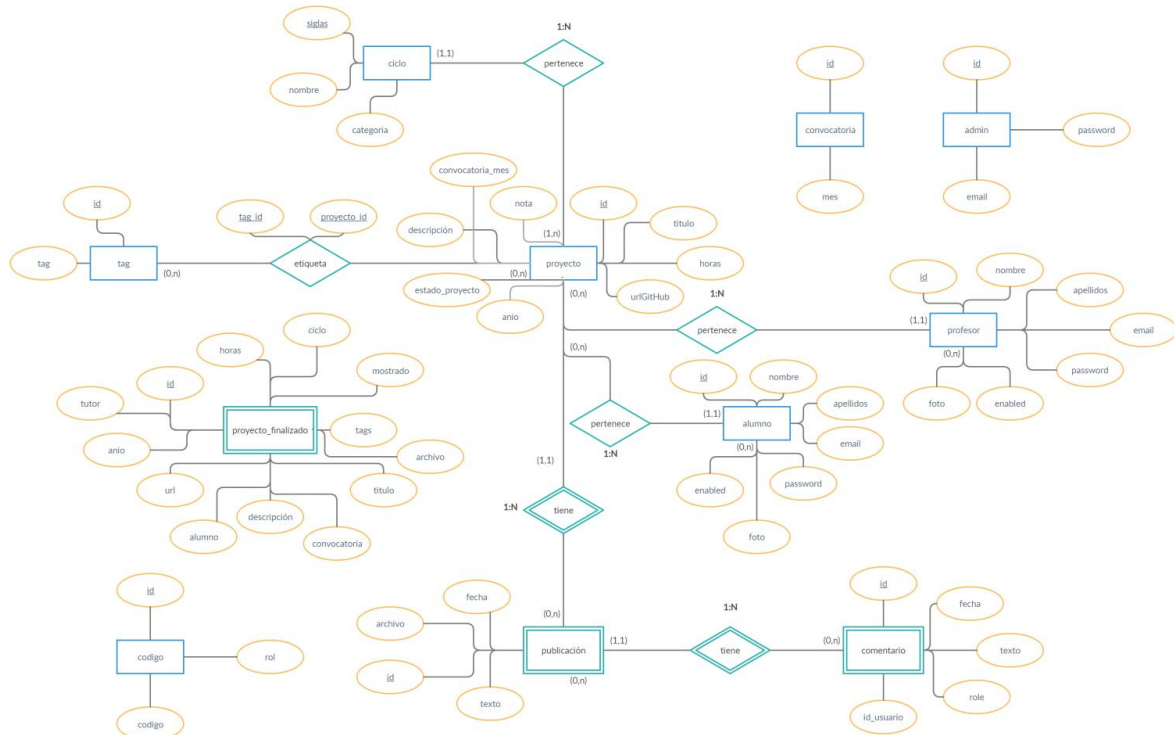
Tras el análisis de los requisitos iniciales expuestos y los casos de uso derivados del mismo, procedo a realizar un modelo entidad-relación de la base de datos:



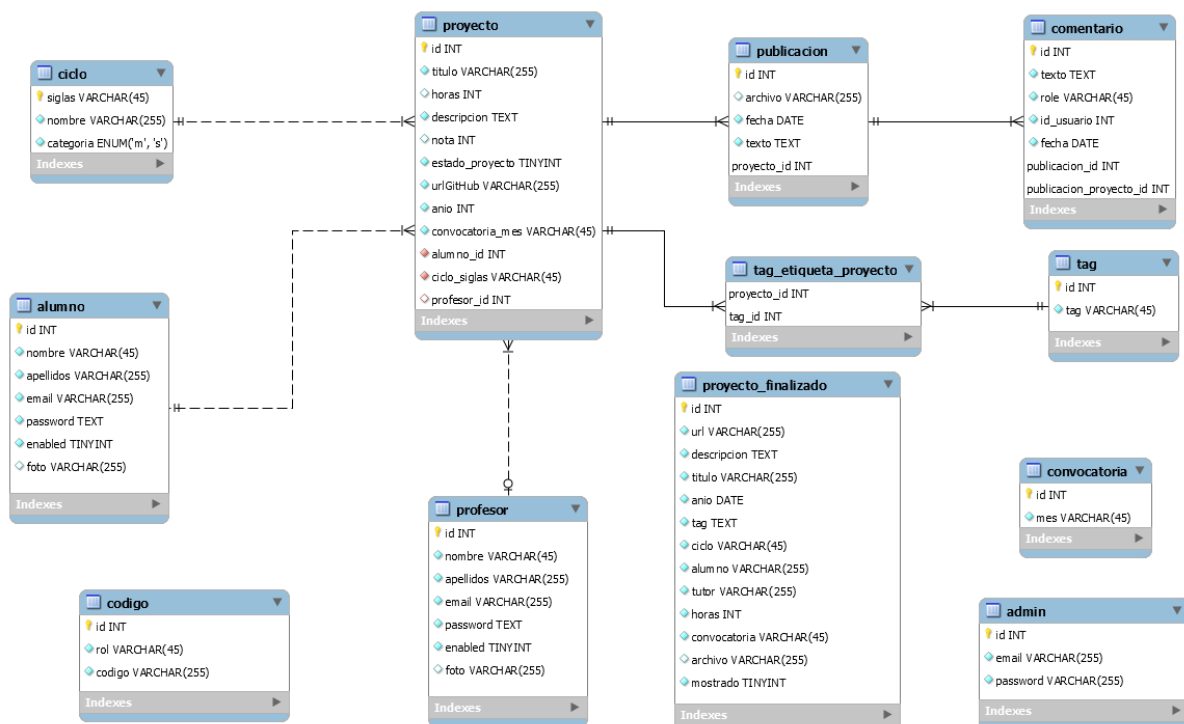
## 5. Diseño de la aplicación

### Diseño de la base de datos:

- Modelo entidad-relación  
(La imagen también se encuentra dentro de la carpeta *anexos*)



- Modelo relacional  
(La imagen también se encuentra dentro de la carpeta *anexos*)



### Diseño de la interfaz:

- **Prototipo realizado con AdobeXD** – Se encuentra en la carpeta ‘anexos’ con el nombre: *‘prototipo\_Hurtado\_Cardoso\_Lucia.xd’*.
- **Guía de estilo de la aplicación** – Se encuentra en la carpeta ‘anexos’ con el nombre: *‘guia\_de\_estilos\_Hurtado\_Cardoso\_Lucia.pdf’*.
- **Medidas utilizadas para garantizar una correcta accesibilidad y usabilidad** – Se encuentra en la carpeta ‘anexos’ con el nombre: *‘accesibilidad\_usabilidad\_Hurtado\_Cardoso\_Lucia.pdf’*.
- **Medidas utilizadas para mejorar el posicionamiento de la página** – Se encuentra en la carpeta ‘anexos’ con el nombre: *‘seo\_Hurtado\_Cardoso\_Lucia.pdf’*.

### Herramientas y tecnologías utilizadas:

- Gestión de la base de datos
  - Para la base de datos he optado por una base de datos relacional, gestionada por **MySQL**. Para el desarrollo de la misma me he apoyado en **MySQL Workbench**, una herramienta visual de diseño de base de datos que integra desarrollo de software, administración de base de datos, diseño de la misma, gestión y almacenamiento...
- BackEnd (servicio REST)
  - El backEnd de la aplicación ha sido realizado con **Spring5**, ya que es uno de los frameworks más potentes de Java y ofrece infinidad de facilidades. El entorno de desarrollo utilizado ha sido **Eclipse** con el plugin **STL (Spring Tool Suite)** que hace que la programación con Spring en Eclipse sea mucho más fluida.
  - Para la conexión entre la base de datos y el backEnd, he utilizado un módulo de Spring llamado **‘Spring Data’**, cuya misión se basa en proporcionar un modelo de programación familiar y consistente y facilita el uso de tecnologías de acceso a datos.
  - Spring Data entre otras cosas, nos proporciona un potente repositorio para el mapeo de objetos, **‘JpaRepository’**, el cual ha sido de gran utilidad a la hora de realizar operaciones con la base de datos.
  - Para la salida de los datos al frontEnt, he utilizado un módulo de Spring llamado **‘Spring Rest’**, que facilita la devolución de dichos datos al frontEnd en forma de JSON.

- También, he implementado '**Spring Security**' con **JWT**, para la gestión del acceso a los recursos, otorgándole a los usuarios un rol específico.

JWT funciona mediante un **token** compuesto por tres partes: cabecera del token, cuerpo del token (donde se encuentra la información referente al usuario) y claves (consta de una clave privada y una clave pública). Dicho token se genera y se firma en el momento en el que el usuario realiza Login de forma correcta y se almacena en el sessionStorage. Cada vez que el frontEnd quiera realizar cualquier petición al backEnd, deberá adjuntar en una cabecera del tipo "Authorization: Bearer *token*" el token. Si el backEnd no reconoce el token o no lo encuentra, la petición no se efectuará.

- Para el envío de correos he usado una librería de Java llamada '**Spring Boot Starter Mail**'.
- ✓ Estructuración del código - El código está dividido en:
  - Entidades (entity): Son aquellas que van acompañadas de la anotación @Entity y hacen referencia a una tabla en la base de datos. En ellas se encuentran los atributos dicha clase y los métodos Get y Set.
  - Dao: Son las interfaces encargadas de gestionar las consultas a la base de datos. En mi aplicación, estás interfaces extienden de JpaRepository.
  - Servicios: Contienen una interfaz en la que se encuentra los métodos que cualquier implementación de la misma ha de cumplir y una clase que implementa dicha interfaz, la cual va acompañada de la anotación @Service.
  - Controladores (controllers): Son los encargados de configurar la ruta (RequestMapping) a la que llega una petición, la envía al servicio, con la respuesta de este genera un JSON y lo devuelve al frontEnd.



- ✓ Todo esto, conlleva a que el backEnd de la aplicación sea, en definitiva, una **API REST**.



- ✓ Los recursos son almacenados en el backEnd divididos en carpetas propias de los distintos proyectos, estas se crean en el momento que se sube el primer archivo referente a un proyecto.

Fuera de estas carpetas, tenemos las fotos de perfil de los usuarios, si el usuario actualiza su foto de perfil, se borrará la anterior, para evitar exceso de archivos.

Además, también podemos encontrar la carpeta 'Final' en la cual se almacenan los .zip que se generan al publicar un proyecto.

- FrontEnd

- El frontEnd de la aplicación ha sido realizado con **Angular**, ya que es uno de los frameworks más potentes en la actualidad para el desarrollo de aplicaciones frontEnd. El entorno de desarrollo ha sido **Visual Studio Code**.

Angular trabaja con **TypeScript**, lo que convierte a **JavaScript** en un lenguaje orientado a objeto.

El código está organizado en componentes, esto ayuda a optimizar el mismo. Cada componente tiene un "html" (vista), un "css" (estilos) y un ".component.ts" (lógica), además, podemos añadirle un ".service.ts" (conexión con el backEnd) y una clase (contiene los atributos).

- He trabajado con **Bootstrap**, para conseguir que mi frontEnd sea responsivo.
- También he usado **Font Awesome**, esta herramienta trata los iconos como texto y no como imágenes, lo que ayuda al tiempo de carga de la página.

Adjunto enlace que explica cómo usar Font Awesome con Angular:

<https://www.npmjs.com/package/@fortawesome/angular-fontawesome>.

- Por último, para los select he usado **bootstrap-select**, ya que ofrece muchas opciones fácilmente aplicables (buscar, marcar/desmarcar todo...) y es muy bonito estéticamente.

<https://developer.snapappointments.com/bootstrap-select/>

## 6. Manuales

Estos documentos se encuentran en la carpeta *anexos*:

- El manual de configuración para poner en funcionamiento la aplicación se encuentra con el nombre *'manual\_configuracion\_Hurtado\_Cardoso\_Lucia.pdf'*.
- El manual de uso de la aplicación se encuentra con el nombre *'manual\_uso\_Hurtado\_Cardoso\_Lucia.pdf'*.

## 7. Gestión del proyecto

Durante la realización del proyecto, he llevado a cabo el backEnd y el frontEnd de forma simultánea.

En los inicios de la aplicación, gran parte del tiempo fue destinado a la configuración de la base de datos. Mantuve varias reuniones con mi tutora de proyecto para crear una base de datos adecuada y fui optimizándola para que hiciera más fácil algunas funcionalidades de la aplicación, por ello, aproximadamente la puesta a punto de la base de datos conllevó unas 10/15 horas.

La realización del backEnd ha sido bastante laboriosa, ya que como he explicado anteriormente, no conocía la tecnología usada (Spring5). Aproximadamente, esta tarea ha podido conllevar unas 150/160 horas.

Por último, la realización del frontEnd y la maquetación del mismo, ha conllevado aproximadamente unas 100 horas.

## 8. Conclusiones

La elaboración de este proyecto, ha sido un gran reto. Gracias a él, me he superado a mí misma y he aprendido muchísimo de tecnologías que están siendo bastante utilizadas en la actualidad.

La idea inicial ha sido superada con creces por el resultado final del proyecto. Cada idea que me planteaba mi tutora de proyecto, ha podido ser realizada de forma 'sencilla' gracias a la gran organización del código y a lo desacoplado que se encuentra el mismo.

Arreglar algunos de los errores que han surgido en el camino, me ha llevado horas, pero cuando dicho problema se solucionaba, la satisfacción valía la pena.

## 9. Bibliografía

Para la realización del proyecto, he tenido que consultar muchísimas dudas en San Google, algunas de las páginas que más he frecuentado han sido:

- <https://angular.io/docs>
- <https://spring.io/>
- <https://getbootstrap.com/docs/4.5/getting-started/introduction/>
- <https://stackoverflow.com/>
- Algunas dudas las he resuelto con los cursos de Udemey realizados para la 'fct especial'. Los cursos son los siguientes:
  - Angular & Spring 5: Creando web app full stack (Angular 9+)
  - Diseño Web Responsive con Foundation for Sites SASS y XYGrid
- <https://jwt.io/>
- <https://color.adobe.com/es/create/color-wheel>
- <https://www.eclipse.org/downloads/>
- <https://code.visualstudio.com/download>
- <https://github.com/>
- <https://www.baeldung.com/spring-boot>
- <https://jquery.com/>
- <https://developer.snapappointments.com/bootstrap-select/>
- <https://fontawesome.com/>