



---

# Universidad de Valladolid

**Desarrollo de simulaciones y juegos serios de planificaciones de proyectos. App Web.**

**TRABAJO FIN DE GRADO EN INGENIERÍA INFORMÁTICA  
MENCION TECNOLOGÍAS DE LA INFORMACIÓN**

**AUTOR: Jose Alberto del Val Acebes**

**TUTOR: César Llamas Bello**

**Escuela de ingeniería informática, Valladolid. JUNIO 2020**





# RESUMEN

Entendemos como proyecto a un desarrollo o esfuerzo temporal que se realiza para obtener un producto, servicio o un resultado único que dará lugar a un cambio positivo o añadirá valor al estado previo a la realización de dicho proyecto. La planificación y gestión de proyectos, como su nombre indica, es la materia que estudia y define la organización, el planteamiento y el control de los determinados recursos de los que se disponen en un proyecto, para alcanzar los objetivos, dentro de un proyecto.

En el campo de la docencia, la planificación y gestión de proyectos, es una materia muy amplia, que requiere diversas herramientas para su correcto desarrollo, debido a las variadas técnicas que dicha materia puede considerar para planificar y gestionar un proyecto. Una herramienta esencial puede ser un simulador de planificaciones o planes, a través del cual un alumno pueda realizar su propia planificación de un determinado proyecto a través de plantillas prefijadas por el profesor y así obtener sus propios resultados, para próximamente valorar dicha planificación.

Es esto lo que me motiva a desarrollar una aplicación que realice lo citado anteriormente.

En esta memoria se detalla el diseño e implementación de una aplicación web basada en el desarrollo de simulaciones y juegos serios de planificaciones de proyectos.

El desarrollo de simulaciones y juegos se muestra productivo debido a la capacidad de apoyo que proporciona al alumno el manejar los datos de su propia planificación viendo así su evolución y pudiendo considerar los aspectos en los que los resultados están siendo peores.

# ABSTRACT

We understand as a project a development or temporary effort that is made to obtain a product, service or a unique result that will lead to a positive change or add value to the state prior to the completion of said project. Project planning and management, as its name indicates, is the subject that studies and defines the organization, approach and control of the certain resources available in a project, to achieve the objectives, within a project.

In the field of teaching, project planning and management is a very broad subject, which requires various tools for its proper development, due to the various techniques that this subject may consider to plan and manage a project. An essential tool can be a planning simulator or plans, through which a student can carry out their own planning of a certain project through templates prefixed by the teacher and thus obtain their own results, in order to soon assess said planning.

This is what motivates me to develop an application that does the above.

This report details the design and implementation of a web application based on the development of simulations and serious project planning games. The development of simulations and games is productive due to the support capacity that the student provides to manage the data of his own planning, thus seeing its evolution and being able to consider the aspects in which the results are being worse.

# MOTIVACIÓN

Entendemos como proyecto a un desarrollo o esfuerzo temporal que se realiza para obtener un producto, servicio o un resultado único que dará lugar a un cambio positivo o añadirá valor al estado previo a la realización de dicho proyecto. La planificación y gestión de proyectos, como su nombre indica, es la materia que estudia y define la organización, el planteamiento y el control de los determinados recursos de los que se disponen en un proyecto, para alcanzar los objetivos, dentro de un proyecto.

En el campo de la docencia, la planificación y gestión de proyectos, es una materia muy amplia, que requiere diversas herramientas para su correcto desarrollo, debido a las variadas técnicas que dicha materia puede considerar para planificar y gestionar un proyecto. Una herramienta esencial puede ser un simulador de planificaciones o planes, a través del cual un alumno pueda realizar su propia planificación de un determinado proyecto a través de plantillas prefijadas por el profesor y así obtener sus propios resultados, para próximamente valorar dicha planificación. Es esto lo que me motiva a desarrollar una aplicación que realice lo citado anteriormente.

En esta memoria se detalla el diseño e implementación de una aplicación web basada en el desarrollo de simulaciones y juegos serios de planificaciones de proyectos. El desarrollo de simulaciones y juegos se muestra productivo debido a la capacidad de apoyo que proporciona al alumno el manejar los datos de su propia planificación viendo así su evolución y pudiendo considerar los aspectos en los que los resultados están siendo peores.

La planificación y gestión de proyectos, como su nombre indica, es la materia que estudia y define la organización, el planteamiento y el control de los determinados recursos de los que se disponen en un proyecto, para alcanzar los objetivos, dentro de un proyecto. Además es una de las partes que más importancia tienen en el ciclo de vida de realización de los mismos, siendo un factor determinante en el éxito del proyecto y en la obtención de un buen producto.

En mi caso, la asignatura de PGPI ha sido una de las que más interesantes me ha resultado, por tanto tenía claro que este Trabajo de fin de Grado iría relacionado con esta asignatura.

Trás varios intentos, surgió la idea del tutor de realizar una aplicación relacionada con el aprendizaje a planificar, concretamente de una herramienta que usarían los futuros alumnos para practicar o incluso obtener evaluaciones por parte del profesor en la asignatura citada anteriormente. Esta idea me llevó por fin a decidirme a elegir mi TFG cuyos objetivos relato en el siguiente apartado.

# ESTRUCTURA DE LA MEMORIA DEL TFG

En este documento se recoge, en la medida de lo posible lo necesario para comprender totalmente la estructura de la aplicación, la forma en la que se ha desarrollado con las determinadas tecnologías utilizadas y demás aspectos importantes. El documento posee como introducción, un pequeño resumen de lo que es este trabajo de fin de grado, junto con un índice el cual nos es de mucha ayuda para movernos por el documento.

Como comienzo del propio contenido y en el primer capítulo del documento se contempla un planteamiento de lo que la aplicación abarca, junto con un extenso análisis de posibles herramientas similares en las que apoyarnos para el desarrollo, un breve reconocimiento de los diferentes requisitos y un breve estudio de cómo podemos contemplarlos con las diferentes tecnologías de desarrollo que tenemos al alcance, además se da un vistazo a muy alto nivel de cómo es la aplicación.

En el segundo capítulo del documento encontramos un extenso análisis de requisitos, junto con una propuesta de arquitectura de la aplicación.

Cómo tercer capítulo encontramos el plan de trabajo que se ha llevado a cabo para el desarrollo del proyecto junto con un estudio de viabilidad en función de sus costes.

En el cuarto capítulo encontraremos un análisis en profundidad del dominio del software, en él se incluyen diversos diagramas de secuencia, de estado, etc. y se estudian los diferentes casos de uso, detallando y profundizando en los más importantes.

Cómo quinto capítulo del documento, encontraremos un análisis muy en profundidad del diseño en alto nivel de la aplicación. En él se detalla y estudia todo lo relacionado con la arquitectura de la aplicación.

En los capítulos adicionales se encuentra un amplio y detallado estudio de la solución llevada a cabo para diseñar el algoritmo que simula las planificaciones, esto es, la simulación basada en eventos discretos, junto con el uso de la probabilidad de Pert.

Para concluir lo que es el contenido principal del documento, se añade una sección en la que se comenta lo aprendido en dicho desarrollo, y se plantea una línea de trabajo futuro, en la que se puede seguir trabajando para ampliar las funcionalidades de dicha aplicación.

Cómo anexos a este documento podemos encontrar un manual de uso además de un manual muy completo de instalación y configuración de la herramienta web. También se añaden todas las referencias bibliográficas citadas a lo largo del documento.

# AGRADECIMIENTOS

Primeramente quiero agradecer a mi familia por su apoyo incondicional desde el momento en el que comencé estos estudios, proporcionando todo lo necesario para poder llevarlos a cabo.

Como se suele decir, los amigos son la familia que uno mismo elige y así es, estos años han servido además de para estudiar en profundidad la informática, para conocer a gente increíble, con la que he tenido el placer de compartir muchos momentos inolvidables. A todos esos amiguetes Fer, Kike, Jesus, Alvaro, Javier, Alex, etc...

También quiero agradecer a mis compañeros de piso por hacerme más llevaderos estos años tan duros y a su vez divertidos, gracias Javi y Gabriel.

Por último y no por eso menos importante, sino todo lo contrario, ya que ellos son los principales culpables del aprendizaje, agradecer por supuesto, a todo el profesorado de esta magnífica escuela de ingeniería informática de Valladolid, que tanto nos ayuda y forma para nuestro futuro laboral de una manera lo más llevadera posible, haciendo una mención especial al que para mí ha sido el profesor con el que quizás haya tenido más relación, él es D. César Llamas y ha sido mi tutor en las dos prácticas en empresas que he llevado a cabo en estos estudios y como no podía ser de otra manera también de este Trabajo Fin de Grado. Su trato ha sido inigualable, gracias César.

# ÍNDICE DE CONTENIDO

<b>RESUMEN</b>	<b>3</b>
<b>ABSTRACT</b>	<b>4</b>
<b>MOTIVACIÓN</b>	<b>5</b>
<b>ESTRUCTURA DE LA MEMORIA DEL TFG</b>	<b>6</b>
<b>AGRADECIMIENTOS</b>	<b>7</b>
<b>ÍNDICE DE CONTENIDO</b>	<b>8</b>
<b>ÍNDICE DE TABLAS</b>	<b>12</b>
<b>ÍNDICE DE FIGURAS Y GRÁFICOS</b>	<b>14</b>
<b>Capítulo 1. Objetivos, contexto y propuesta.</b>	<b>15</b>
1.1 Objetivos	15
1.2 Contexto	16
1.2.1 Metodologías de planificación	16
1.2.2 Software relacionado	19
<b>Capítulo 2. Análisis de requisitos.</b>	<b>19</b>
2.1 Contexto general.	19
2.2 Requisitos Funcionales	20
2.3 Requisitos No Funcionales	22
2.3.1 Restricciones	23
2.4 Usabilidad de la aplicación y diagramas de Casos de Uso	23
2.4.1 Análisis del funcionamiento general de la aplicación.	23
2.4.2 Casos de uso Profesor.	25
2.4.3 Casos de uso Alumno.	26
2.5 Matriz de incidencia entre casos de uso y requisitos	27
2.5.1 Matriz de incidencia de Requisitos Funcionales.	28
<b>Capítulo 3. Plan de trabajo y seguimiento.</b>	<b>29</b>
3.1 Contexto y planificación seguida.	29
3.2 Detalle de las diferentes tareas o fases.	30
3.2.1 Fase previa.	30
3.2.2 Primer prototipo	30
3.2.3 Segundo prototipo	31
3.2.4 Tercer Prototipo	31
3.2.5 Prototipo final	31
3.2.6 Diagrama de Gantt	33
3.3 Gestión de Riesgos	34



3.3.1 Contexto gestión de riesgos	34
3.3.2 Plan de gestión de riesgos	35
3.4 Gestión de Costes	40
3.4.1 Introducción y contexto	40
3.4.2 Costes de recursos humanos	40
3.4.3 Costes hardware/software	41
3.4.4 Costes totales del proyecto	42
3.5 Seguimiento	42
3.5.1 Contexto y detalle	42
3.5.2 Estimación vs Realidad	42
3.5.2.1 Fase previa.	42
3.5.2.2 Primer prototipo.	43
3.5.2.3 Segundo prototipo.	43
3.5.2.4 Tercer prototipo.	44
3.5.2.5 Prototipo final.	44
<b>Capítulo 4. Análisis del dominio software.</b>	<b>46</b>
4.1 Introducción	46
4.2 Definición de los Casos de uso	46
4.2.1 Casos de uso usuario tipo profesor	46
4.2.2 Casos de uso usuario tipo alumno	52
4.3 Modelo de dominio	59
4.4 Análisis de los casos de uso más relevantes.	60
4.4.1 Modificar planificación.	60
4.4.2 Avanzar días en la simulación.	61
<b>Capítulo 5. Diseño de bajo nivel, tecnologías.</b>	<b>62</b>
5.1 Contexto de las tecnologías web.	62
5.2 Angular 8.	62
5.3 Angular CLI.	63
5.4 Node Js.	63
5.5 Npm.	64
5.6 Servicios Firebase de Google.	65
5.6.1 Firebase Cloud Database.	66
5.6.2 Firebase Cloud Functions.	66
5.6.3 Firebase Auth.	66
5.7 Bootstrap 4.4.	67
5.8 Git.	67
<b>Capítulo 6. Diseño de la arquitectura.</b>	<b>68</b>
6.1 Contexto	68
6.2 Arquitectura facilitada por Angular	68
6.2.1 Módulos	68
6.2.2 Servicios	69

6.2.3 Componentes y Templates.	69
6.2.4 Resumen.	70
6.3 Arquitectura de la aplicación desarrollada.	71
6.3.1 Diagrama de despliegue y diagrama acceso a servicios.	71
6.3.2 Detalle de cada componente.	73
6.3.2.1 LoginComponent.	73
6.3.2.2 AlumnoPanelComponent.	75
6.3.2.3 AdminPanelComponent.	77
6.3.3 Detalle de cada servicio.	79
6.3.3.1 AuthenticationService.	79
6.3.3.2 FirestoreService.	79
<b>Capítulo 7. Evaluación y pruebas.</b>	<b>80</b>
7.1 Contexto	80
7.2 Pruebas unitarias	81
7.3 Pruebas de integración	86
7.4 Pruebas de sistema	88
7.4.1 Pruebas de compatibilidad	88
7.4.1 Pruebas de aceptación	88
<b>Capítulo 8. Conclusiones y trabajo futuro.</b>	<b>89</b>
8.1 Contexto	89
8.2 Conclusiones y aprendizaje	89
8.3 Trabajo y propuestas futuras	90
<b>Referencias Bibliográficas.</b>	<b>92</b>
<b>ANEXO I. Algoritmo de simulación</b>	<b>93</b>
1. Contexto.	93
2. Simulación basada en eventos discretos.	93
2.1 Introducción.	93
2.2 Componentes de un sistema Basado en eventos discretos.	93
3. Probabilidad de Pert	94
3.1 Probabilidad de finalización de una tarea o proyecto mediante Pert.	95
4. Algoritmo desarrollado	96
4.1 Visión General. Aplicación de eventos Discretos	96
4.1.1 Estados de la simulación	96
4.1.2 Estados de las tareas	97
4.2 Bucle general de funcionamiento del algoritmo	99
<b>ANEXO II. Manual de uso.</b>	<b>100</b>
1. Inicio de sesión.	100
2. Uso panel del alumno.	100
2.1 Crear proyecto.	100
2.2 Borrar datos proyecto.	102

2.3 Crear planificación del proyecto.	103
2.4 Asignar recursos a la planificación.	104
2.5 Simular planificación.	104
2.6 Modificar planificación del proyecto.	105
2.7 Ver y descargar resultados.	106
2. Uso panel del profesor o administrador.	107
2.1 Validar Recursos	107
2.2 Ver y crear plantillas de planificación	108
2.3 Ver proyectos en simulación	109
2.4 Ver proyectos entregados y descarga de resultados.	109
2.5 Ver/Registrar/Eliminar Usuarios	110
<b>ANEXO III. Manual de despliegue.</b>	<b>111</b>
1. Requisitos previos.	111
2. Creación de un proyecto en Firebase.	111
3. Creación de una aplicación dentro del proyecto creado.	113
4. Configuración del proyecto angular.	114
5. Configuraciones y creación de servicios en la app firebase.	116
6. Compilación y despliegue.	120
<b>ANEXO IV. Repositorio del código fuente.</b>	<b>123</b>

# ÍNDICE DE TABLAS

*Tabla 2.1 Requisitos funcionales.*

*Tabla 2.2 Requisitos no funcionales.*

*Tabla 2.3 Requisitos no funcionales (restricciones).*

*Tabla 2.4 Matriz de incidencia entre requisitos funcionales y casos de uso.*

*Tabla 3.1 Detalle del riesgo incumplimiento de la planificación.*

*Tabla 3.2 Detalle del riesgo gestión pobre de los requisitos.*

*Tabla 3.3 Detalle del riesgo falta de comunicación.*

*Tabla 3.4 Detalle del riesgo indisponibilidad del alumno o tutor.*

*Tabla 3.5 Detalle del riesgo Indisponibilidad hardware.*

*Tabla 3.6 Cambio en los requisitos de la aplicación.*

*Tabla 3.7 Detalle del riesgo Falta de experiencia con tecnologías de desarrollo.*

*Tabla 3.8 Detalle del riesgo Manual de usuario complejo.*

*Tabla 3.9 Detalle del riesgo Pérdida de la información.*

*Tabla 3.10 Detalle del riesgo Pruebas poco efectivas.*

*Tabla 3.11 Detalle del coste humano.*

*Tabla 3.12 Detalle del coste hardware.*

*Tabla 3.13 Detalle del coste software.*

*Tabla 3.14 Detalle del coste total.*

*Tabla 3.15 Estimación vs Realidad Fase Previa.*

*Tabla 3.16 Estimación vs Realidad Primer Prototipo.*

*Tabla 3.17 Estimación vs Realidad Segundo Prototipo.*

*Tabla 3.18 Estimación vs Realidad Tercer Prototipo.*

*Tabla 3.19 Estimación vs Realidad Prototipo Final.*

*Tabla 4.1 Detalle caso de uso P1.*

*Tabla 4.2 Detalle caso de uso P2.*

*Tabla 4.3 Detalle caso de uso P3.*

*Tabla 4.4 Detalle caso de uso P4.*

*Tabla 4.5 Detalle caso de uso P5.*

*Tabla 4.6 Detalle caso de uso P6.*

*Tabla 4.7 Detalle caso de uso P7.*

*Tabla 4.8 Detalle caso de uso P8.*

*Tabla 4.9 Detalle caso de uso P9.*

*Tabla 4.10 Detalle caso de uso A1.*

*Tabla 4.11 Detalle caso de uso A2.*

*Tabla 4.12 Detalle caso de uso A3.*

*Tabla 4.13 Detalle caso de uso A4.*

*Tabla 4.14 Detalle caso de uso A5.*

*Tabla 4.15 Detalle caso de uso A6.*

*Tabla 4.16 Detalle caso de uso A7.*

*Tabla 4.17 Detalle caso de uso A8.*  
*Tabla 4.18 Detalle caso de uso A9.*  
*Tabla 4.19 Detalle caso de uso A10.*  
*Tabla 4.20 Detalle caso de uso A11.*

*Tabla 7.1 Detalle caso de Prueba login 1.*  
*Tabla 7.2 Detalle caso de Prueba login 2.*  
*Tabla 7.3 Detalle caso de Prueba getCantidadRecurso.*  
*Tabla 7.4 Detalle caso de Prueba tiempoPesimoOptimo 1.*  
*Tabla 7.5 Detalle caso de Prueba tiempoPesimoOptimo 2.*  
*Tabla 7.6 Detalle caso de Prueba ceraProyectoSimulacion.*  
*Tabla 7.7 Detalle caso de Prueba borrarProyecto.*  
*Tabla 7.8 Detalle caso de Prueba hayUnRecurso 1.*  
*Tabla 7.9 Detalle caso de Prueba hayUnRecurso 2.*  
*Tabla 7.10 Detalle caso de Prueba newRecurso.*  
*Tabla 7.11 Detalle caso de Prueba getAlumnos.*  
*Tabla 7.12 Detalle caso de Prueba cuentaTareas.*  
*Tabla 7.13 Detalle caso de Prueba newAlumno.*  
*Tabla 7.15 Detalle caso de Prueba iniciar Sesión.*  
*Tabla 7.16 Detalle caso de Prueba de Plantilla con error en recursos.*  
*Tabla 7.17 Detalle caso de Prueba crear Proyecto.*  
*Tabla 7.18 Detalle caso de Prueba avanzar simulación.*  
*Tabla 7.19 Detalle caso de Prueba crear alumno/grupo alumnos.*

# ÍNDICE DE FIGURAS Y GRÁFICOS

*Figura 1.1. Línea de tiempo de las metodologías de planificación*

*Figura 1.2. Ejemplo de planificación en la app desarrollada.*

*Figura 2.1. Esquema uso sistema a desarrollar.*

*Figura 2.2. Casos de uso usuario tipo profesor.*

*Figura 2.3. Casos de uso usuario tipo alumno.*

*Figura 3.1. Diagrama de Gantt planificación TFG.*

*Figura 4.1. Modelo de Dominio del sistema.*

*Figura 4.2. Diagrama de secuencia de análisis Caso de Uso “modificar planificación”.*

*Figura 4.3. Diagrama de secuencia de análisis Caso de Uso “simular planificación”.*

*Figura 5.1. Imágen Angular.*

*Figura 5.2. Imágen AngularCLI.*

*Figura 5.3. Imágen NodeJS.*

*Figura 5.4. Imágen npm.*

*Figura 5.5. Imágen Google Firebase.*

*Figura 5.6. Imágen Bootstrap.*

*Figura 5.7. Imágen git.*

*Figura 6.1. Diagrama funcionamiento Angular.*

*Figura 6.2. Diagrama de despliegue app desarrollada.*

*Figura 6.3. Diagrama de acceso a servicios.*

*Figura 6.4. Diagrama de arquitectura LoginComponent.*

*Figura 6.5. Detalle funciones y atributos LoginComponent.*

*Figura 6.6. Diagrama de arquitectura alumnoPanelComponent.*

*Figura 6.7. Detalle funciones y atributos alumnoPanelComponent.*

*Figura 6.8. Diagrama de arquitectura adminPanelComponent.*

*Figura 6.9. Detalle funciones y atributos adminPanelComponent.*

*Figura 6.10. Detalle funciones y atributos AuthenticationService.*

*Figura 6.11. Detalle funciones y atributos FirestoreService.*

# Capítulo 1. Objetivos, contexto y propuesta.

## 1.1 Objetivos

El TFG como su nombre indica es la última asignatura del grado, en la que se deben aplicar la mayoría de conocimientos adquiridos durante los años de estudio del grado. Sus principales objetivos de formación son:

- Realizar un proyecto en su totalidad, teniendo un cliente real, en este caso el departamento de informática de la Universidad.
- Elaborar un informe completo de dicho proyecto.
- Realizar una presentación pública de dicho proyecto.

Además el proyecto software realizado en este caso, tiene sus propios objetivos que hay que llegar a conseguir:

- Facilitar el estudio de las metodologías de planificación y gestión de proyectos.
- Estudio del framework y tecnologías elegidas para realizar la aplicación.
- Permitir realizar una planificación de un proyecto con sus determinadas dependencias entre tareas.
- Permitir asignar recursos a las diferentes tareas definidas en la planificación
- Permitir simular una planificación de un proyecto.
- Permitir replanificar la asignación de recursos a las diferentes tareas.
- Permitir simular riesgos surgidos a través de la simulación.
- Permitir consultar el estado de la simulación en todo momento.
- Obtener un informe de la simulación en formato CSV una vez esta ha finalizado.
- Permitir desplegar la aplicación en un servidor.
- Conseguir una aplicación lo más usable e intuitiva posible.

## 1.2 Contexto

En este subcapítulo comentaremos y daremos un repaso a las diferentes metodologías de planificación existentes, además de explicar la metodología, seleccionada para realizar las diferentes simulaciones en dicha aplicación. Por otro lado, realizaremos un breve estudio de software similar o que nos pueda servir de apoyo para realizar dicha aplicación.

### 1.2.1 Metodologías de planificación

Una buena planificación es la clave del éxito de un proyecto de cualquier tipo, pero en el caso de este trabajo de fin de grado, nos vamos a centrar principalmente en los proyectos informáticos. Actualmente existen multitud de metodologías para planificar un proyecto, encuadradas en dos grandes grupos, las metodologías tradicionales, basadas en la predicción y las metodologías ágiles, basadas en la adaptabilidad [PMBOK]. A lo largo de la historia de la planificación se ha ido evolucionando y han surgido muchos cambios, debido a el gran avance tecnológico y la investigación.

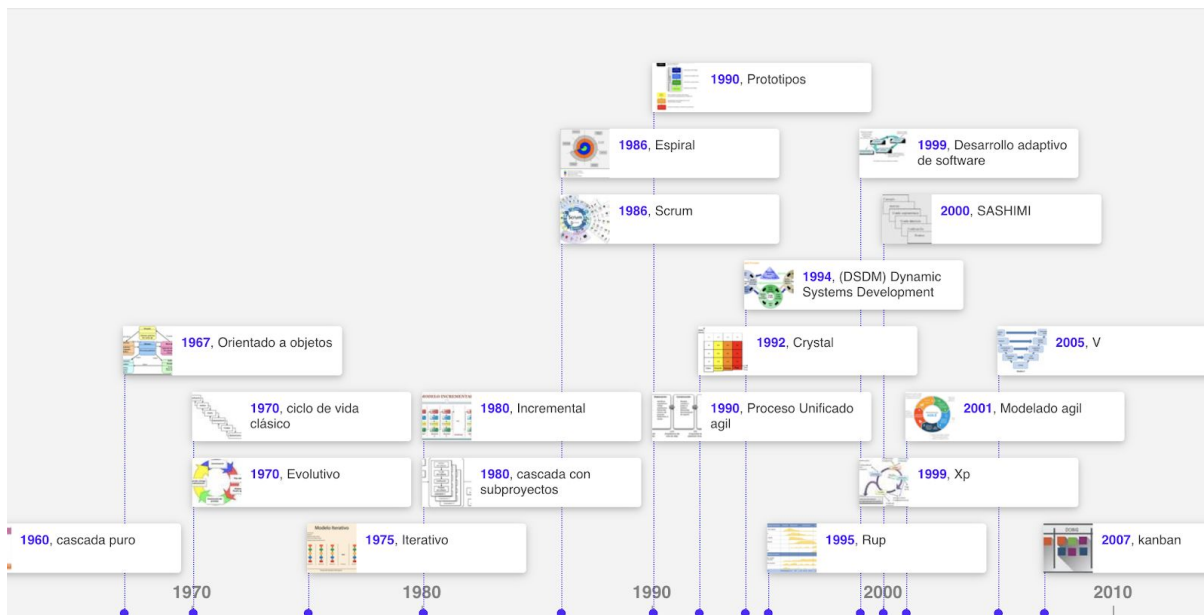


Figura 1.1 Línea de tiempo de las metodologías de planificación [TIMETOAST-WEB]

Como se puede ver en la figura 1.1, la época más lejana a la actual presenta principalmente metodologías tradicionales, pero según vamos avanzando en el tiempo vemos como cada vez estas van degradándose para convertirse en ágiles. Actualmente las



metodologías tradicionales están casi en desuso, pero se siguen utilizando en algunos lugares.

Centrándonos un poco más en el tema de las planificaciones, vamos a explicar más a fondo lo que es una planificación tradicional y lo que es una planificación ágil, realizando una comparación entre las características de ambas.

### **Planificaciones tradicionales:**

-Las planificaciones tradicionales, se basan principalmente en la previsibilidad, es decir, al inicio del proyecto se establece una planificación intocable, esto conlleva muchos riesgos en caso de que algo falle, ya que los costes, duración y demás parámetros pueden incrementarse notablemente hasta el punto de que el proyecto ya no sea viable y sea necesario su aborto.

-Planteamiento en cascada, una tarea depende de otra.

-Durante todo el desarrollo del proyecto existe una inquietud de si todo irá según lo previsto.

-Tratan de esconder el error para que el proyecto siga con su cauce normal.

-Tienen normalmente una efectividad baja, no es recomendable aplicarlas a no ser que se tenga mucha experiencia con un tipo determinado de producto o proyecto y ya se sepa estimar correctamente todos y cada uno de sus parámetros.

-Es recomendable su uso en proyectos donde los requisitos son muy claros y no cambian a lo largo del desarrollo del proyecto.

### **Planificaciones ágiles:**

-Las planificaciones ágiles, son todo lo contrario a las tradicionales y se basan en la adaptabilidad, cuando se elige este tipo de planificación o metodología se pretende mantener un contacto firme con el cliente.

-Suelen dividirse en iteraciones, cada iteración tiene un resultado tangible, es decir, en cada una de ellas se va entregando poco a poco secciones del proyecto totalmente funcionales para obtener de ellas una retroalimentación que sirva para mejorar en las siguientes iteraciones.

-Tratan de encontrar el error para solucionarlo cuanto antes.

-Tienen normalmente una efectividad alta, debido al firme contacto con el cliente que permite adaptarse fácil e inmediatamente a cualquier cambio, o corregir un error que surja durante el desarrollo.

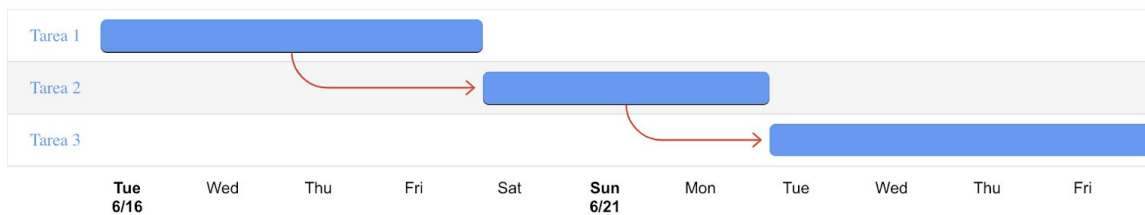
-Es recomendable su uso en proyectos en los que los requisitos puedan cambiar en el tiempo o no están claramente definidos.

### **Conclusión:**

En mi opinión y enfocando esto a los proyectos informáticos, son mucho más usables las metodologías ágiles, ya que actualmente estamos ante un mundo tecnológico muy cambiante y en constante desarrollo.

Como hemos comentado anteriormente, la aplicación software que se trata de construir en este TFG permitirá al alumno realizar planificaciones y simular dichas planificaciones corrigiendo posibles inconvenientes que vayan sucediéndose en el desarrollo de la misma, por tanto en dicha aplicación se simularán planificaciones en cascada dependientes de una serie de factores que se comentarán más adelante y que se pueden modificar en cualquier momento, por tanto, estamos ante algo evolutivo, es decir, las planificaciones simuladas serán con planteamiento ágil.

Así se muestra una planificación en la aplicación desarrollada:



*Figura 1.2. Ejemplo de planificación en la app desarrollada.*

### 1.2.2 Software relacionado

#### **Microsoft Project**

Microsoft Project es un software de administración de proyectos desarrollado por Microsoft, este software es utilizado para contribuir a un buen desarrollo de planificaciones, asignación de recursos a tareas, seguimientos, costes etc. [MS Project-Wikipedia]

Principalmente me he basado en este software para la realización de las planificaciones, es decir, en dividir el trabajo en tareas y asignar a cada una de ellas un tipo de recurso. Además he considerado que Project no implementa bien las simulaciones de planificaciones, es decir, únicamente realiza estimaciones sobre las cuales el usuario puede interactuar.

# Capítulo 2. Análisis de requisitos.

## 2.1 Contexto general.

Como ya hemos comentado, el objetivo de este proyecto es desarrollar una herramienta web que permita la simulación de proyectos software en el contexto del desarrollo de los laboratorios de las asignaturas de Planificación de Proyectos. En ella intervienen profesores y alumnos de la asignatura de las citadas asignaturas, donde los alumnos plasman una planificación de un determinado proyecto a través de unas plantillas de proyecto predeterminadas definidas por el profesor. Una vez definida la planificación, la aplicación permitirá simular la misma replanificando cuando sea necesario para finalmente obtener unos resultados que se pueden descargar en formato CSV, para que el profesor los trate como él crea oportuno.

En los siguientes apartados de este capítulo, comentaremos los requisitos que hemos definido para dicha aplicación, dividiéndolos en requisitos funcionales, requisitos no funcionales y requisitos de información, a su vez, añadiremos un diagrama de casos de uso de la aplicación junto con la oportuna matriz de incidencia que surgen con dichos casos de uso y los requisitos definidos.

Para cada tipo de requisitos, se añade una breve explicación de que es ese tipo de requisitos, además de una tabla en la que cada requisito posee una fila con los siguientes parámetros:

- ID: un identificador que será útil en la matriz de incidencia.
- Nombre: nombre del requisito.
- Descripción: descripción del requisito.
- Prioridad: prioridad del requisito, que puede ser: Alta, Media o Baja.

## 2.2 Requisitos Funcionales

Los requisitos funcionales son aquellos requisitos que establecen el comportamiento del Software, es decir, define una función del sistema de software o sus componentes, requisitos funcionales típicos pueden ser cálculos, detalles técnicos, manipulación de datos y otras funcionalidades específicas que se supone, un sistema debe cumplir. [*Requisito no funcional-Wikipedia*]

En la siguiente tabla se pueden contemplar los requisitos funcionales que hemos obtenido al analizar los casos de uso que se exponen más adelante.

ID	Nombre	Descripción	Prioridad
F01	Usuarios	El sistema permitirá registrar usuarios	Alta
F02	Plantillas de proyecto	El sistema permitirá crear y albergar plantillas de proyecto.	Alta
F03	Planificación de proyecto	Las planificaciones de proyecto se basarán en las plantillas y a su vez asignarán duraciones y recursos a las tareas.	Alta
F04	Modificación de la planificación	El sistema deberá permitir modificar la planificación inicial si esta lo requiere.	Alta
F05	Validación de la planificación	El sistema permitirá validar al profesor la planificación creada por el alumno antes de comenzar la simulación.	Alta
F06	Consulta planificación	El sistema permitirá consultar el diagrama Pert de cada planificación realizada.	Alta
F07	Simulación de la planificación	El sistema permitirá simular una planificación por parte del alumno.	Alta
F08	Monitorización de la planificación	El sistema permitirá monitorizar el estado de cada planificación en detalle por parte de un profesor en tiempo real.	Media
F09	Visualización de resultados	El sistema permitirá visualizar los resultados de una simulación.	Alta
F010	Descarga de resultados	El sistema permitirá descargar los resultados de una simulación.	Media
F11	Visualización de plantillas creadas.	El sistema permitirá ver las plantillas de proyecto predeterminadas que existen.	Alta

F12	Visualización del detalle de plantillas creadas.	El sistema permitirá ver en detalle las plantillas de proyecto predeterminadas que existen.	Media
F13	Almacenamiento de datos	El sistema deberá almacenar todos los datos relativos a las plantillas, usuarios, proyectos y simulaciones que se vayan realizando.	Alta
F14	Deshacer y Eliminar	El sistema permitirá eliminar o modificar cambios que se han propuesto en cualquier paso de la creación o simulación de las planificaciones.	Alta
F15	Eliminación de la planificación (running)	El sistema permitirá eliminar todos los datos acerca del proyecto creado cuando está en simulación o cuando esta aún no ha comenzado.	Alta
F16	Eliminación de la planificación (finish)	El sistema eliminará de forma automática todos los datos acerca del proyecto cuando el profesor haya realizado una descarga del archivo de resultados	Alta
F17	Roles de usuario	El sistema proporcionará 2 tipos de usuarios, alumnos o grupos de alumnos y profesores.	Alta
F18	Porcentaje completado de tareas en simulación	El sistema deberá permitir observar el estado en el que se encuentra cada tarea cuando hay una iteración de la simulación	Media

*Tabla 2.1 requisitos funcionales.*

## 2.3 Requisitos No Funcionales

Los requisitos no funcionales son aquellos requisitos que no definen información a guardar, ni funciones a realizar, sino características de funcionamiento, algunos requisitos funcionales pueden estar relacionados con el rendimiento, con la escalabilidad, con la estabilidad, con la seguridad, etc.

En la siguiente tabla se pueden contemplar los requisitos funcionales que hemos obtenido al analizar los casos de uso que se exponen más adelante.

ID	Nombre	Descripción	Prioridad
NF01	Aplicación web	El sistema deberá implementarse en forma de aplicación web.	Media
NF02	Formato resultados simulación	El sistema deberá proporcionar los resultados en formato CSV.	Media
NF03	Base de Datos no relacional	La aplicación almacenará sus datos en una Base de Datos no relacional	Alta
NF04	Uso de un framework moderno	La aplicación deberá desarrollarse usando un framework relativamente moderno, como puede ser Angular, Laravel, etc	Alta
NF05	Compatibilidad con navegadores	El sistema podrá utilizarse en cualquiera de los navegadores más utilizados, como son google chrome, safari, opera y firefox.	Media
NF06	Usabilidad	El sistema deberá ser intuitivo y usable, un usuario podrá usar la aplicación con facilidad recibiendo una breve introducción o leyéndose el manual adjunto.	Alta
NF07	Plantillas	Las plantillas de un proyecto tendrán un nombre y descripción. Estarán formadas por recursos y tareas.	Alta
NF08	Usuarios	Un usuario deberá tener un nombre, un email y un correo electrónico. Además de un rol ( profesor o usuario).	Alta

*Tabla 2.2 requisitos no funcionales.*

### 2.3.1 Restricciones

En lo que sigue, se expone una tabla también de requisitos no funcionales, pero que he decidido llamarla tabla de restricciones, ya que más que características del sistema, son propiamente restricciones, es decir, limitaciones en determinados asuntos del sistema, que se han decidido implantar según se ha ido desarrollando.

ID	Nombre	Descripción	Prioridad
R01	Un proyecto por alumno	Cada alumno únicamente podrá tener un proyecto activo dentro del sistema	Alta
R02	Restricciones alumno	Un alumno únicamente podrá planificar, simular y replanificar un proyecto	Alta
R03	Restricciones profesor	Un profesor únicamente podrá crear usuarios y plantillas y validar planificaciones del alumno, además de monitorizar las simulaciones.	Alta

*Tabla 2.3 requisitos no funcionales (restricciones).*

## 2.4 Usabilidad de la aplicación y diagramas de Casos de Uso

### 2.4.1 Análisis del funcionamiento general de la aplicación.

En la figura 2.1 se muestra un esquema global de uso de la aplicación y detallamos en ámbitos de usabilidad y para enfocarlo a obtener los casos de uso su funcionamiento general. Dicho esquema muestra lo que sería una iteración normal del sistema.

En la aplicación existen dos roles de usuario, estos son profesor o administrador y alumno.

El profesor será el usuario que administre las cuentas de usuario del sistema y que a su vez cree plantillas de planificaciones, apruebe las planificaciones realizadas por los alumnos y monitorice las simulaciones de esas planificaciones, cuando estas acaben, el profesor podrá imprimir un fichero en formato csv con los resultados de la simulación.

El alumno será el encargado de crear planificaciones con las tareas de la plantilla y asignarles recursos para realizar una simulación de esa planificación, basándose en una planificación previa realizada en otros programas de planificación de proyectos como Gantt Project o Microsoft Project, etc. Mientras esta simulación esté teniendo lugar, el alumno podrá modificar y reasignar o crear nuevos recursos para que esta simulación obtenga mejores resultados.

Un flujo normal de uso de la aplicación será:

- 1- El profesor crea una plantilla en el sistema, compuesta de tipos de recursos y tipos de tareas.
- 2- El alumno crea un proyecto en un programa de planificación de proyectos auxiliar.

3- El alumno inserta dicho proyecto en el sistema.

4- El profesor valida el plan.

## 5. BUCLE DE PLANIFICACIÓN

5.1- El alumno simula un paso de la planificación de dicho proyecto.

5.2- El tutor monitoriza.

5.3- El alumno añade cambios en el software auxiliar.

5.4- El alumno añade cambios en el sistema y de nuevo se realiza el bucle hasta finalizar el proyecto.

6- Se obtienen los resultados de la simulación en formato CSV.

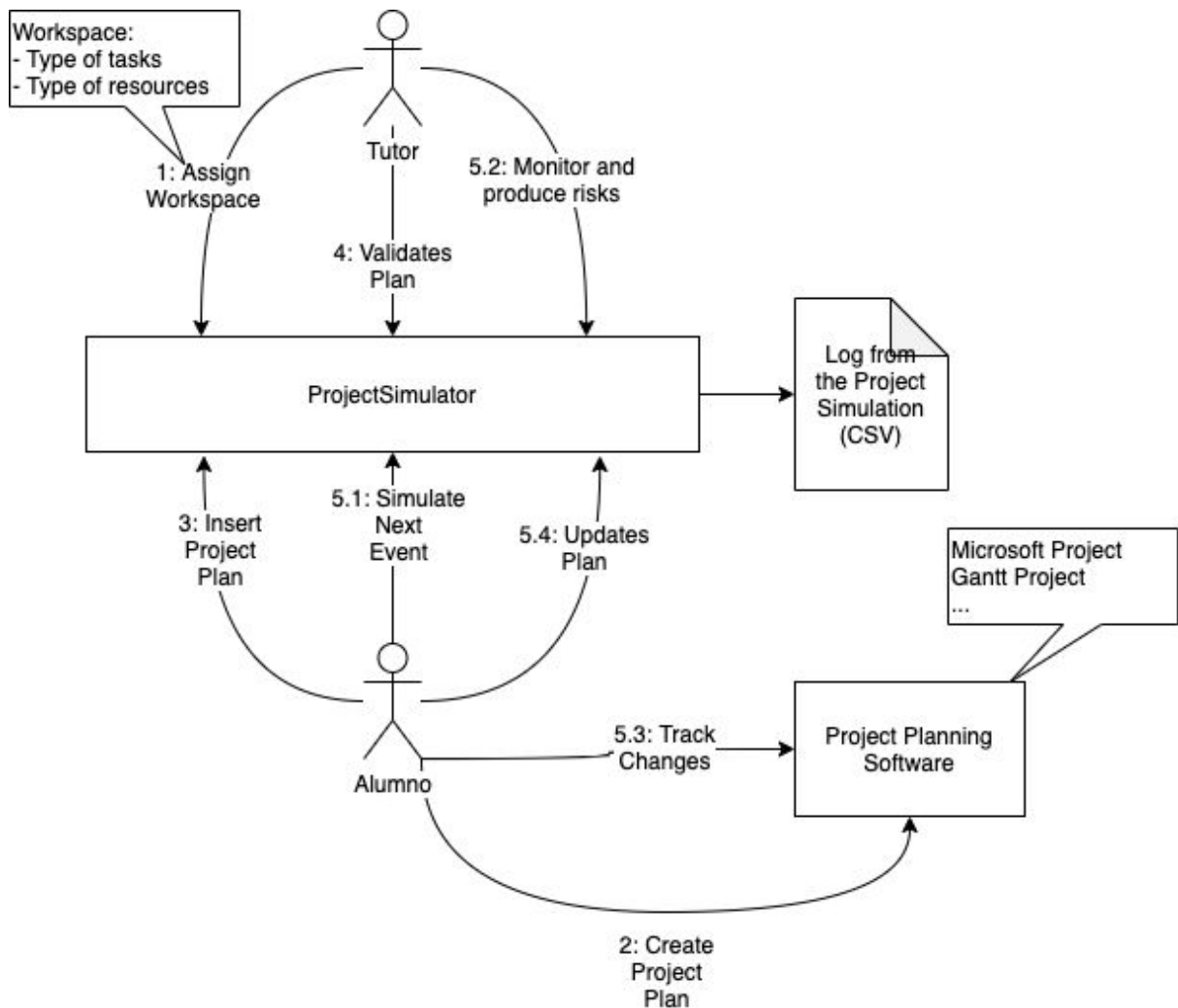


Figura 2.1. Esquema uso sistema a desarrollar.



## 2.4.2 Casos de uso Profesor.

A continuación se pueden observar el diagrama de casos de uso para un usuario con el rol Profesor. Como se puede observar, los casos de uso P2, P5 y P8, tienen que ver con la administración de usuarios del sistema, a su vez los casos de uso P1 y P4 tienen que ver con la creación de plantillas para las planificaciones de los alumnos. Por último, destacar que el resto tienen que ver con la monitorización de las planificaciones y simulaciones.



Figura 2.2. Casos de uso usuario tipo profesor.

### 2.4.3 Casos de uso Alumno.

A continuación se pueden observar el diagrama de casos de uso para un usuario con el rol Alumno. Como se puede ver, los casos de uso A1, A2, A3 tienen relación con la creación y modificación de los datos de la planificación, por otro lado los casos de uso A4, A5, A6, A7 y A8 tienen que ver con la propia simulación y para finalizar, los casos de uso A9, A10 y A11 tienen que ver con la post-simulación y resultados.

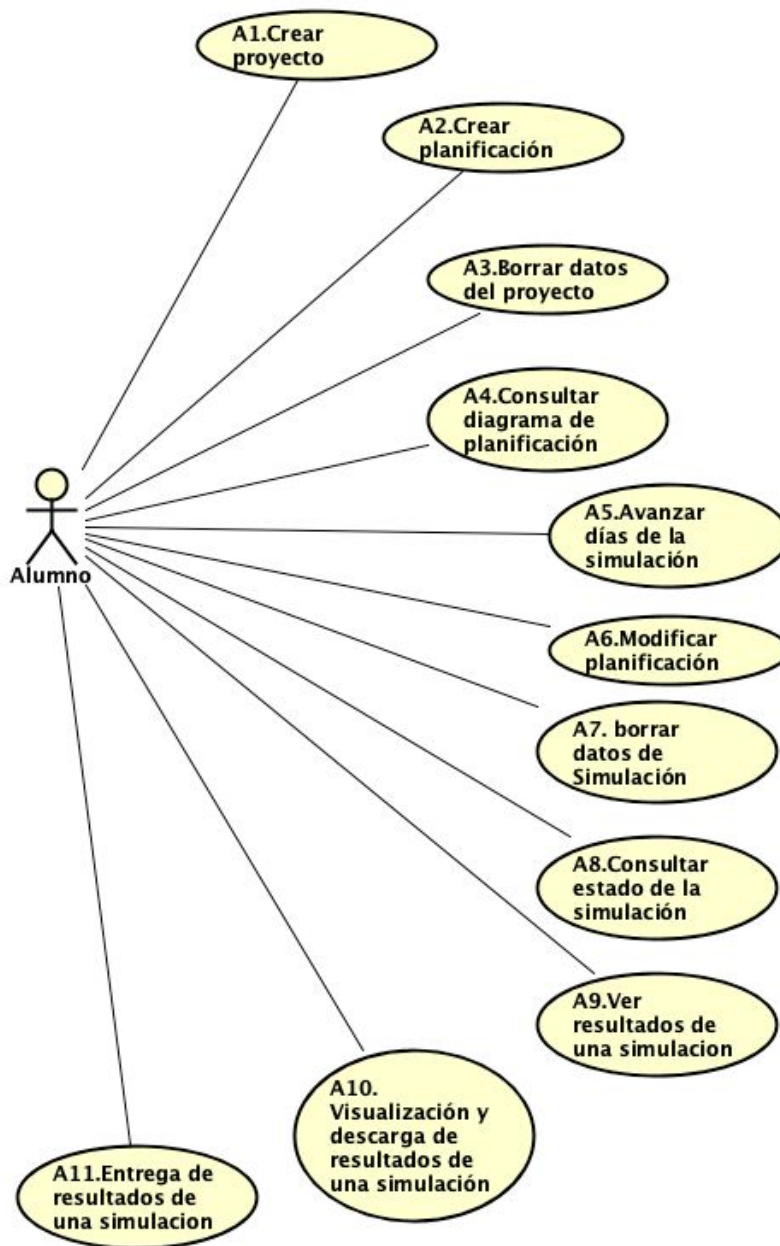


Figura 2.3. Casos de uso usuario tipo alumno.



<b>4</b>																		
<b>A5</b>				X	X	X	X											
<b>A6</b>													X					
<b>A7</b>																X		
<b>A8</b>																		
<b>A9</b>							X											
<b>A10</b>											X							
<b>A11</b>								X										

*Tabla 2.4 Matriz de incidencia entre requisitos funcionales y casos de uso.*

# Capítulo 3. Plan de trabajo y seguimiento.

## 3.1 Contexto y planificación seguida.

Para que un proyecto sea viable y funcione según lo previsto es necesaria una buena planificación, en nuestro caso hemos elegido una planificación por prototipos, en la cual cada prototipo es un tablero kanban con requisitos a implementar surgidos de una fase previa de estudio, análisis y planificación, que pueden variar con una retroalimentación. He tomado este modelo de planificación, debido a que el tutor, al comienzo del proyecto, no tenía muy claro cuales iban a ser los requisitos finales y qué iba a salir de este desarrollo, por tanto considero que esta planificación podría ser muy adaptativa en el futuro.

## 3.2 Detalle de las diferentes tareas o fases.

En este subcapítulo se detalla el plan que se ha llevado a cabo para la realización de este proyecto, junto con cada una de sus fases.

### 3.2.1 Fase previa.

La fase previa se utiliza para analizar los requisitos y realizar la planificación del proyecto, así como estudiar las tecnologías usadas para la elaboración del mismo.

-Se ha tomado como base una media de trabajo de 3h efectivas diarias, trabajando solo diariamente, descansando fines de semana y días festivos.

-La fecha de inicio de elaboración del TFG es el día 3 de Febrero de 2020.

-La fecha estimada de finalización será el día 5 de Junio de 2020.

La estimación para esta fase ha sido la siguiente:

-Duración: 30 h

-Fecha de inicio: 03/02/2020

-Fecha de fin: 14/02/2020

-Tareas:

-Planificación (6 horas).

-Análisis de requisitos y casos de uso (7 horas).

-Modelo de dominio (5 horas).

-Estudio arquitectura del sistema (3 horas).

-Estudio de angular 8 y bootstrap (9 horas).

### 3.2.2 Primer prototipo

Para realizar este prototipo se ha utilizado un tablero kanban con las actividades indicadas a continuación, para las que se ha estimado una duración determinada en cada una de ellas.

La estimación para esta fase ha sido la siguiente:

-Duración: 45 h

-Fecha de inicio: 17/02/2020

-Fecha de fin: 06/03/2020

-Tareas:

-Desarrollo sistema de Login de usuarios (20 horas).

-Desarrollo sistema de Registro de usuarios (15 horas).

-Creación paneles de los diferentes tipos de usuarios (10 horas).

### 3.2.3 Segundo prototipo

Para realizar este prototipo se ha utilizado un tablero kanban con las actividades indicadas a continuación, para las que se ha estimado una duración determinada en cada una de ellas.

La estimación para esta fase ha sido la siguiente:

-Duración: 60 h

-Fecha de inicio: 06/03/2020

-Fecha de fin: 03/04/2020

-Tareas:

-Corrección bugs primer prototipo con feedback del profesor (15 horas).

-Desarrollo funcionalidades creación plantillas de proyectos y creación de planificaciones de proyectos (30 horas).

-Desarrollo interfaz diagrama Gantt de las planificaciones(15 horas).

### 3.2.4 Tercer Prototipo

Para realizar este prototipo se ha utilizado un tablero kanban con las actividades indicadas a continuación, para las que se ha estimado una duración determinada en cada una de ellas.

La estimación para esta fase ha sido la siguiente:

-Duración: 75 h

-Fecha de inicio: 03/04/2020

-Fecha de fin: 15/05/2020

-Tareas:

- Corrección bugs segundo prototipo con feedback del profesor (15 horas).
- Estudio generación eventos discretos y probabilidad Pert (10 horas).
- Desarrollo algoritmo simulación (50 horas).

### 3.2.5 Prototipo final

Para realizar este prototipo se ha utilizado un tablero kanban con las actividades indicadas a continuación, para las que se ha estimado una duración determinada en cada una de ellas.

La principal tarea en este prototipo final es la realización de pruebas y solución de bugs, para obtener el producto final del desarrollo.

La estimación para esta fase ha sido la siguiente:

-Duración: 45 h

-Fecha de inicio: 15/05/2020

-Fecha de fin: 05/06/2020

-Tareas:

- Elaboración batería de pruebas y ejecución (25 horas).
- Corrección de errores (20 horas).

### 3.2.6 Diagrama de Gantt

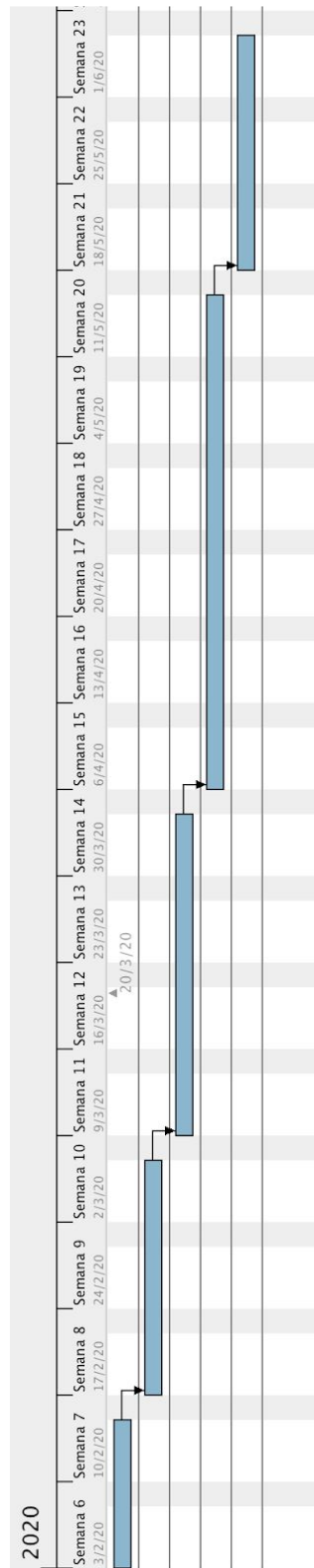


Figura 3.1. Diagrama de Gantt planificación TFG.



## 3.3 Gestión de Riesgos

### 3.3.1 Contexto gestión de riesgos

La gestión de riesgos es una técnica o parte de la gestión de un proyecto en la cual se maneja la incertidumbre relativa a una amenaza con el objetivo de reducir el posible impacto negativo del riesgo en el propio desarrollo del proyecto.

La gestión continuada de riesgos permite aumentar la eficiencia de su estudio y gestión, para realizar este continuo análisis debemos evaluar constantemente lo que puede ir mal, evaluar los riesgos y determinar estrategias de resolución y asegurar la eficacia de estas estrategias.

En este subcapítulo analizaremos los posibles riesgos que puedan darse en la realización de este TFG, cada riesgo analizado contendrá una serie de campos o atributos relevantes, que nos permitirán clasificarlos e identificarlos para una buena gestión de ellos, estos campos serán los que se detallan a continuación:

1. Nombre: nombre significativo del riesgo.
2. Descripción: breve descripción del riesgo.
3. Tipo: debido a que este TFG es un proyecto software, clasificaremos los riesgos según los siguientes tipos:
  - a. Riesgos del Proyecto: son los riesgos que afectan a la planificación y al coste del proyecto, pueden identificar problemas graves de presupuesto, calendario, personal y recursos.
  - b. Riesgos Técnicos: son los riesgos que amenazan la calidad del software y la funcionalidad que debería implementar el mismo, pueden identificar problemas en cuanto al diseño e implementación, en cuanto al mantenimiento del sistema, etc.
  - c. Riesgos de Negocio: son los riesgos que amenazan que el sistema desarrollado sea viable, incluyen riesgos de mercado, riesgos de ventas, riesgos de presupuesto, cambios en los requerimientos, etc.
4. Probabilidad de ocurrencia: probabilidad de que el riesgo ocurra. Normalmente la probabilidad tiene una relación inversa con el impacto del riesgo, es decir, si un riesgo tiene probabilidad alta de ocurrencia, es muy probable que su impacto sea leve.
5. Impacto en el desarrollo proyecto: este campo mide una magnitud de incisión del riesgo en el proyecto en caso de que este ocurra, en esta gestión se medirá a través de la siguiente escala, que es la más común en proyectos software, de menor a mayor impacto: despreciable < marginal < crítico < catastrófico, siendo el primero un impacto casi intangible y este último un impacto desastroso para el desarrollo correcto del proyecto.

6. **Estrategia de prevención:** en este campo comentamos la estrategia de prevención que se usará para intentar evitar el riesgo concreto.
7. **Consecuencias y plan de acción:** en este campo se medirán las consecuencias que la aparición del riesgo lleve consigo y se detalla un plan de acción que sirva para paliar dichas consecuencias sin perder la viabilidad del proyecto.

### 3.3.2 Plan de gestión de riesgos

<b>Nombre</b>	Incumplimiento de la planificación
<b>Descripción</b>	La planificación realizada sufre variaciones y retrasos que afectan negativamente al desarrollo del proyecto.
<b>Tipo</b>	Riesgo de proyecto
<b>Probabilidad Ocurrencia</b>	60%
<b>Impacto en el desarrollo del proyecto</b>	Crítico
<b>Estrategia de prevención</b>	Realizar un estudio muy exhaustivo de la planificación del proyecto, e intentar que sea lo más exacta posible dejando un margen de fallo disponible.
<b>Consecuencias</b>	Retraso en las fechas de desarrollo y despliegue, que conllevan un retraso en el normal camino del proyecto.
<b>Plan de acción</b>	Reorganizar el proyecto para conseguir subsanar los retrasos en la planificación.

*Tabla 3.1 Detalle del riesgo incumplimiento de la planificación.*

<b>Nombre</b>	Gestión pobre de los requisitos
<b>Descripción</b>	Mala gestión y análisis de los requisitos del proyecto que se quiere desarrollar.
<b>Tipo</b>	Riesgo de proyecto
<b>Probabilidad Ocurrencia</b>	40%
<b>Impacto en el desarrollo del proyecto</b>	Crítico
<b>Estrategia de prevención</b>	Realizar un estudio muy exhaustivo de los requisitos del proyecto, preguntando al cliente las veces que sea necesario si existiera algún tipo de duda.
<b>Consecuencias</b>	Retraso en las fechas de desarrollo y despliegue, que conllevan un retraso en el normal camino del proyecto.
<b>Plan de acción</b>	Reorganizar el proyecto para conseguir subsanar los requisitos que no se habían contemplado.

*Tabla 3.2 Detalle del riesgo gestión pobre de los requisitos.*

<b>Nombre</b>	Falta de comunicación
<b>Descripción</b>	Mala compenetración y comunicación entre los integrantes que forman en el equipo, en este caso entre el tutor y el alumno.
<b>Tipo</b>	Riesgo de proyecto
<b>Probabilidad Ocurrencia</b>	30%
<b>Impacto en el desarrollo del proyecto</b>	Crítico
<b>Estrategia de prevención</b>	Realizar un estudio muy exhaustivo de la planificación del proyecto, e intentar que sea lo más exacta posible dejando un margen de fallo disponible.
<b>Consecuencias</b>	Retraso en las fechas de desarrollo y despliegue, que conllevan un retraso en el normal camino del proyecto.
<b>Plan de acción</b>	Intentar mantener una comunicación estable con el tutor del proyecto, que en este caso es el propio cliente.

*Tabla 3.3 Detalle del riesgo falta de comunicación.*

<b>Nombre</b>	Indisponibilidad del alumno o tutor
<b>Descripción</b>	Cualquier indisponibilidad por parte del alumno o tutor, retrasaría la planificación inicial del proyecto
<b>Tipo</b>	Riesgo de proyecto
<b>Probabilidad Ocurrencia</b>	40%
<b>Impacto en el desarrollo del proyecto</b>	Marginal
<b>Estrategia de prevención</b>	Realizar un estudio muy exhaustivo de la planificación del proyecto, e intentar que sea lo más exacta posible dejando un margen de fallo disponible.
<b>Consecuencias</b>	Retraso en las fechas de desarrollo y despliegue, que conllevan un retraso en el normal camino del proyecto.
<b>Plan de acción</b>	Planificar de nuevo teniendo en cuenta los fallos cometidos para no volver a soportarlos

*Tabla 3.4 Detalle del riesgo indisponibilidad del alumno o tutor.*

<b>Nombre</b>	Indisponibilidad del equipo de programación
<b>Descripción</b>	Cualquier indisponibilidad del equipo de programación supone una parada y pérdida de días en el desarrollo del mismo.
<b>Tipo</b>	Riesgo de proyecto
<b>Probabilidad Ocurrencia</b>	15%
<b>Impacto en el desarrollo del proyecto</b>	Crítico
<b>Estrategia de prevención</b>	Disponer de más de un equipo de programación en los cuales se almacene el código actualizado o usen una plataforma de control de versiones en la que se almacene el proyecto actualizado.
<b>Consecuencias</b>	Imposibilidad de continuar con el desarrollo del proyecto.
<b>Plan de acción</b>	Seguir el proyecto desde otro equipo.

*Tabla 3.5 Detalle del riesgo Indisponibilidad hardware.*

<b>Nombre</b>	Cambio en los requisitos de la aplicación
<b>Descripción</b>	Cambio en los requisitos planificados inicialmente para la aplicación, tanto como requisitos nuevos que aparecen como requisitos que cambian de alguna forma.
<b>Tipo</b>	Riesgo de negocio
<b>Probabilidad Ocurrencia</b>	45%
<b>Impacto en el desarrollo del proyecto</b>	Marginal
<b>Estrategia de prevención</b>	Llevar a cabo un buen análisis del proyecto a desarrollar, teniendo claros sus requisitos desde el inicio del mismo.
<b>Consecuencias</b>	Demora en el tiempo de desarrollo.
<b>Plan de acción</b>	Planificar de nuevo el desarrollo del proyecto si fuera necesario, debido al alcance del requisito cambiado.

*Tabla 3.6 Cambio en los requisitos de la aplicación.*

<b>Nombre</b>	Falta de experiencia con las tecnologías de desarrollo.
<b>Descripción</b>	El desarrollador es nuevo en las tecnologías que se van a usar para implementar el proyecto.
<b>Tipo</b>	Riesgo Técnico
<b>Probabilidad Ocurrencia</b>	70%
<b>Impacto en el desarrollo del proyecto</b>	Riesgo Marginal
<b>Estrategia de prevención</b>	Realizar un amplio estudio de las tecnologías que se vayan a utilizar para realizar el proyecto.
<b>Consecuencias</b>	Demora en el tiempo de desarrollo.
<b>Plan de acción</b>	Aumento de las horas de estudio de la tecnología que cause problemas.

*Tabla 3.7 Detalle del riesgo Falta de experiencia con tecnologías de desarrollo.*

<b>Nombre</b>	Manual de usuario complejo
<b>Descripción</b>	Manual demasiado técnico y complejo para la utilización de los usuarios/clientes.
<b>Tipo</b>	Riesgo de negocio
<b>Probabilidad Ocurrencia</b>	40%
<b>Impacto en el desarrollo del proyecto</b>	Riesgo Marginal
<b>Estrategia de prevención</b>	Realizar un estudio de los conocimientos de los usuarios y adecuar el manual a ello.
<b>Consecuencias</b>	Difícil uso de la aplicación por parte de los usuarios
<b>Plan de acción</b>	Realizar sesiones formativas con los usuarios finales de la aplicación, en las cuales se detalle el funcionamiento de la misma y se instruya al uso del manual de instrucciones.

*Tabla 3.8 Detalle del riesgo Manual de usuario complejo.*

<b>Nombre</b>	Pérdida de la información
<b>Descripción</b>	Pérdida no intencionada de los datos del proyecto ocasionada por algún fallo eléctrico o de cualquier tipo.
<b>Tipo</b>	Riesgo de Proyecto
<b>Probabilidad Ocurrencia</b>	15%
<b>Impacto en el desarrollo del proyecto</b>	Catastrófico
<b>Estrategia de prevención</b>	Realizar continuamente copias de seguridad de todos los datos involucrados con el proyecto
<b>Consecuencias</b>	Pérdida de la información.
<b>Plan de acción</b>	Recuperar la última versión almacenada de los datos del proyecto.

*Tabla 3.9 Detalle del riesgo Pérdida de la información.*

<b>Nombre</b>	Pruebas poco efectivas
<b>Descripción</b>	La realización de la batería de pruebas ha sido poco efectiva y se han olvidado detalles de funcionalidad.
<b>Tipo</b>	Riesgo Técnico
<b>Probabilidad Ocurrencia</b>	50%
<b>Impacto en el desarrollo del proyecto</b>	Crítico
<b>Estrategia de prevención</b>	Diseñar un plan de pruebas cuidadosamente, fijándose en cada uno de los detalles que el sistema debe de cumplir.
<b>Consecuencias</b>	Entregar un artefacto o servicio con fallos o problemas.
<b>Plan de acción</b>	Modificar el plan de pruebas para detectar todos los errores posible y así poder solucionar los errores.

*Tabla 3.10 Detalle del riesgo Pruebas poco efectivas.*

## 3.4 Gestión de Costes

### 3.4.1 Introducción y contexto

La gestión de costes de un proyecto es el proceso por el cual se determinan, estiman y asignan los costes de un proyecto, este estudio permite a las empresas, personas o entidades tener una relación de costes antes del inicio del proyecto.

Según el [PMBOK], la gestión de costes se realiza en 4 fases :

Fase 1-Planificación del coste del proyecto: en esta fase se estudian las actividades que se van a realizar, junto con los materiales o recursos que estas necesitan.

Fase 2-Estimación de los costes del proyecto: una vez conocidos los recursos y bienes que se van a utilizar, se puede realizar un sondeo del coste del proyecto.

Fase 3-Estimación del presupuesto: el presupuesto sondea el total de los costes estimados.

Fase 4-Control de los costes: la monitorización en varios hitos determinados de la consecución de los gastos del proyecto también es muy importante.

En este capítulo se estudian los costes determinados de este proyecto.

### 3.4.2 Costes de recursos humanos

En la realización de este proyecto, solo ha sido necesaria la involucración de un ingeniero informático, teniendo en cuenta que el sueldo base por convenio en la provincia de valladolid en el año 2021 es de 21.008,47€ brutos anuales [BOPVA-A2020] y teniendo en cuenta la estimación de horas para la realización de este proyecto, se estima que:

1. Un ingeniero informático cobra 21008,47 €/1950 h anuales = 10,77€/h
2. El total de Costes humanos asciende a : 10,77 x 255 h =2746,35€

Descripción	Cantidad	Precio Unidad (€)
Ingeniero Informático	1	2746,35
TOTAL		2746,35

Tabla 3.11 Detalle del coste humano.

### 3.4.3 Costes hardware/software

Para realizar este proyecto, como hardware se ha usado un ordenador portátil MacBook Pro 2018, un teclado y un ratón, cuyos precios ascienden respectivamente a: 1499€, 16€ y 6€.

En el ámbito del software se ha utilizado una licencia de Astah UML para realizar el estudio del dominio y arquitectura y una licencia profesional de Microsoft Project, el precio asciende respectivamente a: 69\$ (62,01€) y 1300€.

Se han usado software de programación y hostings totalmente gratuitos, por tanto no se incluyen en este apartado.

A estos equipos y softwares, se les ha estimado una vida útil de 6 años, por tanto calculamos la parte proporcional de su coste:

-6 años = 72 meses.

-Tiempo de desarrollo de la aplicación: 4,5 meses.

Tabla de precios por mes y producto teniendo en cuenta lo anterior.

Descripción	Precio Mes	Precio 4,5 meses (€)
Macbook Pro	20,82€	93,69€
Teclado	0,22€	1€
Ratón	0,08€	0,38€
TOTAL	21,12€	95,07€

Tabla 3.12 Detalle del coste hardware.

Descripción	Precio Mes	Precio 4,5 meses (€)
Astah UML	0,86€	3,87€
Microsoft Project Professional	18,06€	81,25€
TOTAL	18,92€	85,12€

Tabla 3.13 Detalle del coste software.



### 3.4.4 Costes totales del proyecto

En la siguiente tabla se reúnen los costes totales de los subapartados anteriores, generando un presupuesto del proyecto a realizar.

Descripción	Cantidad	Precio Unidad (€)
Costes Humanos	1	2746,35 €
Costes Hardware	1	95,07 €
Costes Software	1	85,12 €
TOTAL		2926,54 €

*Tabla 3.14 Detalle del coste total.*

## 3.5 Seguimiento

### 3.5.1 Contexto y detalle

Como se ha comentado en el inicio de este capítulo, la realización de este proyecto debía de llevar consigo un seguimiento detallado ya que los requisitos finales no se llevaban muy claros desde el inicio del mismo y podrían variar incluso aparecer otros nuevos que no estaban ni tenidos en cuenta.

En la construcción del proyecto, aunque ya se tenía en cuenta en la planificación inicial un margen de horas en cada iteración de la misma para sobrellevar estos cambios que podían darse, ha sido necesaria una leve replanificación sobre la marcha, en la que las fechas planeadas de realización del proyecto no se han visto afectadas gravemente.

En el siguiente subapartado se muestra comparación de las horas estimadas frente a las horas reales, además, por cada cambio se añadirá una breve explicación del mismo.

### 3.5.2 Estimación vs Realidad

#### 3.5.2.1 Fase previa.

El principal cambio a destacar fué que las horas de estudio de las tecnologías aumentaron en 3 sobre las planificadas, esto fué debido a que nunca las había utilizado. En cuanto al análisis de requisitos y el modelo de dominio, se estimó más tiempo del que se ha tardado en completar la tarea.

Las horas totales han aumentado en 1 para las estimadas en esta fase, se hizo una hora extra un día y la fecha se mantuvo.

Tarea	Estimación (h)	Realidad (h)	Diferencia
Planificación	6	8	+2
Análisis requisitos y casos de uso	7	5	-2
Modelo de dominio	5	3	-2
Arquitectura del sistema	3	3	0
Estudio tecnologías	9	12	+3
TOTAL	30	31	+1

*Tabla 3.15 Estimación vs Realidad Fase Previa.*

### 3.5.2.2 Primer prototipo.

En este primer prototipo todo se desarrolló con normalidad, esto fue debido a que los requisitos estaban muy bien definidos y claros a la hora de desarrollar.

Tarea	Estimación (h)	Realidad (h)	Diferencia
Sistema login	20	20	0
Sistema registro	15	15	0
Panel usuario	10	10	0
TOTAL	45	45	0

*Tabla 3.16 Estimación vs Realidad Primer Prototipo.*

### 3.5.2.3 Segundo prototipo.

Este prototipo finalizó 4 días antes de lo previsto, esto fue debido a la sobreestimación de horas en corregir detalles del feedback del anterior prototipo, solo fueron necesarios unos breves cambios que llevaron 12h menos de lo previsto.

Fecha fin estimada: 03/04/2020

Fecha fin real : 30/03/2020

Tarea	Estimación (h)	Realidad (h)	Diferencia
Corrección feedback	15	3	-12
Creación de plantillas y de planificaciones	30	20	-10
Interfaz diagrama de Gantt de las planificaciones	15	25	+10
TOTAL	60	48	-12

*Tabla 3.17 Estimación vs Realidad Segundo Prototipo.*

### 3.5.2.4 Tercer prototipo.

Este prototipo comenzó 4 días antes de lo previsto, es decir el 31/03/2020. Los requisitos de la creación de plantillas y de planificaciones desarrolladas cambiaron y fue necesario invertir 12 h en su corrección, 3 menos de las previstas. Por otro lado en cuanto al desarrollo del algoritmo de simulación, costó más de lo previsto y esto se tradujo en 5 días más, la fecha de finalización de este prototipo no se modificó, debido a que comenzó 4 días antes.

Tarea	Estimación (h)	Realidad (h)	Diferencia
Corrección feedback	15	12	-3
Estudio eventos discretos y pert	10	10	0
Desarrollo algoritmo de simulación	50	65	+15
TOTAL	75	87	+12

*Tabla 3.18 Estimación vs Realidad Tercer Prototipo.*

### 3.5.2.5 Prototipo final.

Este prototipo empezó y acabo en la fecha prevista, tan sólo nos encontramos una ligera variación en el tiempo de la elaboración de la batería de pruebas que se dedicó a la corrección de algún error de desarrolló basado en el diseño que se encontró.

<b>Tarea</b>	<b>Estimación (h)</b>	<b>Realidad (h)</b>	<b>Diferencia</b>
Elaboración batería de pruebas y ejecución	25	20	-5
Corrección de errores	20	25	+5
TOTAL	45	45	0

*Tabla 3.19 Estimación vs Realidad Prototipo Final.*

# Capítulo 4. Análisis del dominio software.

## 4.1 Introducción

El análisis del dominio se define como una actividad dentro de la planificación gestión y desarrollo de un proyecto que consiste en identificar los objetos y operaciones de un tipo de sistemas particulares dentro de un dominio de problema particular.

En los siguientes subapartados se realiza un amplio estudio tanto del modelo de dominio de la aplicación como de los casos de uso que este abarca.

## 4.2 Definición de los Casos de uso

Basándome en la definición de los casos de uso identificados en el capítulo 2 de este documento, detallaremos a continuación cada uno de ellos, aportando además los correspondientes diagramas de secuencia más significativos de esta aplicación.

### 4.2.1 Casos de uso usuario tipo profesor

En todos los casos de uso detallados a continuación, se sobreentiende que el actor principal es un usuario con rol profesor.

<b>NOMBRE</b>	P1.Crear plantilla de proyectos.
<b>PRECONDICIÓN</b>	-El usuario está logueado como profesor en el sistema.
<b>POSTCONDICIÓN</b>	-Hay una nueva plantilla de proyecto creada y almacenada en el sistema.
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de creación de una plantilla de proyecto.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"><li>1. El usuario selecciona la opción Ver/crear plantillas de la barra de tareas.</li><li>2. El sistema muestra las plantillas creadas y da la opción de crear una nueva.</li><li>3. El usuario selecciona crear una nueva plantilla.</li><li>4. El sistema muestra una introducción e instrucciones.</li><li>5. El usuario selecciona comenzar.</li><li>6. El sistema muestra un formulario de identificación de la plantilla.</li><li>7. El usuario lo rellena y pulsa continuar.</li><li>8. El sistema muestra una ventana de creación de recursos.</li><li>9. El usuario pulsa en añadir recurso tantas veces como recursos quiera añadir, después rellena sus campos y selecciona continuar.</li><li>10. El sistema muestra un formulario de tareas.</li></ol>

	<ol style="list-style-type: none"> <li>11. El usuario pulsa en añadir tarea tantas veces como tareas quiera añadir, después rellena sus campos y selecciona continuar.</li> <li>12. El sistema muestra un formulario de asignación mínima de recursos a tareas.</li> <li>13. El profesor lo cumplimenta y pulsa crear.</li> <li>14. El sistema solicita confirmación de creación.</li> <li>15. El usuario la acepta.</li> <li>16. El sistema almacena la plantilla y este caso de uso finaliza.</li> </ol>
<b>EXCEPCIONES</b>	<ul style="list-style-type: none"> <li>• 7,9,11,13. El usuario pulsa volver.</li> <li>• 7A,9A,11A,13A. El sistema muestra el formulario anterior respectivamente.</li> <li>• 15. El usuario selecciona cancelar.</li> <li>• 15A. El caso de uso vuelve al paso 12.</li> </ul>

*Tabla 4.1 Detalle caso de uso P1.*

<b>NOMBRE</b>	P2.Registro de usuarios.
<b>PRECONDICIÓN</b>	-El usuario está logueado como profesor en el sistema. -El usuario no existe ya en el sistema.
<b>POSTCONDICIÓN</b>	-Hay un nuevo usuario o grupo de usuarios creado y almacenado en el sistema.
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de creación de un usuario o grupo de usuario del sistema.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Ver/Registrar usuarios de la barra de tareas.</li> <li>2. El sistema muestra los usuarios existentes y da la opción de crear uno nuevo.</li> <li>3. El usuario selecciona crear usuario.</li> <li>4. El sistema muestra un formulario de registro.</li> <li>5. El usuario introduce los datos y selecciona Registrar.</li> <li>6. El sistema solicita confirmación de registro.</li> <li>7. El usuario la acepta.</li> <li>8. El sistema almacena la plantilla y este caso de uso finaliza.</li> </ol>
<b>EXCEPCIONES</b>	<ul style="list-style-type: none"> <li>• 5.1. El usuario selecciona volver.</li> <li>• 5.1A. El caso de uso retrocede al paso 2.</li> <li>• 5.2. El usuario introduce el email de un usuario que ya está en el sistema.</li> <li>• 5.2A. El sistema lo notifica y permanece en el paso 5.</li> <li>• 7. El usuario selecciona cancelar.</li> <li>• 7A. El caso de uso vuelve al paso 5.</li> </ul>

*Tabla 4.2 Detalle caso de uso P2.*

<b>NOMBRE</b>	P3. Ver proyecto en simulación.
<b>PRECONDICIÓN</b>	-El usuario está logueado como profesor en el sistema.
<b>POSTCONDICIÓN</b>	NINGUNA
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de visualización de una planificación en simulación por parte de un profesor.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Ver proyectos en simulación de la barra de tareas.</li> <li>2. El sistema muestra un resumen de los proyectos que actualmente están en simulación o que ya han finalizado y están sin entregar por parte del alumno.</li> <li>3. El usuario selecciona el proyecto que quiere ver en detalle</li> <li>4. El sistema muestra una tabla de detalle de la simulación y el caso de uso finaliza.</li> </ol>
<b>EXCEPCIONES</b>	<ul style="list-style-type: none"> <li>• 2. No existen proyectos en simulación en el sistema.</li> <li>• 2A. El sistema notifica que no existen proyectos en simulación en este momento.</li> </ul>

*Tabla 4.3 Detalle caso de uso P3.*

<b>NOMBRE</b>	P4.Ver plantillas creadas.
<b>PRECONDICIÓN</b>	-El usuario está logueado como profesor en el sistema.
<b>POSTCONDICIÓN</b>	NINGUNA
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de visualización de una plantilla existente en el sistema por parte de un profesor.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Ver/crear plantillas de la barra de tareas.</li> <li>2. El sistema muestra un resumen de las plantillas que actualmente existen en el sistema.</li> <li>3. El usuario selecciona la plantilla que quiere ver en detalle.</li> <li>4. El sistema muestra una tabla de detalle de la plantilla y el caso de uso finaliza.</li> </ol>
<b>EXCEPCIONES</b>	<ul style="list-style-type: none"> <li>• 2. No existen plantillas definidas en el sistema.</li> <li>• 2A. El sistema notifica que no existen plantillas en este momento.</li> </ul>

*Tabla 4.4 Detalle caso de uso P4.*

<b>NOMBRE</b>	P5.Ver usuarios del sistema.
<b>PRECONDICIÓN</b>	-El usuario está logueado como profesor en el sistema.
<b>POSTCONDICIÓN</b>	NINGUNA
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de visualización de los usuarios existentes en el sistema.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Ver/Registrar usuarios de la barra de tareas.</li> <li>2. El sistema muestra los usuarios existentes y el caso de uso finaliza.</li> </ol>
<b>EXCEPCIONES</b>	<ul style="list-style-type: none"> <li>• 2. No existen usuarios registrados en el sistema.</li> <li>• 2A. El sistema notifica que no existen usuarios registrados en este momento.</li> </ul>

*Tabla 4.5 Detalle caso de uso P5.*

<b>NOMBRE</b>	P6.Vista de resultados de una simulación.
<b>PRECONDICIÓN</b>	-El usuario está logueado como profesor en el sistema. -Existe una simulación finalizada en el sistema.
<b>POSTCONDICIÓN</b>	NINGUNA
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de visualización de los resultados de una simulación finalizada en el sistema.
<b>SECUENCIA NORMAL</b>	La secuencia a seguir es la misma que el caso de uso P3.
<b>EXCEPCIONES</b>	Las excepciones son las mismas que el caso de uso P3.

*Tabla 4.6 Detalle caso de uso P6.*

<b>NOMBRE</b>	P7.Descarga de resultados de una simulación.
<b>PRECONDICIÓN</b>	-El usuario está logueado como profesor en el sistema. -Existe una simulación finalizada en el sistema.
<b>POSTCONDICIÓN</b>	-Existe un archivo csv en su computador que muestra los resultados de la simulación.
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de descarga de los



	resultados de una simulación finalizada en el sistema.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Ver/Registrar usuarios de la barra de tareas.</li> <li>2. El sistema muestra los resultados que han sido enviados junto con el nombre del grupo al que pertenece.</li> <li>3. El usuario solicita ver detalle de uno de ellos.</li> <li>4. El sistema muestra los resultados de dicha simulación.</li> <li>5. El usuario pulsa el botón descargar.</li> <li>6. El sistema descarga automáticamente un archivo csv en el computador del usuario .</li> </ol>
<b>EXCEPCIONES</b>	NINGUNA

*Tabla 4.7 Detalle caso de uso P7.*

<b>NOMBRE</b>	P8.Borrado de datos de un alumno del sistema.
<b>PRECONDICIÓN</b>	-El usuario está logueado como profesor en el sistema. -Existen usuarios en el sistema.
<b>POSTCONDICIÓN</b>	-El usuario eliminado ya no existe en el sistema.
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de borrado de usuario del sistema.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Ver/Registrar usuarios de la barra de tareas.</li> <li>2. El sistema muestra los usuarios existentes y da la opción de eliminar cualquiera de ellos.</li> <li>3. El usuario selecciona eliminar en el usuario deseado.</li> <li>4. El sistema solicita confirmación.</li> <li>5. El usuario la acepta.</li> <li>6. El sistema borra el usuario y notifica que la operación se a completado, el caso de uso finaliza.</li> </ol>
<b>EXCEPCIONES</b>	<ul style="list-style-type: none"> <li>• 5. El usuario selecciona cancelar.</li> <li>• 5A. El caso de uso vuelve al paso 2.</li> </ul>

*Tabla 4.8 Detalle caso de uso P8.*

<b>NOMBRE</b>	P9. Validar recursos planificación alumno.
<b>PRECONDICIÓN</b>	-El usuario está logueado como profesor en el sistema. -Existen usuarios en el sistema. -Existe un alumno con una planificación creada.
<b>POSTCONDICIÓN</b>	-El alumno ya puede asignar recursos a las tareas.
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de borrado de usuario del sistema.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Validación de Recursos de la barra de tareas.</li> <li>2. El sistema muestra las planificaciones que necesitan de esta acción.</li> <li>3. El usuario selecciona validar recursos en la planificación deseada.</li> <li>4. El sistema muestra una tabla con cada tipo de recurso y su cantidad modificable por parte del usuario.</li> <li>5. El usuario realiza o no las modificaciones oportunas y selecciona registrar selección.</li> <li>6. El sistema pide confirmación.</li> <li>7. El usuario la acepta.</li> <li>8. El sistema guarda la operación y el caso de uso finaliza.</li> </ol>
<b>EXCEPCIONES</b>	<ul style="list-style-type: none"> <li>• 5. El usuario selecciona cancelar.</li> <li>• 5A. El caso de uso vuelve al paso 2.</li> <li>• 6. El usuario cancela.</li> <li>• 6A. El caso de uso se mantiene en el paso 5.</li> </ul>

*Tabla 4.9 Detalle caso de uso P9.*

#### 4.2.2 Casos de uso usuario tipo alumno

<b>NOMBRE</b>	A1.Crear proyecto.
<b>PRECONDICIÓN</b>	-El usuario está logueado como alumno en el sistema. -El alumno no tiene ningún proyecto ya creado.
<b>POSTCONDICIÓN</b>	-El alumno tiene un proyecto creado en el sistema pendiente de validar.
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de creación de un proyecto en el sistema por parte de un alumno.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Crear Proyecto en su ventana de bienvenida.</li> <li>2. El sistema muestra un formulario de datos generales del proyecto.</li> <li>3. El usuario lo cumplimenta y pulsa en siguiente.</li> <li>4. El sistema muestra un formulario de recursos para que el alumno elija las cantidades correspondientes.</li> <li>5. El usuario cumplimenta el formulario y pulsa en crear.</li> <li>6. El sistema pide confirmación de creación.</li> <li>7. El usuario la acepta.</li> <li>8. El sistema almacena el nuevo proyecto y el caso de uso finaliza.</li> </ol>
<b>EXCEPCIONES</b>	<ul style="list-style-type: none"> <li>• 5. El usuario selecciona volver.</li> <li>• 5A. El caso de uso retorna al paso 2.</li> <li>• 6. El usuario declina la confirmación.</li> <li>• 6A. El caso de uso vuelve al paso 2.</li> </ul>

*Tabla 4.10 Detalle caso de uso A1.*

<b>NOMBRE</b>	A2.Crear planificación.
<b>PRECONDICIÓN</b>	-El usuario está logueado como alumno en el sistema. -El alumno ha creado un proyecto. -Un usuario profesor ha validado el proyecto creado por el alumno.
<b>POSTCONDICIÓN</b>	-El alumno tiene un proyecto planificado en el sistema listo para simular.
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de creación de una planificación de un proyecto en el sistema por parte de un alumno.

<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Añadir Tareas en su ventana de bienvenida.</li> <li>2. El sistema muestra un formulario de tareas.</li> <li>3. El usuario pulsa en añadir tarea tantas veces como tareas quiera añadir, después rellena sus campos y selecciona Asignar Recursos.</li> <li>4. El sistema muestra una ventana de asignación de recursos.</li> <li>5. El usuario selecciona una cantidad de recursos de cada tipo para asignar a cada tarea y pulsa crear.</li> <li>6. El sistema pide confirmación de creación.</li> <li>7. El usuario la acepta.</li> <li>8. El sistema almacena la nueva planificación del proyecto y el caso de uso finaliza.</li> </ol>
<b>EXCEPCIONES</b>	<ul style="list-style-type: none"> <li>• 5. El usuario selecciona volver.</li> <li>• 5A. El caso de uso retorna al paso 2.</li> <li>• 6. El usuario declina la confirmación.</li> <li>• 6A. El caso de uso vuelve al paso 2.</li> </ul>

*Tabla 4.11 Detalle caso de uso A2.*

<b>NOMBRE</b>	A3.Borrar Datos del Proyecto.
<b>PRECONDICIÓN</b>	-El usuario está logueado como alumno en el sistema. -El profesor ha dado el visto bueno a los recursos creados.
<b>POSTCONDICIÓN</b>	-El proyecto, planificación y simulación del mismo si existiera previamente, se eliminan por completo del sistema.
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento del borrado completo de los datos de un proyecto.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Borrar Datos del Proyecto en su ventana de bienvenida.</li> <li>2. El sistema pide confirmación de la operación.</li> <li>3. El usuario la acepta.</li> <li>4. El sistema elimina todos los datos del proyecto y el caso de uso finaliza.</li> </ol>
<b>EXCEPCIONES</b>	<ul style="list-style-type: none"> <li>• 3. El usuario declina la confirmación.</li> <li>• 3A. El caso de uso vuelve al paso 1.</li> </ul>

*Tabla 4.12 Detalle caso de uso A3.*

<b>NOMBRE</b>	A4.Consultar diagrama de planificación
<b>PRECONDICIÓN</b>	-El usuario está logueado como alumno en el sistema. -El alumno tiene un proyecto planificado o en estado de simulación.
<b>POSTCONDICIÓN</b>	NINGUNA
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de cómo un usuario de tipo alumno puede ver y consultar el diagrama de gantt de su planificación.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Ver estado en su ventana de bienvenida.</li> <li>2. El sistema muestra por pantalla el diagrama de gantt de la planificación que el usuario ha creado si la simulación no ha comenzado o de el estado de simulación que se encuentre la planificación en ese momento.</li> </ol>
<b>EXCEPCIONES</b>	NINGUNA

*Tabla 4.13 Detalle caso de uso A4.*

<b>NOMBRE</b>	A5.Avanzar días en la simulación.
<b>PRECONDICIÓN</b>	-El usuario está logueado como alumno en el sistema. -El alumno tiene un proyecto planificado o en estado de simulación.
<b>POSTCONDICIÓN</b>	- La simulación de la planificación se encuentra en nuevo estado. Puede que alguna tarea finalice o incluso el proyecto finalice.
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento del avance de días en una simulación de una planificación.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Simular Planificación en su ventana de bienvenida.</li> <li>2. El sistema muestra por pantalla la vista principal de la simulación.</li> <li>3. El usuario Selecciona Avanzar días.</li> <li>4. El sistema realiza la simulación mediante el algoritmo que se detalla en el <b>Anexo I de este documento</b> y : <ul style="list-style-type: none"> <li>• -determina la finalización de una tarea notificándose al usuario dicho evento y el caso de uso finaliza.</li> <li>• -notifica que la simulación ha finalizado si ya no quedan tareas por finalizar y el caso de uso queda finalizado.</li> </ul> </li> </ol>

<b>EXCEPCIONES</b>	<ul style="list-style-type: none"> <li>• 3.1. El usuario selecciona volver.</li> <li>• 3.1.A El caso de uso queda sin efecto.</li> <li>• 3.2. El usuario selecciona Borrar datos de Simulación</li> <li>• 3.2.A El sistema realiza el caso de uso <u>Borrar datos de Simulación</u> y este caso de uso vuelve a el paso 2.</li> <li>• 4.1 El sistema determina que no hay suficientes recursos para que una tarea pueda comenzar la simulación y lo notifica.</li> <li>• 4.1.A El usuario deberá hacer clic en Replanificar y realizar el caso de uso Modificar planificación.</li> <li>• 4.1.B El caso de uso vuelve al paso 2.</li> <li>• 4.2 El sistema notifica una sobreasignación de recursos.</li> <li>• 4.2.A El usuario deberá hacer clic en Replanificar y realizar el caso de uso Modificar planificación.</li> <li>• 4.1.B El caso de uso vuelve al paso 2.</li> <li>• 4.3 Un riesgo sucede y un Recurso deja de estar disponible.</li> <li>• 4.3.A El usuario deberá determinar si es necesaria una replanificación o no.</li> </ul>
--------------------	---

*Tabla 4.14 Detalle caso de uso A5.*

<b>NOMBRE</b>	A6. Modificar planificación.
<b>PRECONDICIÓN</b>	-El usuario está logueado como alumno en el sistema. -El usuario tiene un proyecto en estado de simulación. -El usuario se encuentra en la vista principal de simulación.
<b>POSTCONDICIÓN</b>	-Los recursos asignados a las tareas de la simulación del alumno han cambiado.
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de una modificación en la asignación de recursos a las diferentes tareas de la planificación de un alumno.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Replanificar en su vista principal de la simulación.</li> <li>2. El sistema muestra una pantalla en la que el usuario puede añadir nuevos recursos e incluso modificar la asignación de ellos con respecto a las tareas..</li> <li>3. El usuario modifica las asignaciones a su gusto y selecciona modificar.</li> <li>4. El sistema pide confirmación de creación.</li> <li>5. El usuario la acepta.</li> <li>6. El sistema almacena la nueva planificación del proyecto y el caso de uso finaliza.</li> </ol>
<b>EXCEPCIONES</b>	<ul style="list-style-type: none"> <li>• 3. El usuario selecciona volver.</li> <li>• 3A. El caso de uso vuelve al paso 1.</li> <li>• 5. El usuario declina la confirmación.</li> <li>• 5A. El caso de uso vuelve al paso 2.</li> </ul>

*Tabla 4.15 Detalle caso de uso A6.*

<b>NOMBRE</b>	A7. Borrar Datos de Simulación.
<b>PRECONDICIÓN</b>	-El usuario está logueado como alumno en el sistema. -El usuario tiene un proyecto en estado de simulación. -El usuario se encuentra en la vista principal de simulación.
<b>POSTCONDICIÓN</b>	-Los datos de simulación de la planificación del alumno se eliminan por completo del sistema. Quedando disponibles los datos de planificación.
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento del borrado completo de los datos de la simulación de una planificación.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Borrar Datos de simulación en su vista principal de la simulación.</li> <li>2. El sistema pide confirmación de la operación.</li> <li>3. El usuario la acepta.</li> <li>4. El sistema elimina todos los datos de la simulación de la planificación del proyecto y el caso de uso finaliza.</li> </ol>
<b>EXCEPCIONES</b>	NINGUNA

*Tabla 4.16 Detalle caso de uso A7.*

<b>NOMBRE</b>	A8. Consultar estado de simulación.
<b>PRECONDICIÓN</b>	-El usuario está logueado como alumno en el sistema. -El alumno tiene un proyecto planificado o en estado de simulación.
<b>POSTCONDICIÓN</b>	NINGUNA
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de la consulta del estado de una simulación.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Simular Planificación en su ventana de bienvenida.</li> <li>2. El sistema muestra por pantalla la vista principal de la simulación, en la que el usuario puede observar los datos principales de su simulación. El caso de uso finaliza.</li> </ol>
<b>EXCEPCIONES</b>	NINGUNA

*Tabla 4.17 Detalle caso de uso A8.*

<b>NOMBRE</b>	A9. Ver resultados de una simulación.
<b>PRECONDICIÓN</b>	-El usuario está logueado como alumno en el sistema. -El usuario alumno tiene un proyecto en estado finalizado.
<b>POSTCONDICIÓN</b>	NINGUNA
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de la consulta de los resultados de una simulación.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Ver resultados en su ventana de bienvenida.</li> <li>2. El sistema muestra por pantalla los diferentes eventos que el la simulación de la planificación del proyecto ha ido generando hasta finalizar y el caso de uso finaliza</li> </ol>
<b>EXCEPCIONES</b>	NINGUNA

*Tabla 4.18 Detalle caso de uso A9.*

<b>NOMBRE</b>	A10.Descarga de resultados de una simulación
<b>PRECONDICIÓN</b>	- El usuario está logueado como alumno en el sistema. - El usuario alumno tiene un proyecto en estado finalizado. - El usuario ha realizado el caso de uso Ver Resultados de una simulación.
<b>POSTCONDICIÓN</b>	- El usuario recibirá por descarga directa un archivo CSV con los resultados del proyecto contenidos en la tabla de esta vista.
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de la descarga de los resultados de una simulación.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción descargar tabla en formato CSV en su ventana de resultados.</li> <li>2. El sistema lanza una descarga de los contenidos de la tabla hacia el host del usuario y el caso de uso finaliza.</li> </ol>
<b>EXCEPCIONES</b>	NINGUNA

*Tabla 4.19 Detalle caso de uso A10.*



<b>NOMBRE</b>	A11. Entrega de resultados de una simulación
<b>PRECONDICIÓN</b>	- El usuario está logueado como alumno en el sistema. -El usuario alumno tiene un proyecto en estado finalizado.
<b>POSTCONDICIÓN</b>	- El usuario profesor recibe los resultados como entregados en su panel de usuario. - El alumno puede crear otro proyecto y comenzar otra simulación.
<b>DESCRIPCIÓN</b>	En este caso de uso se detalla el funcionamiento de la entrega de los resultados de una simulación.
<b>SECUENCIA NORMAL</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción Enviar Resultados en su ventana de resultados.</li> <li>2. El sistema guarda los resultados como definitivos y los hace accesibles al profesor en el apartado de finalizados y entregados.</li> </ol>
<b>EXCEPCIONES</b>	NINGUNA

*Tabla 4.20 Detalle caso de uso A11.*

## 4.3 Modelo de dominio

A continuación se muestra el diagrama de clases que he construido para cubrir los objetivos que muestran los requisitos y diferentes casos de uso de la herramienta que se ha desarrollado en este TFG.

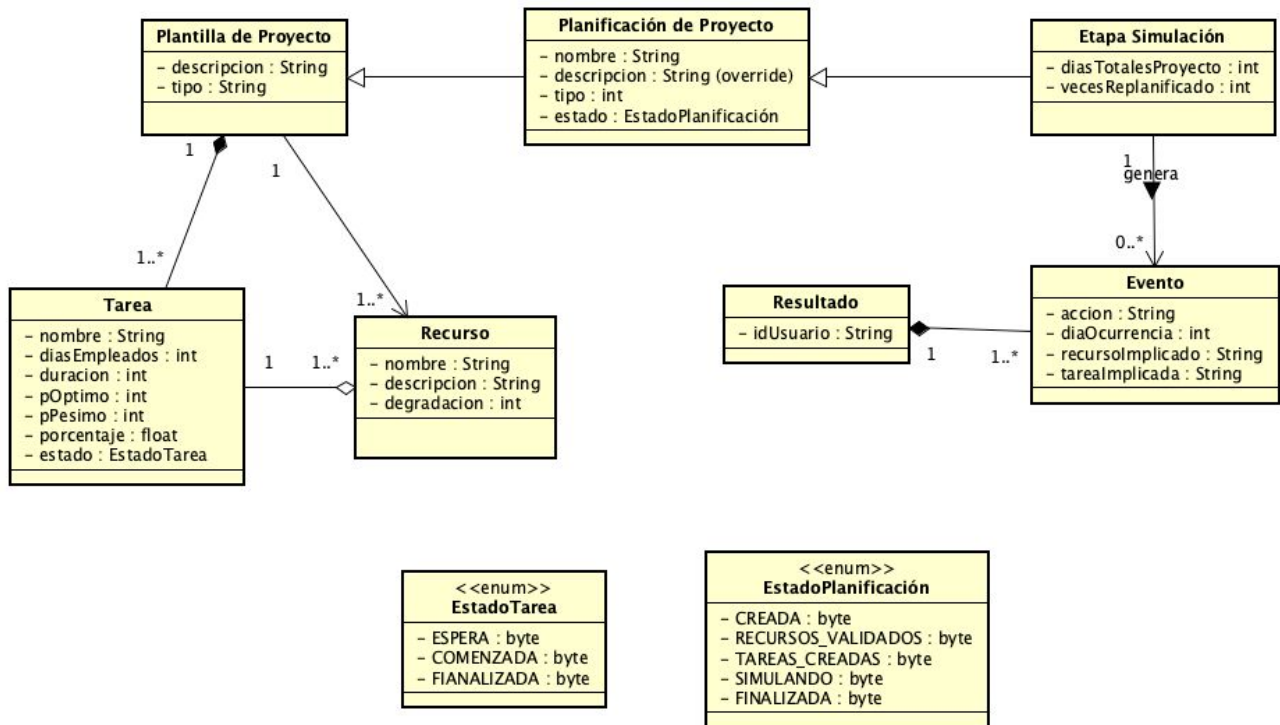


Figura 4.1. Modelo de Dominio del sistema.

## 4.4 Análisis de los casos de uso más relevantes.

En este subcapítulo se adjuntan los diagramas de secuencia para los casos de uso modificar planificación y avanzar días en la simulación, ya que se consideran los más relevantes de la aplicación.

### 4.4.1 Modificar planificación.

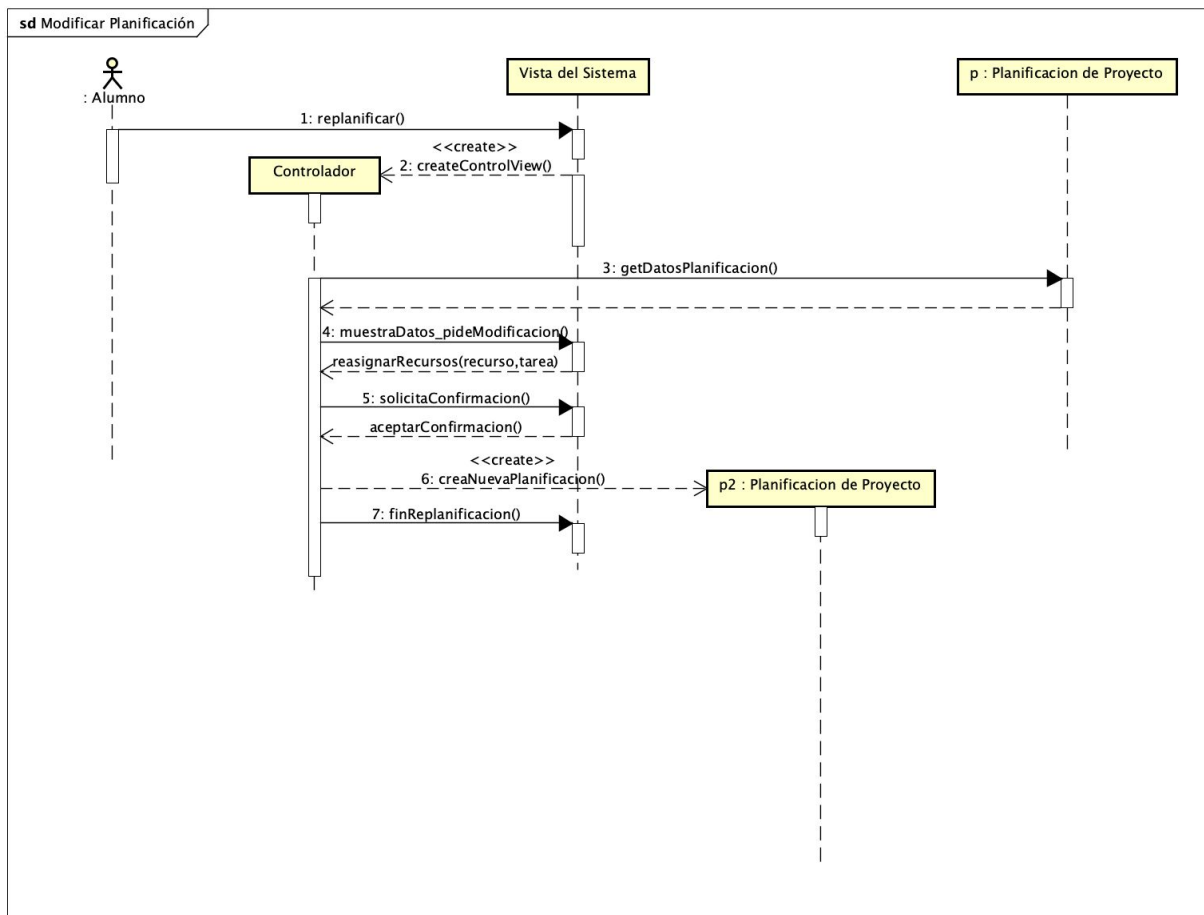


Figura 4.2. Diagrama de secuencia de análisis Caso de Uso “modificar planificación”.

#### 4.4.2 Avanzar días en la simulación.

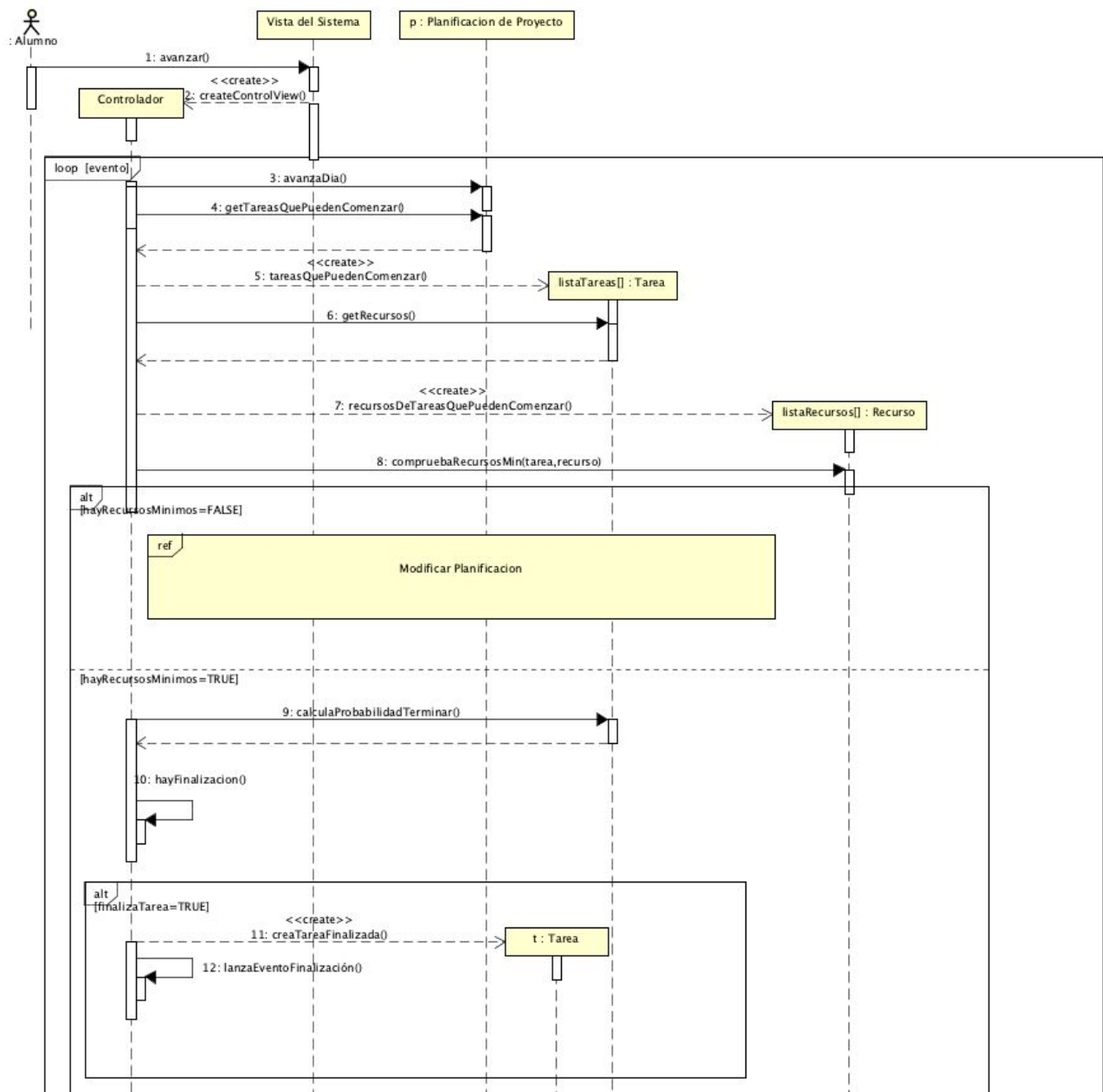


Figura 4.3. Diagrama de secuencia de análisis Caso de Uso “simular planificación”.

# Capítulo 5. Diseño de bajo nivel, tecnologías.

## 5.1 Contexto de las tecnologías web.

Para llevar a cabo el desarrollo de esta herramienta se ha tomado como mejor alternativa su desarrollo como aplicación web.

Una aplicación web, es una herramienta usable accediendo a un servidor web alojado en algún dominio mediante un navegador. Son prácticas debido al uso del navegador como cliente, que consume muchos menos recursos que si el propio cliente es el Sistema Operativo.

Aunque hay varios modelos de arquitectura, una aplicación web suele estar estructurada en tres capas:

- Capa del navegador: es la primera capa, encargada de interpretar el código.
- Capa del servidor: la segunda capa la forma el servidor y todo el código que este envía al navegador como respuesta de las peticiones.
- Capa de datos: la tercera y última capa es una base de datos, a la que únicamente tiene acceso el servidor.

## 5.2 Angular 8.

Angular es un framework de código libre implementado por google para desarrollar aplicaciones web SPA, es decir para desarrollar aplicaciones de una sola página, en la cual la navegación entre las distintas secciones de la misma o cualquier operación como acceso a la base de datos, cálculos, etc... se realizan dinámicamente casi de modo instantáneo, llamando al servidor de manera asíncrona y sin necesidad de recargar la misma.

Tiene varias características que detallo a continuación, que hacen de él uno de los mejores frameworks, en mi opinión, para el desarrollo de este tipo de aplicaciones:

-MVC: Angular proporciona un modelo-vista-controlador muy bien definido, ya que separa completamente el frontend del backend. Su funcionamiento básico es cambiar la vista a través del controlador, en este caso el servidor.

-Modular y escalable: este framework permite crear nuevas etiquetas html, por medio de componentes, que pueden usarse como cualquier otra ya predefinida, además cómo utiliza el estándar web, es perfectamente escalable.

-Programación en Typescript, algo ventajoso, ya que es apoyado por grandes empresas como google o microsoft, esto nos da un gran soporte y durabilidad. Por otro lado, typescript es un lenguaje orientado a objetos, lo que nos permite de nuevo hacer un código modular que a su vez permite la escalabilidad.

-Programación reactiva: angular permite que en el desarrollo la vista se actualice automáticamente con cada cambio que se realice en el código, tanto en el back-end como en el front-end. [Angular]



*Figura 5.1. Imágen angular.*

### 5.3 Angular CLI.

Angular CLI es básicamente una herramienta que funciona a través de comandos usada en aplicaciones implementadas a través del framework angular, que nos permite realizar acciones básicas como crear proyectos, crear componentes, crear servicios, arrancar el servidor local de nuestro proyecto, compilarlo, etc. [Angular CLI]



*Figura 5.2. Imágen angularCLI.*

### 5.4 Node Js.

Node Js está ideado como un entorno de ejecución de JavaScript orientado a eventos asíncronos, Node.js está diseñado para crear aplicaciones de red escalables. Su modo de funcionamiento es estar activo únicamente cuando se recibe alguna conexión, si esto no se produce los procesos de Node JS permanecen dormidos. Como principal característica destaca la capacidad de atender muchas conexiones simultáneas, sin problemas de concurrencia. [Node Js]



*Figura 5.3. Imágen NodeJS.*

## 5.5 Npm.

En su esencia podemos definir a Npm (Node Packaged Manager) como un gestor de paquetes y/o módulos codificado en JavaScript, este gestor suele utilizarse junto con Node.js, es más, node.js adquiere potencia gracias a el.

En cuanto a su composición es básicamente un repositorio online de paquetes de los que se hace uso mediante línea de comandos.

Su utilidad junto a angular o node js, es descargar mediante comandos las dependencias actualizadas que la app en desarrollo requiera. [Npm]



*Figura 5.4. Imágen npm.*

## 5.6 Servicios Firebase de Google.

Firebase es una plataforma ubicada en la nube y creada por google utilizada para desarrollar tanto apps web como apps móviles. Dicha plataforma proporciona a los usuarios cada vez más servicios entre ellos, almacenamiento en la nube, hosting, autenticación, base de datos en tiempo real, etc. [Firebase]

Firebase posee una serie de ventajas que hace de ella un uso muy factible por parte de los desarrolladores, a continuación se detallan las más importantes:

- Usa infraestructura de Google, que como se puede imaginar es muy potente, por ello hace que los proyectos sean muy escalables.

- No es necesario el uso de un servidor o incluso su creación para desarrollar un proyecto, ya que las herramientas de firebase se incluyen en su sdk.

- Se puede integrar fácilmente con multitud de lenguajes incluso con multitud de frameworks modernos de desarrollo web, como angular, react, etc, gracias a sus herramientas multiplataforma.

- Aporta a los usuarios un gran manual de uso de cada una de sus herramientas para cada uno de los lenguajes de programación que admite.

- Es muy sencillo sincronizar datos sin tener que administrar conexiones o escribir código de sincronización o consulta complejo.



*Figura 5.5. Imágen Google Firebase.*

Para el desarrollo de la aplicación que se trata en este TFG se han utilizado tres de las herramientas proporcionadas que detallamos en los subapartados siguientes.



### 5.6.1 Firebase Cloud Database.

Cloud Firestore es una base de datos NoSQL flexible y escalable para la programación en servidores, dispositivos móviles y aplicaciones Web desde Firebase y Google Cloud Platform.

Cloud Firestore mantiene los datos sincronizados en tiempo real entre cualquier app cliente y proporciona algún servicio sin conexión, al igual que la mayoría de las herramientas de google Firebase, permite una sencilla integración con otros productos de la citada plataforma y de google Cloud Platform.

### 5.6.2 Firebase Cloud Functions.

Cloud Functions para Firebase permite ejecutar código backend creado y almacenado en la nube en respuesta a solicitudes HTTP creadas por eventos activados en el código local. El código de estas funciones se almacena en la nube de Google y se ejecuta también en ella en un entorno muy seguro al que sólo se puede acceder mediante unas credenciales determinadas de administración del proyecto en el que está albergada la app.

En el caso de este proyecto, google firebase cloud functions se ha utilizado para realizar operaciones seguras como por ejemplo eliminar usuarios de la BBDD, eliminar planificaciones de la base datos, siempre desde el perfil del profesor, que en este caso es el administrador del sistema.

### 5.6.3 Firebase Auth.

Firebase auth es un servicio de autenticación de usuarios que utiliza únicamente el código del cliente para ello, sin ser necesario realizar implementaciones del lado del servidor, esto aporta al desarrollador mucha facilidad a la hora de implementar su proyecto, ya que no se tiene que preocupar de desarrollar ningún método de autenticación.

Para autenticar a los usuarios Firebase auth proporciona varios métodos entre los cuales: se encuentran, el uso de proveedores de inicio de sesión como Facebook, Twitter, Google..., un sistema de email y contraseña que se almacenan en Firebase, identificación mediante sms y numero de telefono, etc.

Firebase Auth aporta además del servicio de autenticación, varias funcionalidades extra como la recuperación de cuentas y su verificación, cuotas de registro de usuarios, etc, todo ello gestionado en la propia plataforma de Google.

## 5.7 Bootstrap 4.4.

Bootstrap es una biblioteca multiplataforma de código abierto utilizado para el diseño de sitios y aplicaciones web, la cual contiene plantillas de diseño para crear tablas, menús, barras de navegación, formularios, botones de navegación y otros elementos basados html y css con estilos propios.

Cómo detalle de lo potente y funcional que es Bootstrap podemos decir que se le admira por ser el segundo proyecto más destacado en GitHub y por ser usado por organizaciones reconocidas mundialmente como la NASA. [*Bootstrap*]



*Figura 5.6. Imágen Bootstrap.*

## 5.8 Git.

Git es una herramienta de control de versiones gratuito y de código abierto diseñado para manejar y controlar cualquier proyecto de desarrollo de software con rapidez y eficiencia. Git fue creado por Linus Torvalds, creador de Linux. Su uso es muy sencillo y fácil de aprender y a su vez aporta gran seguridad.

Varias empresas han desarrollado su propia herramienta a través del código de Git, algunos ejemplos son GitHub, GitLab, Tortoise Git, etc. [*Git*]



*Figura 5.7. Imágen git.*

# Capítulo 6. Diseño de la arquitectura.

## 6.1 Contexto

En este capítulo se analiza el diseño arquitectónico de la herramienta, es decir se estudia el diseño a alto nivel de la plataforma creada en este TFG.

Como introducción, la arquitectura software analiza el sistema a implementar detallando el mismo, como un conjunto de patrones ya conocidos a través de los cuales se puede acceder e interactuar con el código de la herramienta, la arquitectura software también relaciona los citados componentes software y explica cómo se comunican entre ellos.

La arquitectura software se selecciona en base a diferentes criterios y necesidades del proyecto en desarrollo, en este caso se ha determinado utilizar la arquitectura que angular 8 proporciona, después de estudiar los requisitos y restricciones expuestas en el capítulo 2 del presente documento.

## 6.2 Arquitectura facilitada por Angular

Como ya se ha comentado en alguna ocasión a lo largo del documento, angular es un framework basado en TypeScript para escribir aplicaciones web, mobile y de escritorio. Angular es un framework apoyado en librerías, la mayoría de ellas son parte de su núcleo y son necesarias para el funcionamiento de cualquier aplicación, luego hay otras que puede instalar cada desarrollador dependiendo de la funcionalidad que este desee implementar.

A modo de introducción, una aplicación angular arquitectónicamente hablando está formada por: módulos, servicios y componentes. En los siguientes subcapítulos detallo cada uno de estos elementos y redacto sus características más importantes.

### 6.2.1 Módulos

Un módulo en Angular forma un grupo de compilación para una serie de componentes, es decir, un módulo relaciona sus componentes para formar unidades o conjuntos funcionales en una aplicación.

Todas las aplicaciones angular tienen como mínimo un módulo, este módulo es llamado App Module o módulo root. En este módulo se aloja el proceso de arranque de cualquier aplicación angular.

En mi caso, en la herramienta creada, solo se dispone de dicho módulo, el cuál importa las funcionalidades de todos los componentes y librerías necesarias y además las exporta para que dichas funcionalidades sean accesibles desde cualquiera de ellos.

### 6.2.2 Servicios

Un servicio es uno de los elementos clave en las aplicaciones angular, en ellos se escribe toda la lógica que no está asociada directamente a una vista y desea ser accesible por cada uno de los componentes de la aplicación.

Los servicios al igual que los componentes pueden ser exportados como clases TypeScript, además gracias a portar el decorador `@Injectable` permiten ser inyectados en uno o varios componentes como una propia dependencia, de esta forma cualquiera de los componentes que tenga inyectado este servicio puede acceder a las funcionalidades que este mismo implementa.

### 6.2.3 Componentes y Templates.

Un componente es el elemento base de Angular, cada componente contiene una clase lógica en TypeScript y un template HTML, al igual que ocurre con los módulos, todas las aplicaciones en Angular contienen al menos un componente llamado AppComponent o root component que se encarga de jerarquizar el resto de componentes de la aplicación.

Hablar de componentes Angular implica hablar de Templates, de directivas y de data binding, por ello se comentará a continuación cada uno de estos elementos.

Un template en angular es el conjunto de un archivo HTML junto con su diseño (CSS) en el que se añaden directivas angular que enlazan la lógica de la aplicación junto a sus datos con la vista de la misma.

En angular existen 2 tipos de data binding, property binding y event binding, el primero de ellos permite agregar un valor de nuestro modelo de dominio a la propia vista HTML de la aplicación, los segundos, son eventos provocados por el usuario al modificar o añadir algún elemento de un input de la vista que actualizan o crean valores en el data del modelo de dominio de la aplicación.

## 6.2.4 Resumen.

En la siguiente imagen extraída de la información que angular proporciona en su propia web [[angular](#)] se puede ver el funcionamiento básico del framework, en el cual apreciamos todos los elementos detallados anteriormente que juntos forman una aplicación del mismo, es decir vemos como los componentes acceden a Servicios y comparten datos a través de ellos, esos datos se pueden mostrar en los templates HTML, que a su vez envían eventos a los componentes.

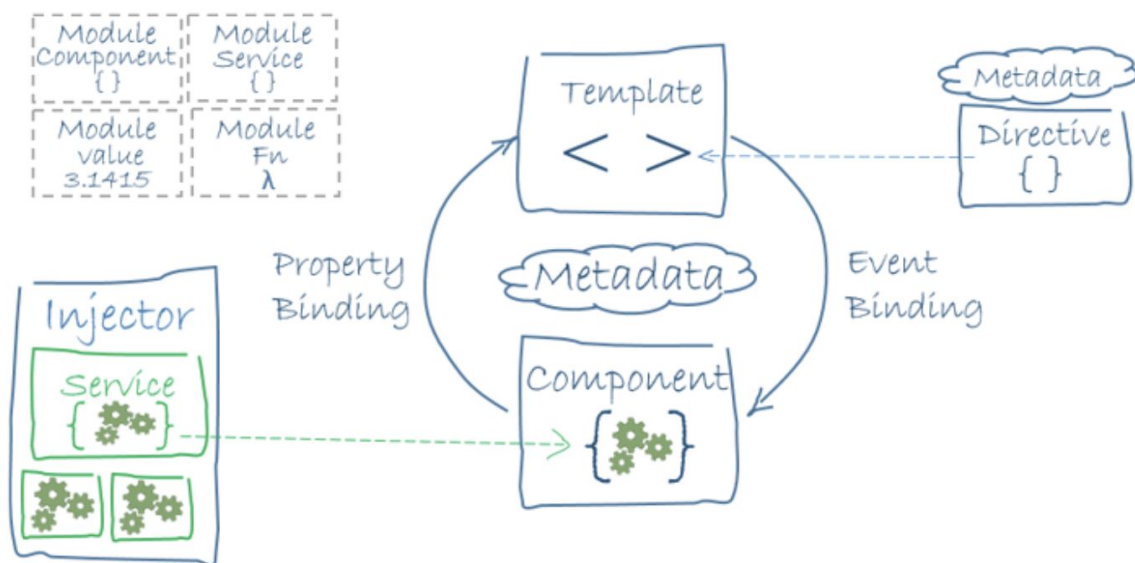


Figura 6.1. Diagrama funcionamiento Angular.

## 6.3 Arquitectura de la aplicación desarrollada.

En este subcapítulo se muestra la arquitectura interna de la aplicación desarrollada que incluye un diagrama de despliegue junto con uno de relación entre componentes y servicios, así como un detalle de cada componente y cada servicio.

### 6.3.1 Diagrama de despliegue y diagrama acceso a servicios.

#### Diagrama de despliegue

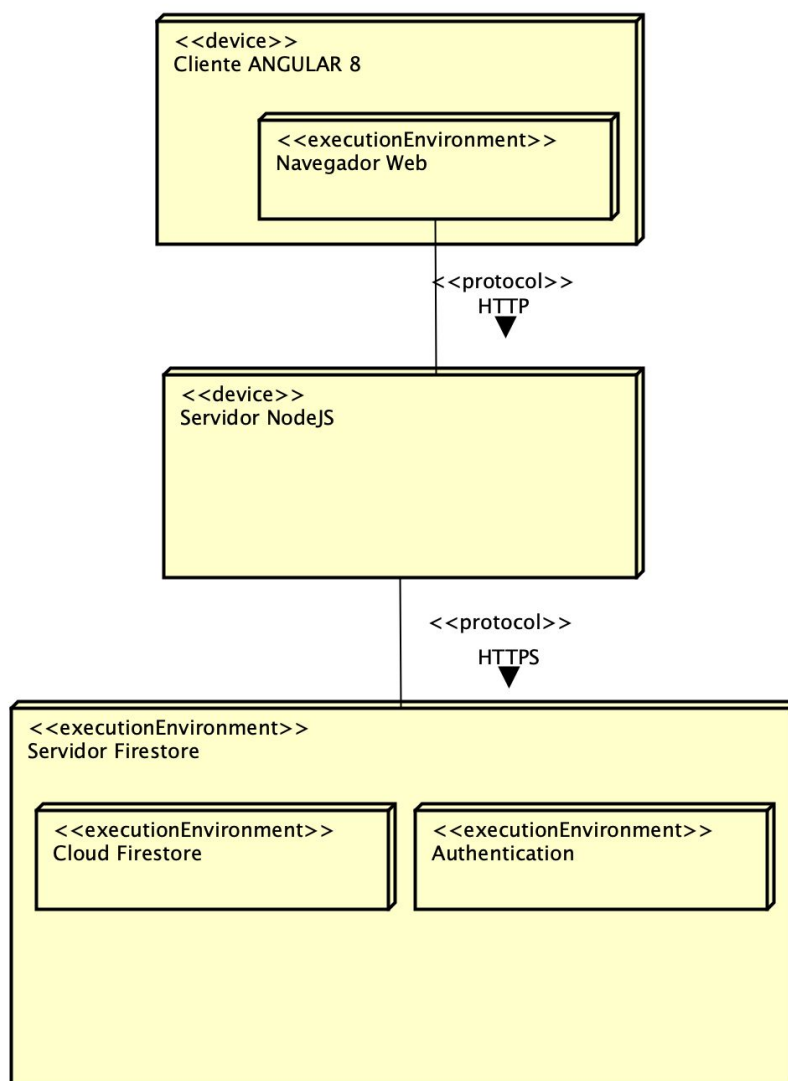


Figura 6.2. Diagrama de despliegue app desarrollada.

## Acceso a servicios

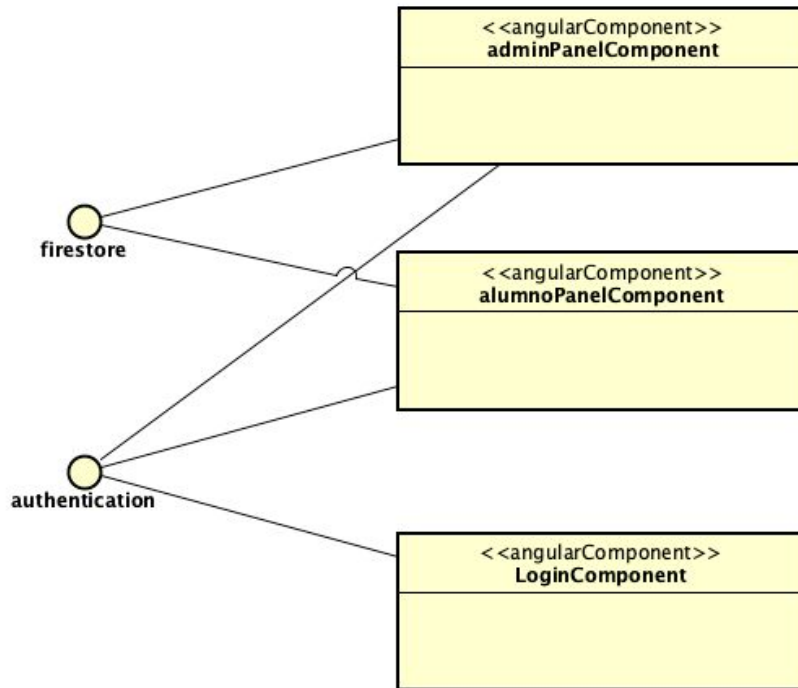


Figura 6.3. Diagrama de acceso a servicios.

## 6.3.2 Detalle de cada componente.

### 6.3.2.1 LoginComponent.

#### **Visión general.**

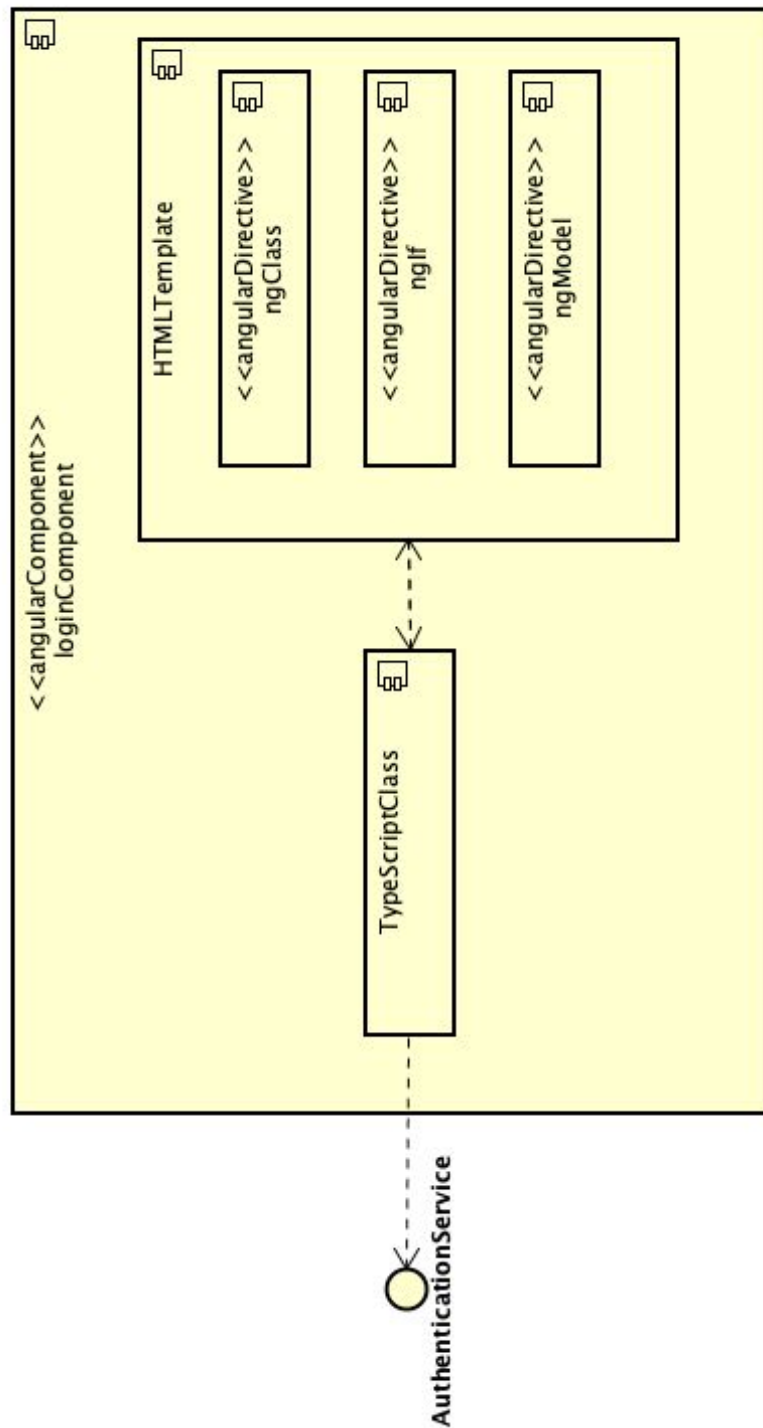


Figura 6.4. Diagrama de arquitectura LoginComponent.



**Detalle de atributos y funciones.**

<b>&lt;&lt;angularComponent&gt;&gt; LoginComponent</b>
<ul style="list-style-type: none"><li>- loginUserForm : FormGroup</li><li>- usuarios : Array</li><li>- password : String</li><li>- email : String</li><li>- loginError : boolean</li></ul>
<ul style="list-style-type: none"><li>+ login() : void</li><li>+ tipoUsuario() : String</li></ul>

*Figura 6.5. Detalle funciones y atributos LoginComponent.*

### 6.3.2.2 AlumnoPanelComponent.

#### Visión general.

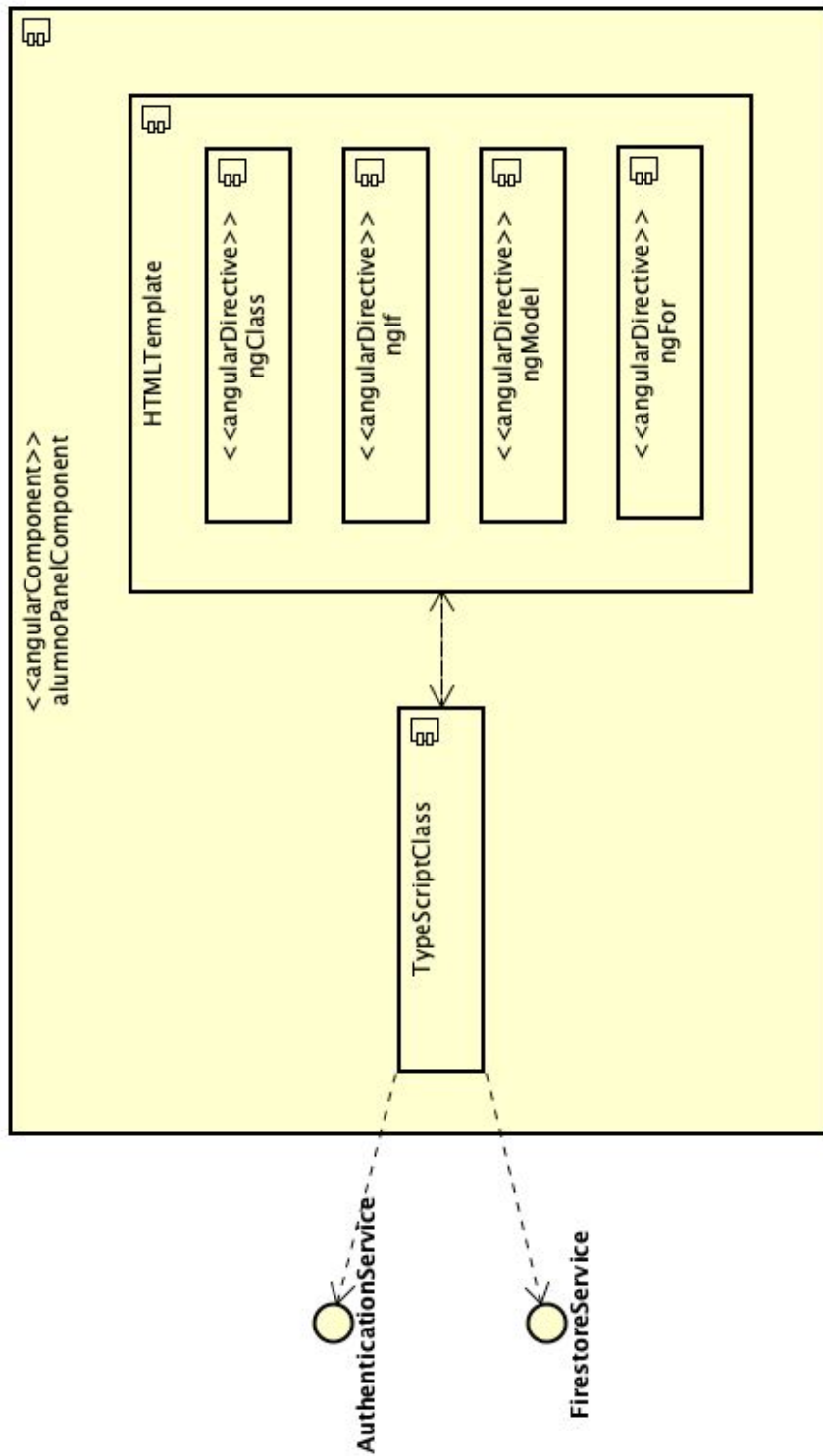


Figura 6.6. Diagrama de arquitectura alumnoPanelComponent.

## Detalle de atributos y funciones.

<code>&lt;&lt;angularComponent&gt;&gt;</code> <b>AlumnoPanelComponent</b>
<ul style="list-style-type: none"><li>- ESTADOS : enum</li><li>- ESTADOS_TAREAS : enum</li><li>- proyecto : object</li><li>- cantidadVersiones : int</li><li>- numRecursosMaximo : int</li><li>- tareasGant1 : Array</li><li>- estado : ESTADOS</li><li>- proyectoSimulacion : object</li><li>- proyectoComprobacionFinal : object</li><li>- proyectoFin : object</li><li>- plantillas : Array</li><li>- plantillaSeleccionada : object</li><li>- mostrarTablaRecursos : boolean</li><li>- hayGant : boolean</li><li>- proyectoTareas : object</li><li>- simulacionPantalla : boolean</li><li>- estadosSimulacion : Array</li><li>- tareasEnProceso : Array</li><li>- tareasFinalizadas : Array</li><li>- tareasQuePuedenComenzar : Array</li><li>- tareasEnEsperaPorDependencia : Array</li><li>- tareasQuePuedenAcabar : Array</li><li>- tareasNoFinalizadas : Array</li><li>- hayEvento : boolean</li><li>- probabilidades : Array</li><li>- eventos : Array</li></ul>
<ul style="list-style-type: none"><li>+ logOut() : void</li><li>+ compruebaEstadoProyecto() : void</li><li>+ selectPlantilla() : void</li><li>+ nuevoProyecto() : void</li><li>+ eliminarElemento(index : int, option : int) : void</li><li>+ addTarea() : void</li><li>+ addRecurso() : void</li><li>+ guardarProyecto() : void</li><li>+ guardarProyectoInit() : void</li><li>+ guardarProyectoTareas() : void</li><li>+ getCantidadRecurso(recurso : object, tarea : object, index : int) : void</li><li>+ asignarRecursosTareas() : void</li><li>+ dibujaGantt() : void</li><li>+ tiempoPesimoOptimo(tiempo : int, porcentaje : float, opcion : int) : float</li><li>+ simular() : void</li><li>+ creaProyectoSimulacion() : void</li><li>+ comprobacionMinRecursos() : void</li><li>+ avanzarSimulacion() : void</li><li>+ getProyectoEsFinalizado() : void</li><li>+ borrarProyecto() : void</li><li>+ borrarSimuacion() : void</li><li>+ almacenarResultados() : void</li><li>+ creaTareasProceso() : void</li><li>+ getRandom() : float</li><li>+ modificarPlanificacion() : void</li><li>+ dowloadCSV(csv : Object, filename : String) : void</li><li>+ exportTableToCSV() : void</li><li>+ tablaFin() : void</li></ul>

Figura 6.7. Detalle funciones y atributos alumnoPanelComponent.

### 6.3.2.3 AdminPanelComponent.

#### Visión general.

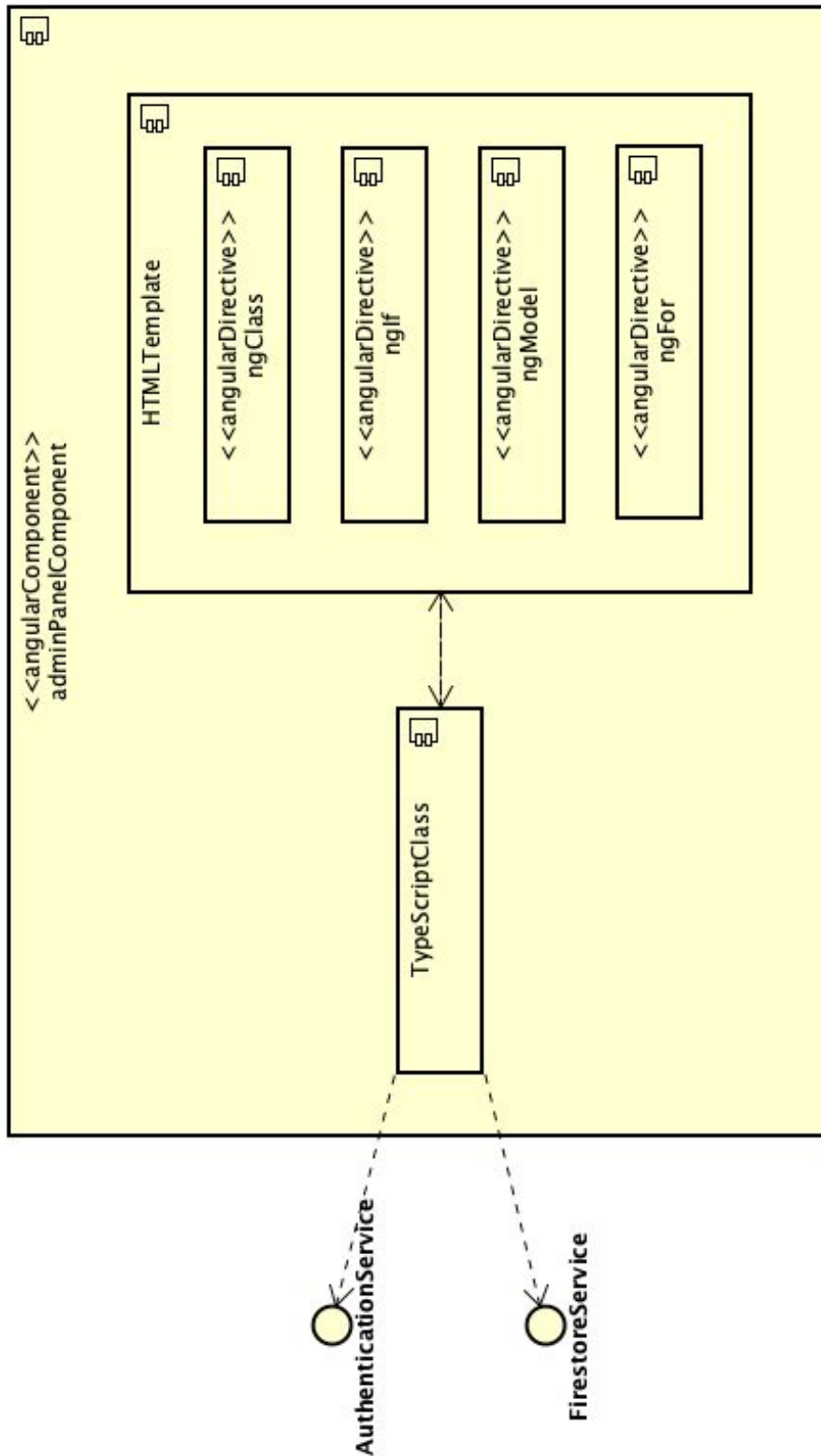


Figura 6.8. Diagrama de arquitectura `adminPanelComponent`.

## Detalle de atributos y funciones.

<<angularComponent>> AdminPanelComponent
<ul style="list-style-type: none"> <li>- creacionPlantilla : boolean</li> <li>- previaCreacionPlantilla : boolean</li> <li>- verDetallePlantilla : boolean</li> <li>- plantillas : Array</li> <li>- mostrarTablaRecursos : boolean</li> <li>- introduccion : boolean</li> <li>- tercerpaso : boolean</li> <li>- alumnosConProyectos : Array</li> <li>- alumnoMoment : Object</li> <li>- recursosProyectoActual : Array</li> <li>- roles : Array</li> <li>- email : String</li> <li>- password : String</li> <li>- mensajeRegistro : String</li> <li>- alumnosEnSimulacion : Array</li> <li>- proyectosFinalizados : Array</li> <li>- eventos : Array</li> </ul>
<ul style="list-style-type: none"> <li>+ nuevoProyecto() : void</li> <li>+ hayUnRecurso() : boolean</li> <li>+ newRecurso() : void</li> <li>+ eliminarElemento(index : int, tipo : String) : void</li> <li>+ guardarProyecto(proyecto : Object) : void</li> <li>+ addTarea() : void</li> <li>+ logout() : void</li> <li>+ getAlumnos() : void</li> <li>+ cuentaTareas(tareas : Array) : void</li> <li>+ previaAsignacionRecursos(alumno : Object, proyecto : Object, recursos : Array, descripcion : String, tareas : Array) : void</li> <li>+ newAlumno() : void</li> <li>+ finRegistro() : void</li> <li>+ actualizarProyecto() : void</li> <li>+ descargaProyectosSimulacion() : void</li> <li>+ descargaProyectosFinalizados() : void</li> <li>+ operation45() : void</li> <li>+ tablaFin() : void</li> <li>+ downloadCSV() : void</li> <li>+ exportTableToCSV() : void</li> <li>+ calculaSimulacionesAcabadas(id : String) : int</li> </ul>

Figura 6.9. Detalle funciones y atributos adminPanelComponent.

### 6.3.3 Detalle de cada servicio.

#### 6.3.3.1 AuthenticationService.

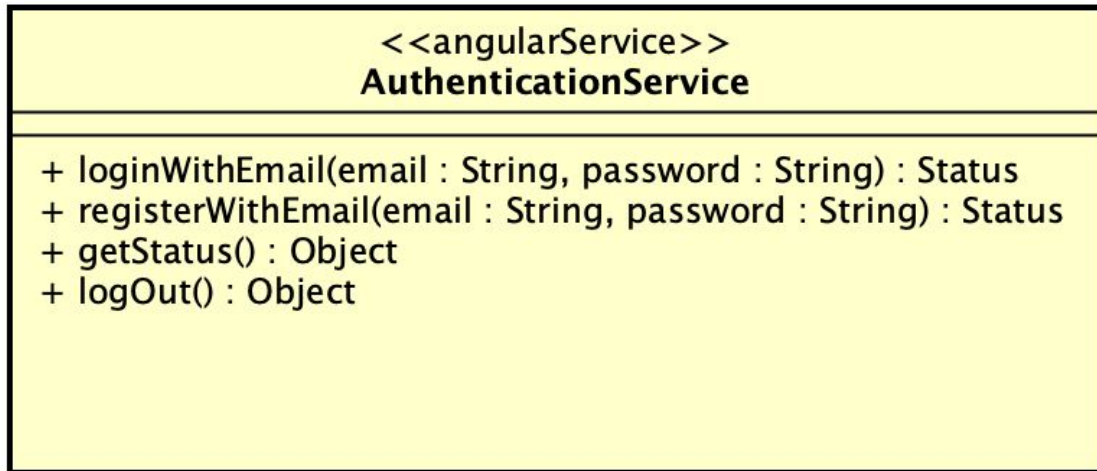


Figura 6.10. Detalle funciones y atributos AuthenticationService.

#### 6.3.3.2 FirestoreService.

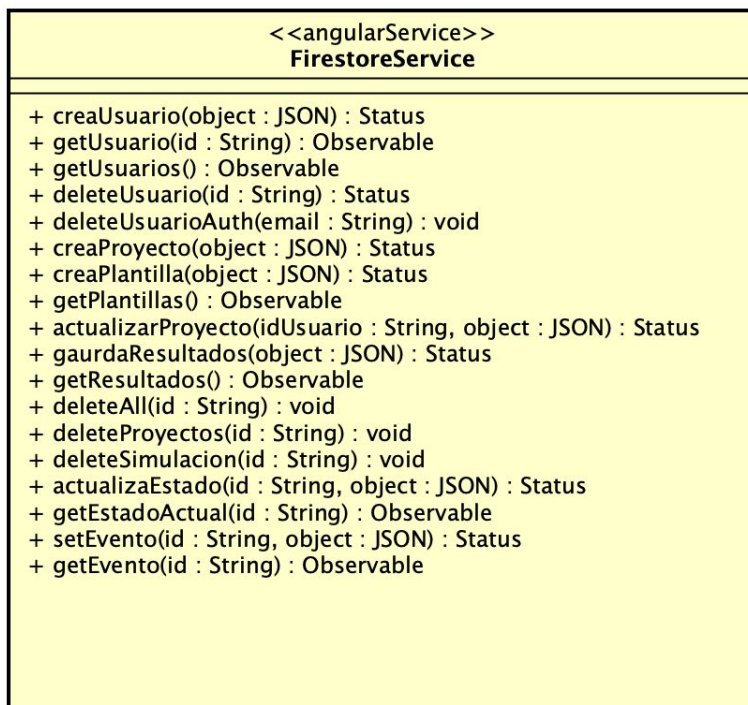


Figura 6.11. Detalle funciones y atributos FirestoreService.

# Capítulo 7. Evaluación y pruebas.

## 7.1 Contexto

Para comprobar el correcto funcionamiento de la aplicación en diferentes aspectos, se han realizado una serie de pruebas de diferentes tipos, pretendiendo así, que abarquen el máximo posible de parámetros de la propia aplicación.

Las pruebas que se han usado son :

1. **Pruebas unitarias:** este tipo de pruebas son las realizadas a cada componente individual de código y tienen como objetivo verificar su correcta funcionalidad, en nuestro caso, cada prueba unitaria comprobará el correcto funcionamiento de las diferentes funciones del código.
2. **Pruebas de integración:** su objetivo es comprobar que la aplicación y el conjunto de sus componentes, mantienen una buena comunicación para cubrir con éxito el conjunto de la funcionalidad requerido de la misma.
3. **Pruebas de sistema:** su objetivo es probar el sistema al completo como un bloque, se centran en los requisitos no funcionales como compatibilidades, usabilidad, etc.

En este apartado únicamente se han incluido las pruebas satisfactorias como ejemplo, es de reconocer que nada ha salido a la primera y se han obtenido errores a lo largo del desarrollo de esta aplicación.

## 7.2 Pruebas unitarias

Como es lógico, en este documento se añaden únicamente los casos de prueba más relevantes, para evitar su extensión.

En este subcapítulo, se indicará la función a probar junto con la especificación de la prueba y el resultado obtenido. Para cada caso de prueba se añade una tabla que almacena estos parámetros citados.

### **Función `loginComponent.login(email,password)` incorrecto**

<b>Especificación del caso de prueba</b>	Se llama a la función con un mail y contraseña incorrectos, o que no existen en el sistema.
<b>Resultado esperado</b>	Se lanza un error.
<b>Resultado obtenido</b>	Se lanza un error.

*Tabla 7.1 Detalle caso de Prueba login 1.*

### **Función `loginComponent.login(email,password)` correcto**

<b>Especificación del caso de prueba</b>	Se llama a la función con un mail y contraseña correctos, es decir, que están asociados a un usuario.
<b>Resultado esperado</b>	Se crea una sesión para el usuario al que corresponde el email y la contraseña.
<b>Resultado obtenido</b>	Se crea una sesión para el usuario al que corresponde el email y la contraseña.

*Tabla 7.2 Detalle caso de Prueba login 2.*

### **Función `AlumnoPanelComponent.getCantidadRecurso(recurso,tareas,index)`**

<b>Especificación del caso de prueba</b>	Se llama a la función con un recurso del cual se dispone de 10 unidades de las cuales 7 están asignadas a diferentes tareas.
<b>Resultado esperado</b>	3
<b>Resultado obtenido</b>	3

*Tabla 7.3 Detalle caso de Prueba `getCantidadRecurso`.*



**Función AlumnoPanelComponent.tiempoPesimoOptimo(tiempo, porcentaje, opt) caso tiempo óptimo**

<b>Especificación del caso de prueba</b>	Se llama a la función con un tiempo de 10, un porcentaje óptimo de 50% y opt=true
<b>Resultado esperado</b>	5
<b>Resultado obtenido</b>	5

*Tabla 7.4 Detalle caso de Prueba tiempoPesimoOptimo 1.*

**Función AlumnoPanelComponent.tiempoPesimoOptimo(tiempo, porcentaje, opt) caso tiempo pésimo**

<b>Especificación del caso de prueba</b>	Se llama a la función con un tiempo de 10, un porcentaje óptimo de 50% y opt=false
<b>Resultado esperado</b>	15
<b>Resultado obtenido</b>	15

*Tabla 7.5 Detalle caso de Prueba tiempoPesimoOptimo 2.*

**Función AlumnoPanelComponent.creaProyectoSimulacion(proyecto)**

<b>Especificación del caso de prueba</b>	Se llama a la función con un objeto proyecto determinado debidamente inicializado y se imprime por consola la variable global proyectoSimulacion.
<b>Resultado esperado</b>	La variable imprimida contiene un objeto y no es undefined.
<b>Resultado obtenido</b>	La variable imprimida contiene un objeto y no es undefined.

*Tabla 7.6 Detalle caso de Prueba ceraProyectoSimulacion.*

### **Función AlumnoPanelComponent.borrarProyecto()**

<b>Especificación del caso de prueba</b>	Se llama a la función teniendo una variable global proyecto debidamente inicializada y almacenada en BBDD y luego se imprime por consola dicha variable.
<b>Resultado esperado</b>	La variable global imprimida es undefined.
<b>Resultado obtenido</b>	La variable global imprimida es undefined.

*Tabla 7.7 Detalle caso de Prueba borrarProyecto.*

### **Función AdminPanelComponent.hayUnRecurso() con proyecto con recursos creados**

<b>Especificación del caso de prueba</b>	Se llama a la función teniendo una variable global proyecto debidamente inicializada y con recursos asignados.
<b>Resultado esperado</b>	true
<b>Resultado obtenido</b>	true

*Tabla 7.8 Detalle caso de Prueba hayUnRecurso 1.*

### **Función AdminPanelComponent.hayUnRecurso() con proyecto sin recursos creados**

<b>Especificación del caso de prueba</b>	Se llama a la función teniendo una variable global proyecto debidamente inicializada y sin recursos asignados.
<b>Resultado esperado</b>	false
<b>Resultado obtenido</b>	false

*Tabla 7.9 Detalle caso de Prueba hayUnRecurso 2.*

### **Función AdminPanelComponent.newRecurso()**

<b>Especificación del caso de prueba</b>	Se llama a la función teniendo una variable global proyecto de tipo objeto creada y se imprime dicha variable por consola.
<b>Resultado esperado</b>	Se imprime un objeto el cual contiene un Array llamado recursos y este a su vez un objeto con los parámetros nombre, descripción y degradación, todos ellos con valor null.
<b>Resultado obtenido</b>	Se imprime un objeto el cual contiene un Array llamado recursos y este a su vez un objeto con los parámetros nombre, descripción y degradación, todos ellos con valor null.

*Tabla 7.10 Detalle caso de Prueba newRecurso.*

### **Función AdminPanelComponent.getAlumnos()**

<b>Especificación del caso de prueba</b>	Se llama a la función y se imprime una variable global llamada alumnos por consola.
<b>Resultado esperado</b>	Si existen alumnos registrados en el sistema dicha variable será un Array relleno de objetos alumno.
<b>Resultado obtenido</b>	Si existen alumnos registrados en el sistema dicha variable será un Array relleno de objetos alumno.

*Tabla 7.11 Detalle caso de Prueba getAlumnos.*

**Función AdminPanelComponent.cuentaTareas(tareas)**

<b>Especificación del caso de prueba</b>	Se llama a la función con un array que contiene 5 tareas.
<b>Resultado esperado</b>	5
<b>Resultado obtenido</b>	5

*Tabla 7.12 Detalle caso de Prueba cuentaTareas.*

**Función**

**AdminPanelComponent.newAlumno(nombre,componentes,email,password)**

<b>Especificación del caso de prueba</b>	Se llama a la función con los datos: Grupo 1,Juan y pedro, <a href="mailto:prueba@prueba.com">prueba@prueba.com</a> , prueba.
<b>Resultado esperado</b>	Se crea un objeto alumno con los parámetros especificados y se almacena dicho objeto en BD.
<b>Resultado obtenido</b>	Se crea un objeto alumno con los parámetros especificados y se almacena dicho objeto en BD.

*Tabla 7.13 Detalle caso de Prueba newAlumno.*

## 7.3 Pruebas de integración

En este subcapítulo, se indicará el caso de uso a probar junto con la especificación de la prueba y el resultado obtenido. Para cada caso de prueba se añade una tabla que almacena estos parámetros citados.

A continuación se muestran una serie de pruebas todas ellas con un resultado positivo, durante el desarrollo del proyecto se ha obtenido una serie de pruebas erróneas, debido a errores de implementación en la parte de lógica de la aplicación, todos ellos se han ido solucionando sin mayor problema.

### Caso de uso iniciar Sesión.

<b>Procedimiento</b>	En la pantalla inicial del sistema, se introduce un email y password.
<b>Resultado esperado</b>	Se inicia la sesión y se muestra el menú del usuario determinado por esa sesión.
<b>Resultado obtenido</b>	Se inicia la sesión y se muestra el menú del usuario determinado por esa sesión.

*Tabla 7.15 Detalle caso de Prueba iniciar Sesión.*

### Caso de uso crear Plantilla con error en los recursos.

<b>Procedimiento</b>	En la pantalla inicial del profesor, se hace clic en ver/crear plantillas y se selecciona crear plantilla. Después sin introducir ningún recurso para la plantilla, se hace clic en siguiente.
<b>Resultado esperado</b>	El sistema detecta que no existen recursos y no deja continuar.
<b>Resultado obtenido</b>	El sistema detecta que no existen recursos y no deja continuar.

*Tabla 7.16 Detalle caso de Prueba de Plantilla con error en recursos.*

### **Caso de uso crear Proyecto.**

<b>Procedimiento</b>	En la pantalla inicial del alumno, se hace clic en crear Proyecto y se rellenan todos los campos de tareas y recursos.
<b>Resultado esperado</b>	El sistema muestra por pantalla que el proyecto está esperando la validación del tutor.
<b>Resultado obtenido</b>	El sistema muestra por pantalla que el proyecto está esperando la validación del tutor.

*Tabla 7.17 Detalle caso de Prueba crear Proyecto.*

### **Caso de uso avanzar simulación.**

<b>Procedimiento</b>	En la pantalla inicial del alumno, se hace clic en pantalla simulación y se selecciona avanzar días.
<b>Resultado esperado</b>	El sistema simulará y notificará la acción necesaria o un suceso determinado.
<b>Resultado obtenido</b>	El sistema simula y notifica la acción necesaria o un suceso determinado.

*Tabla 7.18 Detalle caso de Prueba avanzar simulación.*

### **Caso de uso crear grupo alumnos.**

<b>Procedimiento</b>	En la pantalla inicial del profesor, se hace clic en ver/registrar usuarios, después se selecciona registrar y se rellenan todos los campos.
<b>Resultado esperado</b>	El sistema creará un nuevo grupo de usuarios con los parámetros introducidos.
<b>Resultado obtenido</b>	El sistema ha creado un nuevo grupo de usuarios con los parámetros introducidos.

*Tabla 7.19 Detalle caso de Prueba crear alumno/grupo alumnos.*

## 7.4 Pruebas de sistema

En este subcapítulo se explica básicamente las pruebas que se han realizado para comprobar la compatibilidad requerida con diferentes sistemas o navegadores y otros aspectos como la aceptación de la app entre los futuros usuarios.

### 7.4.1 Pruebas de compatibilidad

Como se cita en los requisitos, esta aplicación web, debe de ser usable y ejecutable en cualquier navegador y dispositivo móvil o fijo existente, además de poder mantener sesiones de un mismo usuario activas en varios dispositivos en un mismo instante.

Para probar el primer aspecto, se accedió a la app desde varias máquinas y varios navegadores, como son un smartphone con android y navegador google chrome, un ordenador portátil macbook pro con ios y navegador safari, y un ordenador con windows y google chrome como navegador. En ninguno de ellos se mostró algún error y la app pudo ejecutarse con éxito.

### 7.4.1 Pruebas de aceptación

Una vez finalizado el desarrollo de la app, se realizaron las pruebas de aceptación. Para proceder con estas pruebas se contactó con varias fuentes, entre ellas el tutor del proyecto César Llamas ( usuario final de la app ) y otros compañeros de clase, a todos ellos se les encargó un uso de la aplicación, desde el inicio hasta la simulación de una planificación, es decir, probar toda la funcionalidad del sistema.

Para extraer conclusiones de estas pruebas se les pidió responder a las siguientes preguntas después de proceder con el uso:

1. Nota a la interfaz 0-5.
2. Nota usabilidad 0-5.
3. Nota funcionalidad 0-5.
4. Comentarios generales

Como conclusiones se han obtenido que la aplicación es muy intuitiva y usable, como aspectos a mejorar se destacó la interfaz y quizás una ampliación futura de funcionalidad.

Se obtiene en general un feedback positivo, por lo que se dá por concluida la primera versión de esta aplicación.

Cabe destacar que en el siguiente curso esta app será usada por alumnos de la asignatura PGP, esto nos traerá consigo nuevos aportes en cuanto a su uso.

# Capítulo 8. Conclusiones y trabajo futuro.

## 8.1 Contexto

Después de unos meses de duro trabajo en esta aplicación, es necesario dejar constancia de aquello que se ha aprendido durante la realización de este trabajo, así como un vistazo de aquello que considero que podría mejorarse en una versión futura de la app. En este último capítulo se realiza precisamente esto, se obtienen unas conclusiones de lo que ha sido el trabajo y se realizan unas propuestas para el futuro en caso de que esta aplicación tuviera una segunda versión.

## 8.2 Conclusiones y aprendizaje

Tras finalizar este proyecto, se realiza un balance con los objetivos propuestos al inicio del mismo, que eran:

- Realizar un proyecto en su totalidad, teniendo un cliente real, en este caso el departamento de informática de la Universidad.
- Elaborar un informe completo de dicho proyecto.
- Realizar una presentación pública de dicho proyecto.
- Facilitar el estudio de las metodologías de planificación y gestión de proyectos.
- Estudio del framework y tecnologías elegidas para realizar la aplicación.
- Permitir realizar una planificación de un proyecto con sus determinadas dependencias entre tareas.
- Permitir asignar recursos a las diferentes tareas definidas en la planificación
- Permitir simular una planificación de un proyecto.
- Permitir replanificar la asignación de recursos a las diferentes tareas.
- Permitir simular riesgos surgidos a través de la simulación.
- Permitir consultar el estado de la simulación en todo momento.
- Obtener un informe de la simulación en formato CSV una vez esta ha finalizado.
- Permitir desplegar la aplicación en un servidor.
- Conseguir una aplicación lo más usable e intuitiva posible.

En mi opinión, se han superado todos con éxito, aunque es posible que se puedan añadir mejoras y nuevos objetivos para el desarrollo de una app más funcional, esto se detallará en el siguiente subcapítulo.



A parte de cumplir con estos objetivos, de estos meses puedo obtener varias conclusiones relacionadas con el proyecto en general y con el modo de trabajo.

He podido realizar un proyecto informático completo, pasando por todas las fases que ello conlleva, es decir, desde el planteamiento y planificación hasta el desarrollo iterativo, con reuniones periódicas en las que se obtenía feedback o retroalimentación del cliente ( en este caso el propio tutor ), durante este camino me he encontrado con grandes obstáculos debidos a la inexperiencia en la realización de proyectos ordenados y al completo, pero se han resuelto sin mayor problema, teniendo constancia y con la ayuda del tutor.

Por mi parte, ha sido necesario reestudiar conceptos perdidos como los tipos de planificaciones de proyectos, las dependencias entre tareas, diagramas de Gantt, es decir, conceptos que tienen que ver con la asignatura de Planificación y Gestión de Proyectos.

Por otro lado, puedo sacar una conclusión positiva del correcto aprendizaje de las tecnologías utilizadas en el desarrollo del proyecto, ya que se han utilizado tecnologías desconocidas por mi hasta ahora, esta acción de aprendizaje puede ser muestra de cómo proceder a enfrentarme a nuevas tecnologías con éxito.

Por último cito una conclusión positiva y en mi opinión la más usable en un futuro, se ha aprendido a trabajar por uno mismo, es decir, a no tener una fecha fijada de entrega como se venía haciendo en el resto de asignaturas de la carrera, sino a depender de uno mismo, esto ha conllevado consigo una mejora en la planificación y en la forma de trabajo que tenía hasta ahora.

### 8.3 Trabajo y propuestas futuras

Durante el desarrollo de esta aplicación, han surgido varias inquietudes y ideas de futuras funcionalidades, pero no han sido llevadas a cabo para ajustarse al modelo de 300h estimadas de duración del TFG, ya que sino podría haberse alargado y el tiempo de desarrollo es limitado, algunas de ellas y las que me parecen más interesantes se detallan a continuación.

1. Actualmente, solo se puede planificar la asignación de recursos a una tarea de una determinada planificación durante la simulación, pero sería muy útil, que además se pudieran modificar las fechas de inicio e incluso las dependencias entre tareas.
2. En esta primera versión de la app únicamente se han tenido en cuenta las dependencias de tipo Fin-Inicio para las tareas de las planificaciones a simular, en un futuro podrían añadirse otro tipo de dependencias a la aplicación.
3. Como mejora visible, propondría quizás mejorar un poco el tablero de juego y simulación, creando una interfaz más colorida y bonita hacia el usuario.

4. Actualmente el sistema solo permite tener una planificación activa para cada grupo de alumnos, pero en un futuro se tendrá en cuenta la posibilidad de albergar más de una planificación activa.

Para realizar la mayoría de estas diferentes funcionalidades, no sería necesario un gran cambio dentro de la app, simplemente al estar dividida en componentes sería añadir nueva lógica que responda a estas necesidades en su componente determinado.

# Referencias Bibliográficas.

[**PMBOK**] - PMI Standards Committee, & Project Management Institute. (1996). A guide to the project management body of knowledge. Project Management Institute.

Última visita. 28 de Junio de 2020.

[**TIMETOAST-WEB**] - Gustavo Paredes. Metodologías en el desarrollo del software.

<https://www.timetoast.com/timelines/metodologias-en-el-desarrollo-de-software>

Última visita 27 de Junio de 2020.

[**MS Project-Wikipedia**] - Microsoft Project. (2019, 21 de diciembre). Wikipedia, La enciclopedia libre.

[https://es.wikipedia.org/w/index.php?title=Microsoft Project&oldid=122186169](https://es.wikipedia.org/w/index.php?title=Microsoft_Project&oldid=122186169)

Última visita 29 de Junio de 2020.

[**Requisito no funcional-Wikipedia**] - Leffingwell, Dean and Don Widrig. Managing Software Requirements: A Use Case Approach (second edition). Addison-Wesley, 2003.

Última visita 29 de Junio de 2020.

[**BOPVA-A2020**] - BOE diputación provincial de valladolid, año 2020.

<https://bop.sede.diputaciondevalladolid.es/boletines/2020/febrero/05/BOPVA-A-2020-00407.pdf> (Página 8)

Última visita 29 de Junio de 2020.

[**Angular**] - <https://angular.io/> Última visita 29 de Junio de 2020.

[**Angular CLI**] - <https://cli.angular.io/> Última visita 29 de Junio de 2020.

[**Node Js**] - OpenJS Foundation - <https://nodejs.org/es/>

Última visita 29 de Junio de 2020.

[**Npm**] - <https://www.npmjs.com/> Última visita 29 de Junio de 2020.

[**Firebase**] - <https://firebase.google.com/> Última visita 29 de Junio de 2020.

[**Bootstrap**] - <https://getbootstrap.com/> Última visita 29 de Junio de 2020.

[**Git**] - <https://git-scm.com/> Última visita 29 de Junio de 2020.

[**Simulación por eventos discretos-Wikipedia**] - Simulación por eventos discretos. (2019, 15 de julio). Wikipedia, La enciclopedia libre.

[https://es.wikipedia.org/w/index.php?title=Simulaci%C3%B3n\\_por\\_eventos\\_discretos&oldid=117433570](https://es.wikipedia.org/w/index.php?title=Simulaci%C3%B3n_por_eventos_discretos&oldid=117433570)

Última visita 28 de Junio de 2020.

# ANEXO I. Algoritmo de simulación

## 1. Contexto.

Para la realización de la parte simulada del sistema se ha utilizado una simulación de eventos discretos generados por determinadas probabilidades basadas en probabilidades PERT, en este anexo, se da un vistazo general a la simulación basada en eventos discretos, a la probabilidad de PERT y se detalla el algoritmo creado para dar solución al problema de esta aplicación de simulación de planificaciones.

## 2. Simulación basada en eventos discretos.

### 2.1 Introducción.

La simulación basada en eventos discretos es una técnica usada en informática que se caracteriza por un control temporal que permite a un sistema avanzar en intervalos que varían, en función de la ocurrencia de determinados eventos. Cabe destacar que entre la ocurrencia de dos eventos del sistema es necesario que no se produzca ningún cambio en el mismo, para asegurar que la simulación pueda saltar directamente al siguiente intervalo [*Simulación por eventos discretos-Wikipedia*].

En resumen, una simulación o sistema basado en eventos discretos se refiere a sistemas que pueden ser representados por una sucesión de eventos.

### 2.2 Componentes de un sistema Basado en eventos discretos.

Un sistema basado en eventos discretos consta de los siguientes elementos:

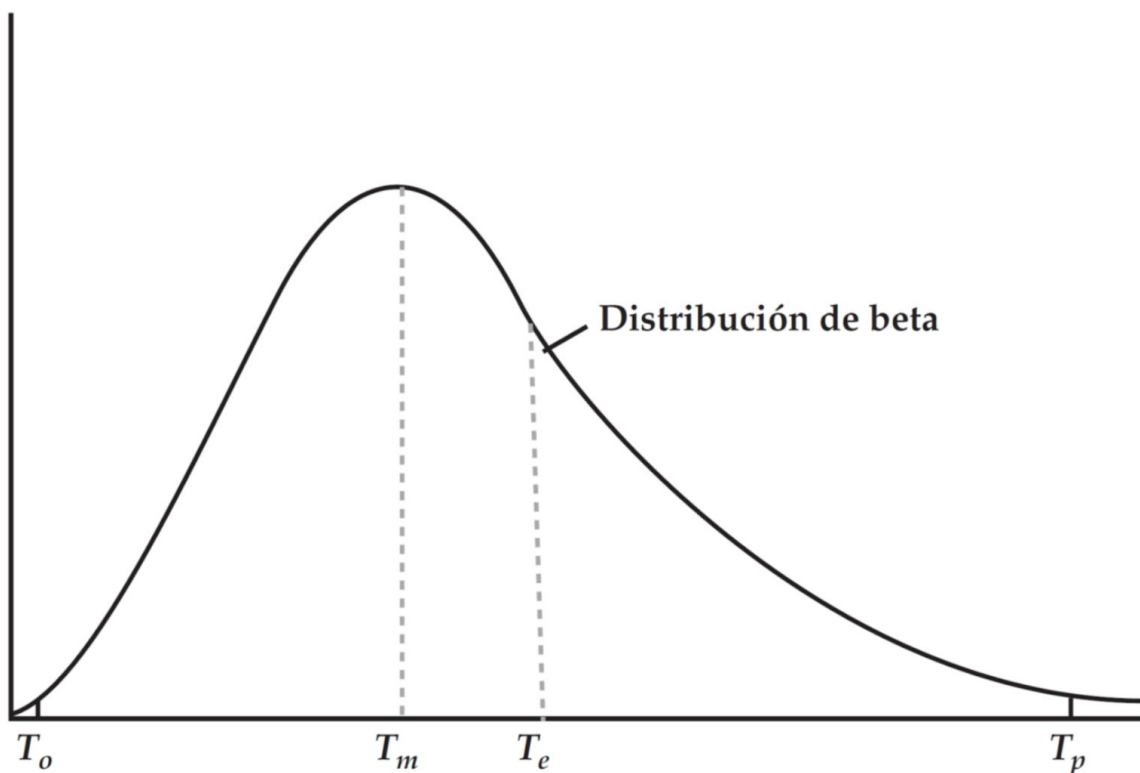
1. **Estado:** Un estado, es un conjunto de condiciones del sistema en un instante determinado que no varían hasta la producción de un evento.
2. **Evento:** es un suceso en el sistema que lleva como consecuencia el cambio en las condiciones de un determinado estado e implica pasar a otro.
3. **Tiempo de simulación:** es el valor de las unidades de tiempo en las que el sistema puede avanzar, estas pueden ser superiores a las reales proporcionando así una aceleración del avance del sistema.

4. **Generador de números aleatorios:** para que un evento ocurra, es necesario que exista un generador de números aleatorios que dependiendo de condiciones generen o no un evento que conlleve el cambio de estado del sistema.

### 3. Probabilidad de Pert

Gracias al análisis estadístico basado en PERT o CPM, es posible determinar la probabilidad de que una actividad o tarea finalice en un determinado instante, esto puede ser muy útil debido a la incertidumbre que la finalización de una tarea puede causar debido a la aparición de riesgos o imprevistos durante el desarrollo de la misma.

La probabilidad PERT está basada en una distribución de probabilidad Beta sesgada a la derecha como se muestra en la siguiente imagen.



Esta probabilidad necesita de tres estimaciones temporales para su cálculo, estas estimaciones, representadas también en la anterior gráfica son las siguientes:

1. **Tiempo optimista ( $T_o$ ):** tiempo o duración que tomaría una actividad o tarea si todo sale perfecto, es decir es el caso ideal. Hay una pequeña variación de alrededor del 1% en la que una tarea puede tardar menos de ese tiempo.

2. **Tiempo pesimista ( $T_p$ ):** tiempo o duración que tomaría una actividad o tarea si todo sale lo más desfavorable posible. Hay una pequeña variación de alrededor del 1% en la que una tarea puede tardar más de ese tiempo.
3. **Tiempo más probable ( $T_m$ ):** también llamado tiempo medio de realización de una tarea, es la estimación más realista de la duración de la tarea, se refiere al tiempo que tardaría en finalizar la tarea en condiciones normales.

Cabe destacar que estos tiempos deben de ser determinados o estimados por el encargado del proyecto. El tiempo real ( $T_e$ ) es normalmente superior al tiempo más probable, esto es debido al optimismo que las personas tienden a tener al calcular o estimar estos tiempos.

La relación entre todas estas variables temporales viene dada por la siguiente fórmula:

$$T_e = \frac{T_o + 4T_m + T_p}{6}$$

### 3.1 Probabilidad de finalización de una tarea o proyecto mediante Pert.

Para determinar la probabilidad de que un determinado proyecto finalice en un determinado plazo, hay que aplicar primeramente el teorema del límite central, según el cual la suma de un grupo de variables aleatorias independientes, distribuidas idénticamente, se aproxima a una distribución normal a medida que el número de variables va aumentando.

**Teorema (del límite central):** Sea  $X_1, X_2, \dots, X_n$  un conjunto de variables aleatorias, independientes e idénticamente distribuidas de una distribución con media  $\mu$  y varianza  $\sigma^2 \neq 0$ . Entonces, si  $n$  es suficientemente grande, la variable aleatoria

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$$

tiene aproximadamente una distribución normal con  $\mu_{\bar{X}} = \mu$  y  $\sigma_{\bar{X}}^2 = \frac{\sigma^2}{n}$ .

Por otro lado sabemos que la media de la distribución normal del tiempo total del proyecto es igual a la suma de los tiempos esperados ( $T_e$ ) de las tareas que se encuentran englobadas por el camino crítico del proyecto.

$$T_E(\text{Proyecto}) = \sum_{\text{ruta crítica}} T_e = \text{Media de la distribución normal}$$

Obviando que la desviación típica del tiempo total es la raíz cuadrada de la varianza de ese tiempo, obtenemos esta fórmula que nos permite calcular el estadístico z de la distribución normal, a partir del cual obtenemos la probabilidad esperada de que el proyecto o tarea finalice en un determinado tiempo T.

$$z = \frac{T - T_E}{\sigma_p}$$

## 4. Algoritmo desarrollado

### 4.1 Visión General. Aplicación de eventos Discretos

Para simular la planificación de un determinado proyecto en el sistema desarrollado, se ha utilizado una sistema generación de eventos discretos lanzados con una determinada probabilidad de Pert.

En la aplicación existen varios estados para una simulación de una planificación de un proyecto, estos estados básicos son “simulando” y “finalizado”. Durante el estado simulando pueden ocurrir varios eventos que determinan a su vez el estado de una tarea que pueden ser: “en espera”, “en proceso” y “finalizada”. A continuación detallo cada uno de los estados y cómo puede pasarse de uno a otro, es decir, que eventos implican este paso.

#### 4.1.1 Estados de la simulación

La simulación puede tener dos estados:

1. **Simulando:** la planificación ha comenzado su simulación y hay tareas en proceso o no finalizadas.
2. **Finalizado:** todas las tareas de la planificación han finalizado.

El paso de un estado a otro funciona como se muestra en la siguiente imagen, es decir, cada vez que surge el evento de fin de una tarea se comprueba si todas han finalizado para pasar al siguiente estado.



#### 4.1.2 Estados de las tareas

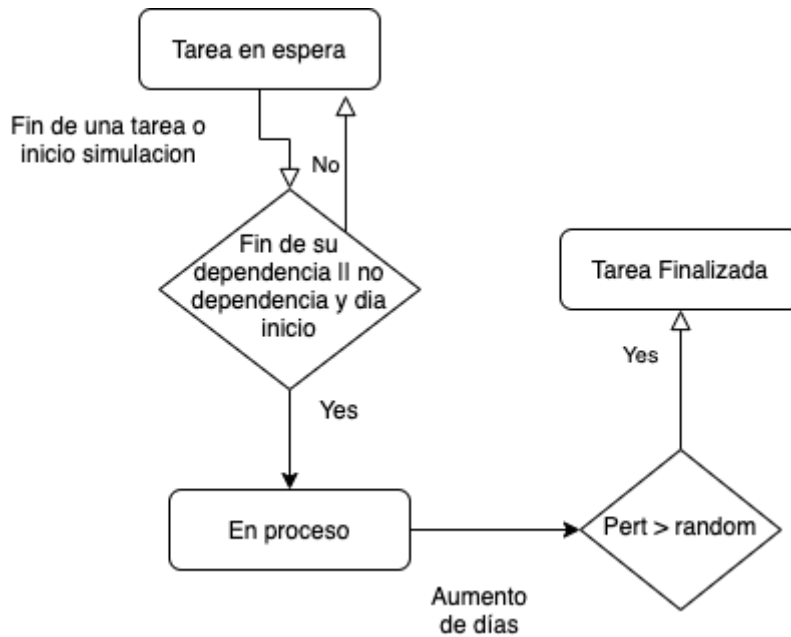
Las diferentes tareas de la planificación pueden tener tres estados:

1. **En espera:** la tarea está esperando que llegue su turno para comenzar su ejecución.
2. **En proceso:** la tarea ha comenzado.
3. **Finalizada:** la tarea ha concluido con éxito.

El paso de un estado a otro funciona como se muestra en la siguiente imagen.

Es destacable decir, que cuando una tarea finaliza, los recursos asociados a ella son liberados y pueden ser asignados a otra tarea.





Como explicación, una tarea puede comenzar cuando se provoca un evento de tipo inicio de simulación o de fin de otra tarea, entonces se comprueba para cada tarea en espera que se cumplan las condiciones de que no tengan dependencia y el día de simulación permite su inicio o que sus dependencias estén finalizadas. Cuando se provoca un evento de avance de días se calcula la estadística de Pert para comprobar qué probabilidad tiene esa tarea de acabar en los días que actualmente lleva en proceso y además se lanza un número random, si se da la condición de que la probabilidad Pert es mayor que el número Random, esa tarea finalizará.

Cabe destacar que cuando una tarea está en proceso puede ocurrir un evento de Riesgo y algún recurso deje de estar disponible o simplemente no haya recursos suficientes para que la tarea continúe, en ese caso es necesario planificar de nuevo y la simulación no podrá continuar hasta realizar esta acción. todo esto se detalla en el capítulo 4.2 de este anexo.

## 4.2 Bucle general de funcionamiento del algoritmo

A continuación se detalla el bucle general de funcionamiento del algoritmo indicando cada uno de sus pasos.

MIENTRAS NO HAY EVENTO SE REPITE:

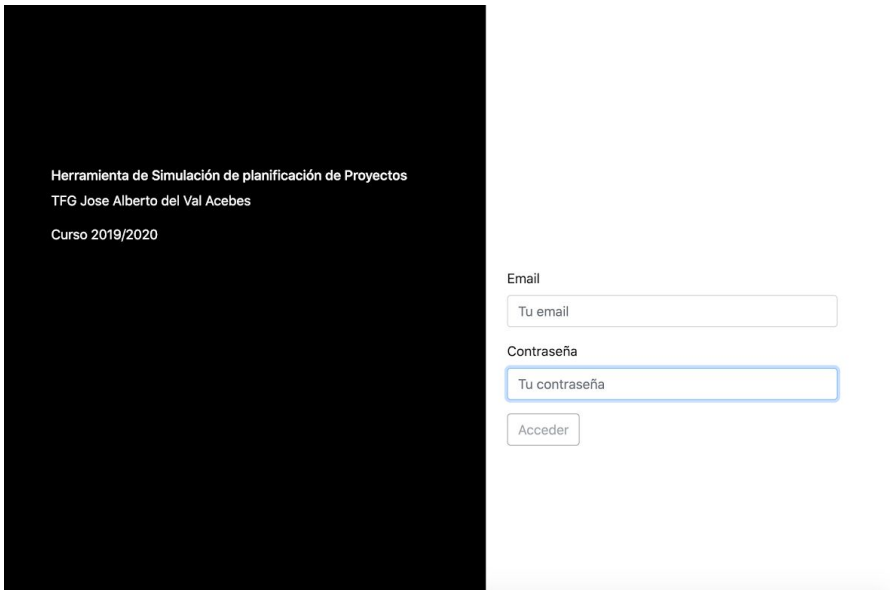
1. Avanza un día.
2. ¿Hay sobreasignación de recursos?
  - 2.1. Si. Lanzamiento evento Sobreasignación. Necesaria replanificación y fin bucle.
  - 2.2. No. Continúa en el paso 3.
3. Formación de una lista de tareas que pueden comenzar (ver 4.1.2 de este anexo).
4. Formación de una lista con tareas comenzadas ( suma de las que pueden comenzar y las que ya estaban en modo running ).
5. ¿Las tareas tienen recursos mínimos para ejecutarse?
  - 5.1. Si. Continúa en el paso 6.
  - 5.2. No. Lanzamiento evento falta recursos. Necesaria replanificación y fin bucle.
6. ¿Ha dejado algún recurso de estar disponible?
  - 6.1. Si. Lanzamiento evento indisponibilidad recurso. Necesaria replanificación y fin bucle.
  - 6.2. No. Continúa en el paso 7.
7. Para las tareas comenzadas cálculo probabilidad pert para finalizar con los días que lleva ejecutándose la tarea.
8. Cálculo de un número Random.
9. Para las tareas comenzadas: ¿Es mayor Pert que el número Random?
  - 9.1. Si. Lanzamiento evento fin de tarea o fin proyecto (ver 4.1.1 de este anexo)
  - 9.2. No. Vuelta al paso 1.

Cabe destacar que los eventos de rotura del bucle pueden lanzarse en los pasos 2.1, 5.2, 6.1 y 9.1,

# ANEXO II. Manual de uso.

## 1. Inicio de sesión.

Para acceder al sistema únicamente es necesario introducir el email del profesor o grupo de alumnos y su contraseña haciendo clic posteriormente a Acceder.



The screenshot shows a login form on a black background. On the left, white text reads: 'Herramienta de Simulación de planificación de Proyectos', 'TFG Jose Alberto del Val Acebes', and 'Curso 2019/2020'. On the right, there are three input fields: 'Email' with 'Tu email', 'Contraseña' with 'Tu contraseña', and an 'Acceder' button.

## 2. Uso panel del alumno.

### 2.1 Crear proyecto.

Para crear una planificación de proyecto, el alumno deberá iniciar sesión con su cuenta grupal y posteriormente deberá hacer clic en la barra de tareas donde se indica crear proyecto o hacer clic en el botón principal que indica la misma operación.

Para crear un proyecto es necesario que el alumno no tenga ninguno activo en el momento de la creación.

Crear un proyecto Panel del Alumno • Cerrar Sesión

Hola Jose Alberto del Val Acebes, no tienes proyectos activos, ¡Crea Uno!

Una vez comenzado esta operación, el sistema mostrará la pantalla inicial de creación del proyecto, el alumno o su grupo en efecto, deberá rellenar el formulario, escoger un tipo de proyecto predefinido por el profesor y hacer clic en siguiente.

Crear un proyecto Panel del Alumno • Cerrar Sesión

### Creación de un proyecto.

Para crear un proyecto cumplimenta el formulario siguiente sin saltarte ningún paso.

Define un nombre y da una descripción a tu proyecto.

Nombre del proyecto.
Descripción del proyecto.

Elige un tipo de proyecto predefinido.

El siguiente paso mostrará los tipos de recursos predefinidos para ese tipo de plantilla y simplemente el alumno deberá introducir una cantidad para cada recurso que crea necesaria para ese tipo de proyecto y pulsará a crear, en caso de que se pulse el botón anterior el sistema retornará la pantalla anterior. Es importante destacar que si algún campo no se rellena el sistema lo tomará como 0.

## Creación de un proyecto.

Para crear un proyecto cumplimenta el formulario siguiente sin saltarte ningún paso.

### Recursos

A continuación propón una lista de recursos que crees que vas a usar para realizar este proyecto.

Nombre del Recurso	Descripción del Recurso	Unidades disponibles
Tutor académico	Profesor de la universidad	<input type="text" value="5"/>
Tutor externo	Tutor de alguna entidad externa a la universidad	<input type="text" value="4"/>
Alumno	Alumno que hace el TFG	<input type="text" value="7"/>
Plataforma de desarrollo	Recursos para realizar el TFG	<input type="text" value="5"/>
Usuarios de prueba	Usuarios para el caso de un proyecto software	<input type="text" value="4"/>
Administración	Entidad administrativa para tramitar el TFG	<input type="text" value="Unidades disponibles."/>

## 2.2 Borrar datos proyecto.

Si existe un proyecto creado o en estado de simulación, el grupo de alumnos puede borrar los datos del proyecto si así lo consideran oportuno, para ello con el proyecto creado simplemente deben hacer clic en Borrar Datos del Proyecto, esto eliminará de forma permanente los datos del proyecto creado o en simulación.

Hola Jose Alberto del Val Acebes, este es tu proyecto abierto:

Proyecto 1

Esperando validación del profesor

\*Al hacer click en Borrar Datos del Proyecto se eliminará todo lo relacionado con este proyecto, tanto lo relacionado con planificación como lo relacionado simulación si esta hubiera comenzado.

## 2.3 Crear planificación del proyecto.

Una vez que el registro de recursos ha sido validado por el profesor, el alumno podrá crear una planificación para este proyecto, añadiendo tareas y asignando recursos a esas tareas, para iniciar este proceso simplemente es necesario hacer clic en Añadir tareas.

Crear un proyecto Panel del Alumno • Cerrar Sesión

---

Hola Jose Alberto del Val Acebes, este es tu proyecto abierto:

Proyecto 1 Añadir Tareas Borrar Datos del Proyecto

\*Al hacer click en Borrar Datos del Proyecto se eliminará todo lo relacionado con este proyecto, tanto lo relacionado con planificación como lo relacionado simulación si esta hubiera comenzado.

Esta operación nos mostrará una pantalla en la que deberemos añadir tareas y sus dependencias entre ellas. Para ello simplemente hacemos click en Añadir otra tarea, y el sistema nos mostrará una tabla con los diferentes campos necesarios a cumplimentar que tienen las tareas. Tipo de tarea es un seleccionable de tipos de tareas predefinidas por el profesor al realizar la plantilla de simulación, nombre de la tarea es el nombre que queremos dar a la tarea, día de inicio, es el día N de simulación en el que estimamos que la tarea puede dar comienzo, es un número entero. Duración de la tarea es un número entero que estima los días que esa tarea va a durar y depende de es un seleccionable en el que se puede escoger una tarea que sea la predecesora de esta misma, esto indicará que dicha tarea no puede comenzar hasta que la predecesora finalice aunque el día de inicio ya haya pasado.

Por otro lado, cómo se ve en la imagen, si se hace clic en eliminar esa tarea creada se borrará, es posible también modificar tareas anteriores.

Una vez estén creadas todas las tareas simplemente debemos hacer clic en asignar recursos que será el paso que se explica a continuación.

Crear un proyecto Panel del Alumno • Cerrar Sesión

---

Hola Jose Alberto del Val Acebes, en este formulario deberás añadir las tareas de las que se compondrá el proyecto:

	TIPO DE TAREA	NOMBRE DE TAREA	DÍA DE INICIO*	DURACIÓN (DÍAS)	DEPENDENCIA DE:	
Tarea 1	Registro de Tema TF	Tarea 1	0	7		Eliminar
Tarea 2	Adscripción alumno	Tarea 2	3	5	Tarea 1	Eliminar

\*DIA DE INICIO: se refiere al día en el que podrá dar comienzo la tarea, siempre y cuando las tareas de las que dependen estén finalizadas.

Añadir otra tarea Asignar Recursos

## 2.4 Asignar recursos a la planificación.

Para asignar recursos a las diferentes tareas nos aparecerá una tabla como esta para cada una de las tareas, simplemente se selecciona la cantidad necesaria que se cree que será la idónea para la ejecución de la tarea y se selecciona confirmar.

Crear un proyecto Panel del Alumno Cerrar Sesión

A continuación realice una primera aproximación de los recursos que utilizará cada tarea.

NOMBRE DE LA TAREA	DURACIÓN DE LA TAREA	RECURSOS																					
Tarea 1	7	<table border="1"><thead><tr><th>NOMBRE DEL RECURSO</th><th>CANTIDAD DISPONIBLE</th><th>CANTIDAD SELECCIONADA</th></tr></thead><tbody><tr><td>Tutor académico</td><td>5</td><td>0</td></tr><tr><td>Tutor externo</td><td>4</td><td>0</td></tr><tr><td>Alumno</td><td>7</td><td>0</td></tr><tr><td>Plataforma de desarrollo</td><td>5</td><td>0</td></tr><tr><td>Usuarios de prueba</td><td>4</td><td>0</td></tr><tr><td>Administración</td><td>0</td><td>0</td></tr></tbody></table>	NOMBRE DEL RECURSO	CANTIDAD DISPONIBLE	CANTIDAD SELECCIONADA	Tutor académico	5	0	Tutor externo	4	0	Alumno	7	0	Plataforma de desarrollo	5	0	Usuarios de prueba	4	0	Administración	0	0
NOMBRE DEL RECURSO	CANTIDAD DISPONIBLE	CANTIDAD SELECCIONADA																					
Tutor académico	5	0																					
Tutor externo	4	0																					
Alumno	7	0																					
Plataforma de desarrollo	5	0																					
Usuarios de prueba	4	0																					
Administración	0	0																					

Ahora ya se podrá ver el estado (diagrama de gantt de la planificación) simplemente haciendo clic en ver estado o entrar en la pantalla de simulación (haciendo clic en Simular Planificación)

Crear un proyecto Panel del Alumno Cerrar Sesión

Hola Jose Alberto del Val Acebes, este es tu proyecto abierto:

Proyecto manual Ver estado Simular Planificación Borrar Datos del Proyecto

\*Al hacer click en Borrar Datos del Proyecto se eliminará todo lo relacionado con este proyecto, tanto lo relacionado con planificación como lo relacionado simulación si esta hubiera comenzado.

## 2.5 Simular planificación.

Para avanzar días en la simulación de la planificación creada simplemente es necesario hacer clic en avanzar días dentro del panel de simulación.

¡Bienvenido a la pantalla de simulación!

Este es el estado de tus tareas, pincha en cualquiera de los botones bajo la tabla para realizar una acción.

NOMBRE DE TAREA:	ESTADO:	DIAS ESTIMADOS*:	DIAS EMPLEADOS*:
Tarea 1	En espera	7	0
Tarea 2	En espera	5	0

Días totales de ejecución del plan:

0

\*El número de días estimados se refiere a la cantidad de días en los que la tarea finalizará según tu estimación.

\*El número de días empleados se refiere a la cantidad de días que la tarea lleva en proceso o a la cantidad de días que ha consumido para completarse en caso de que ya este finalizada.

Una vez que hemos hecho clic en avanzar días, esperamos a que la app nos diga que es lo que ha sucedido, en caso de ser necesario modificar la planificación, simplemente hacemos clic en replanificar y accederemos al menú de modificación de las planificaciones.

## 2.6 Modificar planificación del proyecto.

Crear un proyecto
Panel del Alumno
• Cerrar Sesión

Hola Jose Alberto del Val Acebes, este es tu proyecto abierto:

En la siguiente tabla podrás modificar la cantidad de recursos asignados a cada tarea.

NOMBRE DE LA TAREA	DURACIÓN DE LA TAREA	RECURSOS				
		NOMBRE DEL RECURSO	CANTIDAD TOTAL	CANTIDAD DISPONIBLE	CANTIDAD SELECCIONADA	CANTIDAD INICIAL ESTIMADA
Tarea 1	7	Tutor académico	<input type="text" value="5"/>	2	<input type="text" value="3"/>	3
		Tutor externo	<input type="text" value="4"/>	2	<input type="text" value="2"/>	2
		Alumno	<input type="text" value="7"/>	2	<input type="text" value="5"/>	5
		Plataforma de desarrollo	<input type="text" value="5"/>	3	<input type="text" value="2"/>	2
		Usuarios de prueba	<input type="text" value="4"/>	2	<input type="text" value="2"/>	2
		Administración	<input type="text" value="0"/>	0	<input type="text" value="0"/>	0



En este menú tan solo tendremos que modificar la cantidad total si necesitamos más recursos de cada tipo (alguno ha fallado) o modificar la cantidad seleccionada, hay que tener en cuenta que cuantas más modificaciones más penalización tendremos por parte del tutor cuando utilicemos esta app, ya que todo queda reflejado en el log.

Los recursos son liberados cuando una tarea finaliza.

## 2.7 Ver y descargar resultados.

Una vez la simulación ha finalizado, el alumno puede ver los resultados y entregarlos haciendo clic en ver resultados.

Crear un proyecto Panel del Alumno • Cerrar Sesión

---

Hola Jose Alberto del Val Acebes, este es tu proyecto abierto:

Proyecto manual Ver Resultados Borrar Datos del Proyecto

\*Al hacer click en Borrar Datos del Proyecto se eliminará todo lo relacionado con este proyecto, tanto lo relacionado con planificación como lo relacionado simulación si esta hubiera comenzado.

Para realizar las anteriores acciones simplemente una vez en el menú de resultados se hace clic en el botón correspondiente.

Crear un proyecto Panel del Alumno • Cerrar Sesión

---

Hola Jose Alberto del Val Acebes, este es tu proyecto abierto:

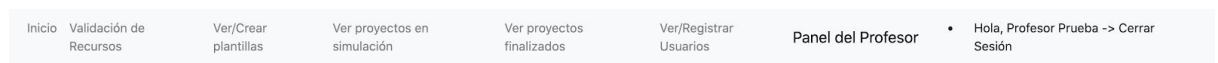
DIA OCURRENCIA	ACCIÓN	RECURSO IMPLICADO	TAREA IMPLICADA
0	Error mínimo recursos	Tutor académico	Tarea 1
0	Error mínimo recursos	Administración	Tarea 1
0	replanificacion		No procede
9	Fin de una tarea	No procede	Tarea 1
9	replanificacion		No procede
9	Error mínimo recursos	Tutor académico	Tarea 2
16	Fin de la simulación, el proyecto acabó	No procede	No procede
16	Fin de una tarea	No procede	Tarea 2

Volver Descargar tabla en formato csv Enviar Resultados

## 2. Uso panel del profesor o administrador.

En este subapartado del anexo, se muestran las instrucciones para realizar cualquier acción siendo un usuario de tipo profesor.

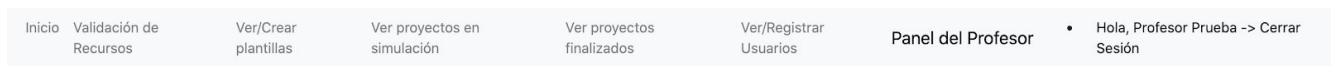
Una vez logeado como profesor aparecerá el siguiente menú.



Hola Profesor Prueba, bienvenido!!:

### 2.1 Validar Recursos

Para validar los recursos de algún proyecto creado por algún alumno simplemente se deberá hacer clic en validación de recursos, así aparecerá la ventana principal de validación, en la que se muestran todos los proyectos necesarios de esta acción.



A continuación se muestra una lista de proyectos pendientes de validar sus recursos.

Nombre del alumno.	Nombre del proyecto.	Tipo del proyecto.	Acción.
Jose Alberto del Val Acebes	Proyecto de prueba	TFG	<input type="button" value="Validar Recursos"/>

Una vez elegido el proyecto que se quiere validar simplemente se hace clic en Validar recursos del proyecto seleccionado y se mostrará la ventana de detalle de ese proyecto, en la que aparece una tabla con los recursos y su cantidad seleccionada por el alumno, en esta pantalla simplemente se modifica lo que se quiera y para finalizar se hace clic en registrar selección.

## VALIDACIÓN DE RECURSOS

Realiza los cambios oportunos. Si consideras que no es necesario realizar cambios deja los campos tal cuál.

Nombre del recurso.	Descripción.		
Tutor académico	Profesor de la universidad	<input type="text" value="5"/>	<input type="button" value="Eliminar Recurso"/>
Tutor externo	Tutor de alguna entidad externa a la universidad	<input type="text" value="4"/>	<input type="button" value="Eliminar Recurso"/>
Alumno	Alumno que hace el TFG	<input type="text" value="2"/>	<input type="button" value="Eliminar Recurso"/>
Plataforma de desarrollo	Recursos para realizar el TFG	<input type="text" value="4"/>	<input type="button" value="Eliminar Recurso"/>
Usuarios de prueba	Usuarios para el caso de un proyecto software	<input type="text" value="2"/>	<input type="button" value="Eliminar Recurso"/>
Administración	Entidad administrativa para tramitar el TFG	<input type="text" value="7"/>	<input type="button" value="Eliminar Recurso"/>

## 2.2 Ver y crear plantillas de planificación

Para ver o crear una plantilla el profesor deberá hacer clic en Ver/crear plantillas, se mostrará un menú general con todas las plantillas desde el que se puede acceder al detalle de cada una o crear otra nueva.

Para acceder al detalle simplemente se hace clic en ver detalle en la plantilla seleccionada, para crear una nueva se hará clic en crear plantilla y se seguirán los pasos.

Estas son las plantillas que actualmente están creadas, para crear una nueva haz clic en crear al final de esta lista.

Tipo	Descripción	Acción
Proyecto de construcción	Es un proyecto para probar la funcionalidad completa de esta aplicación	<input type="button" value="Ver detalle"/>
TFG	Son las fases que tenemos que realizar para un buen Trabajo de Fin de Grado	<input type="button" value="Ver detalle"/>
Implantación red inalámbrica	Implantación de una red inalámbrica en una nave y oficinas	<input type="button" value="Ver detalle"/>

## 2.3 Ver proyectos en simulación

Para ver y monitorizar una simulación, el profesor deberá hacer clic en Ver Proyectos en simulación, se mostrará un menú general con todos los proyectos que están actualmente en estado de simulación o que han finalizado y aún no se han entregado para evaluar por medio de los alumnos o grupos de ellos.

Para acceder al detalle simplemente se hace clic en ver detalle o ver resultados en el botón correspondiente al proyecto elegido.

Inicio	Validación de Recursos	Ver/Crear plantillas	Ver proyectos en simulación	Ver proyectos finalizados	Ver/Registrar Usuarios	Panel del Profesor	Hola, Profesor Prueba -> Cerrar Sesión
--------	------------------------	----------------------	-----------------------------	---------------------------	------------------------	--------------------	--

A continuación se muestra una lista de proyectos en estado de simulación, haz clic en ver detalle para más información.

Nombre del alumno.	Email del alumno.	Nombre del proyecto.	Días empleados en el proyecto.	Tareas completadas.	Estado.	Acción.
Grupo 01	grupo@prueba.com	Prueba proyecto	10	2	finalizado	<a href="#">Ver resultados</a>
Pepito	pepito@palot.es	Mision Saturno	0	0	simulando	<a href="#">Ver detalle</a>

## 2.4 Ver proyectos entregados y descarga de resultados.

Para ver los resultados que un alumno ha entregado de una simulación, el profesor deberá hacer clic en Ver Proyectos finalizados, se mostrará un menú general con todos los proyectos que están actualmente que han finalizado y han sido entregados por parte del alumno para evaluar.

Para acceder al detalle simplemente se hace clic en ver detalle o ver resultados en el botón correspondiente al proyecto elegido, dentro de este detalle se podrá descargar el fichero csv correspondiente a los resultados de ese proyecto.

Inicio	Validación de Recursos	Ver/Crear plantillas	Ver proyectos en simulación	Ver proyectos finalizados	Ver/Registrar Usuarios	Panel del Profesor	Hola, Profesor Prueba -> Cerrar Sesión
--------	------------------------	----------------------	-----------------------------	---------------------------	------------------------	--------------------	--

Estos son las simulaciones que actualmente están entregadas para evaluar, para ver el detalle de cada una puedes hacer clic en Ver detalle.

Nombre Grupo	Componentes	E-mail	Acción
Jose Alberto del Val Acebes		pepeino2@hotmail.com	<a href="#">Ver detalle</a>
miguel delibes		miguel@delibes.com	<a href="#">Ver detalle</a>

## 2.5 Ver/Registrar/Eliminar Usuarios

Para ver los usuarios que hay registrados en el sistema, el profesor deberá hacer clic en Ver/Registrar Usuarios, se mostrará un menú general con todos los usuarios actualmente activos en el sistema.

Para eliminar un usuario simplemente se hará clic en eliminar en el usuario seleccionado, para crear uno nuevo se hará clic en crear usuario y se rellenará el formulario de registro como el que aparece a continuación.

Inicio   Validación de Recursos   Ver/Crear plantillas   Ver proyectos en simulación   Ver proyectos finalizados   Ver/Registrar Usuarios   Panel del Profesor   • Hola, Profesor Prueba -> Cerrar Sesión

Estos son los alumnos que actualmente están creados, para crear uno nuevo haz clic en crear al final de esta lista.

Grupo	Componentes	E-mail	Rol	Acción
Grupo 01	Juan Fernandez, Miguel Lopez, Jose Luis Rodriguez	grupo@prueba.com	Alumno	<input type="button" value="Eliminar"/>
Pepito	Palotes	pepito@palot.es	Alumno	<input type="button" value="Eliminar"/>
Profesor Prueba		profesor@prueba.com	Profesor	
Jose Alberto del Val Acebes		pepeino2@hotmail.com	Alumno	<input type="button" value="Eliminar"/>

# ANEXO III. Manual de despliegue.

## 1. Requisitos previos.

En este anexo se dan una serie de instrucciones para desplegar la aplicación en el **hosting** de **google firebase**, servicio totalmente gratuito.

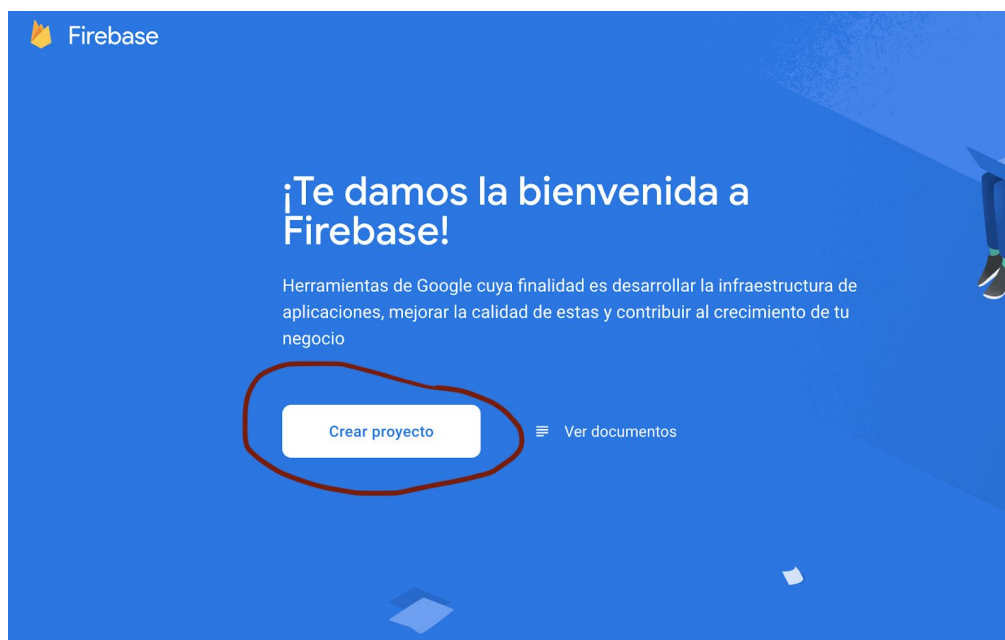
Para poder desplegar la aplicación en dicho hosting, es necesario tener una cuenta de google activa, que será la que administre dicho proyecto, para crear una cuenta de google se puede acceder a este enlace

<https://accounts.google.com/signup/v2/webcreateaccount?flowName=GlifWebSignIn&flowEntry=SignUp> y seguir los pasos que se indican.

## 2. Creación de un proyecto en Firebase.

Para que la aplicación pueda usar los servicios de **firebaseAuth**, firestore y a su vez el **hosting** es necesario crear un proyecto en la consola de firebase, para ello se seguirán los siguientes pasos:

1. Acceder a <https://console.firebase.google.com/?pli=1> y logearse con la cuenta de google.
2. Hacer clic en crear proyecto



3. Introducir el nombre del proyecto y hacer clic en continuar

× Crear proyecto(paso 1 de 2)

---

## Empieza por ponerle un nombre al proyecto <sup>?</sup>

### Introduce el nombre de tu proyecto

Debes indicar el nombre del proyecto. !

Acepto los [términos de Firebase](#).

Continuar

4. Es aconsejable deshabilitar google Analytics, por último se hace clic en Crear Proyecto.

× Crear proyecto(paso 2 de 2)

informes y realizar tareas de segmentación (entre otras acciones) en Firebase Crashlytics, Cloud Messaging, In-App Messaging, Remote Config, A/B Testing, Predictions y Cloud Functions.

Google Analytics incluye estos servicios:

<input checked="" type="checkbox"/> Pruebas A/B <sup>?</sup>	<input checked="" type="checkbox"/> Usuarios sin fallos <sup>?</sup>
<input checked="" type="checkbox"/> Segmentación de usuarios en productos de Firebase <sup>?</sup>	<input checked="" type="checkbox"/> Activadores de Cloud Functions basados en eventos <sup>?</sup>
<input checked="" type="checkbox"/> Predicción del comportamiento de los usuarios <sup>?</sup>	<input checked="" type="checkbox"/> Informes gratuitos ilimitados <sup>?</sup>

**Habilitar Google Analytics en este proyecto**  
Recomendado

[Anterior](#) Crear proyecto

### 3. Creación de una aplicación dentro del proyecto creado.

Una vez creado el proyecto de **firebase**, es necesario crear una aplicación web dentro de él como se indica a continuación.

1. Acceder a la consola de **firebase** y luego acceder al proyecto creado.



2. Hacer clic en añadir app web (símbolo `</>`)

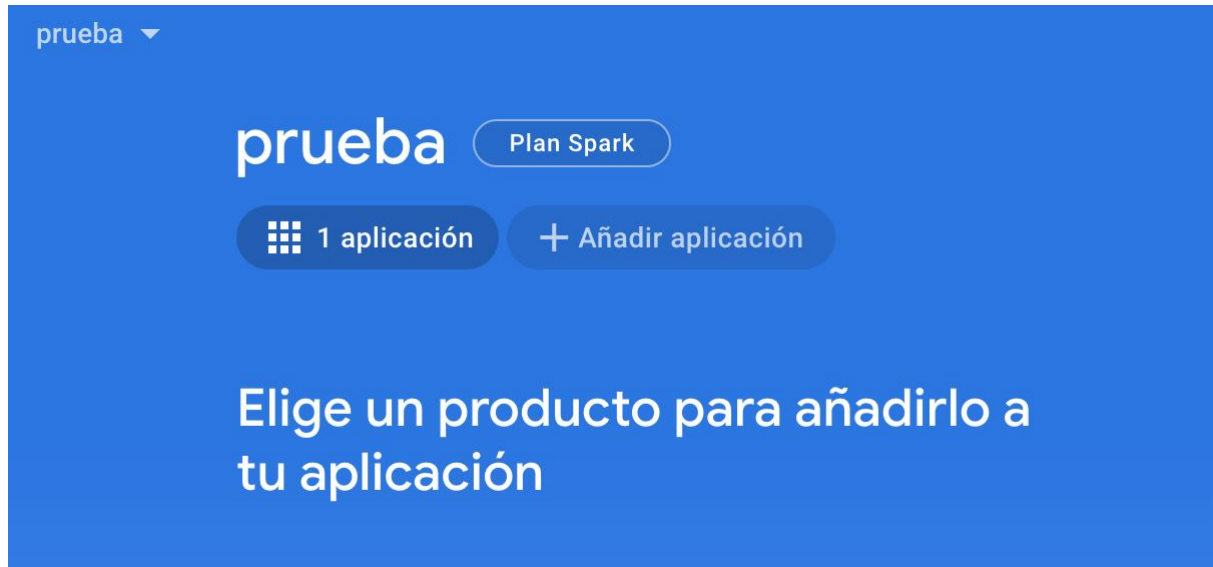




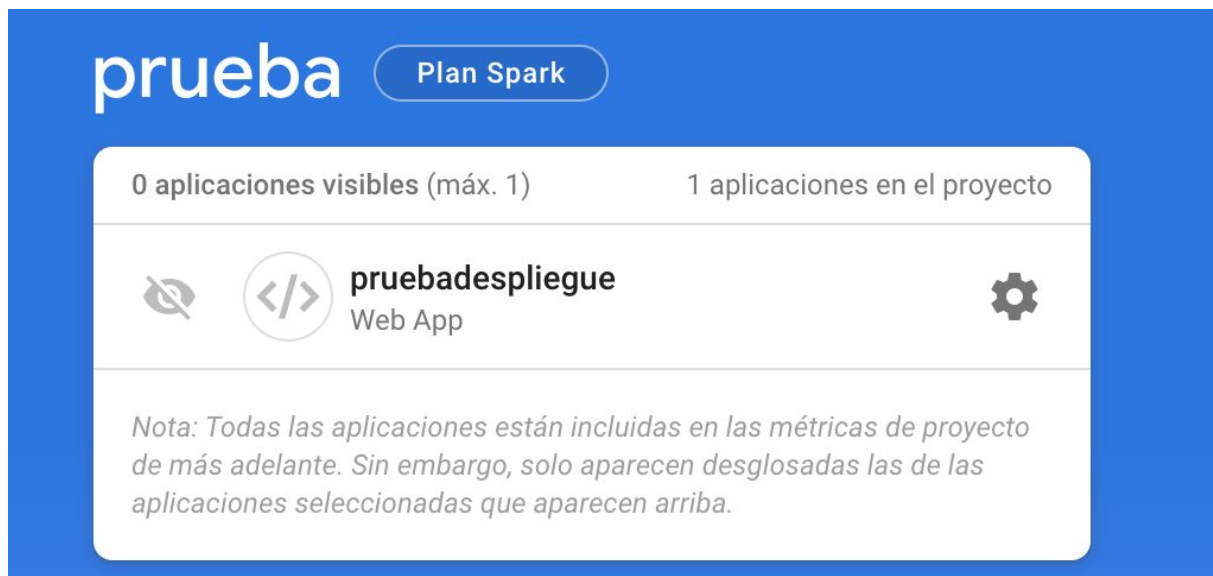
## 4. Configuración del proyecto angular.

Una vez creada la aplicación dentro del proyecto de **firebase**, debemos configurar nuestro proyecto angular para conectarlo con dicha aplicación de **firebase** para ello una vez más dentro del proyecto en la consola de firebase, seguimos estos pasos:

1. Hacemos clic en 1 aplicación.



2. Hacer clic en la ruletita de ajustes.



3. En la pantalla principal de ajustes navegamos hacia el final y copiamos el key de la app:

## 2 Añadir SDK de Firebase

Antes de utilizar cualquier servicio de Firebase, copia y pega estas secuencias de comandos en la parte inferior de la etiqueta <body>:

```
<!-- The core Firebase JS SDK is always required and must be listed first -->
<script src="https://www.gstatic.com/firebasejs/7.14.6/firebase-app.js"></scrip

<!-- TODO: Add SDKs for Firebase products that you want to use
      https://firebase.google.com/docs/web/setup#available-libraries -->

<script>
  // Your web app's Firebase configuration
  var firebaseConfig = {
    apiKey: "AIzaSyAaoHg1Rr4IR_2BTrq8yRp0LHNlp-ta-Pc",
    authDomain: "prueba-964b5.firebaseio.com",
    databaseURL: "https://prueba-964b5.firebaseio.com",
    projectId: "prueba-964b5",
    storageBucket: "prueba-964b5.appspot.com",
    messagingSenderId: "298643516179",
    appId: "1:298643516179:web:cbc72a0a23a4ebdf6a08e6"
  };
  // Initialize Firebase
  firebase.initializeApp(firebaseConfig);
</script>
```

Más información sobre Firebase para aplicaciones web: [Primeros pasos](#), [Referencia de APIs del SDK web](#) y [Muestras](#)

[Ir a la consola](#)

El key es la variable ***firebaseConfig***, que debemos de copiar y pegar en nuestro proyecto angular en los ficheros `src/app/environments/environment.ts` y `src/app/environments/environment.prod.ts` como el proyecto ya estaba configurado para una anterior key no habrá mayor problema, únicamente se deberá de sustituir la key de tal forma que respectivos ficheros queden de la siguiente manera:

```

export const environment = {
  production: true,

  firebaseConfig : {
    apiKey: "AIzaSyAaoHg1Rr4IR_2BTrq8yRp0LHNlp-ta-Pc",
    authDomain: "prueba-964b5.firebaseio.com",
    databaseURL: "https://prueba-964b5.firebaseio.com",
    projectId: "prueba-964b5",
    storageBucket: "prueba-964b5.appspot.com",
    messagingSenderId: "298643516179",
    appId: "1:298643516179:web:cbc72a0a23a4ebdf6a08e6"
  }
};

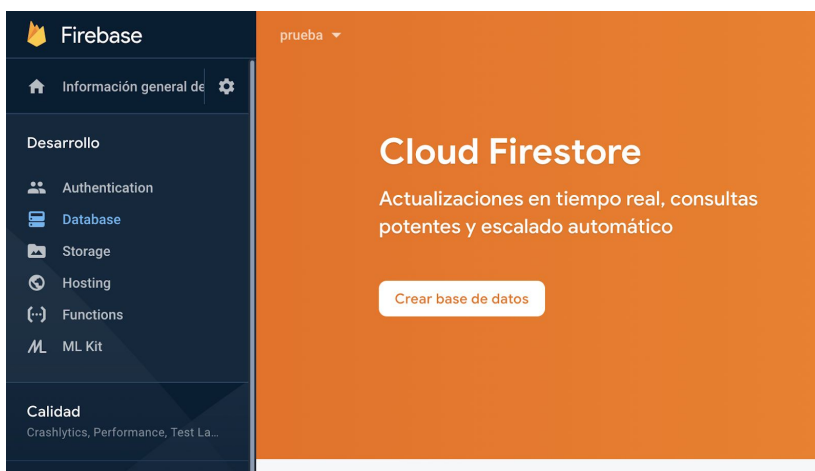
```

Esta imagen corresponde al fichero environment.prod.ts, el environment.ts deberá ser igual solo que con la variable production: false

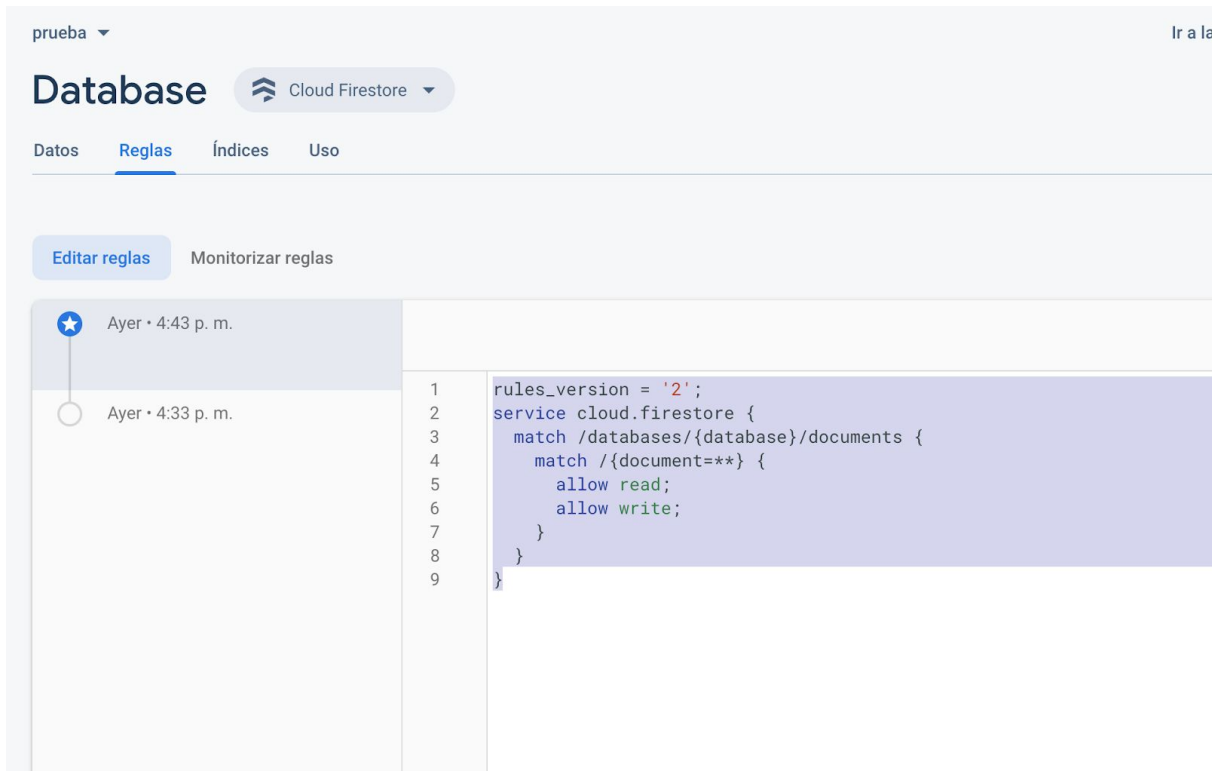
## 5. Configuraciones y creación de servicios en la app firebase.

Antes de compilar y desplegar la aplicación, se deben de configurar todos los servicios de firebase que utilizaremos, estos son firebaseAuth y firestore, para ello nos acercamos a la consola de firebase, accedemos a la aplicación de nuestro proyecto y seguimos los siguientes pasos:

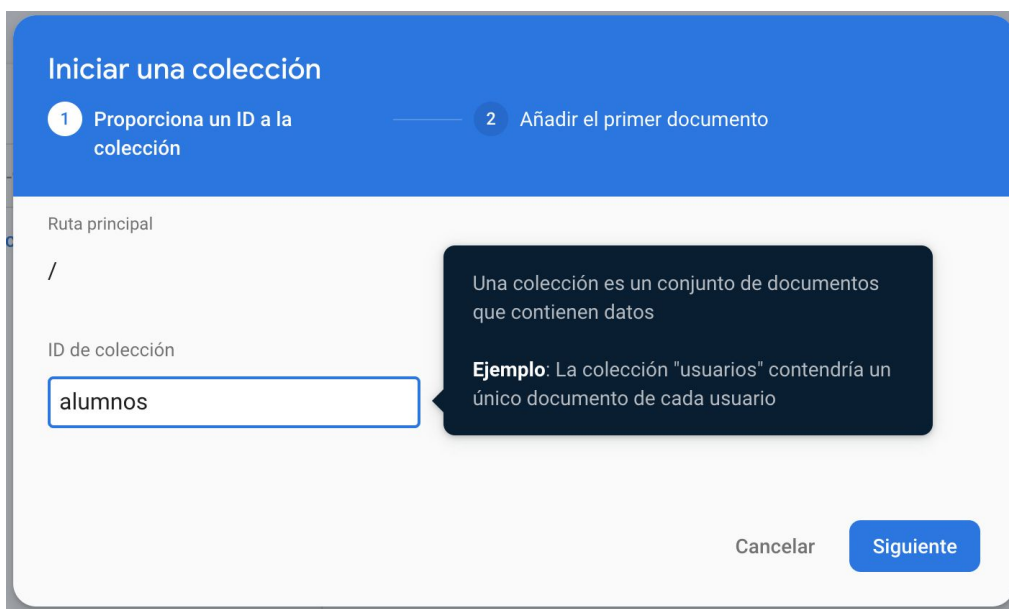
1. Creación BBDD, para ello únicamente es necesario hacer clic en Database y seguidamente en Crear Base de Datos.



2. Cambio de reglas de la BBDD, para ello una vez creada la BBDD únicamente es necesario hacer clic en Reglas y modificarlas como se indica en la siguiente imagen.



3. Población de la BBDD con un usuario administrador, para ello en la pestaña Datos de la imagen anterior hacemos clic en iniciar Colección y creamos una colección nueva llamada alumnos, posteriormente poblamos esta colección con un usuario que será administrador del sistema.



Para poblar la colección con el primer usuario debemos crear un documento con los campos de la imagen, el email, password son los datos con los que el futuro usuario iniciará sesión en el sistema, el rol debe de ser profesor.

Ruta principal del documento  
/alumnos

ID del documento  
ID automático

Campo	Tipo	Valor
email	string	admin@sistema.
password	string	admin
rol	string	profesor
nombre	string	Profesor admin

+ Añadir campo


Cancelar Guardar

4. Creación servicio de autenticación. Accedemos a la pestaña de Authentication dentro de la página principal de nuestra app y hacemos clic en configura el método de inicio de sesión.

Buscar por dirección de correo electrónico, número de teléfono o UID de usuario

Añadir usuario

Identificador	Proveedores	Fecha de creación	Inicio de sesión	UID de usuario ↑
---------------	-------------	-------------------	------------------	------------------



Autentica y administra usuarios de una gran variedad de proveedores sin necesidad de utilizar ningún código del servidor

[Más información](#) [Consultar la documentación](#)

Configura el método de inicio de sesión

Habilitamos correo electrónico/contraseña como medio de inicio de sesión y hacemos clic en guardar.

✉ Correo electrónico/contraseña

Habilitar

Permite a los usuarios registrarse con su dirección de correo electrónico y contraseña. Nuestros SDK también proporcionan la verificación de la dirección de correo, la recuperación de contraseñas y las primitivas de cambio de dirección de correo. [Más información](#)

Enviar enlace por correo electrónico (inicio de sesión sin contraseña)  Habilitar

Cancelar **Guardar**

5. Creación del usuario administrador en el servicio de autenticación. Para ello es necesario hacer clic en añadir usuario e introducir como email y contraseña los introducidos anteriormente en la BBDD y volver a hacer clic en Añadir usuario.

🔍 Buscar por dirección de correo electrónico, número de teléfono o UID de usuario **Añadir usuario** ↻ ⋮

Identificador	Proveedores	Fecha de creación	Inicio de sesión	UID de usuario ↑
Este proyecto aún no tiene usuarios				

**Añade un usuario de correo electrónico/contraseña**

Correo electrónico

Contraseña

Se debe indicar un correo electrónico válido ! Se debe indicar una contraseña !

Cancelar **Añadir usuario**

## 6. Compilación y despliegue.

Para finalizar el despliegue es necesario loguearse e iniciar el servicio de **firebase** en nuestro proyecto, para ello es necesario abrir una terminal en la carpeta raíz del código fuente y realizar estos pasos:

1. Login en **firebase**.
2. Inicio de **firebase**.
3. Compilación del proyecto angular.
4. Despliegue.

Veamos a continuación en detalle cada uno de estos pasos, ilustrados con una captura de pantalla en la que se puede ver lo comentado en la descripción:

1. **Login en firebase.** Para iniciar el login ponemos el comando : `$ firebase login` en la terminal que tenemos abierta, este comando nos lanzará una pregunta a la cual debemos de responder con un carácter 'y' y posteriormente presionar Enter. Esto nos abrirá una pantalla de login en el que habrá que introducir los datos correspondientes a nuestra cuenta de google, en la cual tenemos el proyecto **firebase** creado.

```
MacBook-Pro-de-Jose:proyectoPlanificacion 2 josealbertodelval$ firebase login
i Firebase optionally collects CLI usage and error reporting information to help improve our products. Data is collected in accordance with Google's privacy policy (https://policies.google.com/privacy) and is not used to identify you.
? Allow Firebase to collect CLI usage and error reporting information? (Y/n)
```

2. **Inicio de firebase.** Una vez estamos logeados ya en **firebase**, es hora de iniciar el asistente de **firebase** para vincular el proyecto de **firebase** con nuestro proyecto, para ello tecleamos el comando : `$ firebase init` el cual nos arrojará una pantalla como la de la imagen, en la cual debemos desplazarnos hasta la opción **hosting** presionar espacio para seleccionarla y presionar enter para confirmar.

Una vez realizado esto, el asistente nos pedirá elegir un proyecto para vincular, en nuestro caso solo tendremos uno, seleccionamos usar un proyecto existente y pulsamos intro.

```
First, let's associate this project directory with a Firebase project.
You can create multiple project aliases by running firebase use --add,
but for now we'll just set up a default project.

? Please select an option: (Use arrow keys)
> Use an existing project
  Create a new project
  Add Firebase to an existing Google Cloud Platform project
  Don't set up a default project
```

El siguiente paso será configurar el directorio público que usará el **hosting** para desplegar nuestra aplicación, para ello, debido a que es un proyecto **angular** denominado **tfg**, debemos responder con una dirección como esta : `dist/tfg`, tal cual se muestra en la imagen y pulsar intro de nuevo.

### === Hosting Setup

```
Your public directory is the folder (relative to your project directory) that
will contain Hosting assets to be uploaded with firebase deploy. If you
have a build process for your assets, use your build's output directory.
```

```
? What do you want to use as your public directory? dist/tfg
```

El siguiente paso, está relacionado directamente con las aplicaciones de página única o mono páginas (*SPA, Single Page Applications*), en este caso Angular tiene su propio router de páginas así que se escribe “**y**” y presionamos Intro.

```
? Configure as a single-page app (rewrite all urls to /index.html)? (y/N) y
```

Después de esto la herramienta se dará cuenta de que ya existe un archivo `index.html`, por tanto pregunta que si queremos sobrescribir, a lo que respondemos “**N**” y pulsamos Intro, para conservar el `index` de nuestro proyecto y ya habríamos acabado por fin la configuración.

```
? File dist/tfg/index.html already exists. Overwrite? (y/N) N
```

### 3. Compilación del proyecto Angular.

Antes de iniciar el despliegue es necesario que compilemos nuestra aplicación Angular, esto se realiza con el comando: `$ng build`, este proceso tomará aproximadamente medio minuto.

### 4. Despliegue.

Ya tenemos el proyecto compilado y solo falta desplegarlo, para ello simplemente utilizamos el comando: `$ firebase deploy`, este proceso tomará aproximadamente entre 2 y 5 minutos.



Al finalizar nos mostrará una URL del **hosting** a través de la cual se encontrará la aplicación alojada y desplegada para su uso.

```
✓ Deploy complete!  
Project Console: https://console.firebase.google.com/project/prueba-964b5/overview  
Hosting URL: https://prueba-964b5.web.app
```

## ANEXO IV. Repositorio del código fuente.

Para consultar el código fuente de dicho TFG se puede acceder al siguiente repositorio de GitHub, en el que se encuentra alojado:

<https://github.com/Joserto97/tfgPlanificaciones>