



**ULPGC**  
Universidad de  
Las Palmas de  
Gran Canaria

Escuela de  
Ingeniería Informática



---

# **Desarrollo de una Aplicación Web para la Elaboración de Proyectos Docentes**

Grado en Ingeniería Informática

Iván Jorge Dos Santos Lopes

---

TUTORIZADO POR:

Agustín Salgado de la Nuez

Nelson Monzón López

Julio 2021

# Agradecimientos

Ante todo, me gustaría agradecer a mis amigos más íntimos y, en especial, a mi familia que me incentivaron a terminar la carrera y apoyarme en todo el desarrollo del proyecto. En 2015 fui de Erasmus a Polonia por un periodo de un año, y donde, a finales del Erasmus encontré una oportunidad de empezar mi primer trabajo como un profesional. Empecé a trabajar aquí en Polonia, donde resido actualmente, como Júnior en Desarrollo Software. Los años fueron pasando y deje un poco la carrera hacia un lado. Con el trabajo fue un poco difícil para mí centrarme en las dos cosas, pero como me faltaba solo terminar el Trabajo Fin de Grado, tenía que terminar todo.

Así que este año me puse todas las ganas y fuerzas para terminar el proyecto. Un agradecimiento en especial a mi Padre *Antonio Manuel Da Silva Adrião Lopes*, que me proporcionó la oportunidad de ir a estudiar en Las Palmas. Sin Él hoy no podría haberlo logrado.

También, agradecer a mi tutor Agustín Salgado de la Nuez, por el seguimiento a lo largo del proyecto, preparar el diseño de la interfaz, preparar la documentación necesaria para los trámites del Trabajo Fin de Título y por la paciencia para terminar el trabajo, dado que todo fue hecho de forma remota, y con tutorías por *Microsoft Teams*.

Por último, agradecer a mi novia Magdalena que siempre estuvo ahí, recordándome, apoyándome para poder terminar el proyecto.

# Índice general

<b>1. Introducción .....</b>	<b>5</b>
<b>2. Estado del arte .....</b>	<b>7</b>
<b>3. Objetivos .....</b>	<b>10</b>
<b>4. Competencias.....</b>	<b>12</b>
<b>5. Aportaciones .....</b>	<b>14</b>
<b>6. Metodología .....</b>	<b>15</b>
<b>7. Recursos Utilizados.....</b>	<b>18</b>
<b>8. Planificación .....</b>	<b>24</b>
<b>9. Desarrollo de la Aplicación .....</b>	<b>25</b>
<b>9.1 Pila de Producto -----</b>	<b>25</b>
<b>9.2. Sprint Zero-----</b>	<b>27</b>
<b>9.3. Sprint 1 -----</b>	<b>37</b>
<b>9.4. Sprint 2 -----</b>	<b>43</b>
<b>9.5 Sprint 3 -----</b>	<b>51</b>
<b>10. Presupuesto.....</b>	<b>54</b>
<b>11. Conclusiones .....</b>	<b>55</b>
<b>12. Trabajo Futuro .....</b>	<b>56</b>
<b>13. Anexo I. Pila de Producto .....</b>	<b>57</b>
<b>14. Anexo II. Mockups.....</b>	<b>60</b>
<b>15. Anexo III. Manual de Usuario .....</b>	<b>66</b>
<b>16. Bibliografía .....</b>	<b>99</b>

# 1. Introducción

Anualmente el coordinador de cada asignatura de la ULPGC debe acceder a una aplicación web para la confección del PDA de dicha asignatura. Esta aplicación web incorpora una serie de campos de texto que el profesor coordinador debe rellenar. Una vez finalizada la cumplimentación del PDA ésta se envía y, posteriormente, es examinada por la Comisión de Asesoramiento Docente que dictamina si cumple con los requisitos exigidos o se detecta algún tipo de anomalía (leve o grave) que requiere una corrección por parte del coordinador. Muchos de los motivos de rechazo de los PDA se deben a causas de forma y no de contenido.

Dado este problema, este TFG se centra en el diseño y desarrollo de una aplicación web para la gestión de los Proyectos Docentes, con una interfaz que facilita su elaboración. Para el desarrollo de esta aplicación se ha intentado seguir los principios utilizados en la metodología de Kanban, definiendo el proyecto en múltiples tareas menores.

La aplicación está definida con los 5 apartados (menús) utilizados en la Guía Docente en la Universidad de Las Palmas de Gran Canaria: *Datos Generales, Plan de Enseñanza, Plan de Aprendizaje, Plan Tutorial, Bibliografía*. En cada apartado, el usuario tiene a su disposición una lista de campos de texto para la definición adecuada de cada asignatura que directamente es guardada en una base de datos. Con eso se tiene un sistema único y una interfaz intuitiva.

El proyecto está dividido en varias iteraciones, donde se desarrollan las funcionalidades que componen la aplicación final. La aplicación ha sido desarrollada en *Firebase* de *Google*.

## 1.1 Motivaciones

La herramienta actual es susceptible de una serie de mejoras, comentadas a continuación, que serán abordadas en el presente trabajo fin de título.

- *Desperdicio de esfuerzo de trabajo.*

Dado que muchos de los motivos de rechazo se deben a causas de forma, sería deseable que una herramienta informática detectara y notificara al coordinador esos errores evitando que la CAD sea quién lo haga. El PDA no se podría enviar hasta la corrección

de esos defectos. De esta forma se agilizaría el proceso de aprobación de los PDA's y se descargaría de trabajo a los miembros de la CAD de cada centro en la verificación de esos aspectos formales.

- *Modelado y unificación de aspectos de diseño del PDA.*

La inmensa mayoría de los PDA's comparten aspectos comunes en lo referente a las "Fuentes de Evaluación", "Acciones Formativas", "Recursos", etc. Facilitaría al profesor de la asignatura la disponibilidad de un catálogo de elementos que pudiera incorporar con un mínimo de esfuerzo. Además, sería muy fácil contrastar esta información con la presente en otros campos del PDA, mediante el establecimiento de referencias cruzadas.

- *Sistema Centralizado.*

Obtener una plataforma única para la gestión de todas las asignaturas del PDA y que permite enfrentar mejor cualquier problema en cualquiera de ellas. Con este método se tiene un padrón de diseño único y estandarizado para todos, así como el formato de los datos será igual para cualquier usuario.

## 2. Estado del arte

Este proyecto se basa en la aplicación web para la elaboración de los proyectos docentes de la ULPGC. En la herramienta actual, el coordinador de cada asignatura debe rellenar, de forma manual, una serie de campos de textos. Es un proceso que conlleva mucho esfuerzo por parte del coordinador y desperdicio de tiempo. La aplicación propuesta en este TFG tiene como objetivo de mejorar y facilitar la elaboración de PDA's al coordinador interconectando información dentro del proyecto docente reduciendo la posibilidad de cometer errores.

The screenshot shows a web form titled "METODOLOGÍAS DEL DESARROLLO ÁGIL". In the top right corner, it displays "Proyecto Docente: 47359" and "Pantalla: Datos Generales". Below the title, there is a link: "Si desea consultar la guía básica asociada a este proyecto docente haga clic [aquí](#)". A navigation menu includes "Datos Generales" (selected), "Plan de Enseñanza", "Plan de Aprendizaje", "Plan Tutorial", "Bibliografía", and "volver". A text input field for "Código Unesco:" contains the value "1203". Below this is a section titled "Requisitos Previos" with a text area containing "Ingeniería del Software I" and "Ingeniería del Software II". Further down is the section "Lengua de Impartición (Especificar número de créditos en cada Lengua)", which includes a table for "Nº Créditos":

Nº Créditos	
Español	6
Inglés	
Otra	

Figura. 1. Ejemplo de la herramienta actual – Datos generales

Contribución de la asignatura al perfil profesional
La asignatura de Metodologías de Desarrollo Ágil se enmarca dentro de la intensificación de Ingeniería del Software. Esta asignatura se centra en el estudio de las metodologías que permiten organizar el desarrollo de productos software, especialmente las metodologías denominadas ágiles, que persiguen la obtención de un producto final de manera rápida y eficiente. En esta asignatura se imparten los conceptos básicos que subyacen a las metodologías ágiles, prestando especial atención a varias de las metodologías de desarrollo más conocidas.
Competencias que tiene asignadas
G1, G2, G3, G4, G5, N1, N2, N3, N4, N5, IS01, IS03, IS04, IS06
Objetivos
<p>Ob1. Conocer los principios que subyacen a las metodologías ágiles y estudiar las metodologías de desarrollo más conocidas.</p> <p>Ob2. Entender cómo se organizan, de manera eficiente, los equipos de desarrollo y cuál es el papel de las metodologías de desarrollo dentro de los procesos de la Ingeniería del Software.</p> <p>Ob3. Conseguir que el estudiante adquiera las capacidades necesarias para implantar una metodología dentro de una organización.</p>

Figura. 2. Ejemplo de la herramienta actual – Plan enseñanza

En la figura 1 y 2 se muestra que todos los datos deben ser introducidos de forma manual. Por ejemplo, las competencias introducidas en el apartado de “Competencias que tienen asignadas” pueden ser diferentes a las especificadas en otros apartados. Esta situación genera errores en el proyecto docente que deben ser detectados y solucionados a posteriori. El *Plan aprendizaje*, *Plan tutorial* y *bibliografía*, ocurren situaciones similares con otros datos.

<b>Diferencias entre la herramienta actual y la aplicación desarrollada</b>	
<b>Herramienta actual</b>	<b>Aplicación web</b>
Algunos los datos generales de cada asignatura se introducen de forma manual	Los datos generales de cada asignatura son cargadas directamente desde la base de datos
Hay que introducir los datos de cada asignatura para completar los planes: <i>plan enseñanza, plan aprendizaje, plan tutorial, bibliografía</i>	Existen tablas predefinidas para cada componente con ventanas emergentes. El usuario solamente elige las casillas de selección a su necesidad
El usuario tiene que introducir los códigos de las competencias asignadas a cada asignatura	La lista de competencias es cargada directamente desde la base de datos. El usuario seleccione la competencia.
No se puede importar datos de otras asignaturas	Importación de datos de otras asignaturas
El usuario debe introducir todos los requisitos previos respecto a una asignatura	El usuario solamente debe seleccionar los requisitos previos respecto a una asignatura. La lista es automatizada con respecto al semestre y curso en que se encuentra la asignatura

## **3. Objetivos**

### **3.1 Objetivos académicos**

Uno de los objetivos principales de este trabajo es que el estudiante tenga una experiencia mayor en la realización, diseño y desarrollo de una aplicación. Enfocado en un entorno más profesional, utilizando las herramientas que se usa en el día a día para una mejor implementación con calidad. También cabe destacar la preparación para un entorno ágil de desarrollo y aprendizaje.

Saber encontrar puntos débiles de código, capacidad para resolver alguna tarea durante el desarrollo es otro de los objetivos a alcanzar. La idea es que el alumno no esté bloqueado con una tarea, y que puede avanzar en diferentes puntos del proyecto maximizando así el tiempo de desarrollo. Este proyecto ha seguido el principio *KISS (Keep It Simple Stupid)*. El principio de diseño KISS establece que la mayoría de los sistemas funcionan mejor si se mantienen simples que si se hacen complejos. Por ello, hacer que el estudiante empiece a utilizar este método para un entorno clave del diseño en el futuro profesional.

### **3.2 Objetivos generales del proyecto**

Teniendo en cuenta las posibles dificultades que puede entrañar este proyecto, se plantea los siguientes objetivos:

- Aprendizaje, familiarización y uso de tecnologías web e Ingeniería del Software.
- Definir los requerimientos del sistema, de los distintos perfiles de usuarios que harán uso de la aplicación diseñada, y de la base de datos y sus relaciones.
- Diseñar y desarrollar una interfaz gráfica de acceso y manejo de la información, que contemplará parámetros de usabilidad y accesibilidad.
- Definir un modelo de datos genérico con la capacidad suficiente para almacenar y gestionar la información.

- Elaboración de toda la documentación necesaria en cada una de las fases del proyecto de desarrollo software.

Desde el punto de vista del sistema de elaboración de los PDA's de la ULPGC propiamente dicho los objetivos serán:

- Confección del Proyecto Docente.
- Verificación de la información incorporada en el PDA.
- Consulta y exportación de los datos del PDA.

## 4. Competencias

- **IS01**

**Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos y prácticas de la ingeniería del software.**

Esta competencia se queda cubierta en los apartados de definición de los objetivos y la metodología necesaria para la realización del proyecto, siguiendo los métodos ágiles presentes en el mercado.

- **IS02**

**Capacidad para valorar las necesidades del cliente y especificar los requisitos software para satisfacer estas necesidades, reconciliando objetivos en conflicto mediante la búsqueda de compromisos aceptables dentro de las limitaciones derivadas del coste, del tiempo, de la existencia de sistemas ya desarrollados y de las propias organizaciones.**

Esta segunda competencia se abarca en la definición de la pila de productos que definen los requisitos del sistema y también con las revisiones con mi Tutor.

- **IS03**

**Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles.**

Esta competencia queda demostrada en la sección de configuración del proyecto entre *Firebase* y *Angular*. Al definir la integración usando el paquete *Angularfire*, que como el nombre indica, su uso es solo para proyectos desarrollados en *Angular*.

- **IS04**

**Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales.**

La siguiente competencia se queda reflejada en los apartados al fijar el diseño del trabajo, proyectando la vista para para cada menú, así como el diseño de la base datos y la planificación previa a la implementación.

- **TFG01**

**Ejercicio original para realizar individualmente y presentar y defender ante un tribunal universitario, consistente en un proyecto en el ámbito de las tecnologías específicas de la Ingeniería en Informática de naturaleza profesional en el que se sinteticen e integren las competencias adquiridas en las enseñanzas.**

Con la realización completa de este trabajo fin de grado se obtiene esta competencia, siguiendo todas las prácticas, patrones de diseño definidos para el desarrollo software.

## **5. Aportaciones**

La gran aportación de este proyecto es la facilidad, de disponer un sistema único y estándar para todos a la hora de gestionar la elaboración del proyecto docente de cada asignatura. Una interfaz simple y común, con funcionalidad para relacionar asignaturas similares, en el que el usuario sólo tendrá que seleccionar lo deseable.

La gran particularidad de este proyecto está en la perspectiva técnico-científico. Aporta una gran facilidad y mayor rendimiento a la hora de preparar y definir los proyectos docentes; permitiendo un menor desperdicio de esfuerzo de trabajo, modelado y unificación de aspectos de diseño del PDA.

## 6. Metodología

Es importante destacar que en el marco de gestión del proyecto se empleará un enfoque ágil en el desarrollo de la aplicación mediante una Kanban y una revisión constante.

### 6.1 ¿Dónde surgió el Kanban?

Es un método que es utilizado para la gestión del trabajo que surgió en la empresa de Toyota. En los años 40, fue implementado por Toyota en la producción, un sistema llamado “*Just In Time*” (Justo a tiempo). Es decir, la producción se basa en la petición de los clientes y no en la práctica tradicional de fabricar productos e intentar venderlos en el mercado abierto.

Su exclusivo sistema de producción puso las bases del *Lean Manufacturing* (producción ajustada). Su propósito fundamental consiste en minimizar los desperdicios sin afectar la producción. El objetivo principal es crear más valor para el cliente sin generar más gastos.

### 6.2 ¿Qué significa Kanban?

El método Kanban es un sistema que utiliza tarjetas de diferentes colores o tamaños para designar y especificar tareas. De esta forma se mejora la administración desde tarjetas de señalización para control de flujo. Por lo tanto, uno sabe qué tareas deben realizarse, se están realizando y cuáles se han completado. Suele utilizarse en empresas de producción para indicar cómo funcionan los flujos de trabajo.

La Tabla Kanban (Figura 2) más básica consta de tres columnas: "Por hacer", "En proceso" y "Hecho". Si se aplica correctamente, serviría como fuente de información ya que muestra dónde están los cuellos de botella del proceso y qué impide que el flujo de trabajo sea continuo e ininterrumpido.



Figura. 3. Ejemplo de una tabla de Kanban

## 6.3 Principios del Método Kanban

- **Empezar con lo que hace ahora**

No requiere configuración y se puede aplicar a flujos de trabajo reales o procesos activos para identificar problemas. Por tanto, es fácil implementar Kanban en cualquier tipo de organización, ya que no es necesario realizar cambios drásticos.

- **Buscar e implementar cambios incrementales y evolutivos**

Estos son pequeños y continuos cambios incrementales y evolutivos en el proceso actual. En general, no se consideran cambios radicales ya que suelen encontrar resistencias debido a la incertidumbre del proceso.

- **Responsabilidades y los cargos actuales**

Los puestos y responsabilidades existentes pueden tener valor y vale la pena mantenerlos. El método Kanban no prohíbe los cambios, pero tampoco los prescribe. Fomenta cambios incrementales ya que no da tanto miedo como para ralentizar el progreso.

- **Animar el liderazgo en todos los niveles**

Este es el principio más innovador de Kanban. Algunos de los dos principales líderes surgen de las actividades del día a día de las personas que están al frente de sus equipos. Es importante que todos fomentemos una mentalidad de melodía continua (Kaizen) para alcanzar o ejecutar un equipo / departamento / empresa fuera de nivel. No pueden ser una actividad de nivel gerencial.

- **Aplicación de los estados en el proyecto**

Como se ha mencionado, la tabla básica del Kanban son tres columnas (por hacer, en proceso, hecho). En esta aplicación se ha definido uno estado más para la verificación del estado del proyecto por parte del tutor:

<b>Estados</b>	<b>Definición</b>
Por hacer	Lista de las tareas necesarias para definir una funcionalidad del proyecto. En este estado todavía no se empezado el desarrollo.
En proceso	Tareas que se encuentran en desarrollo. Las tareas en este estado deben ser pocas, para no cargar mucho al desarrollador y poder centrarse más en una ejecución y ver si hay dependencias con otras tareas.
Revisión con el tutor	Revisión de cada tarea con el Tutor mediante tutorías online, enseñando la correcta funcionalidad de cada una. En caso del visto bueno de parte del Tutor la tarea pasa al estado Hecho, en caso de no, vuelve al primer estado con los comentarios de la revisión.
Hecho	Tareas que ya están terminadas y aprobadas por el Tutor. En este estado el desarrollador sigue con la siguiente tarea.

*Tabla 1. Tabla de los estados de tareas*

## **6.4 Pila de producto**

Todos los requisitos de la aplicación se encuentran definidas en la pila de producto. En el anexo I de esta memoria se ilustra la lista de todas las funcionalidades necesarias para la realización del proyecto.

## **6.5 Iteraciones**

El proyecto se ha seguido una secuencia de iteraciones y revisiones con el tutor para la correcta verificación de cada una de las funcionalidades definidas en la pila de producto. Al finalizar cada iteración algunas funcionalidades han sido refinadas, según la revisión realizada con el Tutor.

## 7. Recursos Utilizados

En este apartado se ilustrarán los diferentes recursos utilizados para la realización de este proyecto. Cabe destacar que todo el proyecto ha sido desarrollado en el sistema operativo Windows 10. A continuación se enumeran los distintos recursos:

- **HTML5**

HTML es el componente básico de la web, te permite insertar contenido y establecer la estructura básica de un sitio web. Por tanto, sirve para dar sentido y organizar la información en una página web. Sin esto, el navegador no podría mostrar texto como elementos o cargar imágenes y otro contenido.

Los hipertextos son conjuntos de elementos conectados. Pueden ser palabras, imágenes, videos, documentos, etc. Cuando están conectados, forman una red de información que permite la comunicación de datos, organizar el conocimiento y guardar información. [6]



Figura. 4. HTML5

- **CSS3**

CSS se denomina lenguaje de hojas de estilo en cascada (*Cascading Style Sheet*) y se utiliza para diseñar elementos escritos en un lenguaje de marcado como HTML. CSS separa el contenido de la representación visual del sitio web. Es posible cambiar el color del texto y el fondo, fuente y espaciado entre párrafos. También puede crear tablas, usar variaciones de diseño, ajustar imágenes para sus respectivas pantallas, etc. [7]



Figura. 5. CSS3

- **Brave**

Es un nuevo navegador web, con código abierto. Está basado en Chromium. Fue creado por la empresa Brave Software en el año 2016. El navegador también evita que aparezcan anuncios invasivos al acceder a redes sociales y sitios web. Este sistema de bloqueo evita el seguimiento de búsquedas, historial de navegación y otros datos utilizados para este tipo de anuncios.



*Figura. 6. Brave*

- **Visual Studio Code**

Visual Studio Code, o simplemente VSCode, es una herramienta multiplataforma ligera que está disponible para Windows, Mac OS y Linux y se adapta a una amplia gama de proyectos. Además, el editor tiene soporte para la sintaxis de varios lenguajes de programación y es totalmente libre.



*Figura. 7. Visual Studio Code*

- **NPM**

Npm es el Node Package Manager, desarrollado por el lenguaje JavaScript, utiliza varias herramientas para instalar librerías, módulos y administrar sus dependencias, con tan solo en una línea de código.



*Figura. 8. Node Package Manager*

- **BOOTSTRAP**

Bootstrap es el framework más utilizado para crear diseños y pantallas para sistemas web, paneles de control, sitios receptivos y sitios comunes también. Facilita mucho el trabajo de front-end. [1]



*Figura. 9. Bootstrap 4*

- **JavaScript**

Es un lenguaje de programación utilizado principalmente en páginas web. Con JavaScript puedes mostrar mensajes y otra información interesante, hacer comprobaciones o cambiar dinámicamente la presentación visual de las páginas, según el comportamiento que quieras que tenga tu página. Todo a través del navegador del usuario. Es la tercera capa del pastel de las tecnologías web estándar, dos de las cuales (HTML y CSS).



*Figura. 10. JavaScript*

- **Firebase**

*Google Firebase* es una plataforma digital utilizada para facilitar el desarrollo de aplicaciones web o móviles, de forma eficaz, rápida y sencilla. Gracias a sus diversas funciones, se utiliza como técnica de Marketing Digital, con el fin de incrementar la base de usuarios y generar mayores beneficios económicos. Su principal objetivo es mejorar el rendimiento de las aplicaciones implementando varias funcionalidades que harán de la aplicación una herramienta mucho más flexible, segura y de fácil acceso para los usuarios.

## ○ **Funciones de Firebase**

*Firebase* proporciona diversas funciones para el desarrollo de aplicaciones. Sin embargo, para la implementación de nuestro proyecto hemos centrado más en dos, de las distintas funcionalidades que nos ofrece:

### 1. **Autenticación de usuarios**

Se ejecuta para identificar a los usuarios a través de correo electrónico o redes sociales. *Firebase* puede guardar en la nube los datos de inicio de sesión con total seguridad, evitando que una persona tenga que identificarse cada vez que abra la aplicación.

### 2. **Cloud Firestore**

*Cloud Firestore* es una base de datos NoSQL, flexible y escalable para el desarrollo centrada en dispositivos móviles, la Web y servidores de *Firebase* y *Google Cloud*. Mantiene sus datos sincronizados en las aplicaciones cliente a través de escuchas en tiempo real. Además, ofrece soporte sin conexión para dispositivos móviles y web para que pueda crear aplicaciones receptivas que funcionen independientemente de la latencia de la red o la conectividad a Internet. [4]



*Figura. 11. Firebase*

## ● **Angular**

Angular es un framework de código abierto utilizado para crear aplicaciones web basadas en una sola página dinámica. Los desarrolladores también utilizan esta herramienta para crear menús animados para sitios web basados en HTML. Implementa la arquitectura Model-View-Controller (MCV), que se utiliza en el desarrollo de aplicaciones para Internet.

También hay que destacar que es un framework bastante modular y escalable adaptándose a nuestras necesidades y al estar basado en el estándar de componentes web.

[2] [8]



Figura. 12. Angular

- **Typescript**

*Typescript* es un superset del lenguaje JavaScript creado por Microsoft para permitir la escritura de scripts utilizando escritura estática, orientación a objetos y facilitar la escritura de código con una sintaxis fácil de entender. Se dispone con varios recursos que mejor apoyan el uso de la programación orientada a objetos. [3]



Figura. 13. Typescript

- **AngularFire**

*AngularFire* es una librería utilizada para el *Angular* con el objetivo de suavizar y eliminar la complejidad que se puede encontrar al implementar el SDK de *Firebase*. Proporciona al desarrollador una experiencia más natural al cumplir con las convenciones de *Angular*. [5]



Figura. 14. Librería AngularFire

- **Angular CLI**

Es una línea de comandos que facilita la creación, gestión, ejecución, prueba, implementación. Herramienta muy potente a la hora de empezar a desarrollar, de forma rápida y sencilla, al realizar test, añadir nuevos componentes en el proyecto.



Figura. 15. Angular CLI

- **JQuery**

Es una biblioteca de *JavaScript* que se puede agregar a proyectos de codificación. Básicamente, *jQuery* permite a los desarrolladores web conectar recursos *JavaScript* rutinarios a una página web para que puedan dedicar más tiempo a centrarse en funciones complicadas que son exclusivas del sitio.



Figura. 16. JQuery

## 8. Planificación

En la tabla siguiente se muestra, por fases, la realización del trabajo, la duración estimada y real de las mismas.

Fases	Duración Estimada (horas)	Duración Real (horas)	Actividades
<b>Creación del Producto Backlog</b>	20	30	Análisis de la situación actual (10h)
			Primera toma de requisitos de usuario y priorización de necesidades (10h)
<b>Sprint Zero</b>	40	64	Definición de requisitos del software (10h)
			Diseño de la base de datos en <i>Firebase</i> (15h)
			Diseño de la interfaz de usuario - Mockups (15h)
<b>Desarrollo de las iteraciones</b>	200	250	<p>Se estime el desarrollo de tres versiones que sucederán fases de implementación de la interacción con la base de datos, implementación iterativa de la interfaz de usuario según las necesidades que se estimen. A finales de cada iteración, se hará una revisión de las funcionalidades con el tutor.</p> <p>La estimación de puesta a producción es la siguiente:</p> <ul style="list-style-type: none"> <li>• iteración 1 (80h)</li> <li>• iteración 2 (60h)</li> <li>• iteración 3 (60h)</li> </ul>
<b>Documentación / Presentación</b>	40	90	Documentación de la aplicación desarrollada
			Edición de la memoria del proyecto realizado
			Preparación de la defensa pública

Tabla 2. Tabla de planificación

En la tabla 2 se indica la duración en horas estimadas y reales para la realización del TFG. Como se puede observar en algunas tareas se ha excedido el tiempo estimado. Esto es debido al aprendizaje de las distintas tecnologías utilizadas a la hora de llevar a cabo la implementación.

## 9. Desarrollo de la Aplicación

En este apartado se explica todos los pasos seguidos para la implementación del proyecto. El proyecto docente está dividido en cinco bloques o apartados, que forman los datos completos de cada asignatura: **Datos Generales, Plan de Enseñanza, Plan de Aprendizaje, Plan Tutorial y Bibliografía.**

Inicialmente se hizo un diseño del mockup de cada apartado. Cabe mencionar que el mockup fue preparado con la ayuda de mi tutor. Posteriormente, se hizo la configuración y diseño de la base de datos en *Firebase console*, basándose en la estructura y campos de datos de cada apartado. Con la configuración hecha en *Firebase* se conectó la base de datos a nuestro proyecto en *Angular*. En este punto, se puso en práctica el desarrollo de la interfaz del diseño reflejado en los mockups.

En segundo lugar, se llevó a cabo la implementación de la funcionalidad de la aplicación. Cada dos semanas se hizo una revisión con el tutor para la validación de las funcionalidades hechas y definir la siguiente iteración. Se hicieron tres iteraciones y un *End to End test (E2E)* de toda la aplicación.

### 9.1 Pila de Producto

En la tabla 3 se expone la pila de producto simplificada de la funcionalidad implementada en la aplicación. En el anexo I se puede encontrar la tabla completa de la pila de producto con la información detallada.

<b>ID</b>	<b>Nombre</b>	<b>Descripción</b>
HU-1	Login	Acceso al sistema a través de correo electrónico y contraseña
HU-2	Listado asignaturas	Pantalla inicial con la lista de todas las asignaturas disponibles en el Grado
HU-3	Editar Asignatura [Datos generales]	Editar los datos generales que definen la asignatura
HU-4	Visualizar Asignatura [Datos generales]	Ver los datos generales que definen la asignatura. El usuario no tiene permiso para editar

HU-5	Editar Asignatura [Plan de enseñanza]	Editar los datos plan enseñanza que definen la asignatura
HU-6	Visualizar Asignatura [Plan de enseñanza]	Ver los datos plan enseñanza que definen la asignatura. El usuario no tiene permiso para editar
HU-7	Editar Asignatura [Plan de aprendizaje]	Editar los datos plan de aprendizaje que definen la asignatura
HU-8	Visualizar Asignatura [Plan de aprendizaje]	Ver los datos plan de aprendizaje que definen la asignatura. El usuario no tiene permiso para editar
HU-10	Editar Asignatura [Plan tutorial]	Editar los datos plan tutorial que definen la asignatura
HU-11	Visualizar Asignatura [Plan tutorial]	Ver los datos plan tutorial que definen la asignatura. El usuario no tiene permiso para editar
HU-12	Editar Asignatura [Bibliografía]	Editar los datos bibliográficos que definen la asignatura
HU-13	Visualizar Asignatura [Bibliografía]	Ver los datos bibliográficos que definen la asignatura. El usuario no tiene permiso para editar
HU-14	Logout	Cierre del acceso a la aplicación. El logout es hecho desde el Firebase
HU-15	Importar datos asignaturas	Importar datos de otras asignaturas en caso de que las asignaturas comparten datos similares
HU-16	Borrar datos	Se borra un elemento en la base datos. Se sale un dialogo de confirmación al usuario

Tabla 3. Tabla de Pila producto

## 9.2. Sprint Zero

En el **Sprint Zero** el objetivo fue preparar todos los recursos necesarios para iniciar el desarrollo de la aplicación. Las distintas tareas a realizar están: la plataforma para la sintaxis de edición de proyecto, instalación de librerías y frameworks requeridos, la configuración del proyecto en la consola de *Firebase* de Google, así como la configuración en el entorno local del proyecto para la correcta conexión con los servidores de *Firebase*.

### 9.2.1 Crear un proyecto en *Firebase* Google

Para iniciar nuestro TFG con la base de datos se define crea un proyecto en la consola de *Google Firebase*. Para crear el proyecto es necesario tener previamente una cuenta valida en Google. Se ha seguido los siguientes pasos.

<https://console.firebase.google.com/?hl=es-419>

#### 1- Dirigir al enlace para crear el proyecto

Página web oficial de Google para la gestión y exploración de proyectos hechos en *Firebase*. Funciona como un front-end de los proyectos definidos en la cuenta.

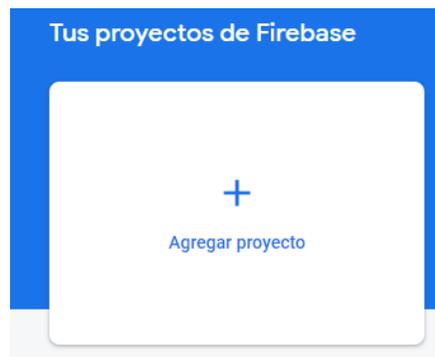


Figura. 17. Ejemplo crear un nuevo proyecto *Firebase*

#### 2 - Accedido al enlace de arriba, hacer *click* en **Agregar proyecto**

En la figura 17 se ilustra la captura de pantalla desde el enlace de *Firebase console* para definir un proyecto nuevo.



Figura. 18. Ejemplo de nombre del proyecto

## 2- Definir el nombre del proyecto y hacer clic en Continuar

Como se puede ver en la figura 18 se define un nombre al proyecto para identificarlo de forma global.



Figura. 19. Ejemplo de carga del proyecto

## 3- Se define todos los parámetros para la creación del proyecto

En esta etapa el proceso empieza a preparar todos los recursos y configuración necesarios para el ambiente de trabajo, como se aprecia en la figura 19.



Figura. 20. Ejemplo de la vista general del proyecto

#### 4- Al terminar el proceso de creación, se presenta la vista general de nuestro proyecto en la consola

En la figura 20 se puede ver ya el proyecto listo con las descripciones generales. El usuario ya puede definir la estructura de base de datos. Además de definir la base datos, se tiene a disposición todas las funcionalidades que ofrece *Firebase* para realizar una aplicación ya sea para una página web o aplicación móvil.

### 9.2.2 Crear la aplicación base en Angular

Con posterioridad se creó la base de nuestra aplicación en Angular usando en editor *Visual Studio Code*. Como se ha referido anteriormente, se hace uso del angular CLI para una implementación rápida y más flexible a la hora de generar y gestionar los componentes. El primer paso a ser instalar el CLI de angular.

#### Requisitos previos:

Instalado el Node.js - <https://nodejs.org/es/download/>

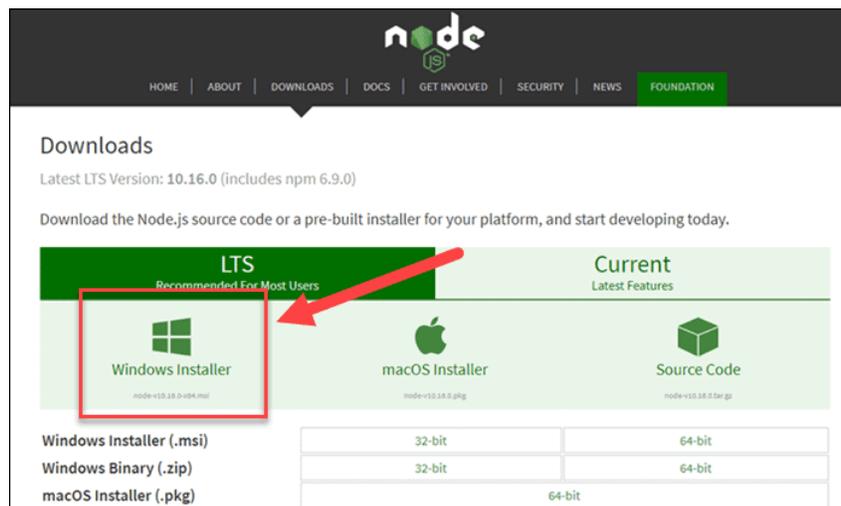


Figura. 21. Ejemplo de la descarga Node.js

Después de acceder al enlace arriba mencionado, se abre la pantalla para descargar el Node.js como se ve en la Figura 21. Para este trabajo se hizo la descarga para la plataforma Windows.

## Verificación de la instalación

```
$ node -v
v14.15.0

Ivan@DESKTOP-V067437 MINGW64 /g/bootstrap_sandbox
$ npm -v
6.14.8
```

Figura. 22. Verificación de la instalación

En Figura 22, se abre el terminal de línea de comandos (nuestro caso en *Visual Studio Code*) y se ejecuta los comandos `node -v` y `npm -v`. Como se observa el Node.js y el paquete NPM fueron instalados correctamente.

## Instalación de Angular CLI

```
$ npm install -g @angular/cli
```

Figura. 23. Comando para instalar paquete Angular CLI

Con NPM instalado, se ejecuta el comando para iniciar la instalación. En este punto, se instala todos los módulos básicos de Angular. En la Figura 23 se puede ver la facilidad de la librería Angular CLI, con tan solo un comando él se encarga de instalar los paquetes necesarios.

```
$ ng new APP
```

Figura. 24. Comando para crear nuevo proyecto

Al ejecutar el comando de la Figura 24 se crea un nuevo proyecto con la estructura inicial de carpetas y los módulos “core” de Angular.

```
$ ng serve
```

Figura. 25. Arrancar Servidor

A partir de este momento (Figura 25) la aplicación se está ejecutando en el ordenador local. El servidor reconstruye automáticamente la aplicación y vuelve a cargar la página cuando cambia cualquiera de los archivos de origen. La página es alojada en <http://localhost:4200/>.

Toda la subestructura de carpetas y ficheros es creada bajo el primer nodo **APP**, con un sólo archivo de configuración del espacio de trabajo.

### 9.2.3 Configurar el proyecto Firebase en Angular

Una gran facilidad que demuestra *Firebase* es que es muy rápido y sencillo a la hora de conectar la APP de angular con el servidor de la consola Firebase.

Se sigue los siguientes pasos:

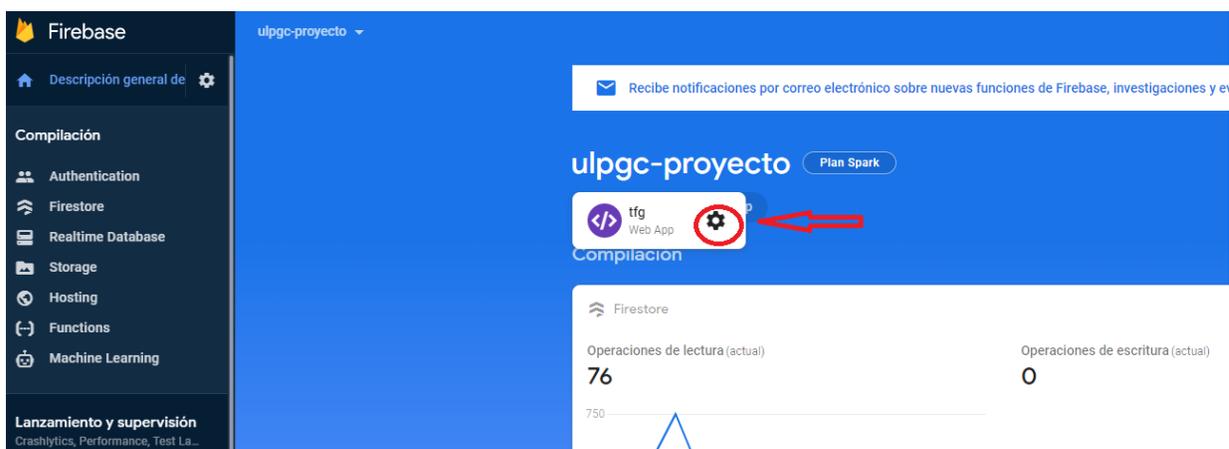


Figura. 26. Ejemplo de configuración del proyecto en Firebase

#### 1- Hacer clic en el icono de configurar de la aplicación

En este paso el usuario accede a toda la configuración del proyecto. En la figura 26 se puede ver el icono para acceder a las definiciones, como, el nombre proyecto, el ID para identificar el proyecto. Por ejemplo, si se usa en un API, número de identificación global, Identificación de nuestra, y entre otros.

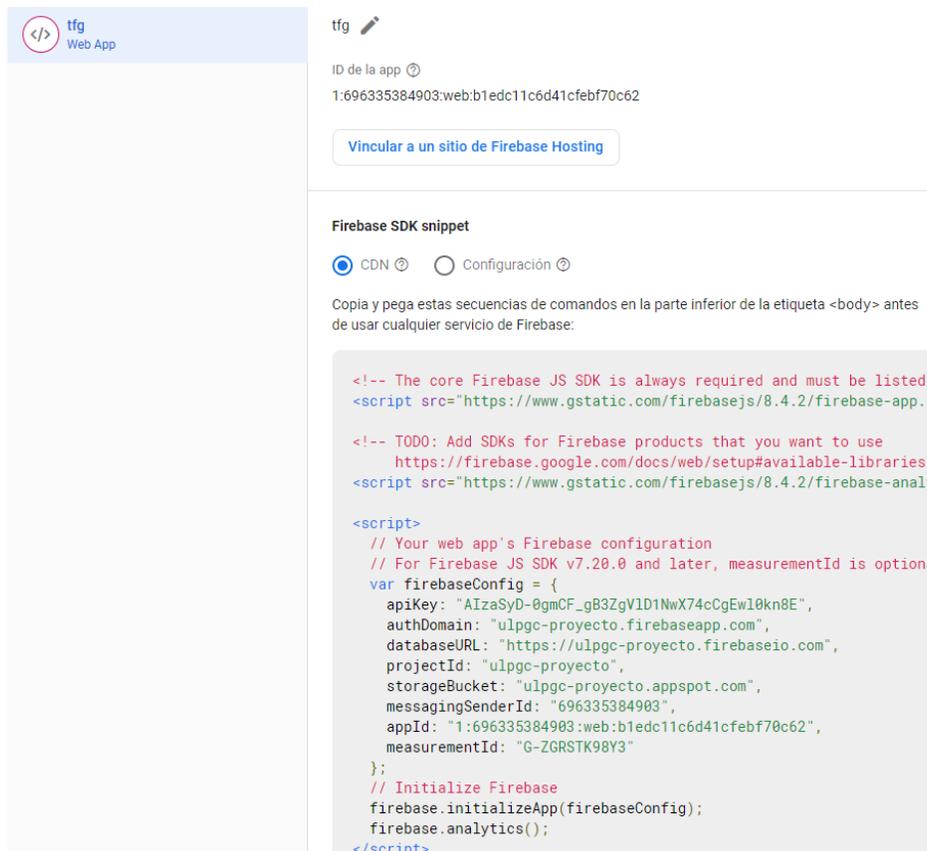


Figura. 27. Configuración de la APP en Firebase

## 2- Se abre una ventana con información sobre la configuración

Lo único que hay que hacer es copiar el código y adjúntalo en Angular. Con la configuración ilustrada en la Figura 27, se dispone de lo necesario para identificar el proyecto en Google y así poder hacer peticiones hacia el proyecto desde la aplicación local de Angular.

```

4   export const environment = {
5     production: false,
6     firebase: {
7       apiKey: "AIzaSyD-0gmCF_gB3ZgVlD1NwX74cCgEwl0kn8E",
8       authDomain: "ulpgc-proyecto.firebaseio.com",
9       databaseURL: "https://ulpgc-proyecto.firebaseio.com",
10      projectId: "ulpgc-proyecto",
11      storageBucket: "ulpgc-proyecto.appspot.com",
12      messagingSenderId: "696335384903",
13      appId: "1:696335384903:web:b1edc11c6d41cfebf70c62",
14      measurementId: "G-ZGRSTK98Y3"
15    }
16  };

```

Figura. 28. Código del Firebase en Angular

### 3- Al extraer el código de configuración de *Firebase* se pega en la ruta de carpeta *src/environments/environments.ts* dentro de la aplicación Angular

Al realizar todos los pasos listados previamente, la aplicación en Angular ya está configurada y lista para uso conjunto con *Firebase*.

#### 9.2.4 Estructura de la base datos en Firebase

La estructura y el modelo de base de datos hace uso de la función *Cloud Firestore* en *Firebase*. *Cloud Firestore* es una base de datos NoSQL orientada a documentos. Al contrario de una base de datos en SQL no define tablas ni filas. A cambio, los datos están almacenados en documentos, que se encuentra organizados en colecciones.

La unidad de almacenamiento es el **documento**. Un documento es un registro ligero que contiene campos, que se asignan a valores. Cada documento está identificado por un nombre. Las colecciones son simplemente un conjunto de varios documentos almacenados. Dado nuestra aplicación se define una colección padre llamada **asignaturas** formado por un conjunto de documentos que viene a representar todas las asignaturas del *Grado en Ingeniería Informática*.

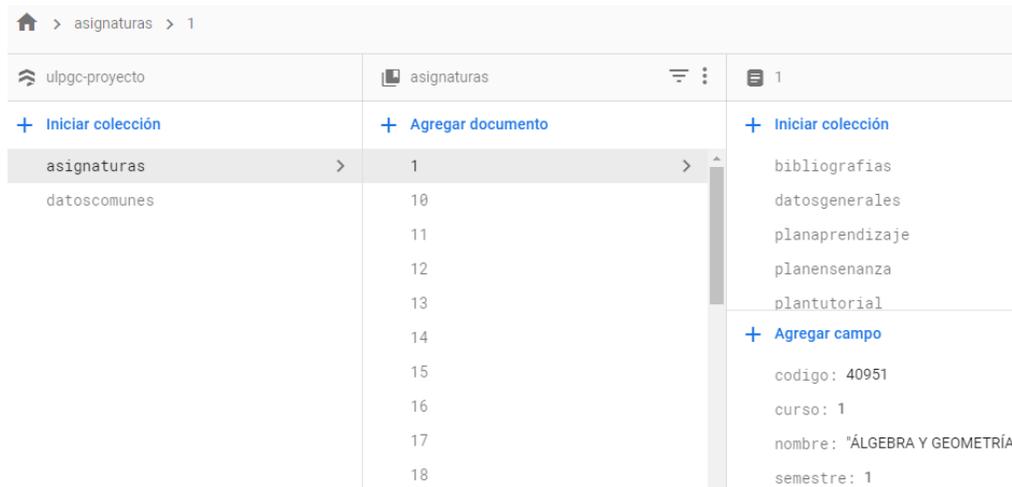


Figura. 29. Estructura de la base datos en Firebase

En *Firestore* también se puede definir datos jerárquicos. Por ejemplo, definiendo colecciones de documentos y dentro de cada documento definir otras subcolecciones. En la Figura 29, se puede ver la colección **asignaturas** (padre) compuesto por varios documentos que viene a representar las asignaturas.

En la aplicación se ha creado como subcolecciones los menús (dentro cada documento asignatura) **datos generales**, **plan de enseñanza**, **plan de aprendizaje**, **plan tutorial**, **bibliografía**.

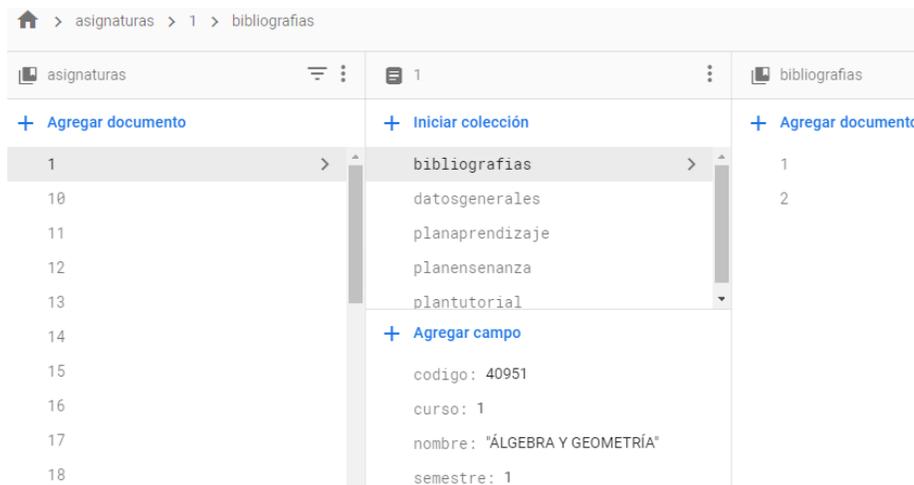


Figura. 30. Estructura de la base datos (subcolecciones)

En la Figura 30 se define para cada documento, los menús de cada asignatura. Así que para acceder a los menús el usuario tiene que elegir una asignatura en primer plan y en seguida poder leer los datos de dicha asignatura.

## 9.2.5 Componentes

La aplicación está hecha en *Angular*, siguiendo las pautas de arquitectura definidas en *Angular*. Para eso, existe un fichero padre **app.module.ts** de la toda aplicación, donde se define todos los componentes, servicios, rutas que se hace uso dentro de la aplicación. En total se ha definido 24 subcomponentes para el proyecto.

```
// My components
import { AppComponent } from './app.component';
import { HeadersComponent } from './components/headers/headers.component';
import { DatosGeneralesComponent } from './components/datos-generales/datos-generales.component';
import { PlanEnsenanzaComponent } from './components/plan-ensenanza/plan-ensenanza.component';
import { AddCompetenciasComponent } from './components/plan-ensenanza/add-competencias/add-competencias.component';
import { AddObjetivosComponent } from './components/plan-ensenanza/add-objetivos/add-objetivos.component';
import { AddEvaluacionComponent } from './components/plan-ensenanza/add-evaluacion/add-evaluacion.component';
import { PlanAprendizajeComponent } from './components/plan-aprendizaje/plan-aprendizaje.component';
import { AddTaresComponent } from './components/plan-aprendizaje/add-tares/add-tares.component';
import { SignInComponent } from './components/authentication/sign-in/sign-in.component';
import { AsignaturasComponent } from './components/asignaturas/asignaturas.component';
import { SignUpComponent } from './components/authentication/sign-up/sign-up.component';
import { AddRequisitosComponent } from './components/datos-generales/add-requisitos/add-requisitos.component';
import { AddRecursosComponent } from './components/plan-aprendizaje/add-recursos/add-recursos.component';
import { AddResultadosComponent } from './components/plan-aprendizaje/add-resultados/add-resultados.component';
import { PlanTutorialComponent } from './components/plan-tutorial/plan-tutorial.component';
import { AtencionIndividualComponent } from './components/plan-tutorial/atencion-individual/atencion-individual.component';
import { AtencionGruposComponent } from './components/plan-tutorial/atencion-grupos/atencion-grupos.component';
import { AtencionTelefonciaComponent } from './components/plan-tutorial/atencion-telefoncia/atencion-telefoncia.component';
import { AtencionVirtualComponent } from './components/plan-tutorial/atencion-virtual/atencion-virtual.component';
import { BibliografiaComponent } from './components/bibliografia/bibliografia.component';
import { AddBibliografiaComponent } from './components/bibliografia/add-bibliografia/add-bibliografia.component';
import { ImportAsignaturaComponent } from './components/asignaturas/import-asignatura/import-asignatura.component';
import { SharedDeleteComponent } from './components/shared-delete/shared-delete.component';
```

Figura. 31. Lista de Componentes

En la Figura 31, se muestra la lista completa de todos los componentes necesarios para ejecutar la aplicación. En los componentes se define los servicios, la vista a representar, la interfaz del modelo, siguiendo el patrón de *Angular Modelo-Vista-Controlador*. Así que un componente puede comunicar con otro componente, compartir datos, estar a la escucha de datos provenientes de otros componentes. También existe la jerarquía entre componentes por lo que se puede establecer componentes padres y subcomponentes (componentes hijos).

## 9.2.6 Ficheros comunes de los componentes

Para definir un componente hace falta los ficheros bases: **component.css**, **component.html**, **component.ts**.

- Component.css – fichero para definir los códigos en CSS respecto a un componente.
- Component.html – fichero para definir los códigos HTML respecto a un componente.
- Component.ts – fichero para fijar los códigos utilizados a concretar las funcionalidades de cada componente. Se hace uso del lenguaje Typescript.

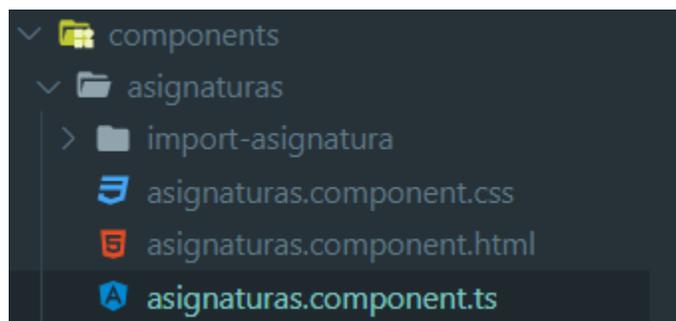


Figura. 32. Ejemplo de estructura de carpetas

Figura 32: Estos ficheros componen la base inicial de los componentes. Todavía se puede exponer otros ficheros dependiendo de la función de cada, por ejemplo:

- **Service.ts** – definir los servicios encargados de cada componente.
- **Model.ts** – Modelo a definir para el uso de datos de cada componente.
- **Guards.ts** – Actúa como una protección al intentar acceder en un determinado componente. El fichero Guards.ts evalúa si el usuario cumple los requisitos a acceder en el componente.

## 9.3. Sprint 1

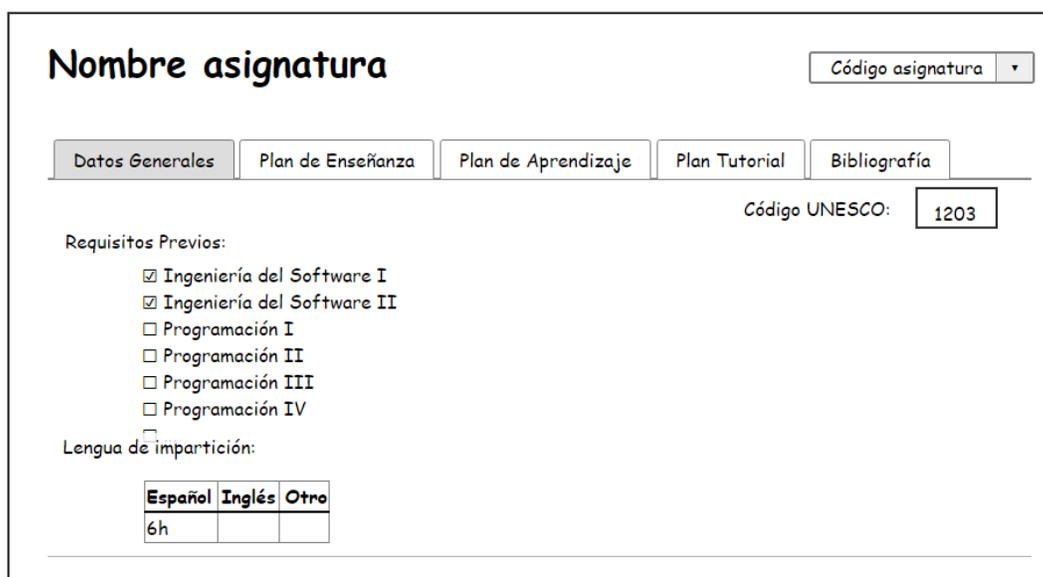
En el Sprint 1 se inició el desarrollo de la funcionalidad descrita en la pila de producto. En este sprint se abordaron las siguientes historias de usuario que requirieron la implementación de las vistas, modelos de interfaces de base datos y el código de la lógica de negocio.

ID	Nombre
HU-3	Editar Asignatura [Datos generales]
HU-5	Editar Asignatura [Plan de enseñanza]
HU-7	Editar Asignatura [Plan de aprendizaje]

A continuación, se describe el desarrollo de las historias de usuario realizadas en este sprint. Seguidamente se mostrará los mockups de cada historia de usuario, como modo de diseño previo de la interfaz gráfica, así como las capturas de pantallas de la aplicación.

### 9.3.1 HU-5 - Datos Generales

Como ejercicio inicial se hizo un diseño previo de cómo sería la vista de los datos generales de cada asignatura (Figura 32).



Nombre asignatura Código asignatura ▼

Datos Generales | Plan de Enseñanza | Plan de Aprendizaje | Plan Tutorial | Bibliografía

Código UNESCO: 1203

Requisitos Previos:

- Ingeniería del Software I
- Ingeniería del Software II
- Programación I
- Programación II
- Programación III
- Programación IV

Lengua de impartición:

Español	Inglés	Otro
6h		

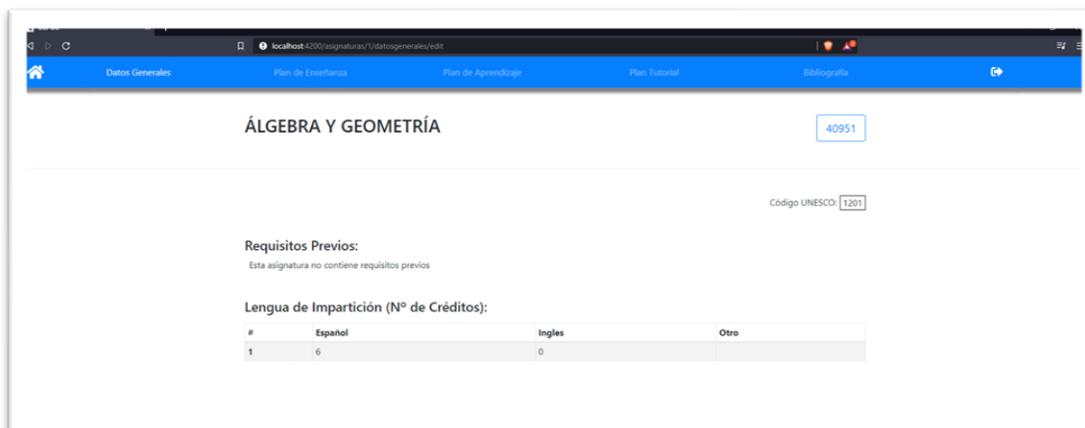
Figura. 33. Mockup de Datos generales

Código fuente responsable para filtrar las asignaturas de los requisitos previos (Figura 33). El método coge el ID de la asignatura en que se encuentra el usuario y hace un filtro de todas las asignaturas de la base datos. El mecanismo general de la función es que una asignatura X (ID de la asignatura) del curso X tiene como conocimiento previo todas las asignaturas de cursos y semestres anteriores de acuerdo con el plan de estudios oficial de la ULPGC.

```
ngAfterViewInit() {
  this.LoadDatos();
  this.reqSubscription = this.asignaturaService.asignaturasChanged.subscribe(asignaturas => { // filter requisitos
    const id = this.route.snapshot.params['id'];
    var curso: number;
    var semestre: number;
    for (let asignatura of asignaturas) {
      if (asignatura.id === id) {
        curso = asignatura.curso;
        semestre = asignatura.semestre;
        break;
      }
    }
    //2 < 3 e 2 < 1
    asignaturas.filter(item => ((item.curso === curso && item.semestre < semestre) || (item.curso < curso)))
      .map(ar => this.defaultRequisitos.push({nombre: ar.nombre, checked: false}));
  });
}
```

Figura. 34. Código del filtro de requisitos previos

Vista de los datos generales implementada. Se ve que la implementación de la vista se basa en el diseño del mockup inicial.



### 9.3.2 HU-7 - Plan de Enseñanza

En las figuras 34 y 35 se demuestra el diseño del apartado Plan enseñanza. En este menú se implementa todos los campos requeridos para el plan de enseñanza.

Figura. 35. Mockup de Plan enseñanza (1)

Figura. 36. Ventana emergente para añadir competencias

El código recorre el vector con las competencias y verifica las casillas que estén seleccionadas y las que no estén (Figura 36). Según la selección del usuario guarda la información en la base de datos y se ilustra las seleccionadas en la vista.

```

this.datosPlanEnsenanza.checkedValues.length = 0;
this.datosPlanEnsenanza.descriptions.length = 0;
this.datosPlanEnsenanza.checkedList.length = this.competencias.competencias.length;
for (var i = 0; i < this.datosPlanEnsenanza.checkedList.length; i++) {
    if (this.datosPlanEnsenanza.checkedList[i] === undefined) this.datosPlanEnsenanza.checkedList[i] = false;
    if (this.datosPlanEnsenanza.checkedList[i]) {
        this.datosPlanEnsenanza.checkedValues.push(this.competencias.competencias[i].id);
        this.datosPlanEnsenanza.descriptions.push(this.competencias.competencias[i].descripcion);
    }
}
this.pEService.updateDatos(this.datosPlanEnsenanza);
}

```

Figura. 37. Código fuente para guardar las competencias asignadas

Figura 37: El código recorre el vector con las competencias inicialmente asignadas en el menú plan enseñanza (previamente guardadas en la base de datos). Toma sólo valores que estén seleccionadas y se los añade en un segundo vector que por su vez se ilustra en la vista.

```
this.subscriptionPE = this.pEService.datosChanged.subscribe(datos => {
  if (!datos) datos = {}; // In case data base is empty
  if (!datos.checkedValues) datos.checkedValues = []; // If data base is empty and not initiate
  if (!datos.descriptions) datos.descriptions = [];
  this.lengthArr = this.datosFuentes.competenciaschecked.length;
  if (this.lengthArr !== datos.checkedValues.length) {
    for (let i = 0; i < datos.checkedValues.length; i++) {
      for (let k = 0; k < this.datosFuentes.competenciasIds.length; k++) {
        if (datos.checkedValues[i] === this.datosFuentes.competenciasIds[k]) {
          this.datosFuentes.competenciaschecked[i] = true;
        } else { this.datosFuentes.competenciaschecked[i] = false }
      }
    }
  }
  this.competenciasComunes.competencias.length = 0;
  for (let i = 0; i < datos.checkedValues.length; i++) {
    this.competenciasComunes.competencias.push({id: datos.checkedValues[i], descripcion: datos.descriptions[i]})
  }
});
this.pEService.fetchDatosPlanEnseanza();
```

Figura. 38. Código de filtro de las competencias para los criterios y fuentes de evaluación

Vista del plan de enseñanza en relación al mockup inicial. La vista será explicada detalladamente en el manual de usuario.



### 9.3.3 HU-9 - Plan de Aprendizaje

Igual que en el menú “Plan enseñanza”, en las figuras 38 y 39, se demuestra el diseño del apartado “Plan aprendizaje”. En este menú se implementa todos los campos requeridos para el plan de aprendizaje.

ID	Tipo	Contexto
Ta1	Reuniones de grupo	(P)
Ta2	...	...
Ta3	...	...

Figura. 39. Mockup del Plan de aprendizaje (1)

Tareas y actividades:

Tipo:

Reuniones de grupo

...

Otros: campo otros ...

Contexto:

(C): Científico

(P): Profesional

(I): Institucional

(S): Social

OK Cancel

Figura. 40. Ventana emergente para añadir tareas y actividades

El código a destacar en la vista del plan aprendizaje es el ilustrado a continuación. Simplemente existe un único método en el componente principal encargado de cargar las tablas responsables de los subcomponentes.

```
ngOnInit() {  
  this.loadTareasActividades();  
  this.loadDatosPlanAprendizaje();  
  this.loadRecursos();  
  this.loadResultados();  
}
```

Vista del plan de aprendizaje en relación al mockup inicial. La vista será explicada detalladamente en el manual de usuario.

The screenshot shows a web application interface for 'ÁLGEBRA Y GEOMETRÍA'. The top navigation bar includes 'Datos Generales', 'Plan de Enseñanza', 'Plan de Aprendizaje', 'Plan Tutorial', and 'Bibliografía'. The main content area displays the course title and a box with the number '40951'. Below this, there is a section titled 'Tareas y actividades que realizará según distintos contextos profesionales' with a '+ ?' icon. Underneath, a table lists three tasks with their IDs, types, contexts, and edit/delete actions.

ID	Tipo	Contexto	Editar	Borrar
TA1	Consulta de material bibliográficos	(C), (P)		
TA2	Asistir y participar en las actividades docentes	(P), (I), (S)		
TA3	Realización de ejercicios de autoevaluación	(C), (P)		

## 9.4. Sprint 2

Finalizado el **Sprint 1**, se pasa al siguiente sprint, ajustando la pila de producto. La experiencia del Sprint 1 fue un poco exigente, para organizar el diseño con la implementación utilizando las tecnologías necesarias. Requirió un poco de investigación pero que al final se ha logrado obtener el resultado esperado.

Se define la pila de sprint desarrollada durante el sprint:

ID	Nombre
HU-10	Editar Asignatura [Plan tutorial]
HU-12	Editar Asignatura [Bibliografía]
HU-16	Borrar datos
HU-2	Listado asignaturas
HU-15	Importar datos asignaturas

### 9.4.1 HU-11 - Plan Tutorial

El diseño del apartado “*Plan tutorial*” (Figuras 40 y 41). En este menú se implementa todos los campos requeridos para el plan de tutorial. Para el menú “*Plan tutorial*”, los campos son muy sencillos. Se compone simplemente de ventanas emergentes disponibles al usuario con opciones de casillas para seleccionar.

The image shows a web form titled "Nombre asignatura". At the top right, there is a dropdown menu labeled "Código asignatura". Below the title, there are five tabs: "Datos Generales", "Plan de Enseñanza", "Plan de Aprendizaje", "Plan Tutorial" (which is highlighted), and "Bibliografía". Under the "Plan Tutorial" tab, there is a section labeled "Atención presencial individualizada:" followed by a text input field containing the placeholder text "Horario de tutoría, cita previa del alumno, ...". To the right of this input field is a small square button with a plus sign (+). A yellow callout box with a pointer to the plus button contains the text "se abre la ventana cuando se pulsa el botón".

Figura. 41. Mockup del Plan tutorial (1)

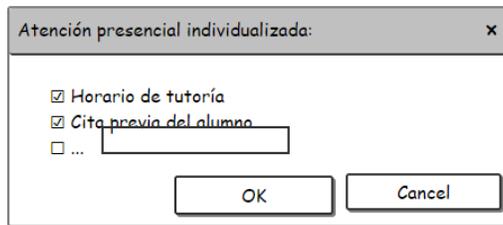


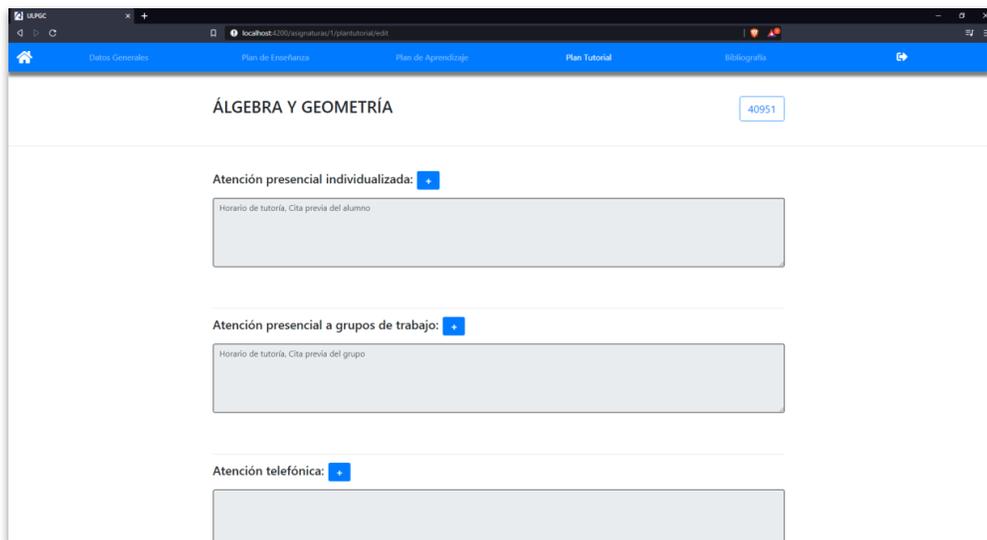
Figura. 42. Ventana emergente para añadir una atención presencial individual

El fichero *plan-tutorial.service.ts* se encarga de guardar los datos en la base datos de *Firebase*, con las casillas seleccionadas por el usuario.

El servicio del componente Plan tutorial carga los subcomponentes en la pantalla inicial. Existe 4 componentes para para la atención individual, grupos, telefónica y virtual. El código carga los datos desde la base de datos y, en caso que exista datos en la base de datos, los componentes son inicializados a vacío.

```
loadDatosPlanTutorial() {
  this.subscription = this.planTService.datosChanged.subscribe(datos => {
    this.datosPlanTutorial = datos;
    if(!this.datosPlanTutorial) {
      this.initEmptyData();
    }
    if(!this.datosPlanTutorial.atencionIndividualVal) this.datosPlanTutorial.atencionIndividualVal= [];
    if(!this.datosPlanTutorial.atencionGruposVal) this.datosPlanTutorial.atencionGruposVal= [];
    if(!this.datosPlanTutorial.atencionTelefonaVal) this.datosPlanTutorial.atencionTelefonaVal= [];
    if(!this.datosPlanTutorial.atencionVirtualVal) this.datosPlanTutorial.atencionVirtualVal= [];
  });
  (method) PlanTutorialService.fetchDatosPlantutorial(): void
  this.planTService.fetchDatosPlantutorial();
}
```

Vista del plan de tutorial en relación al mockup inicial. La vista será explicada detalladamente en el manual de usuario.



## 9.4.2 HU-13 – Bibliografía

El menú de “*Bibliografía*” contiene una única tabla en el que usuario puede añadir las bibliografías de cada asignatura. Los datos a incluir serían el tipo, título y el código ISBN del libro.

El fichero *bibliografía.service.ts* obtiene los datos introducidos por el usuario en la vista y simplemente envía los datos a la base de datos.

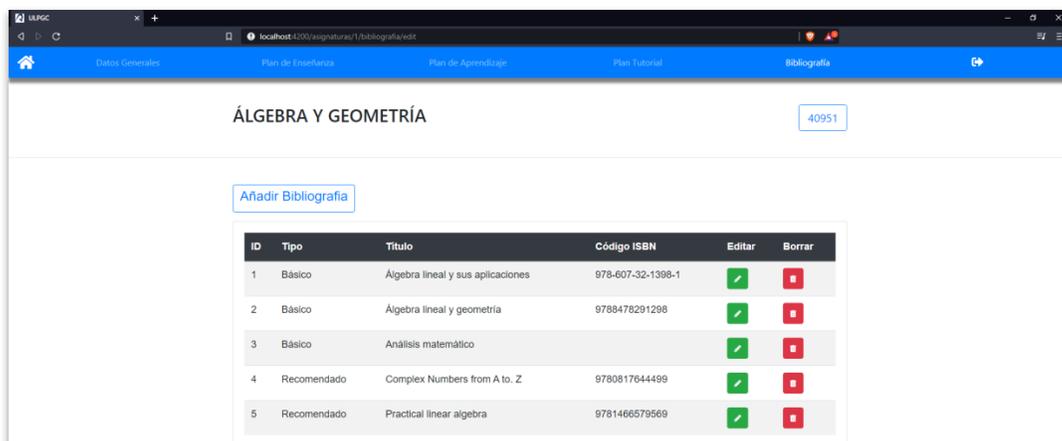
```
pattern="\d{13}" |  
[(ngModel)]="datosBibliografia.codigoIcbn">
```

Figura. 43. Patrón de verificación del ISBN

Dado que el ISBN puede contener 10 ó 13 dígitos se definió en la aplicación un patrón general de verificación en el que el campo debe tener máximo 13 dígitos. El componente de Bibliografía es el más simple ya que consta de una única tabla. El back-end se conecta en la base de datos de *Firebase* y carga los datos en la vista.

```
loadDatosBibliografia() {  
  this.subscription = this.biblioService.datosChanged.subscribe(datos => {  
    this.datosBibliografia = datos;  
    this.ID = Math.max.apply(Math, this.datosBibliografia.map(o => { return o.id }));  
    if (this.ID === -Infinity) this.ID = 0;  
  });  
  this.biblioService.fetchBibliografias();  
}
```

Vista de la Bibliografía. La vista será explicada detalladamente en el manual de usuario.



ID	Tipo	Título	Código ISBN	Editar	Borrar
1	Básico	Álgebra lineal y sus aplicaciones	978-607-32-1398-1	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	Básico	Álgebra lineal y geometría	9788478291298	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	Básico	Análisis matemático		<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	Recomendado	Complex Numbers from A to Z	9780817644499	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	Recomendado	Practical linear algebra	9781466579569	<input checked="" type="checkbox"/>	<input type="checkbox"/>

## 9.4.2 HU-17 – Borrar datos

Se creó un sólo componente en la aplicación capaz de borrar cualquier elemento en la base de datos independiente del tipo de dato.

```
this.subscription = this.deleteService.delData.subscribe(item => {
  this.delData = item;
  if (this.delData && this.delData.object === 'objetivo') {
    this.message = '¿Seguro que quieres borrar el objetivo?';
    this.messageID = 'OB' + this.delData.id;
  } else if (this.delData && this.delData.object === 'fuentes') {
    this.message = '¿Seguro que quieres borrar la fuente para la evaluación?';
    this.messageID = 'FE' + this.delData.id;
  } else if (this.delData && this.delData.object === 'tareas') {
    this.message = '¿Seguro que quieres borrar la tarea y actividad?';
    this.messageID = 'TA' + this.delData.id;
  } else if (this.delData && this.delData.object === 'recursos') {
    this.message = '¿Seguro que quieres borrar el recurso?';
    this.messageID = 'RE' + this.delData.id;
  } else if (this.delData && this.delData.object === 'resultados') {
    this.message = '¿Seguro que quieres borrar el resultado de aprendizaje?';
    this.messageID = 'RA' + this.delData.id;
  } else if (this.delData && this.delData.object === 'bibliografias') {
    this.message = '¿Seguro que quieres borrar la bibliografía?';
    this.messageID = this.delData.id;
  }
});
onOK() {
  this.deleteService.deleteObject(this.delData);
}
```

Figura 44. Código para borrar un elemento

Figura 44: El método ya contiene los mensajes predefinidos y dependiendo del Id del mensaje, cada mensaje se activa y es proyectado en la vista (con el Id adecuado). En caso el usuario confirme borrar el elemento, simplemente se llama a la función ‘delete’ de *Firestore*, que a su vez borra por completo el elemento en la base de datos.

En la siguiente figura se muestra varios ejemplos de mensajes a hora de borrar un elemento.



Ejemplos de mensajes de borrar

### 9.4.3 HU-5 - Pantalla general de gestión de asignaturas

Para obtener una capa superior a los menús de las asignaturas se creó una pantalla inicial con todas las asignaturas del curso. Aquí se puede gestionar cada asignatura, así como ver los datos de cada uno.

La página inicial de asignaturas (Figura 44) se carga el componente asignaturas de *Angular*. El componente contiene un servicio (*asignaturas.service.ts*) que se encarga de conectar con la base datos del *Firebase* y buscar todas las asignaturas disponibles. Las asignaturas son añadidas en un vector que a su vez es reflejado en la tabla del front-end de la página inicial, usando el componente *asignaturas.component.html*.

```
fetchAsignaturas() {
  this.firestore.collection('asignaturas').snapshotChanges()
    .pipe(
      map(
        docArray => {
          return docArray.map(
            doc => {
              const data = doc.payload.doc.data() as Asignatura;
              return {
                id: doc.payload.doc.id,
                codigo: data.codigo,
                nombre: data.nombre,
                curso: data.curso,
                semestre: data.semestre
              };
            }
          );
        }
      )
    )
    .subscribe((asignaturas: Asignatura[]) => {
      this.availableAsignaturas = asignaturas;
      this.asignaturasChanged.next([...this.availableAsignaturas]); // create a copy of original array
    });
}
```

Figura. 45. Código para cargar las asignaturas desde la base de datos

La página **asignaturas** consta de 4 funcionalidades: **buscador, editar, visualizar, importar**.

```
onEdit(asignatura: Asignatura) {
  this.router.navigate(['asignaturas', asignatura.id, 'datosgenerales', 'edit']);
}
```

Figura. 46. Código para cargar la vista Datos generales en modo editar

A través del mecanismo definido en *Angular* (router), el código carga el menú de “*Datos generales*” con el Id de la asignatura y en el modo de *editar* (Figura 46).

En las Figuras 47, 48, 49, 50 se muestra el código completo que se ejecuta a la hora de importar los datos de una asignatura.

```
import(docFrom: string, docTo: string, datosImportar: string[]) {
  for (let i = 0; i < datosImportar.length; i++) {
    if (datosImportar[i] === 'datosgenerales') {
      let docGenerales: AngularFirestoreDocument<DatosGenerales>;
      docGenerales = this.db.collection('asignaturas').doc(docFrom).collection('datosgenerales').doc('datosgenerales');
      docGenerales.ref.get().then(docSnap => {
        if (docSnap.exists && docSnap.data()) {
          docGenerales.valueChanges().subscribe(datos => {
            this.db.doc('asignaturas/' + docTo + '/datosgenerales/datosgenerales').update({requisitos: datos.requisitos})
              .then(() => this.notifier.notify('success', '!Guardado Correctamente!'))
              .catch((error) => this.notifier.notify('error', error));
          });
        }
      });
    } else if (datosImportar[i] === 'planensenanza') {
      let docPE: AngularFirestoreDocument<DatosPlanEnsenanza>;
      docPE = this.db.collection('asignaturas').doc(docFrom).collection('planensenanza').doc('planensenanza');
      docPE.ref.get().then(docSnap => {
        if (docSnap.exists && docSnap.data()) {
          docPE.valueChanges().subscribe(datos => {
            this.db.doc('asignaturas/' + docTo + '/planensenanza/planensenanza').set(datos)
              .then(() => this.notifier.notify('success', '!Guardado Correctamente!'))
              .catch((error) => this.notifier.notify('error', error));
          });
        }
      });
    }
  }
};
```

Figura. 47. Código para importar datos de una asignatura (1)

```
let collObj: AngularFirestoreCollection<DatosObjetivos>;
collObj = this.db.collection('asignaturas/' + docFrom + '/planensenanza/objetivos/objs');
collObj.valueChanges().subscribe(datos => {
  // if (datos.length === 0) alert(datos);
  for (let dat of datos) {
    this.db.collection('asignaturas/' + docTo + '/planensenanza/objetivos/objs')
      .doc(dat.id).set(dat);
  }
});
let collFuentes: AngularFirestoreCollection<DatosFuentesEvaluacion>;
collFuentes = this.db.collection('asignaturas/' + docFrom + '/planensenanza/fuentesevaluacion/fuentes');
collFuentes.valueChanges().subscribe(datos => {
  for (let data of datos) {
    this.db.collection('asignaturas/' + docTo + '/planensenanza/fuentesevaluacion/fuentes')
      .doc(data.id).set(data);
  }
});
} else if (datosImportar[i] === 'planaprendizaje') {
  let docPA: AngularFirestoreDocument<DatosPlanAprendizaje>;
  docPA = this.db.collection('asignaturas').doc(docFrom).collection('planaprendizaje').doc('planaprendizaje');
  docPA.ref.get().then(docSnap => {
    if (docSnap.exists && docSnap.data()) {
      docPA.valueChanges().subscribe(datos => {
        this.db.doc('asignaturas/' + docTo + '/planaprendizaje/planaprendizaje').set(datos);
      });
    }
  });
};
```

Figura. 48. Código para importar datos de una asignatura (2)

```

let collTareas: AngularFireStoreCollection<DatosTareasActividades>;
collTareas = this.db.collection('asignaturas/' + docFrom + '/planaprendizaje/tareasactividades/tareas');
collTareas.valueChanges().subscribe(datos => {
  for (let data of datos) {
    this.db.collection('asignaturas/' + docTo + '/planaprendizaje/tareasactividades/tareas')
      .doc(data.id).set(data);
  }
});
let collRecursos: AngularFireStoreCollection<DatosRecursos>;
collRecursos = this.db.collection('asignaturas/' + docFrom + '/planaprendizaje/recursos/recursos');
collRecursos.valueChanges().subscribe(datos => {
  for (let data of datos) {
    this.db.collection('asignaturas/' + docTo + '/planaprendizaje/recursos/recursos')
      .doc(data.id).set(data);
  }
});
let collResultados: AngularFireStoreCollection<DatosResultadosAprendizaje>;
collResultados = this.db.collection('asignaturas/' + docFrom + '/planaprendizaje/resultadosaprendizaje/resultados');
collResultados.valueChanges().subscribe(datos => {
  for (let data of datos) {
    this.db.collection('asignaturas/' + docTo + '/planaprendizaje/resultadosaprendizaje/resultados')
      .doc(data.id).set(data);
  }
});

```

Figura. 49. Código para importar datos de una asignatura (3)

```

} else if (datosImportar[i] === 'plantutorial') {
  let docPT: AngularFireStoreDocument<DatosPlanTutorial>;
  docPT = this.db.collection('asignaturas').doc(docFrom).collection('plantutorial').doc('plantutorial');
  docPT.ref.get().then(docSnap => {
    if (docSnap.exists && docSnap.data()) {
      docPT.valueChanges().subscribe(datos => {
        this.db.doc('asignaturas/' + docTo + '/plantutorial/plantutorial').set(datos);
      });
    }
  });
} else if (datosImportar[i] === 'bibliografia') {
  let collBiblio: AngularFireStoreCollection<DatosBibliografia>;
  collBiblio = this.db.collection('asignaturas/' + docFrom + '/bibliografias');
  collBiblio.valueChanges().subscribe(datos => {
    for (let data of datos) {
      this.db.collection('asignaturas/' + docTo + '/bibliografias').doc(data.id).set(data);
    }
  });
}
}

```

Figura. 50. Código para importar datos de una asignatura (4)

El proceso de importación es mantenido por un servicio (*import.service.ts*) que se encarga de analizar si contiene alguna asignatura a importar y qué apartados tiene a importar. Se recorre todo el vector en el que contiene el nombre del contenido a importar. Dado que todo está estructurado por colecciones y documentos en la base de datos, el servicio tiene como objetivo filtrar dichas colecciones, verificar si el documento(s) existe(n) y si contiene algún dato para ser importado a la nueva asignatura. En caso que no se encuentre datos en el documento la operación es cancelada.

El objetivo de esta funcionalidad es de ahorrar el tiempo al usuario ya no tendrá que ir a cada asignatura, cada menú de ésta, para editar los datos. Si una asignatura contiene varios datos en comunes, se puede hacer uso de esta funcionalidad para replicar dichos datos y, en un par de segundos, tendrá todos los contenidos rellenos.



The screenshot displays a web interface for managing course data. At the top, there is a 'Mostrar' dropdown set to '10' and a 'Buscador' search box. Below this is a table with columns for 'Código', 'Nombre', 'Editar', 'Visualizar', and 'Importar'. The table lists 10 courses, each with a unique code and name. Each row includes three action buttons: a blue pencil for editing, a green magnifying glass for viewing, and a blue document with a plus sign for importing. At the bottom, a pagination bar shows 'Mostrando 1 a 10 de 68 entradas' and a series of page numbers from 1 to 7, with '1' being the active page.

	Código	Nombre	Editar	Visualizar	Importar
1	40951	ÁLGEBRA Y GEOMETRÍA			
2	40959	ESTRUCTURA DE COMPUTADORES			
3	40954	FUNDAMENTOS DE COMPUTADORES			
4	40953	FUNDAMENTOS DE PROGRAMACIÓN I			
5	40958	FUNDAMENTOS DE PROGRAMACIÓN II			
6	40957	FUNDAMENTOS FÍSICOS DE LA INFORMÁTICA			
7	40955	HABILIDADES PROFESIONALES PARA INGENIEROS			
8	40960	LA EMPRESA Y SUS PROCESOS			
9	40952	MATEMÁTICA DISCRETA			
10	40956	MATEMÁTICAS PARA LA COMPUTACIÓN			

Mostrando 1 a 10 de 68 entradas      Anterior 1 2 3 4 5 6 7 Siguiente

## 9.5 Sprint 3

El Sprint 3 es el último sprint de proyecto donde se termina las restantes funcionalidades de la pila de producto y se refina alguna de las funcionalidades de los sprints anteriores.

Se define la pila de sprint desarrollada durante el sprint:

ID	Nombre
HU-4	Visualización Asignaturas [Datos generales]
HU-6	Visualización Asignaturas [Plan de enseñanza]
HU-8	Visualización Asignaturas [Plan de aprendizaje]
HU-11	Visualización Asignaturas [Plan tutorial]
HU-13	Visualización Asignaturas [Bibliografía]
HU-1	Login
HU-14	Logout

Originalmente las historias de usuario HU-4, HU-6, HU-8, HU-11 eran muy simples ya se usaba las mismas vistas de cada menú para editar. La vista, aparentemente es la misma para el usuario, pero al hacer clic en editar o visualizar, se activa el modo de lectura o escritura.

```
onView(asignatura: Asignatura) {  
  this.router.navigate(['asignaturas', asignatura.id, 'datosgenerales', 'read']);  
}
```

Figura 51. Código para cargar la vista Datos generales en modo lectura

Figura 51: El sistema carga el mismo menú que en el código de la Figura 116, pero esta vez en el modo de sólo lectura. Los elementos para modificar los datos están deshabilitados.

### 9.5.1 HU-1 – Login

Para acceder a la plataforma el usuario debe tener una cuenta de correo institucional. En la Figura 51 se muestra el código fuente para que un usuario pueda iniciar sesión. Tiene como parámetros el email y contraseña introducidos por el usuario. Los datos son verificados en la base de datos de *Firebase* y, si son correctos, el usuario accede al sistema. En caso de error *Firebase* envía un aviso.

```

login(authData: AuthData) {
  this.authFire.auth.signInWithEmailAndPassword(
    authData.email, authData.password
  )
  .catch(error => {
    alert(error);
  });
  this.initAuthListener();
}

```

Figura. 52. Código del Guard de la autenticación

Este código (Figura 53) se encarga de verificar si el usuario todavía se encuentra conectado a *Firebase*. En caso que no esté se redirecciona a la página de inicio de sesión.

```

initAuthListener(){
  this.authFire.authState.subscribe(user => {
    if (user) {
      this.isAuthenticated = true;
      if (this.router.url === '/login') {
        this.router.navigate(['asignaturas']);
      } else {
        this.router.navigate([this.router.url]);
      }
      //this.router.navigate(['asignaturas']);
    } else {
      this.asignaturaService.cancelSubscriptions();
      this.isAuthenticated = false;
      this.router.navigate(['/login']);
    }
  });
}

```

Figura. 53. Código para verificar si el usuario esta logado en *Firebase*

Vista de la interfaz de ‘Login’ (inicio de sesión). La vista será explicada detalladamente en el manual de usuario.

## 9.5.2 HU-14 - Log Out

Para desconectar del sistema el usuario tan sólo tiene que pulsar el botón de cerrar sesión presente en el menú.

```
logout() {  
    this.authFire.auth.signOut();  
}
```

Figura. 54. Código para desconectar de la aplicación

En la Figura 54 se muestra el código responsable enviar la petición al *Firebase* para hacer el “*Log Out*” (cierre de sesión) del sistema.

## 10. Presupuesto

A continuación, se mostrará una tabla con una estimación del presupuesto que supondría el desarrollo de este TFG.

<b>Presupuesto Hardware</b>			
<b>Componente</b>	<b>Duración (meses)</b>	<b>Cantidad</b>	<b>Coste</b>
Ordenador Portátil	12	1	€ 233
Total:		1	€ 233

<b>Presupuesto Software</b>		
<b>Componente</b>	<b>Cantidad</b>	<b>Coste</b>
Registro del dominio	1	€ 11,95
Alojamiento Web	1	€ 25,00
Certificados SSL	1	€ 4,95
Base de Datos (basado en Firebase de Google)	1	€ 0,00
Total:		€ 41,90

<b>Presupuesto Personal</b>				
<b>Componente</b>	<b>Cantidad</b>	<b>Horas</b>	<b>Coste</b>	<b>Total</b>
Diseño Interfaz	1	15	€ 23,00	€ 345,00
Desarrollador de la aplicación	1	168	€ 23,00	€ 3. 864,00
Tester	1	40	€ 10,00	€ 50,00
Total:	3	223		€ 4. 259,00

La tabla anterior se divide en tres bloques en función del tipo de gasto: hardware, software y gastos de personal. Aunque el bloque de gasto de personal muestra un total de 223 horas totales, se reservara con un margen de 300 horas totales. Los costes de software se asumen como mínimo un periodo de 1 año pudiendo renovar el servicio al alcanzar el tiempo de 1 año.

Los precios han sido consultados con la empresa Atxvisions (<https://atxvisions.com/>). Atxvisions es una empresa localizada en Holanda con el objetivo de crear y entrega de aplicaciones Web a sus Clientes llevando a cabo alta calidad.

## 11. Conclusiones

Llegado a esto punto me siento mucho halagado por concluir el Trabajo Fin de Título. Dado que en mi situación he empezado a trabajar antes, y con el trabajo todos los días, me hizo un poco más difícil preparar todo para terminarlo. Pero al final con un poco de esfuerzo se ha conseguido alcanzar el objetivo final.

En este TFG se ha conseguido desarrollar todas las funcionalidades planteadas al inicio del proyecto. Con la supervisión de mi Tutor se ha logrado obtener la solución completa y con la interfaz gráfica deseable. También se ha podido incorporar alguna función más de la planteada inicialmente como es “*Importar datos de las asignaturas*”. Esto permite al usuario importar datos de otras asignaturas, ahorrando así tiempo de editar los campos. También le da al usuario la flexibilidad de elegir el tipo de datos a importar.

Por mi parte, me llena de complacencia del trabajo realizado. Me llevo tiempo y dedicación para aprender el uso de las tecnologías utilizadas, ya que prácticamente no había usado Angular, HTML o CSS. El *Angular* es una herramienta muy potente ya que contiene una arquitectura sólida, núcleo bastante grande de funcionalidades y con una capacidad de customizar en la mayoría de los casos.

El proyecto me ha aportado un gran conocimiento durante la fase de desarrollo, conocer más la programación Web, saber buscar las soluciones, usar los patrones necesarios para el desarrollo de una aplicación, dar soluciones simples que facilite para su posterior mantenimiento. Me gustaría seguir en esta rama de programación para el mercado laboral y profesional. En conclusión, fue una gran satisfacción personal al ver el proyecto finalizado.

## 12. Trabajo Futuro

En este apartado se comenta algunas ideas que se podrían incluir en la aplicación en el futuro. En este TFG se ha creado una herramienta para mejorar la elaboración de los proyectos docentes en la ULPGC. Los profesores tienen a su disposición una plataforma sólida y única con el objetivo uniforme de gestionar las asignaturas del proyecto docente. No solamente traería beneficios a los profesores, pero también a la universidad en general.

Alguna de las mejoras a incluir en la aplicación actual estarían:

- Crear una base de datos más robusta, capaz de guardar más información de las asignaturas, aplicado en un entorno de SQL, obtenido así la estructura completa de base de datos definiendo las tablas y sus relaciones.
- Incorporar más funcionalidades a la aplicación
- Crear más roles para los tipos de usuario
- Mejorar la Interfaz de usuario
- Optimización de código fuente

### 13. Anexo I. Pila de Producto

ID	Nombre	Descripción	Esfuerzo	Tiempo	Prioridad	Criterios de Validación	Sprint
HU-1	Login	Acceso al sistema a través de correo electrónico y contraseña	10h	6h	Muy Alta	Se debe poder acceder a la plataforma mediante un correo y contraseña, previamente registrada en el sistema. Tras la verificación de las credenciales el usuario es redireccionado a la página general de las asignaturas.	Sprint 3
HU-2	Listado asignaturas	Lista de todas las asignaturas disponibles	20h	15h	Alta	Disponer de una página para la gestión (listado) de las asignaturas que tiene acceso el profesor. Para cada asignatura se tiene permiso de edición, lectura del contenido de éstas.	Sprint 2
HU-3	Edición Asignaturas [Datos generales]	Editar los datos generales que definen la asignatura	40h	30h	Muy Alta	Disponer de una vista de los campos del apartado “ <i>Datos generales</i> ”. Se guarda los cambios en la base datos y se actualiza la vista. Se muestra una notificación indicando que el dato se ha guardado correctamente.	Sprint 1
HU-4	Visualización Asignaturas [Datos generales]	Ver los datos generales que definen la asignatura en modo lectura.	30h	10h	Media	Disponer de una vista para leer los campos del apartado “ <i>Datos generales</i> ”. No se puede modificar ningún dato en esta vista.	Sprint 3
HU-5	Edición Asignaturas [Plan de enseñanza]	Editar los datos plan enseñanza que definen la asignatura	40h	45h	Muy Alta	Disponer de una vista para los campos del apartado “ <i>Plan de enseñanza</i> ”. Se guarda los cambios en la base datos y se actualiza la vista. Se muestra una notificación indicando que el dato se ha guardado correctamente.	Sprint 1

HU-6	Visualización del [Plan de enseñanza]	Ver el plan de enseñanza de la asignatura en modo lectura.	30h	10h	Media	Disponer de una vista para leer los campos del apartado “ <i>Plan enseñanza</i> ”. No se puede modificar ningún dato en esta vista.	Sprint 3
HU-7	Edición Asignaturas [Plan de aprendizaje]	Editar el plan de aprendizaje de la asignatura	40h	40h	Muy Alta	Disponer de una vista para leer los campos del apartado “ <i>Plan de aprendizaje</i> ”. Se guarda los cambios en la base datos y se actualiza la vista. Se muestra una notificación indicando que el dato se ha guardado correctamente.	Sprint 1
HU-8	Visualización del [Plan de aprendizaje]	Ver los datos del plan de aprendizaje la asignatura en modo lectura.	30h	10h	Media	Disponer de una vista para leer los campos del apartado “ <i>Plan aprendizaje</i> ”. No se puede modificar ningún dato en esta vista.	Sprint 3
HU-10	Edición Asignaturas [Plan tutorial]	Editar los datos del plan tutorial de la asignatura.	40h	30h	Muy Alta	Disponer de una vista para leer los campos del apartado “ <i>Plan tutorial</i> ”. Se guarda los cambios en la base datos y se actualiza la vista. Se muestra una notificación indicando que el dato se ha guardado correctamente.	Sprint 2
HU-11	Visualización del [Plan tutorial]	Ver los datos del plan tutorial de la asignatura en modo lectura.	30h	10h	Media	Disponer de una vista para leer los campos del apartado “ <i>Plan tutorial</i> ”. No se puede modificar ningún dato en esta vista.	Sprint 3
HU-12	Edición Asignaturas [Bibliografía]	Editar los datos bibliográficos de la asignatura.	40h	20h	Muy Alta	Disponer de una vista para leer los campos del apartado “ <i>Bibliografía</i> ”. Se guarda los cambios en la base datos y se actualiza la vista. Se muestra una notificación indicando que el dato se ha guardado correctamente.	Sprint 2

HU-13	Visualización de la [Bibliografía]	Ver los datos bibliográficos de la asignatura en modo lectura.	30h	8h	Media	Disponer de una vista para leer los campos del apartado “ <i>Bibliografía</i> ”. No se puede modificar ningún dato en esta vista.	Sprint 3
HU-14	Logout	Cerrar sesión en la aplicación.	10h	5h	Media	Se tiene que poder cerrar la sesión del profesor actual. El logout se hace en el sistema de autenticación de Firebase. Tras activar la acción de cierre de sesión el usuario es redireccionado a la página de inicio de sesión ( <i>login</i> ).	Sprint 3
HU-15	Importar datos asignaturas	Importar datos de otras asignaturas.	45h	35h	Media	Se debe importar datos de otras asignaturas. Se debe indicar la asignatura a importar (fuente) y la asignatura a ser importada (destino). Se especifica los datos a importar (bloque de datos). Tras activar la acción de importación se guarda los datos en la asignatura destino.	Sprint 2
HU-16	Borrar datos	Borrar un dato de la aplicación.	5h	3h	Baja	Se borra un elemento en la base datos. Se muestra un mensaje de confirmación al usuario antes de realizar la acción de borrado.	Sprint 2

## 14. Anexo II. Mockups

- **Datos Generales**

Se define el diseño previo de los contenidos para los datos generales.

The mockup shows a form titled "Nombre asignatura" with a dropdown menu for "Código asignatura". Below the title are four tabs: "Datos Generales" (selected), "Plan de Enseñanza", "Plan de Aprendizaje", and "Bibliografía". To the right of the tabs is a text input field for "Código UNESCO" containing the value "1203". Under the "Datos Generales" tab, there is a section for "Requisitos Previos:" with a list of checkboxes: "Ingeniería del Software I" (checked), "Ingeniería del Software II" (checked), "Programación I" (unchecked), "Programación II" (unchecked), "Programación III" (unchecked), and "Programación IV" (unchecked). Below this is a section for "Lengua de impartición:" with three buttons: "Español" (selected), "Inglés", and "Otro". At the bottom left, there is a text input field containing "6h".

- **Plan de Enseñanza**

Se define el modelo de diseño con todos los campos necesarios para definir el plan enseñanza. La interacción con el usuario es mediante ventanas emergentes de diálogo para operaciones de *guardar*, *selección*, *modificar*.

### Nombre asignatura Código asignatura ▾

Datos Generales | **Plan de Enseñanza** | Plan de Aprendizaje | Plan Tutorial | Bibliografía

Contribución de la asignatura:

Texto ....

Competencias asignadas: +

G1, G2, IS01, IS02

se abre la ventana cuando se pulsa el botón

Competencias asignadas: x

- N1: Descripción N1
- N2: Descripción N2
- G1: Descripción G1
- G2: ...
- IS01: ...
- IS02: ...
- ...

OK
Cancel

Objetivos:

ID	Descripción	
Obj1	Conocer los principios ...	+
Obj2	...	-
Obj3	...	x

Contenido teórico:

Texto ....

Contenido práctico:

Texto ....

Metodología:

Texto ....

se abre la ventana cuando se pulsa el botón

Objetivo 1 x

Descripción:

Texto descriptivo del objetivo ...

OK
Cancel

El Mockup para el Plan enseñanza es más complejo que de los Datos generales, ya que se compone de varios campos de textos a rellenar, tablas predefinidas. El usuario puede definir usando las ventanas emergentes de diálogo de entre ellos, campos a rellenar, casillas de selección.

Criterios y fuentes para la evaluación:

ID	Tipo	Descripción	Act. Formativas	Competencias
FE1	Pruebas escrita	Descripción 1 ...	AF1, AF2	G1, G2
FE2	...	...	...	...
FE3	...	...	...	...

Sistemas de evaluación:

Texto ....

Criterios de calificación:

Texto ....

se abre la ventana cuando se pulsa el botón

Fuentes para la evaluación:

Tipo:

Pruebas escritas

Lecturas obligatorias

... campo otros ...

Otros:

Descripción:

Texto descriptivo ...

Actividades formativas:

AF1: Descripción AF1

AF2: ...

AF3: ...

AF4: ...

Competencias:

G1: Descripción G1

G2: ...

IS01: ...

IS02: ...

OK Cancel

## • Plan de Aprendizaje

En caso del Mockup de 'Plan de Aprendizaje' sigue el patrón similar de diseño del 'Plan enseñanza' con campos de textos, tablas, casillas de selección. El contenido de los datos se compone información necesaria a definir el 'Plan aprendizaje'.

## Nombre asignatura

Código asignatura ▾

Datos Generales

Plan de Enseñanza

Plan de Aprendizaje

Plan Tutorial

Bibliografía

Tareas y actividades que realizará según distintos contextos profesionales  
(Científico, Profesional, Institucional, Social)

ID	Tipo	Contexto	
Ta1	Reuniones de grupo	(P)	+
Ta2	...	...	-
Ta3	...	...	x

se abre la ventana cuando se pulsa el botón

Temporalización semanal de tareas y actividades:

Texto ....

Tareas y actividades: x

Tipo:

Reuniones de grupo

...

...

Otros:

Contexto:

(C): Científico

(P): Profesional

(I): Institucional

(S): Social

OK Cancel

Recursos que tendrá que utilizar adecuadamente en cada uno de los contextos profesionales:

ID	Tipo	Contexto	
Re1	Ordenador PC	(P)	+
Re2	...	...	-
Re3	...	...	x

se abre la ventana cuando se pulsa el botón

Resultados de aprendizaje que tendrá que alcanzar al finalizar las distintas tareas:

ID	Descripción	Act. Formativas	
Ra1	Descripción 1 ...	AF1, AF2	+
Ra2	...	...	-
Ra3	...	...	x

se abre la ventana cuando se pulsa el botón

Resultados de aprendizaje

Descripción:

Texto descriptivo ...

Actividades formativas:

AF1: Descripción AF1

AF2: ...

AF3: ...

AF4: ...

OK Cancel



Recursos

Tipo:

Ordenador PC

...

...

Otros: campo otros ...

Contexto:

(C): Científico

(P): Profesional

(I): Institucional

(S): Social

OK Cancel

## • Plan Tutorial

El Mockup del ‘Plan tutorial’ se compone básicamente de ventanas emergentes en el que el usuario puede elegir entre las opciones dependiendo del tipo de dato.

Nombre asignatura

Código asignatura ▾

Datos Generales Plan de Enseñanza Plan de Aprendizaje **Plan Tutorial** Bibliografía

Atención presencial individualizada: +

Horario de tutoría, cita previa del alumno, ...

se abre la ventana cuando se pulsa el botón

Atención presencial individualizada:

Horario de tutoría

Cita previa del alumno

...

OK Cancel



Atención presencial a grupos de trabajo: +

Clases de laboratorio, horarios de tutoría, cita previa del grupo, ...

se abre la ventana cuando se pulsa el botón

Atención presencial a grupos de trabajo: x

- Clases de laboratorio
- Horarios de tutoría
- Cita previa del grupo
- ...

Atención telefónica: +

Horario de tutoría, ...

se abre la ventana cuando se pulsa el botón

Atención presencial individualizada: x

- Horario de tutoría
- ...

Atención virtual (on-line): +

Horario de tutoría, correo electrónico, campus virtual, ...

se abre la ventana cuando se pulsa el botón

Atención virtual (on-line) x

- Horario de tutoría
- Correo electrónico
- Campus virtual
- ...

La ventana emergente “*Atención virtual (on-line)*” define las opciones a ser seleccionadas por el usuario que estén relacionadas con el entorno virtual.

## 15. Anexo III. Manual de Usuario

### Pantalla de inicio de sesión

La aplicación comienza con la pantalla del inicio de sesión.



Figura. 55. Ventana de Login en la aplicación

En la Figura 55 se muestra la ventana de inicio de sesión de la aplicación. El acceso se hace mediante la funcionalidad de autenticación de *Firebase*. El usuario debe tener una cuenta de correo institucional y contraseña válida para conectar en la plataforma. Si el usuario no se encuentra en la base de datos, el *login* es denegado.

Validación en front-end de los campos:

- Correo electrónico – El sistema sólo aceptará correos electrónicos que contiene **@ulpgc.es**.
- Contraseña – La longitud mínima de la contraseña es de **6 caracteres**.

### Pantalla general de gestión de asignaturas

Para obtener una capa superior a los menús de las asignaturas se creó una pantalla inicial con todas las asignaturas del curso. Aquí se puede gestionar cada asignatura, así como ver los datos de cada uno. En las Figuras 55, 56, 57 se demuestra lo que viene a ser la interfaz gráfica

de gestión con la lista de todas las asignaturas del *Grado en Ingeniería informática* y sus funcionalidades disponibles. También se puede apreciar el logo de la ULPGC en la cabecera y en el pie de página el mensaje de derechos de autor.

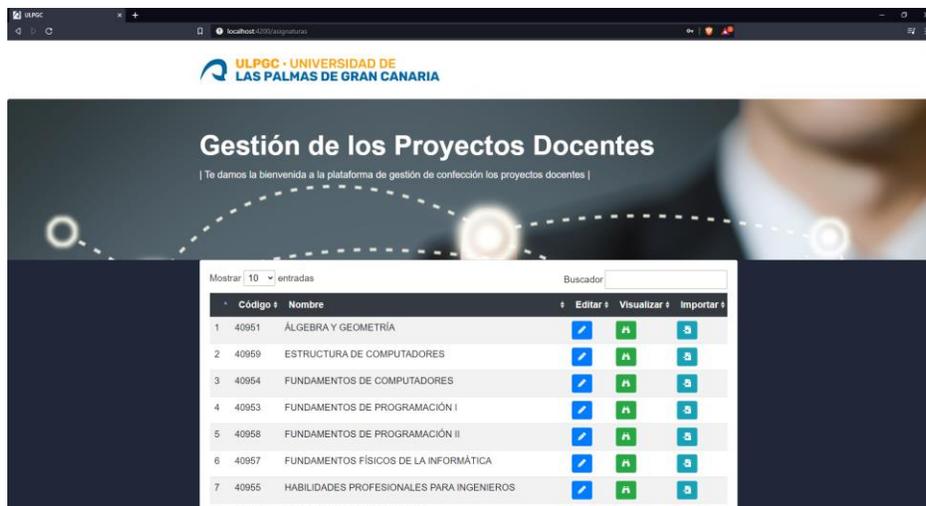


Figura. 56. Vista de la Pantalla general de asignaturas (1)

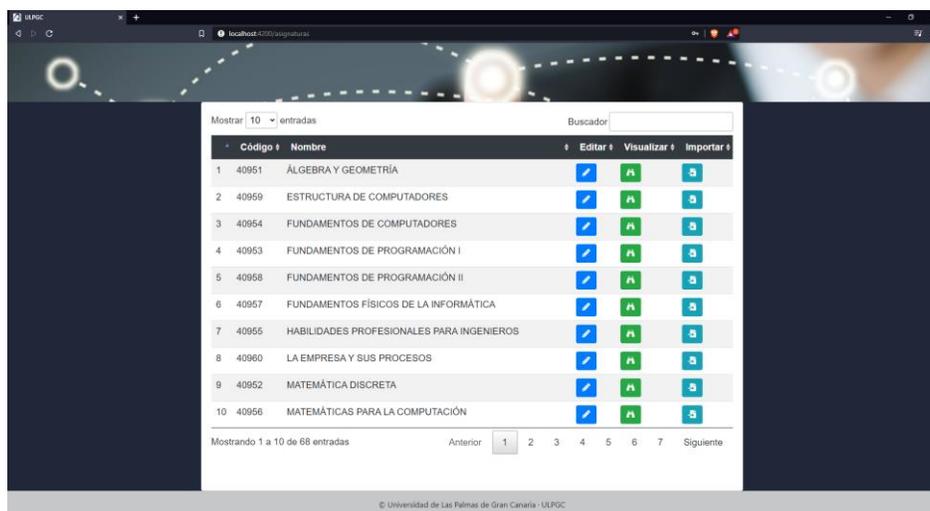


Figura. 57. Vista de la Pantalla general de asignaturas (2)

Mostrar 10 entradas Buscador

	Código	Nombre	Editar	Visualizar	Importar
1	40951	ÁLGEBRA Y GEOMETRÍA			
2	40959	ESTRUCTURA DE COMPUTADORES			
3	40954	FUNDAMENTOS DE COMPUTADORES			
4	40953	FUNDAMENTOS DE PROGRAMACIÓN I			
5	40958	FUNDAMENTOS DE PROGRAMACIÓN II			
6	40957	FUNDAMENTOS FÍSICOS DE LA INFORMÁTICA			
7	40955	HABILIDADES PROFESIONALES PARA INGENIEROS			
8	40960	LA EMPRESA Y SUS PROCESOS			
9	40952	MATEMÁTICA DISCRETA			
10	40956	MATEMÁTICAS PARA LA COMPUTACIÓN			

Mostrando 1 a 10 de 68 entradas Anterior  2 3 4 5 6 7 Siguiente

Figura. 58. Vista de la Pantalla general de asignaturas (3)

La lista de las asignaturas está dividida de acuerdo con los planes [PLAN 41 / 2019](#) y [PLAN 40 / 2010](#). Así que contiene las asignaturas de **primero y segundo** del plan 41 (2019) y las de **tercero y cuarto** del plan 40 (2010). Esto se debe a que en el momento de realización de este trabajo fin de grado es como se está impartiendo. Los códigos y los nombres de la asignatura son los mismos que están reflejados en la estructura por curso de la página oficial de la ULPGC.

## Buscador

El buscador es una funcionalidad que permite al usuario buscar cualquiera asignatura, ya sea por el nombre o por el código de la asignatura. En la Figura 59 se muestra que, al introducir el nombre, se hace un filtro en la tabla buscando la mejor combinación. La búsqueda no es sensible a mayúsculas o minúsculas.



Figura. 59. Ejemplo de búsqueda por el nombre de la asignatura

En la Figura 60, se muestra la búsqueda por el código de asignatura. Al introducir el código, se hace un filtro en la tabla buscando la mejor combinación por el número.



Figura. 60. Ejemplo de búsqueda por el código de asignatura

Por último, en la Figura 61 se muestra un ejemplo en el que el término buscado no existe. En el caso que el usuario introduzca un elemento que no esté en la tabla, se muestra un mensaje que indique que no se encuentra tal registro.

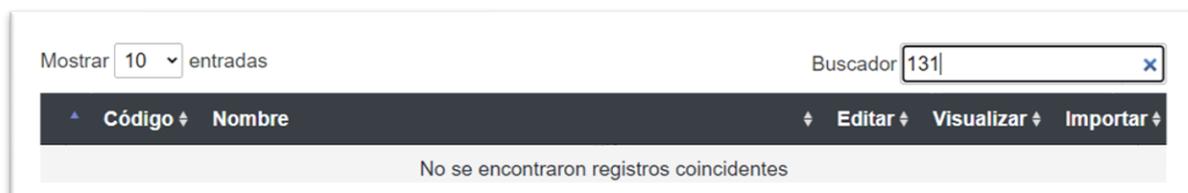


Figura. 61. Ejemplo de búsqueda fallida de un elemento que no está en la tabla

## Editar una asignatura (lectura/escritura)

Para modificar los datos de una asignatura se hace clic en botón **Editar** presente en la tabla de asignaturas (Figura 62). Al pulsar el botón, el sistema carga la ruta específica a esa asignatura con el menú principal para modificar cada apartado de esta.

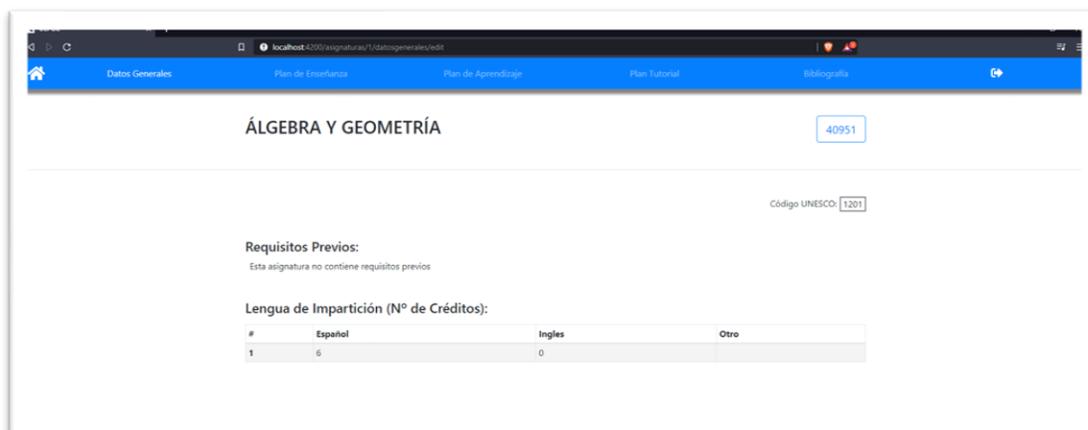


The screenshot shows a table with columns for 'Código' and 'Nombre'. The first row is '40951 ÁLGEBRA Y GEOMETRÍA'. The 'Editar' button (a blue pencil icon) for this row is highlighted with a red square. Other buttons for 'Visualizar' and 'Importar' are also visible.

	Código	Nombre	Editar	Visualizar	Importar
1	40951	ÁLGEBRA Y GEOMETRÍA			
2	40959	ESTRUCTURA DE COMPUTADORES			
3	40954	FUNDAMENTOS DE COMPUTADORES			

Figura. 62. Ejemplo de edición de una asignatura

Después de pulsar el botón (Figura 62), la información mostrada corresponde con los “*Datos generales*” (primer menú). A partir de este punto el usuario puede moverse entre otros apartados como “*Plan tutorial*” dentro del menú principal (situado en la parte superior de la página).



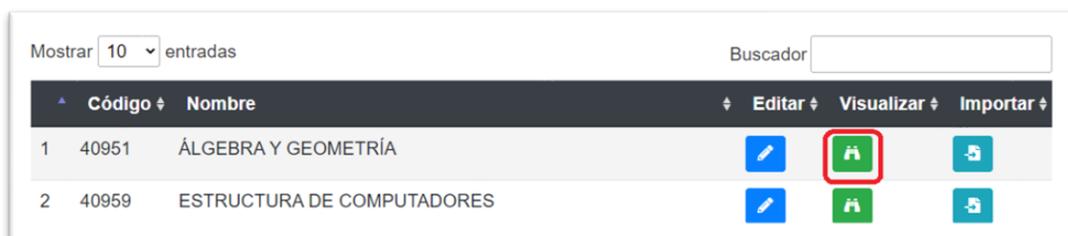
The screenshot shows the 'Datos generales' view for the course 'ÁLGEBRA Y GEOMETRÍA' (code 40951). It includes a navigation menu at the top, the course title and code, the UNESCO code (1201), and a table for 'Requisitos Previos' and 'Lengua de Impartición (Nº de Créditos)'. The 'Requisitos Previos' section states 'Esta asignatura no contiene requisitos previos'. The 'Lengua de Impartición' table shows 6 credits in Spanish and 0 in English.

#	Español	Inglés	Otro
1	6	0	

Figura. 63. Ejemplo de la vista Datos generales

## Visualizar una asignatura (sólo lectura)

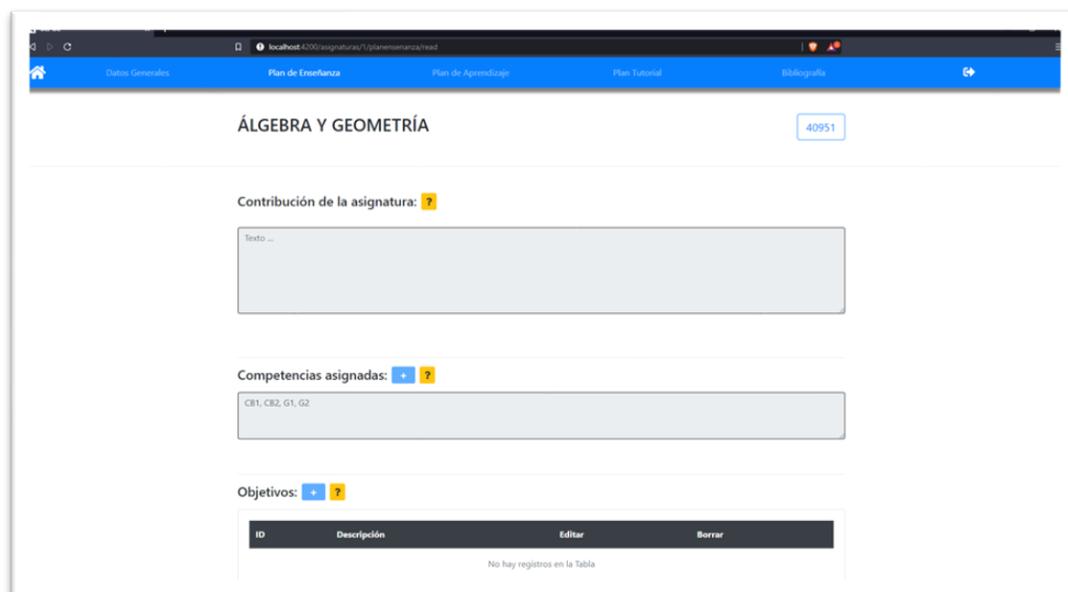
Para visualizar la información de una determinada asignatura en modo lectura se emplea la función *visualizar* (Figura 64).



	Código	Nombre	Editar	Visualizar	Importar
1	40951	ÁLGEBRA Y GEOMETRÍA			
2	40959	ESTRUCTURA DE COMPUTADORES			

Figura. 64. Ejemplo para visualizar una asignatura

En el modo visualizar el usuario no puede hacer modificaciones en los campos. Se le permite solamente ver y leer el contenido relacionado a la asignatura (Figura 65). Los botones para editar se encuentran deshabilitados.



ÁLGEBRA Y GEOMETRÍA 40951

Contribución de la asignatura: ?

Texto ...

Competencias asignadas: + ?

CB1, CR2, G1, G2

Objetivos: + ?

ID	Descripción	Editar	Borrar
No hay registros en la Tabla			

Figura. 65. Ejemplo de vista en modo lectura

## Importar los datos de otra asignatura

En muchas ocasiones el contenido de dos asignaturas se asemeja bastante. Sería deseable que la aplicación disponga de una funcionalidad que permitiera al usuario importar los datos directamente de otra asignatura. De esta forma, sólo habría que concentrarse en modificar aquella información que las diferencias. Esta funcionalidad permite reducir drásticamente el esfuerzo del profesorado en la confección del proyecto docente.



Figura. 66. Ejemplo para importar datos de otra asignatura

En la Figura 66 se muestra la funcionalidad que permite al usuario importar los datos directamente de otra asignatura en caso compartan datos similares. Al pulsar el botón importar de una asignatura, se abre una ventana emergente (Figura 67), donde el usuario puede elegir qué asignatura y los campos del menú que desea importar los datos. Se trata de una operación muy cómoda para el usuario pero costosa computacionalmente hablando. El sistema tiene que buscar todos los campos en la base de datos (si están disponibles), recorriendo varias colecciones, subcolecciones y documentos de la misma. Así que si el usuario no elige la asignatura o elige la asignatura pero no elige un apartado, la operación de importación no se inicia en backend.

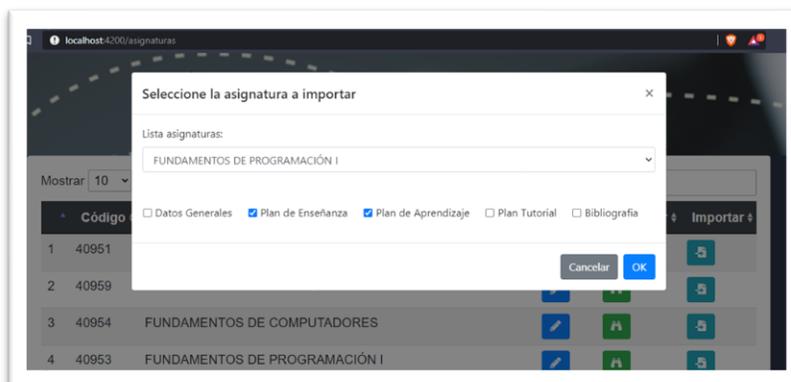
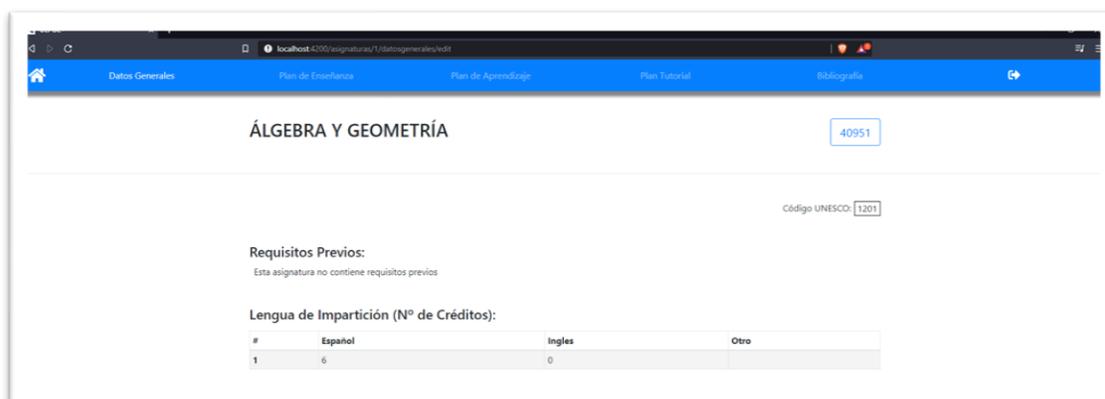


Figura. 67. Ventana para importar datos de otras asignaturas

## Vista “*Datos generales*” de una asignatura

En este menú se definen los datos generales de la asignatura (Figura 68). Se define el código Unesco de la asignatura, los requisitos previos, y una tabla que contiene el número de créditos a impartir relacionada con la lengua. Los datos del código Unesco y las lenguas de impartición se han tomados desde los documentos oficiales del proyecto docente de cada asignatura presente en la ULPGC. Así que cada asignatura tendrá el código Unesco oficial.



The screenshot shows a web interface for editing course data. The title is 'ÁLGEBRA Y GEOMETRÍA' with a course ID of 40951. The UNESCO code is 1201. The 'Requisitos Previos' section indicates that this course has no prerequisites. The 'Lengua de Impartición (Nº de Créditos)' table shows 6 credits in Spanish and 0 in English.

#	Español	Inglés	Otro
1	6	0	

Figura. 68. Vista Datos generales

El campo **requisitos previos** actúa como un filtro, dependiendo de la asignatura seleccionada. En la Figura 69 se observa que todas las asignaturas de primer curso y semestre no tienen ninguna asignatura como conocimiento previo ya que es una asignatura del primer curso. Se muestra el mensaje al usuario de no contener requisitos previos.



The screenshot shows the 'Requisitos Previos' section for the course 'FUNDAMENTOS DE COMPUTADORES' (ID 40954, UNESCO code 1203). It displays the message 'Esta asignatura no contiene requisitos previos' and the same language credit table as in Figure 68.

#	Español	Inglés	Otro
1	6	0	

Figura. 69. Requisitos previos Fundamentos de computadores

Figura 70: En caso de otras asignaturas (que no esté en el primer curso y semestre), se muestra una lista de asignaturas disponibles en el que el usuario podrá elegir los requisitos a esa asignatura.

### ESTRUCTURA DE COMPUTADORES

40959

Código UNESCO: 1203

**Requisitos Previos:**

- ÁLGEBRA Y GEOMETRÍA
- FUNDAMENTOS DE COMPUTADORES
- FUNDAMENTOS DE PROGRAMACIÓN I
- HABILIDADES PROFESIONALES PARA INGENIEROS
- MATEMÁTICA DISCRETA

**Lengua de Impartición (Nº de Créditos):**

#	Español	Inglés	Otro
1	6	0	

Figura. 70. Requisitos previos de la asignatura “Estructura de Computadores”

## Vista “Plan de Enseñanza” de una asignatura

En la Figura 71 se define los primeros campos para el plan de enseñanza:

- **Contribución de la asignatura** - Texto descriptivo a definir la contribución la asignatura.
- **Competencias asignadas** – se compone de un botón que al hacer clic se abre una ventana para que el usuario elegir las competencias necesarias para dicha asignatura.



Figura. 71. Vista de Plan enseñanza (1)

El usuario puede seleccionar las competencias adecuadas a la asignatura seleccionada (Figura 72). Dado que la lista oficial con las competencias contiene más caracteres, se ha ajustado los textos con un margen de 100 caracteres por fila. El código y la descripción de cada competencia se ha tonado del enlace oficial de la universidad - [https://www.eii.ulpgc.es/tb\\_university\\_ex/?q=objtivos-y-competencias-del-gii#Segundo Enlace](https://www.eii.ulpgc.es/tb_university_ex/?q=objtivos-y-competencias-del-gii#Segundo%20Enlace).

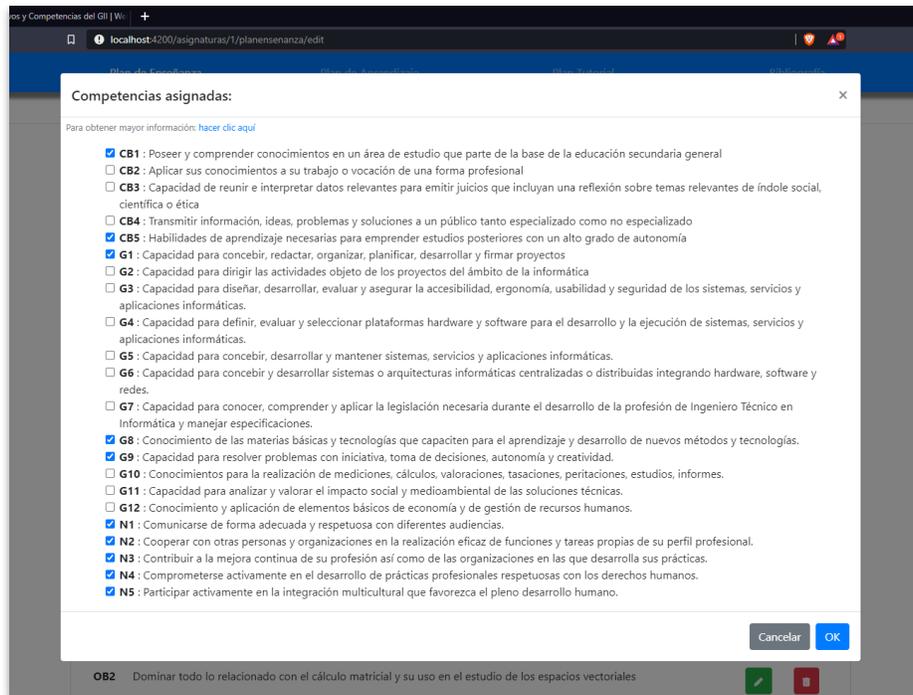


Figura. 72. Ventana para añadir competencias

Al elegir las competencias, estas son guardadas en el campo de texto, solamente con la identificación de cada competencia (Figura 73). El campo de texto está deshabilitado para su edición. En caso de modificar el usuario debe abrir la ventana de competencias y usar la casilla de selección.



Figura. 73. Ejemplo de competencias asignadas

El siguiente campo del plan enseñanza son los *objetivos* (Figura 74):

- **Objetivos** – Se compone por una tabla en que el usuario puede describir los objetivos de las asignaturas. La tabla contiene el *Id del objetivo* y la *descripción*.

Objetivos: + ?

ID	Descripción	Editar	Borrar
OB1	Introducir al alumnado en el uso del lenguaje matemático y de algunos métodos del razonamiento abstracto	✓	✗
OB2	Dominar todo lo relacionado con el cálculo matricial y su uso en el estudio de los espacios vectoriales	✓	✗
OB3	Estudiar las transformaciones o aplicaciones lineales y sus propiedades, haciendo énfasis en la representación matricial de las mismas	✓	✗
OB4	Introducir el concepto de producto escalar y estudiar y analizar los espacios euclídeos dentro del marco general de los espacios vectoriales	✓	✗
OB5	Estudiar las formas bilineales y cuadráticas, profundizando en lo relativo a la diagonalización de las mismas	✓	✗
OB6	Estudiar la geometría afín, en concreto los espacios métricos E2 y E3 y los espacios puntuales	✓	✗
OB7	Aprender a expresarse con fluidez y precisión	✓	✗

Figura. 74. Vista del Plan enseñanza (2)

El usuario debe escribir un texto descriptivo en la ventana emergente para caracterizar el nuevo objetivo a alcanzar con la asignatura (Figura 75).

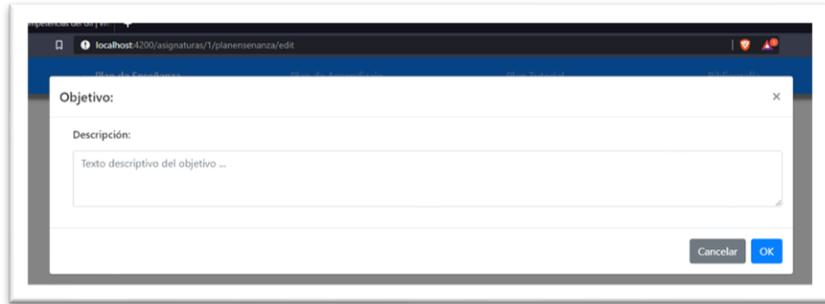


Figura. 75. Ventana para añadir objetivos

Para editar un objetivo se debe seleccionar el objetivo y hacer uso del botón editar (Figura 76). Se abre al usuario la misma ventana para añadir un objetivo, pero con la diferencia que contiene el texto del objetivo seleccionado (Figura 77). Al hacer clic en OK de la ventana emergente los datos se guardan en la base de datos.

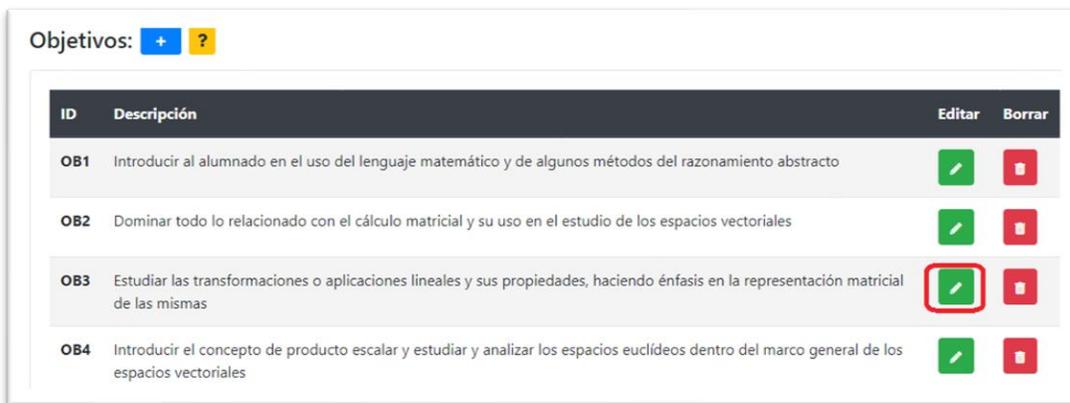


Figura. 76. Editar un objetivo

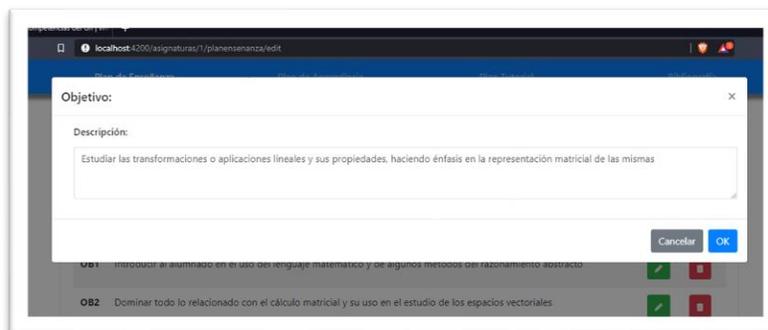


Figura. 77. Ventana para editar un objetivo

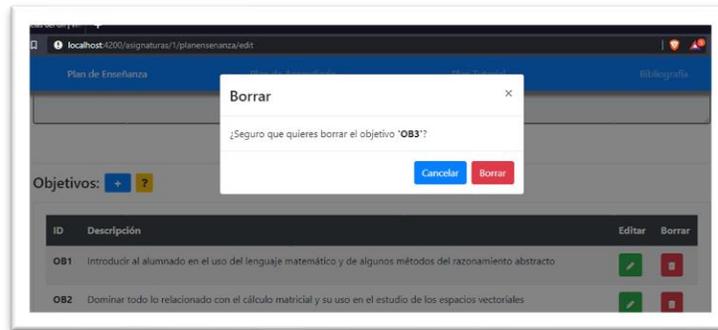


Figura. 78. Borrar un objetivo

Al hacer clic en borrar, se abre una ventana para que el usuario puede confirmar si está seguro para borrar el registro (Figura 78). En caso confirme el dato se borra desde la base datos.

En la Figura 79 se ilustra los siguientes campos:

- **Contenido teórico** - Texto descriptivo a definir los contenidos teóricos. En el siguiente ejemplo se expone los contenidos de la asignatura “*Álgebra y Geometría*” estructurada por temas.
- **Contenido práctico** - Texto descriptivo a definir los contenidos prácticos. Se debe listar la estructura a usar para las prácticas.

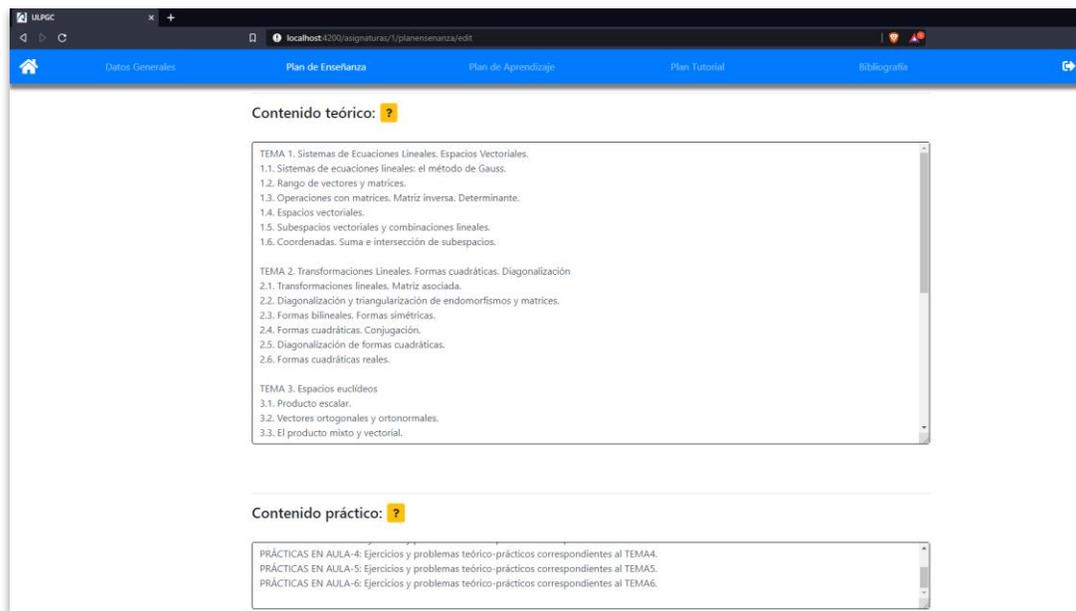


Figura. 79. Vista del Plan enseñanza (3)

Figura 80: A continuación, se expone los campos:

- **Metodología** - Texto descriptivo a definir la metodología. Debe contener las metodologías docentes básicas que forman las asignaturas de la titulación.
- **Criterios y fuentes de evaluación** – contiene una tabla con los campos necesarios para definir las fuentes para la evaluación.
- **Sistemas de evaluación** - Texto descriptivo a definir el tipo de sistema para evaluación ya sea de forma presencial o no presencial.

The screenshot shows a web interface for editing a 'Plan de Enseñanza'. It has a blue header with navigation tabs: 'Plan de Enseñanza', 'Plan de Aprendizaje', 'Plan Tutorial', and 'Bibliografía'. The main content is divided into three sections:

- Metodología:** A text area containing three paragraphs:
  - AF1.- Sesiones académicas de fundamentación; exposición de los contenidos teóricos.
  - AF2.- Sesiones académicas de interacción: se resolverán ejercicios y problemas en una clase práctica. La resolución se realizará preferiblemente por los estudiantes, valorándose su participación. Cuando sea necesario o conveniente, se discutirán en detalle aspectos concretos por toda la clase.
  - AF3.- Sesiones de tutorización.
- Criterios y fuentes para la evaluación:** A table with columns: ID, Tipo, Descripción, Act. Formativas, Competencias, Editar, and Borrar.
 

ID	Tipo	Descripción	Act. Formativas	Competencias	Editar	Borrar
FE1	Exámenes y ejercicios no presenciales	Se realizará un examen de conjunto, en el que las cuestiones podrán ser de carácter teórico, práctico o teórico-práctico.	AF1, AF2, AF3, AF4	CB1, CB5, G1, G8, G9		
FE2	Asistencia y participación		AF1, AF2, AF3	N1, N2, N3, N4, N5		
- Sistemas de evaluación:** A text area containing one paragraph:
 

Para cada una de las convocatorias oficiales, ordinarias o extraordinarias, se realizará un examen de conjunto, en el que las cuestiones podrán ser de carácter teórico, práctico o teórico-práctico. El examen se complementa con las actividades en clase y tutorías a lo largo del curso y la participación en el Campus Virtual.

Figura. 80. Vista del Plan enseñanza (4)

Figura 81: Para definir una fuente de evaluación, el usuario debe de completar el formulario de la presente figura:

- **Tipo** – Casilla de selección para definir el tipo de la evaluación. La lista de opciones se cogió con base del documento de la página oficial de la Escuela de Ingeniería Informática.

[https://www.eii.ulpgc.es/tb\\_university\\_ex/sites/default/files/files/Grado%20en%20Ingenier%C3%ADa%20Inform%C3%A1tica.pdf](https://www.eii.ulpgc.es/tb_university_ex/sites/default/files/files/Grado%20en%20Ingenier%C3%ADa%20Inform%C3%A1tica.pdf) (5.1.5 Descripción de los sistemas de

evaluación). Si el usuario no encuentra el tipo deseado en la lista, puede seleccionar la casilla ‘Otros’ y definir el Tipo (Figura 82).

Figura. 81. Ventana para añadir una “fuente para la evaluación”

- **Descripción** – texto descriptivo del tipo de fuente evaluación seleccionada.
- **Actividades Formativas** - Casilla de selección para definir las actividades formativas. La lista de opciones se cogió con base del documento de la página oficial de la Escuela informática.

[https://www.eii.ulpgc.es/tb\\_university\\_ex/sites/default/files/files/Grado%20en%20Ingenier%C3%ADa%20Inform%C3%A1tica.pdf](https://www.eii.ulpgc.es/tb_university_ex/sites/default/files/files/Grado%20en%20Ingenier%C3%ADa%20Inform%C3%A1tica.pdf) (5.1.3 Descripción de las actividades formativas). Lo mismo que en el campo Tipo, el usuario también puede definir una actividad que no esté en la lista.

**Tipo:**

- Exámenes y ejercicios presenciales
- Exámenes y ejercicios no presenciales
- Trabajos prácticos
- Presentaciones y defensas
- Evaluación por pares
- Participación activa
- Otros

campo otros...

Figura. 82. Ejemplo de otro tipo de fuente de evaluación

- **Competencias** – la funcionalidad es la misma que en el campo Competencias asignadas del Plan enseñanza, pero la lista de selección está condicionada a las competencias que el usuario elige en el primer paso de asignar competencias (Figura 83.b). O sea, todas las competencias que el usuario seleccione en la Figura 83.a, se refleja en este apartado. Al igual si el usuario quite una selección previamente seleccionada el filtro se actualiza automáticamente.

**Competencias asignadas:** + ?

CB1, CB5, G1, G8, G9, N1, N2, N3, N4, N5

**Competencias:**

- CB1** : Poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general
- CB5** : Habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía
- G1** : Capacidad para concebir, redactar, organizar, planificar, desarrollar y firmar proyectos
- G8** : Conocimiento de las materias básicas y tecnologías que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías.
- G9** : Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad.
- N1** : Comunicarse de forma adecuada y respetuosa con diferentes audiencias.
- N2** : Cooperar con otras personas y organizaciones en la realización eficaz de funciones y tareas propias de su perfil profesional.
- N3** : Contribuir a la mejora continua de su profesión así como de las organizaciones en las que desarrolla sus prácticas.
- N4** : Comprometerse activamente en el desarrollo de prácticas profesionales respetuosas con los derechos humanos.
- N5** : Participar activamente en la integración multicultural que favorezca el pleno desarrollo humano.

Figura. 83. Ejemplo de (a) competencias seleccionadas, (b) competencias a seleccionar

Figura 83.a: Contiene la lista de todas competencias seleccionadas en la sección de competencias asignadas.

Figura 84: Con la tabla de “*Criterios y fuentes para la evaluación*” definida, al crear un registro el botón de **Editar** será visible para el usuario, en el que al clicar se puede editar los datos designados al ID del registro.

riterios y fuentes para la evaluación: + ?

ID	Tipo	Descripción	Act.		Editar	Borrar
			Formativas	Competencias		
FE1	Exámenes y ejercicios no presenciales	Se realizará un examen de conjunto, en el que las cuestiones podrán ser de carácter teórico, práctico o teórico-práctico.	AF1, AF2, AF3, AF4	CB1, CB5, G1, G8, G9		
FE2	Asistencia y participación		AF1, AF2, AF3	N1, N2, N3, N4, N5		

Figura. 84. Editar fuentes para la evaluación

Figura 85. Al igual que los objetivos, para editar las fuentes de evaluación se abre la ventana con los datos que ya se encuentran en la tabla. El usuario puede modificar cada campo.

localhost:4200/asignaturas/1/plansemananza/edit

Tipo:

- Exámenes y ejercicios presenciales
- Exámenes y ejercicios no presenciales
- Trabajos prácticos
- Presentaciones y defensas
- Evaluación por pares
- Participación activa
- Otros

Descripción:

Se realizará un examen de conjunto, en el que las cuestiones podrán ser de carácter teórico, práctico o teórico-práctico.

Actividades formativas:

- AF1 : Sesiones académicas de fundamentación
- AF2 : Sesiones académicas de interacción
- AF3 : Sesiones académicas de aplicación
- AF4 : Sesiones de tutorización
- AF5 : Trabajos
- AF6 : Estudio
- AF7 : Otro

Competencias:

- CB1 : Poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general
- CB5 : Habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía
- G1 : Capacidad para concebir, redactar, organizar, planificar, desarrollar y firmar proyectos
- G8 : Conocimiento de las materias básicas y tecnologías que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías.
- G9 : Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad.
- N1 : Comunicarse de forma adecuada y respetuosa con diferentes audiencias.
- N2 : Cooperar con otras personas y organizaciones en la realización eficaz de funciones y tareas propias de su perfil profesional.
- N3 : Contribuir a la mejora continua de su profesión así como de las organizaciones en las que desarrolla sus prácticas.
- N4 : Comprometerse activamente en el desarrollo de prácticas profesionales respetuosas con los derechos humanos.
- N5 : Participar activamente en la integración multicultural que favorezca el pleno desarrollo humano.

Cancelar OK

Figura. 85. Ventana para editar una fuente para la evaluación

Para todos los campos de edición, cuando el usuario pulsa los botones de confirmación, se sale un ‘popover’ en el canto inferior derecho notificando al usuario que el dato ha sido guardado correctamente (Figura 86).

Criterios y fuentes para la evaluación: + ?

ID	Tipo	Descripción	Act. Formativas	Competencias	Editar	Borrar
FE1	Exámenes y ejercicios presenciales	Se realizará un examen de conjunto, en el que las cuestiones podrán ser de carácter teórico, práctico o teórico-práctico.	AF1, AF2, AF3, AF4	CB1, CB5, G1, G8, G9	<span style="color: green;">✎</span>	<span style="color: red;">✖</span>
FE2	Asistencia y participación		AF1, AF2, AF3	N1, N2, N3, N4, N5	<span style="color: green;">✎</span>	<span style="color: red;">✖</span>

Sistemas de evaluación: ?

Para cada una de las convocatorias oficiales, ordinarias o extraordinarias, se realizará un examen de conjunto, en el que las cuestiones podrán ser de carácter teórico, práctico o teórico-práctico. El examen se complementa con las actividades en clase y tutorías a lo largo del curso y la participación en el Campus Virtual.

!Guardado Correctamente! ✕

Figura. 86. Ejemplo de editar una fuente para la evaluación

Borrar

¿Seguro que quieres borrar la fuente para la evaluación "FE2"?

Cancelar Borrar

ID	Tipo	Descripción	Act. Formativas	Competencias	Editar	Borrar
FE1	Exámenes y ejercicios presenciales	Se realizará un examen de conjunto, en el que las cuestiones podrán ser de carácter teórico, práctico o teórico-práctico.	AF1, AF2, AF3, AF4	CB1, CB5, G1, G8, G9	<span style="color: green;">✎</span>	<span style="color: red;">✖</span>

Figura. 87. Ejemplo borrar una fuente evaluación

Figura 87: Para borrar un elemento de la tabla el usuario debe confirmar en el dialogo de la ventana emergente.

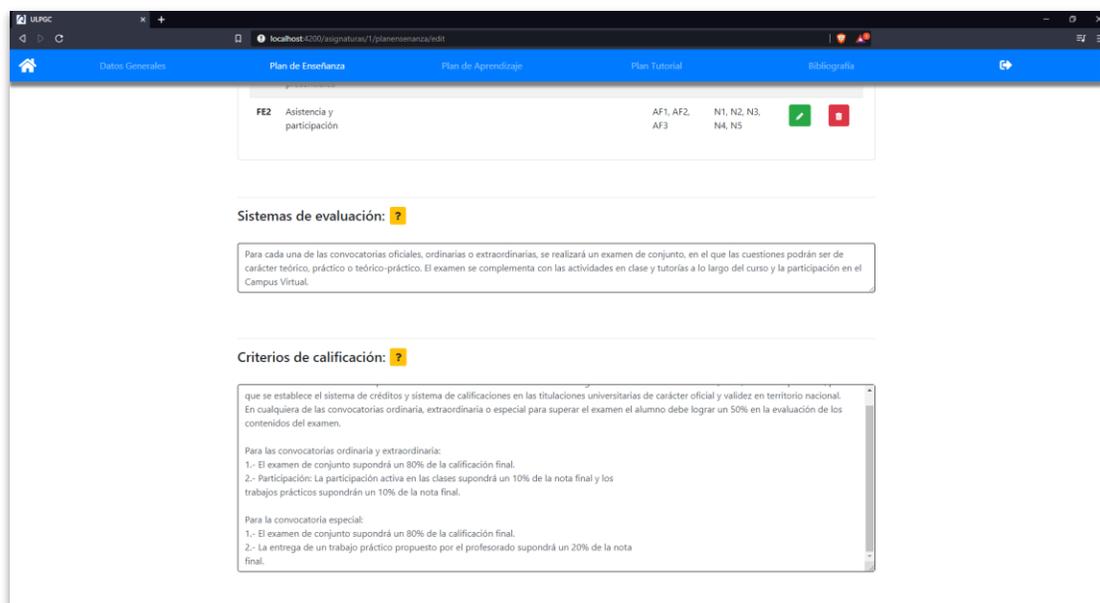


Figura. 88. Vista del Plan enseñanza (5)

Figura 88: Los últimos campos que componen el Plan enseñanza son:

- **Sistemas de evaluación** - Texto descriptivo a definir el tipo sistema para la evaluación.
- **Criterios de calificación** - Texto descriptivo a definir los criterios a calcular la calificación.

## Vista “Plan de Aprendizaje”

Para el menú “Plan de Aprendizaje” (Figura 89) primeramente se define el siguiente dato:

- **Tareas y actividades que realizará según distintos contextos profesionales** – se componen de una tabla simple con los datos a definir el tipo y el contexto. Según la documentación oficial de los proyectos docentes los contextos pueden ser *científico*, *profesional*, *institucional* y *social*.

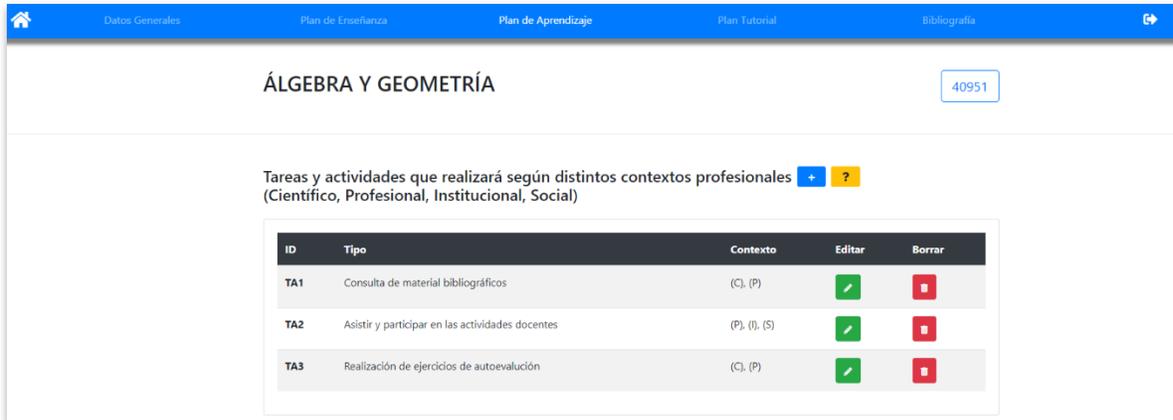


Figura. 89. Vista del Plan de Aprendizaje (1)

Al clicar en el botón añadir se abre la ventana emergente (Figura 90) en el que usuario tiene a elegir entre las casillas para definir el tipo y contexto de cada tarea. Una tarea puede tener más de un contexto. La lista de tareas fue tomada de los proyectos docentes de la ULPGC.

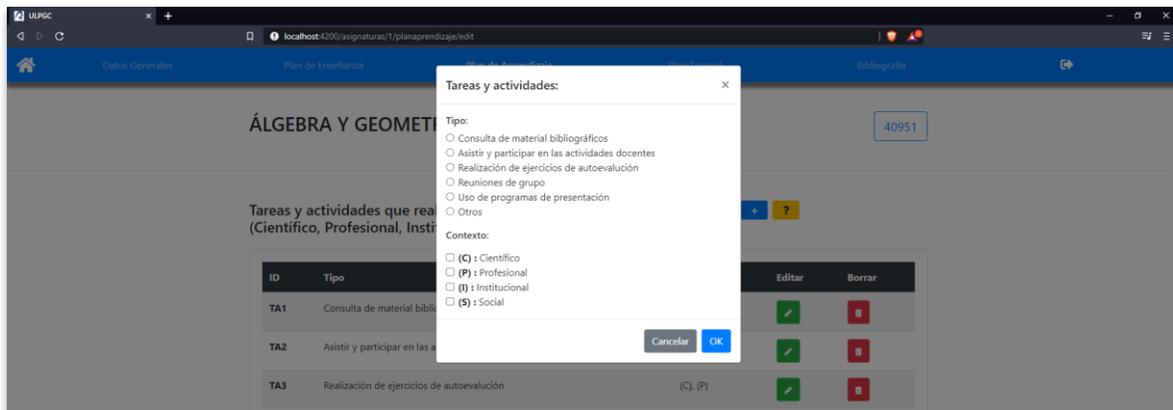


Figura. 90. Añadir tareas y actividades

En la Figura 91 se muestra que al igual que los otros campos, el usuario puede definir un tipo de tareas que no esté en la lista.

**Tareas y actividades:** ×

---

**Tipo:**

Consulta de material bibliográficos  
 Asistir y participar en las actividades docentes  
 Realización de ejercicios de autoevaluación  
 Reuniones de grupo  
 Uso de programas de presentación  
 Otros

campo otros...

**Contexto:**

(C) : Científico  
 (P) : Profesional  
 (I) : Institucional  
 (S) : Social

Cancelar
OK

Figura. 91. Ejemplo del campo otro para tareas y actividades

Figura 92: Ejemplo al pulsar el botón **Editar** para modificar los datos del registro TA2 de la tabla.

**Tareas y actividades que realizará según distintos contextos profesionales** + ?

(Científico, Profesional, Institucional, Social)

ID	Tipo	Contexto	Editar	Borrar
TA1	Consulta de material bibliográficos	(C), (P)	<span style="color: green;">✎</span>	<span style="color: red;">✖</span>
TA2	Asistir y participar en las actividades docentes	(P), (I), (S)	<span style="color: green; border: 2px solid red;">✎</span>	<span style="color: red;">✖</span>
TA3	Realización de ejercicios de autoevaluación	(C), (P)	<span style="color: green;">✎</span>	<span style="color: red;">✖</span>

Figura. 92. Editar una tarea

Al igual que las otras tablas para editar un registro, se abre la ventana responsable para la edición (Figura 93). Los campos registrados se reflejan en cada ítem de la tabla.

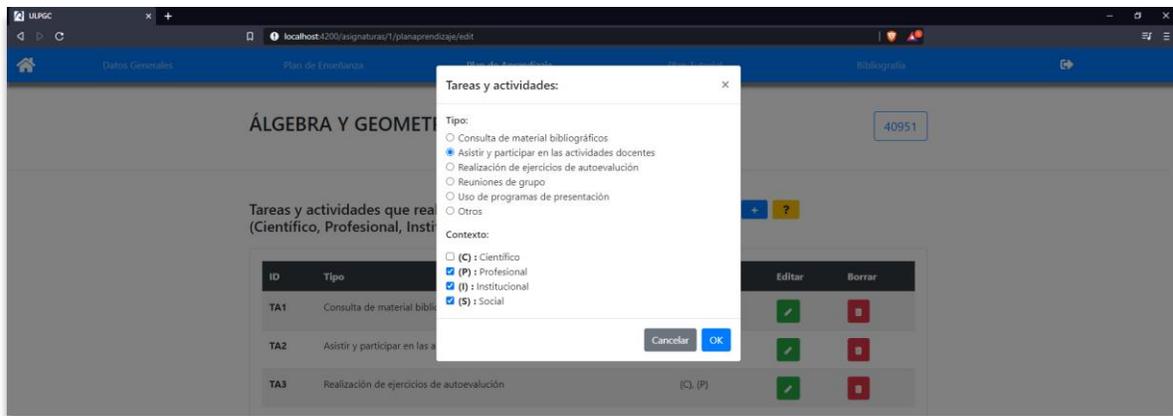


Figura. 93. Ventana para editar una tarea

El siguiente dato a definir dentro del plan de aprendizaje (Figura 94):

- **Temporalización semanal de tareas y actividades** – Texto descriptivo con una la sincronización estructura de las tareas.



Figura. 94. Vista del Plan de Aprendizaje (2)

Para completar el apartado Plan de aprendizaje se define los datos (Figura 95):

- **Recursos** – representa una tabla con el tipo de recurso que tendrá que utilizar adecuadamente en los contextos. La lista de recursos fue tomada a partir de la información presente en los proyectos docentes de la ULPGC.

- **Resultados de aprendizaje** – representa los resultados a alcanzar después de finalizar las distintas tareas. Lo componen la tabla con los campos descripción y actividades formativas.

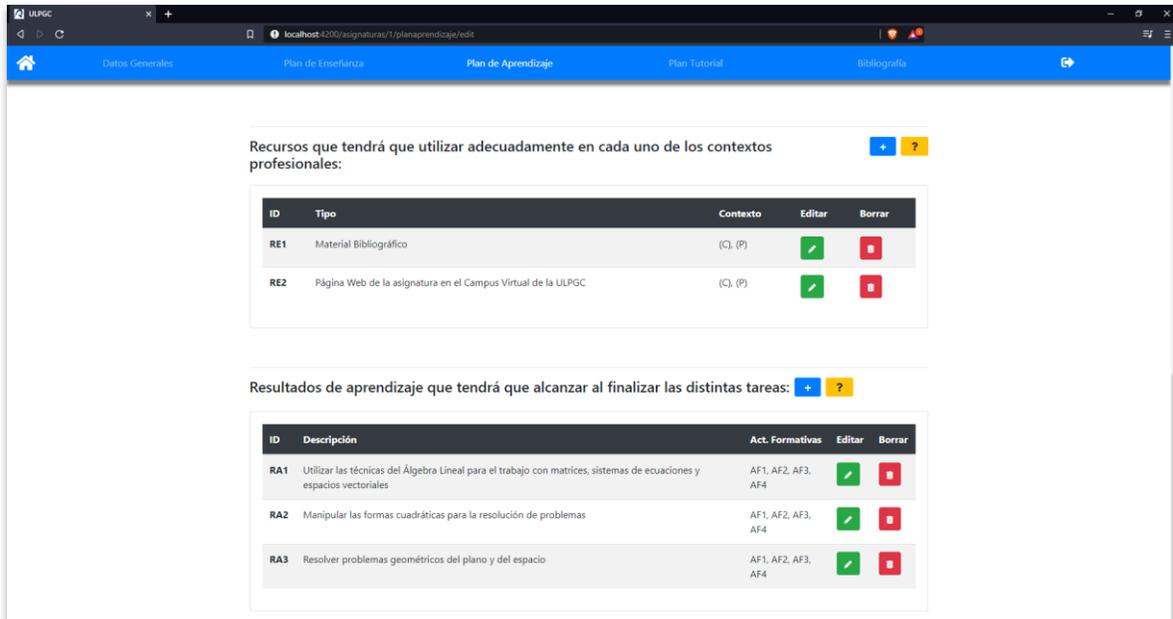


Figura. 95. Vista del Plan de Aprendizaje (3)

Las actividades formativas son las mismas que se ha definido en el apartado de *Plan de enseñanza*. La lista de resultados de aprendizaje fue tomada a partir de la información presente en los proyectos docentes de la universidad.

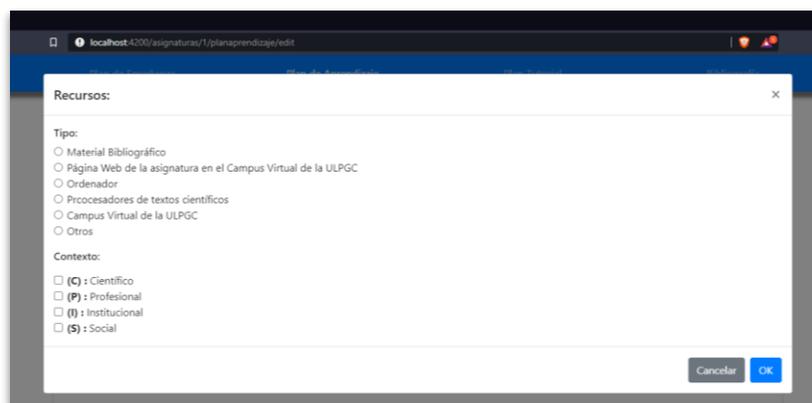


Figura. 96. Añadir un recurso

Para añadir un recurso nuevo en la tabla, se abre un dialogo (Figura 96) al usuario con los datos requeridos para añadir un recurso. Se trata de casillas que el usuario habilita/deshabilita.

**Recursos:**

---

**Tipo:**

- Material Bibliográfico
- Página Web de la asignatura en el Campus Virtual de la ULPGC
- Ordenador
- Procesadores de textos científicos
- Campus Virtual de la ULPGC
- Otros

**Contexto:**

- (C) : Científico
- (P) : Profesional
- (I) : Institucional
- (S) : Social

Figura. 97. Ejemplo del campo otros para recursos

En caso el usuario no encuentre el Tipo recurso deseado, se puede hacer uso de la casilla *Otros* (Figura 97). Al hacer clic se abre un campo de texto para definir el nuevo recurso.

localhost:4200/asignaturas/1/planaprendizaje/edit

**Recursos:**

---

**Tipo:**

- Material Bibliográfico
- Página Web de la asignatura en el Campus Virtual de la ULPGC
- Ordenador
- Procesadores de textos científicos
- Campus Virtual de la ULPGC
- Otros

**Contexto:**

- (C) : Científico
- (P) : Profesional
- (I) : Institucional
- (S) : Social

Cancelar OK

Figura. 98. Ventana para editar un recurso

El componente para editar un recurso es la misma que para añadir un nuevo recurso (Figura 98). La diferencia radica en que, para editar un recurso, los campos que ya estén seleccionados son visibles al usuario. El usuario puede modificarlos.

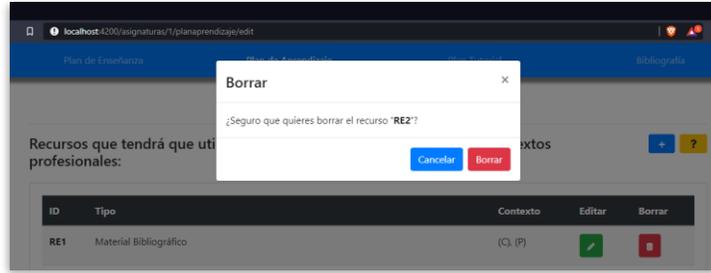


Figura. 99. Ejemplo de Borrar un recurso

La lista de actividades formativas (Figura 100) es la misma que en el primer apartado del *Plan de Enseñanza*. También contiene con el campo “*Otros*” para definir una actividad que no esté en la lista.

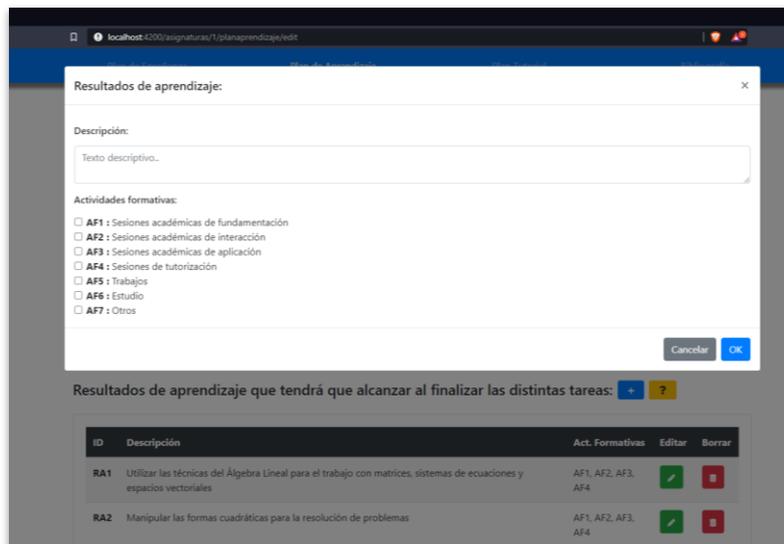


Figura. 100. Añadir un resultado de aprendizaje

Figura 101: Ejemplo al pulsar el botón **Editar** para modificar los datos del registro RA1 de la tabla. Se abre la misma ventana la Figura 100 con los datos para modificar.

Resultados de aprendizaje que tendrá que alcanzar al finalizar las distintas tareas: + ?

ID	Descripción	Act. Formativas	Editar	Borrar
RA1	Utilizar las técnicas del Álgebra Lineal para el trabajo con matrices, sistemas de ecuaciones y espacios vectoriales	AF1, AF2, AF3, AF4		
RA2	Manipular las formas cuadráticas para la resolución de problemas	AF1, AF2, AF3, AF4		
RA3	Resolver problemas geométricos del plano y del espacio	AF1, AF2, AF3, AF4		

Figura. 101. Editar un resultado de aprendizaje

## Vista “Plan Tutorial” de una asignatura

En el apartado de “Plan tutorial” (Figura 102) se define una serie de campos simples para su representación:

- **Atención presencial individualizada** – En vez de usar un campo de texto con una descripción, se optó por crear una ventana emergente en que el usuario solamente tiene a seleccionar las casillas de opciones.
- **Atención presencial a grupos de trabajo** – Define la lista de opciones relacionadas con atención a grupos de trabajo.
- **Atención telefónica** - Define la lista de opciones relacionadas con atención a telefónica.
- **Atención virtual (on-line)** - Define la lista de opciones relacionadas con atención virtual.

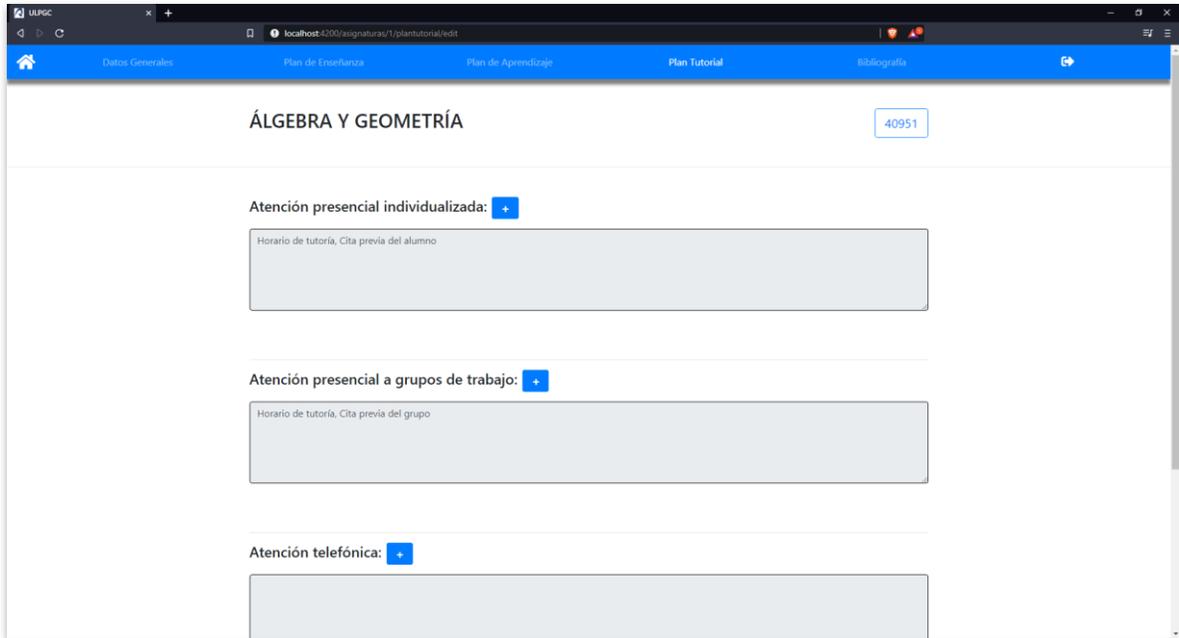


Figura. 102. Vista del Plan Tutorial (1)

Al pulsar el botón añadir, se abre una ventana emergente (Figura 103) en el que el usuario contiene casillas de selección para definir la atención individualizada.

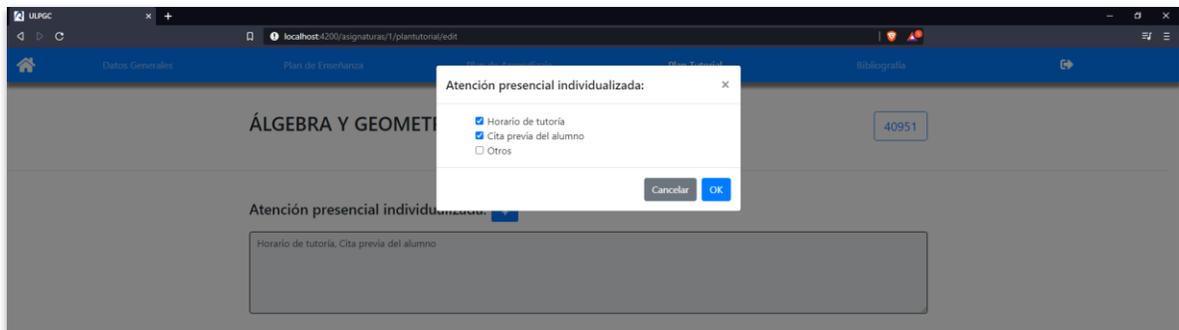


Figura. 103. Añadir una atención individualizada

Para cada apartado (atención presencial individualizada, atención presencial a grupo de trabajo, atención telefónica, atención virtual), contiene el campo “*Otros*” para definir en caso no esté en la lista de ejemplos (Figura 104).

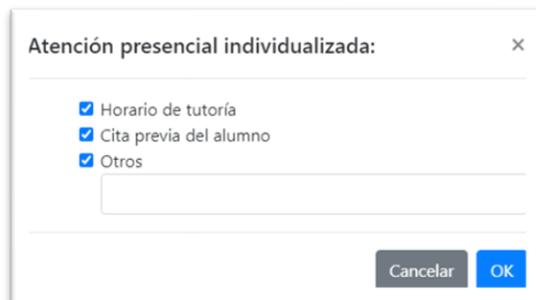


Figura. 104. Ejemplo para definir un otro campo de atención individualizada

En la Figura 105, la lista de las casillas de selección está definida de acuerdo con la atención presencial a grupos de trabajo.

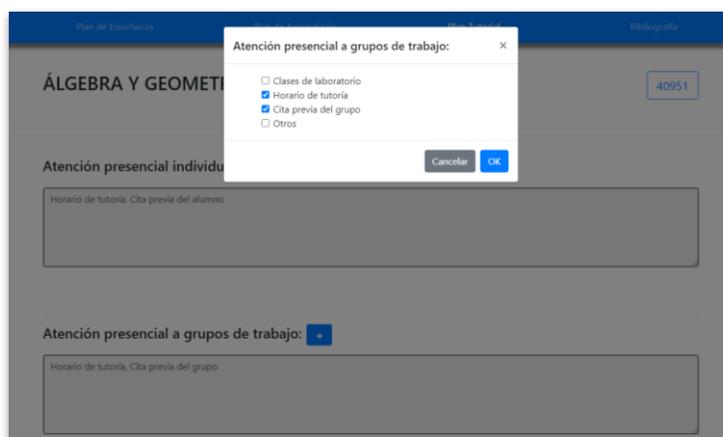


Figura. 105. Añadir una atención a grupos

La lista de las casillas de selección está definida de acuerdo con la atención telefónica (Figura 106).

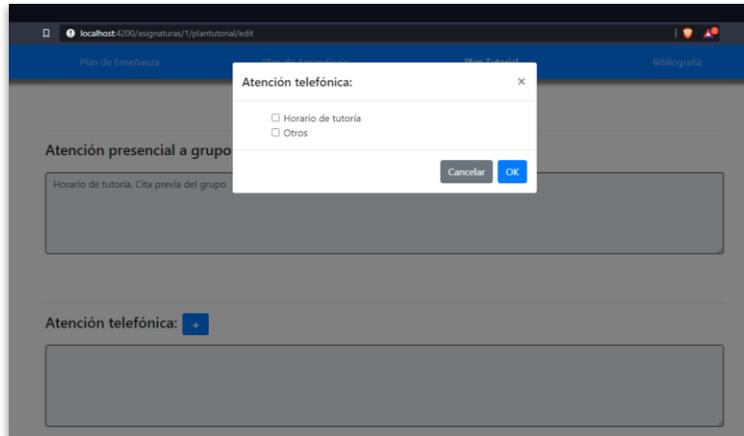


Figura. 106. Añadir una atención telefónica

La lista de las casillas de selección está definida de acuerdo con la atención virtual (Figura 107).

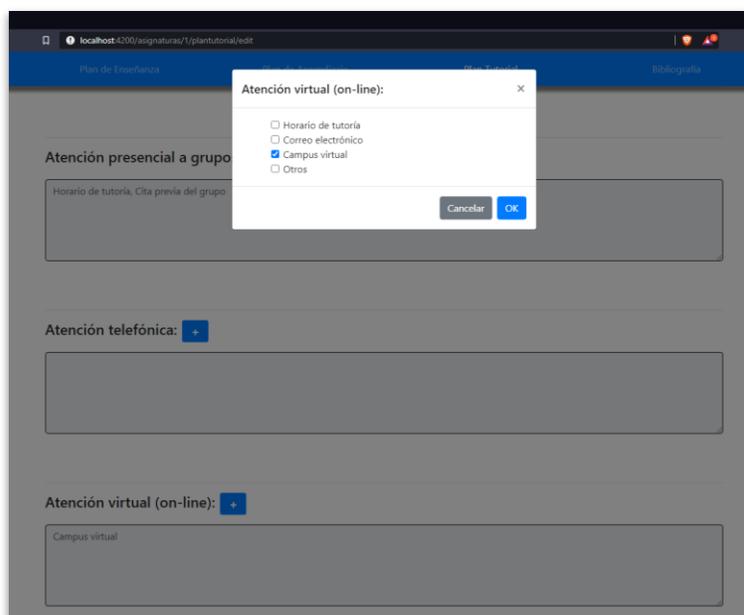


Figura. 107. Añadir una atención virtual

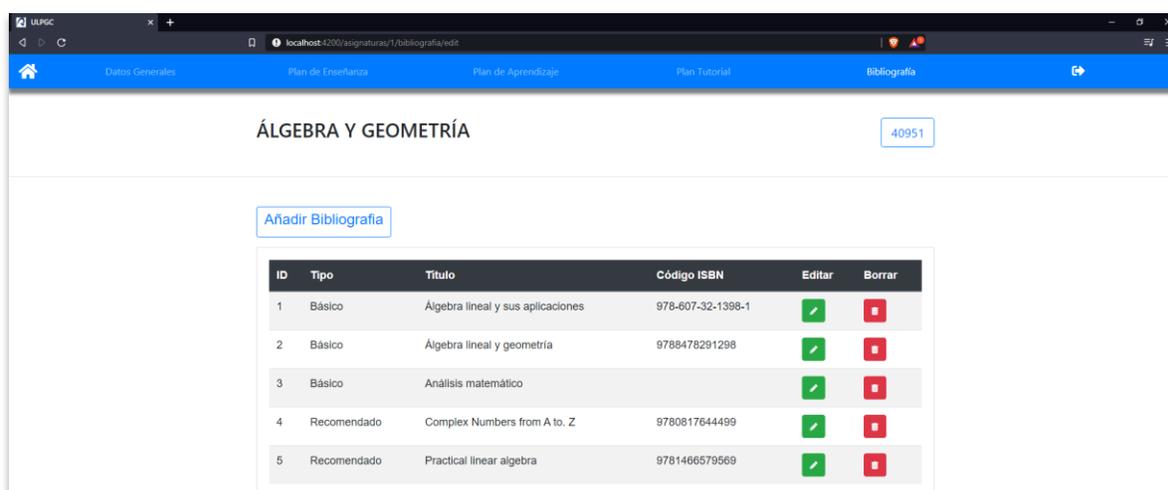
## Vista “Bibliografía” de una asignatura

El ultimo menú a definir para las asignaturas es la “Bibliografía” (Figura 108). Se compone de una simple tabla con los datos necesarios para identificar la bibliografía de una asignatura:

**1. Tipo** – tipo del libro. El usuario debe introducir si un tipo **básico** o **recomendado**.

**2. Título** – El usuario debe introducir el título del libro.

**3. ISBN** – De las siglas en ingles ISBN (International Standard Book Number) es un identificador numérico único para libros comerciales. Por ejemplo, un libro electrónico, un folleto. El ISBN tiene 10 dígitos si se asigna antes de 2007 y 13 dígitos si se asigna después del 1 de enero de 2007. Un ISBN de 13 dígitos se puede dividir en: elemento de prefijo, grupo de registro, registro, publicación y dígito de control.



ID	Tipo	Título	Código ISBN	Editar	Borrar
1	Básico	Álgebra lineal y sus aplicaciones	978-607-32-1398-1		
2	Básico	Álgebra lineal y geometría	9788478291298		
3	Básico	Análisis matemático			
4	Recomendado	Complex Numbers from A to Z	9780817644499		
5	Recomendado	Practical linear algebra	9781466579569		

Figura. 108. Vista de Bibliografía

Para añadir una nueva bibliografía (botón añadir) se abre una ventana emergente (Figura 109) en el que el usuario define los campos arriba mencionados. Al rellenar todos los campos y pulsar “OK”, los datos se guardan en la base datos y automáticamente se reflejan en la tabla con el nuevo registro.

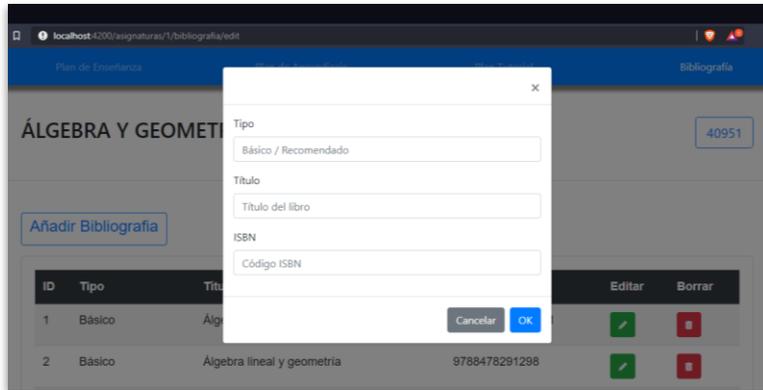


Figura. 109. Añadir una bibliografía

Al pulsar el botón del icono editar (Figura 110) se abre una ventana emergente que le permite al usuario modificar los datos de la tabla (Figura 111).

[Añadir Bibliografía](#)

ID	Tipo	Título	Código ISBN	Editar	Borrar
1	Básico	Álgebra lineal y sus aplicaciones	978-607-32-1398-1		
2	Básico	Álgebra lineal y geometría	9788478291298		
3	Básico	Análisis matemático			
4	Recomendado	Complex Numbers from A to Z	9780817644499		
5	Recomendado	Practical linear algebra	9781466579569		

Figura. 110. Editar una bibliografía

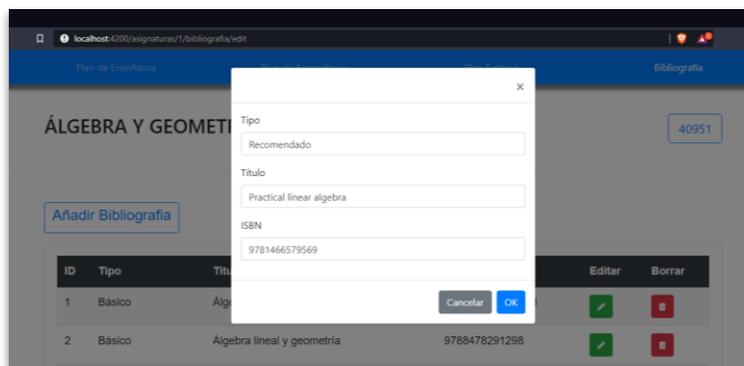


Figura. 111. Ventana emergente para editar una bibliografía

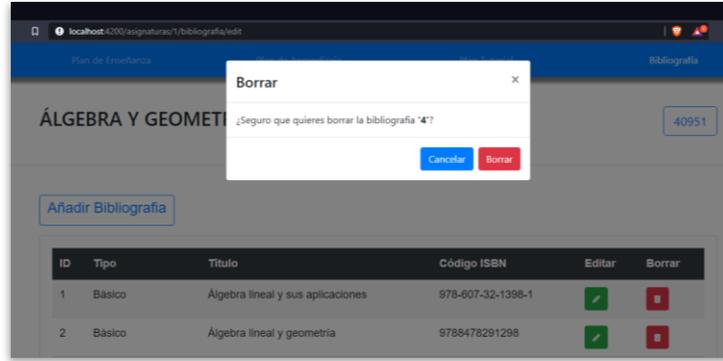


Figura. 112. Ejemplo Borrar una bibliografía

## Redireccionar a la pantalla asignaturas

Al pulsar el icono de Home (Figura 113), la aplicación se redirecciona a la página principal con la lista de todas las asignaturas. Pantalla que contiene todas las listas de asignaturas explicada en apartados anteriores.



Figura. 113. Botón para ir a la pantalla general de asignaturas



## Log Out

Para desconectar del sistema el usuario tan sólo tiene que pulsar el botón de cerrar sesión presente en el menú (Figura 114). Al pulsar el botón la aplicación envía una petición al *Firebase* (autenticación) para hacer el “*Log Out*”. En respuesta si el token es correcto *Firebase* desconecta el usuario del sistema en la página es redireccionada a la página inicial de *Login*.

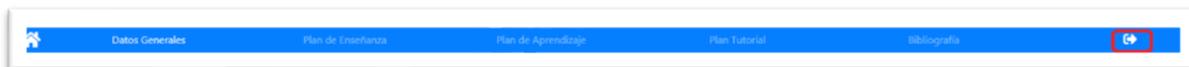


Figura. 114. Botón para cerrar sesión

## 16. Bibliografía

- [1] Documentación oficial Bootstrap <https://getbootstrap.com/docs/4.0/getting-started/introduction/>
- [2] Documentación oficial de Angular <https://angular.io/docs>
- [3] Documentación oficial de Typescript <https://www.typescriptlang.org/docs/>
- [4] Documentación oficial de Firebase <https://firebase.google.com/docs?hl=es>
- [5] Librería oficial del AngularFire <https://github.com/angular/angularfire>
- [6] Documentación de W3schools para HTML <https://www.w3schools.com/html/default.asp>
- [7] Documentación de W3schools para CSS <https://www.w3schools.com/css/default.asp>
- [8] Documentación oficial de la librería RxJs <https://www.learnrxjs.io/#content>
- [9] Wikipedia Enciclopedia <http://www.wikipedia.org/>
- [10] Documentación oficial de iconos Font Awesome:  
<https://fontawesome.com/icons?d=gallery&p=2>