



GRADO EN INGENIERÍA EN TECNOLOGÍAS DE  
TELECOMUNICACIÓN

TRABAJO FIN DE GRADO- MEMORIA

**Aplicación Web para la Gestión de Procesos de  
Ingesta de Datos en Entornos de BI-Big Data**

Autor: Constantino Fernández Pérez

Director: Alberto Ayora Pais

Codirector: Atilano Fernández-Pacheco Sánchez-Migallón

*Madrid*

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título **Aplicación Web para la Gestión de Procesos de Ingesta de Datos en Entornos de BI-Big Data** en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el curso académico 2019-2020 es de mi autoría, original e inédito y no ha sido presentado con anterioridad a otros efectos.

El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido tomada de otros documentos está debidamente referenciada.

Fdo.: Constantino Fernández Pérez

Fecha: 09/ 07/ 2020



Autorizada la entrega del proyecto

EL DIRECTOR DEL PROYECTO



Fdo.: Atilano Fernández-Pacheco Sánchez Migallón

Fecha: 09/ 07/ 2020

## **AUTORIZACIÓN PARA LA DIGITALIZACIÓN, DEPÓSITO Y DIVULGACIÓN EN RED DE PROYECTOS FIN DE GRADO, FIN DE MÁSTER, TESIS O MEMORIAS DE BACHILLERATO**

### **1º. Declaración de la autoría y acreditación de la misma.**

El autor D. *Constantino Fernández Pérez* DECLARA ser el titular de los derechos de propiedad intelectual de la obra: *Aplicación Web para la Gestión de Procesos de Ingesta de Datos en Entornos de BI-Big Data*, que ésta es una obra original, y que ostenta la condición de autor en el sentido que otorga la Ley de Propiedad Intelectual.

### **2º. Objeto y fines de la cesión.**

Con el fin de dar la máxima difusión a la obra citada a través del Repositorio institucional de la Universidad, el autor **CEDE** a la Universidad Pontificia Comillas, de forma gratuita y no exclusiva, por el máximo plazo legal y con ámbito universal, los derechos de digitalización, de archivo, de reproducción, de distribución y de comunicación pública, incluido el derecho de puesta a disposición electrónica, tal y como se describen en la Ley de Propiedad Intelectual. El derecho de transformación se cede a los únicos efectos de lo dispuesto en la letra a) del apartado siguiente.

### **3º. Condiciones de la cesión y acceso**

Sin perjuicio de la titularidad de la obra, que sigue correspondiendo a su autor, la cesión de derechos contemplada en esta licencia habilita para:

- a) Transformarla con el fin de adaptarla a cualquier tecnología que permita incorporarla a internet y hacerla accesible; incorporar metadatos para realizar el registro de la obra e incorporar “marcas de agua” o cualquier otro sistema de seguridad o de protección.
- b) Reproducir la en un soporte digital para su incorporación a una base de datos electrónica, incluyendo el derecho de reproducir y almacenar la obra en servidores, a los efectos de garantizar su seguridad, conservación y preservar el formato.
- c) Comunicarla, por defecto, a través de un archivo institucional abierto, accesible de modo libre y gratuito a través de internet.
- d) Cualquier otra forma de acceso (restringido, embargado, cerrado) deberá solicitarse expresamente y obedecer a causas justificadas.
- e) Asignar por defecto a estos trabajos una licencia Creative Commons.
- f) Asignar por defecto a estos trabajos un HANDLE (URL *persistente*).

### **4º. Derechos del autor.**

El autor, en tanto que titular de una obra tiene derecho a:

- a) Que la Universidad identifique claramente su nombre como autor de la misma
- b) Comunicar y dar publicidad a la obra en la versión que ceda y en otras posteriores a través de cualquier medio.
- c) Solicitar la retirada de la obra del repositorio por causa justificada.
- d) Recibir notificación fehaciente de cualquier reclamación que puedan formular terceras personas en relación con la obra y, en particular, de reclamaciones relativas a los derechos de propiedad intelectual sobre ella.

### **5º. Deberes del autor.**

El autor se compromete a:

- a) Garantizar que el compromiso que adquiere mediante el presente escrito no infringe ningún derecho de terceros, ya sean de propiedad industrial, intelectual o cualquier otro.
- b) Garantizar que el contenido de las obras no atenta contra los derechos al honor, a la intimidad y a la imagen de terceros.
- c) Asumir toda reclamación o responsabilidad, incluyendo las indemnizaciones por daños, que pudieran ejercitarse contra la Universidad por terceros que vieran infringidos sus derechos e intereses a causa de la cesión.
- d) Asumir la responsabilidad en el caso de que las instituciones fueran condenadas por infracción

de derechos derivada de las obras objeto de la cesión.

**6º. Fines y funcionamiento del Repositorio Institucional.**

La obra se pondrá a disposición de los usuarios para que hagan de ella un uso justo y respetuoso con los derechos del autor, según lo permitido por la legislación aplicable, y con fines de estudio, investigación, o cualquier otro fin lícito. Con dicha finalidad, la Universidad asume los siguientes deberes y se reserva las siguientes facultades:

- La Universidad informará a los usuarios del archivo sobre los usos permitidos, y no garantiza ni asume responsabilidad alguna por otras formas en que los usuarios hagan un uso posterior de las obras no conforme con la legislación vigente. El uso posterior, más allá de la copia privada, requerirá que se cite la fuente y se reconozca la autoría, que no se obtenga beneficio comercial, y que no se realicen obras derivadas.
- La Universidad no revisará el contenido de las obras, que en todo caso permanecerá bajo la responsabilidad exclusiva del autor y no estará obligada a ejercitar acciones legales en nombre del autor en el supuesto de infracciones a derechos de propiedad intelectual derivados del depósito y archivo de las obras. El autor renuncia a cualquier reclamación frente a la Universidad por las formas no ajustadas a la legislación vigente en que los usuarios hagan uso de las obras.
- La Universidad adoptará las medidas necesarias para la preservación de la obra en un futuro.
- La Universidad se reserva la facultad de retirar la obra, previa notificación al autor, en supuestos suficientemente justificados, o en caso de reclamaciones de terceros.

Madrid, a 12 de Mayo de 2020

**ACEPTA**

Fdo.....

Motivos para solicitar el acceso restringido, cerrado o embargado del trabajo en el Repositorio Institucional:



GRADO EN INGENIERÍA EN TECNOLOGÍAS DE  
TELECOMUNICACIÓN

TRABAJO FIN DE GRADO- MEMORIA

**Aplicación Web para la Gestión de Procesos de  
Ingesta de Datos en Entornos de BI-Big Data**

Autor: Constantino Fernández Pérez

Director: Alberto Ayora Pais

Codirector: Atilano Fernández-Pacheco Sánchez-Migallón

*Madrid*



# Agradecimientos

A mis padres y familia. Por la oportunidad, por el apoyo incondicional desde el día que puse mi primer pie en Madrid. No tengo palabras para agradeceros todo lo que habéis hecho por mí. No habría sido posible sin vosotros.

A la universidad. Institución que más me ha definido y moldeado, y sin duda la mejor elección de mi vida. Estaré siempre agradecido por los valores y conocimientos adquiridos y orgulloso de formar parte de ella.

A mis amigos. Los de siempre por seguir ahí, y los que han llegado nuevos por hacer de estos 4 años que Dios sabe que han sido duros, los mejores de mi vida.

A mis profesores, todos y cada uno de ellos. Mención especial a Jose Luis Rodriguez Marrero. Creo que nunca he tenido un profesor que me lo haya hecho pasar tan mal y tan bien a la vez. Me hiciste disfrutar de cada una de las clases que me diste, me hiciste trabajar duro para mejorar, me hiciste crecer como estudiante y como persona.

A Atilano, por su respeto, por su cercanía, por ser un profesor y un “amigo”, por las mejores clases de la universidad. El profesor con diferencia que más me ha ayudado, y del que más he aprendido.

A Altia y a Alberto, por su constancia, por su dedicación, por su tiempo.

Gracias.





# **Aplicación Web para la Gestión de Procesos de Ingesta de Datos en Entornos de BI-Big Data**

**Autor: Fernández Pérez, Constantino.**

Director: Ayora País, Alberto

Codirector: Fernández-Pacheco Sánchez-Migallón, Atilano

Entidad Colaboradora: Altia Consultores.

## **RESUMEN DEL PROYECTO**

El presente proyecto denominado Aplicación Web para la Gestión de Procesos de Ingesta de Datos en Entornos de Big Data, consiste en una plataforma web que implementa un sistema de gestión (subida, ejecución y revisión) de ficheros .jar, que al ejecutarse llevan a cabo el proceso de ingesta de datos en ellos definida.

**Palabras clave:** Big Data, Ficheros Ejecutables, Gestión de ficheros, Gestión de proyectos, Gestión de Usuarios.

### **1. Introducción**

El concepto de Big Data aplica para toda la información que, por su gran tamaño, no puede ser procesada o analizada utilizando procesos o herramientas tradicionales. El creciente avance de las tecnologías proporciona una también creciente cantidad de información, información que las tecnologías Big Data pueden aprovechar para obtener respuestas que ayuden a tomar decisiones de negocio.

El primer paso en cualquier arquitectura Big Data es la Ingesta de datos, que consiste en introducir datos de diferentes fuentes, estructura o características dentro de otros sistemas de almacenamiento o procesamiento de datos.

Lo que la aplicación desarrollada pretende es gestionar distintos procesos de ingesta de datos mediante la gestión de ficheros que, al ejecutarse, lleven a cabo alguno de estos procesos.

### **2. Definición del Proyecto**

El proyecto desarrollado por lo tanto es una aplicación web que implementa un sistema de gestión (subida, ejecución y revisión) de ficheros ejecutables.

La gestión de ficheros implica una gestión de los proyectos y directorios a los que estos pertenecen, así como de los usuarios que tendrán acceso a ellos, sus perfiles, entornos, y permisos.

El conjunto de tareas individuales se complementarán para cumplir con el objetivo final de la aplicación: la gestión de procesos de ingesta de datos.

### **3. Descripción del modelo/sistema/herramienta**

La aplicación desarrollada cuenta con dos tipos de perfiles de usuario: administradores y desarrolladores, y tres entornos distintos: desarrollo, preproducción y producción.

Los entornos a los que cada usuario tenga acceso definirán sus perfiles asociados, y con ello sus permisos a la hora de gestionar los distintos proyectos. Existen tres permisos distintos: despliegue, ejecución, y programación, y cada uno está asociado a una o más acciones dentro de la aplicación:

- El permiso de despliegue permite crear y modificar la estructura de directorios dentro de los proyectos, crear o subir ficheros de texto, importar ficheros ejecutables, editarlos, y eliminarlos.
- El permiso de ejecución permite ejecutar al momento los ficheros ejecutables, así como ver la información relativa a ejecuciones anteriores.
- El permiso de programación permite la programación de ejecuciones de forma puntual o periódica. Incluye también la revisión de la información relativa a ejecuciones pasadas (igual que el permiso de ejecución).

Cada usuario por lo tanto tendrá uno o más perfiles asociados, y cada perfil está a su vez asociado a un entorno. El entorno de desarrollo brinda permisos completos a sus usuarios, y el de preproducción y producción únicamente tienen asociado el permiso de programación.

La creación, modificación y eliminación de los usuarios y proyectos se realizará por parte de los usuarios administradores, que tienen acceso a todos los entornos y por lo tanto permisos completos en la aplicación.

#### 4. Resultados

La aplicación desarrollada cumple con todos los objetivos descritos, gestionando distintos usuarios y proyectos, y permitiendo la subida, ejecución y revisión de ficheros tanto de texto como ejecutables.

La siguiente captura muestra una captura del menú de un usuario administrador (con permisos completos en la aplicación):



*Ilustración 1. Menú usuario administrador*

A continuación se muestra un ejemplo de la estructura de directorios creada dentro de un proyecto de prueba. La información sobre los directorios y los ficheros se guardará tanto en base de datos como en el servidor (configurable en fichero .properties).



*Ilustración 2. Árbol de directorios de proyecto de prueba.*

Las siguientes capturas de pantalla muestran las ventanas que permiten a usuarios con permisos de ejecución y programación ejecutar cualquier fichero ejecutable (fichero que llevará a cabo un proceso de ingesta de datos):



Ilustración 3. Ventana para ejecutar un fichero



Ilustración 4. Ventana para programar la ejecución de un fichero.

Por último muestro una captura de la tabla de ejecuciones de ficheros .jar del proyecto de prueba, en la que se muestra toda la información relativa a una ejecución junto con un botón que permite descargar el fichero log con la traza de ejecución completa:

Fichero	Fecha	Hora inicio	Hora Fin	Resultado	Usuario	Parámetros	Log
fichero2.jar	2020-07-07	20:50:00.004	20:50:00.150	SUCCESS	TU	-Xms1g -Xmx2048m	
fichero2.jar	2020-07-07	20:45:08.580	20:45:10.045	SUCCESS	TU	-Xms1g -Xmx2048m	

Entradas de 1 a 1 de 2 de 2 totales

Ilustración 5. Tabla de ejecuciones pasadas en proyecto de prueba.

## 5. Conclusiones

La aplicación desarrollada proporciona un base robusta para una aplicación que podría facilitar enormemente esa primera fase en cualquier arquitectura Big Data. Funciona independientemente del proceso elegido ya que en sí no implementa ninguna arquitectura de Big Data, y permite la programación de la ejecución de los ficheros, permitiendo a su vez la programación de los procesos de ingesta de datos para que se realicen de forma periódica sin necesidad de realizarlos de forma manual.

Aunque todavía se encuentra en su primera versión, supone una base sólida para lo que podría ser una aplicación de uso extendido en entornos de Business Intelligence.

## 6. Referencias

[1] ¿Por qué el Big Data actualmente es tan importante? Universidad de Alcalá.

<https://www.master-bigdata.com/big-data-actualmente-tan-importante/>

# **Web Application for the Management of Data Intake in Big Data Environments**

**Author: Fernández Pérez, Constantino.**

Supervisor 1: Ayora País, Alberto

Supervisor 2: Fernández-Pacheco Sánchez-Migallón, Atilano

Collaborating Entity: Altia Consultores.

## **ABSTRACT**

This Project called Web Application for the Management of Data Intake in big Data Environments, consists of a web platform that implements a management system (upload, execution and review) of .jar files, which when executed carry out the process of data ingestion defined in them.

**Key words:** Big Data, Executable files, Files management, Projects management, User management.

## **1. Introduction**

The Big Data concept applies to all information that, due to its large size, cannot be processed or analyzed using traditional processes or tools. The increasing advancement of technologies provides an increasing amount of information, information that Big Data technologies can take advantage of to obtain answers that help to make business decisions.

The first step in any Big Data architecture is Data Ingestion, which consists of introducing data from different sources, structure or characteristics into other data storage or processing systems.

What the developed application intends is to manage different data ingestion processes by managing files that, when executed, carry out any of these processes.

## **2. Project definition**

The project developed is therefore a web application that implements a management system (upload, execution and review) of executable files.

File management involves managing the projects and directories to which they belong, as well as the users who will have access to them, their profiles, environments, and permissions.

The set of individual tasks will be complemented to fulfill the final objective of the application: the management of data ingestion processes.

## **3. Descripción del modelo/sistema/herramienta**

The developed application has two types of user profiles: administrators and developers, and three different environments: development, pre-production and production.

The environments to which each user has access will define their associated profiles, and with it their permissions when managing different projects. There are three different permissions: deployment, execution, and schedule, and each one is associated with one or more actions within the application:

- The deployment permission allows the creation and modification of the directory structure within projects, creating or uploading text files, importing executable files, editing them, and deleting them.
- The execution permission allows to execute the executable files at the moment, as well as to see the information related to previous executions.
- The scheduling permission allows the scheduling of executions on a regular or periodic basis. It also includes the review of information regarding past executions (same as the execution permission).

Each user will therefore have one or more associated profiles, and each profile is in turn associated with an environment. The development environment provides full permissions to its users, and the pre-production and production environment only have the programming permission associated with it.

The creation, modification and deletion of users and projects will be carried out by administrator users, who have access to all environments and therefore full permissions in the application.

#### 4. Resultados

The developed application fulfills all the described objectives, managing different users and projects, and allowing the upload, execution and revision of both text and executable files.

The following screenshot shows a screenshot of an administrator user menu (with full permissions in the app):



*Illustration 1. Administrator user menu*

Below is an example of the directory structure created within a test project. The information about the directories and files will be saved both in the database and on the server (configurable in .properties file).



*Illustration 6. Test project directory tree.*

The following screenshots show the windows that allow users with execution and programming permissions to execute any executable file (a file that will carry out a data ingestion process):



Illustration 3. Window to execute a file



Illustration 4. Window to program the execution of a file.

Finally, I show a capture of the execution table of .jar files of the test project, in which all the information related to an execution is shown together with a button that allows downloading the log file with the complete execution trace:

Ejecuciones pasadas:							
Fecha	Hora inicio	Hora fin	Resultado	Usuario	Parámetros	Log	
2020-07-07	20:50:00.094	20:50:00.159	SUCCESS	TU	-Xms1g -Xmx2048m		
2020-07-07	20:45:09.590	20:45:10.045	SUCCESS	TU	-Xms1g -Xmx1024k		

Entradas de 1 a 2 de 2 totales

Anterior 1 Siguiente

Illustration 5. Table of past executions in test project.

## 5. Conclusions

The developed application provides a robust foundation for an application that could greatly facilitate that first phase in any Big Data architecture. It works independently of the chosen process since in itself it does not implement any Big Data architecture, and allows the scheduling of the execution of the files, in turn allowing the scheduling of the data intake processes to be carried out periodically without need to do them manually.

Although it is still in its first version, it provides a solid foundation for what could be a widely used application in Business Intelligence environments.

## 6. References

[1] ¿Por qué el Big Data actualmente es tan importante? Universidad de Alcalá.

<https://www.master-bigdata.com/big-data-actualmente-tan-importante/>



## *Índice de la memoria*

<b>Capítulo 1. Introducción.....</b>	<b>22</b>
1.1 Motivación del proyecto .....	22
1.2 Estructura del documento .....	24
<b>Capítulo 2. Descripción de las Tecnologías.....</b>	<b>26</b>
2.1 Java .....	26
2.2 Eclipse.....	28
2.3 MySQL y MySQL Workbench.....	29
2.4 Maven .....	30
2.5 JavaScript.....	31
<b>Capítulo 3. Estado de la Cuestión .....</b>	<b>32</b>
<b>Capítulo 4. Definición del Trabajo.....</b>	<b>35</b>
4.1 Justificación .....	35
4.2 Objetivos.....	35
4.3 Metodología.....	37
<b>Capítulo 5. Sistema/Modelo Desarrollado .....</b>	<b>40</b>
5.1 Arquitectura y Patrón de diseño.....	40
5.2 Requisitos de la aplicación .....	44
5.2.1 Requisitos de usuario .....	44
5.2.2 Requisitos de sistema.....	52
5.3 Descripción del sistema .....	66
5.3.1 Funcionamiento de la aplicación .....	66
5.3.2 Diagrama de casos de uso.....	70
5.3.3 Descripción de casos de uso.....	72
5.3.4 Diagrama de clases .....	96
5.3.5 Descripción de clases .....	97
5.4 Modelo de datos.....	101



---

5.4.1 Modelo Físico de datos .....	101
5.4.2 Descripción de entidades.....	102
<b>Capítulo 6. Análisis de Resultados.....</b>	<b>109</b>
<b>Capítulo 7. Conclusiones y Trabajos Futuros.....</b>	<b>111</b>
<b>Capítulo 8. Bibliografía .....</b>	<b>114</b>
<b>Anexo I: Manual de instalación.....</b>	<b>115</b>
<b>Anexo II: Manual de la aplicación.....</b>	<b>124</b>
<b>Anexo III: Alineación proyecto con los ODS .....</b>	<b>146</b>

## Índice de figuras

Figura 1. Logo Java .....	26
Figura 2. Logo eclipse. ....	28
Figura 3. Logo MySQL .....	29
Figura 4. Logo Maven .....	30
Figura 5. Ejemplo estructura de directorios. ....	37
Figura 6. Diagrama de Gantt proyecto .....	39
Figura 7. Arquitectura de 3 capas .....	41
Figura 8. Patrón MVC .....	42
Figura 9. Vistazo Árbol directorios proyecto .....	67
Figura 10. Menú desplegable directorio raiz .....	67
Figura 11. Menú desplegable directorio .....	67
Figura 12. Menú desplegable fichero de texto .....	68
Figura 13. Menú desplegable fichero ejecutable .....	68
Figura 14. Digrama casos de uso WebApp[1].....	70
Figura 15. Digrama casos de uso WebApp[2].....	71
Figura 16. Diagrama de clases WebApp .....	96
Figura 17. Modelo de datos WebApp.....	101
Figura 18. Instalación Java. Manual Instalación. ....	115
Figura 19. Instalador MySQL. Manual de instalación[1].....	118
Figura 20. Página Login WebApp.....	124
Figura 21. Mensaje Error autenticación[3].....	125
Figura 22. Mensaje Error autenticación[2].....	125
Figura 23. Mensaje Error autenticación[1].....	125
Figura 24. Página Inicio Administrador WebApp.....	126
Figura 25. Página Inicio Desarrollador WebApp .....	127
Figura 26. Página Gestión de Usuarios WebApp .....	128
Figura 27. Ventana Nuevo Usuario WebApp[1] .....	129
Figura 29. Ventana Nuevo Usuario WebApp[2] .....	130

Figura 28 . Mensaje confirmación Nuevo Usuario WebApp .....	130
Figura 30. Usuario creado WebApp .....	131
Figura 31. Ventana Dar de Baja Usuario WebApp .....	132
Figura 32. Ventana Editar Usuario WebApp.....	132
Figura 33. Página Gestión de Proyectos WebApp .....	133
Figura 34. Ventana Editar Proyecto WebApp .....	134
Figura 35. Ventana Nuevo Proyecto WebApp .....	134
Figura 36. Ventana Confirmar Eliminación de Proyecto .....	135
Figura 37. Ventana Editar Perfil WebApp .....	136
Figura 38. Ventana Proyecto de Prueba[1].....	137
Figura 39. Ventana Proyecto de Prueba[3].....	138
Figura 40. Ventana Proyecto de Prueba[2].....	138
Figura 41. Ventana nuevo Directorio WebApp.....	139
Figura 42. Cambiar nombre directorio/fichero WebApp .....	139
Figura 43. Ventana Proyecto de Prueba[4].....	140
Figura 44. Ventana nuevo fichero de texto WebApp .....	140
Figura 45. Nuevo fichero de texto WebApp[2].....	141
Figura 46. Importar fichero menú desplegable WebApp .....	141
Figura 47. Ventana Proyecto de Prueba WebApp[5] .....	142
Figura 48. Ventana selección de fichero importado WebApp.....	142
Figura 49. Ventana Ejecutar fichero WebApp .....	143
Figura 50. Ficheros ejecutados WebApp.....	144
Figura 51. Ventana programar ejecución fichero WebApp.....	144
Figura 52. Confirmación resetear tabla ejecuciones pasadas WebApp.....	145
Figura 53. Cerrar Sesión WebApp .....	145

## *Índice de tablas*

Tabla 1. Requisitos de usuario.....	51
Tabla 2. Requisitos de sistema: funcionales.....	64
Tabla 3. Requisitos de sistema: no funcionales.....	65
Tabla 4. Descripción de caso de uso: Autenticación.....	72
Tabla 5. Descripción de caso de uso: editar perfil propio.....	73
Tabla 6. Descripción de caso de uso: ver listado de proyectos propios.....	74
Tabla 7. Descripción de caso de uso: abrir proyecto.....	75
Tabla 8. Descripción de caso de uso: crear directorio.....	76
Tabla 9. Descripción de caso de uso: eliminar directorio.....	77
Tabla 10. Descripción de caso de uso: modificar nombre directorio.....	78
Tabla 11. Descripción de caso de uso: crear fichero de texto.....	79
Tabla 12. Descripción de caso de uso: abrir fichero de texto.....	79
Tabla 13. Descripción de caso de uso: modificar fichero de texto.....	80
Tabla 14. Descripción de caso de uso: importar fichero.....	81
Tabla 15. Descripción de caso de uso: eliminar fichero.....	82
Tabla 16. Descripción de caso de uso: modificar nombre fichero.....	83
Tabla 17. Descripción de caso de uso: ejecutar fichero.....	84
Tabla 18. Descripción de caso de uso: programar ejecución fichero.....	86
Tabla 19. Descripción de caso de uso: descargar fichero log.....	86
Tabla 20. Descripción de caso de uso: ver ejecuciones anteriores.....	87
Tabla 21. Descripción de caso de uso: cerrar sesión.....	87
Tabla 22. Descripción de caso de uso: ver lista de usuarios.....	88
Tabla 23. Descripción de caso de uso: crear nuevo usuario.....	89
Tabla 24. Descripción de caso de uso: editar usuario.....	90
Tabla 25. Descripción de caso de uso: dar de baja usuario.....	91
Tabla 26. Descripción de caso de uso: ver lista completa de proyectos.....	92
Tabla 27. Descripción de caso de uso: crear nuevo proyecto.....	93
Tabla 28. Descripción de caso de uso: editar proyecto.....	94

---

Tabla 29. Descripción de caso de uso: eliminar proyecto .....	95
Tabla 30. Descripción de clase: User .....	97
Tabla 31. Descripción de clase: Permiso .....	97
Tabla 32. Descripción de clase: Entorno .....	98
Tabla 33. Descripción de clase: Perfil .....	98
Tabla 34. Descripción de clase: Proyecto.....	98
Tabla 35. Descripción de clase: Directorio .....	99
Tabla 36. Descripción de clase: Fichero.....	99
Tabla 37. Descripción de clase: Ejecución .....	100
Tabla 38. Notación tablas entidad. ....	102
Tabla 39. Descripción de entidades: user .....	102
Tabla 40. Descripción de entidades: userprofile .....	103
Tabla 41. Descripción de entidades: userproject.....	103
Tabla 42. Descripción de entidades: profile .....	104
Tabla 43. Descripción de entidades: profilepermit.....	104
Tabla 44. Descripción de entidades: permit .....	105
Tabla 45. Descripción de entidades: project.....	105
Tabla 46. Descripción de entidades: environment.....	106
Tabla 47. Descripción de entidades: directorio .....	106
Tabla 48. Descripción de entidades: fichero .....	107
Tabla 49. Descripción de entidades: ejecución .....	108

## **Capítulo 1. INTRODUCCIÓN**

### ***1.1 MOTIVACIÓN DEL PROYECTO***

Se estima que al finalizar 2020 habremos producido 44 veces más datos de los que producimos en el 2010. El creciente avance de la tecnología proporciona una también creciente cantidad de información. Esta información puede ser almacenada y analizada, proporcionando respuestas a preguntas que ni sabíamos que teníamos.

Las empresas cada vez tienen más arraigada la cultura del dato: obtener información de todo lo posible, y almacenarla en grandes bases de datos. La tecnología Big Data sin embargo no reside únicamente en la posibilidad de almacenar esta información, sino que reside en la capacidad para limpiar y tratar estos millones de datos, y aquí es donde entran en juego los procesos ETL (Extract, Transform, Load). Las herramientas ETL automatizan las operaciones de extracción de los datos de los sistemas de origen, su transformación para usos de procesamiento y analítica, y posterior carga en destino.

La Ingesta de datos es el primer paso y una de las piezas fundamentales en una arquitectura Big Data. Se trata del proceso mediante el cual se introducen datos, de diferentes fuentes, estructura o características dentro de otros sistemas de almacenamiento o procesamiento de datos.

Existen distintos procesos para llevar a cabo esta mudanza de datos en cualquier arquitectura Big Data, y una forma de poner en marcha estos procesos es mediante el uso de ficheros de parametrización, que, al ejecutarse, realizan el proceso de ingesta de datos que en ellos se ha definido.

El proyecto al que esta memoria hace referencia, denominado *Aplicación Web para la Gestión de Procesos de Ingesta de Datos en Entornos de BI-Big Data*, consiste en una

plataforma Web para la gestión (subida, revisión y ejecución) de estos ficheros de parametrización, y por consiguiente, gestión de distintos procesos de ingesta de datos.

Dicha aplicación podría ser incluida en las propuestas de ofertas tanto en entornos de BI puros como en Big Data, como un componente de gestión para la capa de ingesta de datos, así como ser comercializado de manera independiente.

Esta aplicación debe permitir la subida y revisión de ficheros .txt y .jar y la ejecución de estos últimos, así como gestionar su organización en estructuras de directorios para cada uno de los proyectos. La ejecución de los ficheros debe además mostrar si la ejecución ha sido exitosa y generar un fichero log para su posterior revisión.

La gestión de estos ficheros y los directorios que los contienen requieren a su vez una gestión de los proyectos a los que pertenecen, además de los usuarios que a ellos tienen acceso y sus permisos dentro de cada entorno. Es por eso por lo que la aplicación a desarrollar se divide en 4 partes, cuyas tareas individuales se complementarán para cumplir con los requisitos del proyecto al completo.

La primera parte se encargará de la gestión de diferentes perfiles de usuarios: desarrolladores y administradores, y la segunda gestionará los equipos, los entornos, y los proyectos asignados a cada usuario. Los usuarios de administración tendrán permisos completos en cada uno de los diferentes entornos, y serán los encargados además de dar de alta proyectos y vincular a los desarrolladores a dichos proyectos.

La tercera parte será la encargada de la administración de las planificaciones de ejecución para cada proceso de ingesta de datos, es decir, de los ficheros ejecutables para cada proyecto dentro de la aplicación.

Es importante destacar que la aplicación ejecuta los ficheros independientemente de su contenido, y por lo tanto es independiente del proceso de ingesta de datos que el fichero contenga. La ejecución del fichero será la que realizará esa mudanza de datos, de un origen a un destino que tampoco conciernen a la aplicación desarrollada, ya que esta no implementa ninguna arquitectura Big Data.

De hecho, ni siquiera tendría por qué llevar a cabo un proceso de ingesta de datos, ya que la aplicación de lo único de lo que se encargará será de ejecutar el fichero y guardar la traza de ejecución para su posterior revisión.

Esta última necesidad nos conduce a la cuarta y última parte, que se centrará en la recuperación de los logs de trazabilidad de las ejecuciones realizadas, guardándose en base de datos para su posterior visualización y revisión.

El proyecto realizado proporciona una herramienta de trabajo a los usuarios finales y se desarrolla para cumplir con la necesidad establecida de gestión de procesos de ingesta de datos en un entorno de Big Data.

## ***1.2 ESTRUCTURA DEL DOCUMENTO***

La presente memoria se estructura por capítulos, cuyo contenido se describe a continuación:

**Capítulo 1. Introducción:** este capítulo presenta una explicación genérica del proyecto desarrollado, partiendo de la necesidad de generación de la aplicación y estableciendo unos objetivos generales para aportar una visión global en la que situar al lector.

**Capítulo 2. Descripción de tecnologías:** expone las tecnologías, protocolos, y demás herramientas específicas que se vayan a tratar durante el proyecto para facilitar su lectura y comprensión.

**Capítulo 3. Estado de la cuestión:** en este capítulo revisaré qué trabajos o soluciones existen en el ámbito de mi proyecto. Ante la idea inicial de desarrollar un proyecto, siempre debemos realizarnos la pregunta: ¿hay algo similar en el mercado? ¿Hay algún trabajo de investigación que haya aportado los resultados que quiero alcanzar?

**Capítulo 4. Definición del Trabajo:** A la vista del capítulo anterior debemos realizar un análisis crítico sobre los trabajos previos realizados y saber ya claramente por qué se va a



desarrollar el proyecto. Se pretende responder a la pregunta ¿Por qué alguien puede querer comprar este proyecto?

También se comentarán objetivos concretos del proyecto, y se expondrá el camino dividido en fases tomado para conseguirlos.

Por último, se expondrá una presentación temporal de las actividades a realizar.

**Capítulo 5. Sistema/ Modelo desarrollado:** en este capítulo se realiza una descripción detallada de la solución implementada. Contiene información de análisis y diseño de la solución que proporcionan la información necesaria para la comprensión de su funcionamiento e implementación.

**Capítulo 6. Análisis de resultados:** se destacan los resultados más relevantes del proyecto y se realiza un análisis crítico de los mismos.

**Capítulo 7. Conclusiones y trabajos futuros:** se comentan las conclusiones del proyecto, destacando lo que se ha hecho, los objetivos cubiertos y las aportaciones hechas.

**Capítulo 8. Bibliografía:** referencias.

## Capítulo 2. DESCRIPCIÓN DE LAS TECNOLOGÍAS

### 2.1 JAVA



*Figura 1. Logo Java*

Java es un lenguaje de programación y una plataforma informática rápida, segura y fiable, y uno de los lenguajes de programación más populares en uso, particularmente para aplicaciones de cliente- servidor.

Su sintaxis deriva en gran parte de C y C++, pero tiene menos utilidades de bajo nivel que cualquier de ellos. Las aplicaciones Java son compiladas a bytecode (clases Java), que pueden ejecutarse en cualquier máquina virtual Java sin importar la arquitectura de la computadora subyacente.

Precisamente por ser un lenguaje tan conocido, no haré especial hincapié en él, pero sí explicaré algunas de sus *características principales*, ya que ha sido el lenguaje utilizado en el desarrollo de la aplicación, y la naturaleza de su lenguaje explica la organización del proyecto y de esta memoria:

- **Simple:** introduce toda la funcionalidad de un lenguaje potente, pero sin las características menos usadas y más confusas de éstos. Elimina la necesidad de liberar memoria, la aritmética de punteros, o palabras reservadas como struct o typedef por clases.
- **Orientado a objetos:** trabaja con sus datos como objetos y con interfaces a estos objetos. Soporta las tres características propias del paradigma de la orientación a objetos: encapsulación, herencia y polimorfismo. Las plantillas de objetos son llamadas clases, y sus copias instancias.
- **Distribuido:** se ha construido con extensas capacidades de interconexión TCP/IP (con librerías de rutinas para acceder e interactuar con protocolos como http y ftp).

Esto permite a los programadores acceder a la información a través de la red con tanta facilidad como a los ficheros locales. En sí java no es distribuido, pero sí proporciona las librerías y herramientas para que los programas puedan serlo.

- **Robusto:** realiza verificaciones en busca de problema tanto en tiempo de compilación como de ejecución. La comprobación de tipos ayuda a detectar errores lo antes posible en el ciclo de desarrollo. Proporciona, pues comprobación de punteros, de límites de arrays, excepciones, y verificación de bytecodes.
- **Seguro:** las aplicaciones de Java no acceden a zonas delicadas de memoria o de sistema, evitando la interacción de ciertos virus. No posee una semántica específica para modificar la pila de programa o la memoria libre o utilizar objetos y métodos de un programa sin los privilegios de kernel del sistema operativo. Además el cargador de clases puede verificar una firma digital antes de realizar una instancia de un objeto para evitar que se cree o almacene en memoria sin que se hayan validado los privilegios de acceso.
- **Portable:** además de la portabilidad básica por ser de arquitectura independiente, implementa otros estándares de portabilidad para facilitar el desarrollo.
- **Interpretado:** el interprete java (sistema run-time) puede ejecutar directamente el código abierto. Enlazar un programa normalmente consume menos recursos que compilarlo, por lo que los desarrolladores con java pasarán más tiempo desarrollando y menos esperando por el ordenador.
- **Multihilo:** permite múltiples actividades simultáneas en un programa que deriva en un mejor rendimiento interactivo y mejor comportamiento en tiempo real.
- **Dinámico:** java se beneficia todo lo posible de la tecnología orientada a objetos, de forma que no conecta todos los módulos que comprenden una aplicación hasta el mismo tiempo de ejecución. Además, implementa las opciones de persistencia para que no se eliminen los módulos de ByteCode ( o los objetos o nuevas clases) cuando se limpia la caché de la máquina.

## 2.2 ECLIPSE



*Figura 2. Logo eclipse.*

Eclipse es un entorno de desarrollo integrado (IDE), es decir, un entorno de programación empaquetado como un programa de aplicación que consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica.

Está compuesto por un conjunto de herramientas de programación de código abierto multiplataforma, y aunque no tienen en mente ningún lenguaje específico, ofrece un marco de trabajo amigable para la mayoría de los lenguajes de programación tales como C++, Python o Java.

Las principales características de Eclipse son las siguientes:

- Perspectivas, editores y vistas- El concepto de trabajo se basa en las perspectivas, que no es otra cosa que una preconfiguración de ventanas y editores relacionadas entre sí que nos permiten modificar el entorno de trabajo para trabajar de forma óptima.
- Gestión de proyectos: el desarrollo sobre Eclipse se basa en los proyectos, dividido en distintos recursos relacionados entre sí: código fuente, documentación, ficheros de configuración etc. IDE proporciona asistentes y ayudas para la creación de estos proyectos.
- Depurador de código: potente y de uso fácil e intuitivo.
- Extensa colección de *plugins*- Los plugins son componentes de código que amplían las funcionalidades de un programa o herramienta (algo así como complementos en español). Eclipse cuenta con muchísimos de ellos, algunos publicados por ellos mismos y otros por terceros.

Uno de los plugins más utilizados y que además también usamos para desarrollar la aplicación a la que esta memoria hace referencia, es el plugin JDT (Java Development Tools).

Este plugin es el encargado del soporte del IDE al lenguaje Java, incluido en la versión estándar de eclipse por defecto, que como ya expliqué, no está concebido para dar soporte a un lenguaje determinado. Algunas de las ventajas que este plugin aporta son por ejemplo el coloreado del código, code completion, o generadores de esqueletos de clase.

### **2.3 *MYSQL Y MYSQL WORKBENCH***



*Figura 3. Logo MySQL*

MySQL es un sistema de gestión de bases de datos relacional con considerada como la base de datos de código abierto más popular del mundo junto con Oracle y Microsoft SQL Server, sobre todo para entornos desarrollo web.

Como interfaz gráfica de usuario utilicé MySQL Workbench, que es el entorno integrado oficial de MySQL. Esta GUI permite administrar gráficamente las bases de datos MySQL y diseñar visualmente las estructuras de las bases de datos, configuración de servidores, administración de usuarios, copias de seguridad y mucho más.

Soporta completamente las versiones del servidor MySQL 5.5 y superiores, y compatible con algunas versiones anteriores con alguna funcionalidad menos. Este programa nos va a permitir entre otras cosas crear modelos de esquema de base de datos de manera gráfica, observar y modificar nuestras bases de datos en forma de esquemas, y editar todos los aspectos de una base de datos con el editor de tablas. Este editor proporciona facilidades para editar tablas, columnas, índices, inserciones, etc.

En resumen, proporciona un conjunto de herramientas para aprovechar al máximo y mejorar el rendimiento de las aplicaciones MySQL.

Puede descargarse la versión Open Source gratuitamente desde la página oficial de la herramienta.

## 2.4 MAVEN



*Figura 4. Logo Maven*

Maven es una herramienta software desarrollada por Apache Software Foundation para la gestión y construcción de proyectos Java.

Maven utiliza el Project Object Model (POM) para describir el proyecto de software a construir, sus dependencias de otros módulos y componentes externos, así como el orden de construcción de los elementos.

La característica principal de Maven es su capacidad de descargar dinámicamente plugins de un repositorio en la web. Está construido alrededor de la idea de **reutilización**, y más específicamente a la reutilización de la lógica de construcción.

El ciclo de vida de cualquier proyecto Maven se divide en 5 fases:

1. Compile: genera los ficheros .class compilando las fuentes .java.
2. Test: ejecuta los comandos de JUnit existentes, abortando si alguno falla. JUnit es un conjunto de clases (framework) que permite realizar la ejecución de clases Java de manera controlada para así evaluar si el funcionamiento de cada uno de los métodos de la clase se comporta como se espera.
3. Package: genera el fichero .jar con los .class compilados.
4. Install: copia el fichero .jar a un directorio de nuestro ordenador donde maven deja todos los .jar.
5. Deploy: copia el fichero .jar a un servidor remoto.

Cuando se ejecuta cualquiera de los comandos maven, maven irá verificando todas las fases del ciclo de vida desde la primera hasta la del comando. Otras metas como clean (elimina todos los .class y .jar) están fuera del ciclo de vida, pero también existen.

A veces es difícil explicar la necesidad de usar una herramienta como Maven. Normalmente al trabajar con Java/ JavaEE el uso de librerías es algo común (como en cualquier otro lenguaje de programación).

El concepto de librería es un concepto limitado, sin embargo, ya que no vale simplemente con querer utilizar cierta librería, sino que además necesitaremos saber la versión exacta o las librerías de las que depende para funcionar correctamente.

Maven solventa este problema a través del concepto de artefacto (artifact). Un artefacto contiene las clases propias de la librería, así como toda la información necesaria para su correcta gestión: grupo, versión, dependencias, etc.

Todos estos artefactos se definen en el POM.xml, que conectándose al repositorio de maven nos permite una gestión correcta de nuestras librerías, proyectos y dependencias.

## **2.5 JAVASCRIPT**

JavaScript es un lenguaje de programación que no requiere de compilación. Se utiliza en la parte cliente en el desarrollo de páginas Web, permitiendo añadir funcionalidades o mejoras en la interfaz de usuario y páginas Web dinámicas.

Utiliza una sintaxis similar al del lenguaje Java, pero no está orientado a objetos, ya que no dispone de herencia, sino que es un lenguaje basado en prototipos.

Se he utilizado este lenguaje para complementar el desarrollo principal del proyecto añadiendo funciones para validar datos o modificar la interfaz de usuario según la interacción de este con él.

## **Capítulo 3. ESTADO DE LA CUESTIÓN**

El análisis de datos es la ciencia encargada de examinar un conjunto de datos con el propósito de sacar conclusiones sobre la información, que permitan tomar decisiones o ampliar conocimientos sobre diversos temas. El concepto de Big Data aplica para toda la información que, por su gran tamaño, no puede ser procesada o analizada utilizando procesos o herramientas tradicionales. Es ahí donde entran en juego las herramientas de Big Data.

Desde el análisis de fenómenos naturales como el clima, hasta el diagnóstico para la seguridad, turismo, o energía son sectores en los que las herramientas Big Data tienen un papel cada vez más importante.

En el ámbito empresarial, el análisis de datos permite saber a las compañías el próximo paso que deben tomar, descubrir patrones de consumo y correlaciones para darles una percepción clara del mercado y así tomar decisiones inteligentes y seguras.

Una de las compañías más conocidas que utiliza el Big Data para tomar decisiones de negocio es el gigante del Retail online Amazon. Amazon adoptó tecnología de vanguardia para recolectar, analizar y utilizar la cantidad masiva de datos a la que tienen acceso a partir del historial de búsqueda y de compra de una persona, permitiéndoles mostrar a sus clientes los productos que buscan y necesitan, más rápido y económico que cualquiera de sus competidores. Todo esto se reflejó en un crecimiento de ventas del 20%, es decir, 8 billones de dólares aproximadamente.

El Big Data tiene cada vez un papel más importante en el procesamiento de información, y es por eso por lo que en la actualidad se están desarrollando cada vez más herramientas que faciliten su uso. A continuación, hablaré de algunas de las herramientas más utilizadas, y explicaré cómo esta aplicación podría fusionarse con ellas diferenciándose de todo lo existente en el mercado actual.



La Ingesta de datos es el primer paso y una de las piezas fundamentales en una arquitectura Big Data. Se trata del proceso mediante el cual se introducen datos, de diferentes fuentes, estructura o características dentro de otros sistemas de almacenamiento o procesamiento de datos. Una forma de conseguir esta mudanza de datos es mediante el uso de ficheros de parametrización, que, al ejecutarse, utilizan alguna de las herramientas o procesos de ingesta de datos existentes.

Entre las herramientas de ingesta de datos más utilizadas se encuentra Apache Nifi. Apache Nifi es una plataforma integrada de procesamiento y logística de datos en tiempo real, para automatizar el movimiento de datos entre diferentes sistemas de forma rápida, fácil y segura. Carga datos de diferentes fuentes, los pasa por un flujo de procesos para su tratamiento, y los vuelca en otra fuente. La ingesta de información se realiza de forma que no afecta al rendimiento ni de la fuente de origen ni de destino, es capaz de transformar numerosos formatos de datos y admite fuentes dispersas y distribuidas.

Otra herramienta muy utilizada es Apache Sqoop. Esta herramienta de línea de comandos desarrollada para transferir grandes volúmenes de datos de bases de datos leyendo línea por línea cada tabla e importándolas a HDFS (Hadoop Distributed File System). Soporta cargas incrementales de datos, multitud de conectores, y permite programar aplicaciones que la utilicen.

La última herramienta de la que voy a hablar es Apache Flume. Este servicio distribuido mueve de forma fiable y eficiente grandes cantidades de datos, especialmente logs, y es ideal para aplicaciones de analíticas en línea en entornos Hadoop.

En la actualidad, todos estos distintos procesos de ingesta de datos se realizan de forma separada y utilizando sus propios interfaces web. La aplicación desarrollada supondría una interfaz para trabajar con distintos procesos de ingesta de datos mediante la gestión de ficheros ejecutables que utilizarasen el proceso de ingesta de datos más conveniente. La aplicación funcionaría independientemente del proceso ejecutado, por lo que no funcionaría únicamente para un tipo de arquitectura Big Data. Además, la ejecución de

estos ficheros podría realizarse o bien al momento, o bien programarse para realizarse un día y a una hora concreta (puntual o periódicamente).

En la actualidad, no existen en el mercado aplicaciones que permitan esto, por lo que supondría una primera versión para el desarrollo de un sistema que permitiese la gestión de distintos procesos de ingesta de datos.

Como la gestión de distintos procesos implica la gestión de distintos proyectos, la aplicación no solo debe ejecutar ficheros, sino que debe encargarse de que estos se distribuyan y ordenen de la forma más adecuada dentro de directorios, y estos a su vez dentro de proyectos (organizados en distintos entornos). En la aplicación además existirán distintos tipos de usuarios, que según su perfil tendrán unos permisos u otros.

La gestión de usuarios es un área de entidad propia, en la medida en que la relación de estos con los sistemas es crucial a la hora de asegurar el correcto funcionamiento de los servicios puestos a su disposición. Es indispensable mantener un riguroso control de acceso a la información para garantizar la seguridad de los datos de un sistema de gestión. Mediante la concesión de permisos puede garantizarse esa seguridad, permitiendo el acceso a parte o a toda la información a determinados usuarios.

Los clientes del sistema deben ser capaces de autogestionar el mantenimiento de los distintos usuarios, para lo cual la aplicación desarrollada permitirá a los administradores la asignación y modificación de perfiles con sus consiguientes permisos, así como la creación o eliminación de posibles usuarios del sistema. De la misma forma los usuarios administradores podrán crear, modificar o eliminar proyectos del sistema.

Nuestra aplicación por lo tanto se encargará de la gestión de procesos de ingesta de datos mediante la gestión de ficheros ejecutables, proyectos en los que se encuentran, usuarios que los utilizan, entornos a los que pertenecen, y permisos que estos disponen. Esta primera versión se centrará en crear la base para una aplicación que gestione todo lo mencionado, y por lo tanto suponga una herramienta diferencial para aquellas empresas interesadas en el análisis de datos y el Big Data.

## **Capítulo 4. DEFINICIÓN DEL TRABAJO**

### **4.1 JUSTIFICACIÓN**

Impulsado por la constante llegada de nuevos actores y tecnologías, el mercado del análisis de datos está evolucionando a pasos agigantados. Tal y como dijo Antonio Conde, director de IoT y Transformación Digital en Cisco España: “Los datos son *el nuevo petróleo*; pieza clave de la sociedad y la economía”. Esto es así porque el dato es el nuevo valor a gestionar, y dominarlo se ha convertido en el punto de inflexión de las empresas.

En el 2018 New Vantage publicó su sexta edición de Encuesta de Ejecutivos, enfocada al análisis del Big Data y el Business Intelligence en empresas de éxito como Motorola o American Express. La encuesta mostró que el 92.7% invertían en Big Data, y que el 62.5% de los participantes tenían asignado un puesto de Chief Data Officer (CDO).

Estos números apoyan la teoría de la importancia del Big Data, y son la razón por la que se desarrolla un entorno que facilite la gestión de estos procesos. Esta aplicación responde a una necesidad del mundo actual, ayudando a gestionar los procesos de ingesta de datos y facilitando su parametrización, ejecución y revisión.

### **4.2 OBJETIVOS**

El objetivo concreto de este proyecto es el de disponer de una plataforma para la gestión de procesos de ingesta de datos en un entorno de BI/Big Data, y para conseguirlo se fijan cuatro objetivos principales. Estos objetivos se centran en la consecución del objetivo principal de la aplicación: la viabilidad de la implantación de un sistema de gestión de distintos procesos de ingesta de datos.

- Gestión de usuarios y sus perfiles dentro de la aplicación, permitiendo así un control sobre los permisos de acceso a la aplicación y a los diferentes servicios que

ofrece el sistema. Los usuarios finales deberán ser capaces de autogestionar este aspecto del sistema, posibilitando la creación, modificación y baja de usuarios.

- Gestión de proyectos dentro del sistema, así como de la estructura de directorios contenida en ellos y su relación con los distintos usuarios. El mantenimiento de estos proyectos también debe ser autogestionable por parte de los usuarios finales de la aplicación, de forma que se puedan crear, modificar y eliminar proyectos, así como diseñar la estructura de directorios dentro de estos.
- Gestión de ficheros (tanto de texto como ejecutables) dentro de los árboles de directorios de los distintos proyectos. Posibilidad de crear nuevos ficheros o importarlos, editarlos y ejecutarlos. Estas ejecuciones podrán además programarse para realizarse en un momento puntual o periódicamente.
- Recuperación de trazas de ejecución de los ficheros ejecutables para su posterior revisión. Además de la trazas de ejecución deberá recuperarse toda la información que pueda ser relevante para su posterior revisión.

El primer objetivo descrito es el de la gestión de diferentes usuarios (administradores y desarrolladores) y permisos de acceso de los usuarios al sistema de gestión de procesos ETL.

En función del rol que los usuarios posean, estos podrán gestionar los diferentes entornos (desarrollo, preproducción y producción) y permisos asociados a cada uno de los entornos (despliegue, ejecución y programación). Los usuarios de administración tendrán permisos completos en cada uno de los diferentes entornos. Los administradores podrán además dar de alta proyectos en cada uno de los entornos y vincular los usuarios de tipo desarrollador a dichos proyectos. Se da una explicación más detallada sobre los distintos perfiles y permisos de la aplicación en el apartado 5.2.1 (Funcionamiento de la aplicación).



*Figura 5. Ejemplo estructura de directorios.*

Se podrá crear para cada proyecto una estructura de directorios. Algo de este estilo:

Dentro de cada directorio, podrán subirse ficheros de texto (*.txt*) dando la opción de ser editados manualmente, y ficheros ejecutables (*.jar* inicialmente, aunque deberá dejarse abierto para poder ejecutar otro tipo de ficheros ejecutables en el futuro, *.py*, *.rb*, etc.).

Estos ejecutables podrán ser ejecutados en el momento actual, o programar su ejecución de una forma puntual o periódica. Inicialmente la ejecución de los procesos se realizará en la propia máquina/servidor.

Además, la aplicación deberá recuperar todas las trazas de ejecución que se generen durante la ejecución para su posterior visualización y revisión. Deberán mostrarse tiempos de ejecución, excepciones y errores y toda aquella información relevante para la revisión de los procesos.

### **4.3 METODOLOGÍA**

El desarrollo de la aplicación se dividirá en 3 fases.

En la primera fase se deberá crear un portal con acceso para dos tipos de perfiles de usuarios, administradores y desarrolladores. Ambos accederán a través de un formulario mediante usuario y contraseña. Aunque inicialmente la autenticación de usuarios se haga

contra base de datos, deberá dejarse abierta la implementación para su ampliación a otros posibles sistemas de autenticación (LDAP, Directorio Activo, etc.) Los usuarios de tipo administrador podrán gestionar el alta-baja-actualización de usuarios de perfil desarrollador.

Por otro lado, la aplicación deberá permitir a los usuarios de tipo administrador la gestión de diferentes entornos (desarrollo, preproducción, producción) con sus consiguientes permisos (ejecución, despliegue, programación). Los usuarios de administración tendrán siempre permisos completos en cada uno de los diferentes entornos.

Finalmente, los administradores podrán dar de alta proyectos en cada uno de los entornos y vincular los usuarios de tipo desarrollador asociados a dichos proyectos.

La segunda fase del proyecto se centrará en cumplir con el requisito de la creación de una estructura de directorios dentro de los proyectos.

Dentro de cada directorio, los usuarios podrán subir ficheros de texto (*.txt*) dando la opción de ser editados manualmente, y ficheros ejecutables (*.jar* inicialmente, aunque deberá dejarse abierto para poder ejecutar otro tipo de ficheros ejecutables en el futuro, *.py*, *.rb*, etc.). Para los ficheros ejecutables la aplicación dará la opción de ejecutarlos en el momento actual o programar su ejecución a una hora puntual o periódica. Inicialmente, la ejecución de los procesos se realizará en la propia máquina/servidor.

La tercera fase se centrará en la recuperación de las trazas de ejecución (logs) que se generen en las diferentes ejecuciones y su visualización para su posterior revisión. En la visualización, la aplicación deberá mostrar tiempos de ejecución, excepciones/errores y toda aquella información relevante para la revisión de los procesos.

En la siguiente página se muestra una planificación aproximada (Diagrama de Gantt) del proyecto, con inicio en noviembre del 2019 y finalización en julio de 2020.

Las filas representan cada actividad a realizar y las columnas una escala de tiempos.

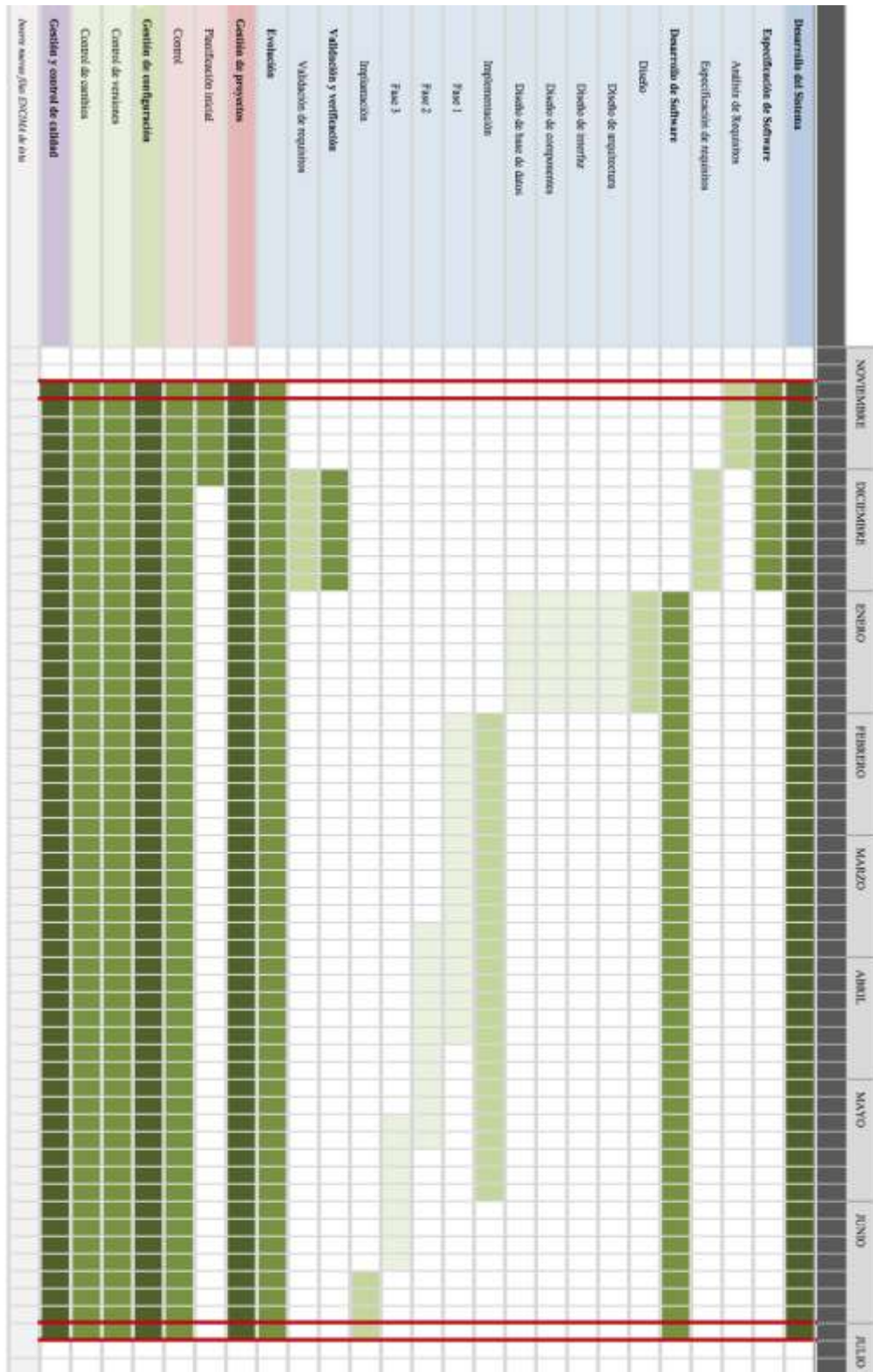


Figura 6. Diagrama de Gantt proyecto



## Capítulo 5. SISTEMA/MODELO DESARROLLADO

### 5.1 ARQUITECTURA Y PATRÓN DE DISEÑO

Una aplicación Web es una aplicación informática que se ejecuta en un entorno Web. La arquitectura Web permite a múltiples clientes o usuarios establecer una comunicación a través de Internet con un servidor Web.

Los clientes mediante un navegador emiten peticiones vía http a un servidor Web que responde a cada una de las solicitudes. El servidor aloja una serie de aplicaciones que ofrecen diferentes servicios a los usuarios conectados. Es aquí donde se centraliza la carga de trabajo, permitiendo al navegador ser un mero presentador de información (modelo de cliente ligero).

La comunicación entre cliente y servidor se efectúa mediante el uso del protocolo HTTP (Hypertext Transfer Protocol).

El usuario interactúa con las aplicaciones Web a través del navegador. Como consecuencia de esta actividad se envían peticiones al servidor, donde se aloja la aplicación Web y que a su vez hace uso de una base de datos que almacena toda la información relacionada con la misma. El servidor procesa la petición y devuelve la respuesta al navegador que la presenta al usuario.

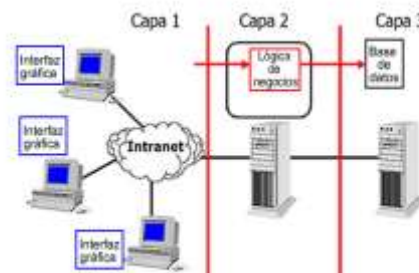
El sistema por lo tanto se distribuye en tres componentes:

- El **navegador (capa de presentación)**: presenta la interfaz al usuario. Comunica la información y captura la información de usuario. Únicamente se comunica directamente con la aplicación o lógica de negocios.
- La **aplicación (capa de negocio)**: realiza las operaciones necesarias según las acciones llevadas a cabo por el usuario. Recibe peticiones del usuario y envía las respuestas tras el proceso. Constituye el verdadero núcleo de la aplicación,



conteniendo toda la lógica que modela los procesos de negocio y realizando todo el procesamiento necesario para atender las peticiones del usuario. Se comunica con el navegador, del que recibe solicitudes y en el que presenta los resultados, y con la base de datos, donde almacena y recupera la información persistente relacionada con la aplicación.

- La *base de datos (capa de datos)*: guarda la información relacionada con la aplicación. Formada por uno o más gestores de datos. Únicamente se comunican con la capa de negocio.

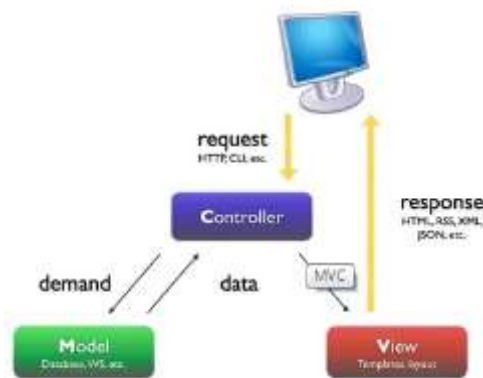


*Figura 7. Arquitectura de 3 capas*

Esta distribución se conoce como el modelo o arquitectura de tres capas, y proporciona algunas ventajas:

- **Facilidad de desarrollo:** puede llevarse a cabo en varios niveles, posibilitando la realización de desarrollos paralelos (en cada capa).
- **Robustez:** gracias al encapsulamiento.
- **Facilidad de mantenimiento y soporte:** el código dividido en módulos facilita la sustitución o modificación de los elementos.
- **Flexibilidad:** de nuevo la modularidad facilita añadir o modificar componentes sin afectar a los restantes.
- **Reutilización:** el código de la capa de negocio puede reutilizarse por múltiples aplicaciones si está diseñado en formato modular.
- **Alta escalabilidad:** capacidad para manejar muchas peticiones con el mismo rendimiento simplemente añadiendo más hardware (sin necesidad de modificar o añadir código).

El patrón compatible con esta distribución, y utilizado para el desarrollo al que esta memoria hace referencia es el patrón MVC (Modelo Vista Controlador). Es un patrón de arquitectura software encargado de separar la lógica de negocio de la interfaz de usuario y es el más utilizado en aplicaciones Web por facilitar la funcionalidad, mantenibilidad y escalabilidad de nuestros sistemas.



*Figura 8. Patrón MVC*

La anterior figura muestra esta arquitectura. En ella se puede ver cómo se relacionan estos tres bloques funcionales entre sí, así como su interacción con el resto de las capas de la aplicación.

### **Controlador:**

Intermedio entre la vista y el modelo, el “cerebro” de la aplicación. Todas las peticiones a la capa intermedia que se realicen desde el cliente son dirigidas al controlador, cuya misión es determinar las acciones a realizar para cada una de las peticiones e invocar al resto de componentes de la aplicación (Modelo y Vista) para que realicen las acciones requeridas en cada caso.

En nuestra aplicación en controlador es el conjunto de Servlets incluidos en el paquete *Controller*.

En definitiva, se encarga de controlar las interacciones del usuario solicitando o pasando datos al modelo para después entregarlos a la vista y que esta lo presente al usuario.

### **Vista:**

Es la encargada de mostrar la información al usuario permitiéndole interactuar con la aplicación. El código de la página deberá ser generado de forma dinámica para así poder incluir datos proporcionados por el controlador.

Las vistas pueden dividirse en: la presentación de datos (gráficos, tablas...), formularios de ingreso de datos, pestañas de navegación (menús, hipervínculos...), o ventanas emergentes o pop-ups.

En una aplicación Web estas vistas son páginas HTML (HTML o JSP en nuestra aplicación) que el usuario visualiza en el navegador y con las que interactúa enviando eventos al servidor a través de peticiones HTTP. El servidor actuará sobre el modelo convenientemente en función al evento concreto y devolverá los resultados al usuario mediante la respuesta HTTP.

### **Modelo:**

La lógica de negocio de la aplicación: acceso a los datos y su manipulación. Aunque hacen uso de él, es independiente al controlador y la vista, permitiendo así su reutilización y desacoplamiento entre las capas.

En el caso de nuestra aplicación incluye las clases incluidas dentro del paquete *Model*. Este paquete se divide a su vez en *service* y *dao* para mejorar la limpieza y división del código evitando que en controlador utilice de forma directa los métodos de las clases DAO (donde se realiza la conexión a la base de datos). El funcionamiento básico del patrón es el siguiente:

1. El usuario realiza una petición a través de la interfaz de usuario.
2. El controlador captura el momento.
3. El controlador llama a los modelos correspondientes y efectúa las modificaciones pertinentes sobre el modelo.
4. El controlador recibe información del modelo y la pasa a la vista.
5. La vista recibe los datos y los muestra al usuario.

## 5.2 REQUISITOS DE LA APLICACIÓN

En este apartado definiré los requisitos de la aplicación, es decir, qué servicios se requieren y las restricciones en la operación del sistema y en el desarrollo. Dividiré esta sección en dos: la especificación de requisitos de usuario, y la de los requisitos de sistema, los cuales a su vez separé en funcionales y no funcionales.

### 5.2.1 REQUISITOS DE USUARIO

Esta sección describe los servicios que se espera que ofrezca el sistema en lenguaje natural. La especificación de requisitos de usuario se realiza en una tabla que contiene las siguientes propiedades:

- Identificador: unívoco del requisito de usuario.
- Descripción: descripción del requisito de usuario.
- Prioridad: indica la prioridad establecida para cada requisito (*alta*, *media* o *baja*) siendo el valor “*alta*” el de mayor prioridad y “*baja*” el de menor.
- Estabilidad: indica si el requisito está sujeto a modificaciones.

Identificador	Descripción	Prioridad	Estabilidad
<b>UR001</b>	Autenticación en la aplicación contra base de datos verificando correo electrónico y contraseña.	Alta	Modificable
<b>UR002</b>	Mostrar mensaje de usuario no existente en caso de que no exista ningún usuario registrado con el correo electrónico introducido.	Media	No modificable
<b>UR003</b>	Mostrar mensaje de contraseña incorrecta en caso de no introducir la contraseña correctamente, pero sí existir un usuario registrado con ese correo electrónico.	Media	No modificable
<b>UR004</b>	Mostrar mensaje de usuario dado de baja en caso de que el usuario y la contraseña introducidos sean correctos pero el usuario haya sido dado de baja.	Media	No modificable
<b>UR005</b>	Mostrar página de inicio con mensaje de	Baja	Modificable

	bienvenida, perfil que posees como usuario de la aplicación y entornos a los que tienes acceso.		
<b>UR006</b>	Mostrar en la página de inicio los proyectos a los que tienes acceso como usuario de la aplicación.	Media	No modificable
<b>UR007</b>	Mostrar proyectos en página de inicio en una tabla paginada.	Alta	Modificable
<b>UR008</b>	Filtro de búsqueda dentro de la tabla paginada de proyectos.	Media	Modificable
<b>UR009</b>	Pestaña de editar perfil para permitir al usuario ver sus datos y modificarlos. Los únicos datos que podrá editar serán su nombre y su contraseña.	Media	Modificable
<b>UR010</b>	Pestaña de cerrar sesión para salir de la aplicación.	Alta	No modificable
<b>UR011</b>	Un usuario con perfil de administrador tendrá dos pestañas que los usuarios de tipo desarrollador no poseen: gestión de usuarios, y gestión de proyectos.	Alta	No modificable
<b>UR012</b>	La página de gestión de usuarios muestra una lista de todos los usuarios de la aplicación con su nombre, perfil, entornos a los que tiene acceso, proyectos asignados y fecha de creación.	Alta	No modificable
<b>UR013</b>	Los administradores aparecerán en la tabla con los mensajes de “Todos” en los entornos accesibles y proyectos asociados.	Baja	Modificable
<b>UR014</b>	Perfil propio marcado de forma distintiva en la tabla.	Baja	Modificable
<b>UR015</b>	La tabla en la que se muestran todos los usuarios debe ser una tabla paginada con posibilidad de filtrar la búsqueda.	Alta	Modificable
<b>UR016</b>	Añadir un usuario.	Alta	No modificable
<b>UR017</b>	Editar un usuario.	Alta	No modificable
<b>UR018</b>	Dar de baja un usuario.	Alta	No modificable
<b>UR019</b>	Al añadir un usuario solicitar nombre y	Alta	No

	apellidos, correo electrónico y seleccionar perfil (desarrollador o administrador). Todos estos campos son obligatorios.		modificable
<b>UR020</b>	En caso de que el usuario sea desarrollador solicitar entornos asociados, y en función a los entornos seleccionados mostrar una lista con los proyectos disponibles.	Alta	No modificable
<b>UR021</b>	Filtro de búsqueda en la lista de proyectos disponibles.	Media	Modificable
<b>UR022</b>	Al añadir un usuario nuevo se generará una contraseña de 7 dígitos aleatoria que se mostrará en una ventana pop up.	Media	Modificable
<b>UR023</b>	Al editar un usuario podremos modificar nombre y apellidos, email, y perfil, así como los entornos a los que tiene acceso y los proyectos asociados.	Alta	No modificable
<b>UR024</b>	Si un usuario es modificado y pierde acceso a un entorno, automáticamente dejará de estar asociado a todos los proyectos asociados a dicho entorno.	Alta	No modificable
<b>UR025</b>	Mismo diseño en la página de editar y añadir usuarios.	Baja	Modificable
<b>UR026</b>	Mostrar mensaje de confirmación al solicitar dar de baja un usuario.	Alta	No modificable
<b>UR027</b>	Al dar de baja un usuario no se elimina completamente de la aplicación, sino que se le bloquea el acceso a esta.	Alta	No modificable.
<b>UR028</b>	Al añadir, editar o eliminar un usuario, la aplicación debe devolver a la página de gestión de usuarios mostrando la página de la tabla de usuarios en la que encontrabas antes del alta/baja/modificación del usuario.	Alta	No modificable
<b>UR029</b>	Las ventanas de añadir, editar y dar de baja un usuario se cerrarán o bien pulsando la cruz en la esquina superior derecha o bien pulsando en cualquier punto de la pantalla que no se encuentre dentro de la ventana.	Media	Modificable
<b>UR030</b>	La página de gestión de proyectos muestra	Alta	No

	una lista de todos los proyectos de la aplicación con su nombre (junto con el identificador único del proyecto), entorno al que pertenece, y usuarios asignados.		modificable
<b>UR031</b>	La tabla en la que se muestran todos los proyectos debe ser una tabla paginada con filtro de búsqueda.	Alta	Modificable
<b>UR032</b>	Posibilidad de añadir un proyecto.	Alta	No modificable
<b>UR033</b>	Posibilidad de editar un proyecto.	Alta	No modificable
<b>UR034</b>	Posibilidad de eliminar un proyecto.	Alta	No modificable
<b>UR035</b>	Al añadir un proyecto solicitar nombre y entorno. Estos dos campos son obligatorios.	Alta	No modificable
<b>UR036</b>	Al seleccionar un entorno para un proyecto mostrar todos los usuarios que tienen acceso a ese entorno en una lista.	Alta	No modificable
<b>UR037</b>	Filtro de búsqueda en la lista de usuarios disponibles.	Media	Modificable
<b>UR038</b>	Al editar un proyecto podremos modificar nombre entorno y usuarios asociados.	Alta	No modificable
<b>UR039</b>	Si un proyecto es modificado y cambia de entorno, automáticamente dejará de estar asociado a todos los usuarios (excepto a los administradores).	Alta	No modificable
<b>UR040</b>	Mismo diseño en la página de editar proyecto y añadir uno nuevo.	Baja	Modificable
<b>UR041</b>	Mostrar mensaje de confirmación al solicitar eliminar un proyecto.	Alta	No modificable
<b>UR042</b>	Al añadir, editar o eliminar un proyecto, la aplicación debe devolver a la página de gestión de proyectos mostrando la página de la tabla de proyectos en la que encontrabas antes del alta/baja/modificación del usuario.	Alta	No modificable
<b>UR043</b>	Las ventanas de añadir, editar y eliminar un proyecto se cerrarán o bien pulsando la cruz en la esquina superior derecha o bien	Media	Modificable

	pulsando en cualquier punto de la pantalla que no se encuentre dentro de la ventana.		
<b>UR044</b>	Al pulsar en un proyecto de la lista de proyectos de la página de inicio la aplicación te mandará a una nueva página para ese proyecto en particular.	Alta	No modificable
<b>UR045</b>	La página con la información relativa al proyecto debe mostrar tres paneles diferenciados: participantes, vistazo de directorios, y ejecuciones pasadas.	Alta	Modificable
<b>UR046</b>	Los participantes deben mostrarse divididos en administradores y desarrolladores.	Baja	Modificable
<b>UR047</b>	El panel de vistazo de directorios debe mostrar la estructura de directorios en forma de árbol.	Alta	No modificable
<b>UR048</b>	Los directorios en el árbol se mostrarán con un icono de una carpeta y el icono para los ficheros será distinto según sea ejecutable o de texto.	Media	Modificable
<b>UR049</b>	Al hacer <i>click</i> derecho en un elemento del árbol de directorios debe mostrar las acciones que se pueden realizar, que variarán según el tipo de elemento pulsado (directorio/fichero de texto/fichero ejecutable)	Alta	No modificable
<b>UR050</b>	Según los permisos de usuario que tengas podrás realizar unas acciones u otras.	Alta	No modificable
<b>UR051</b>	Si seleccionas la opción de cambiar nombre (tanto en un directorio como en un fichero) el nombre se hará editable en el árbol, y podrás cambiarlo y guardarlo pinchando en cualquier punto de la pantalla o pulsando <i>enter</i> .	Alta	No modificable
<b>UR052</b>	Si seleccionas la opción de nuevo directorio se abrirá una ventana que te pedirá el nombre del nuevo directorio y te mostrará la ruta que tendrá.	Alta	Modificable
<b>UR053</b>	Si al seleccionar crear en la ventana de crear directorio no has introducido un nombre saltará el mensaje "Debes	Alta	No modificable



	introducir un nombre”.		
<b>UR054</b>	Si seleccionas la opción de eliminar (tanto en un directorio como en un fichero) se abrirá una ventana para confirmar la eliminación.	Alta	No modificable
<b>UR055</b>	Si seleccionas la opción de nuevo fichero (de texto) se abrirá una ventana que te pedirá el nombre del nuevo fichero y te mostrará la ruta que tendrá.	Alta	No modificable
<b>UR056</b>	Al crear un nuevo fichero se abrirá automáticamente el editor de texto vacío para poder llenarlo de contenido y guardarlo.	Alta	No modificable
<b>UR057</b>	Si al seleccionar crear en la ventana de crear fichero no has introducido un nombre saltará el mensaje “Debes introducir un nombre”.	Alta	No modificable
<b>UR058</b>	Si seleccionas la opción de importar fichero se abrirá una ventana para seleccionar el fichero de tu ordenador.	Alta	No modificable
<b>UR059</b>	Si el fichero seleccionado al importarlo no es .txt o .jar no se guardará.	Alta	Modificable
<b>UR061</b>	Si el fichero importado es .txt se abrirá automáticamente en un panel que te permitirá editarlo.	Media	No modificable
<b>UR062</b>	Si seleccionas la opción de abrir fichero el fichero se abrirá en un panel que te permitirá editarlo.	Alta	No modificable
<b>UR063</b>	Si seleccionas la opción de ejecutar fichero ejecutable se abrirá una ventana que te permitirá introducir ciertos parámetros de ejecución: el grupo inicial y máximo de asignación de memoria.	Alta	Modificable
<b>UR064</b>	Si pulsas en ejecutar en la ventana de “ejecutar fichero”, se ejecutará el fichero que corresponda al momento y el resultado de su ejecución se cargará en la tabla de ejecuciones de la ventana principal del proyecto.	Alta	No modificable
<b>UR065</b>	Si seleccionas la opción de programar ejecución en un fichero ejecutable se	Alta	No modificable

	abrirá una ventana que te permitirá introducir ciertos parámetros de ejecución y la fecha y la hora a la que ejecutar el fichero, así como el tipo de ejecución (puntual o periódica).		
<b>UR066</b>	Si pulsas en programar ejecución en la ventana de “ejecutar fichero”, se configurará una ejecución a la hora y fecha establecida, y una vez se ejecute se cargará el resultado de la ejecución en la tabla de ejecuciones de la ventana principal del proyecto.	Alta	No modificable
<b>UR067</b>	Las ventanas de nuevo directorio, nuevo fichero, eliminar directorio, eliminar fichero, ejecutar fichero y programar ejecución fichero se cerrarán o bien pulsando la cruz en la esquina superior derecha, en cancelar, o en cualquier punto de la pantalla que no se encuentre dentro de la ventana.	Alta	No modificable
<b>UR068</b>	Una ejecución debe guardar todas sus características (hora, fecha, nombre de fichero ejecutado, usuario que lo ha ejecutado, etc..) además de generar un fichero log con la traza de ejecución.	Alta	No modificable
<b>UR069</b>	La tabla que muestra las ejecuciones pasadas debe ser una tabla paginada con filtro de búsqueda.	Media	No modificable
<b>UR070</b>	La tabla que muestra las ejecuciones pasadas en la ventana de un proyecto tendrá las siguientes columnas: Fichero, Fecha, Hora Inicio, Hora fin, Resultado, Usuario, Parámetros de ejecución, y un botón para descargar el fichero log de la ejecución.	Alta	Modificable
<b>UR071</b>	Si la ejecución ha fallado el resultado de la ejecución se mostrará en color rojo, y si ha sido exitosa en color verde.	Baja	Modificable
<b>UR072</b>	Al pulsar en el botón para descargar el log de una ejecución se mostrará el fichero en la parte inferior de la pantalla como fichero descargable.	Alta	No modificable

<b>UR073</b>	Todos los ficheros se guardarán en el servidor con el nombre correspondiente al hash de su nombre real para evitar que alguien sepa que contiene cada fichero en caso de tener acceso al servidor.	Alta	No modificable
<b>UR074</b>	La aplicación será de fácil manejo para un usuario sin previo conocimiento de esta.	Alta	No modificable
<b>UR075</b>	La autenticación de usuarios se hace contra base de datos, aunque deberá dejarse abierta.	Media	No modificable

*Tabla 1. Requisitos de usuario*

## 5.2.2 REQUISITOS DE SISTEMA

Esta sección describe más detallada y consistentemente lo que se espera de la aplicación. ’

He dividido esta especificación en requisitos funcionales, que definen los servicios que el sistema debe proporcionar, y no funcionales, que serán aquellas restricciones que afectan a los servicios o funciones del sistema.

La especificación de requisitos de sistema se realiza en una tabla que contiene las siguientes propiedades:

- Identificador: unívoco del requisito de usuario.
- Descripción: descripción del requisito de usuario.
- Prioridad: indica la prioridad establecida para cada requisito (*alta, media* o *baja*) siendo el valor “*alta*” el de mayor prioridad y “*baja*” el de menor.
- Estabilidad: indica si el requisito está sujeto a modificaciones.
- Fuente: procedencia de la especificación del requisito, es decir, el requisito de usuario al que hace referencia. Esta columna es útil para poder crear más tarde la matriz de trazabilidad.

### 5.2.2.1 REQUISITOS FUNCIONALES:

Identificador	Descripción	Prioridad	Estabilidad	Fuente
<b>SR001</b>	Autenticación mediante correo electrónico y contraseña que serán verificados contra base de datos.	Alta	Modificable	<i>UR001</i>
<b>SR002</b>	Comprobar correo electrónico introducido y mostrar mensaje de usuario no existente en caso de que no exista en la tabla.	Media	No modificable	<i>UR002</i>
<b>SR003</b>	Comprobar correo electrónico y contraseña y mostrar mensaje de contraseña incorrecta en caso de que el usuario exista, pero la contraseña no sea correcta.	Media	No modificable	<i>UR003</i>
<b>SR004</b>	Comprobar correo electrónico y contraseña y mostrar mensaje de usuario dado de baja en caso	Media	No modificable	<i>UR004</i>

	de que su fecha de baja no sea nula.			
<b>SR005</b>	Mostrar información básica al entrar en la aplicación: nombre, perfil y entornos a los que se tiene acceso.	Baja	Modificable	<i>UR005</i>
<b>SR006</b>	Recuperar proyectos a los que el usuario tiene acceso.	Media	No modificable	<i>UR006</i>
<b>SR007</b>	Mostrar proyectos en página de inicio en una tabla paginada.	Alta	Modificable	<i>UR007</i>
<b>SR008</b>	Crear filtro de búsqueda dentro de la tabla paginada de proyectos.	Media	Modificable	<i>UR008</i>
<b>SR009</b>	Mostrar pestaña de editar perfil para permitir al usuario ver sus datos y modificarlos.	Media	Modificable	<i>UR009</i>
<b>SR010</b>	Permitir únicamente la modificación del nombre y la contraseña.	Media	Modificable	<i>UR009</i>
<b>SR011</b>	Para cambiar la contraseña es obligatorio primero introducir correctamente la contraseña antigua. En caso de que esta sea incorrecta se avisará con un mensaje de aviso.	Media	No modificable	<i>UR009</i>
<b>SR012</b>	Al introducir tu nueva contraseña deberás introducirla dos veces, y solo se cambiará si estas coinciden. En caso contrario se avisará con un mensaje de aviso.	Media	No modificable	<i>UR009</i>
<b>SR013</b>	Mostrar pestaña de cerrar sesión para eliminar tu sesión y salir de la aplicación.	Alta	No modificable	<i>UR010</i>
<b>SR014</b>	Comprobar tipo de perfil de usuario y mostrar pestaña de gestión de usuarios en caso de que el perfil sea de administrador.	Alta	No modificable	<i>UR011</i>
<b>SR015</b>	Comprobar tipo de perfil de usuario y mostrar pestaña de	Alta	No modificable	<i>UR011</i>

	gestión de proyectos en caso de que el perfil sea de administrador.			
<b>SR016</b>	La página de gestión de usuarios muestra una lista de todos los usuarios de la aplicación con su nombre, perfil, entornos a los que tiene acceso, proyectos asignados y fecha de creación.	Alta	No modificable	<i>UR012</i>
<b>SR017</b>	En las filas de la tabla en la que el usuario sea administrador, los entornos a los que tiene acceso y los proyectos asignados aparecerán como “Todos (admin)”.	Baja	Modificable	<i>UR013</i>
<b>SR018</b>	En la fila de la tabla de usuarios en la que aparezca tu propio usuario habrá una distinción y saldrá “Tú” al lado de tu nombre.	Baja	Modificable	<i>UR014</i>
<b>SR019</b>	La tabla en la que se muestran todos los usuarios debe ser una tabla paginada con filtro de búsqueda.	Alta	Modificable	<i>UR015</i>
<b>SR020</b>	Crear botón único encima de la tabla y a la izquierda del filtro de búsqueda para añadir nuevos usuarios.	Alta	No modificable	<i>UR016</i>
<b>UR021</b>	Crear un botón de editar usuario para cada usuario en la tabla, de forma que al pinchar en él te abra una página con su información y la posibilidad de editarla.	Alta	No modificable	<i>UR017</i>
<b>SR022</b>	Crear un botón de dar de baja usuario para cada usuario en la tabla.	Alta	No modificable	<i>UR018</i>
<b>SR023</b>	En la página de nuevo usuario deben crearse cuadros de texto para introducir el nombre, los	Alta	No modificable	<i>UR019</i>

	apellidos y el correo electrónico. Deben también mostrarse los tipos de perfiles de usuario existentes (administrador y desarrollador) y permitir seleccionar uno de ellos.			
<b>SR024</b>	En caso de que la selección del perfil de usuario sea de desarrollador, se mostrarán todos los entornos disponibles en casillas, y según la casilla seleccionada se mostrarán los proyectos disponibles en ese entorno.	Alta	No modificable	<i>UR020</i>
<b>SR025</b>	Filtro de búsqueda en la lista de proyectos disponibles.	Media	Modificable	<i>UR021</i>
<b>SR026</b>	Al añadir un nuevo usuario se verificará que se han rellenado todos los campos necesarios. Para cualquier tipo de perfil es obligatorio introducir nombre, apellidos, email y tipo de perfil.	Alta	No modificable	<i>UR019</i>
<b>SR027</b>	En caso de que el perfil a añadir sea de desarrollador se verificará que se ha seleccionado al menos un entorno.	Alta	No modificable	<i>UR019</i>
<b>SR028</b>	Al añadir un usuario nuevo se generará una contraseña de 7 dígitos aleatoria que se mostrará en una ventana pop up.	Media	Modificable	<i>UR022</i>
<b>SR029</b>	En la página de editar usuario deben crearse cuadros de texto con el nombre, los apellidos y el correo electrónico del usuario seleccionado. Deberá también mostrarse el tipo de usuario y los entornos y proyectos asociados en caso de que sea desarrollador. Se podrán modificar todos estos campos.	Alta	No modificable	<i>UR023</i>

<b>SR030</b>	Al editar un usuario se verificará que se han rellenado todos los campos necesarios. Para cualquier tipo de perfil es obligatorio introducir nombre, apellidos, email y tipo de perfil.	Alta	No modificable	<i>UR023</i>
<b>SR031</b>	En caso de que el perfil editado sea desarrollador se verificará que se ha seleccionado al menos un entorno.	Alta	No modificable	<i>UR023</i>
<b>SR032</b>	Si un usuario es modificado y pierde acceso a un entorno, automáticamente dejará de estar asociado a todos los proyectos asociados a dicho entorno.	Alta	No modificable	<i>UR024</i>
<b>SR033</b>	Mismo diseño en la página de editar usuario y añadir uno nuevo.	Baja	Modificable	<i>UR025</i>
<b>SR034</b>	Dar de baja un usuario significa cambiar su fecha de baja, no su eliminación de la tabla de usuarios. De esta forma el usuario no podrá acceder a la aplicación ni se podrá visualizar su información, pero no se eliminará su rastro completamente.	Alta	No modificable.	<i>UR027</i>
<b>SR035</b>	Las ventanas de añadir, editar y dar de baja un usuario se cerrarán o bien pulsando la cruz en la esquina superior derecha o bien pulsando en cualquier punto de la pantalla que no se encuentre dentro de la ventana.	Media	Modificable	<i>UR029</i>
<b>SR036</b>	La página de gestión de proyectos muestra una lista de todos los proyectos de la aplicación con su nombre (junto con el identificador único del proyecto), entorno al que pertenece, y los usuarios asignados al proyecto (equipo).	Alta	No modificable	<i>UR030</i>



<b>SR037</b>	La tabla en la que se muestran todos los proyectos debe ser una tabla paginada.	Alta	Modificable	UR030
<b>SR038</b>	Crear cuadro de texto que permita filtrar la búsqueda.	Alta	Modificable	UR031
<b>SR039</b>	Crear botón único encima de la tabla y a la izquierda del filtro de búsqueda para añadir nuevos proyectos.	Alta	No modificable	UR032
<b>SR040</b>	Crear un botón de editar proyecto para cada proyecto en la tabla, de forma que al pinchar en él te abra una página con su información y la posibilidad de editarla.	Alta	No modificable	UR033
<b>SR041</b>	Crear un botón de dar de eliminar proyecto para cada proyecto en la tabla.	Alta	No modificable	UR034
<b>SR042</b>	Al añadir un nuevo usuario se verificará que se han rellenado todos los campos necesarios. Para cualquier proyecto es obligatorio introducir nombre y entorno, en caso de no hacerlo se mostrará un mensaje apropiado.	Alta	No modificable	UR034
<b>SR043</b>	Los usuarios administradores serán automáticamente añadidos al equipo de todos los nuevos proyectos y no podrán deseccionarse.	Alta	No modificable	UR035
<b>SR044</b>	Al seleccionar un entorno para un proyecto mostrar todos los usuarios que tienen acceso a ese entorno en una lista.	Alta	No modificable	UR036
<b>SR045</b>	Crear filtro de búsqueda en la lista de usuarios disponibles para añadir al equipo del proyecto según el entorno seleccionado.	Media	Modificable	UR037
<b>SR046</b>	Al editar un proyecto podremos	Alta	No	UR038

	modificar nombre entorno y usuarios asociados.		modificable	
<b>SR047</b>	Si un proyecto es modificado y cambia de entorno, automáticamente dejará de estar asociado a todos los usuarios (excepto a los administradores).	Alta	No modificable	<i>UR039</i>
<b>SR048</b>	Mismo diseño en la página de editar proyecto y añadir uno nuevo.	Baja	Modificable	<i>UR040</i>
<b>SR049</b>	Las ventanas de añadir, editar y eliminar un proyecto se cerrarán o bien pulsando la cruz en la esquina superior derecha o bien pulsando en cualquier punto de la pantalla que no se encuentre dentro de la ventana.	Media	Modificable	<i>UR043</i>
<b>SR050</b>	Al pinchar en cada fila de la lista de proyectos de la página de inicio la aplicación te mandará a una nueva página para ese proyecto en particular.	Alta	No modificable	<i>UR044</i>
<b>SR051</b>	Además de los bloques para el menú superior y el título del proyecto, la página mostrará la información del proyecto dividida en tres bloques: vistazo de directorios, participantes, y ejecuciones pasadas.	Alta	Modificable	<i>UR045</i>
<b>SR052</b>	Los participantes deben mostrarse divididos en administradores y desarrolladores.	Baja	Modificable	<i>UR046</i>
<b>SR053</b>	El nombre propio aparecerá en la tabla de participantes como "Tú".	Baja	Modificable	<i>UR046</i>
<b>SR054</b>	Como máximo se mostrará una línea de participantes para cada tipo de perfil, es decir, una línea de participantes de desarrolladores y una de	Baja	Modificable	<i>UR046</i>

	administradores. Si hay demasiados se escribirán puntos suspensivos.			
<b>SR055</b>	El panel de vistazo de directorios debe mostrar la estructura de directorios en forma de árbol.	Alta	No modificable	UR047
<b>SR056</b>	La estructura de árbol se realizará utilizando el plugin de jquery jstree.	Alta	No modificable	UR047
<b>SR057</b>	El icono de los directorios en el árbol de directorios será una carpeta, el de los ficheros ejecutables un fichero con una tuerca, y el de los ficheros de texto un fichero con líneas.	Media	Modificable	UR048
<b>SR058</b>	Al hacer <i>click</i> derecho en el directorio raíz del árbol se desplegará un menú con la opción de “Nuevo directorio”.	Alta	Modificable	UR049
<b>SR059</b>	Al hacer <i>click</i> derecho en un directorio del árbol (que no sea el directorio raíz), se desplegará un menú con las opciones de “Nuevo directorio”, “Nuevo fichero”, “Importar fichero”, “Cambiar nombre” y “Eliminar”.	Alta	Modificable	UR049
<b>SR060</b>	Al hacer <i>click</i> derecho en un fichero de texto se desplegará un menú con las opciones de “Abrir”, “Cambiar nombre”, y “Eliminar”.	Alta	Modificable	UR049
<b>SR061</b>	Al hacer <i>click</i> derecho en un fichero ejecutable se desplegará un menú con la opción de “Ejecutar”, “Programar ejecución”, “Cambiar nombre”, y “Eliminar”.	Alta	Modificable	UR049
<b>SR062</b>	Los usuarios con permiso de despliegue tendrán habilitadas	Alta	No modificable	UR050

	las opciones de “Nuevo directorio”, “Nuevo fichero”, “Cambiar nombre”, “Eliminar”, “Importar fichero”, y “Nuevo fichero”.			
<b>SR063</b>	Los usuarios con permiso de ejecución tendrán habilitada la opción de “Ejecutar” en los ficheros ejecutables.	Alta	No modificable	UR050
<b>SR064</b>	Los usuarios con permiso de programación tendrán habilitada la opción de “Programar ejecución” en los ficheros ejecutables.	Alta	No modificable	UR050
<b>SR065</b>	Si seleccionas la opción de cambiar nombre (tanto en un directorio como en un fichero) el nombre se hará editable en el árbol, y podrás cambiarlo y guardarlo pinchando en cualquier punto de la pantalla o pulsando <i>enter</i> .	Alta	No modificable	UR051
<b>SR066</b>	Debe desactivarse usando javascript la opción de que se envíe un formulario al pulsar en la tecla <i>enter</i> para evitar errores al guardar un nombre.	Alta	No modificable	UR051
<b>SR067</b>	Si seleccionas la opción de nuevo directorio se abrirá una ventana con la ruta y un cuadro de texto para introducir el nombre del directorio a crear.	Alta	Modificable	UR052
<b>SR068</b>	Usando javascript conseguir que a medida que escribes el nombre del directorio se actualice el nombre de la ruta final del directorio a crear.	Baja	Modificable	UR053
<b>SR069</b>	Al pulsar en crear directorio validar que el nombre del directorio a crear no es nulo. Si lo es mostrar el mensaje “Debes introducir un nombre”.	Alta	No modificable	UR053

<b>SR070</b>	Si seleccionas la opción de eliminar (tanto en un directorio como en un fichero) se abrirá una ventana para confirmar la eliminación. Se elimina pulsando en “Eliminar”.	Alta	No modificable	UR054
<b>SR071</b>	Si seleccionas la opción de nuevo fichero se abrirá una ventana con la ruta y un cuadro de texto para introducir el nombre del fichero a crear.	Alta	No modificable	UR055
<b>SR072</b>	Usando javascript conseguir que a medida que escribes el nombre del fichero se actualice el nombre de la ruta final del fichero a crear.	Baja	Modificable	UR055
<b>SR073</b>	Al pulsar en crear fichero validar que el nombre del fichero a crear no es nulo. Si lo es mostrar el mensaje “Debes introducir un nombre”.	Alta	No modificable	UR057
<b>SR074</b>	Al crear un nuevo fichero se abrirá automáticamente un panel que funciona de editor de texto vacío.	Alta	No modificable	UR056
<b>SR075</b>	Al pulsar en la cruz superior derecha del panel del fichero se cierra el contenido del fichero sin guardarse.	Alta	No modificable	UR056
<b>SR076</b>	Al pulsar en guardar en el panel del fichero se cierra el panel del fichero y se guarda su contenido.	Alta	No modificable	UR056
<b>SR077</b>	Si seleccionas la opción de importar fichero se abrirá una ventana para seleccionar el fichero de tu ordenador.	Alta	No modificable	UR058
<b>SR078</b>	Las opciones por defecto que aparecerán de tipos de fichero a buscar deben ser solamente .txt y .jar, pero el usuario podrá	Alta	No modificable	UR058

	cambiarla (no existe la posibilidad de hacerlo de otra forma), por lo que deberá verificarse más tarde el tipo de fichero subido.			
<b>SR079</b>	Se verifica que el tipo de fichero sea .txt o .jar. Si no, no se guarda.	Alta	Modificable	<i>UR059</i>
<b>SR080</b>	Si el fichero importado es .txt se abrirá el panel de edición del fichero.	Media	No modificable	<i>UR061</i>
<b>SR081</b>	Si seleccionas la opción de abrir fichero el fichero se abrirá en el panel de edición de fichero.	Alta	No modificable	<i>UR062</i>
<b>SR082</b>	Si seleccionas la opción de ejecutar fichero ejecutable se abrirá una ventana que te permitirá introducir ciertos parámetros de ejecución: el grupo inicial y máximo de asignación de memoria.	Alta	Modificable	<i>UR063</i>
<b>SR083</b>	Si pulsas en ejecutar en la ventana de “ejecutar fichero”, se ejecutará el fichero que corresponda al momento y el resultado de su ejecución se cargará en la tabla de ejecuciones de la ventana principal del proyecto.	Alta	No modificable	<i>UR064</i>
<b>SR084</b>	Si seleccionas la opción de programar ejecución en un fichero ejecutable se abrirá una ventana que te permitirá introducir ciertos parámetros de ejecución y la fecha y la hora a la que deseas ejecutar el fichero.	Alta	No modificable	<i>UR065</i>

<b>SR085</b>	Si pulsas en programar ejecución en la ventana de “ejecutar fichero”, se configurará una ejecución a la hora y fecha establecida, y una vez se ejecute se cargará el resultado de la ejecución en la tabla de ejecuciones de la ventana principal del proyecto.	Alta	No modificable	<i>UR066</i>
<b>SR086</b>	Las ventanas de nuevo directorio, nuevo fichero, eliminar directorio, eliminar fichero, ejecutar fichero y programar ejecución fichero se cerrarán o bien pulsando la cruz en la esquina superior derecha, en cancelar, o en cualquier punto de la pantalla que no se encuentre dentro de la ventana.	Alta	No modificable	<i>UR067</i>
<b>SR087</b>	Una ejecución debe guardar todas sus características (hora, fecha, nombre de fichero ejecutado, usuario que lo ha ejecutado, etc..).	Alta	No modificable	<i>UR068</i>
<b>SR088</b>	Una ejecución debe guardar la traza de ejecución completa, y generar un nombre para el fichero log donde se almacenará. Este nombre será el nombre del fichero ejecutado junto con la fecha e hora de inicio de la ejecución separado por “_”.	Alta	No modificable	<i>UR068</i>
<b>SR089</b>	La tabla que muestra las ejecuciones pasadas debe ser una tabla paginada con filtro de búsqueda.	Media	No modificable	<i>UR069</i>
<b>SR090</b>	La tabla que muestra las ejecuciones pasadas en la ventana de un proyecto tendrá las siguientes columnas: Fichero, Fecha, Hora Inicio,	Alta	Modificable	<i>UR070</i>

	Hora fin, Resultado, Usuario, Parámetros de ejecución, y un botón para descargar el fichero log de la ejecución.			
<b>SR091</b>	Cada ejecución aparece arriba de todo de la tabla (se ordena utilizando la fecha y hora de ejecución descendentemente).	Media	No modificable	UR070
<b>SR092</b>	La tabla paginada no permitirá que se muestren los resultados ordenados de otra forma que no sea la predeterminada.	Media	No modificable	UR070
<b>SR093</b>	Si la ejecución ha fallado el resultado de la ejecución se mostrará en color rojo(FAIL), y si ha sido exitosa en color verde(SUCCESS).	Baja	Modificable	UR071
<b>SR094</b>	Al pulsar en el botón para descargar el log de una ejecución se mostrará el fichero en la parte inferior de la pantalla como fichero descargable.	Alta	No modificable	UR072
<b>SR095</b>	La aplicación será de fácil manejo para un usuario sin previo conocimiento de esta.	Alta	No modificable	UR074
<b>SR096</b>	La autenticación de usuarios se hace contra base de datos, aunque deberá dejarse abierta.	Media	No modificable	UR075

*Tabla 2. Requisitos de sistema: funcionales*

### **5.2.2.2 REQUISITOS NO FUNCIONALES:**

Identificador	Descripción	Prioridad	Estabilidad	Fuente
<b>SR097</b>	Mostrar mensaje de confirmación al solicitar dar de baja un usuario.	Alta	No modificable	UR026
<b>SR098</b>	Al añadir, editar o eliminar un	Alta	No	UR028



	usuario, la aplicación debe devolver a la página de gestión de usuarios mostrando la página de la tabla de usuarios en la que encontrabas antes del alta/baja/modificación del usuario.		modificable	
<b>SR099</b>	Mostrar mensaje de confirmación al solicitar eliminar un proyecto.	Alta	No modificable	<i>UR041</i>
<b>SR100</b>	Al añadir, editar o eliminar un proyecto, la aplicación debe devolver a la página de gestión de proyectos mostrando la página de la tabla de proyectos en la que encontrabas antes del alta/baja/modificación del usuario.	Alta	No modificable	<i>UR042</i>
<b>SR101</b>	Todos los ficheros se guardarán en el servidor con el nombre correspondiente al hash de su nombre real para evitar que alguien sepa que contiene cada fichero en caso de tener acceso al servidor (mejorando la seguridad).	Alta	No modificable	<i>UR073</i>
<b>SR102</b>	La aplicación será de fácil manejo para un usuario sin previo conocimiento de esta.	Alta	Modificable	<i>UR074</i>
<b>SR103</b>	La autenticación de usuarios se hace contra base de datos, aunque deberá dejarse abierta su ampliación a otros posibles sistemas de autenticación como LDAP.	Media	No modificable	<i>UR075</i>

*Tabla 3. Requisitos de sistema: no funcionales*

## **5.3 DESCRIPCIÓN DEL SISTEMA**

### **5.3.1 FUNCIONAMIENTO DE LA APLICACIÓN**

Como ya se ha explicado, el objetivo final de la aplicación es el de la gestión de procesos de Ingesta de datos en entornos de Business Intelligence. Esta gestión se realizará haciendo uso de ficheros ejecutables. Esta aplicación por lo tanto debe permitir la subida, revisión y ejecución de estos ficheros, ordenados en estructuras de directorios para cada uno de los proyectos. La ejecución de los ficheros .jar debe además mostrar si la ejecución ha sido exitosa y generar un fichero log para su posterior revisión.

La gestión de estos ficheros y los directorios que los contienen requieren a su vez una gestión de los proyectos a los que pertenecen, además de los usuarios que a ellos tienen acceso y sus permisos dentro de cada entorno.

Existen dos posibles perfiles de usuario en la aplicación. Usuarios administradores, y desarrolladores. Para entender los permisos que cada usuario tiene es importante conocer el concepto de entorno. Esta aplicación cuenta con tres: desarrollo, preproducción, y producción, y según el entorno al que se tiene acceso se tienen unos permisos u otros.

En estos momentos nuestra aplicación cuenta con tres permisos distintos: despliegue, programación, y ejecución, y según los permisos que tengas podrás realizar unas acciones u otras en cada proyecto. El permiso de despliegue te permite crear y eliminar directorios en los proyectos, así como subir ficheros de texto(.txt) y ejecutables(.jar), editarlos, o modificar sus nombres. El permiso de ejecución te permite ejecutar los ficheros ejecutables en el momento, y el de programación te da la posibilidad de programar su ejecución en un momento concreto en el futuro (o periódicamente).

Los usuarios desarrolladores con acceso al entorno de desarrollo tendrán permisos completos en cada proyecto al que estén asociados, es decir, podrán desplegar, ejecutar y programar. Los usuarios desarrolladores con acceso al entorno de preproducción y producción tienen únicamente permiso de programación. Por último, los usuarios

administradores tienen acceso a todos los entornos, y por lo tanto permisos completos en todos los proyectos. A continuación, se muestra una captura del árbol de directorios de un proyecto de ejemplo, y las acciones que un usuario con permisos completos podría realizar



Figura 9. Vistazo Árbol directorios proyecto

al hacer click derecho en cualquier directorio o fichero.

Al pinchar sobre el directorio raíz solo puedes añadir un nuevo directorio. Este directorio raíz no puede eliminarse ni cambiarse su nombre, ya que se crea por defecto al crear el proyecto y lleva el mismo nombre que él.

Al pinchar sobre un directorio que no sea el directorio raíz las opciones aumenta. Podemos o bien crear un nuevo directorio dentro de este, un nuevo fichero de texto, importar un fichero (.jar o .txt) de tu ordenador, cambiar el nombre del directorio o eliminarlo.

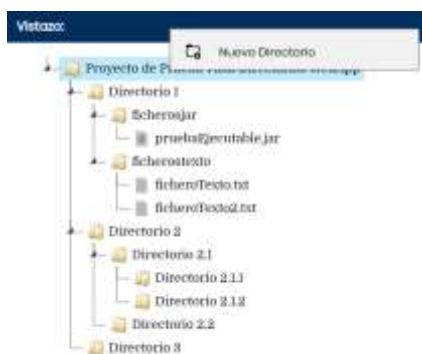


Figura 11. Menú desplegable directorio raíz

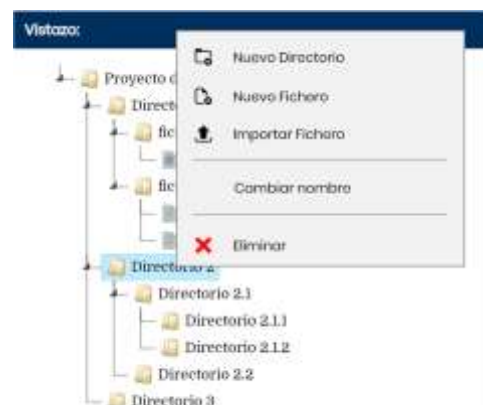
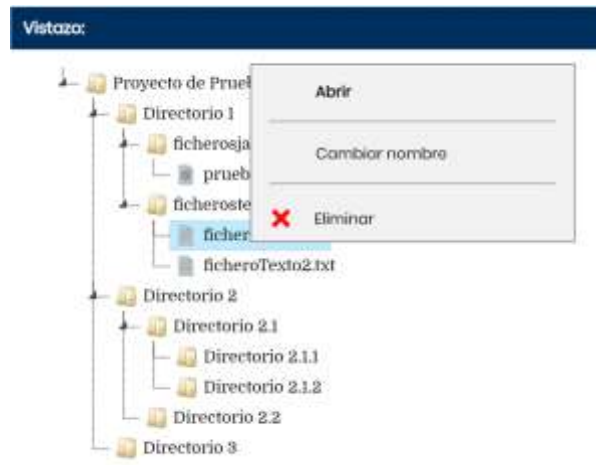


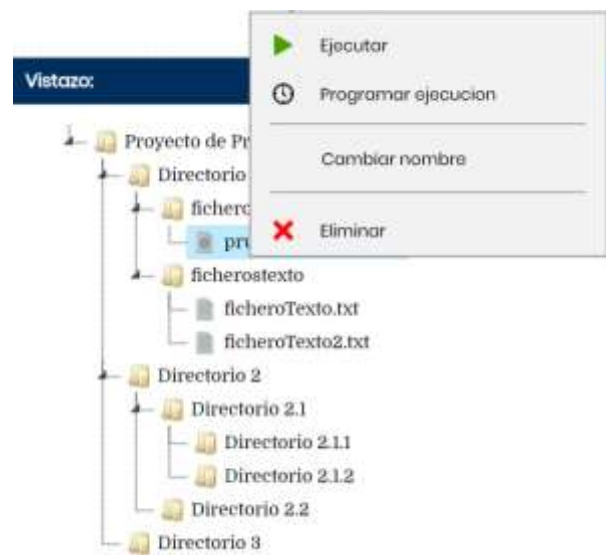
Figura 10. Menú desplegable directorio

Al pinchar sobre un fichero de texto podrás abrirlo (y modificarlo), cambiar su nombre, o eliminarlo.



*Figura 12. Menú desplegable fichero de texto*

Por último, al pinchar sobre un fichero ejecutable, podrá ejecutarse en el momento (introduciendo más tarde los parámetros de ejecución correspondientes), programar su ejecución para que se realice un día a una hora determinada, cambiar su nombre o eliminarlo:



*Figura 13. Menú desplegable fichero ejecutable*

Además de la gestión de ficheros y directorios dentro de los proyectos, se requiere de una gestión de estos proyectos que incluya la eliminación o creación de nuevos proyectos, así como la modificación de su nombre o el entorno al que pertenecen, y el equipo de usuarios que en él trabaja. Para esto existe una pestaña de gestión de proyectos a la que únicamente los perfiles administradores tienen acceso.

Además de los proyectos, se requiere también una gestión de los usuarios, y para ello existe otra pestaña a la que de nuevo sólo los administradores tienen acceso, y que permite crear nuevos usuarios y dar de baja o editar existentes, modificando su nombre o dirección de correo electrónico, su tipo de perfil o los entornos a los que tienen acceso, así como sus proyectos asociados. La última pestaña creada fue la editar perfil, a la que todos los usuarios con independencia de su perfil tienen acceso, y que permite modificar tu contraseña. El manual de uso de la aplicación del Anexo II describe con capturas y explicaciones más detalladas el uso y las acciones que se pueden realizar en cada una de las ventanas.

El número de perfiles existentes, así como los permisos que tienen asociados pueden modificarse únicamente modificando las bases de datos directamente. Una de las mejoras que podría realizarse y de la que hablaré más tarde en el apartado de “conclusiones y trabajos futuros” sería la creación de una pestaña a la que tendrían acceso los usuarios administradores y que permitiese crear nuevos entornos y permisos, asociándolos para crear nuevos perfiles de usuario.

### 5.3.2 DIAGRAMA DE CASOS DE USO

Los casos de uso describen la funcionalidad del sistema basándose en los requisitos previamente establecidos. Se representan los actores y los casos de uso principales del sistema, detallando las relaciones que entre ellos existen.

A continuación, se muestra el diagrama de casos de uso del sistema:

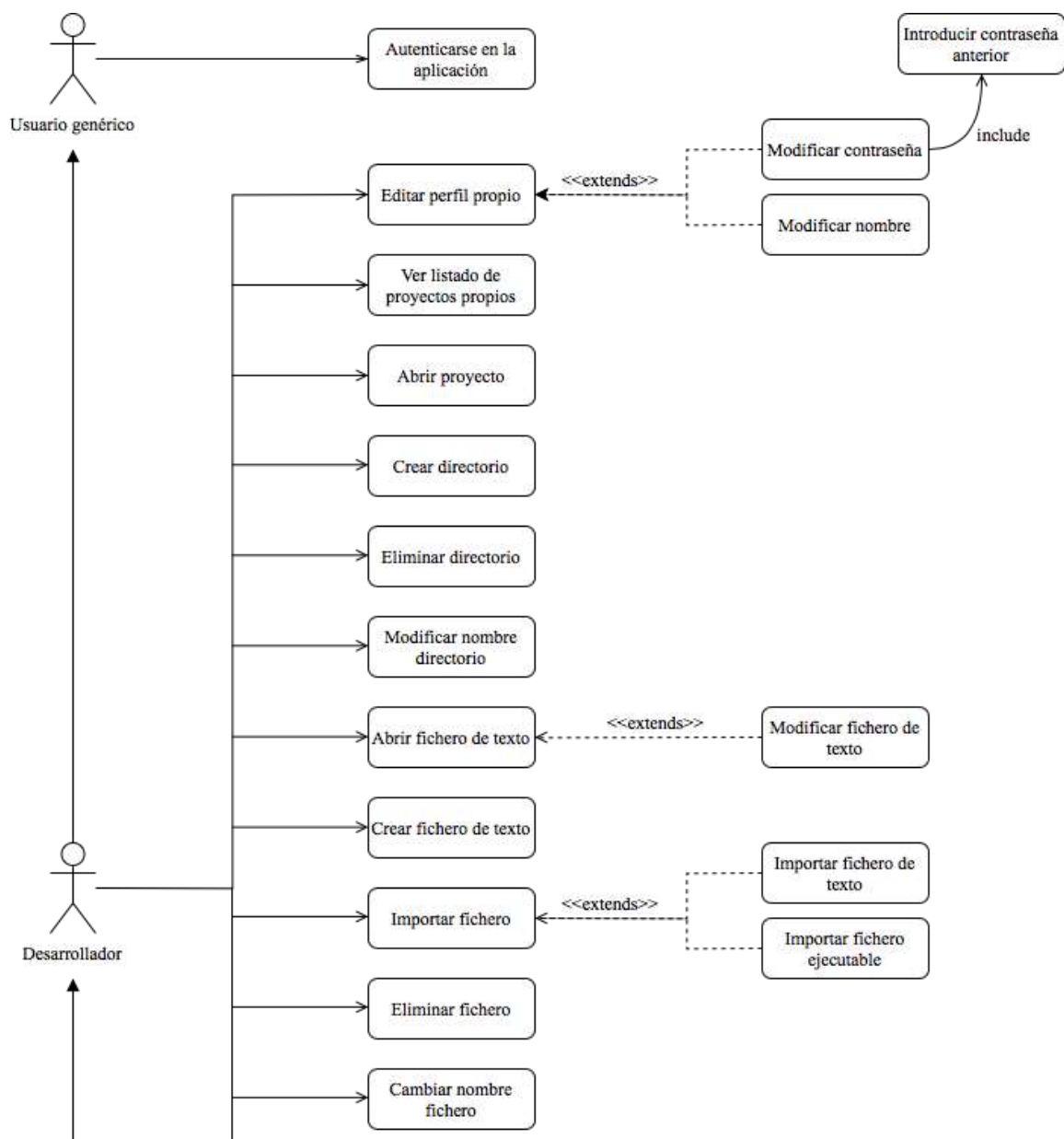


Figura 14. Diagrama casos de uso WebApp[1]

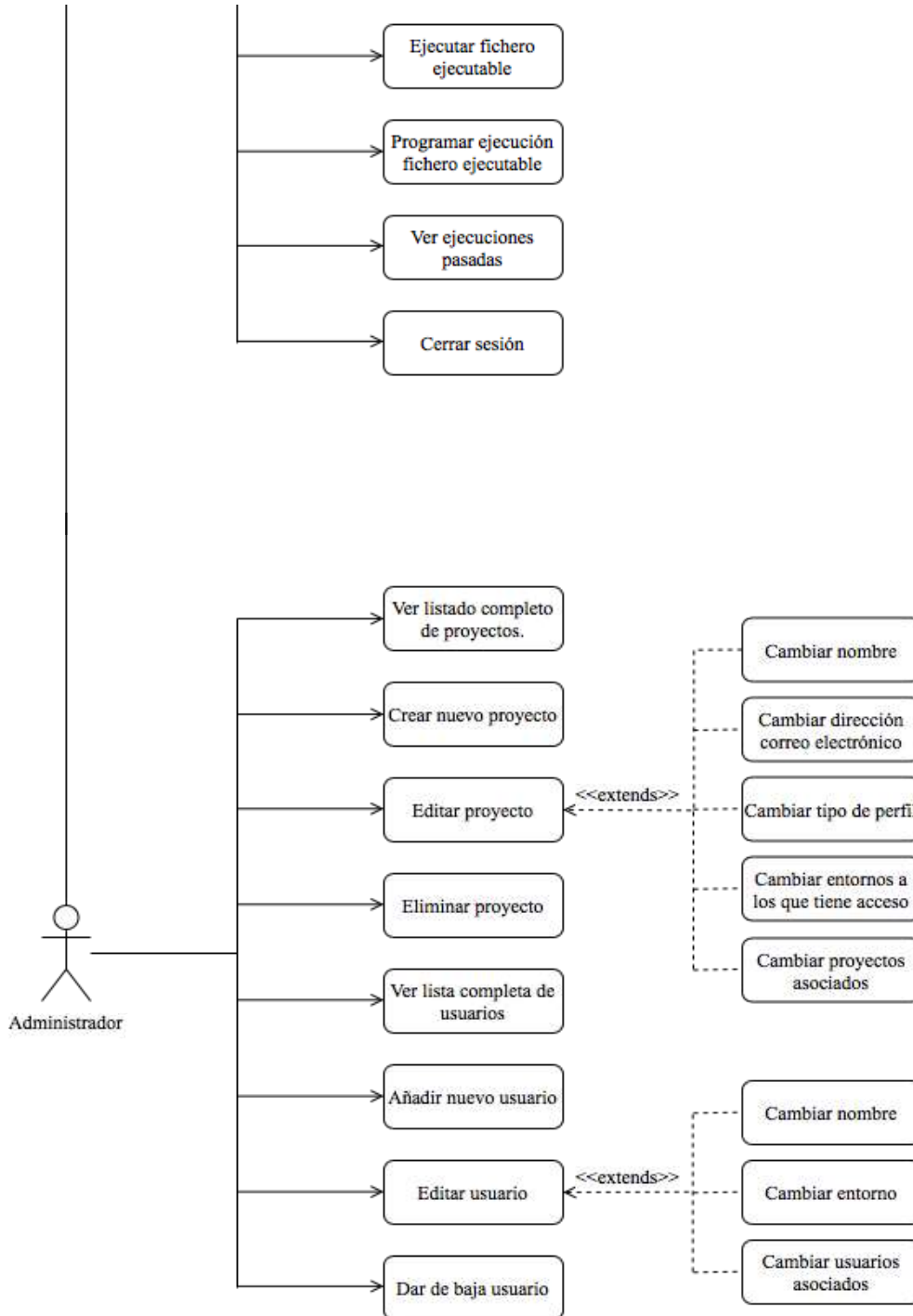


Figura 15. Digrama casos de uso WebApp[2]

### 5.3.3 DESCRIPCIÓN DE CASOS DE USO

Este apartado facilita una descripción detallada de los diferentes casos de uso de la aplicación.

UC001 AUTENTICACIÓN	
<b>Caso de uso</b>	Autenticación.
<b>Actores</b>	Usuario genérico.
<b>Objetivo</b>	Controlar el acceso al sistema, permitiendo que accedan únicamente usuarios registrados y diferenciando las ventanas a las que tendrás acceso según el tipo de perfil que tengas.
<b>Descripción</b>	Autenticación en la aplicación.
<b>Precondiciones</b>	El usuario debe estar en la página de autenticación de la aplicación.
<b>Postcondiciones</b>	Permitir o denegar el acceso a la aplicación.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario introduce su email y contraseña.</li> <li>2. El sistema valida contra base de datos si ese email y contraseña introducidos existen y son correctos.</li> <li>3. El sistema permite el acceso a la aplicación, comprueba el tipo de perfil del usuario e inicializa la página de inicio teniendo en cuenta los permisos que tiene.</li> <li>4. Página de inicio con mensaje de bienvenida, información sobre el tipo de perfil y entornos asociados y lista de proyectos asociados. Pestañas para editar tu perfil, cerrar sesión, gestión de usuarios, y gestión de proyectos.</li> </ol>
<b>Flujo alternativo</b>	<p>(2.1) Si el correo introducido no existe en base de datos la aplicación niega el acceso y lanza el mensaje “El usuario introducido no existe”.</p> <p>(2.2) Si el correo introducido existe en base de datos, pero la contraseña es errónea la aplicación niega el acceso y lanza el mensaje “Contraseña incorrecta”.</p> <p>(2.3) Si el correo introducido existe en base de datos, pero se ha dado de baja la aplicación niega el acceso y lanza el mensaje “El usuario introducido ha sido dado de baja”.</p> <p>(4.1) Si el usuario es desarrollador no mostrará las pestañas de gestión de usuarios ni proyectos.</p>

*Tabla 4. Descripción de caso de uso: Autenticación*



UC002 EDITAR PERFIL PROPIO	
<b>Caso de uso</b>	Editar perfil propio.
<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Permitir la modificación de tu nombre y de tu contraseña.
<b>Descripción</b>	Ver tus datos y modificarlos.
<b>Precondiciones</b>	El usuario debe haberse autenticado en la aplicación (UC001).
<b>Postcondiciones</b>	Nombre o contraseña modificada.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario pincha en la pestaña de editar perfil.</li> <li>2. El sistema muestra todos sus datos: nombre completo, email, tipo de perfil y entornos a los que tiene acceso.</li> <li>3. El sistema permite la modificación del nombre y la contraseña.               <ol style="list-style-type: none"> <li>3.1. Para modificar la contraseña, el usuario debe introducir la contraseña anterior.</li> <li>3.2. Al introducir la contraseña anterior el usuario introduce su nueva contraseña en dos campos de texto distintos.</li> </ol> </li> <li>4. El usuario pincha en guardar cambios.</li> </ol>
<b>Flujo alternativo</b>	<p>(0.1) Si el usuario pincha en cancelar se reinicia el formulario con los datos que consten en base de datos.</p> <p>(3.1.1) Si no se introduce nada en el campo de contraseña anterior el cuadro de texto para introducir la nueva contraseña estará bloqueado.</p> <p>(4.1) Si se intenta guardar los cambios y el nombre de usuario está vacío la aplicación no guardará los cambios y avisará con el mensaje de “El campo de nombre de usuario debe estar completo”.</p> <p>(4.2) Si se intenta guardar los cambios y la contraseña anterior es incorrecta no guardará los cambios y avisará con el mensaje de “Contraseña incorrecta”.</p> <p>(4.3) Si al introducir la contraseña nueva en los dos campos de texto esta no coincide la aplicación avisará con el mensaje de “Debes introducir la misma contraseña”</p>

*Tabla 5. Descripción de caso de uso: editar perfil propio*

<b>UC003 VER LISTADO DE PROYECTOS PROPIOS</b>	
<b>Caso de uso</b>	Ver listado de proyectos propios.
<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Mostrar una tabla con todos los proyectos que tienes asignados. En caso de ser administrador mostrará todos los proyectos que existen. Al pulsar en el proyecto abrirá una nueva ventana con el proyecto en cuestión, sus participantes, sus directorios y ficheros, ejecuciones realizadas, etc.
<b>Descripción</b>	Ver un listado de proyectos asociados facilitando el acceso a ellos.
<b>Precondiciones</b>	El usuario debe haberse autenticado en la aplicación (UC001).
<b>Postcondiciones</b>	Al pinchar en un proyecto determinado te lo abrirá en una nueva página.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario entra la aplicación.</li> <li>2. El sistema muestra en la página de inicio una tabla paginada con todos los proyectos asociados al usuario autenticado. Los proyectos aparecerán ordenados por alfabéticamente.</li> </ol>
<b>Flujo alternativo</b>	(2.1) Si el usuario pincha en un proyecto en particular se abrirá el proyecto en una ventana nueva.

*Tabla 6. Descripción de caso de uso: ver listado de proyectos propios*

<b>UC004 ABRIR PROYECTO</b>	
<b>Caso de uso</b>	Abrir proyecto.
<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Abrir un proyecto para ver toda la información que contiene e interactuar con él.
<b>Descripción</b>	Abrir proyecto existente y asociado.
<b>Precondiciones</b>	El usuario debe haberse autenticado en la aplicación (UC001) y encontrarse en la página de Inicio.

	El usuario debe tener acceso al proyecto (sólo se mostrará en la lista paginada si así es).
<b>Postcondiciones</b>	Ninguna.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario pincha en el proyecto que corresponda en la lista paginada de la pestaña de Inicio.</li> <li>2. El sistema comprueba los permisos del usuario para filtrar las acciones que puede realizar y la información que puede ver.</li> <li>3. El sistema muestra tres bloques de información: los participantes del proyecto, el árbol de directorios que contiene, y la lista de ejecuciones pasadas.</li> </ol>
<b>Flujo alternativo</b>	<p>(0.1) Si el usuario pincha en cualquier otra pestaña en el menú de navegación se cerrará la ventana del proyecto y se abrirá la que corresponda.</p> <p>(3.1) Si no tienes permiso de ejecución o programación no mostrará el listado de ejecuciones pasadas.</p>

*Tabla 7. Descripción de caso de uso: abrir proyecto*

<b>UC005 CREAR DIRECTORIO</b>	
<b>Caso de uso</b>	Crear directorio.
<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Permitir la creación de un directorio dentro de un proyecto para facilitar su organización.
<b>Descripción</b>	Crear directorio dentro de un proyecto.
<b>Precondiciones</b>	El usuario debe haberse autenticado en la aplicación (UC001). El usuario debe encontrarse en la página del proyecto (UC004). El usuario debe tener permiso de despliegue.
<b>Postcondiciones</b>	Directorio creado y guardado en servidor y base de datos.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click derecho en el directorio padre donde quiere crear un nuevo directorio y selecciona la opción de “Nuevo directorio” del menú que aparece.</li> <li>2. Se abre una ventana para introducir el nombre del directorio</li> </ol>

	<p>a crear.</p> <ol style="list-style-type: none"> <li>El usuario escribe el nombre y pulsa “Crear”.</li> <li>El sistema comprueba que el nombre del directorio a crear no exista ya dentro del directorio padre. Si es así añadirá un número (X) al nombre del directorio. Ese número será X=2, pero si esta versión también existe será X=3, 4, 5...</li> <li>El sistema guarda el directorio y su información en base de datos y en el servidor.</li> </ol>
<b>Flujo alternativo</b>	<p>(2.1) Si el usuario pincha en cancelar se cierra la ventana.</p> <p>(2.2) Si el usuario pincha en cualquier punto de la pantalla fuera de la ventana se cierra.</p> <p>(2.3) Si el usuario pincha en la cruz superior derecha se cierra la ventana.</p> <p>(3.1) Si el campo para introducir el nombre está vacío la aplicación avisa con el mensaje: “Debes introducir un nombre”.</p>

*Tabla 8. Descripción de caso de uso: crear directorio*

<b>UC006 ELIMINAR DIRECTORIO</b>	
<b>Caso de uso</b>	Eliminar directorio.
<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Permitir la eliminación de un directorio dentro de un proyecto.
<b>Descripción</b>	Eliminar directorio dentro de un proyecto.
<b>Precondiciones</b>	El usuario debe haberse autenticado en la aplicación (UC001). El usuario debe encontrarse en la página del proyecto (UC004). El usuario debe tener permiso de despliegue.
<b>Postcondiciones</b>	Directorio seleccionado y todos sus directorios y ficheros interiores eliminados tanto de servidor como de base de datos.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>El usuario hace click derecho en el directorio que se desea eliminar y selecciona la opción de “Eliminar” del menú que aparece.</li> </ol>

	<ol style="list-style-type: none"> <li>2. Se abre una ventana para confirmar la eliminación del directorio.</li> <li>3. El usuario pulsa “Eliminar”.</li> <li>4. El sistema elimina el directorio y todos sus hijos (tanto directorios como ficheros) del servidor y de base de datos.</li> </ol>
<b>Flujo alternativo</b>	<p>(2.1) Si el usuario pincha en cancelar se cierra la ventana.</p> <p>(2.2) Si el usuario pincha en cualquier punto de la pantalla fuera de la ventana se cierra.</p> <p>(2.3) Si el usuario pincha en la cruz superior derecha se cierra la ventana.</p>

*Tabla 9. Descripción de caso de uso: eliminar directorio*

<b>UC007 MODIFICAR NOMBRE DIRECTORIO</b>	
<b>Caso de uso</b>	Modificar nombre directorio.
<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Permitir la modificación del nombre de un directorio de un proyecto.
<b>Descripción</b>	Modificar nombre directorio.
<b>Precondiciones</b>	El usuario debe haberse autenticado en la aplicación (UC001). El usuario debe encontrarse en la página del proyecto (UC004). El usuario debe tener permiso de despliegue.
<b>Postcondiciones</b>	Nombre directorio modificado en base de datos y en servidor.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click derecho en el directorio al que quiere cambiarle el nombre y selecciona la opción de “Cambiar nombre” del menú que aparece.</li> <li>2. El nombre del directorio en el árbol de directorios se hará editable.</li> <li>3. El usuario escribe el nuevo nombre y pulsa la tecla Enter o hace click en cualquier punto de la pantalla para guardarlo.</li> <li>4. El sistema comprueba que el nuevo nombre del directorio no exista ya dentro del directorio padre. Si es así añadirá un número (X) al nombre del directorio. Ese</li> </ol>

	<p>número será X=2, pero si esta versión también existe será X=3, 4, 5...</p> <p>5. El sistema modifica el nombre del directorio en base de datos y en el servidor.</p>
<b>Flujo alternativo</b>	<p>(3.1) Si no se introduce ningún nombre el sistema no guardará ningún cambio y el directorio mantendrá su nombre anterior.</p>

Tabla 10. Descripción de caso de uso: modificar nombre directorio

<b>UC008 CREAR FICHERO DE TEXTO</b>	
<b>Caso de uso</b>	Crear fichero de texto.
<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Permitir la creación de un fichero de texto dentro del directorio de un proyecto y editar su contenido.
<b>Descripción</b>	Crear fichero de texto.
<b>Precondiciones</b>	El usuario debe haberse autenticado en la aplicación (UC001). El usuario debe encontrarse en la página del proyecto (UC004). El usuario debe tener permiso de despliegue.
<b>Postcondiciones</b>	Fichero de texto creado.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click derecho en el directorio en el que quiere crear el fichero de texto y selecciona la opción de “Nuevo fichero” del menú que aparece.</li> <li>2. El nombre del directorio en el árbol de directorios se hará editable.</li> <li>3. Se abre una ventana para introducir el nombre del fichero a crear.</li> <li>4. El sistema comprueba que no exista otro fichero de texto con el mismo nombre en el directorio padre. Si es así añadirá un número (X) al nombre del fichero. Ese número será X=2, pero si esta versión también existe será X=3, 4, 5...</li> <li>5. El sistema generará un hash usando el nombre del fichero para guardarlo con ese nombre en el servidor. Este valor se guardará junto con el resto de información del fichero en base de datos.</li> </ol>

	<p>6. El fichero entonces se abre vacío en el editor de texto de la aplicación para poder escribir en él.</p> <p>7. El usuario rellena el fichero en caso de que quiera hacerlo y pulsa cerrar o guardar para cerrar el fichero (sin guardar) o guardar su contenido.</p> <p>8. El sistema guarda el contenido del fichero en el servidor.</p>
<b>Flujo alternativo</b>	<p>(2.1) Si el usuario pincha en cancelar se cierra la ventana.</p> <p>(2.2) Si el usuario pincha en cualquier punto de la pantalla fuera de la ventana se cierra.</p> <p>(2.3) Si el usuario pincha en la cruz superior derecha se cierra la ventana.</p> <p>(3.1) Si el campo para introducir el nombre está vacío la aplicación avisa con el mensaje: “Debes introducir un nombre”.</p>

*Tabla 11. Descripción de caso de uso: crear fichero de texto*

<b>UC009 ABRIR FICHERO DE TEXTO</b>	
<b>Caso de uso</b>	Abrir fichero de texto.
<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Permitir abrir un fichero de texto de un proyecto y modificar su contenido.
<b>Descripción</b>	Abrir fichero de texto y modificarlo.
<b>Precondiciones</b>	El usuario debe haberse autenticado en la aplicación (UC001). El usuario debe encontrarse en la página del proyecto (UC004). El usuario debe tener permiso de despliegue.
<b>Postcondiciones</b>	Ninguna.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click derecho en el fichero de texto que desea abrir y selecciona la opción de “Abrir” del menú que aparece.</li> <li>2. El sistema recoge la información del fichero de base de datos y su contenido del servidor y lo muestra en el editor de texto de la aplicación.</li> </ol>
<b>Flujo alternativo</b>	Ninguno.

*Tabla 12. Descripción de caso de uso: abrir fichero de texto*

<b>UC010 MODIFICAR FICHERO DE TEXTO</b>	
<b>Caso de uso</b>	Modificar fichero de texto.
<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Permitir la modificación del contenido de un fichero de texto de un proyecto.
<b>Descripción</b>	Modificar contenido de fichero de texto.
<b>Precondiciones</b>	El usuario debe haberse autenticado en la aplicación (UC001). El usuario debe encontrarse en la página del proyecto (UC004). El usuario debe tener permiso de despliegue. El usuario debe haber abierto el fichero de texto (UC009).
<b>Postcondiciones</b>	Contenido de fichero modificado.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>Habiendo abierto el fichero de texto el contenido del fichero se encontrará en un editor de texto en pantalla.</li> <li>El usuario cambia el contenido del fichero y pulsa en guardar.</li> <li>El sistema guarda el contenido del fichero en el servidor.</li> </ol>
<b>Flujo alternativo</b>	(0.1) Si el usuario pincha en la cruz superior derecha del panel del editor de texto se cierra el editor sin guardar el contenido del fichero.

*Tabla 13. Descripción de caso de uso: modificar fichero de texto*

<b>UC011 IMPORTAR FICHERO</b>	
<b>Caso de uso</b>	Importar fichero.
<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Permitir subir un fichero ejecutable o de texto existente en tu ordenador a un proyecto.
<b>Descripción</b>	Importar fichero de texto o ejecutable.
<b>Precondiciones</b>	El usuario debe haberse autenticado en la aplicación (UC001). El usuario debe encontrarse en la página del proyecto (UC004). El usuario debe tener permiso de despliegue.



	El fichero debe existir en el ordenador.
<b>Postcondiciones</b>	Fichero subido y guardado en servidor y base de datos.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click derecho en el directorio en el que quiere subir el fichero y selecciona la opción de “Importar fichero” del menú que aparece.</li> <li>2. Se abrirá una ventana para seleccionar el fichero de tu ordenador.</li> <li>3. El usuario selecciona el fichero a subir (.txt) o (.jar) y pulsa aceptar.</li> <li>4. El sistema comprueba que es un fichero de texto o ejecutable.</li> <li>5. El sistema comprueba que el nombre del fichero no exista ya dentro del directorio padre. Si es así añadirá un número (X) al nombre del fichero. Ese número será X=2, pero si esta versión también existe será X=3, 4, 5...</li> <li>6. El sistema generará un hash usando el nombre del fichero para guardarlo con ese nombre en el servidor. Este valor se guardará junto con el resto de información del fichero en base de datos.</li> <li>7. Si el fichero es de texto entonces se abrirá en el editor de texto de la aplicación para poder visualizar su contenido y modificarlo si se desea.</li> </ol>
<b>Flujo alternativo</b>	(4.1) Si el fichero no es .txt o .jar no se subirá ni se guardará.

Tabla 14. Descripción de caso de uso: importar fichero

UC012 ELIMINAR FICHERO	
<b>Caso de uso</b>	Eliminar fichero.
<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Permitir eliminar un fichero ejecutable o de texto de un proyecto.
<b>Descripción</b>	Eliminar fichero de texto o ejecutable.
<b>Precondiciones</b>	<p>El usuario debe haberse autenticado en la aplicación (UC001).</p> <p>El usuario debe encontrarse en la página del proyecto (UC004).</p> <p>El usuario debe tener permiso de despliegue.</p>

<b>Postcondiciones</b>	Fichero eliminado del servidor y de base de datos.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click derecho en el fichero que se desea eliminar y selecciona la opción de “Eliminar” del menú que aparece.</li> <li>2. Se abre una ventana para confirmar la eliminación del fichero.</li> <li>3. El usuario pulsa “Eliminar”.</li> <li>4. El sistema elimina el fichero del servidor y de base de datos.</li> </ol>
<b>Flujo alternativo</b>	<p>(2.1) Si el usuario pincha en cancelar se cierra la ventana.</p> <p>(2.2) Si el usuario pincha en cualquier punto de la pantalla fuera de la ventana se cierra.</p> <p>(2.3) Si el usuario pincha en la cruz superior derecha se cierra la ventana.</p>

*Tabla 15. Descripción de caso de uso: eliminar fichero*

<b>UC013 MODIFICAR NOMBRE FICHERO</b>	
<b>Caso de uso</b>	Modificar nombre fichero.
<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Permitir la modificación del nombre de un fichero de un proyecto.
<b>Descripción</b>	Modificar nombre fichero.
<b>Precondiciones</b>	<p>El usuario debe haberse autenticado en la aplicación (UC001).</p> <p>El usuario debe encontrarse en la página del proyecto (UC004).</p> <p>El usuario debe tener permiso de despliegue.</p>
<b>Postcondiciones</b>	Nombre directorio modificado en base de datos y en servidor.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click derecho en el fichero al que quiere cambiarle el nombre y selecciona la opción de “Cambiar nombre” del menú que aparece.</li> <li>2. El nombre del fichero en el árbol de directorios se hará editable.</li> </ol>

	<ol style="list-style-type: none"> <li>3. El usuario escribe el nuevo nombre y pulsa la tecla Enter o hace click en cualquier punto de la pantalla para guardarlo.</li> <li>4. El sistema comprueba que el nuevo nombre del fichero no exista ya dentro del directorio padre. Si es así añadirá un número (X) al nombre del fichero. Ese número será X=2, pero si esta versión también existe será X=3, 4, 5...</li> <li>5. El sistema generará un hash usando el nombre del fichero para guardarlo con ese nombre en el servidor.</li> <li>6. El sistema modifica el nombre del fichero y su hash en base de datos y en el servidor.</li> </ol>
<b>Flujo alternativo</b>	(3.1) Si no se introduce ningún nombre el sistema no guardará ningún cambio y el fichero mantendrá su nombre anterior.

Tabla 16. Descripción de caso de uso: modificar nombre fichero

UC014 EJECUTAR FICHERO	
<b>Caso de uso</b>	Ejecutar fichero.
<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Permitir la ejecución instantánea de un fichero .jar dentro de un proyecto.
<b>Descripción</b>	Ejecuta un fichero.
<b>Precondiciones</b>	El usuario debe haberse autenticado en la aplicación (UC001). El usuario debe encontrarse en la página del proyecto (UC004). El fichero tiene que ser ejecutable. El usuario debe tener permiso de ejecución.
<b>Postcondiciones</b>	Fichero ejecutado y ejecución guardada en base de datos. Se genera un fichero .log con toda la información relevante sobre la ejecución.

<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click derecho en el fichero que desea ejecutar y selecciona la opción de “Ejecutar” del menú que aparece.</li> <li>2. Se abre una ventana para introducir los parámetros de ejecución. Estos parámetros son únicamente dos: grupo inicial y máximo de asignación de memoria.</li> <li>3. El sistema ejecuta el fichero en el momento utilizando los parámetros introducidos y guarda en base de datos toda la información de la ejecución: el nombre del fichero ejecutado, la hora de inicio y de fin de la ejecución (y fecha), el resultado, el usuario que le ha ejecutado, los parámetros introducidos, y la traza de ejecución completa, así como el nombre del fichero log en el que se guardará la traza.</li> <li>4. El sistema recarga la lista de ejecuciones mostrando la nueva ejecución en la tabla de ejecuciones (siempre aparece la última ejecución de primera).</li> </ol>
<b>Flujo alternativo</b>	<p>(2.1) Si el usuario pincha en cancelar se cierra la ventana.</p> <p>(2.2) Si el usuario pincha en cualquier punto de la pantalla fuera de la ventana se cierra.</p> <p>(2.3) Si el usuario pincha en la cruz superior derecha se cierra la ventana.</p>

*Tabla 17. Descripción de caso de uso: ejecutar fichero*

<b>UC015 PROGRAMAR EJECUCION FICHERO</b>	
<b>Caso de uso</b>	Programar ejecución fichero.
<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Programar la ejecución de un fichero .jar dentro de un proyecto para que se haga periódicamente o a una hora concreta.
<b>Descripción</b>	Programa la ejecución de un fichero.
<b>Precondiciones</b>	<p>El usuario debe haberse autenticado en la aplicación (UC001).</p> <p>El usuario debe encontrarse en la página del proyecto (UC004).</p> <p>El fichero tiene que ser ejecutable.</p> <p>El usuario debe tener permiso de programación.</p>
<b>Postcondiciones</b>	Fichero ejecutado y ejecución guardada en base de datos. Se genera un fichero .log con toda la información relevante sobre la

<p><b>Flujo normal</b></p>	<p>ejecución.</p> <ol style="list-style-type: none"> <li>1. El usuario hace click derecho en el fichero al que desea programar su ejecución y selecciona la opción de “Programar ejecución” del menú que aparece.</li> <li>2. Se abre una ventana para introducir los parámetros de ejecución, la hora, y la periodicidad.</li> <li>3. El usuario introduce los parámetros, la hora, y cuándo quiere que se repita la ejecución (si es que quiere que se repita), y pulsa en “Programar Ejecución”.</li> <li>4. El sistema calcula la primera fecha y hora de ejecución del fichero así como su periodo de ejecución y genera la tarea. <ul style="list-style-type: none"> <li>- Si se ha seleccionado la opción de no repetir la ejecución nunca la ejecución se realizará “hoy” a la hora seleccionada. Si la hora introducida es anterior a la hora actual, la ejecución se programará para el día siguiente (a la hora seleccionada).</li> <li>- Si se ha seleccionado la opción de todos los días y la hora introducida es anterior a la hora actual, la primera ejecución será al día siguiente (a la hora seleccionada).</li> <li>- Si se ha seleccionado la opción de repetir un día específico de la semana la primera ejecución se programará primer día que sea ese día específico de la semana a la hora seleccionada.</li> <li>- Si el día de la semana seleccionado coincide con el día en el que te encuentras se ejecutará ese mismo día, pero si la hora introducida es anterior a la hora actual la primera ejecución se realizará la semana siguiente.</li> </ul> </li> <li>5. Al llegar a la fecha y hora establecida se realizará la ejecución que corresponda.</li> </ol>
<p><b>Flujo alternativo</b></p>	<ol style="list-style-type: none"> <li>(2.1) Si el usuario pincha en cancelar se cierra la ventana.</li> <li>(2.2) Si el usuario pincha en cualquier punto de la pantalla fuera de la ventana se cierra.</li> <li>(2.3) Si el usuario pincha en la cruz superior derecha se cierra la ventana.</li> </ol>

	(3.1) Si el usuario no introduce una hora de ejecución el sistema avisa con el mensaje “Debes introducir una hora”.
--	---

*Tabla 18. Descripción de caso de uso: programar ejecución fichero*

UC016 DESCARGAR FICHERO LOG	
<b>Caso de uso</b>	Descargar fichero .log
<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Descargar la traza de ejecución de cualquier ejecución realizada.
<b>Descripción</b>	Permite la descarga de la traza de ejecución de un fichero .log en tu ordenador pulsando en el botón de descargar fichero de la tabla ejecuciones.
<b>Precondiciones</b>	El usuario debe haberse autenticado en la aplicación (UC001). El usuario debe encontrarse en la página del proyecto (UC004). Debe haberse ejecutado un fichero (UC014 o UC015).
<b>Postcondiciones</b>	Guardado en tu ordenador un fichero .log con contenido sobre la traza de la ejecución.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click en el botón de la derecha de la fila de la ejecución con el icono de fichero log.</li> <li>2. El sistema genera un fichero log con el nombre del fichero log y su contenido obtenidos de base de datos.</li> <li>3. El sistema muestra en la misma página un “pop-up” con la opción de descargar fichero.</li> <li>4. El usuario pincha y se abre en su ordenador fichero log con su contenido.</li> </ol>
<b>Flujo alternativo</b>	(4.1) El usuario guarda el fichero en el sistema o lo cierra.

*Tabla 19. Descripción de caso de uso: descargar fichero log*

UC017 VER EJECUCIONES ANTERIORES	
<b>Caso de uso</b>	Ver ejecuciones anteriores.

<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Permitir ver y revisar toda la información referente a una ejecución pasada.
<b>Descripción</b>	Ver trazas de ejecución de un fichero ejecutado.
<b>Precondiciones</b>	El usuario debe haberse autenticado en la aplicación (UC001). El usuario debe encontrarse en la página del proyecto (UC004). El fichero tiene que ser ejecutable. El usuario debe tener permiso de programación o ejecución.
<b>Postcondiciones</b>	Ninguna.
<b>Flujo normal</b>	1. Encontrándose en la página de un proyecto y con los permisos adecuados aparecerá una tabla con todas las ejecuciones anteriores.
<b>Flujo alternativo</b>	Ninguno.

*Tabla 20. Descripción de caso de uso: ver ejecuciones anteriores*

<b>UC018 CERRAR SESIÓN</b>	
<b>Caso de uso</b>	Cerrar sesión.
<b>Actores</b>	Usuario desarrollador, usuario administrador.
<b>Objetivo</b>	Cerrar la sesión en la aplicación.
<b>Descripción</b>	Cierra sesión.
<b>Precondiciones</b>	El usuario debe haberse autenticado en la aplicación (UC001).
<b>Postcondiciones</b>	Sesión cerrada.
<b>Flujo normal</b>	1. El usuario pincha en la pestaña de Cerrar Sesión. 2. El sistema elimina la sesión del usuario y devuelve a la página de autenticación.
<b>Flujo alternativo</b>	Ninguno.

*Tabla 21. Descripción de caso de uso: cerrar sesión*

<b>UC019 VER LISTA USUARIOS</b>	
<b>Caso de uso</b>	Ver lista de usuarios.
<b>Actores</b>	Usuario administrador.
<b>Objetivo</b>	Permitir ver la lista completa de usuarios de la aplicación con su información principal.
<b>Descripción</b>	Ver lista de usuarios de la aplicación.
<b>Precondiciones</b>	El usuario administrador debe haberse autenticado en la aplicación (UC001).
<b>Postcondiciones</b>	Ninguna.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario pincha en la pestaña de Gestión de Usuarios.</li> <li>2. El sistema muestra todos los usuarios de la aplicación en una tabla paginada con las siguientes columnas: nombre, email, perfil, entornos, proyectos, y fecha de creación.</li> </ol>
<b>Flujo alternativo</b>	(0.1) Si la sesión del usuario ha expirado la aplicación volverá a la página de inicio.

Tabla 22. Descripción de caso de uso: ver lista de usuarios

<b>UC020 CREAR NUEVO USUARIO</b>	
<b>Caso de uso</b>	Crear nuevo usuario.
<b>Actores</b>	Usuario administrador.
<b>Objetivo</b>	Permitir la creación de un nuevo usuario.
<b>Descripción</b>	Creación de nuevo usuario.
<b>Precondiciones</b>	<p>El usuario administrador debe haberse autenticado en la aplicación (UC001).</p> <p>El usuario administrador debe encontrarse en la pestaña de Gestión de Usuarios.</p>
<b>Postcondiciones</b>	Usuario creado y guardado en base de datos y página de gestión de usuarios recargada.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario pincha en el botón superior izquierdo de la tabla paginada.</li> <li>2. Se abre una ventana de nuevo usuario con cuatro datos a</li> </ol>



	<p>introducir: tres cuadros de texto para el nombre, los apellidos y el email y un botón radio para seleccionar el perfil (administrador o desarrollador).</p> <ol style="list-style-type: none"> <li>3. Si el perfil seleccionado es de administrador el usuario podrá pulsar el botón de añadir usuario y al nuevo usuario se le habrán asignado todos los entornos, permisos, y proyectos disponibles.</li> <li>4. Si el perfil seleccionado es de desarrollador aparecerán otros dos campos más a completar: entornos y proyectos. Los entornos aparecerán en una lista de checkboxes, y los proyectos en una lista con buscador que en un principio estará vacía.</li> <li>5. Al seleccionar un entorno se mostrarán en la lista de proyectos los proyectos existentes en dichos entornos, y al deseccionarlos desaparecen.</li> <li>6. El usuario selecciona los entornos y proyectos a los que quiere que tenga acceso el nuevo usuario.</li> <li>7. Pulsa añadir usuario y la aplicación guarda el usuario en base de datos.</li> <li>8. El sistema genera una contraseña aleatoria de 7 dígitos que aparece en un mensaje de alerta en pantalla y recarga la tabla paginada.</li> </ol>
<p><b>Flujo alternativo</b></p>	<ol style="list-style-type: none"> <li>(2.1) Si el usuario pincha en cualquier punto de la pantalla fuera de la ventana se cierra.</li> <li>(2.2) Si el usuario pincha en la cruz superior derecha se cierra la ventana.</li> <li>(3.1) o (7.1) Si no se introduce nada en el campo de nombre, apellidos o email salta el mensaje “Campo Obligatorio”.</li> <li>(3.2) o (7.2) Si no se selecciona ningún perfil salta el mensaje “Debes seleccionar un tipo de perfil”.</li> <li>(7.3) Si no se ha seleccionado ningún entorno salta el mensaje “Debes seleccionar al menos un entorno”.</li> <li>(7.4) Si ya existe un usuario con el correo introducido saltará un mensaje de alerta en pantalla: “ERROR, ya existe un usuario con ese email” y devuelve a la tabla paginada de usuarios sin crear el nuevo usuario.</li> </ol>

Tabla 23. Descripción de caso de uso: crear nuevo usuario

<b>UC021 EDITAR USUARIO</b>	
<b>Caso de uso</b>	Editar usuario.
<b>Actores</b>	Usuario administrador.
<b>Objetivo</b>	Permitir editar la información de un usuario existente en la aplicación.
<b>Descripción</b>	Editar información de usuario.
<b>Precondiciones</b>	El usuario administrador debe haberse autenticado en la aplicación (UC001). El usuario administrador debe encontrarse en la pestaña de Gestión de Usuarios.
<b>Postcondiciones</b>	Usuario modificado con datos actualizados en base de datos y página de gestión de usuarios recargada.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario pincha en el botón de editar usuario en la izquierda de la fila del usuario que desea editar.</li> <li>2. Se abre una ventana de editar usuario con toda la información del usuario seleccionado.</li> <li>3. Si el perfil era desarrollador se verán los entornos y proyectos que tiene asociados.</li> <li>4. Si el perfil era administrador no se muestran entornos ni proyectos ya que tiene todos asociados.</li> <li>5. El usuario realiza los cambios que desee y pulsa en el botón de “Editar Usuario”.</li> <li>6. El sistema guarda la información y recarga la tabla paginada.</li> </ol>
<b>Flujo alternativo</b>	<p>(2.1) Si el usuario pincha en cualquier punto de la pantalla fuera de la ventana se cierra.</p> <p>(2.2) Si el usuario pincha en la cruz superior derecha se cierra la ventana.</p> <p>(3.1) o (6.1) Si no se introduce nada en el campo de nombre, apellidos o email salta el mensaje “Campo Obligatorio”.</p> <p>(6.2) Si no se ha seleccionado ningún entorno y el perfil es de desarrollador salta el mensaje “Debes seleccionar al menos un entorno”.</p> <p>(6.3) Si ya existe un usuario con el correo introducido saltará un mensaje de alerta en pantalla: “ERROR, ya existe un usuario con ese email” y devuelve a la tabla paginada de usuarios sin crear el nuevo usuario.</p>

Tabla 24. Descripción de caso de uso: editar usuario

<b>UC022 DAR DE BAJA USUARIO</b>	
<b>Caso de uso</b>	Dar de baja usuario
<b>Actores</b>	Usuario administrador.
<b>Objetivo</b>	Permitir dar de baja un usuario. Al darle de baja no se elimina su información de base de datos, sino que se deja de mostrar como usuario de la aplicación.
<b>Descripción</b>	Dar de baja un usuario existente.
<b>Precondiciones</b>	El usuario administrador debe haberse autenticado en la aplicación (UC001). El usuario administrador debe encontrarse en la pestaña de Gestión de Usuarios.
<b>Postcondiciones</b>	El usuario dado de baja tendrá la fecha de baja fijada, lo que evitará que pueda entrar en la aplicación o que aparezca como usuario de la aplicación.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario pincha en la cruz de la izquierda en la fila del usuario que desea eliminar.</li> <li>2. Se abre una ventana de confirmación para dar de baja al usuario seleccionado.</li> <li>3. El usuario pincha en “Dar de Baja”.</li> </ol>
<b>Flujo alternativo</b>	<p>(2.1) Si el usuario pincha en “Cancelar” la ventana se cierra sin dar de baja al usuario.</p> <p>(2.2) Si el usuario pincha en cualquier punto de la pantalla fuera de la ventana se cierra sin dar de baja al usuario.</p> <p>(2.3) Si el usuario pincha en la cruz superior derecha se cierra la ventana sin dar de baja al usuario.</p>

*Tabla 25. Descripción de caso de uso: dar de baja usuario*

<b>UC023 VER LISTA COMPLETA DE PROYECTOS</b>	
<b>Caso de uso</b>	Ver lista de proyectos.
<b>Actores</b>	Usuario administrador.
<b>Objetivo</b>	Permitir ver la lista completa de proyectos de la aplicación con su información principal.

<b>Descripción</b>	Ver lista de proyectos de la aplicación.
<b>Precondiciones</b>	El usuario administrador debe haberse autenticado en la aplicación (UC001).
<b>Postcondiciones</b>	Ninguna.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario pincha en la pestaña de Gestión de Proyectos.</li> <li>2. El sistema muestra todos los proyectos de la aplicación en una tabla paginada con las siguientes columnas: nombre, entorno y equipo (lista de participantes).</li> </ol>
<b>Flujo alternativo</b>	(0.1) Si la sesión del usuario ha expirado la aplicación volverá a la página de autenticación.

*Tabla 26. Descripción de caso de uso: ver lista completa de proyectos*

<b>UC024 CREAR NUEVO PROYECTO</b>	
<b>Caso de uso</b>	Crear nuevo proyecto.
<b>Actores</b>	Usuario administrador.
<b>Objetivo</b>	Permitir la creación de un nuevo proyecto.
<b>Descripción</b>	Crear un nuevo proyecto.
<b>Precondiciones</b>	<p>El usuario administrador debe haberse autenticado en la aplicación (UC001).</p> <p>El usuario administrador debe encontrarse en la pestaña de Gestión de Proyectos.</p>
<b>Postcondiciones</b>	Proyecto creado y guardado en base de datos y directorio raíz creado en el servidor y en base de datos.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario pincha en el botón de nuevo proyecto en la esquina superior izquierda de la tabla paginada.</li> <li>2. Se abre una ventana de Nuevo Proyecto con dos campos a introducir: el nombre en un cuadro de texto, y tres botones para seleccionar su entorno.</li> <li>3. Al seleccionar alguno de los entornos se mostrarán los usuarios con acceso a ese entorno en una lista con checkboxes para añadirlos al proyecto. Los usuarios de tipo administrador estarán seleccionados por defecto y no</li> </ol>

	<p>podrán deseleccionarse.</p> <ol style="list-style-type: none"> <li>4. El usuario pincha en “Añadir Proyecto”.</li> <li>5. El sistema guarda la información del nuevo proyecto en base de datos y crea un directorio raíz para el proyecto con el mismo nombre que éste. Guarda el directorio en base de datos y lo crea en el servidor y devuelve a la página de Gestión de Proyectos.</li> </ol>
<b>Flujo alternativo</b>	<p>(2.1) Si el usuario pincha en cualquier punto de la pantalla fuera de la ventana se cierra.</p> <p>(2.2) Si el usuario pincha en la cruz superior derecha se cierra la ventana.</p> <p>(4.1) Si no se introduce nada en el campo de nombre salta el mensaje “Campo Obligatorio”.</p> <p>(4.2) Si no se ha seleccionado ningún entorno salta el mensaje “Debes seleccionar un entorno”.</p>

*Tabla 27. Descripción de caso de uso: crear nuevo proyecto*

<b>UC025 EDITAR PROYECTO</b>	
<b>Caso de uso</b>	Editar proyecto.
<b>Actores</b>	Usuario administrador.
<b>Objetivo</b>	Permitir la modificación de un proyecto.
<b>Descripción</b>	Modificar proyecto existente.
<b>Precondiciones</b>	<p>El usuario administrador debe haberse autenticado en la aplicación (UC001).</p> <p>El usuario administrador debe encontrarse en la pestaña de Gestión de Proyectos.</p>
<b>Postcondiciones</b>	Proyecto modificado en base de datos y nombre de directorio raíz modificado en el servidor y en base de datos.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario pincha en el botón de editar proyecto a la izquierda de la fila del proyecto a editar.</li> <li>2. Se abre una ventana de Editar Proyecto con la información del proyecto completada.</li> <li>3. El usuario modifica la información a su gusto.</li> </ol>

	<ol style="list-style-type: none"> <li>4. El usuario pincha en “Editar Proyecto”.</li> <li>5. El sistema actualiza la información del proyecto en base de datos, actualiza el nombre del directorio raíz en base de datos y en el servidor y devuelve a la página de Gestión de Proyectos.</li> </ol>
<b>Flujo alternativo</b>	<p>(2.1) Si el usuario pincha en cualquier punto de la pantalla fuera de la ventana se cierra sin modificar el usuario.</p> <p>(2.2) Si el usuario pincha en la cruz superior derecha se cierra la ventana sin modificar el usuario.</p> <p>(4.1) Si no se introduce nada en el campo de nombre salta el mensaje “Campo Obligatorio”.</p>

Tabla 28. Descripción de caso de uso: editar proyecto

UC026 ELIMINAR PROYECTO	
<b>Caso de uso</b>	Eliminar proyecto.
<b>Actores</b>	Usuario administrador.
<b>Objetivo</b>	Permitir la eliminación de un proyecto y de todo lo que este contiene de base de datos.
<b>Descripción</b>	Eliminar un proyecto, sus directorios, ficheros, y ejecuciones pasadas.
<b>Precondiciones</b>	<p>El usuario administrador debe haberse autenticado en la aplicación (UC001).</p> <p>El usuario administrador debe encontrarse en la pestaña de Gestión de Proyectos.</p>
<b>Postcondiciones</b>	Proyecto y todo lo que este contiene eliminados de base de datos y servidor.
<b>Flujo normal</b>	<ol style="list-style-type: none"> <li>1. El usuario pincha en la cruz a la izquierda de la fila del proyecto a eliminar.</li> <li>2. Se abre una ventana para confirmar la eliminación del proyecto.</li> <li>3. El usuario pincha en “Eliminar Proyecto”.</li> <li>4. El sistema elimina el directorio raíz de base de datos y del servidor. La aplicación está desarrollada de forma que se eliminen a su vez todos los ficheros y directorios hijos del</li> </ol>

	<p>directorio padre. Eliminará también todas las ejecuciones pasadas que se hayan realizado en el proyecto y por último el proyecto en sí. Una vez eliminado todo devuelve a la página de Gestión de Proyectos.</p>
<p><b>Flujo alternativo</b></p>	<p>(2.1) Si el usuario pincha en “Cancelar” la ventana se cierra sin eliminar el proyecto.</p> <p>(2.2) Si el usuario pincha en cualquier punto de la pantalla fuera de la ventana se cierra sin eliminar el proyecto.</p> <p>(2.3) Si el usuario pincha en la cruz superior derecha se cierra la ventana sin eliminar el proyecto.</p>

*Tabla 29. Descripción de caso de uso: eliminar proyecto*

### 5.3.4 DIAGRAMA DE CLASES

Ya que el lenguaje utilizado para el desarrollo de la aplicación es Java, y se trata de programación orientada a objetos, el diseño del sistema se ha realizado con una estructura de clases.

A continuación, se muestra un diagrama en el que se detallan las principales clases de la aplicación:

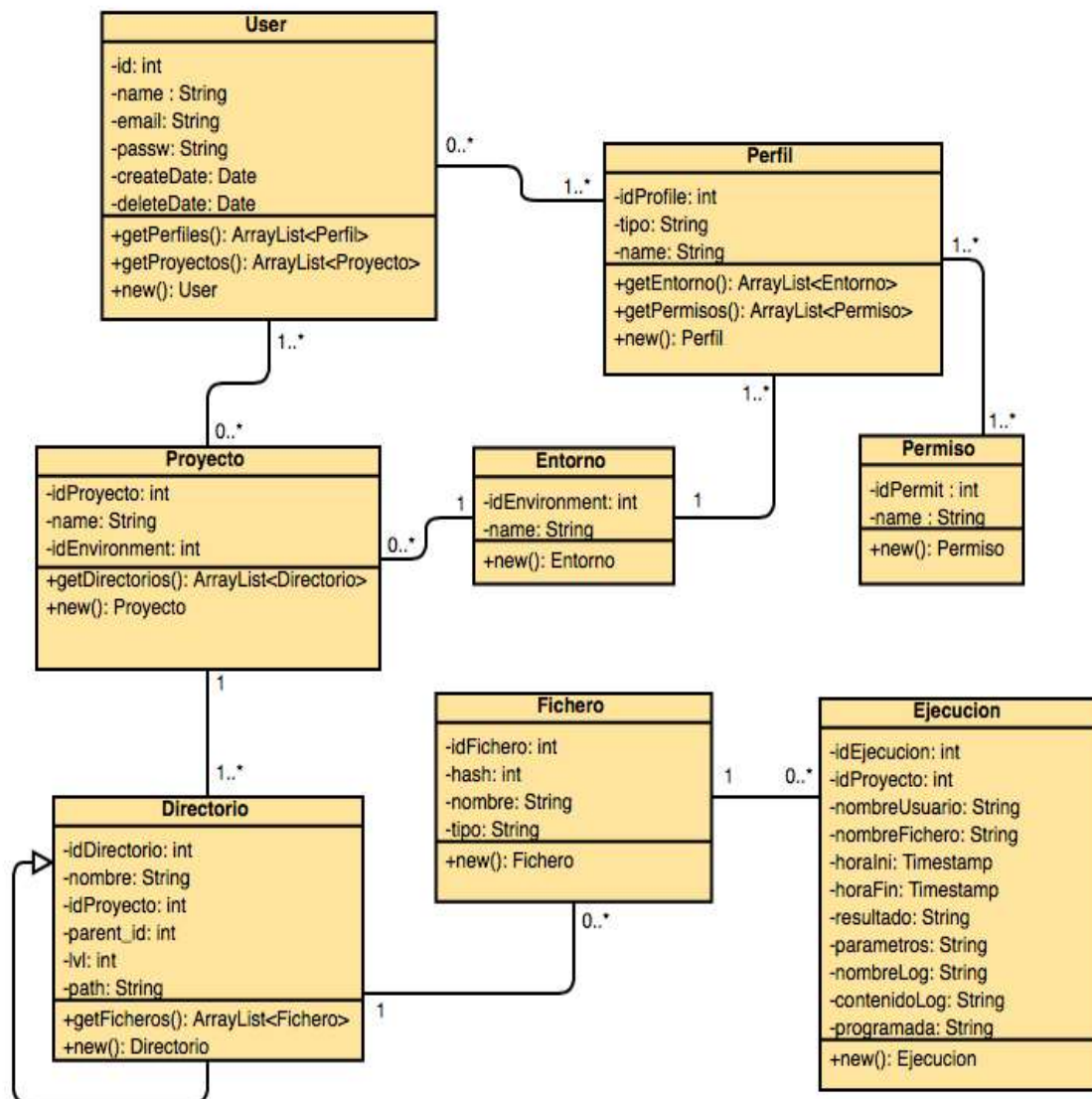


Figura 16. Diagrama de clases WebApp



### 5.3.5 DESCRIPCIÓN DE CLASES

En las siguientes tablas se representa información detallada de las clases definidas en el apartado anterior.

Para cada clase tenemos su nombre, descripción, atributos y lo que estos significan, y operaciones y lo que hacen.

CL001 USER	
<b>Nombre</b>	User
<b>Descripción</b>	Clase que representa la información de un usuario del sistema.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>▪ <b>id</b>: identificador unívoco de usuario.</li> <li>▪ <b>name</b>: nombre de usuario.</li> <li>▪ <b>email</b>: email de usuario.</li> <li>▪ <b>passw</b>: clave interna del usuario.</li> <li>▪ <b>createDate</b>: fecha de creación del usuario.</li> <li>▪ <b>deleteDate</b>: fecha de baja del usuario.</li> </ul>
<b>Operaciones</b>	<ul style="list-style-type: none"> <li>▪ <b>getPerfiles()</b>: devuelve la lista de perfiles del usuario.</li> <li>▪ <b>getProyectos()</b>: devuelve la lista de proyectos asociados al usuario.</li> <li>▪ <b>new()</b>: crea un nuevo usuario.</li> </ul>

*Tabla 30. Descripción de clase: User*

CL002 PERMISO	
<b>Nombre</b>	Permiso
<b>Descripción</b>	Clase que representa la información de un permiso de usuario del sistema.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>▪ <b>idPermit</b>: identificador unívoco de permiso.</li> <li>▪ <b>name</b>: nombre del permiso.</li> </ul>
<b>Operaciones</b>	<ul style="list-style-type: none"> <li>▪ <b>new()</b>: crea un nuevo permiso.</li> </ul>

*Tabla 31. Descripción de clase: Permiso*

CL003 ENTORNO	
<b>Nombre</b>	Entorno
<b>Descripción</b>	Clase que representa la información de un entorno de la aplicación.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>▪ <b>idEnvironment</b>: identificador unívoco de entorno.</li> <li>▪ <b>name</b>: nombre del entorno.</li> </ul>
<b>Operaciones</b>	<ul style="list-style-type: none"> <li>▪ <b>new()</b>: crea un nuevo entorno.</li> </ul>

*Tabla 32. Descripción de clase: Entorno*

CL004 PERFIL	
<b>Nombre</b>	Perfil
<b>Descripción</b>	Clase que representa la información de un perfil de usuario del sistema.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>▪ <b>idProfile</b>: identificador unívoco del perfil.</li> <li>▪ <b>tipo</b>: tipo de perfil: administrador o desarrollador.</li> <li>▪ <b>name</b>: nombre del perfil.</li> </ul>
<b>Operaciones</b>	<ul style="list-style-type: none"> <li>▪ <b>getEntornos()</b>: devuelve la lista de entornos del perfil.</li> <li>▪ <b>getProyectos()</b>: devuelve la lista de permisos asociados al perfil.</li> <li>▪ <b>new()</b>: crea un nuevo perfil.</li> </ul>

*Tabla 33. Descripción de clase: Perfil*

CL005 PROYECTO	
<b>Nombre</b>	Proyecto
<b>Descripción</b>	Clase que representa la información de un proyecto existente.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>▪ <b>idProyecto</b>: identificador unívoco de proyecto.</li> <li>▪ <b>name</b>: nombre de proyecto.</li> <li>▪ <b>idEnvironment</b>: identificador del entorno al que pertenece.</li> </ul>
<b>Operaciones</b>	<ul style="list-style-type: none"> <li>▪ <b>getDirectorios()</b>: devuelve la lista de directorios dentro del proyecto.</li> <li>▪ <b>new()</b>: crea un nuevo proyecto.</li> </ul>

*Tabla 34. Descripción de clase: Proyecto*

CL006 DIRECTORIO	
<b>Nombre</b>	Directorio
<b>Descripción</b>	Clase que representa la información de un directorio dentro de un proyecto.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>▪ <b>idDirectorio</b>: identificador unívoco de directorio.</li> <li>▪ <b>nombre</b>: nombre del directorio.</li> <li>▪ <b>idProyecto</b>: id del proyecto donde se encuentra el directorio.</li> <li>▪ <b>parent_id</b>: identificador del directorio padre del fichero. Los directorios raíz guardan el valor 0.</li> <li>▪ <b>lvl</b>: numero entero que indica el nivel en el que se encuentra el directorio dentro del árbol de directorios.</li> <li>▪ <b>path</b>: indica la ruta del directorio dentro del proyecto. Junto con el atributo lvl permiten mostrar el árbol de directorios.</li> </ul>
<b>Operaciones</b>	<ul style="list-style-type: none"> <li>▪ <b>getFicheros ()</b>: devuelve la lista de ficheros dentro del directorio.</li> <li>▪ <b>new()</b>: crea un nuevo directorio.</li> </ul>

*Tabla 35. Descripción de clase: Directorio*

CL007 FICHERO	
<b>Clase</b>	Fichero
<b>Descripción</b>	Clase que representa la información de un fichero dentro de un directorio y a su vez de un proyecto.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>▪ <b>idFichero</b>: identificador unívoco de fichero.</li> <li>▪ <b>hash</b>: código hash para el nombre del fichero. Utilizado para guardar los ficheros en el servidor.</li> <li>▪ <b>nombre</b>: nombre del fichero.</li> <li>▪ <b>tipo</b>: indica el tipo de fichero: de texto(.txt) o ejecutable (.jar).</li> </ul>
<b>Operaciones</b>	<ul style="list-style-type: none"> <li>▪ <b>new()</b>: crea un nuevo fichero.</li> </ul>

*Tabla 36. Descripción de clase: Fichero*

CL008 EJECUCION	
<b>Clase</b>	Ejecución
<b>Descripción</b>	Clase que representa toda la información relativa a una ejecución de un fichero jar.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>▪ <b>idEjecucion:</b> identificador unívoco de la ejecución.</li> <li>▪ <b>idProyecto:</b> identificador del proyecto en el que se ha realizado la ejecución.</li> <li>▪ <b>nombreUsuario:</b> nombre del usuario que ha realizado la ejecución.</li> <li>▪ <b>nombreFichero:</b> nombre del fichero que se ha ejecutado.</li> <li>▪ <b>horaIni:</b> hora de inicio de la ejecución.</li> <li>▪ <b>horaFin:</b> hora de finalización de la ejecución.</li> <li>▪ <b>resultado:</b> resultado de la ejecución (success o fail).</li> <li>▪ <b>parametros:</b> parámetros de ejecución introducidos.</li> <li>▪ <b>nombreLog:</b> nombre del fichero .log que se creará en caso de que el usuario quiera descargarlo.</li> <li>▪ <b>contenidoLog:</b> traza de ejecución completa del fichero que se guardará en el fichero log.</li> <li>▪ <b>Programada:</b> indica si la ejecución se ha programado o ejecutado al momento.</li> </ul>
<b>Operaciones</b>	<ul style="list-style-type: none"> <li>▪ <b>new():</b> crea una nueva ejecución.</li> </ul>

*Tabla 37. Descripción de clase: Ejecución*

## 5.4 MODELO DE DATOS

### 5.4.1 MODELO FÍSICO DE DATOS

A continuación se muestra el diseño físico de la base de datos del sistema realizado con la ayuda de MySQL Workbench:

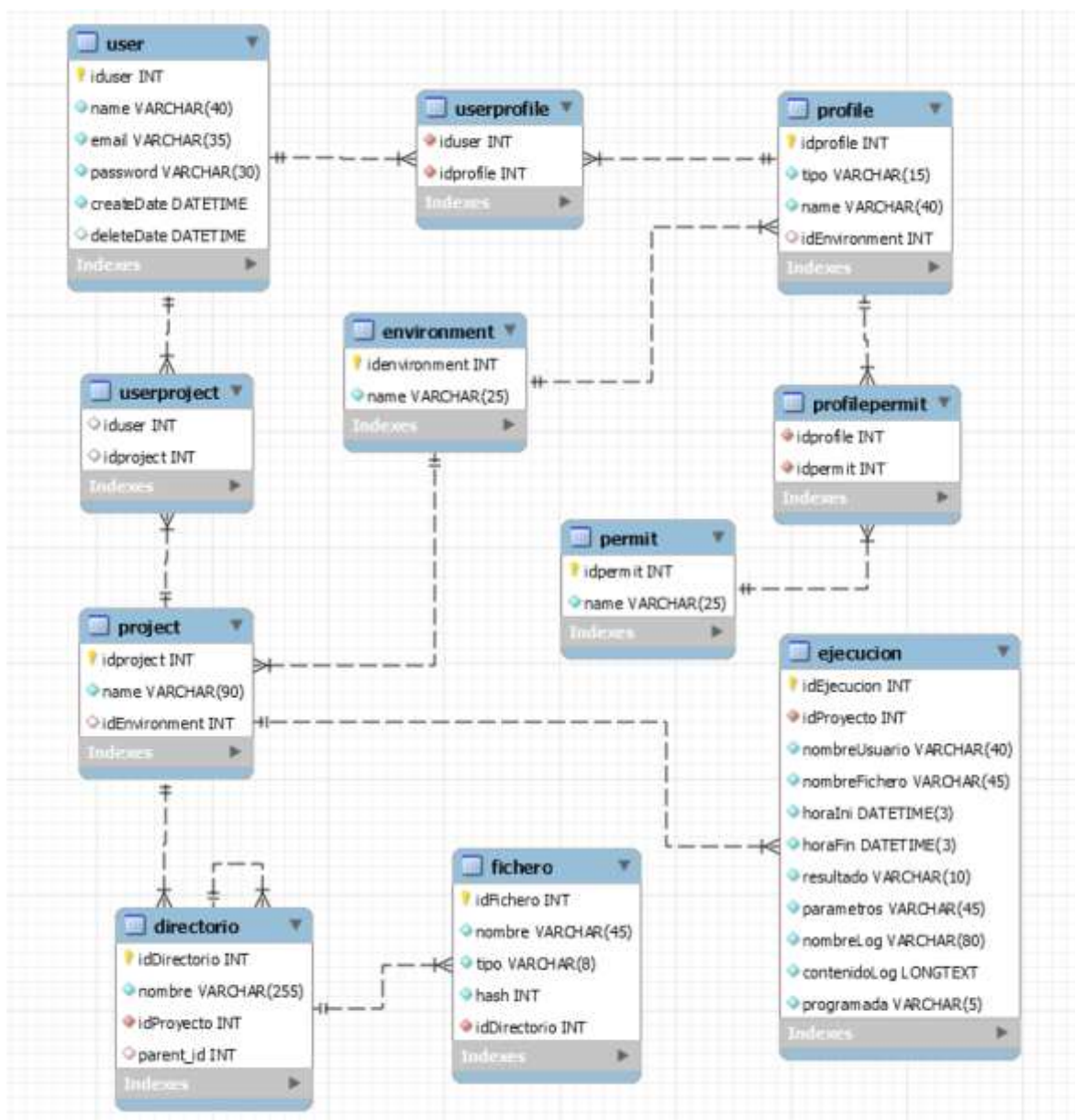


Figura 17. Modelo de datos WebApp

## 5.4.2 DESCRIPCIÓN DE ENTIDADES

A continuación, se describen las entidades representadas en el modelo físico de datos. Para cada entidad se ha hecho una tabla con sus atributos, el tipo del atributo, la descripción, otras características (columna “otros”), y sus restricciones.

La notación utilizada en la columna de “Otros” para cada identificador de la entidad es la siguiente:

ABREVIATURA	SIGNIFICADO
<b>PK</b>	Clave primaria
<b>FK</b>	Clave foránea.
<b>NN</b>	No nulo.
<b>AI</b>	Incremento automático.

*Tabla 38. Notación tablas entidad.*

<b>E001 USER</b>			
La tabla USER guarda la información de los usuarios de la aplicación en base de datos. Conectada usando su identificador único con las tablas USERPROFILE y USERPROJECT, que constituyen los perfiles de cada usuario y sus proyectos asignados.			
ATRIBUTO	TIPO	DESCRIPCIÓN	OTROS
<b>idUser</b>	INT	Identificador de usuario.	PK, NN, AI
<b>name</b>	VARCHAR(40)	Nombre de usuario.	NN
<b>email</b>	VARCHAR(35)	Correo electrónico de usuario.	NN
<b>password</b>	VARCHAR(30)	Contraseña de usuario.	NN
<b>createDate</b>	DATETIME	Fecha de creación de usuario.	NN
<b>deleteDate</b>	DATETIME	Fecha de baja de usuario.	-
<b>RESTRICCIONES</b>			
<ul style="list-style-type: none"> <li>- El atributo <i>createDate</i> debe inicializarse con el valor por defecto del momento en el que se crea, es decir, su Default Expression es CURRENT_TIMESTAMP.</li> <li>- El atributo <i>deleteDate</i> debe inicializarse con el valor por defecto nulo, es decir, su Default Expression es NULL.</li> </ul>			

*Tabla 39. Descripción de entidades: user*

### E002 USERPROFILE

La tabla USERPROFILE guarda la información sobre los perfiles de cada usuario. Como ya vimos en el diagrama de clases, un usuario puede tener varios perfiles, y un perfil puede asignarse a más de un usuario. Es por eso por lo que se necesitaba una tabla nueva que guardase en cada fila cada par de identificador de usuario e identificador de perfil.

ATRIBUTO	TIPO	DESCRIPCIÓN	OTROS
<b>idUser</b>	INT	Identificador de usuario.	NN, FK
<b>idProfile</b>	INT	Identificador de perfil.	NN, FK

#### RESTRICCIONES

- El atributo *idUser* es clave foránea del atributo idUser de la tabla USER.
- El atributo *idProfile* es clave foránea del atributo idProfile de la tabla PROFILE.

*Tabla 40. Descripción de entidades: userprofile*

### E003 USERPROJECT

La tabla USERPROJECT guarda la información sobre los proyectos a los que cada usuario tiene acceso. Como ya vimos en el diagrama de clases, un usuario puede tener varios proyectos, y un proyecto puede estar asignado a más de un usuario. Es por eso por lo que se necesitaba una tabla nueva que guardase en cada fila cada par de identificador de usuario e identificador de proyecto.

ATRIBUTO	TIPO	DESCRIPCIÓN	OTROS
<b>idUser</b>	INT	Identificador de usuario.	NN, FK
<b>idProject</b>	INT	Identificador de proyecto.	NN, FK

#### RESTRICCIONES

- El atributo *idUser* es clave foránea del atributo idUser de la tabla USER.
- El atributo *idProject* es clave foránea del atributo idproject de la tabla PROJECT.

*Tabla 41. Descripción de entidades: userproject*

### E004 PROFILE

La tabla PROFILE guarda la información de los perfiles disponibles en la aplicación en base de datos. Conectada usando su identificador único con la tabla USERPROFILE, y usando su atributo idEnvironment con la tabla ENVIRONMENT (ya que cada perfil tiene asignado un único entorno en la aplicación).

ATRIBUTO	TIPO	DESCRIPCIÓN	OTROS
<b>idProfile</b>	INT	Identificador de perfil.	PK, NN, AI
<b>tipo</b>	VARCHAR(15)	Tipo de perfil (admin/ des).	NN
<b>name</b>	VARCHAR(40)	Nombre del perfil.	NN
<b>idEnvironment</b>	INT	Identificador de entorno.	FK

#### RESTRICCIONES

- El atributo *idEnvironment* es clave foránea del atributo idenvironment de la tabla ENVIRONMENT.
- El atributo *idEnvironment* debe inicializarse con el valor por defecto nulo, es decir, su Default Expression es NULL.

Tabla 42. Descripción de entidades: profile

### E005 PROFILEPERMIT

La tabla PROFILEPERMIT guarda la información sobre los permisos que incluye cada perfil de usuario. Como ya vimos en el diagrama de clases, un perfil puede tener varios permisos, y un permiso puede estar asignado a más de un perfil. Es por eso por lo que se necesitaba una tabla nueva que guardase en cada fila cada par de identificador de perfil e identificador de permiso.

ATRIBUTO	TIPO	DESCRIPCIÓN	OTROS
<b>idprofile</b>	INT	Identificador de usuario.	NN, FK
<b>idpermit</b>	INT	Nombre de usuario.	NN, FK

#### RESTRICCIONES

- El atributo *idprofile* es clave foránea del atributo idprofile de la tabla PROFILE.
- El atributo *idpermit* es clave foránea del atributo idpermit de la tabla PERMIT.

Tabla 43. Descripción de entidades: profilepermit



<b>E006 PERMIT</b>			
La tabla PERMIT guarda la información de los permisos de la aplicación en base de datos. Conectada usando su identificador único con la tabla PROFILEPERMIT.			
ATRIBUTO	TIPO	DESCRIPCIÓN	OTROS
<b>idPermit</b>	INT	Identificador de usuario.	PK, NN, AI
<b>name</b>	VARCHAR(25)	Nombre de usuario.	NN
<b>RESTRICCIONES</b>			
Ninguna.			

*Tabla 44. Descripción de entidades: permit*

<b>E007 PROJECT</b>			
La tabla PROJECT guarda la información de los proyectos de la aplicación en base de datos. Conectada usando su identificador único con la tabla USERPROJECT, y usando su atributo idEnvironment con la tabla ENVIRONMENT (un proyecto está asociado a un único entorno).			
ATRIBUTO	TIPO	DESCRIPCIÓN	OTROS
<b>idProject</b>	INT	Identificador de proyecto.	PK, NN, AI
<b>name</b>	VARCHAR(90)	Nombre de proyecto.	NN
<b>idEnvironment</b>	INT	Identificador de entorno.	FK
<b>RESTRICCIONES</b>			
<ul style="list-style-type: none"> <li>- El atributo <i>idEnvironment</i> debe inicializarse con el valor por defecto nulo, es decir, su Default Expression es NULL.</li> <li>- El atributo <i>idEnvironment</i> es clave foránea del atributo idenvironment de la tabla ENVIRONMENT.</li> </ul>			

*Tabla 45. Descripción de entidades: project*

### E008 ENVIRONMENT

La tabla ENVIRONMENT guarda la información de los entornos existentes en la aplicación. Conectada usando su identificador único con las tablas PROJECT y PROFILE.

ATRIBUTO	TIPO	DESCRIPCIÓN	OTROS
<b>idenvironment</b>	INT	Identificador de entorno.	PK, NN, AI
<b>name</b>	VARCHAR(25)	Nombre de entorno.	NN

#### RESTRICCIONES

- El atributo *createDate* debe inicializarse con el valor por defecto del momento en el que se crea, es decir, su Default Expression es CURRENT\_TIMESTAMP.
- El atributo *deleteDate* debe inicializarse con el valor por defecto nulo, es decir, su Default Expression es NULL.

Tabla 46. Descripción de entidades: environment

### E009 DIRECTORIO

La tabla DIRECTORIO guarda la información de los usuarios de la aplicación en base de datos. Conectada consigo misma usando su identificador único y su atributo parent\_id, con la tabla FICHERO usando su identificador único, y con la tabla proyecto usando el identificador idProyecto.

ATRIBUTO	TIPO	DESCRIPCIÓN	OTROS
<b>idDirectorio</b>	INT	Identificador de directorio.	PK, NN, AI
<b>nombre</b>	VARCHAR(255)	Nombre de directorio.	NN
<b>idProyecto</b>	INT	Identificador de proyecto.	NN, FK
<b>parent_id</b>	INT	Identificador del fichero padre.	-

#### RESTRICCIONES

- El atributo *parent\_id* debe inicializarse con el valor por defecto nulo, es decir, su Default Expression es NULL. De esta forma podrán distinguirse los directorios sin directorio padre como directorios raíz.
- El atributo *idProyecto* es clave foránea del atributo idproject de la tabla PROJECT.
- El atributo *parent\_id* es clave foránea del atributo idDirectorio de la misma tabla DIRECTORIO.

Tabla 47. Descripción de entidades: directorio

### E010 FICHERO

La tabla FICHERO guarda la información de los ficheros existentes en cada proyecto en base de datos. Conectada usando su atributo idDirectorio con la tabla DIRECTORIO.

ATRIBUTO	TIPO	DESCRIPCIÓN	OTROS
<b>idFichero</b>	INT	Identificador de fichero.	PK, NN, AI
<b>nombre</b>	VARCHAR(45)	Nombre del fichero.	NN
<b>tipo</b>	VARCHAR(8)	Tipo de fichero (.txt/.jar).	NN
<b>hash</b>	INT	Código hash para guardar el fichero en el servidor.	NN
<b>idDirectorio</b>	INT	Identificador del directorio en el que se encuentra.	NN, FK

### RESTRICCIONES

- El atributo *idDirectorio* es clave foránea del atributo idDirectorio de la tabla DIRECTORIO.

Tabla 48. Descripción de entidades: fichero

### E011 EJECUCION

La tabla EJECUCION guarda la información de las ejecuciones realizadas en cada proyecto. Conectada usando su atributo idProyecto con la tabla PROJECT.

ATRIBUTO	TIPO	DESCRIPCIÓN	OTROS
<b>idEjecucion</b>	INT	Identificador de ejecución	PK, NN, AI
<b>idProyecto</b>	INT	Identificador de proyecto.	NN, FK
<b>nombreUsuario</b>	VARCHAR(40)	Nombre de usuario que realizó la ejecución.	NN
<b>nombreFichero</b>	VARCHAR(45)	Nombre del fichero que se ejecutó creando la ejecución.	NN
<b>horaIni</b>	DATETIME(3)	Hora (y fecha) a la que se inició la ejecución.	NN
<b>horaFin</b>	DATETIME(3)	Hora (y fecha) a la que acabó la ejecución.	NN

<b>resultado</b>	VARCHAR(10)	Resultado de la ejecución (success o fail).	NN
<b>parametros</b>	VARCHAR(45)	Parámetros de ejecución.	NN
<b>nombreLog</b>	VARCHAR(80)	Nombre con el que se guardará el fichero log con la traza de ejecución.	NN
<b>contenidoLog</b>	LONGTEXT	Traza de ejecución.	NN
<b>programada</b>	VARCHAR(5)	Sí o no según si la ejecución se hizo al instante o fue programada.	NN
<b>RESTRICCIONES</b>			
- El atributo <i>idProyecto</i> es clave foránea del atributo <i>idproject</i> de la tabla PROYECTO.			

*Tabla 49. Descripción de entidades: ejecución*

## **Capítulo 6. ANÁLISIS DE RESULTADOS**

En el apartado 4.2 de esta memoria se definieron una serie de objetivos que el sistema debía de cumplir. Una vez acabadas todas las etapas del proyecto se puede determinar que estos objetivos se han cumplido de manera satisfactoria.

Pasaré a definir estos objetivos y la forma en la que se han alcanzado más concretamente.

El primer objetivo era el de la creación de una plataforma que fuese capaz de gestionar usuarios, perfiles y permisos. Empezando desde la creación de las tablas en bases de datos, hasta el diseño de las interfaces que permiten realizar todas las tareas relacionadas con la gestión de usuarios: creación de nuevos usuarios, baja y modificación de usuarios existentes, y asignación de entornos y perfiles con sus consiguientes permisos.

El segundo objetivo aumentaba las funcionalidades del sistema con la gestión de los proyectos. Para ello hubo que crear nuevas tablas en bases de datos y diseñar nuevas interfaces que permitiesen crear, modificar o eliminar proyectos, además de gestionar sus relaciones con los usuarios de la aplicación. Una vez conseguido, se crearon el resto de tablas e interfaces necesarias para posibilitar el diseño y la creación de un árbol de directorios dentro de cada proyecto por parte de los usuarios con los permisos correspondientes.

El tercer objetivo fue, aunque no el más largo de conseguir, el más importante, y el que aporta a la aplicación su funcionalidad principal: la gestión de los ficheros. Los usuarios con los permisos adecuados, debían ser capaces de crear, importar, modificar, eliminar, o ejecutar ficheros en la aplicación. Para ello no solo tuve que crear nuevas tablas en base de datos y nuevas clases en la aplicación, sino que trabajé con funciones y clases java que me permitiesen guardar la información y la estructura de los directorios en una dirección en el servidor (configurable en el fichero “.properties”), y ejecutar los ficheros .jar.

Este tercer objetivo estaba estrechamente ligado al último: la recuperación de las trazas y el resto de información relevante de las ejecuciones realizadas para su posterior revisión. De nuevo la información de estas ejecuciones se almacenó en base de datos, y se mostró en la aplicación junto con la opción de descarga del fichero “. pom” con la traza de ejecución completa.

Al finalizar este objetivo se había conseguido generar un sistema robusto, fácilmente mantenible y escalable. Se han seguido las directrices establecidas por su patrón de diseño al que ya hice referencia y expliqué en el capítulo 5.1 de esta misma memoria, consiguiendo cumplir la lista completa de requisitos definidos en el capítulo 5.2.

Por último, indicar que la aplicación desarrollada, aunque aún está en su primera versión, podría incluirse ya en las propuestas de ofertas tanto en entornos de BI puros como en Big Data, así como ser comercializado de manera independiente.

A nivel personal, tener la posibilidad de realizar un proyecto tan completo desde su inicio, fase a fase y con la perspectiva de los diferentes roles implicados, no sólo me ha permitido aplicar, ampliar y afianzar mis conocimientos, si no que ha aumentado mi visión global sobre un proyecto real, mejorando mi entendimiento y seguridad en el desempeño de mis funciones como ingeniero.

## Capítulo 7. CONCLUSIONES Y TRABAJOS FUTUROS

La aplicación diseñada, aunque podría ser comercializada en estos instantes, abre un abanico enorme de posibilidades y ampliaciones, algunas de las cuales me habría gustado poder haber realizado yo mismo.

Dividiré las mejoras y trabajos futuros en 4 secciones: nuevas funcionalidades, entorno tecnológico, explotación de los datos, y mejoras de seguridad.

### **Nuevas funcionalidades o mejoras sobre las existentes:**

- Permitir la gestión de los distintos perfiles de la aplicación. Se crearía una nueva ventana como la de “Gestión de Usuarios” o “Gestión de Proyectos” que se llamase por ejemplo “Gestión de perfiles” en la que pudiesen gestionarse los distintos perfiles existentes en la aplicación mediante la creación de nuevos entornos (por ejemplo, entorno de pruebas) y permisos, y la asignación de relaciones entre estos.
- Para que tuviese algún sentido utilizar esta opción deberían existir además nuevas funcionalidades que requiriesen de un nuevo perfil de usuario que las realizase. Es por eso por lo que no lo desarrollamos en esta primera versión, pero se deja abierto para una posible ampliación.
- Permitir ver los usuarios dados de baja. Sería una funcionalidad nueva para los administradores en su ventana de “Gestión de Usuarios”. Actualmente los usuarios administradores pueden dar de baja a cualquier usuario, y, aunque su información no es eliminada de base de datos, el acceso a la aplicación usando sus credenciales se bloquea y su información deja de mostrarse en cualquier tabla de la aplicación. Es por eso por lo que para ver la información de cualquier usuario dado de baja es necesario acceder directamente a base de datos, y en caso de querer darlo de alta de nuevo modificar los datos desde ahí con un comando UPDATE de SQL. Una posible solución a esto sería un botón en la página de gestión de Usuarios que se

llamase “Usuarios dados de baja” que al pulsarse mostrase una tabla con todos los usuarios dados de baja y un botón para dar de alta de nuevo.

- Generación de correo electrónico con la contraseña generada automáticamente al crear un nuevo usuario en vez de mostrarla en pantalla con un mensaje de alerta.
- Permitir ver y gestionar las ejecuciones configuradas en una tabla paginada. De esta forma podrían modificarse o cancelarse las ejecuciones programadas. Para ello debería guardarse la información de las programaciones en base de datos.
- Aumentar los tipos de ficheros ejecutables que pueden subirse y ejecutarse en la aplicación. De las ampliaciones que menos tiempo llevarían, pero una de las más interesantes por afectar directamente al objetivo principal de la aplicación.
- Ampliar los parámetros de ejecución de los ficheros .jar. Podría incluso crear la opción de añadir un nuevo tipo de parámetro, pero para ello deberían guardarse los parámetros existentes en base de datos.

Cabe destacar que el código se ha desarrollado tratando de conseguir que cualquier mejora o modificación sobre éste requiera un mínimo esfuerzo. Tanto el nombre de las clases como el de los paquetes que las contienen facilitan además la comprensión del esquema de la aplicación, así como los comentarios sobre el código.

#### **Entorno tecnológico:**

- Adaptación a dispositivos móviles.
- Adaptación a distintos sistemas operativos.

#### **Explotación de datos:**

- Permitir llevar una gestión sobre la información registrada en el sistema mediante la generación de estadísticas e informes de seguimiento relativos a nuevos usuarios, usuarios dados de baja, proyectos dados de alta o eliminados, ficheros ejecutados, etc.



### **Seguridad:**

- La autenticación actualmente se realiza contra base de datos y es vulnerable a distintos ataques como bien podría ser inyección de SQL o la elevación de privilegios no autorizados aprovechando una vulnerabilidad de desbordamiento de búfer. Estos problemas podrían resolverse con una combinación de control de acceso a nivel de consulta, auditoría de base de datos e IPS (Sistemas de Prevención de Intrusiones) tradicionales.
- Modificar la autenticación de aplicación. En vez de contra base de datos, podría realizarse usando LDAPS. LDAP es un conjunto de protocolos de licencia abierta utilizados a nivel de aplicación para acceder a la información almacenada de forma centralizadas en un directorio remoto. LDAPS aumenta la seguridad además evitando ataques como el del “man in the middle”.

Aunque el trabajo realizado cumple con todos los objetivos propuestos, y yo personalmente estoy muy satisfecho y orgulloso del trabajo conseguido, la aplicación aún cuenta con un abanico enorme de posibilidades y ampliaciones que pueden hacer de ella una aplicación de uso extendido en entornos de Business Intelligence.

## Capítulo 8. BIBLIOGRAFÍA

- [1] [https://www.ecured.cu/Eclipse\\_entorno\\_de\\_desarrollo\\_integrado](https://www.ecured.cu/Eclipse_entorno_de_desarrollo_integrado)
- [2] <https://www.genbeta.com/desarrollo/eclipse-ide>
- [3] <https://es.wikipedia.org/wiki/MySQL>
- [4] <https://www.hostinger.es/tutoriales/que-es-mysql/>
- [5] <https://ubunlog.com/mysql-workbench-bases-datos/>
- [6] <https://es.wikipedia.org/wiki/Maven>
- [7] <https://www.genbeta.com/desarrollo/que-es-maven>
- [8] <http://www.t3cnologic.com/blog/2014/02/27/que-es-maven-que-ventajas-tiene-en-proyectos-java/>
- [9] <https://laurmolina7821.wordpress.com/1-1-3-aplicaciones-de-2-3-y-n-capas/>
- [10] <https://sites.google.com/site/aspnetutb/home/asp-net-y-mvc-3-1>
- [11] <https://www.profesionalreview.com/2019/01/05/ldap/>
- [12] [https://es.wikipedia.org/wiki/Java\\_\(lenguaje\\_de\\_programaci3n\)](https://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci3n))
- [13] [https://www.mundojava.net/caracteristicas-del-lenguaje.html?Pg=java\\_inicial\\_4\\_1.html](https://www.mundojava.net/caracteristicas-del-lenguaje.html?Pg=java_inicial_4_1.html)
- [14] <http://dis.um.es/~bmoros/Tutorial/parte2/cap2-5.html>
- [15] <http://bigdatabytecnalia.com/2015/07/big-data-desde-los-datos-al-conocimiento/>
- [16] <https://business-intelligence.grupobit.net/blog/empresas-que-usan-big-data-y-han-conseguido-los-mejores-resultados>
- [17] [https://www.corresponsables.com/actualidad/17-objetivos-para-hacer-mas-sostenible-y-habitable-el-mundo?gclid=Cj0KCQjwgJv4BRCrARIsAB17JI6HdnY16rqNqt4EtfDOlOlumld7RFLK4yVwvLkRoUsM-Uk3xuuK54EaAILIEALw\\_wcB](https://www.corresponsables.com/actualidad/17-objetivos-para-hacer-mas-sostenible-y-habitable-el-mundo?gclid=Cj0KCQjwgJv4BRCrARIsAB17JI6HdnY16rqNqt4EtfDOlOlumld7RFLK4yVwvLkRoUsM-Uk3xuuK54EaAILIEALw_wcB)
- [18] <https://www.bbva.com/es/la-contribucion-del-big-data-a-los-objetivos-de-desarrollo-sostenible/>
- [19] <https://www.un.org/es/sections/issues-depth/big-data-sustainable-development/index.html>
- [20] <https://www.foxize.com/blog/big-data-desarrollo-sostenible/>

## ANEXO I: MANUAL DE INSTALACIÓN

Para poder utilizar la aplicación desarrollada deberás tomar 5 pasos en total, y en caso de que alguno de ellos ya lo hayas realizado podrás saltarlo y pasar al siguiente:

1. Descargar e instalar Java en tu equipo.
2. Descargar e instalar Maven en tu equipo.
3. Descargar e instalar MySQL en tu equipo.
4. Configurar las bases de datos.
5. Descargar proyecto, configurar fichero .properties y ejecutarlo.

Este manual te guiará y te aportará toda la información necesaria para la realización de estos 5 pasos y conseguir así así la instalación de la aplicación.

### 1. **Descargar e instalar Java:**

Vaya a la página de descarga <https://www.java.com/es/download/manual.jsp> y seleccione la versión de Java que se adapte al sistema operativo de tu ordenador.

Una vez seleccionado se descargará el instalador, y para ejecutarlo haga click en él y pulse en Ejecutar e Instalar para aceptar los términos de licencia y continuar con la instalación. Durante la instalación se abrirán varios cuadros de diálogo con información para completar las distintas etapas del proceso de instalación. Haga click en siguiente hasta llegar al último cuadro y pulse en “Cerrar”.



*Figura 18. Instalación Java. Manual Instalación.*

Configuramos las variables de entorno de Windows. Para ello hacemos click derecho sobre Equipo> Propiedades> Configuración avanzada del sistema> Opciones avanzadas> Variables de entorno, y añadimos las siguientes:

- CLASSPATH: C:\Program Files\Java\jdk 1.8.0\_161\jre
- JAVA\_HOME: C:\Program Files\Java\jdk 1.8.0\_161
- PATH: C:\Program Files\Java\jdk 1.8.0\_161\bin;

La versión de jdk será la que tu hayas instalado.

Para comprobar que ya tienes java instalado en tu ordenador correctamente escriba en su terminal `java -version` y comprueba que devuelve la versión instalada.

## **2. Descargar e instalar Maven en tu equipo:**

Accedemos a la página web de Maven y descargamos la última versión:

<http://maven.apache.org/download.cgi#>

Descomprimos el archivo en C:/maven, de modo que todos los archivos de maven se encontrarán en C:/maven/apache-maven(versión).

Configuramos las variables de entorno como en el caso anterior:

- M2: %M2\_HOME%\bin
- M2\_HOME: C:\maven\apache-maven-3.6.1
- PATH:C:\maven\apache-maven-3.6.1\bin;

Accedemos al terminal y escribimos `mvn -v`. Asegúrate de que el terminal no lo tenías ya abierto cuando cambiaste las variables de sesión, porque no reconocerá las variables de ser así. Debes cerrarlo y volver a abrirlo. Si todo está bien mostrará la versión de Maven instalada.

Si trabajas con eclipse por ejemplo y ya tienes el plugin de maven instalado no tendrás por qué hacer este paso ya que podrás ejecutar el proyecto desde eclipse.

Maven será la encargada de descargar e instalar automáticamente todos los plugins necesarios de su repositorio definidos en el pom.xml del proyecto para evitar que tengas que descargar e instalar todo tú de forma manual.

### 3. Descargar e instalar MySQL

De nuevo debes ir a la página oficial de MySQL y seleccionar la versión y el sistema operativo que te interesan: <https://dev.mysql.com/downloads/mysql/>

Una vez descargado el paquete procedemos a su ejecución para que de comienzo el asistente de instalación. Seleccionamos en Developer Default en el tipo de instalación y MySQL Server, WorkBench y Connectors en la opción de selección de productos.

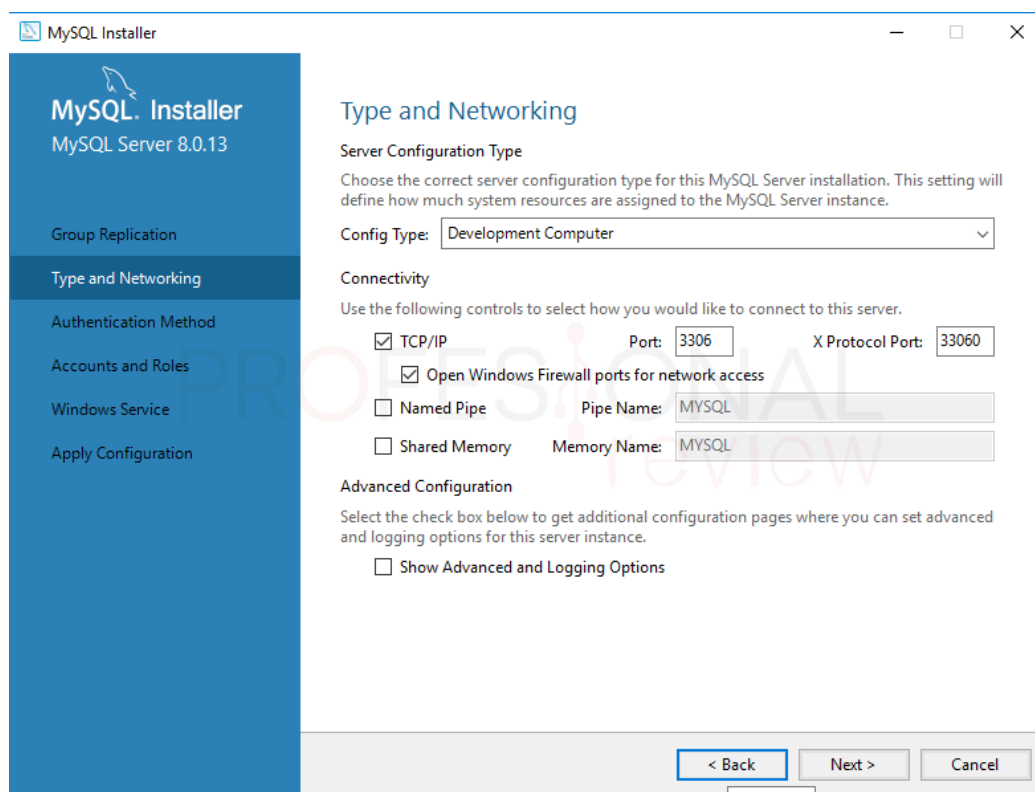
Pulsamos en “Next” y “Execute” y esperamos a que se instalen todos los módulos seleccionados.

Finalizada la instalación será turno de proceder a una configuración inicial antes de ejecutar los correspondientes servicios. Pulsamos en “Next” y elegimos la primera opción: “Standalone MySQL Server/Classic MySQL Replication”

Para la configuración de tipo de ordenador tendremos tres opciones distintas:

- **Development Computer:** Está orientado a ser un equipo en el que está instalado el servidor SQL, pero también el cliente para las consultas de bases de datos. Si nuestro equipo es doméstico y trabajamos de forma normal en él está será la opción que debemos elegir.
- **Server Computer:** esta segunda opción será orientada a ordenadores utilizados para funciones de servidor, por ejemplo, servidor web con bases de datos.
- **Dedicated Computer:** la tercera opción es par el caso en que queremos crear un equipo solo y exclusivamente orientado a bases de datos. Por ejemplo, una máquina virtual en la que se almacenen nuestras bases de datos.

La siguiente opción que tendremos que elegir es la del puerto TCP que utilizaremos para conexiones remotas. Por defecto es el 3306. La opción que marquemos aquí será el puerto que tendremos que abrir en nuestro router para establecer las conexiones remotas.



*Figura 19. Instalador MySQL. Manual de instalación[1]*

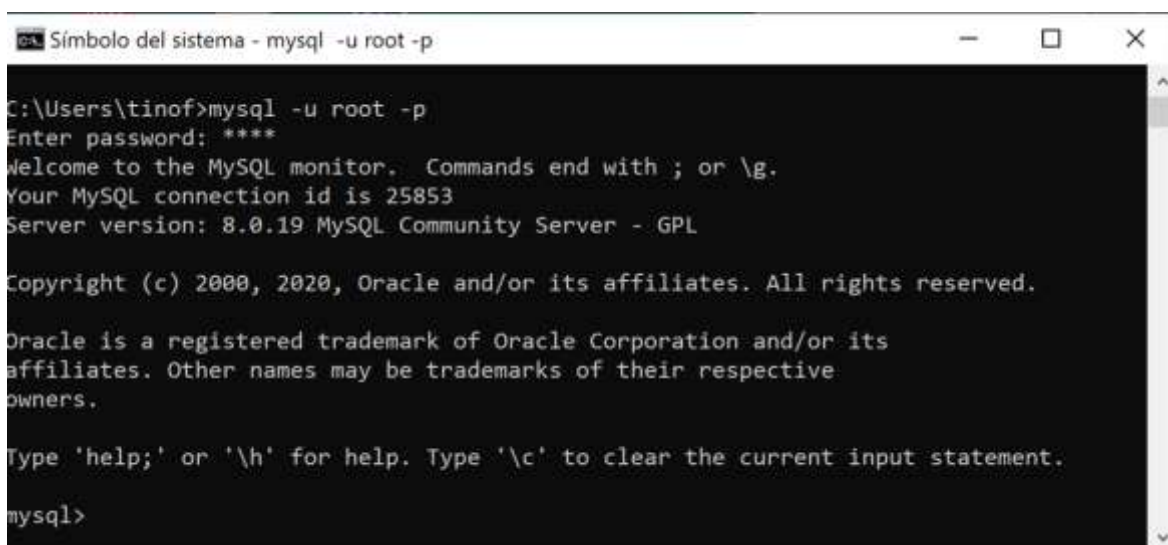
El resto de opciones mejor dejarlas por defecto.

A continuación debemos elegir la contraseña para conectarnos en el servidor SQL. Ojo, este paso es importante ya que usaremos este usuario para crear las tablas de datos y por lo tanto en la aplicación. Habrá que configurarlo en el fichero .properties de nuestro proyecto. Por defecto el usuario y la contraseña será root.

Para finalizar, en la última pantalla pulsamos en “execute” para ejecutar las acciones y activar los servicios correspondientes en el sistema.

#### 4. Configurar las bases de datos:

Entra en la terminal y escribe `mysql -u usuario -p`. Te pedirá tu contraseña introdúcela y ya podrás ejecutar comandos `mysql`.



```
Símbolo del sistema - mysql -u root -p
C:\Users\tinof>mysql -u root -p
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 25853
Server version: 8.0.19 MySQL Community Server - GPL

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Para crear la base de datos de la aplicación simplemente escribe: **CREATE DATABASE ingestmanagementdbtest.**

Una vez creada nuestra base de datos únicamente hay que ejecutar el fichero `.sql` con el siguiente contenido, que creará y rellenará todas las tablas de la aplicación para poder arrancar el programa desde 0.

```
DROP TABLE 'ingestmanagementdbtest'.'ejecucion';
DROP TABLE 'ingestmanagementdbtest'.'fichero';
DROP TABLE 'ingestmanagementdbtest'.'directorio';
DROP TABLE 'ingestmanagementdbtest'.'userproject';
DROP TABLE 'ingestmanagementdbtest'.'userprofile';
DROP TABLE 'ingestmanagementdbtest'.'profilepermit';
DROP TABLE 'ingestmanagementdbtest'.'project';
DROP TABLE 'ingestmanagementdbtest'.'user';
DROP TABLE 'ingestmanagementdbtest'.'permit';
DROP TABLE 'ingestmanagementdbtest'.'profile';
DROP TABLE 'ingestmanagementdbtest'.'environment';
```

```
CREATE TABLE 'user' (  
  'iduser' int NOT NULL AUTO_INCREMENT,  
  'name' varchar(40) NOT NULL,  
  'email' varchar(35) NOT NULL,  
  'password' varchar(30) NOT NULL,  
  'createDate' datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,  
  'deleteDate' datetime DEFAULT NULL,  
  PRIMARY KEY ('iduser')  
);  
  
CREATE TABLE 'environment' (  
  'idenvironment' int NOT NULL AUTO_INCREMENT,  
  'name' varchar(25) NOT NULL,  
  PRIMARY KEY ('idenvironment')  
);  
  
CREATE TABLE 'permit' (  
  'idpermit' int NOT NULL AUTO_INCREMENT,  
  'name' varchar(25) NOT NULL,  
  PRIMARY KEY ('idpermit')  
);  
  
CREATE TABLE 'profile' (  
  'idprofile' int NOT NULL AUTO_INCREMENT,  
  'tipo' varchar(15) NOT NULL,  
  'name' varchar(40) NOT NULL,  
  'idEnvironment' int DEFAULT NULL,  
  PRIMARY KEY ('idprofile'),  
  KEY 'fk_profile_to_environment_idx' ('idEnvironment'),  
  CONSTRAINT 'fk_profile_to_environment' FOREIGN KEY ('idEnvironment') REFERENCES  
'environment' ('idenvironment')  
);  
  
CREATE TABLE 'profilepermit' (  
  'idprofile' int NOT NULL,  
  'idpermit' int NOT NULL,  
  KEY 'fk_profile_to_permit_idx' ('idprofile'),  
  KEY 'fk_permit_to_profile_idx' ('idpermit'),  
  CONSTRAINT 'fk_permit_to_profile' FOREIGN KEY ('idpermit') REFERENCES 'permit'  
( 'idpermit'),  
  CONSTRAINT 'fk_profile_to_permit' FOREIGN KEY ('idprofile') REFERENCES  
'profile' ('idprofile')  
);  
  
CREATE TABLE 'userprofile' (  
  'iduser' int NOT NULL,  
  'idprofile' int NOT NULL,  
  KEY 'fk_user_idx' ('iduser'),  
  KEY 'fk_profile_idx' ('idprofile'),  
  CONSTRAINT 'fk_profile_to_user' FOREIGN KEY ('idprofile') REFERENCES 'profile'  
( 'idprofile'),  
  CONSTRAINT 'fk_user_to_profile' FOREIGN KEY ('iduser') REFERENCES 'user'  
( 'iduser')  
);
```



```
CREATE TABLE 'project' (  
  'idproject' int NOT NULL AUTO_INCREMENT,  
  'name' varchar(90) NOT NULL,  
  'idEnvironment' int DEFAULT NULL,  
  PRIMARY KEY ('idproject'),  
  KEY 'idEnvironment_idx' ('idEnvironment'),  
  CONSTRAINT 'fk_environment_to_project' FOREIGN KEY ('idEnvironment') REFERENCES  
'environment' ('idenvironment')  
);  
CREATE TABLE 'userproject' (  
  'iduser' int NOT NULL,  
  'idproject' int NOT NULL,  
  KEY 'pk_user_to_project_idx' ('iduser'),  
  KEY 'pk_project_to_user_idx' ('idproject'),  
  CONSTRAINT 'pk_project_to_user' FOREIGN KEY ('idproject') REFERENCES 'project'  
('idproject'),  
  CONSTRAINT 'pk_user_to_project' FOREIGN KEY ('iduser') REFERENCES 'user'  
('iduser')  
);  
CREATE TABLE 'directorio' (  
  'idDirectorio' int NOT NULL AUTO_INCREMENT,  
  'nombre' varchar(255) NOT NULL,  
  'idProyecto' int NOT NULL,  
  'parent_id' int DEFAULT NULL,  
  PRIMARY KEY ('idDirectorio'),  
  KEY 'parent_id' ('parent_id'),  
  KEY 'fk_proyectodirectorio_idx' ('idProyecto'),  
  CONSTRAINT 'directorio_ibfk_1' FOREIGN KEY ('parent_id') REFERENCES  
'directorio' ('idDirectorio') ON DELETE CASCADE ON UPDATE CASCADE,  
  CONSTRAINT 'fk_proyectodirectorio' FOREIGN KEY ('idProyecto') REFERENCES  
'project' ('idproject')  
);  
CREATE TABLE 'fichero' (  
  'idFichero' int NOT NULL AUTO_INCREMENT,  
  'nombre' varchar(45) NOT NULL,  
  'tipo' varchar(8) NOT NULL,  
  'hash' int NOT NULL,  
  'idDirectorio' int NOT NULL,  
  PRIMARY KEY ('idFichero'),  
  KEY 'fk_directorio_fichero_idx' ('idDirectorio'),  
  CONSTRAINT 'fk_directorio_fichero' FOREIGN KEY ('idDirectorio') REFERENCES  
'directorio' ('idDirectorio')  
);  
CREATE TABLE 'ejecucion' (  
  'idEjecucion' int NOT NULL AUTO_INCREMENT,  
  'idProyecto' int NOT NULL,  
  'nombreUsuario' varchar(40) NOT NULL,  
  'nombreFichero' varchar(45) NOT NULL,  
  'horaIni' datetime(3) NOT NULL,  
  'horaFin' datetime(3) NOT NULL,  
  'resultado' varchar(10) NOT NULL,  
  'parametros' varchar(45) NOT NULL,
```

```
'nombreLog' varchar(80) NOT NULL,  
'contenidoLog' longtext NOT NULL,  
'programada' varchar(5) NOT NULL,  
PRIMARY KEY ('idEjecucion'),  
KEY 'fk_proyecto_ejecucion_idx' ('idProyecto'),  
CONSTRAINT 'fk_proyecto_ejecucion' FOREIGN KEY ('idProyecto') REFERENCES  
'project' ('idproject')  
);  
-- Creamos un usuario para entrar en la aplicacion tabla USUARIOS  
INSERT INTO 'ingestmanagementdbtest'.'user' ('name', 'email', 'password') VALUES  
( 'Tino Fernandez Perez', 'tinofernandezperez@hotmail.com', '03121998');  
  
-- Rellenamos tabla ENTORNOS  
INSERT INTO 'ingestmanagementdbtest'.'environment' ('name') VALUES  
( 'DESARROLLO');  
INSERT INTO 'ingestmanagementdbtest'.'environment' ('name') VALUES  
( 'PREPRODUCCION');  
INSERT INTO 'ingestmanagementdbtest'.'environment' ('name') VALUES  
( 'PRODUCCION');  
-- Rellenamos table PERMISOS  
INSERT INTO 'ingestmanagementdbtest'.'permit' ('name') VALUES ('DESPLIEGUE');  
INSERT INTO 'ingestmanagementdbtest'.'permit' ('name') VALUES ('EJECUCION');  
INSERT INTO 'ingestmanagementdbtest'.'permit' ('name') VALUES ('PROGRAMACION');  
-- Rellenamos tabla PERFILES;  
INSERT INTO 'ingestmanagementdbtest'.'profile' ('tipo', 'name', 'idEnvironment')  
VALUES ('ADMINISTRADOR', 'ADMINISTRADOR DES', '1');  
INSERT INTO 'ingestmanagementdbtest'.'profile' ('tipo', 'name', 'idEnvironment')  
VALUES ('ADMINISTRADOR', 'ADMINISTRADOR PRE', '2');  
INSERT INTO 'ingestmanagementdbtest'.'profile' ('tipo', 'name', 'idEnvironment')  
VALUES ('ADMINISTRADOR', 'ADMINISTRADOR PRO', '3');  
INSERT INTO 'ingestmanagementdbtest'.'profile' ('tipo', 'name', 'idEnvironment')  
VALUES ('DESARROLLADOR', 'DESARROLLADOR DES', '1');  
INSERT INTO 'ingestmanagementdbtest'.'profile' ('tipo', 'name', 'idEnvironment')  
VALUES ('DESARROLLADOR', 'DESARROLLADOR PRE', '2');  
INSERT INTO 'ingestmanagementdbtest'.'profile' ('tipo', 'name', 'idEnvironment')  
VALUES ('DESARROLLADOR', 'DESARROLLADOR PRO', '3');  
-- Rellenamos tabla PERMISOS PERFILES  
INSERT INTO 'ingestmanagementdbtest'.'profilepermit' ('idProfile', 'idPermit')  
VALUES ('1', '1');  
INSERT INTO 'ingestmanagementdbtest'.'profilepermit' ('idProfile', 'idPermit')  
VALUES ('1', '2');  
INSERT INTO 'ingestmanagementdbtest'.'profilepermit' ('idProfile', 'idPermit')  
VALUES ('1', '3');  
INSERT INTO 'ingestmanagementdbtest'.'profilepermit' ('idProfile', 'idPermit')  
VALUES ('2', '1');  
INSERT INTO 'ingestmanagementdbtest'.'profilepermit' ('idProfile', 'idPermit')  
VALUES ('2', '2');  
INSERT INTO 'ingestmanagementdbtest'.'profilepermit' ('idProfile', 'idPermit')  
VALUES ('2', '3');  
INSERT INTO 'ingestmanagementdbtest'.'profilepermit' ('idProfile', 'idPermit')  
VALUES ('3', '1');  
INSERT INTO 'ingestmanagementdbtest'.'profilepermit' ('idProfile', 'idPermit')  
VALUES ('3', '2');
```

```
INSERT INTO 'ingestmanagementdbtest'.'profilepermit' ('idProfile', 'idPermit')
VALUES ('3', '3');
INSERT INTO 'ingestmanagementdbtest'.'profilepermit' ('idProfile', 'idPermit')
VALUES ('4', '1');
INSERT INTO 'ingestmanagementdbtest'.'profilepermit' ('idProfile', 'idPermit')
VALUES ('4', '2');
INSERT INTO 'ingestmanagementdbtest'.'profilepermit' ('idProfile', 'idPermit')
VALUES ('4', '3');
INSERT INTO 'ingestmanagementdbtest'.'profilepermit' ('idProfile', 'idPermit')
VALUES ('5', '3');
INSERT INTO 'ingestmanagementdbtest'.'profilepermit' ('idProfile', 'idPermit')
VALUES ('6', '3');
-- Rellenamos tabla USUARIOS PERFILES
INSERT INTO 'ingestmanagementdbtest'.'userprofile' ('idUser', 'idProfile') VALUES
('1', '1');
INSERT INTO 'ingestmanagementdbtest'.'userprofile' ('idUser', 'idProfile') VALUES
('1', '2');
INSERT INTO 'ingestmanagementdbtest'.'userprofile' ('idUser', 'idProfile') VALUES
('1', '3');
```

El único usuario creado es un usuario de prueba con nombre Tino Fernández Pérez, usuario [tfp@altia.es](mailto:tfp@altia.es) y contraseña “123456”. Deberás usar estas credenciales para autenticarte en la aplicación, y podrás modificar tu información una vez dentro.

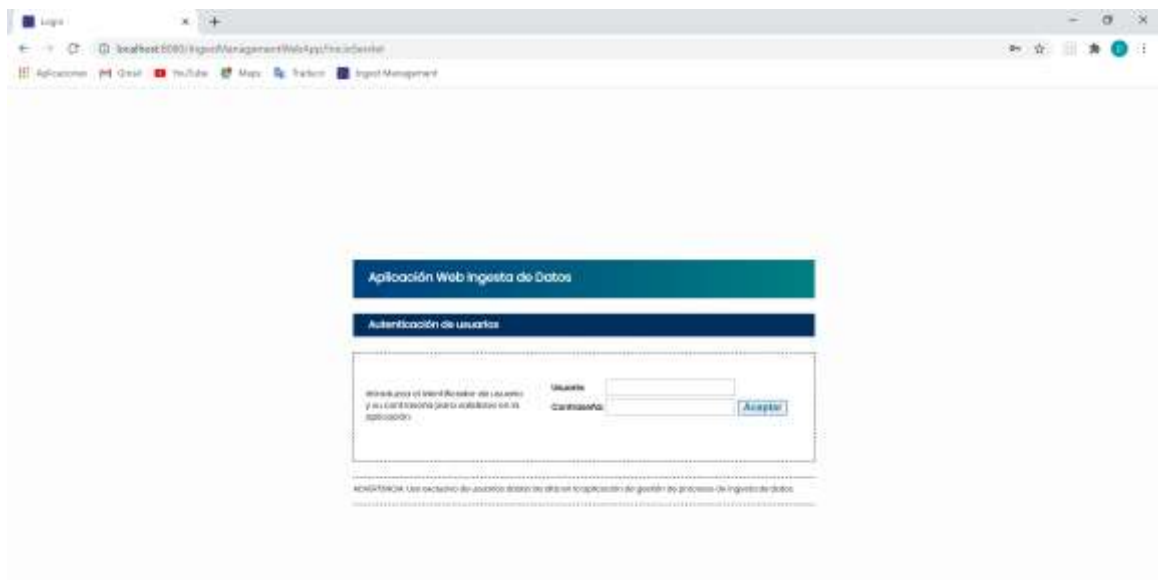
## **5. Configuración fichero .properties y ejecución aplicación:**

Descargas el proyecto en tu ordenador y abres el fichero app.properties dentro de la carpeta src/resource y completa los datos con tu usuario y contraseña de mysql (root) para acceder a las bases de datos, el nombre de la base de datos en datasource.database, y la dirección del servidor donde quieres que se genere toda la estructura de directorios de los proyectos, se suban los ficheros y se ejecuten. Ahora solo debes ir a la terminal de tu ordenador, entrar en el directorio del proyecto y ejecutar el comando **mvn clean jetty:run**. El programa descargará del repositorio de maven todas las librerías necesarias y se desplegará la aplicación. Entra en **localhost:8080/IngestManagementWebApp/** y ya podrás autenticarte y usar la aplicación.

## ANEXO II: MANUAL DE LA APLICACIÓN

### Autenticación en la aplicación:

La siguiente captura muestra la página de *login* del sistema de gestión de procesos de Ingesta de Datos. Para acceder a la aplicación deberás introducir tu correo electrónico en el cuadro de texto de “Usuario” y tu contraseña en “Contraseña” y pulsar “Aceptar”.



*Figura 20. Página Login WebApp*

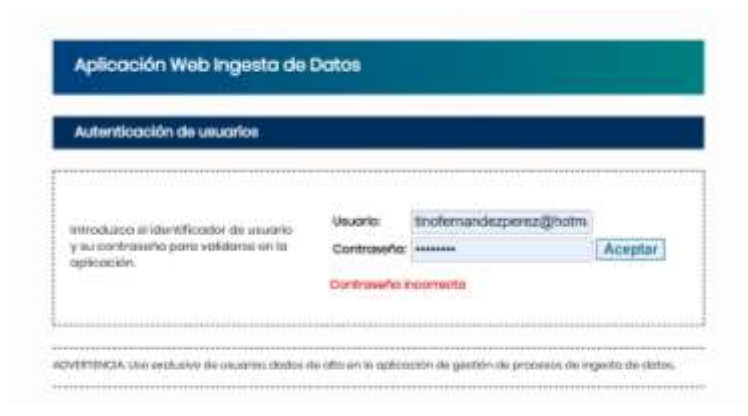
El sistema comprobará tus credenciales y:

1. Mostrará el mensaje “El usuario introducido no existe” sin acceder a la aplicación en caso de que no exista ningún usuario con el correo introducido.
2. Mostrará el mensaje “Contraseña incorrecta” sin acceder a la aplicación en caso de que el usuario introducido exista pero la contraseña introducida sea incorrecta.
3. Mostrará el mensaje “El usuario introducido ha sido dado de baja” sin acceder a la aplicación en caso de que el usuario introducido se haya dado de baja.
4. Accederá a la aplicación si el usuario y contraseña introducidos son correctos.

A continuación se muestran capturas con los mensajes de error de autenticación posibles:



*Figura 23. Mensaje Error autenticación[1]*



*Figura 22. Mensaje Error autenticación[2]*



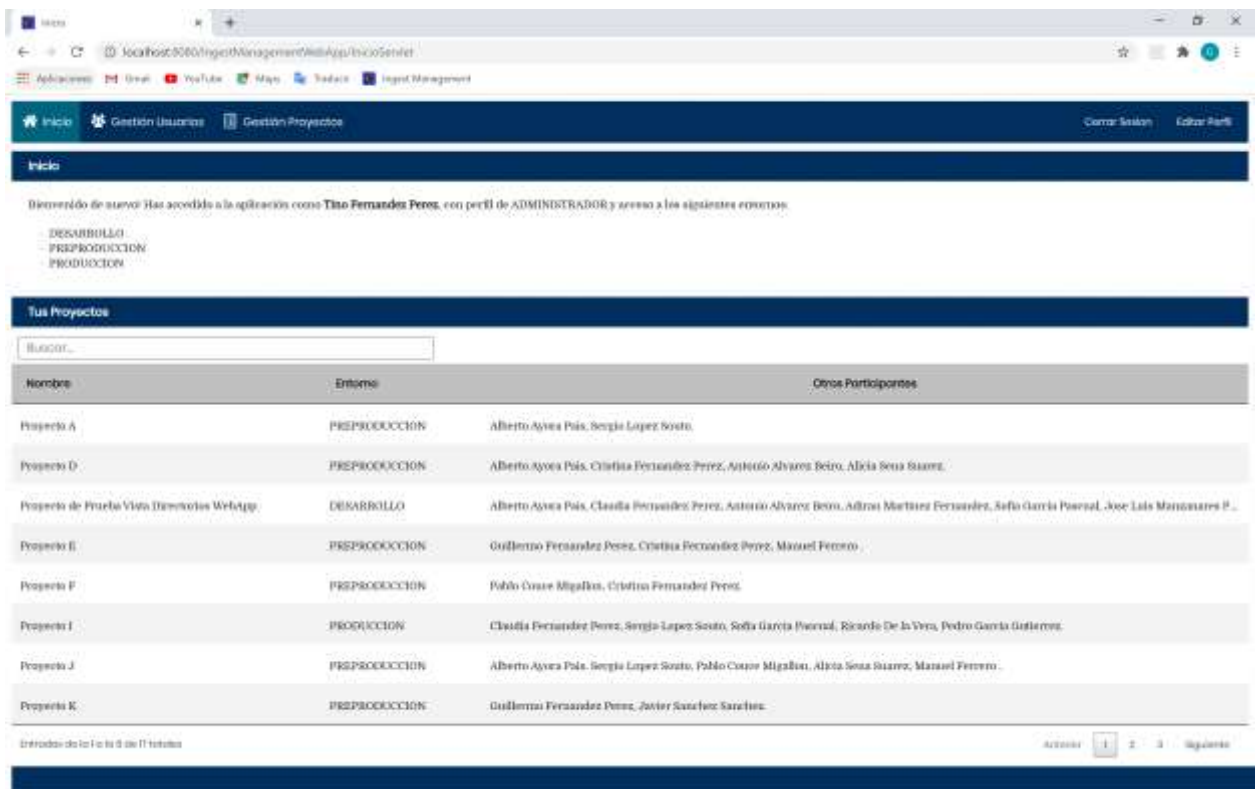
*Figura 21. Mensaje Error autenticación[3]*

Si la autenticación es correcta se abrirá la página de inicio de la aplicación.

## Página de Inicio

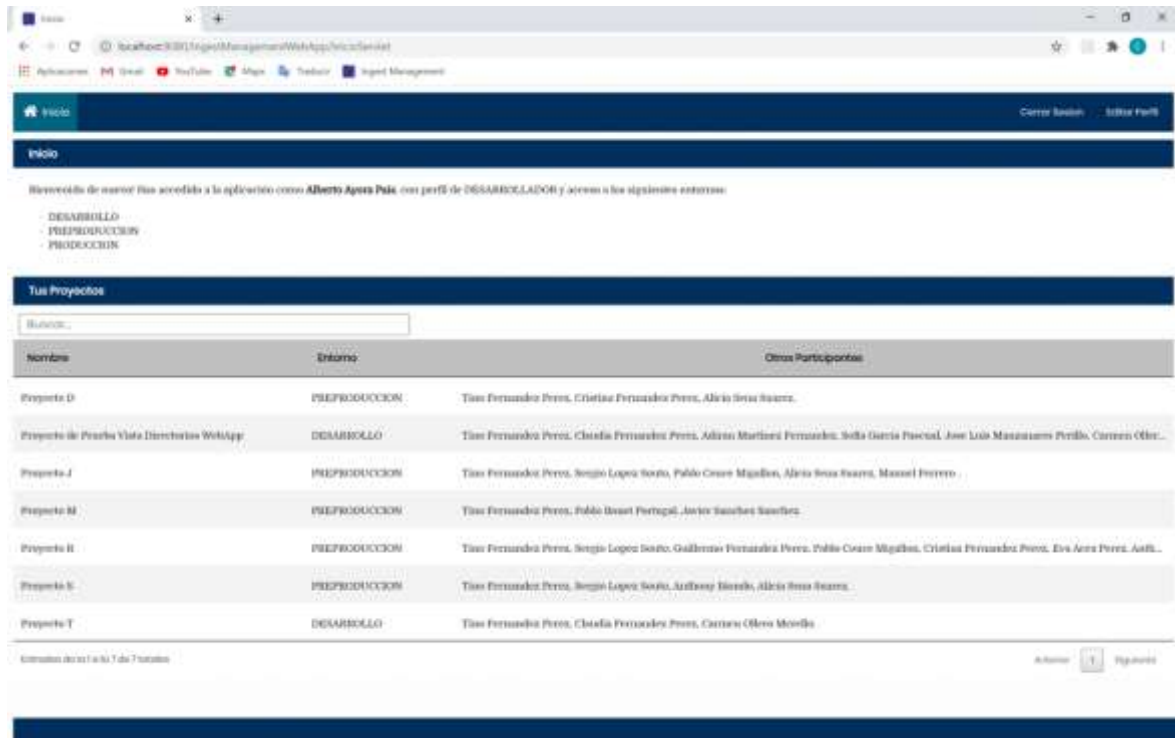
La página de Inicio será igual para todos los usuarios de la aplicación. Cuenta con un mensaje de bienvenida en el que se te informa de los entornos a los que tienes acceso, y la lista de proyectos que tienes asociados.

En la parte superior se encuentra el menú de navegación de la aplicación, que será distinto según tu tipo de perfil de usuario. Los administradores tendrán dos pestañas más que los desarrolladores: una para acceder a la gestión de usuarios, y otra para la gestión de proyectos. Además, todos los usuarios podrán pulsar en “Editar perfil” para ver sus propios datos y modificar su nombre y contraseña, y en “Cerrar Sesión” para salir de la aplicación.



Nombre	Entorno	Otros Participantes
Proyecto A	PREPRODUCCION	Alberto Ayora País, Sergio Lopez Sotelo
Proyecto D	PREPRODUCCION	Alberto Ayora País, Cristina Fernandez Perez, Antonio Alvarez Beiro, Alicia Souza Suarez
Proyecto de Prueba Vista Dimensiones WebApp	DESARROLLO	Alberto Ayora País, Claudia Fernandez Perez, Antonio Alvarez Beiro, Adrian Martinez Fernandez, Sofia Garcia Pascual, Jose Luis Manzanares P.
Proyecto E	PREPRODUCCION	Guillermo Fernandez Perez, Cristina Fernandez Perez, Manuel Ferrero
Proyecto F	PREPRODUCCION	Pablo Couso Migallon, Cristina Fernandez Perez
Proyecto I	PRODUCCION	Cristina Fernandez Perez, Sergio Lopez Sotelo, Sofia Garcia Pascual, Ricardo De la Vera, Pedro Garcia Gutierrez
Proyecto J	PREPRODUCCION	Alberto Ayora País, Sergio Lopez Sotelo, Pablo Couso Migallon, Alicia Souza Suarez, Manuel Ferrero
Proyecto K	PREPRODUCCION	Guillermo Fernandez Perez, Javier Sanchez Sanchez

Figura 24. Página Inicio Administrador WebApp



*Figura 25. Página Inicio Desarrollador WebApp*

La lista de proyectos es una tabla paginada que muestra todos los proyectos que tienes asociado como usuario de la aplicación. En caso de ser administrador mostrará todos los proyectos existentes, aunque no será aquí donde podrás crearlos, modificar su información, o eliminarlos, ya que esa gestión se realiza en la ventana de “Gestión de proyectos” a la que solo los administradores tienen acceso.

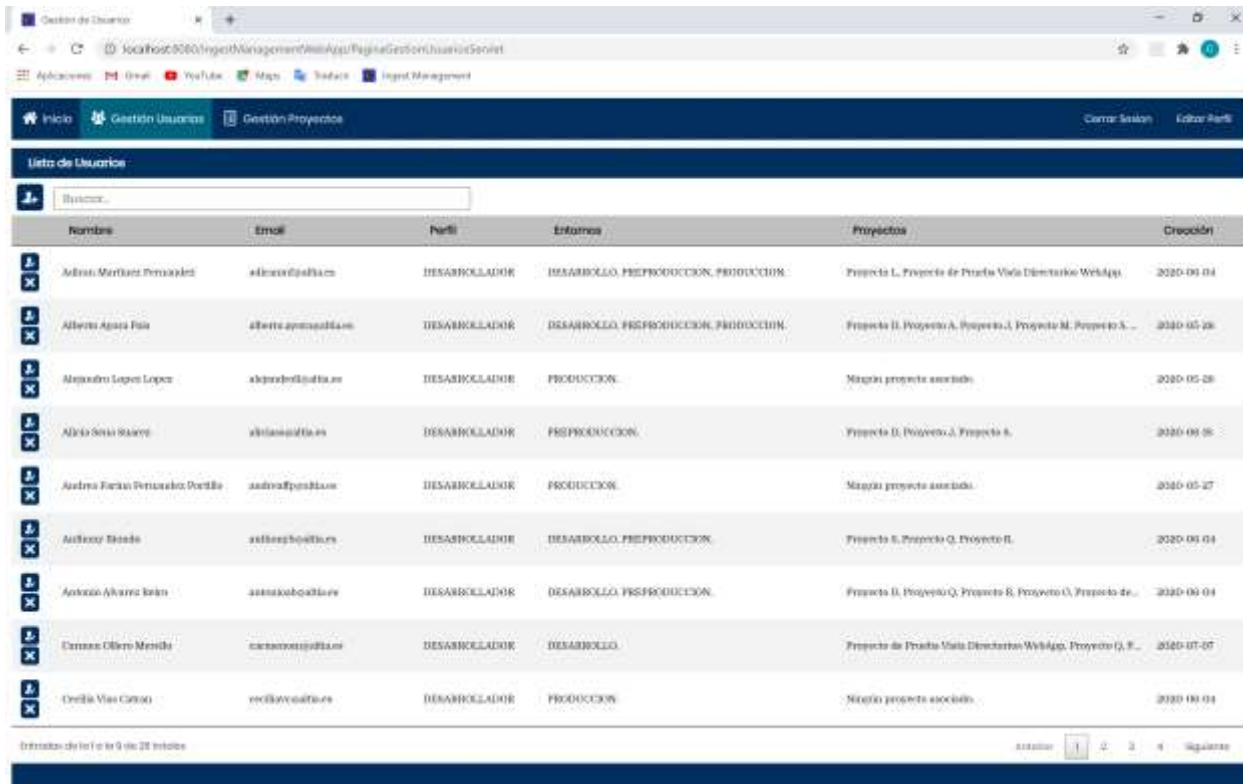
Como usuario con acceso al proyecto que sea podrás pulsar en él dentro de la tabla paginada y se abrirá en una nueva ventana, que explico más tarde en este mismo anexo.

Explicaré ahora las ventanas de Gestión de Usuarios y Gestión de proyectos, a las que recordamos que únicamente los usuarios con permiso de administrador tienen acceso.

### Página de Gestión de Usuarios (Solo administradores)

La página de Gestión de Usuarios permite a los usuarios administradores la gestión de todos los usuarios de la aplicación. Muestra una tabla paginada con todos los usuarios de la aplicación y su información más relevante: nombre y apellidos, correo electrónico, tipo de perfil, entornos a los que tiene acceso, proyectos asociados, y fecha de creación. Por defecto vienen ordenados por nombre, aunque podrán ordenarse también por email, perfil y fecha de creación pulsando en la fila con el título de la columna por la que desees que se ordenen. La tabla tiene un buscador que permite filtrar los resultados por cualquiera de los campos del usuario.

Para cada fila de la lista existen dos botones, uno para editar la información del usuario y otro para darlo de baja. En la esquina superior izquierda de la tabla además existe un botón con un icono de usuario y un más para crear nuevos usuarios.

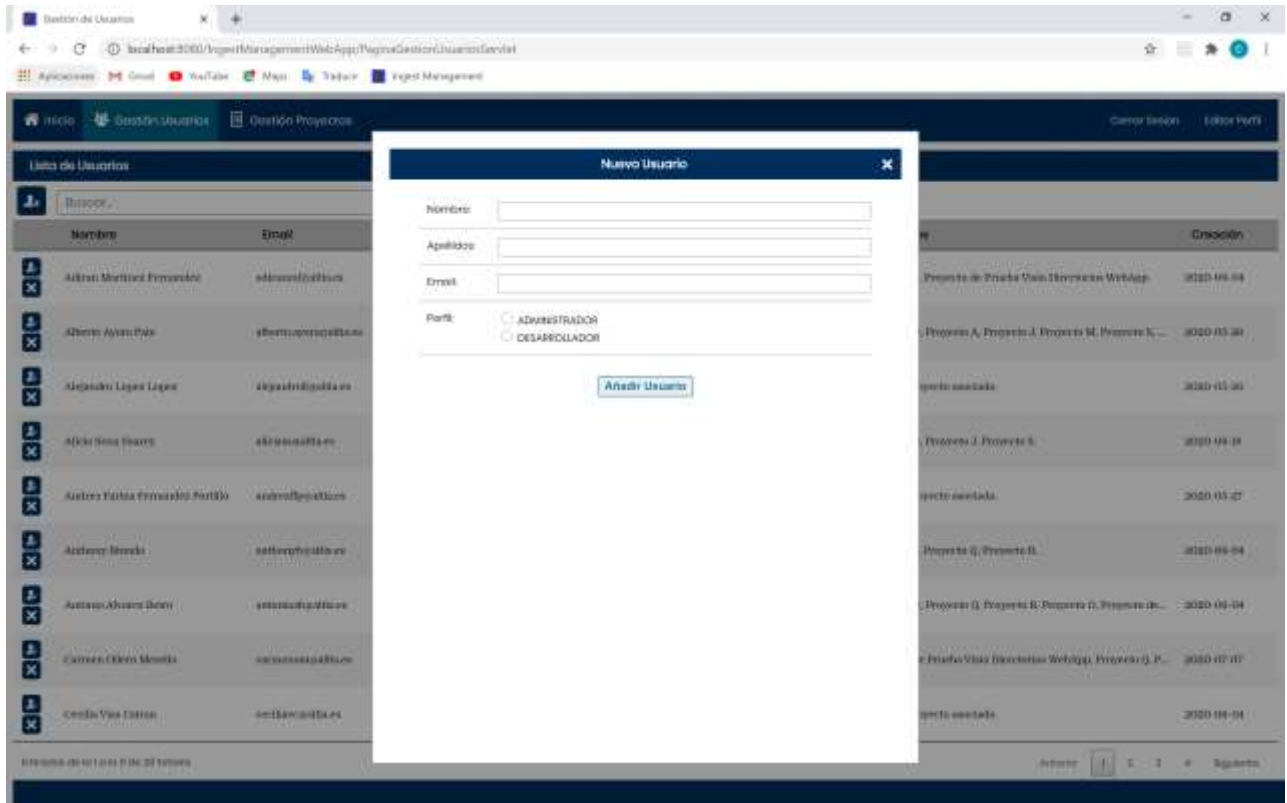


Nombre	Email	Perfil	Entornos	Proyectos	Creación
Andrés Martínez Peralpín	andres@comillas.es	DESARROLLADOR	DESARROLLO, PREPRODUCCION, PRODUCCION	Proyecto L, Proyecto de Prueba Vista Directorio WebApp	2020-09-04
Alberto Aguirre Pina	alberto@comillas.es	DESARROLLADOR	DESARROLLO, PREPRODUCCION, PRODUCCION	Proyecto H, Proyecto A, Proyecto J, Proyecto M, Proyecto S...	2020-05-26
Alejandro Lopez Lopez	alejandro@comillas.es	DESARROLLADOR	PRODUCCION	Ningún proyecto asociado	2020-05-26
Alicia Sierra Suarez	alicia@comillas.es	DESARROLLADOR	PREPRODUCCION	Proyecto D, Proyecto A, Proyecto S...	2020-09-05
Andrea Ferrás Peralpín	andrea@comillas.es	DESARROLLADOR	PRODUCCION	Ningún proyecto asociado	2020-05-27
Antonio Bernal	antonio@comillas.es	DESARROLLADOR	DESARROLLO, PREPRODUCCION	Proyecto S, Proyecto Q, Proyecto R...	2020-09-04
Antonio Alvarez Jimen	antonio@comillas.es	DESARROLLADOR	DESARROLLO, PREPRODUCCION	Proyecto D, Proyecto Q, Proyecto R, Proyecto U, Proyecto de...	2020-09-04
Carmen Otero Merida	carmen@comillas.es	DESARROLLADOR	DESARROLLO	Proyecto de Prueba Vista Directorio WebApp, Proyecto G, F...	2020-07-07
Cecilia Vico Castro	cecilia@comillas.es	DESARROLLADOR	PRODUCCION	Ningún proyecto asociado	2020-09-04

Figura 26. Página Gestión de Usuarios WebApp



Pulsando en el botón para añadir usuarios se abrirá una ventana que te permitirá crear un nuevo usuario introduciendo su nombre, apellidos, email y tipo de perfil:

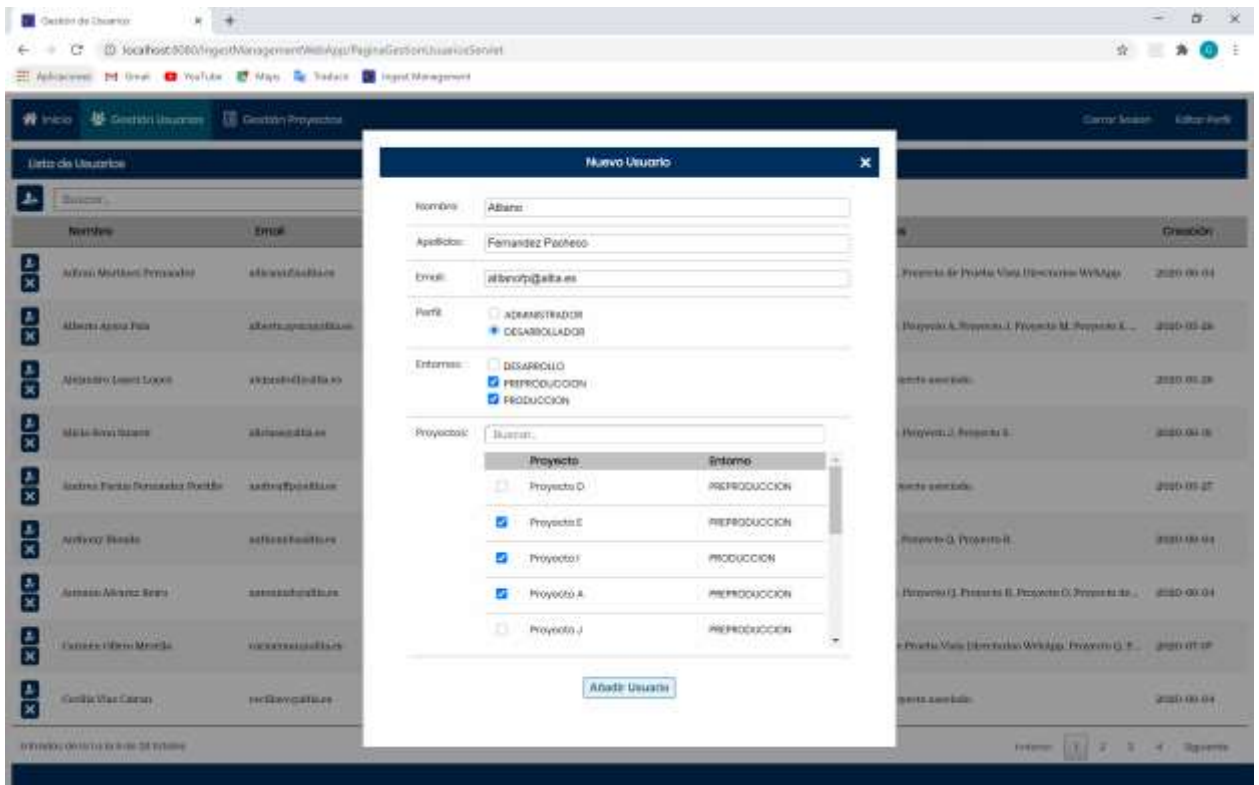


*Figura 27. Ventana Nuevo Usuario WebApp[1]*

Si el usuario a crear es administrador, no se solicitará más información, ya que se le asignarán automáticamente todos los entornos y proyectos de la aplicación, pero si el usuario a crear es de tipo desarrollador deberás introducir además al menos un entorno asociado, y según el o los entornos que selecciones te mostrará los proyectos disponibles con la posibilidad de asociárselos al usuario.

Completando la información del nuevo usuario y pulsando en Añadir Usuario se creará el nuevo usuario y te devolverá a la página de Gestión de Usuarios. El nombre, los apellidos, el email y el perfil son campos obligatorios, por lo que si no introduces alguno de ellos la aplicación no te avisará con el correspondiente mensaje. Si tratas de añadir un nuevo

usuario con perfil de desarrollador y no has seleccionado ningún entorno la aplicación informará de que “Debes seleccionar al menos un entorno”.

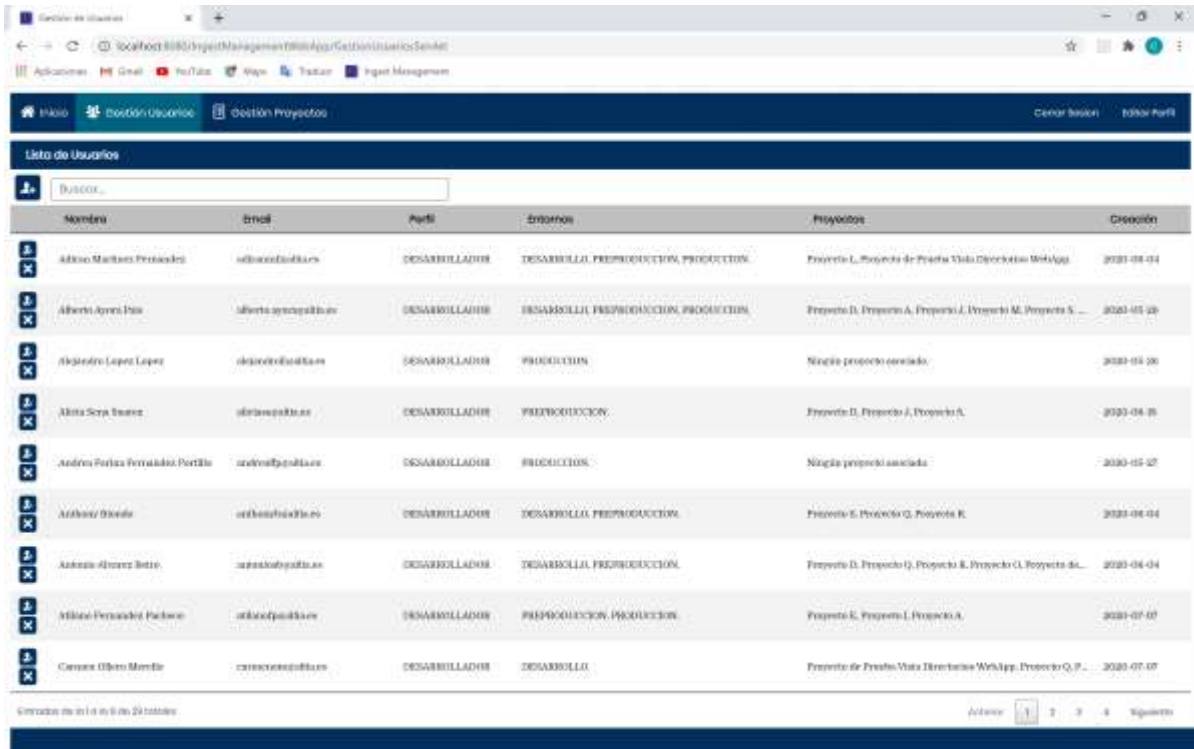


*Figura 28. Ventana Nuevo Usuario WebApp[2]*

Si toda la información introducida es válida, se creará el nuevo usuario con una contraseña aleatoria de 7 dígitos que se mostrará en pantalla en un mensaje de alerta. Esta contraseña no podrá revisarse de nuevo (únicamente accediendo a base de datos directamente) por lo que conviene anotarla.



*Figura 29. Mensaje confirmación Nuevo Usuario WebApp*



Nombre	Email	Papel	Departamento	Proyectos	Creación
Adrián Martínez Fernández	adrianm@comillas.es	DESARROLLADOR	DESARROLLO, PREPRODUCCIÓN, PRODUCCIÓN	Proyecto L, Proyecto de Prueba Vista Directivas WebApp	2020-08-04
Alberto Ayres Díaz	albertoayres@comillas.es	DESARROLLADOR	DESARROLLO, PREPRODUCCIÓN, PRODUCCIÓN	Proyecto D, Proyecto A, Proyecto J, Proyecto M, Proyecto S...	2020-05-28
Alejandro López López	alejandrol@comillas.es	DESARROLLADOR	PRODUCCIÓN	Ningún proyecto asociado	2020-05-28
Alina Sora Suarez	alinasora@comillas.es	DESARROLLADOR	PREPRODUCCIÓN	Proyecto D, Proyecto J, Proyecto A	2020-04-01
Andrés Perlas Fernández Portillo	andresp@comillas.es	DESARROLLADOR	PRODUCCIÓN	Ningún proyecto asociado	2020-05-07
Antonio García	antonio@comillas.es	DESARROLLADOR	DESARROLLO, PREPRODUCCIÓN	Proyecto E, Proyecto G, Proyecto K	2020-04-04
Antonio Álvarez Nieto	antonioa@comillas.es	DESARROLLADOR	DESARROLLO, PREPRODUCCIÓN	Proyecto D, Proyecto J, Proyecto K, Proyecto G, Proyecto de...	2020-04-04
Alfonso Fernández Pacheco	alfonso@comillas.es	DESARROLLADOR	PREPRODUCCIÓN, PRODUCCIÓN	Proyecto E, Proyecto L, Proyecto A	2020-07-07
Carmen Olvera Mendiz	carmenol@comillas.es	DESARROLLADOR	DESARROLLO	Proyecto de Prueba Vista Directivas WebApp, Proyecto Q, P...	2020-07-07

*Figura 30. Usuario creado WebApp*

Para editar un usuario de la aplicación pulsas en el icono con el usuario y el lápiz de la fila del usuario que deseas editar y te abrirá una página con el mismo aspecto que la página de nuevo usuario y la información del usuario completada. Modifica lo que deseas y pulsa “Editar Usuario”. Pulsando en la cruz o fuera de la ventana se cerrará automáticamente sin guardar la información del usuario.

Para dar de baja un usuario deberás pulsar en la cruz de debajo del botón de editar usuario en la fila del usuario que deseas dar de baja. Dar de baja un usuario no borrará su información de base de datos, simplemente dejará de mostrarse su información en la aplicación y se le bloqueará el acceso a esta. Un usuario dado de baja recibiría el mensaje de “Usuario dado de baja” si tratase de autenticarse. Al pulsar en la cruz se abrirá una ventana para confirmar la baja del usuario, y pulsando en “Dar de Baja” se dará de baja al usuario y se volverá a la página principal de Gestión de Usuarios.

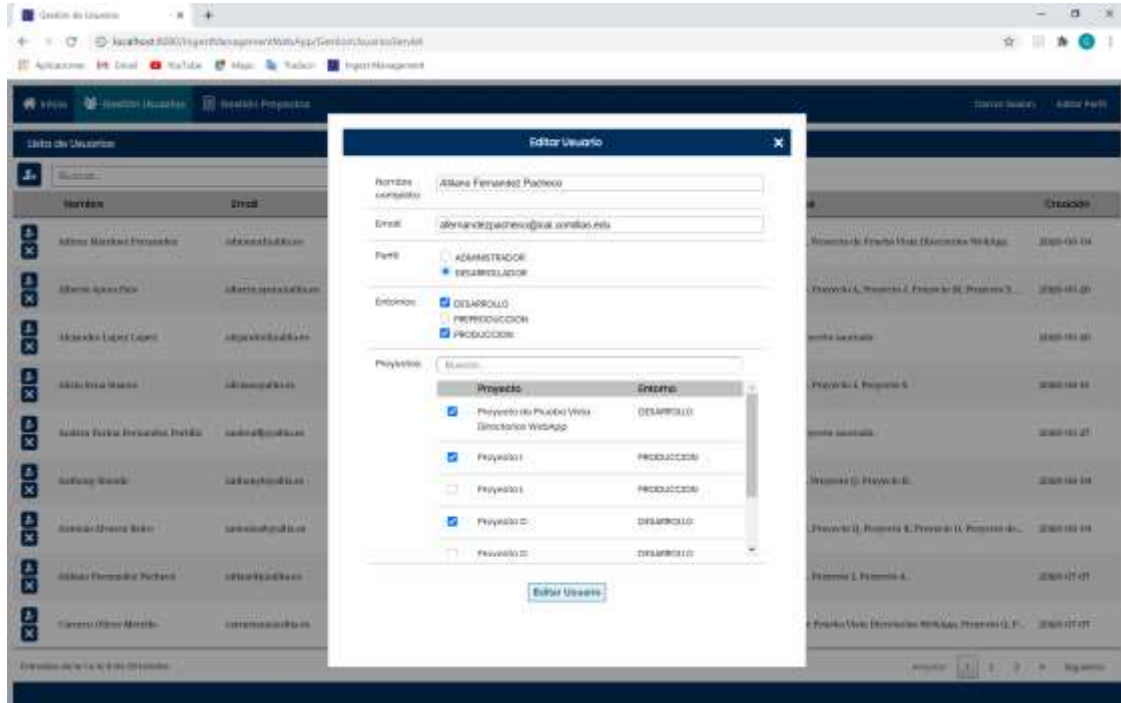


Figura 32. Ventana Editar Usuario WebApp

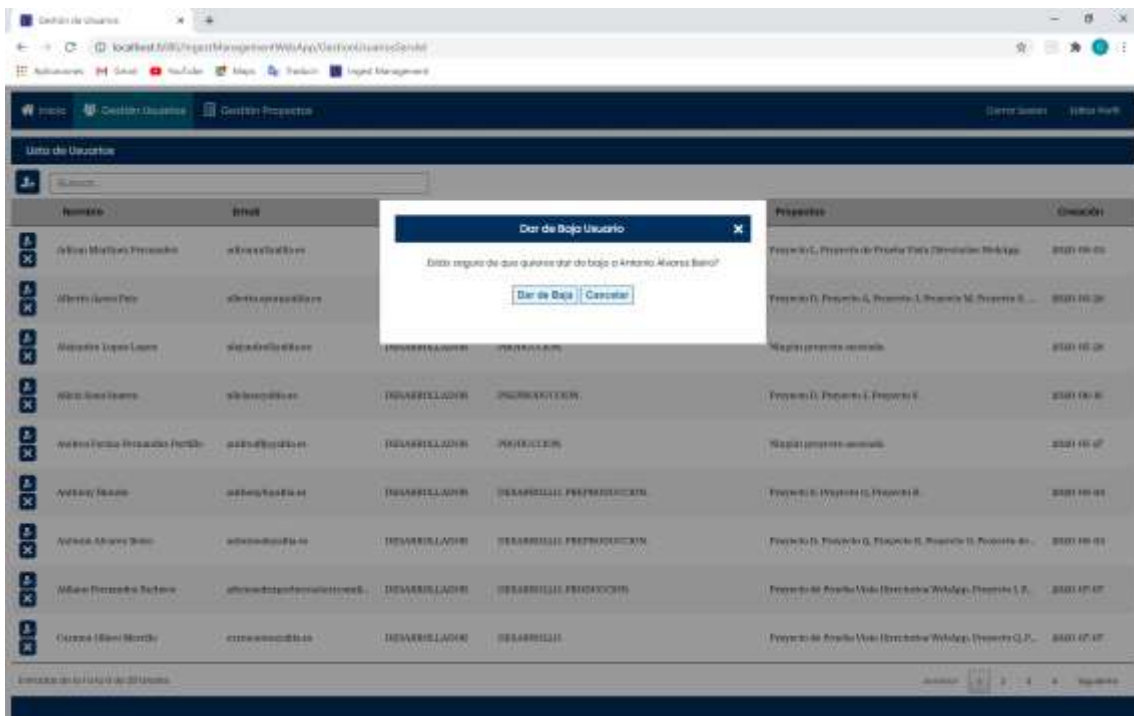
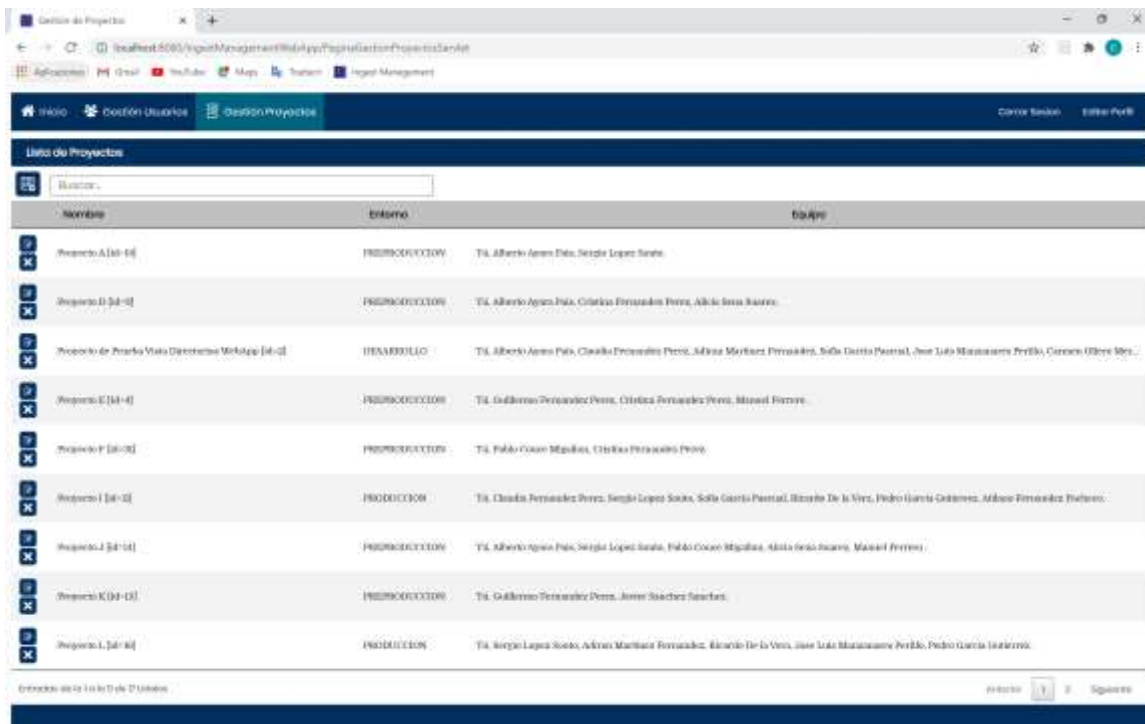


Figura 31. Ventana Dar de Baja Usuario WebApp

### Página de Gestión de Proyectos (Solo administradores)

La página de Gestión de Proyectos permite a los usuarios administradores la gestión de todos los proyectos de la aplicación. Muestra una tabla paginada con todos los proyectos de la aplicación y su información más relevante: nombre, entorno asociado, y participantes que a él tienen acceso.

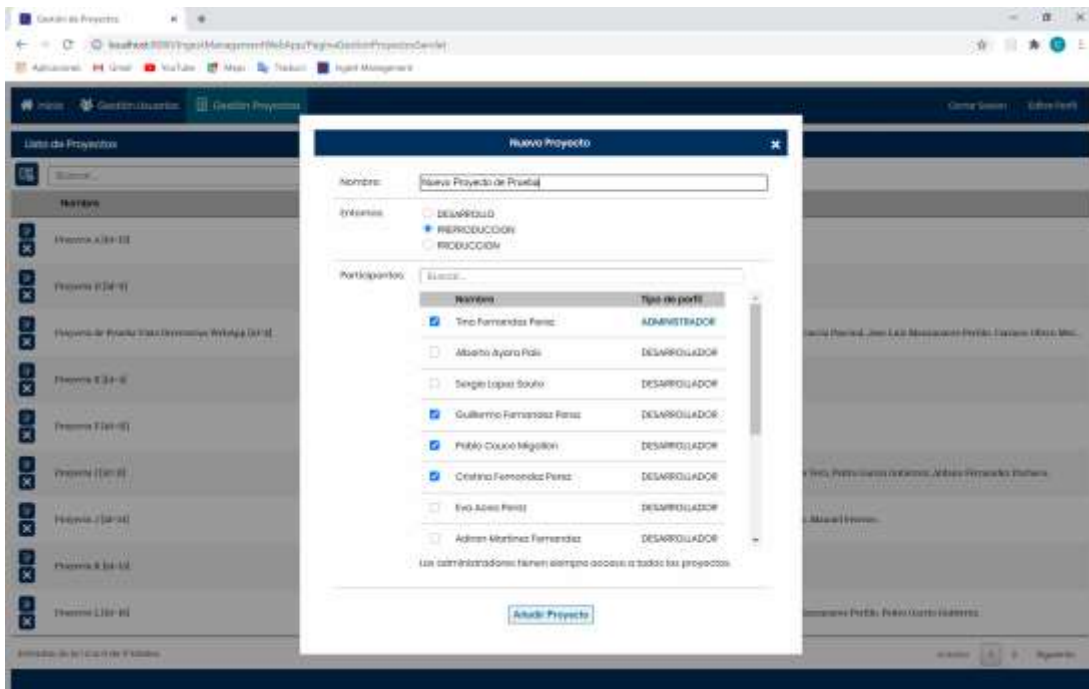
Para cada fila de la lista existen dos botones, uno para editar la información del proyecto y otro para eliminarlo. En la esquina superior izquierda de la tabla además existe un botón con un icono de un proyecto y un más para crear nuevos proyectos.



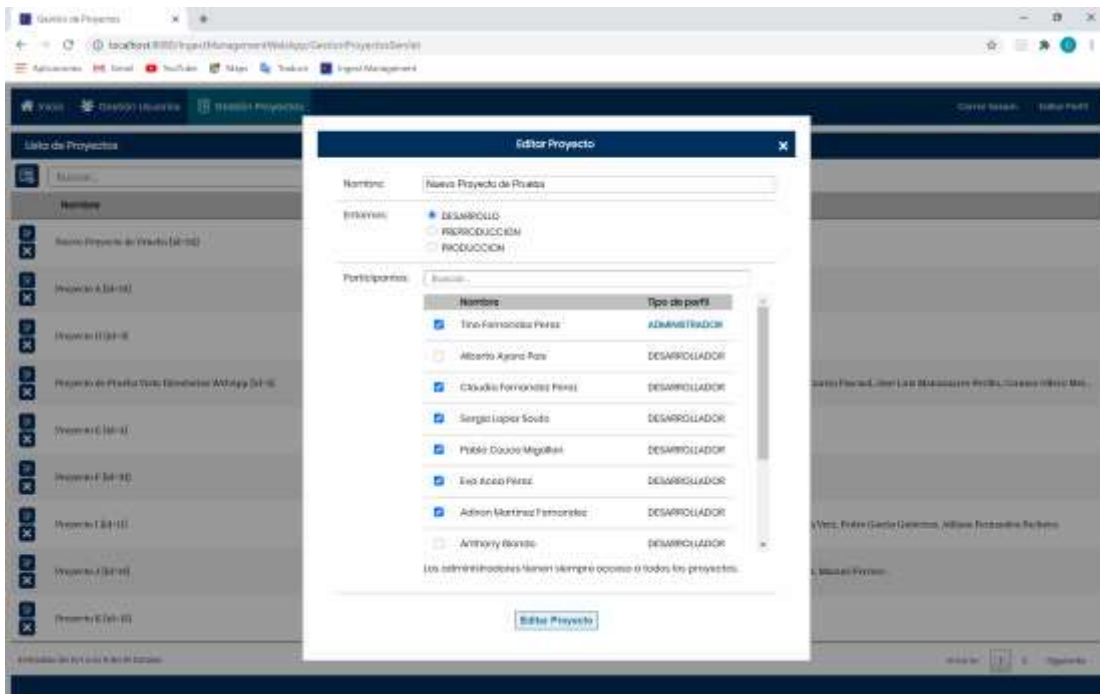
Nombre	Entorno	Equipo
Proyecto A [10-10]	DESARROLLO	Tt. Alberto Aguirre Peña, Sergio Lopez Sotelo
Proyecto B [10-10]	DESARROLLO	Tt. Alberto Aguirre Peña, Carolina Fernandez Perez, Alicia Irujo Izquierdo
Proyecto de Prueba Vista Directiva WebApp [10-10]	DESARROLLO	Tt. Alberto Aguirre Peña, Carolina Fernandez Perez, Miriam Morillas Peralta, Sofia Garcia Pascual, Jose Luis Marañón Perillo, Carmen Olabe Moya
Proyecto E [10-10]	DESARROLLO	Tt. Guillermo Fernandez Perez, Cristina Fernandez Perez, Manuel Ferrer
Proyecto F [10-10]	DESARROLLO	Tt. Pablo Cosco Migallan, Cristina Peralta Perez
Proyecto I [10-10]	PRODUCCION	Tt. Claudia Fernandez Perez, Sergio Lopez Sotelo, Sofia Garcia Pascual, Ricardo De la Veta, Pedro Garcia Gutierrez, Andres Fernandez Pacheco
Proyecto J [10-10]	DESARROLLO	Tt. Alberto Aguirre Peña, Sergio Lopez Sotelo, Pablo Cosco Migallan, Alicia Irujo Izquierdo, Manuel Ferrer
Proyecto K [10-10]	DESARROLLO	Tt. Guillermo Fernandez Perez, Javier Sánchez Sancha
Proyecto L [10-10]	PRODUCCION	Tt. Sergio Lopez Sotelo, Adrian Martinez Fernandez, Ricardo De la Veta, Jose Luis Marañón Perillo, Pedro Garcia Gutierrez

*Figura 33. Página Gestión de Proyectos WebApp*

Como la gestión de proyectos se realiza de forma parecida a la de usuarios mostraré capturas de cada ventana: nuevo proyecto, editar proyecto y confirmar eliminación de proyecto junto con alguna explicación que difiera de la de Gestión de Usuarios.

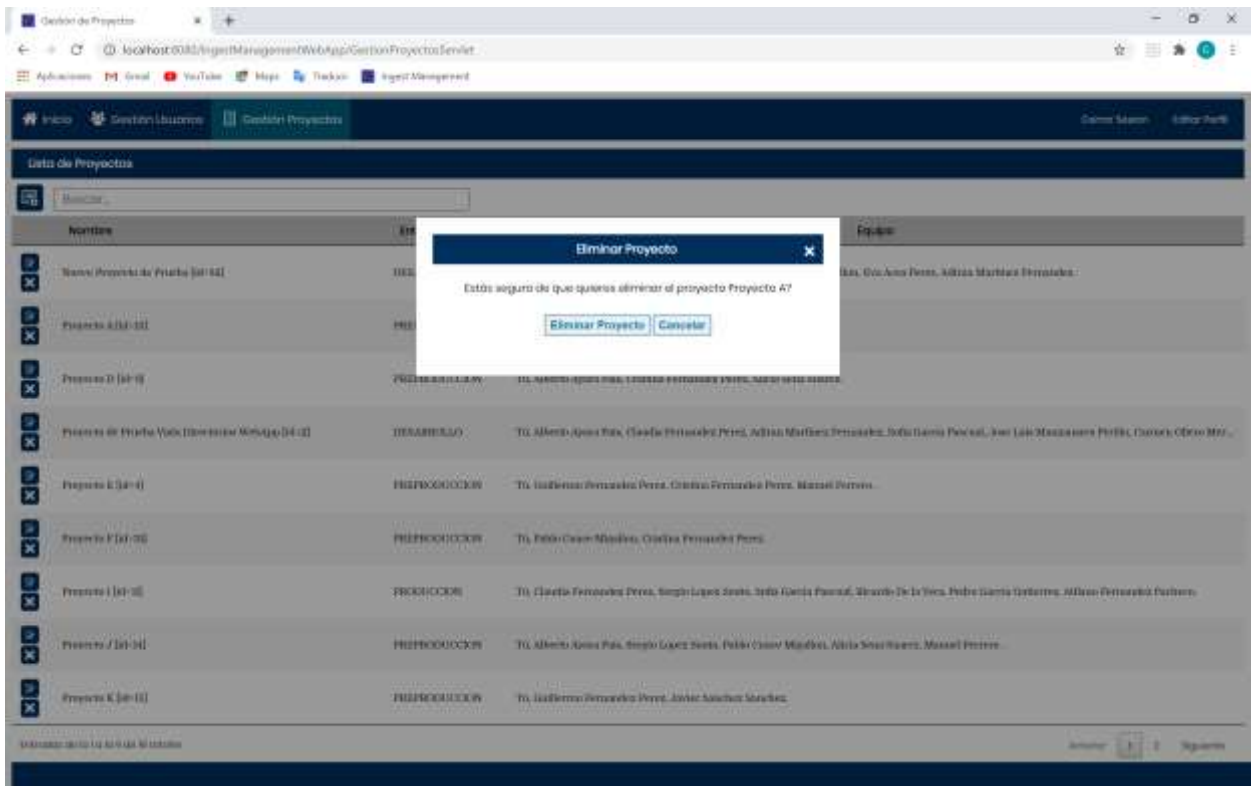


*Figura 35. Ventana Nuevo Proyecto WebApp*



*Figura 34. Ventana Editar Proyecto WebApp*

Los usuarios administradores siempre estará seleccionados como participantes de todos los proyectos que se vayan a crear (y no podrán deseleccionarse).



*Figura 36. Ventana Confirmar Eliminación de Proyecto*

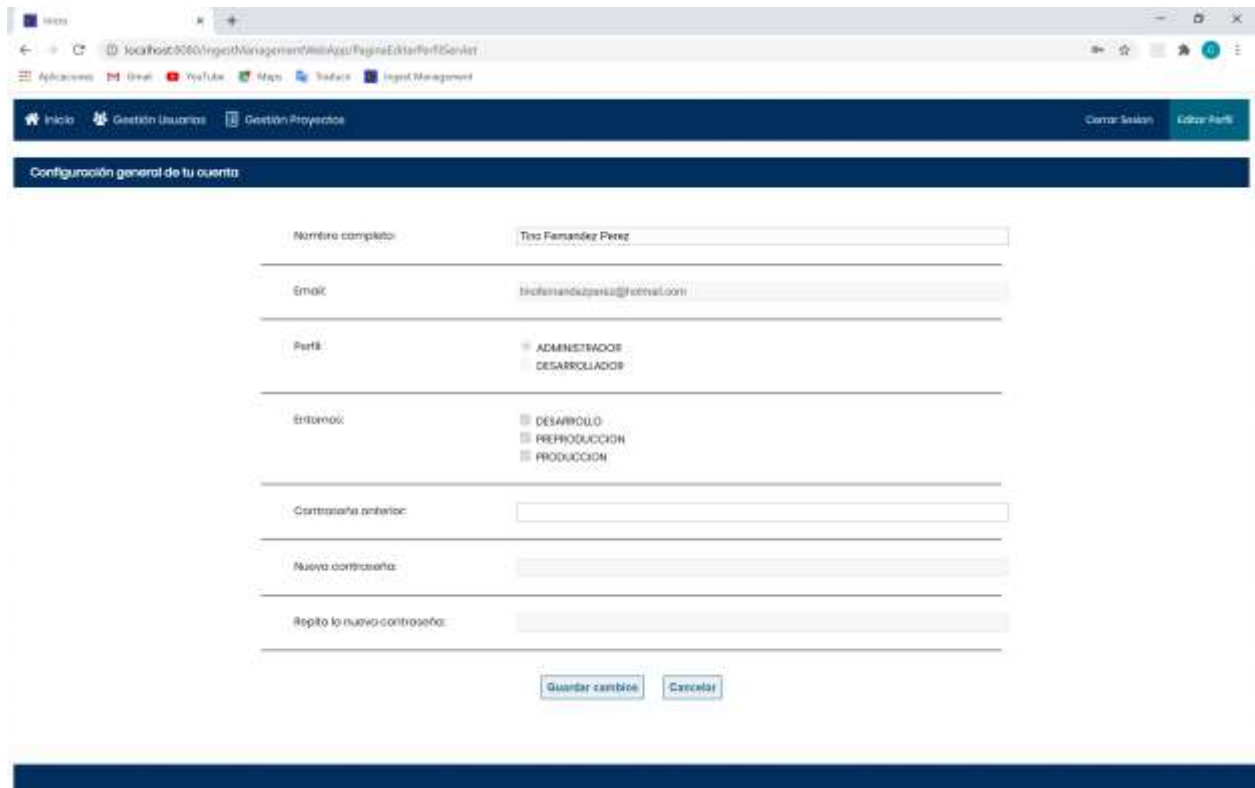
La eliminación de un proyecto conlleva la eliminación de toda la información que contiene. Esto incluye: árbol de directorios, ficheros y ejecuciones pasadas.

### **Página de Editar Perfil**

Esta página te permite ver la información de tu perfil y editar tu nombre y tu contraseña. Se decidió no poder modificar tu usuario desde aquí. En caso de querer hacerlo deberás accederlo un usuario con perfil de administrador desde la página de gestión de usuarios.

Para modificar tu contraseña deberás introducir tu contraseña anterior correctamente. En caso de que no sea así no podrás modificarla. Además, para asegurar que no se comete un error al elegir la nueva contraseña deberás introducirla dos veces, y en caso de que la hayas introducido mal la aplicación te avisará con el mensaje “Debes introducir la nueva contraseña”. Pulsando en “Guardar Cambios” se guardarán los cambios que hayas establecido y pulsando en cancelar se resetearán los datos a los que tenía previamente.





*Figura 37. Ventana Editar Perfil WebApp*

Volviendo a la página de Inicio se explica ahora como será la página para poder gestionar un proyecto. Para abrir cualquier proyecto como ya dijimos deberás tener acceso a él y seleccionarlo de la tabla paginada de Inicio.

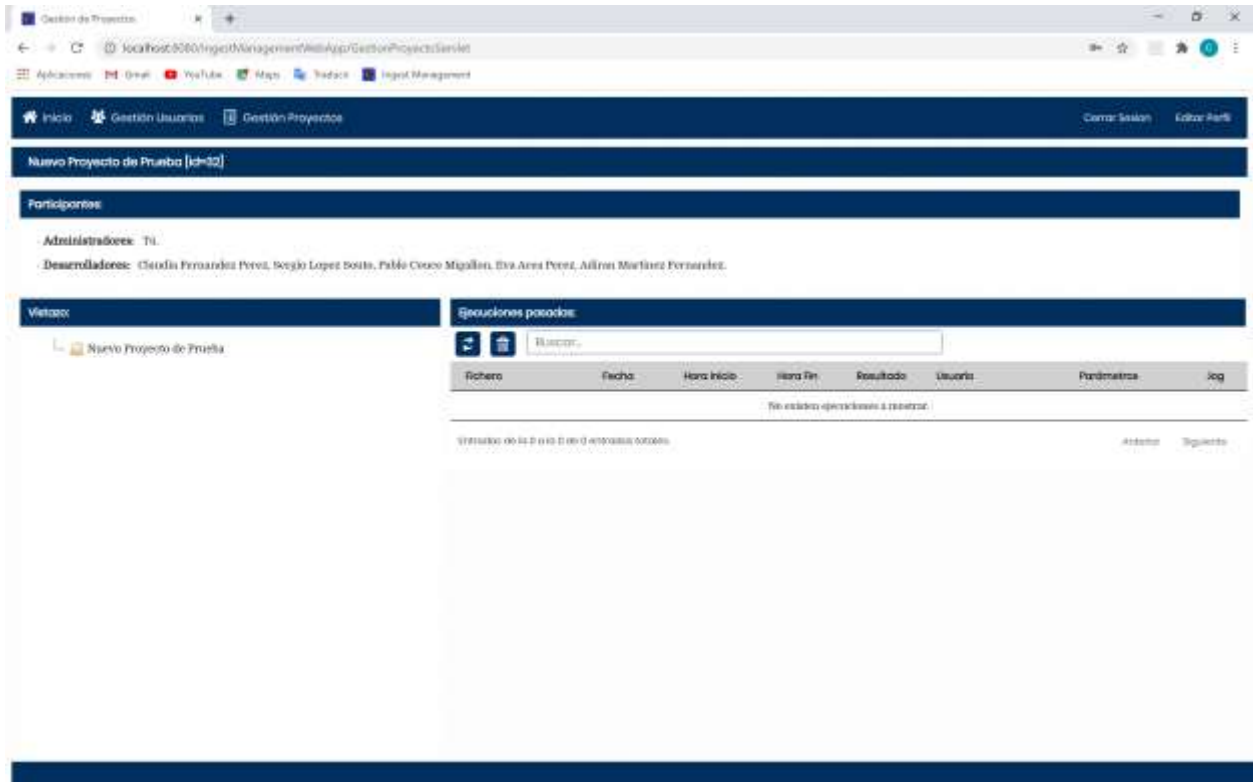
### **Página de Proyecto**

En esta página podrás (Según los permisos que tengas), crear y modificar la estructura de directorios del proyecto, subir ficheros y modificarlos, ejecutar los .jar o programar sus ejecuciones, así como ver la información de las ejecuciones pasadas y descargar los ficheros .log con sus trazas de ejecución.

Los permisos necesarios para realizar cualquiera de las acciones en los proyectos ya se han explicado a lo largo de la memoria, por lo que se mostrarán las acciones utilizando un



perfil de administrador que tiene acceso completos y por lo tanto puede realizar todas las acciones dentro de los proyectos.



*Figura 38. Ventana Proyecto de Prueba[1]*

El proyecto utilizado es un proyecto de prueba vacío:

El directorio raíz siempre estará creado para cualquier proyecto, y no podrá modificarse su nombre ya que tendrá siempre el nombre del proyecto al que pertenece. La única opción que tendremos al hacer click derecho en el directorio será la de crear un nuevo directorio.

Pasaré a crear un árbol de directorios de prueba para ver como se vería. Este árbol de prueba tendrá dos directorios dentro del directorio raíz, y otros dos directorios hijo dentro de cada directorio. En total serán 6 directorios más el directorio raíz.

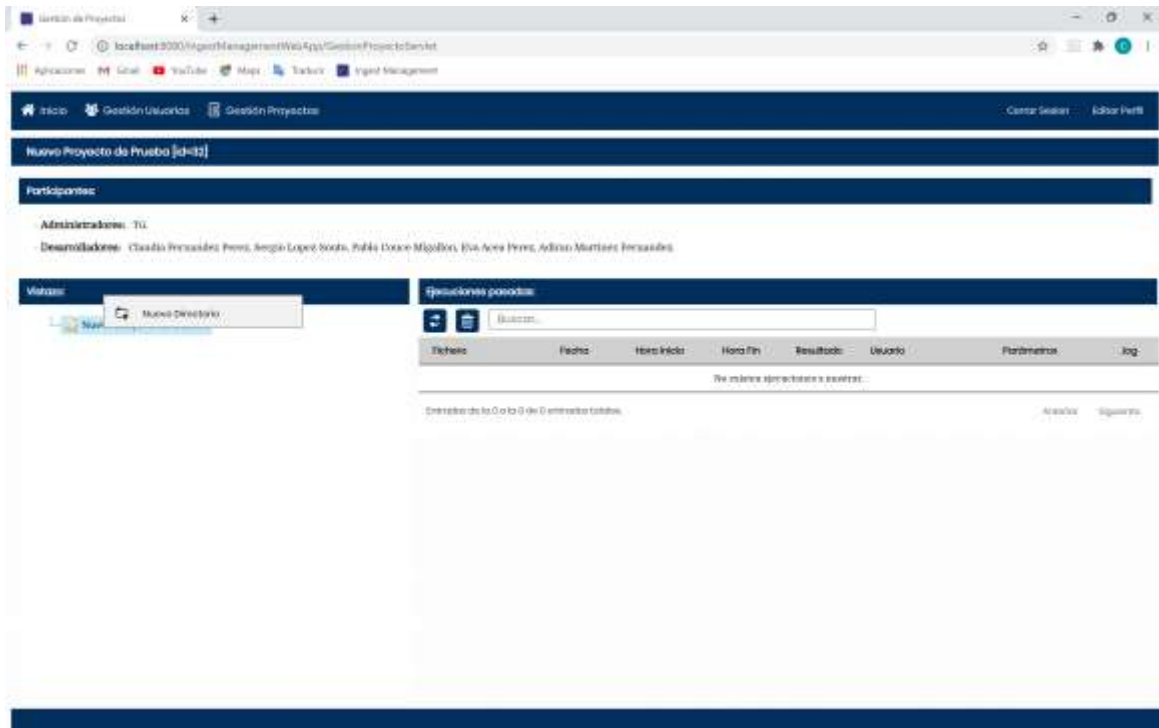


Figura 40. Ventana Proyecto de Prueba[2]

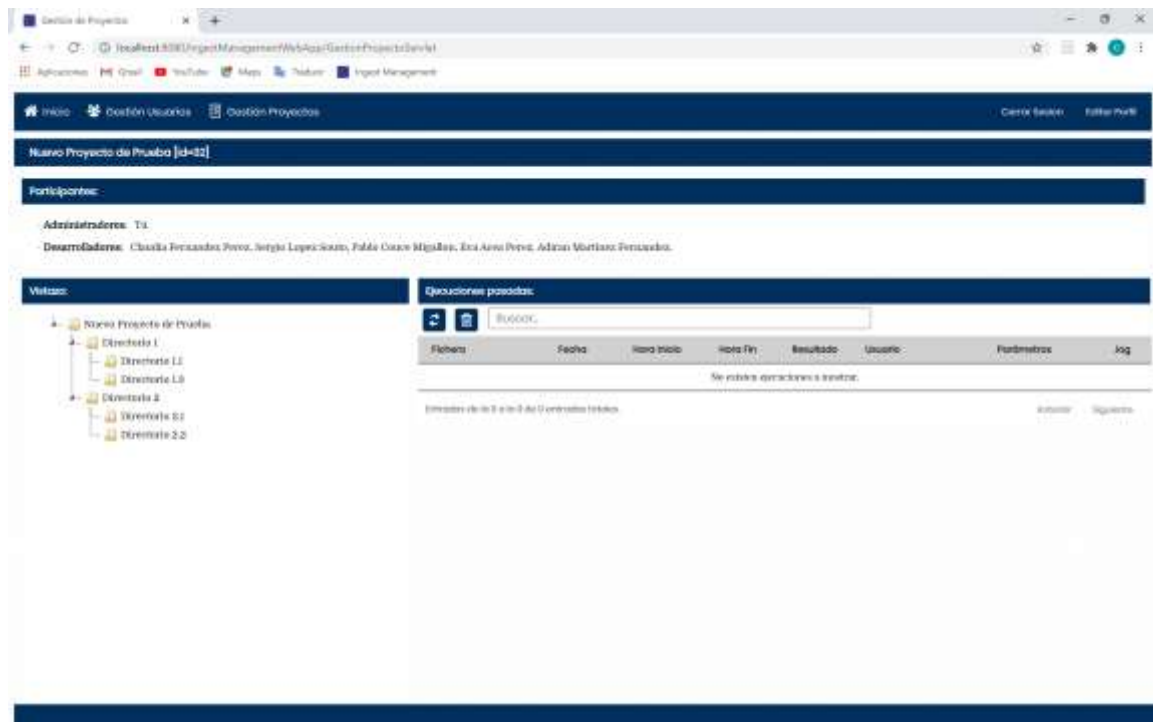
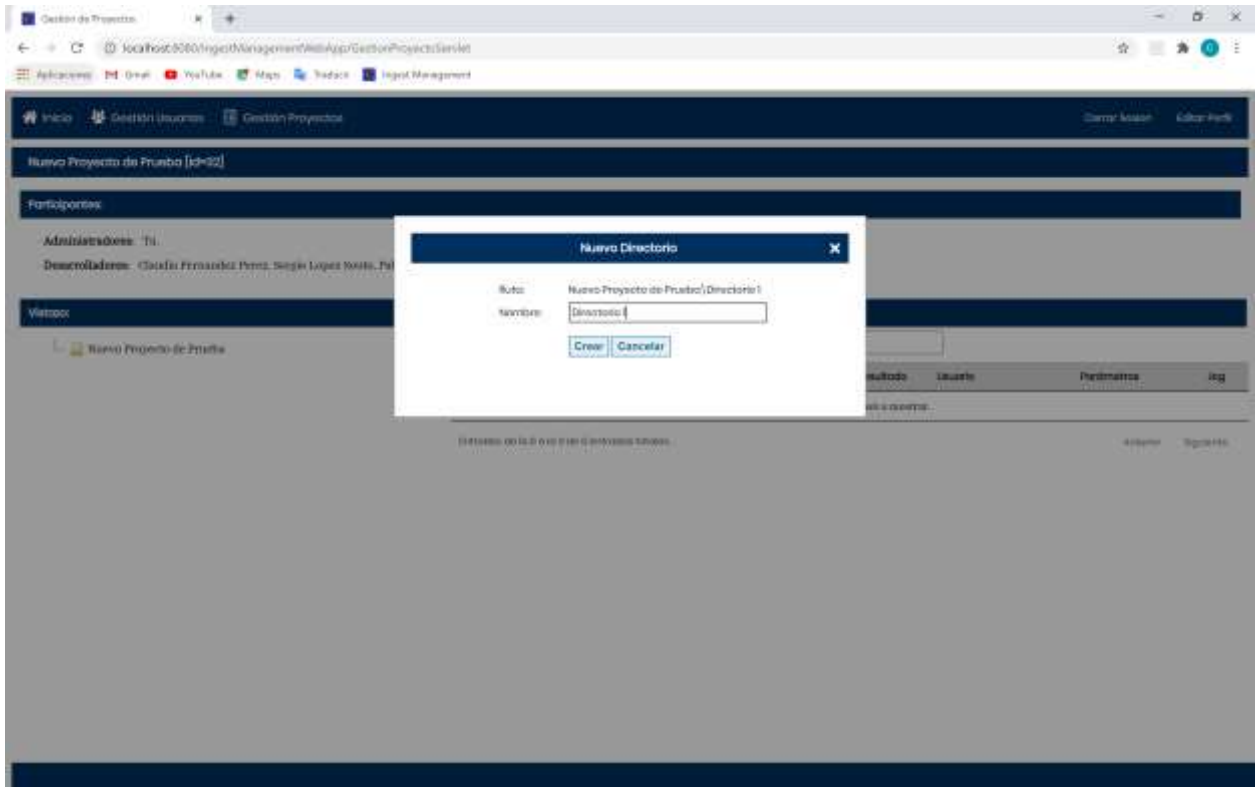


Figura 39. Ventana Proyecto de Prueba[3]



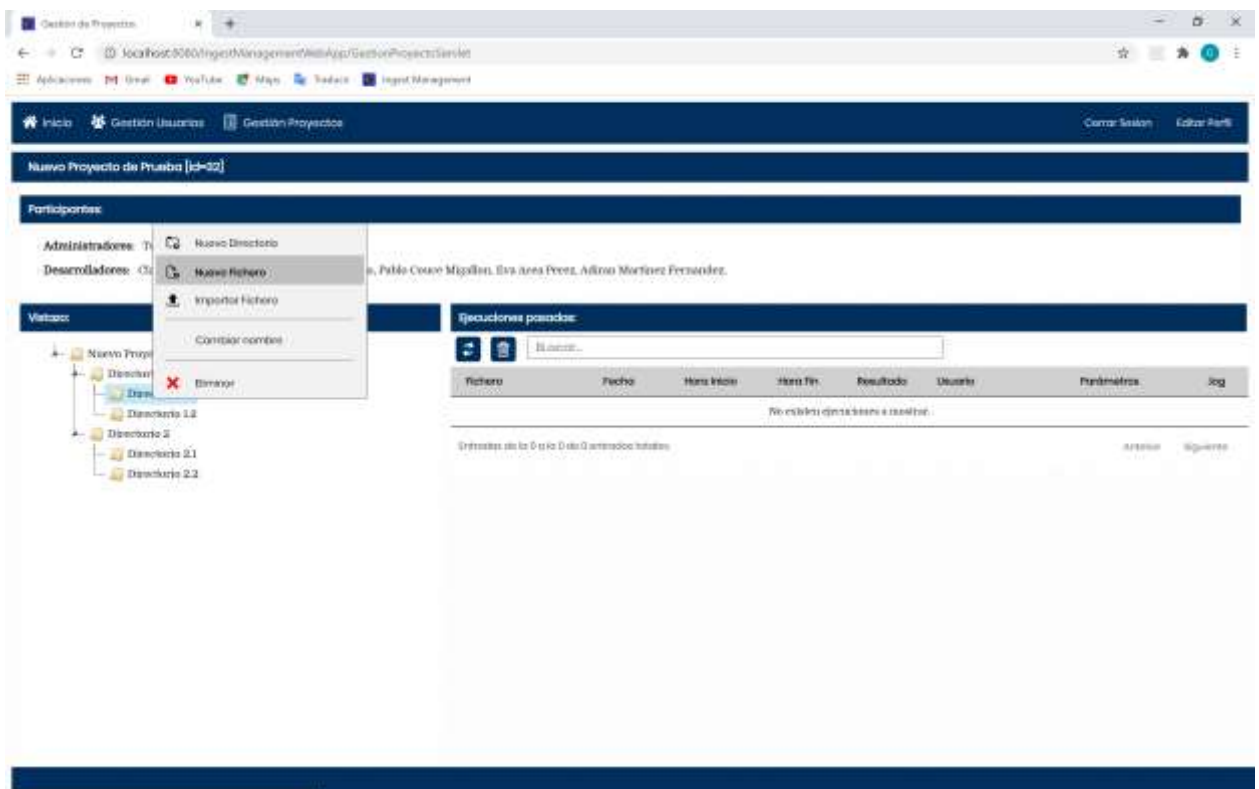
*Figura 41. Ventana nuevo Directorio WebApp*

Para modificar el nombre de un directorio o fichero simplemente selecciona el directorio o fichero a cambiar el nombre, haz click derecho y selecciona la opción de cambiar nombre. El nombre se hará editable y podrás cambiarlo en el mismo árbol pulsando enter o haciendo click en cualquier punto de la pantalla.



*Figura 42. Cambiar nombre directorio/fichero WebApp*

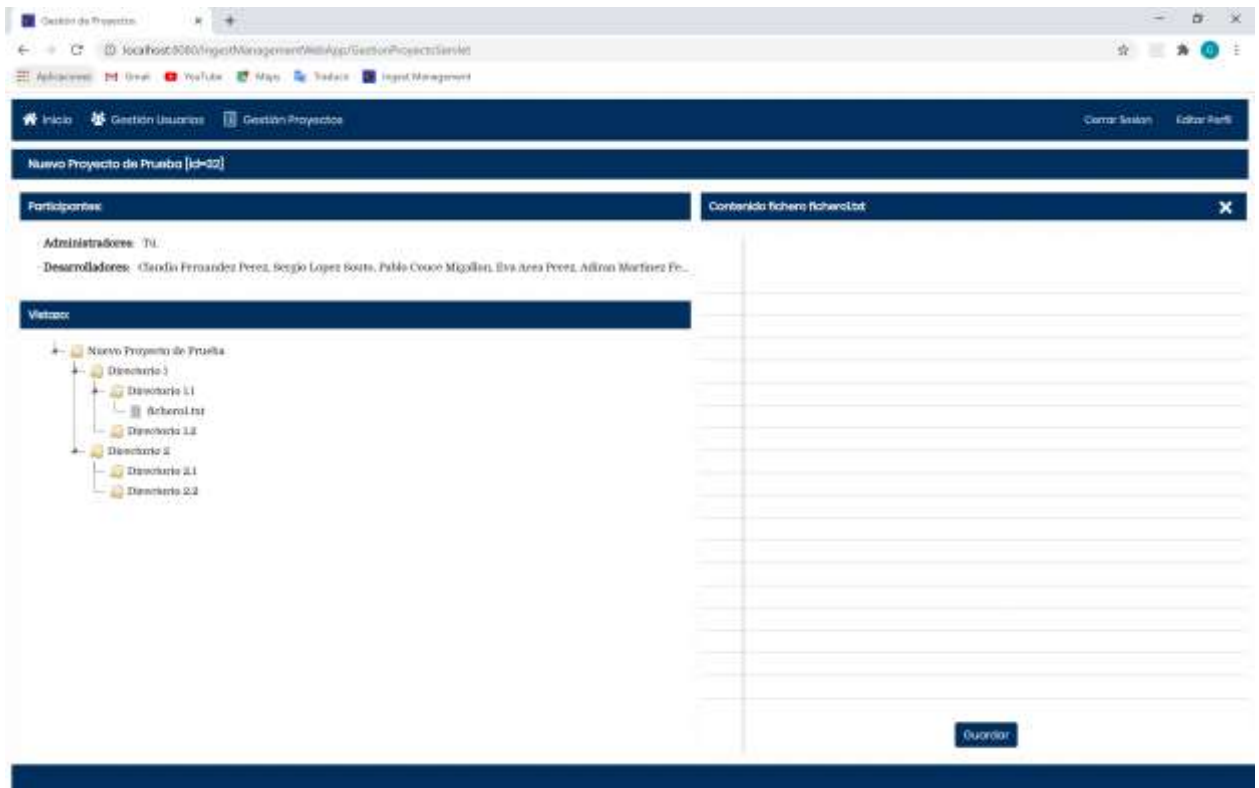
Para crear un nuevo fichero de texto haz click derecho en el directorio donde quieras crearlo y selecciona la opción de Nuevo Fichero. Se abrirá una ventana que te pedirá el nombre del fichero y te mostrará la ruta que tendrá dentro del proyecto, y al pulsar aceptar se abrirá un documento vacío en el editor de texto para escribir contenido en caso de que lo desees. Si no es así pulsa en la cruz superior derecha y se guardará en fichero vacío. Si pulsas en guardar se guardará el contenido que escribas.



*Figura 43. Ventana Proyecto de Prueba[4]*

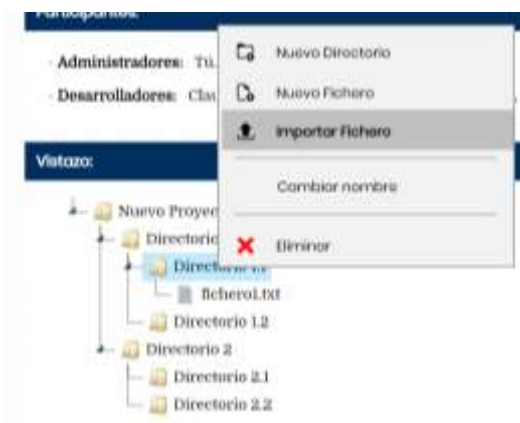


*Figura 44. Ventana nuevo fichero de texto WebApp*



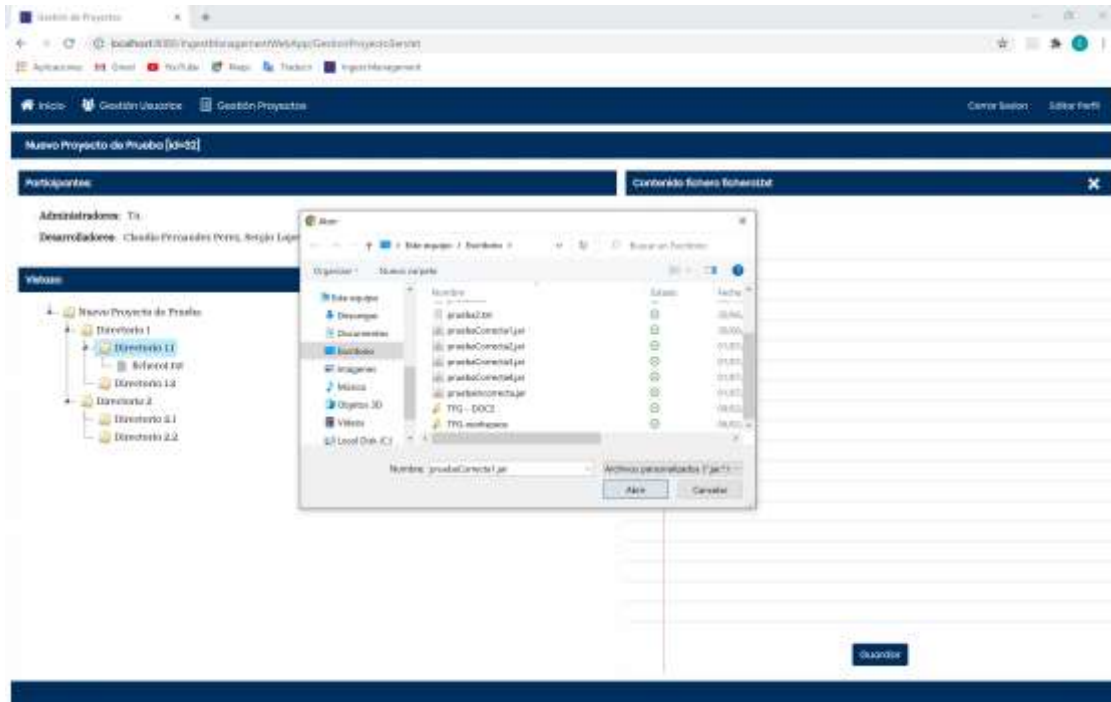
*Figura 45. Nuevo fichero de texto WebApp[2]*

Para importar un fichero (ya sea de texto o ejecutable), deberás hacer click derecho en el directorio donde desees importar el fichero y pulsar en “Importar fichero” del menú que se abra.

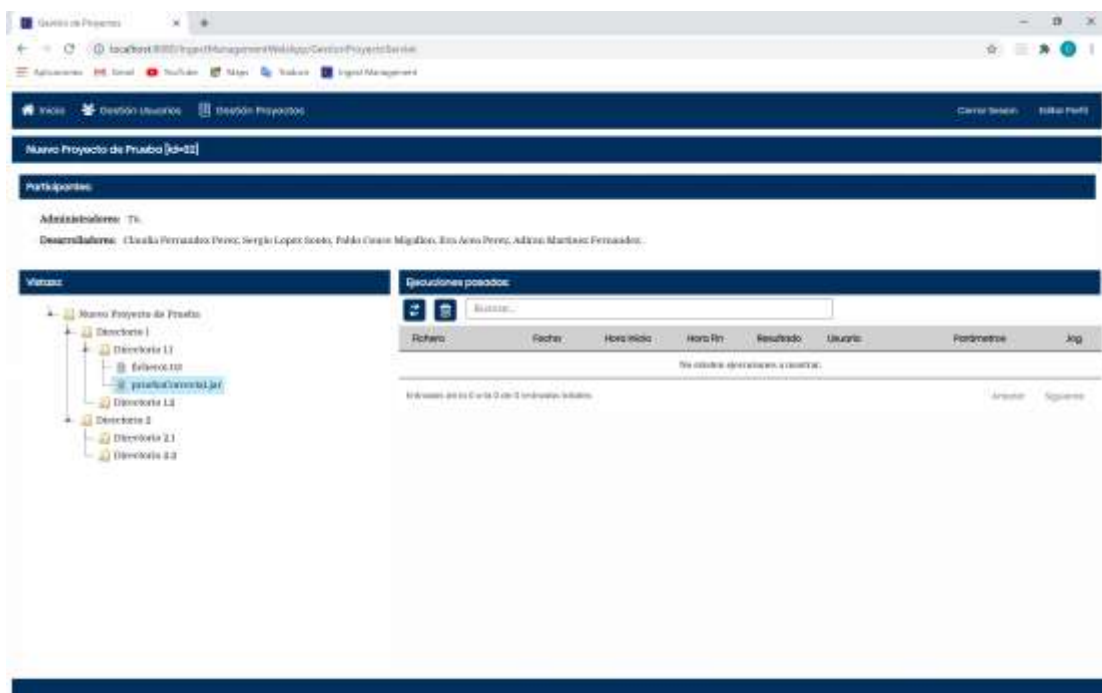


*Figura 46. Importar fichero menú desplegable WebApp*

Se abrirá una ventana que te permitirá elegir un fichero de .txt o .jar de tu ordenador. Si el fichero seleccionado no es .txt o .jar la aplicación no lo importará.

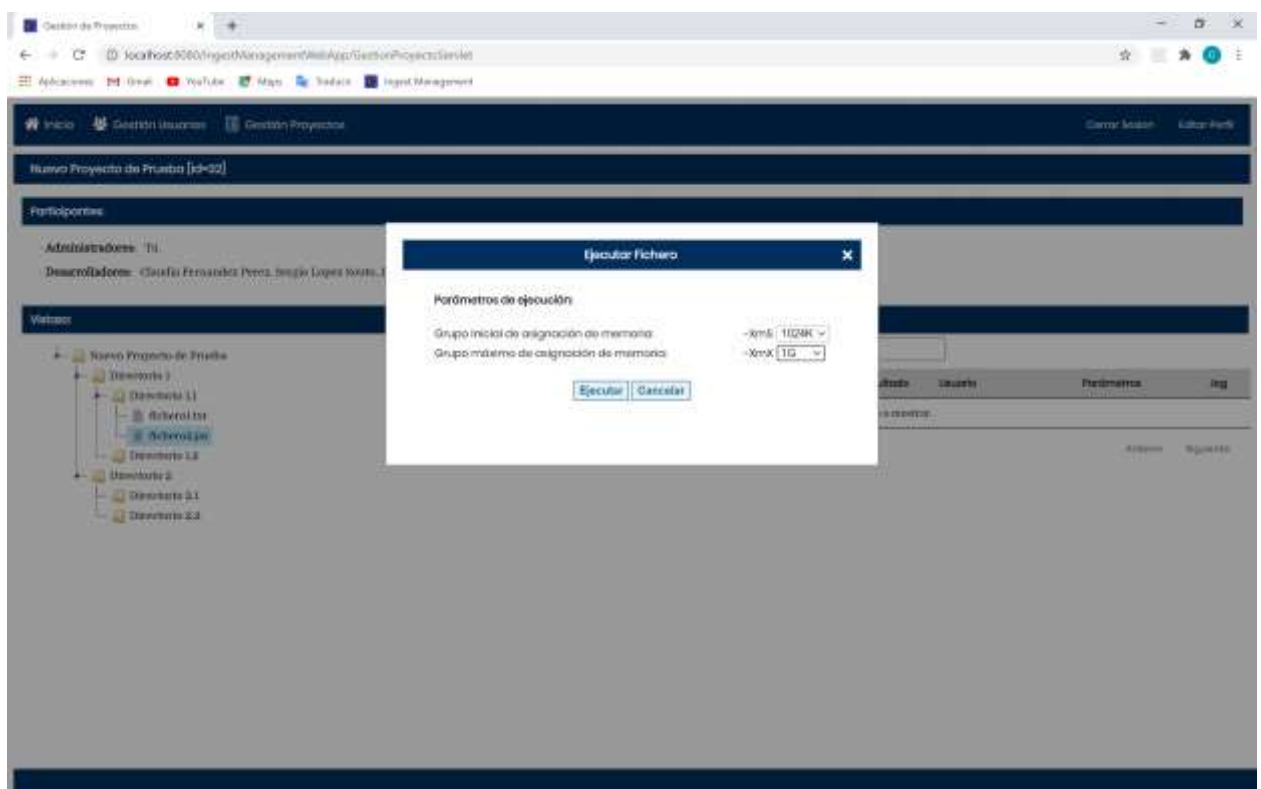


*Figura 48. Ventana selección de fichero importado WebApp*



*Figura 47. Ventana Proyecto de Prueba WebApp[5]*

Una vez subido el fichero ejecutable, se podrá ejecutar o programar su ejecución haciendo click derecho en el fichero que corresponda y seleccionando la opción de “Ejecutar” o “Programar ejecución”. Se abrirá la ventana de ejecutar fichero (con opciones para programar la ejecución de forma puntual o periódica en caso de querer programar la ejecución).



*Figura 49. Ventana Ejecutar fichero WebApp*

Seleccionas los parámetros de ejecución que deseas (grupo inicial y máximo de asignación de memoria) o los dejas vacíos para que use los valores predeterminados, y eliges la hora y cuando quieras que se repita la ejecución en caso de ser programada. Si eliges que no se repita nunca la ejecución se realizará a la hora seleccionada (de hoy si la hora aún no ha pasado o de mañana en caso contrario). Si eliges que se repita todos los días se ejecutará todos los días a esa hora durante un año, y si seleccionas la opción de un día de la semana en particular se ejecutará una vez a la semana a esa hora.

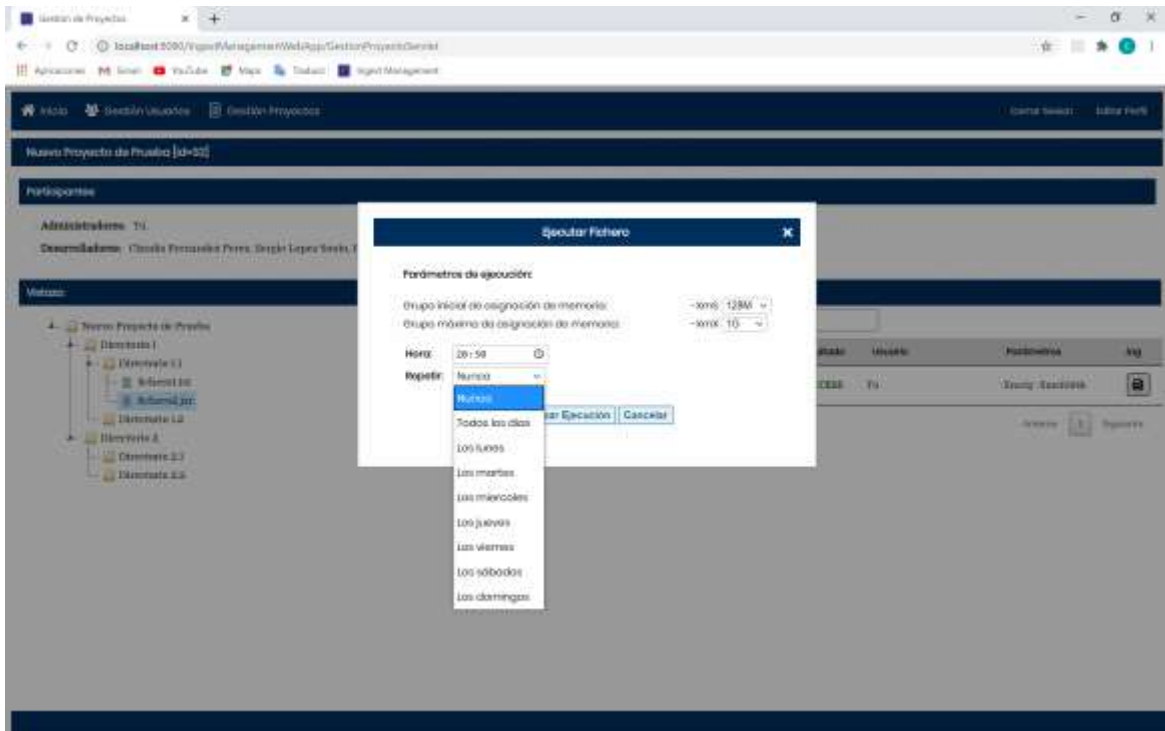


Figura 51. Ventana programar ejecución fichero WebApp

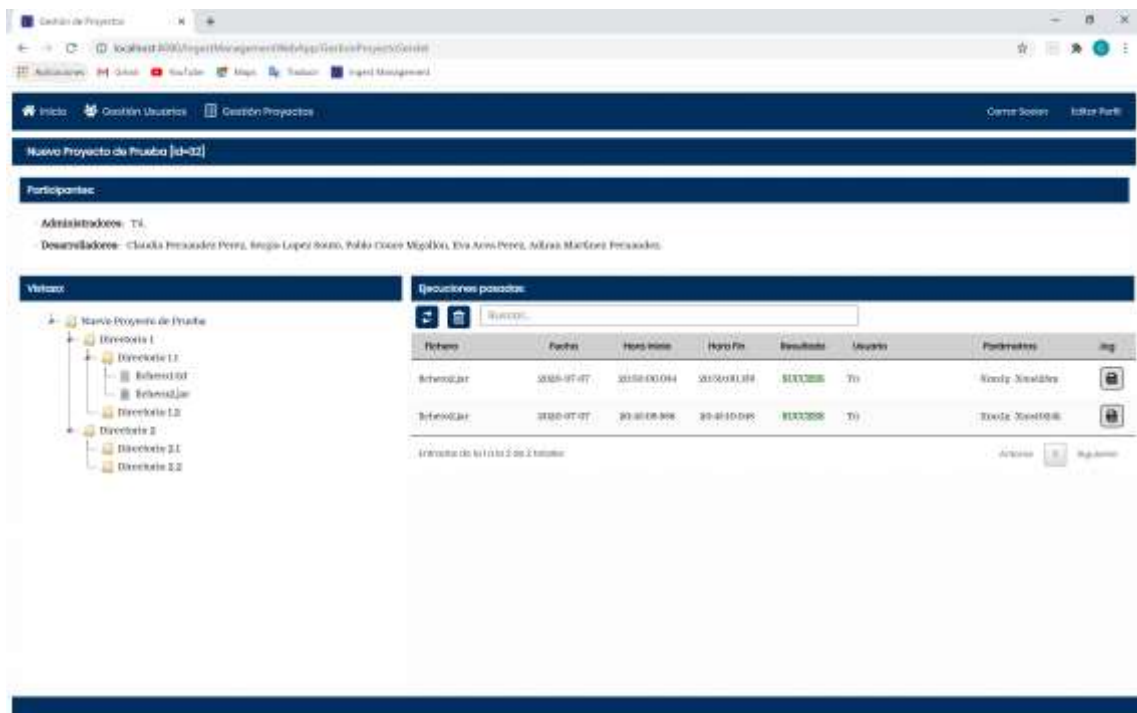
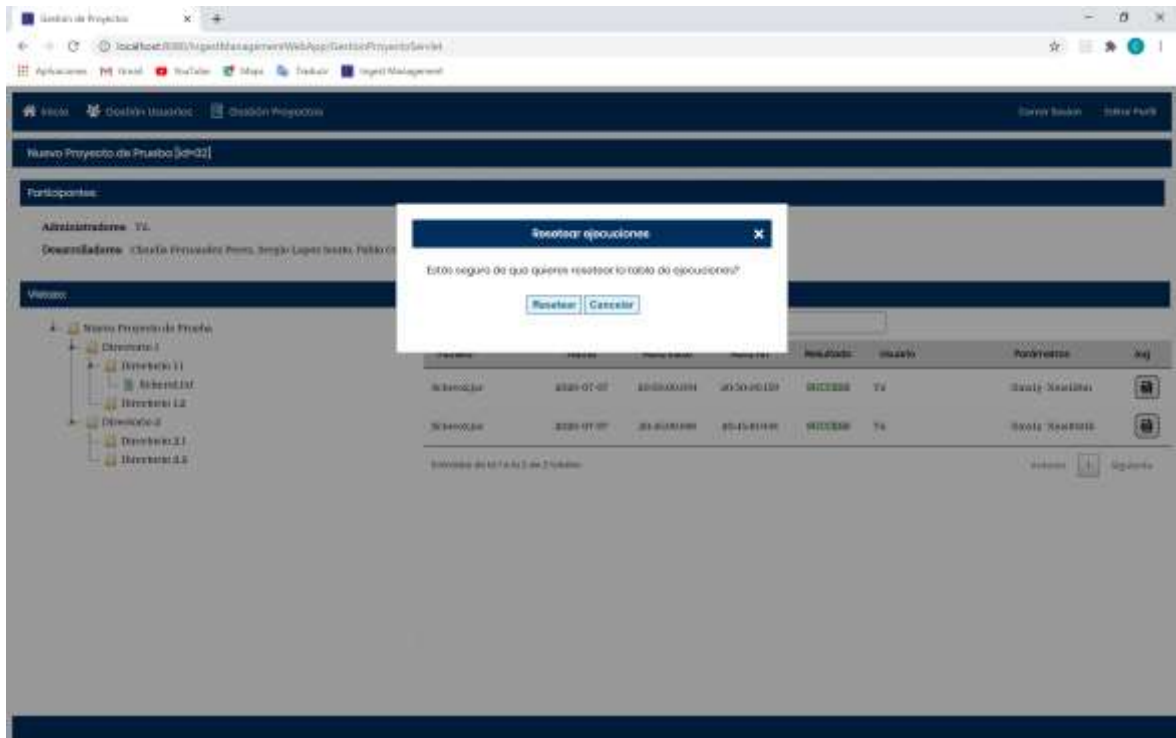


Figura 50. Ficheros ejecutados WebApp



Por último, tabla de ejecuciones te da la opción de resetear las ejecuciones para así poder limpiar las ejecuciones pasadas que ya no interesen de base de datos. Pulsando en la papelera en la esquina superior izquierda:



*Figura 52. Confirmación resetear tabla ejecuciones pasadas WebApp*

Con esta acción se finalizan las acciones que pueden realizarse dentro de un proyecto, y por lo tanto en la aplicación.

Para cerrar sesión y salir de la aplicación simplemente pulsa en “Cerrar Sesión” del menú superior de navegación y se eliminará tu sesión y redirigirá a la página de inicio.



*Figura 53. Cerrar Sesión WebApp*

## **ANEXO III: ALINEACIÓN PROYECTO CON LOS ODS**

Los ODS son 17 “Objetivos de Desarrollo Sostenible” que conforman un plan de acción adoptado por la ONU en el 2015. Este plan tiene como objetivo principal acabar con la pobreza y el hambre, combatir la desigualdad, y luchar contra el cambio climático. 17 objetivos y 169 metas fijadas para los próximos 15 años, y que comprometen tanto a los Estados como a las empresas u organizaciones a realizarlos de manera voluntaria.

Algunos de estos objetivos son por ejemplo: poner fin a la pobreza y al hambre, igualdad de género, educación de calidad, acción por el clima, producción y consumo responsables, o energía asequible y no contaminante.

La aplicación desarrollada, por ser una aplicación cuyo objetivo es facilitar la gestión de procesos de ingesta de datos en entornos de Big Data, puede relacionarse con prácticamente todos estos objetivos.

Los ODS no son sino KPIs de desarrollo social, y todo KPI ha de medirse para monitorizar así su evolución en el tiempo. El papel del Big Data en ámbitos no lucrativos “es el mismo que en el ámbito empresarial: los datos ayudan a navegar con rumbo, con conocimiento y visión de la ruta y de los obstáculos que podemos encontrarnos por el camino”, como bien dijo Juan Murillo, responsable de Divulgación Analítica en el área de Data Strategy & Data Science Innovation de BBVA.

La creación de valor mediante el Big Data es una potencial fuente de beneficios y bienestar para sociedades avanzadas y en desarrollo. El emprendimiento es clave para escalar el beneficio del Big Data. El uso de datos puede permitir alinear objetivos y modelos de negocio con los ODS.

Cuando se considera el uso de datos para el bien social, hay que poner en la balanza también la privacidad y el beneficio que se puede alcanzar, así como el riesgo de no usar

datos en el momento adecuado. Con medidas de seguridad adecuadas y protocolos de privacidad es posible usar datos sensibles para un bien social de gran impacto.

De hecho, el Grupo de las Naciones Unidas para el Desarrollo (GNUM), ya ha ofrecido orientaciones generales sobre la privacidad, la protección y la ética en el uso de los macrodatos, que son recogidos en tiempo real por entidades del sector privado, y compartidos con los miembros del GNUM con el objetivo de fortalecer la aplicación de sus programas diseñados para cumplir la Agenda 2030.

El diseño centrado en humanos es clave para la convivencia con la tecnología digital. La innovación con Big Data debe servir para empoderar a los diferentes actores del Desarrollo y a las personas. Es importante no perder la oportunidad que el Big Data brinda para alcanzar los ODS.

Sin embargo, el uso de la aplicación en entornos de Big Data no es lo único que puede relacionarlo con los Objetivos de Desarrollo Sostenible. Aunque la aplicación está pensada para ser usada con ficheros que ejecuten un proceso de ingesta de datos, puede ejecutar cualquier tipo de proceso con éxito, guardando y permitiendo la revisión de su traza de ejecución.

La aplicación podría ser usada (con pequeñas modificaciones para cambiar el tipo de ficheros que se pueden subir o lo que hacer con ellos), para gestionar cualquier tipo de proyecto, aunque no tuviese nada que ver con el Big Data.

La parte más larga y compleja de desarrollar de nuestra aplicación fue precisamente toda la base del sistema de gestión de usuarios y proyectos, así como de sus entornos asociados y permisos. Al ser una pieza fundamental (y a veces única) en cualquier aplicación de hoy en día, podría reutilizarse gran parte del código desarrollado para crear una aplicación que por ejemplo permitiese monitorizar y gestionar proyectos sociales en cualquier país en desarrollo, permitiendo subir PDFs o imágenes con toda la información sobre la evolución de los proyectos.

*ANEXO III: ALINEACIÓN PROYECTO CON LOS ODS*

---

Al igual que este ejemplo podría poner miles más, en las que el código desarrollado en esta aplicación, aunque no pudiese utilizarse directamente en proyectos directamente ligados con los ODS, pudiese reutilizarse y adaptarse para crear algo que sí lo estuviese. Esta es una de las gran ventajas de las aplicaciones Java, y sobretodo de aquellas desarrolladas con patrones que permitan la reutilización como es el patrón utilizado y explicado en el Capítulo 5 MVC.

En resumen, la aplicación desarrollada puede contribuir, tanto por su utilidad principal en entornos de Big Data como por la capacidad de reutilización de su código, para la monitorización o consecución de los 17 Objetivos de Desarrollo Sostenible.