

Diseño e implementación de un plugin de conexión de Google Calendar con edCrumble

Acedo Vázquez, Pablo

Curs 2020-2021

Directora: Laia Albó

GRAU EN ENGINYERIA DE SISTEMES AUDIOVISUALS

UPF

Trabajo de Fin de Grado

Diseño e implementación de un plugin de conexión de Google Calendar con edCrumble

Pablo Acedo Vázquez

Directora: Laia Albó

Grado en Ingeniería de Sistemas Audiovisuales

Agradecimientos

En primer lugar, quiero agradecer a Laia su disposición y ayuda desde el primer momento de este proyecto. También estoy agradecido a mi familia, por estar siempre presente, y a los compañeros más cercanos que me han acompañado durante estos años.

Resumen

Las herramientas de *Learnign Design* ayudan a los docentes a crear mejores experiencias para los alumnos durante los cursos académicos. *edCrumble* pertenece a este tipo de herramientas. La falta de conexión de esta con aplicaciones más populares entre los docentes, hace que no sea muy utilizada en el día a día de estos. Con el propósito de incrementar su *usabilidad* y ayudar a que sus usuarios tiendan a incorporarla en sus métodos de forma más frecuente, se ha procedido a diseñar e implementar un plugin para esta plataforma, el cual la conecta con una aplicación muy utilizada en docencia, *Google Calendar*. Este permite exportar e importar actividades entre ambas plataformas, utilizando un diseño que garantiza una experiencia de usuario sencilla e intuitiva. Se ha procedido a evaluar la herramienta desarrollada utilizando cuestionarios de *usabilidad*. Los resultados obtenidos han sido positivos, aunque la integración de la herramienta con la plataforma no ha sido posible, hecho que no permite probar con la mayor certeza, que se haya aumentado el grado de *usabilidad* de la plataforma principal, *edCrumble*.

Abstract

The *Learnign Design* tools help teachers create better experiences for students during academic courses. *edCrumble* belongs to this type of tool. Its lack of connection with more popular applications among teachers makes it not widely used in their day-to-day work. In order to increase its usability and help users tend to incorporate it into their methods more frequently, we have proceeded to design and implement a plugin for this platform, which connects it with a widely used application in teaching, Google Calendar. This allows exporting and importing activities between both platforms, using a design that guarantees a simple and intuitive user experience. The developed tool has been evaluated using usability questionnaires. The results obtained have been positive, although the integration of the tool with the platform has not been possible, a fact that does not allow us to prove with the greatest certainty that the degree of usability of the main platform, *edCrumble*, has been increased.

Resum

Les eines de *Learnign Design* ajuden els docents a crear millors experiències per als alumnes durant els cursos acadèmics. *edCrumble* pertany a aquest tipus d'eines. La manca de connexió d'aquesta amb aplicacions més populars entre els docents, fa que no sigui molt utilitzada en el dia a dia d'aquests. Amb el propòsit d'incrementar la seva *usabilitat* i ajudar a que els seus usuaris tendeixin a incorporar-la en els seus mètodes de forma més freqüent, s'ha procedit a dissenyar i implementar un

plugin per a aquesta plataforma, el qual la connecta amb una aplicació molt utilitzada en docència, *Google Calendar*. Aquest permet exportar i importar activitats entre les dues plataformes, utilitzant un disseny que garanteix una experiència d'usuari senzilla i intuïtiva. S'ha procedit a avaluar l'eina desenvolupada utilitzant qüestionaris d'*usabilitat*. Els resultats obtinguts han estat positius, tot i que la integració de l'eina amb la plataforma no ha estat possible, fet que no permet provar amb la major certesa, que s'hagi augmentat el grau d'*usabilitat* de la plataforma principal, *edCrumble*.

Índice

1. Introducción	7
2. Contexto	8
2.1. Aplicación de las TIC en la educación	8
2.2. Diseño de cursos académicos	8
2.3. Análisis de herramientas de <i>Learning Design</i>	10
2.4. Beneficios de incorporar <i>Google Calendar</i>	12
3. Planificación y fases	14
3.1. Diagrama de Gantt	14
3.1.1. Fases del diagrama	15
4. Análisis de Requerimientos	20
4.1. Requerimientos funcionales	20
4.2. Requerimientos no funcionales	21
5. Diseño del sistema	22
5.1. Casos de uso	22
5.2. Interfaz de usuario	24
5.3. Lista de clases y métodos	26
5.4. Formato de actividad	28
6. Implementación	29
6.1. Tecnologías utilizadas	29
6.2. Conexión y utilización de la <i>API</i> de <i>Google Calendar</i>	29
6.3. Conexión con edCrumble	32
6.4. Estructura del formato de evento	34
6.5. Estructura del código	37
7. Evaluación	45
7.1. Plan de prueba de evaluación	45
7.1.1. Características principales	45
7.1.2. Declaración de consentimiento	46
7.1.3. Datos demográficos y conocimientos	46
7.1.4. Cuestionarios de usabilidad	46
7.1.5. Sugerencias de mejora	49
7.2. Participación	49
7.3. Resultados	53
7.3.1. Cuestionarios de usabilidad	53
7.3.2. Sugerencias de mejora	56
7.3.3. Análisis DAFO	56

8. Conclusiones	58
8.1. Conclusiones generales	58
8.2. Limitaciones y trabajo futuro	59
8.3. Valoración personal	60

Índice de figuras

1.	<i>edCrumble</i> , interfaz	12
2.	Diagrama de Gantt, lista de fases y tareas.	14
3.	Diagrama de Gantt, estudio inicial.	15
4.	Diagrama de Gantt, diseño.	16
5.	Diagrama de Gantt, implementación: Fase I.	17
6.	Diagrama de Gantt, implementación: Fase II.	18
7.	Diagrama de Gantt, implementación: Fase III.	19
8.	Caso de uso: Iniciar sesión	22
9.	Caso de uso: Exportar	23
10.	Caso de uso: Importar	24
11.	Interfaz: Exportar	24
12.	Interfaz: Importar	25
13.	Diagrama de clases	26
14.	Título de evento con etiqueta	35
15.	Evento con la configuración descrita	37
16.	mapper: type	42
17.	Género de los encuestados	50
18.	Franja de edad de los encuestados	51
19.	Experiencia en docencia de los encuestados	52
20.	Experiencia de <i>edCrumble</i> de los encuestados	53
21.	Resultados SUS, Exportador	54
22.	Resultados SUS, Importador	55

1. Introducción

En esta sección, se explica el motivo de este trabajo de fin de grado, además de las expectativas que se tienen de él y los objetivos a cumplir.

Este trabajo de fin de grado surge a partir de la posibilidad de ampliación de funcionalidades en la plataforma de *Learning Design* (LD), diseñada e implementada en la Universitat Pompeu Fabra, *edCrumble*¹. La funcionalidad tratada en este TFG, es un plugin que conecta la plataforma con *Google Calendar*.

El contexto de este trabajo, surge a partir de diferentes estudios y cuestionarios realizados a docentes durante el proceso de desarrollo de la plataforma *edCrumble*. En los resultados de dichos estudios, se encontró que la plataforma tenía ciertas limitaciones que, en algunos casos, provocaba que no fuese una herramienta que los usuarios utilizarían en su día a día [12]. Una de las limitaciones es, precisamente, la falta de conexión de dicha herramienta con otras más populares y utilizadas a diario por el perfil de usuario encuestado. En este caso, este trabajo se centra, concretamente, en lo denominado "the *Google apps effect*"[27], que indica que una conexión con una herramienta como *Google Calendar* aumentaría el atractivo de la plataforma. Por ello, el principal motivo y objetivo de este proyecto, es diseñar e implementar dicha herramienta de conexión, la cual podría ayudar a que *edCrumble* tuviese una mayor *usabilidad*. Esta conexión se haría, concretamente, para poder exportar e importar información entre ambas plataformas.

Además del objetivo principal, en este trabajo también se pretende cumplir otros objetivos secundarios, estos se pueden apreciar a continuación.

- Aprender a desarrollar y gestionar un trabajo académico de larga duración.
- Poner en práctica conocimientos y aptitudes adquiridas durante el grado universitario.
- Aprender sobre tecnologías TIC aplicadas en la educación y la aplicación de *Learning Design*.
- Diseñar y desarrollar una herramienta que sea totalmente operativa.
- Que la herramienta desarrollada sea apta para su uso en el contexto de la plataforma, así como para sus usuarios, sin ser su uso, excesivamente complejo.

Una vez introducido el propósito de este trabajo, se continuará con el contexto para entender las bases de este.

¹<https://ilde2.upf.edu/edcrumble/>

2. Contexto

En esta sección se introduce el entorno de la aplicación a diseñar y desarrollar. Este comprende la aplicación de las tecnologías de la información y comunicación en el ámbito educacional, así como el concepto de *Learning Design* y el uso de *Google Calendar* dentro del diseño de cursos académicos. Por lo que también se da importancia al uso del tiempo en estos diseños.

2.1. Aplicación de las TIC en la educación

Las tecnologías de la información y comunicación (TIC) han ido y siguen cambiando la forma y el ritmo de vida de la sociedad conforme evolucionan. Hoy en día, gracias a estas tecnologías, tenemos diversos tipos de herramientas a nuestro alcance para satisfacer necesidades como la comunicación, el trabajo, la diversión, comprar y aprender.

Obviamente, todo esto, ha afectado también a los sistemas educativos de muchos países. Siendo una gran parte de estas herramientas muy recurrentes, incluso clave, para la educación moderna en todos los niveles, desde parvulario hasta la universidad. Cabe destacar también, que la integración de todas estas tecnologías en los sistemas educativos es responsabilidad de los gobernantes y dicha integración conlleva diferentes dificultades, tales como la formación del personal docente, carencia de personal y falta de equipos tecnológicos [1].

Sin ir más lejos, la Universidad Pompeu Fabra (UPF) es un ejemplo de universidad que ha integrado estas tecnologías en la educación. La UPF utiliza las TIC en la gestión de los sistemas de expedientes académicos; la creación de material educativo; tiene un campus virtual donde el alumnado accede al material de los cursos; se utilizan herramientas y aplicaciones específicas según la necesidad de los diferentes cursos y ámbitos de investigación; se utiliza las TIC para los sistemas de comunicación entre alumnado y profesorado.

En el marco de este trabajo de fin de grado, nos centraremos concretamente en tecnologías educativas que dan apoyo a los docentes a la hora de la preparación de sus clases y en el diseño de cursos académicos. Por ello, es importante conocer diferentes definiciones presentes en este contexto, así como diferentes herramientas que existen en el mercado. En el siguiente apartado, se describen los términos involucrados.

2.2. Diseño de cursos académicos

En este apartado definimos los términos *Learning Design* (LD) e *Instructional Design* (ID), para ver que similitudes y diferencias tienen.

Con diseño instruccional (*Instructional Design*) [2], [3] nos podemos referir a un tipo de diseño de transmisión de conocimiento basado en seguir unas instrucciones. Es decir, en este tipo de diseño se analizan las necesidades de aprendizaje y a partir de estas se desarrollan unas instrucciones a seguir. En teoría, si estas se siguen correctamente se conseguirá una transmisión efectiva de conocimientos y aptitudes [2].

Por otro lado, diseño de aprendizaje (*Learning Design*) [2], [3] es un término con el que nos referimos a una estructura o patrón de trabajo para experiencias educativas o de aprendizaje. En este se elige qué, cómo y cuándo enseñar algo, es decir, elegir un modelo pedagógico adecuado para cada situación, el contenido de este, las tareas a realizar, la estructura y secuencia de estas actividades, así como las herramientas y recursos más adecuados para realizar las tareas planteadas, todo esto definido cronológicamente con el objetivo de conseguir un proceso de aprendizaje efectivo [3], [4]. Cabe destacar que está ampliamente extendido dentro del mundo de *e-learning*.

A partir de estas definiciones, podemos diferenciar los dos términos afirmando que, mientras *Instructional Design* está más enfocado a el “cómo” se puede enseñar algo y, por tanto, al profesor, *Learning Design* se enfoca en encontrar la mejor forma para que el estudiante aprenda material de un cierto ámbito [5].

Con *Learning Design* se pretende ir más allá de las instrucciones o entregas pautadas, todo ello pensando primero en una nueva generación de estudiantes acostumbrada a nuevas metodologías y tecnologías, intentado crear el mejor proceso de aprendizaje para ellos. El proceso se convierte en algo más complejo y colaborativo entre estudiantes y docentes respecto a aquellos diseñados con *Instructional Design* [4].

Es verdad que existen definiciones de *Learning Design* que varían en muchos aspectos, pero siguiendo la especificación IMS LD, donde tenemos una descripción más estandarizada de los procesos educativos, tenemos que se desarrolla para cumplir ciertos requerimientos [6], [7], [8]:

- Describir totalmente el proceso de enseñanza y aprendizaje en una unidad de aprendizaje, incluyendo todos aquellos elementos utilizados durante el proceso de enseñanza.
- Deberá expresar todo el significado pedagógico y funcionalidad, así como todos los elementos dentro del contexto de LD.
- Describir aspectos de personalización dentro de LD, tal que los elementos se puedan adaptar a conocimientos previos y circunstancias de los estudiantes.

- Ser compatibles con la introducción de otros estándares siempre que sea posible.
- Que tenga reusabilidad, es decir que se puedan identificar, descontextualizar e intercambiar objetos de aprendizaje útiles, incluso en otros conceptos.
- Deberá proporcionar un lenguaje formal, el cual se pueda procesar automáticamente.
- Que sea reproducible, que el *Learning Design* se describa de una forma abstracta tal que se pueda repetir en varias condiciones y diferentes usuarios.

Llegados a este punto, entendiendo mejor el concepto de LD, podemos poner más énfasis en el uso y la importancia del tiempo dentro de este modelo de diseño. Obviamente, un curso o proceso de aprendizaje lleva un tiempo determinado, no se puede hacer todo a la vez, ya que se necesita tiempo para trabajar y asimilar conceptos y conocimientos. Además, normalmente, para aprender ciertos conocimientos nuevos, se necesitan haber asimilado otros. Por ello, la mayoría de docentes planean o diseñan sus cursos con diferentes herramientas basadas en la temporización, por lo que podemos considerar que la secuencia de aprendizaje está basada en el tiempo [4], [13], [14].

Por ello, consideramos que el tiempo es un factor fundamental dentro de *Learning Design* [13], [14]. A la hora del diseño, se deberá tener en cuenta la cantidad de tiempo disponible para realizar cierto curso, es decir, las actividades a realizar deberán estar espaciadas dentro de una cierta secuencia de tiempo, por ejemplo un trimestre, donde esta se ha decidido en base a los conceptos que se deben introducir antes que otros. Además, el docente también deberá tener en cuenta cuánto tiempo se le deberá dedicar a su curso en horas de trabajo. Siguiendo el ejemplo de las universidades, tenemos que la duración o tiempo que el alumnado debe dedicar a una asignatura está estipulado por el sistema de créditos, y a partir de este dedicar qué duración y peso tendrá cada actividad o tarea a realizar en dicho proceso de aprendizaje.

2.3. Análisis de herramientas de *Learning Design*

Llegados a este punto, podemos hacer un breve análisis de diferentes herramientas de *Learning Design* utilizadas hoy en día, focalizado principalmente en el uso que hacen estas del tiempo.

- **The Learning Designer** [9], [14]: Desarrollado por un equipo liderado por Diana Laurillard, es una herramienta que te permite planificar una sesión, dentro de la cual puedes añadir todas las actividades a desarrollar

en esta, con todos los detalles necesarios, tales como la duración de la sesión, duración de las tareas, descripción de las tareas y formato de estas. Todos los diseños que se realizan se pueden compartir con la comunidad, además de poder exportar la sesión a un documento de texto. Solo permite el diseño de una sesión, donde las actividades se ajustan al tiempo de esta, cada una con una cierta duración. Las sesiones están secuenciadas linealmente pero no encontramos una línea temporal explícita entre sesiones.

- **The Learning Activity Management System (LAMS)** [10], [13]: Herramienta que tiene diferentes modalidades, en este caso nos centraremos en la de autor, que es con la que el docente puede diseñar un curso y sus sesiones. No basan la planificación directamente en el tiempo, sino en actividades que se pueden unir gráficamente entre sí para formar una sesión, actividades que están cronológicamente ordenadas, el profesor puede elegir cuánto tiempo se dedica a la sesión y cómo se reparte este tiempo. La secuenciación del curso se realiza en una lista de sesiones, pero no sobre una línea temporal.
- **Integrated Learning Environment (ILDE)** [11]: Herramienta creada por el grupo TIDE de la UPF. Ideada para crear, co-crear y compartir soluciones de *Learning Design*, integra diferentes herramientas de diseño. En él se puede trabajar el concepto de un curso en una macro-escala, que comprendería una lista conceptual de actividades y objetivos de dicho curso, además de poder diseñar sesiones o actividades de aprendizaje de una forma detallada, incluyendo los recursos y herramientas necesarias para realizarlas. Se pueden conectar estos diseños a plataformas de *e-learning* como *Moodle*, compartir con la comunidad todos los diseños realizados y dar *feedback* a diseños de otros usuarios. Todos los diseños se almacenan en una lista que se puede visualizar en la cuenta de cada usuario, no se encuentra una secuencia temporal clara entre sesiones, pero sí entre actividades/tareas dentro de una sesión.
- **edCrumble** [4], [12]: Herramienta de Learning Design con una interfaz gráfica intuitiva, en la cual se centra este trabajo, también creada por TIDE de la UPF. Permite diseñar cursos siguiendo LD, apoyado por el análisis de datos. Todos los diseños pueden compartirse a través de la comunidad, igual que utilizar, versionar, mejorar y dar *feedback* a proyectos iniciados por otras personas. Al crear un curso, se debe especificar la duración y existe la posibilidad de incluir información de este en macro-escala. Todo el curso se planifica en una línea de tiempo o *timeline*, donde se muestran todos los días y horas dentro del periodo de duración establecido. Tenemos una herramienta que da mucha importancia a este aspecto ya que todas las sesiones o actividades quedarán claramente grabadas en dicha secuencia en un detallado intervalo de tiempo. En el

timeline tenemos dos opciones de tipo de actividades, aquellas que se realizan en el centro de estudios y aquellas actividades complementarias al proceso de aprendizaje fuera del centro. En ambas opciones podemos crear las sesiones o actividades donde podremos especificar la duración de esta, diferentes objetivos, modalidades y tareas. Cada tarea dentro de la sesión tendrá una duración específica, la suma de la duración de todas las tareas deberá ser la de la duración de la actividad o sesión, también se podrán especificar todos recursos asociados a estas tareas, incluso, utilizando diferentes herramientas dentro de *edCrumble*. Dichas tareas siguen un formato concreto, en el que se puede escoger diferentes características que la describen. Todos los diseños y datos de estos se pueden exportar en diferentes documentos. A continuación, se puede apreciar la interfaz del modo de edición de proyecto de la plataforma, en la que hay una actividad abierta, configurada con diferentes tareas.

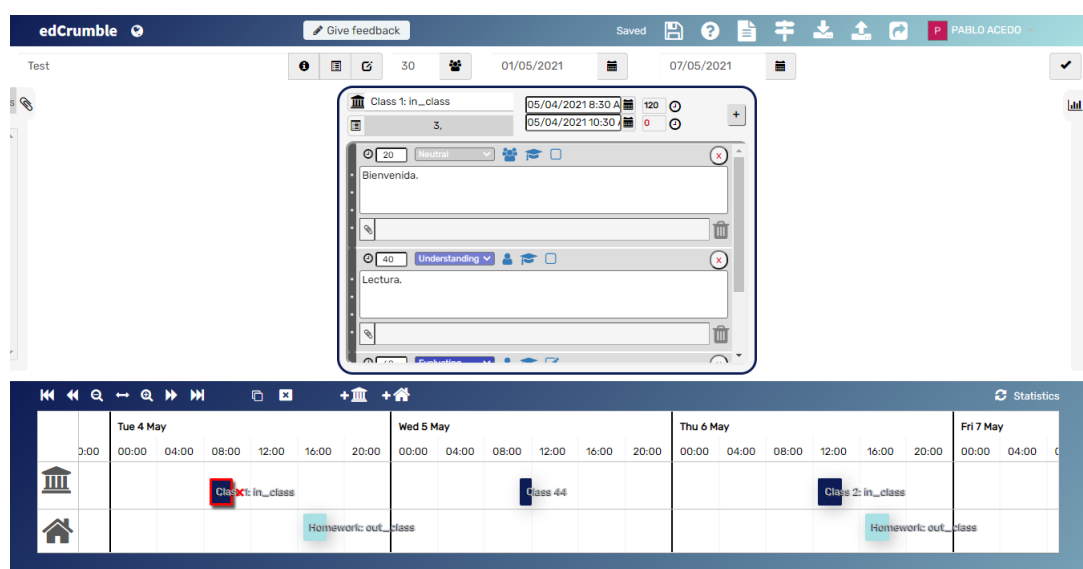


Figura 1: *edCrumble*, interfaz

2.4. Beneficios de incorporar *Google Calendar*

Dentro de las herramientas de *Learning Design* podemos encontrar diferentes problemas de *usabilidad*, ya que no hay mucha conexión con herramientas que normalmente utiliza el personal docente para el diseño de cursos o procesos académicos.

En el caso de *edCrumble*, tal y como se puede apreciar en [12] y [27], se obtiene que uno de los problemas que apuntan diversos profesores que han probado la herramienta, es su falta de conexión con las herramientas externas de *Google*. Este hecho muestra que la *usabilidad* de la herramienta podría mejorar notablemente añadiendo vínculos con las aplicaciones de

Google más utilizadas por los docentes, ya que estos tienden a utilizar estas aplicaciones visualmente atractivas debido a su popularidad, *usabilidad* y grandes posibilidades, destacando que se han convertido en un tipo de estándar, ya que se utilizan en la mayoría de aspectos académicos.

Tal y como se ha señalado anteriormente, *edCrumble* es una herramienta donde todo el planteamiento está basado en un *timeline*, por ello proponemos crear un plugin que conecte la tan utilizada *Google Calendar* con la herramienta de LD. Dicho plugin sería capaz de convertir las sesiones, o grupo de sesiones, creadas en *Google Calendar* a *edCrumble* y al revés. Es decir, plasmar en el *timeline* de la herramienta de *Learning Design*, eventos creados en la popular aplicación de *Google*. Con ello se pretenden los siguientes beneficios:

- Por una parte, conectamos a *edCrumble* una herramienta que podemos considerar clave en el día a día de muchas personas y de los profesores[12], [27]. Con esto facilitamos la conexión entre la herramienta de *Google* y la de LD, haciendo más cómoda la experiencia.
- Según el profesorado que ha utilizado *edCrumble*, el sistema de creación de sesiones y tareas dentro de la línea de tiempo no es del todo ágil, por lo que al crearlos emplean más de tiempo de lo normal en crear estas actividades [4]. Tener la posibilidad de importar dichas actividades directamente de *Calendar*, haría que el proceso de crear un curso dentro de *edCrumble* fuese mucho más rápido y cómodo para los usuarios.
- Se aumentaría la *usabilidad* de *edCrumble*, haciendo que fuese una herramienta más atractiva para docentes acostumbrados a utilizar otros métodos a los que se han acomodado, aunque estos sean menos efectivos para crear modelos de LD. Por lo que la integración del plugin sería un paso más para un posible aumento de usuarios.

3. Planificación y fases

Para planificar las fases del proyecto se ha utilizado un diagrama de Gantt. Gracias a esto se ha obtenido una representación gráfica en la que se pueden distinguir las distintas fases y tareas del proyecto espaciadas en el tiempo, así como el tiempo dedicado a cada una de ellas. El hecho de planificar por fases el proyecto ha sido clave para poder realizar una buen diseño y una buena implementación, dado que cada fase ha sido necesaria para poder avanzar hacia la siguiente. Cabe destacar que el diagrama es una guía temporal sobre las fases del proyecto, pero en la práctica, durante la realización de este, aunque se ha respetado esta guía, ha habido momentos en los que, por necesidad, se han realizado algunas fases a la vez y se ha ido retrocediendo y avanzando de fase según las circunstancias.

3.1. Diagrama de Gantt

A continuación, se pueden apreciar las fases y tareas del diagrama, donde, en color azul tenemos las diferentes fases del proyecto, y en naranja las tareas dentro de cada fase.

Actividades/Fecha(mensual y semanal)
Estudio inicial
TIC aplicadas a educación
Diseño de cursos academicos
Estudio sobre herramientas de Learning Design
Familiarización con edCrumble
Beneficios de incorporar G.C. a edCrumble
Diseño del plugin
Requisitos/Requerimientos
Diagrama de clases
Implementación del plugin
Fase I
Lectura y escritura de datos (JSON)
Exportador (edCrumble a G.C.)
Fase II
Importador (G.C a edCrumble)
Fase III
Test de usabilidad
Redacción de la memoria
Preparación de la defensa

Figura 2: Diagrama de Gantt, lista de fases y tareas.

El período de tiempo en el que se ha realizado este proyecto es el comprendido entre la primera semana de septiembre del año dos mil veinte, hasta el día de la entrega del trabajo, el dieciocho de junio de dos mil veintiuno. A continuación, se explicarán todas las fases y tareas dentro de esta, así como el período cronológico que han ocupado en la realización del trabajo, utilizando

el mes y número de semana dentro de este. Todas ellas vendrán acompañadas de su representación gráfica dentro de este diagrama.

3.1.1. Fases del diagrama

- Estudio inicial

Actividades/Fecha(mensual y semanal)	September				October		
	1	2	3	4	1	2	3
Estudio inicial	[Barra azul que cubre todas las celdas de esta fila]						
TIC aplicadas a educación		[Barra naranja]					
Diseño de cursos académicos				[Barra naranja]			
Estudio sobre herramientas de Learning Design				[Barra naranja]			
Familiarización con edCrumble					[Barra naranja]		
Beneficios de incorporar G.C. a edCrumble					[Barra naranja]		
Diseño del plugin							
Requisitos/Requerimientos							
Diagrama de clases							
Implementación del plugin							
Fase I							
Lectura y escritura de datos (JSON)							
Exportador (edCrumble a G.C.)							
Fase II							
Importador (G.C a edCrumble)							
Fase III							
Test de usabilidad							

Figura 3: Diagrama de Gantt, estudio inicial.

En esta fase, comprendida entre la primera semana de septiembre de dos mil veinte, hasta la tercera semana de octubre del mismo año, para conocer el contexto de la plataforma de LD para la que se va a desarrollar la herramienta, además de para entender mejor el motivo por el que se diseña el plugin, se realiza un estudio sobre el entorno de desarrollo. En este se realiza una búsqueda de diferentes artículos académicos e información en diferentes páginas web. Empezando por las TIC aplicadas a educación, estudio con dos semanas de duración, para entender como se aplican dichas tecnologías en este ámbito, en las siguientes dos semanas se investiga sobre el diseño de cursos académicos y diversas herramientas de *Learning Design*. Finalmente, las últimas dos semanas de esta fase inicial, después de haber entendido el concepto de LD y cómo se aplican las TIC en la educación, se dedican a la familiarización con la plataforma principal *edCrumble* y se realiza un estudio sobre los beneficios de incorporar un plugin de *Calendar* a esta.

- **Diseño del plugin**

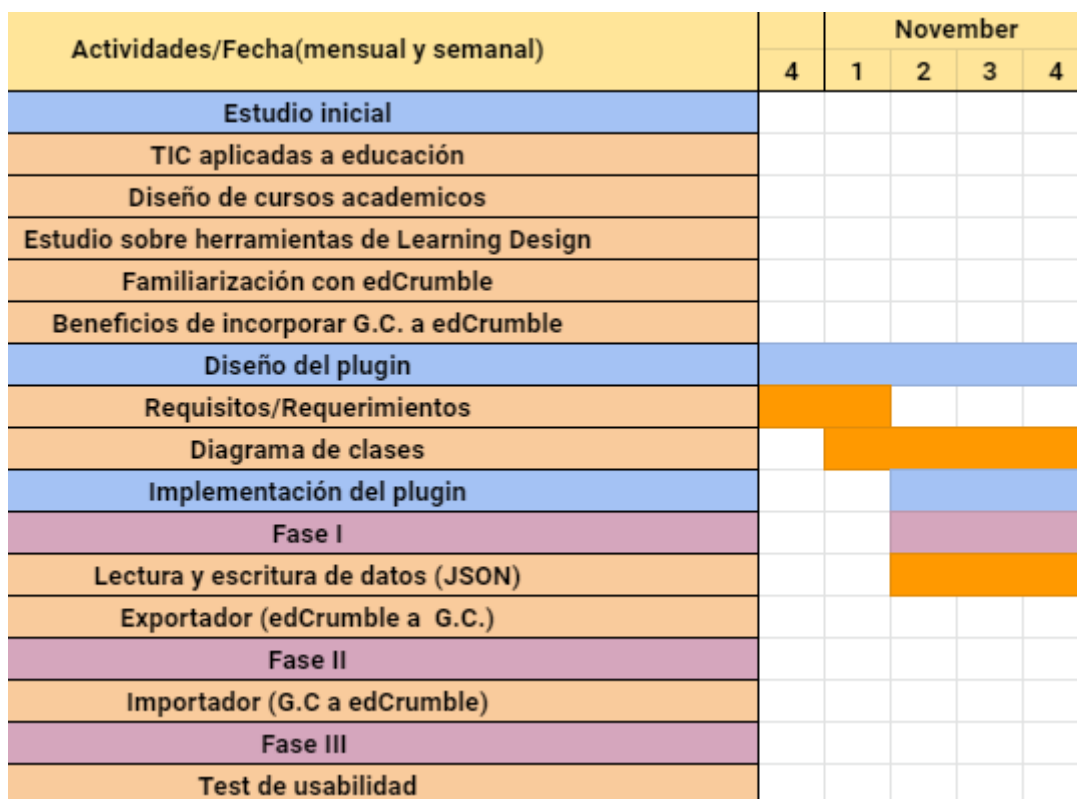


Figura 4: Diagrama de Gantt, diseño.

Tras el estudio inicial, en el período comprendido entre la última semana de octubre de dos mil veinte y todo el mes de noviembre del mismo año, se realiza el diseño del plugin. Entendiendo las necesidades de este y con varias entrevistas con la tutora de este trabajo y responsable de la plataforma de LD, durante dos semanas se discuten y extraen los requerimientos. Una vez extraídos, se procede, durante todo el mes de noviembre, a diseñar el diagrama de clases de la herramienta, la mayoría de semanas en paralelo con la primera fase de implementación, la cual ayuda a definir las clases, métodos y necesidades. Cabe destacar que la fase de diseño del plugin, aunque formalmente dura cinco semanas, está presente también durante la fase de desarrollo o implementación, ya que en esta se encuentran también varios nuevos requerimientos que enriquecen o modifican los ya establecidos, además de modificar ligeramente el diagrama de clases final.

- **Implementación del plugin**

Comprendida entre la segunda semana de noviembre del año dos mil veinte y la segunda semana de junio de dos mil veintiuno, es una de

las fases más largas del trabajo. En ella se buscan y encuentran todas las soluciones para traducir el diseño, en una herramienta funcional. Esta está dividida en tres fases más, las cuales se pueden apreciar a continuación.

- Fase I

Actividades/Fecha(mensual y semanal)	November			December				January			
	2	3	4	1	2	3	4	1	2	3	4
Estudio inicial											
TIC aplicadas a educación											
Diseño de cursos academicos											
Estudio sobre herramientas de Learning Design											
Familiarización con edCrumble											
Beneficios de incorporar G.C. a edCrumble											
Diseño del plugin											
Requisitos/Requerimientos											
Diagrama de clases											
Implementación del plugin											
Fase I											
Lectura y escritura de datos (JSON)											
Exportador (edCrumble a G.C.)											
Fase II											
Importador (G.C a edCrumble)											
Fase III											
Test de usabilidad											

Figura 5: Diagrama de Gantt, implementación: Fase I.

Se realiza entre la segunda semana de noviembre de dos mil veinte, hasta el final del mes de enero de dos mil veintiuno. Durante las tres primeras semanas, se realiza un estudio, además de la conexión, con la interfaz de programación de *Google Calendar*, en paralelo, se estudia y se realiza también, la conexión con la plataforma *edCrumble*, familiarizándose con archivos *JSON*, concretamente el archivo en el que se definen los proyectos de la plataforma. Una vez esta tarea ha sido realizada con éxito, se procede a implementar el exportador de *edCrumble* a *Calendar*. Esta tarea ocupa dos meses enteros del trabajo, en los que se sigue el diseño realizado con anterioridad, además de la familiarización con la utilización de la *API* de *Google*. Al final de esta se consigue ya exportar actividades a *Calendar* con éxito.

- Fase II

Actividades/Fecha(mensual y semanal)	January		February				March			
	3	4	1	2	3	4	1	2	3	4
Estudio inicial										
TIC aplicadas a educación										
Diseño de cursos academicos										
Estudio sobre herramientas de Learning Design										
Familiarización con edCrumble										
Beneficios de incorporar G.C. a edCrumble										
Diseño del plugin										
Requisitos/Requerimientos										
Diagrama de clases										
Implementación del plugin										
Fase I										
Lectura y escritura de datos (JSON)										
Exportador (edCrumble a G.C.)										
Fase II										
Importador (G.C a edCrumble)										
Fase III										
Test de usabilidad										

Figura 6: Diagrama de Gantt, implementación: Fase II.

Comprendida entre finales de enero de dos mil veintiuno hasta finales del mes de marzo del mismo año, con una duración de ocho semanas, se implementa el importador de la herramienta. En ella se hace realidad el diseño de un formato de texto que describe la configuración de una tarea de *edCrumble*, la cual irá en la descripción de los eventos de *Calendar*, y la lectura de este formato. Finalmente se procede a introducir en el archivo del proyecto principal de *edCrumble* los datos importados. Al final de esta fase ya se obtienen las dos funcionalidades principales del plugin. Y una primera versión funcional, la cual se seguirá perfeccionando para obtener la versión final.

- Fase III

Actividades/Fecha(mensual y semanal)	April				May				June			
	1	2	3	4	1	2	3	4	1	2	3	4
Estudio inicial												
TIC aplicadas a educación												
Diseño de cursos academicos												
Estudio sobre herramientas de Learning Design												
Familiarización con edCrumble												
Beneficios de incorporar G.C. a edCrumble												
Diseño del plugin												
Requisitos/Requerimientos												
Diagrama de clases												
Implementación del plugin												
Fase I												
Lectura y escritura de datos (JSON)												
Exportador (edCrumble a G.C.)												
Fase II												
Importador (G.C a edCrumble)												
Fase III												
Test de usabilidad												

Figura 7: Diagrama de Gantt, implementación: Fase III.

Última fase de implementación. Con una duración de ocho semanas, las comprendidas entre el inicio de abril de dos mil veintiuno hasta mediados de junio del mismo año, es una de las últimas fases del proyecto. En esta se diseña un test de usabilidad de la primera versión de la herramienta, se realiza el test con varias personas familiarizadas con *edCrumble*, y una vez se obtienen los resultados se procede a realizar un análisis de estos, además de perfeccionar la herramienta siguiendo en parte la respuesta del test. Al final de esta fase de implementación se obtiene la versión definitiva de la herramienta, la cual contiene todas las funcionalidades indicadas en la fase de diseño, además de cumplir con la estética deseada.

- **Redacción de la memoria:** En paralelo a todas las fases anteriormente descritas se ha ido redactando la memoria del proyecto. En las últimas tres semanas de este es cuando más tiempo se ha dedicado a esta fase, en la que se perfeccionan todos los aspectos para que esté apta para la entrega.
- **Preparación de la defensa:** Tras la entrega de la memoria, se empezará a trabajar en la defensa del proyecto, preparando una presentación en la que se refleje todo el trabajo realizado, con todas las partes que este contiene, así como una demostración del funcionamiento de la herramienta desarrollada.

4. Análisis de Requerimientos

Esta sección se centra en un análisis de los requerimientos del plugin. Es decir, se describen los requisitos que deberá cumplir de la forma más clara y concisa posible, que en su conjunto define que deberá hacer la herramienta.

A partir de estudios hechos previamente, en los que se hicieron cuestionarios, grupos de discusión, entrevistas y observaciones, tal y como podemos ver en [12], surgió la necesidad de conexiones de *edCrumble* con herramientas más populares entre los docentes, así como poder exportar e importar diseños de aprendizaje. Los requerimientos que se enuncian posteriormente, son fruto de las conclusiones a las que se ha llegado a partir de varias entrevistas con la directora del TFG, en las que se ha discutido tanto las necesidades anteriormente mencionadas, como la interacción con la herramienta, la cual deberá mantenerse lo más sencilla posible.

A continuación, se describe la lista de estos requerimientos, que tal y como se hace tradicionalmente en el campo de la ingeniería de software, han sido divididos entre funcionales y no funcionales [15].

4.1. Requerimientos funcionales

Con requerimientos funcionales, nos referimos simplemente a aquellos requerimientos que definen qué es lo que deberá hacer el producto. A continuación, se encuentra el listado de requerimientos de este tipo del plugin:

- El usuario podrá conectarse con su cuenta de *Google*.
- Poder escoger entre modo de exportación y modo de importación.
- Acceder al plugin a través de un icono colocado en el sector de la línea de tiempo del editor de *edCrumble*.
- Modo de exportación:
 - Poder escoger a qué calendario de *Google Calendar* exportar o tener la opción de crear uno nuevo a partir del proyecto de *edCrumble*.
 - Convertir cada actividad creada en el proyecto de *edCrumble* a un evento de *Calendar*.
 - Añadir a los eventos de *Calendar* una etiqueta que diferencie el tipo de actividad, la cual podrá ser etiquetada como fuera o dentro de clase. Esta diferencia también podrá ser visible a partir del color del evento.
 - Traducir cada tarea creada dentro de cada actividad de *edCrumble* en texto dentro de la descripción en el evento correspondiente en

Calendar. Dicho texto deberá seguir un formato en el que se refleje todas las características de la configuración de la tarea, así como el contenido de esta.

- Modo de importación:
 - Escoger desde la herramienta el calendario a importar.
 - Importar actividades sin tareas.
 - Controlar errores de formato en la descripción del evento. En caso de error en la lista de tareas o la falta de algunas características de configuración, se pondrá la configuración *por defecto* de dicha característica.

4.2. Requerimientos no funcionales

Con requerimientos no funcionales, indicamos las restricciones que existen en el sistema y su desarrollo.

- El desarrollo del plugin deberá hacerse en 4 meses dentro del periodo del curso académico.
- El plugin deberá ser compatible con la plataforma *edCrumble* y todos los navegadores en los que funciona.
- El plugin podrá utilizarse únicamente dentro de proyectos ya creados en *edCrumble*.
- La interacción deberá ser sencilla tanto en la exportación como la importación.
- La herramienta será desarrollada utilizando *HTML5*, *JavaScript* y *CSS* para ser compatible con *edCrumble*.
- Todos los datos involucrados en los procesos tanto de exportación como importación serán extraídos y añadidos en el archivo de datos *JSON* del proyecto de *edCrumble* [23].
- Se utilizará la *API* de *Google Calendar* para *JavaScript* para la conexión con la cuenta de Google del usuario.
- La estética del plugin deberá corresponder con la estética de *edCrumble*.
- La sintaxis en la descripción de los eventos de *Google Calendar*, la cual define las tareas dentro de cada actividad, deberá ser la misma tanto en eventos exportados como en eventos a importar.
- Los eventos a importar creados en *Calendar* deberán contener una serie de elementos mínimos obligatorios, los cuales serán el título y la etiqueta diferenciadora entre fuera o dentro de clase.

5. Diseño del sistema

En esta sección se describe el diseño de la herramienta, teniendo en cuenta los diferentes requerimientos extraídos en la sección anterior. En este, se tienen en cuenta los diferentes casos de uso que tendría la herramienta, que ayudarán a extraer las secciones a implementar del plugin, además de la interfaz de usuario.

5.1. Casos de uso

A continuación, se presentan los diferentes casos de uso de la herramienta, con los que se describen las actividades que deberá llevar a cabo el usuario para completar los procesos tanto de exportación como importación. En todos ellos se describe la interacción entre la persona y el sistema. Se asume que este proceso se realiza a partir de haber iniciado sesión en *edCrumble* y estando en el modo de edición de un proyecto, también se asume que el usuario tiene una cuenta de *Google*. Se separan los casos de uso en tres diferentes, inicio de sesión y otros dos para los modos de exportación e importación.

■ Iniciar sesión

1. **Actor:** El usuario abre el plugin.
2. **Sistema:** El sistema pide iniciar sesión con su cuenta de *Google*.
3. **Actor:** El usuario accede con su cuenta de *Google* ya iniciada o introduce su nombre de usuario y contraseña.

Fallo: El sistema no inicia sesión y comunica el error.

Éxito: El sistema se conecta con el *Calendar* del usuario y carga todos sus calendarios. El usuario entra automáticamente en el modo de exportación.

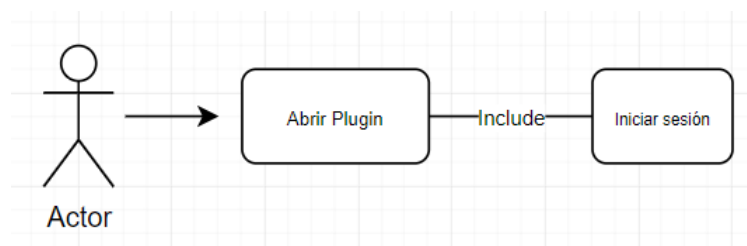


Figura 8: Caso de uso: Iniciar sesión

■ Exportar

1. **Actor:** El usuario, habiendo creado previamente diferentes actividades en la línea de tiempo de *edCrumble*, escoge a qué calendario

de **Google Calendar** quiere exportar las actividades, o crear uno nuevo a partir del nombre del proyecto.

2. **Sistema:** El sistema selecciona el calendario o lee el nombre del proyecto para crear uno nuevo.
3. **Actor:** El usuario hace clic en el botón de exportar.
4. **Sistema:** El sistema crea un calendario nuevo, en caso de tener que hacerlo. Después crea uno a uno los eventos correspondientes a las actividades creadas en *edCrumble* en el calendario correspondiente.

Fallo: Los eventos correspondientes no se crean.

Éxito: Los eventos se crean y se pueden visualizar en el calendario correspondiente en la cuenta de *Google Calendar* del usuario.

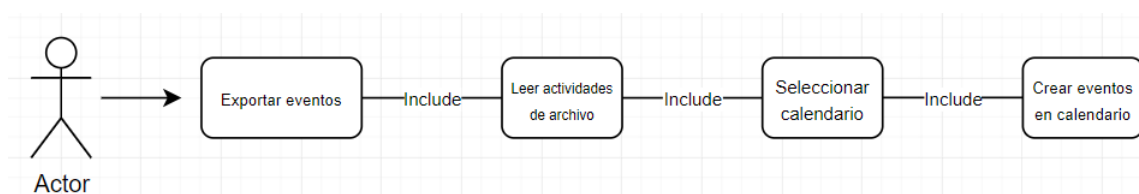


Figura 9: Caso de uso: Exportar

■ Importar

1. **Actor:** El usuario crea, en un calendario de su *Google Calendar* ya creado, los diferentes eventos que desee. En dichos eventos el título y la etiqueta de fuera o dentro de clase es obligatoria, si quiere para las tareas puede seguir la plantilla proporcionada en el plugin. Una vez creados, el usuario escoge en el plugin de *edCrumble* el calendario a importar.
2. **Sistema:** El sistema selecciona el calendario escogido.
3. **Actor:** El usuario hace clic en el botón de importar.
4. **Sistema:** El sistema crea una a una las actividades dentro del *JSON* de *edCrumble* a partir de los eventos del calendario.

Fallo: El sistema no crea las actividades en caso de no tener en el evento de *Calendar* los datos obligatorios. En el caso de fallo en el formato de tareas, el sistema las configura por defecto.

Éxito: Las actividades se crean correctamente en la línea de tiempo del proyecto de *edCrumble* y se pueden visualizar y modificar.

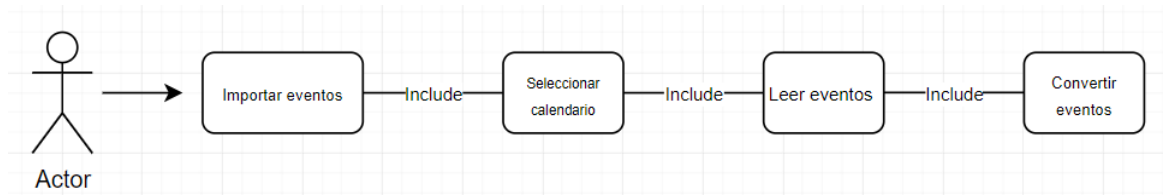


Figura 10: Caso de uso: Importar

5.2. Interfaz de usuario

En este apartado, se explica cómo es la interfaz con la que el usuario tendrá que interactuar. Las imágenes mostradas a continuación son ya la interfaz definitiva de la herramienta. Se ha decidido incluir en la fase de diseño ya que esta encapsula todas las funcionalidades que debería tener el plugin de forma muy resumida y, por tanto, ayuda a poder entender con más facilidad las partes del programa, que se describirán en el siguiente apartado. Todo ello ha sido diseñado siguiendo los casos de uso y requerimientos.

En la interfaz se ha decidido separar en dos secciones las diferentes funciones principales de la herramienta, exportar e importar. Por defecto el usuario se encontrará en el modo de exportar, tal y como se puede apreciar a continuación.

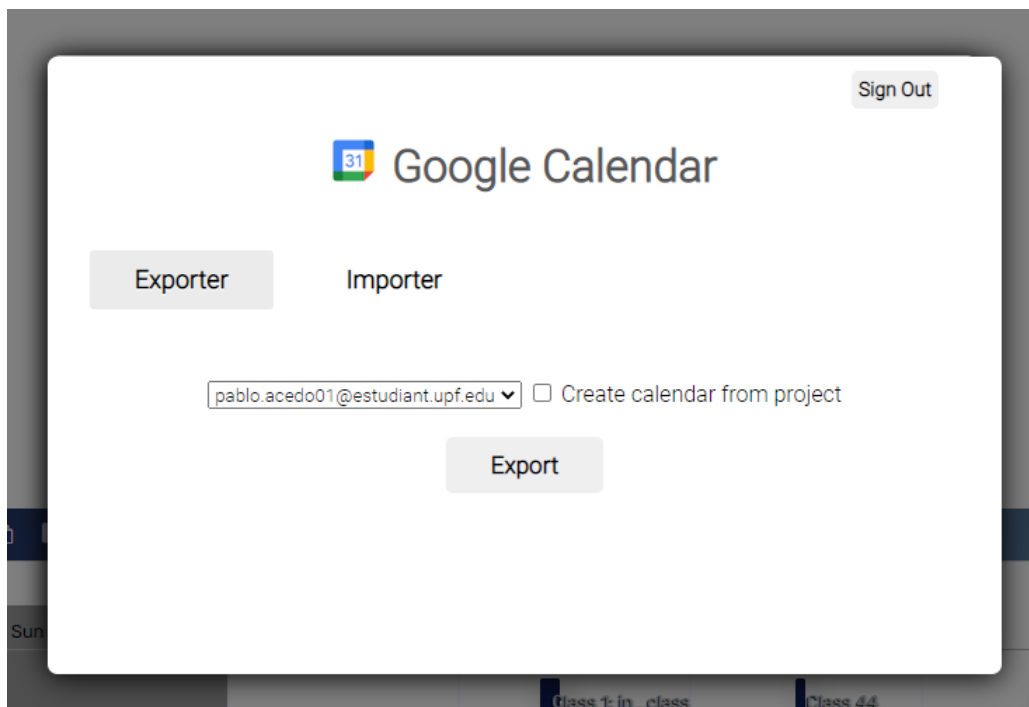


Figura 11: Interfaz: Exportar

En la esquina superior derecha, se encuentra el botón para poder iniciar o cerrar sesión de la cuenta de *Google*. Después encontramos el título,

acompañado de el logotipo actual de *Google Calendar* y, justo debajo de este, se puede observar, marcada en gris, la función seleccionada. En cuanto a la parte de exportación se puede observar que es muy sencilla, con tan solo tres elementos. Estos son un selector con la lista de calendarios disponibles en la cuenta del usuario, junto a este se encuentra una *checkbox* en la que se indica si se quiere crear un calendario nuevo o no a partir del proyecto actual, si esta se marca, desaparece el selector con la lista de calendarios, finalmente se encuentra el botón de exportar, al que se hará clic una vez se haya escogido la configuración deseada.

A continuación, se puede ver el modo de importar.

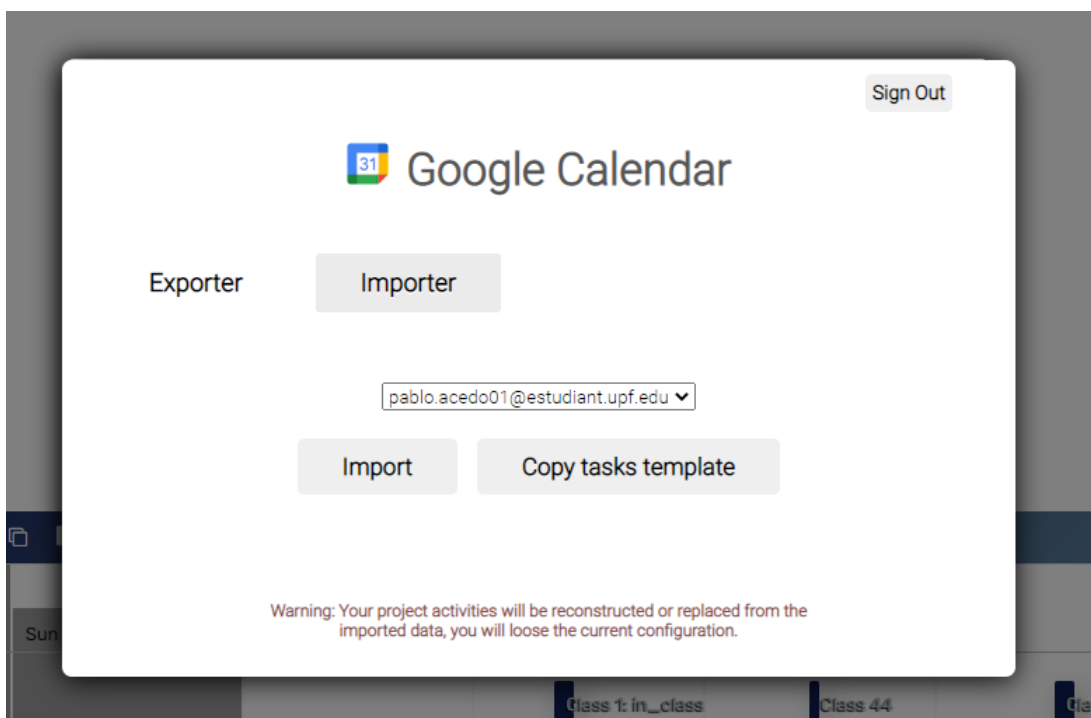


Figura 12: Interfaz: Importar

En este caso se puede observar que hay cuatro elementos. Por una parte tenemos un selector de calendarios para seleccionar a partir de qué calendario se quiere importar, además de dos botones, uno para importar con la configuración seleccionada y otro que servirá para pegar al portapapeles la configuración, en texto, de una tarea de *edCrumble*, la cual está explicada más adelante en el apartado 6.4. Finalmente, en un tono rojizo, se observa un mensaje de aviso al usuario, este mensaje informa de que al hacer la importación, todas las actividades creadas en el proyecto se reemplazarán por aquellas que se importan.

5.3. Lista de clases y métodos

Una vez hecho el análisis de los casos de uso y la interfaz de usuario donde se encapsulan las funcionalidades, se puede describir el esquema y la lista de las principales funciones que deberá tener el sistema a implementar. Para ello se crea una clase para cada una de las dos funcionalidades principales de la herramienta, exportar e importar, además de algunas funciones globales. Primero se visualizará un esquema, el cual se explicará detalladamente a continuación de este. Cabe destacar que, en la implementación, se podrán modificar o añadir más métodos y atributos según la necesidad.

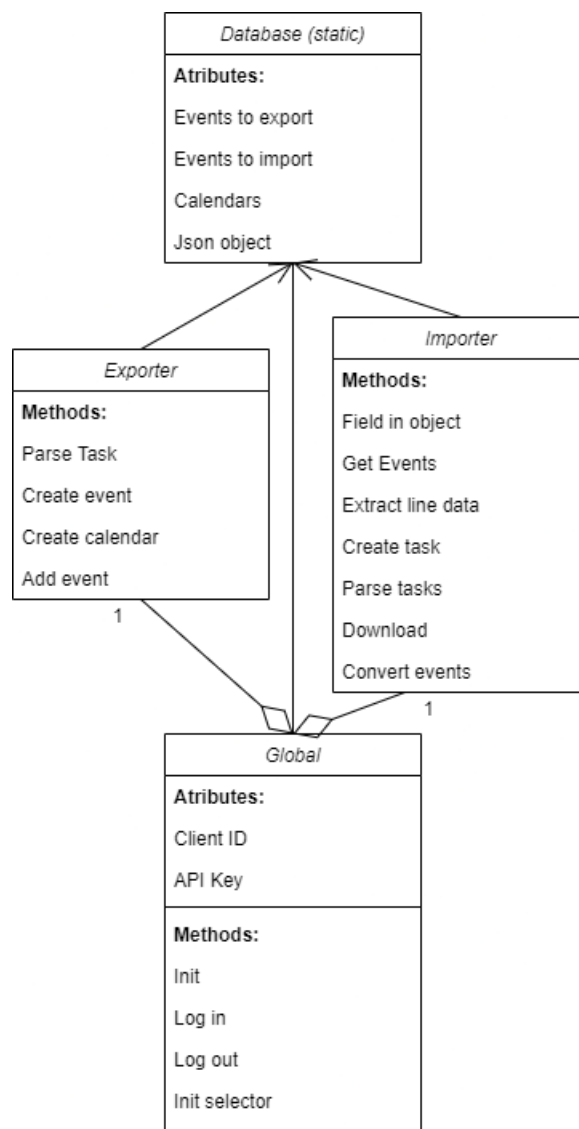


Figura 13: Diagrama de clases

- **Global:** No es una clase como tal, pero se utiliza para representar aquellas variables y funciones globales para todo el sistema. En cuanto a

variables encontramos el número identificador del cliente y la clave para la *API*, estos dos serán utilizados para la conexión con la interfaz de programación de *Google Calendar*. Además de estos atributos, en el sistema global también hay una asociación con un objeto que representa la base de datos, además de dos objetos que son el exportador e importador. A continuación, se describen los métodos globales.

- **Init:** Deberá inicializar los parámetros de la aplicación, tales como la base de datos y conectarse con la *API* de *Google*.
 - **Log in:** Deberá ser capaz de conectarse a la cuenta de *Google* del usuario y pedir permiso para modificar y leer datos de su *Calendar*.
 - **Log out:** Deberá desconectarse de la cuenta de *Google* del usuario y cerrar la herramienta.
 - **Init Selector:** Encargada de pedir la lista de calendarios de la cuenta del usuario, así como crear las opciones referentes a cada uno de ellos en el selector principal de la herramienta.
- **Base de datos:** En ella se encuentran todos los datos que utilizará la aplicación, estos se extraen de *Google Calendar* y del archivo principal del proyecto de *edCrumble*.
 - **Events to export:** Lista que contendrá todos los eventos a exportar.
 - **Events to import:** Lista que contendrá todos los eventos a importar.
 - **Calendars:** Lista que contendrá todos los calendarios de la cuenta de *Calendar* del usuario.
 - **Json object:** En este objeto se almacenarán todos los datos del proyecto de *edCrumble*.
 - **Exportador:** Clase en el que estarán los métodos involucrados en la exportación.
 - **Parse task:** Encargado de convertir una tarea de *edCrumble* a texto.
 - **Create event:** Crea y formatea un evento para *Calendar* a partir de los datos correspondientes.
 - **Create calendar:** Encargado de crear un calendario en la cuenta del usuario utilizando el título del proyecto de *edCrumble*.
 - **Add event:** Añade un evento creado a un calendario de *Google Calendar*.
 - **Importador:** Clase en el que estarán los métodos involucrados en la importación.

- **Field in object:** Encargado de encontrar un campo dentro de un objeto que representa una estructura de datos.
- **Get events:** Extrae los eventos existentes dentro de un calendario de *Google Calendar*.
- **Extract line data:** Extrae diferentes tipos de datos de una línea de texto a partir de un cierto formato.
- **Create task:** Encargado de crear una tarea, con configuración por defecto, para una actividad de un proyecto de *edCrumble*.
- **Parse tasks:** Convierte un texto que sigue un cierto formato, en una lista de tareas con sus correspondientes configuraciones.
- **Download:** Descarga un archivo *JSON*.
- **Convert events:** Convierte eventos de *Google Calendar* a actividades de *edCrumble*.

5.4. Formato de actividad

Para poder crear actividades de *edCrumble* en *Calendar* se necesita un formato que defina la configuración de estas y que sea legible, tanto por el usuario como por el programa, ya que, para simplificar, se utilizará el mismo formato tanto para la exportación como para la importación. A continuación, se encuentra la explicación de dicho formato adaptado para eventos de *Google Calendar*, el cual se detallará en la sección de implementación 6:

- **Título del evento:** En éste, además del título del evento, se debería especificar si la actividad es dentro o fuera de clase utilizando una etiqueta, esta etiqueta será simplemente un texto con 'in' o 'out' como posibles valores.
- **Descripción del evento:** En ella se encontrarán todas las tareas de la actividad de *edCrumble*. Aquí es donde se encuentra la mayor dificultad del plugin, ya que este formato deberá contener toda la configuración de cada tarea.

6. Implementación

Esta sección se centra en la fase de implementación del plugin. En ella tenemos en cuenta las tecnologías utilizadas, la conexión con las diferentes aplicaciones involucradas, y la implementación de todas las funcionalidades descritas en la sección 5.

6.1. Tecnologías utilizadas

Las tecnologías utilizadas debían ser compatibles tanto con *edCrumble* como con la *API* de *Google Calendar*. Teniendo en cuenta que *edCrumble* es una aplicación web, que todos los datos que hay que extraer de la aplicación están en el lado del cliente y que la *API* de *Google* utilizada se conecta y utiliza automáticamente las funcionalidades de *backend* de *Google*, las tecnologías involucradas son las más utilizadas en el lado cliente, también denominado *Front End*, las cuales son:

- **HTML:** Es un lenguaje de marcas de hipertexto, *HyperText Markup Language* por sus siglas en inglés. Con él se define la estructura del contenido web [16] y se enlazan todos los elementos involucrados, como los botones, selectores y archivos que definen la apariencia y funcionalidad del plugin.
- **CSS:** *Cascading Style Sheets* por sus siglas en inglés. Es un lenguaje de estilos [17] que se utiliza para describir la apariencia y presentación de los documentos *HTML*. En este caso se utilizan los mismos parámetros que los utilizados en *edCrumble* por coherencia estética tal y como se ha mencionado en los requerimientos.
- **JavaScript:** Es uno de los lenguajes de programación más utilizados para páginas web. En este caso es utilizado para programar todas las funcionalidades del plugin, desde la utilización de los datos necesarios, hasta la conexión con la *API* [26].

6.2. Conexión y utilización de la *API* de *Google Calendar*

Tal y como se ha mencionado anteriormente en la sección de requerimientos, para integrar *Google Calendar* en el plugin, se utiliza la *API* que proporciona *Google* para este fin [18]. Gracias a esta interfaz de programación cada usuario podrá conectarse a su cuenta de *Google* y la aplicación tendrá acceso tanto para modificar el calendario que el usuario requiera, como para leer los datos necesarios para los procesos de exportación e importación. Este proceso se ha realizado utilizando como base la plantilla de ejemplo proporcionada por *Google* para *JavaScript* [19]. A continuación, se encuentra la explicación de como se hace la conexión y la utilización de diferentes funciones de la *API*:

- **Conexión con la *API*:** La conexión con la interfaz se hace en una inicialización una vez la aplicación empieza su ejecución. Para esta inicialización se necesitarán diferentes parámetros, los cuales se explican a continuación, así como su proceso de obtención. Para la conexión se necesita pasar por el proceso de autenticación y autorización. Estos mecanismos permitirán dar acceso al usuario a su cuenta de *Google*, además de validar el acceso a diferentes datos de la cuenta desde una aplicación externa, en este caso datos de su *Calendar*. Para ello, primero se ha procedido a registrar la aplicación en desarrollo, el plugin, en la llamada *Google Cloud Console*, con la cual obtenemos unas credenciales que solo serán conocidas por la aplicación y por el servidor de *Google* que se encarga de la autenticación. Las credenciales obtenidas son las siguientes:
 - ***Client ID*:** Con esta credencial podremos tener acceso a datos privados del usuario, siempre que este lo consienta mediante una ventana de aviso y aceptación. Cada vez que se requiera información del *Calendar* a través de funciones de la interfaz de programación, se estará utilizando este parámetro.
 - ***API key*:** Se utiliza para tener acceso a datos públicos.

Estas credenciales son dos de los parámetros necesarios para la inicialización. Además de ellas, tenemos otros dos, los cuales son:

- **Documentos de descubrimiento:** En inglés *Discovery Documents*. Estos existen para diferentes *APIs* de *Google* [20]. En ellos se describe la superficie de la interfaz, así como la estructura de requerimientos y respuestas y cómo acceder a ella. Incluye también los métodos a utilizar. En este caso se utiliza un único documento, el referente a *Calendar*.
- **Ámbito de aplicación:** En inglés *Scope*. Con este parámetro se define a qué *API* queremos acceder y el nivel de acceso necesario. En este caso necesitamos acceso a *Calendar*, así como leer, crear y modificar datos [21].

Una vez se realice la inicialización con estos parámetros, se comprueba si el usuario ya ha iniciado sesión dentro de la aplicación, en caso negativo, este deberá identificarse con su cuenta a través de *Google Sign in*. Una vez hecho, si es la primera vez que entra en el plugin, se le requerirá que acepte los permisos necesarios para la aplicación, los cuales serán poder modificar y leer información de *Calendar*. A partir de aquí, el plugin ya está plenamente funcional.

- **Cargar calendarios disponibles:** Dentro de *Google Calendar* tenemos la opción de tener varios calendarios. La aplicación necesita estos calendarios, ya que el usuario tendrá la opción tanto de exportar actividades

como importar eventos a uno de ellos. Para obtener dichos calendarios se utiliza una función de la clase *CalendarList*, llamada *list*. Con ella se realiza una petición para obtener la lista de calendarios, en la respuesta de esta se obtiene dicha lista, donde cada calendario, entre sus parámetros, tiene un *ID* o código identificador, el cual se utilizará para poder operar dentro de esos calendarios, además de un campo llamado *summary*, que simplemente indica el nombre de cada uno de ellos. Este último parámetro será el que se mostrará al usuario dentro de un selector para que este elija sobre cual va a realizar la acción. El proceso de obtener la lista de calendarios se realiza una sola vez, justo después de haber iniciado sesión correctamente.

- **Crear calendarios:** A la hora de exportar, el usuario tiene la opción de hacerlo creando un calendario nuevo, el nombre del cual sería el del proyecto de *edCrumble* en el que está trabajando. Para ello, se utiliza una función, de la clase *Calendar*, llamada *insert*. Dicha función tiene como parámetro un objeto, en el cual tan solo se indica el nombre que queremos que el calendario tenga, si la petición se realiza con éxito, obtenemos el objeto calendario correspondiente al que se acaba de crear y se añade a la lista mencionada en el punto anterior.
- **Crear eventos:** En el proceso de exportación, se necesita traducir cada actividad en la línea de tiempo del proyecto de *edCrumble* en un evento del calendario seleccionado, por tanto necesitamos crear eventos dentro de un calendario. Para ello se utiliza la función, de la clase *Events*, llamada *insert*. Esta función, con la que se realiza una petición de crear un evento, recibe como parámetro un objeto en el cual se indica, utilizando su *ID*, una referencia al calendario en el cual se creará el evento, además del evento en sí, este último es un objeto con diferentes características, las cuales se definirán más adelante a la hora de explicar el código. Si la petición se resuelve con éxito, obtenemos una respuesta con el objeto evento.
- **Leer eventos:** Para la importación necesitamos realizar el proceso inverso al anterior, es decir, traducir los eventos del calendario indicado a actividades de *edCrumble*. Para esto se utiliza la función, de la clase *Events*, llamada *list*. Con esta función se requiere la lista de eventos creados dentro de un calendario, por tanto, el parámetro introducido en ella es el *ID* del calendario indicado. Si la respuesta es positiva, se recibe un objeto, del cual solo nos interesa la lista de eventos del calendario, de los que se detallará su utilización más adelante dentro de esta sección.

6.3. Conexión con edCrumble

La conexión con la aplicación web para la que desarrolla este plugin, *edCrumble*, se realiza utilizando un archivo *JSON*, *JavaScript Object Notation* por sus siglas en inglés. Este tipo de archivo, el cual se ha convertido en un estándar en el intercambio de datos entre cliente y servidor, almacena objetos y estructuras de datos [22]. Este está basado en texto y es fácil de leer por el ser humano además de ser muy fácil de leer y modificar en aplicaciones desarrolladas con *JavaScript*.

Los proyectos de *edCrumble* tienen toda la información necesaria guardada en este formato, es decir, cada proyecto tiene un archivo *JSON* asociado. No toda la información que contiene dicho archivo es necesaria para el propósito del plugin desarrollado, por tanto a continuación se definirán los campos utilizados, además del propósito que tiene cada uno de ellos dentro del plugin [23].

- **designTitle:** Este campo contiene el título del proyecto. Sería utilizado en el proceso de exportación en el caso de querer crear un nuevo calendario en *Calendar* a partir del proyecto.
- **startDate y endDate:** Definen las fechas tanto de inicio como final del proyecto, se utilizan para filtrar, en el proceso de importación, los eventos a leer dentro de dicho intervalo de tiempo, ya que añadir eventos fuera de dicho intervalo a la lista de eventos a importar no tendría sentido.
- **resourcesList:** Es una lista que hace referencia a los recursos que se utilizarán durante el curso. Para el desarrollo y funcionamiento de este plugin, de cada uno de ellos tan sólo tiene importancia su número identificador, ya que se podrá hacer referencia a este en un parámetro perteneciente a las tareas, el cual está explicado más adelante.
- **itemsList:** Este parámetro es el más importante para el plugin, ya que, es una lista de todas las actividades o sesiones del proyecto, las cuales se pueden visualizar en la línea de tiempo. Esta lista está presente en los dos procesos de los que es capaz el plugin, ya que en el caso de exportación, exportaremos todos los que haya dentro de ella, y en el caso de importación se añadirán los eventos correspondientes en dicha lista. Ahora se procederá a definir los diferentes parámetros, que tengan importancia para el plugin, del tipo de objeto de los que está formado la lista, llamado ítem o actividad.
 - **title:** Indica el título de la actividad. Se utilizará, en el proceso de exportación como título del evento creado en el calendario. En el proceso de importación se extraerá del título del evento.
 - **start y end:** Parámetros que indican el inicio y final de la sesión. Dado que estos parámetros tienen reciprocidad con los utilizados

en los eventos de *Google Calendar*, se utilizan sin dificultad en los dos procesos de la aplicación. El formato utilizado para ambos es fecha y hora.

- **group:** En este campo se indica si la sesión es fuera o dentro de clase. A la hora de exportar esto se traduce a una etiqueta de texto al lado del título del evento además de poder indicar el color del evento, el cual sería diferente según del tipo al que nos estamos refiriendo, esta última característica es opcional. En la importación se traducirá la etiqueta del evento a este parámetro. El formato utilizado es un entero con dos posibles valores, 1 dentro de clase y 2 fuera de clase.
- **tasks:** Cada actividad creada en el proyecto puede tener una lista de tareas, que es la que se almacena en este parámetro. Cada tarea contiene diferentes características, las cuales definiremos a continuación.
 - **type:** Este parámetro, de tipo entero, hace referencia a la taxonomía de Bloom que el usuario ha seleccionado para la tarea. Puede tener un valor de 1 a 7, haciendo referencia a *Neutral*, *Remembering*, *Understanding*, *Applying*, *Analysing*, *Evaluating* y *Creating* respectivamente. Cabe especificar, tal y como se puede suponer por cada uno de los indicadores mencionados, que la taxonomía de Bloom es una lista de habilidades y procesos del pensamiento que pueden aparecer en tareas escolares [28] tal y como es el caso.
 - **student_role:** Representa el tipo de trabajo que el alumno deberá hacer en esta tarea. Está representado por un número entero en formato de texto, el cual puede ser 1, 2 y 3, haciendo referencia a trabajo individual, en grupos o en un grupo formado por toda la clase respectivamente.
 - **students_group:** Con este campo, se hace referencia al número de alumnos que habrá en cada grupo, en el caso que el trabajo a hacer en la tarea sea en grupo. Está representado por un entero, cuyo valor por defecto es 2.
 - **teacher_role:** Describe la disponibilidad del profesor/a o profesores en la tarea. Su valor es un entero, dentro del rango de 1 a 3, en formato de texto. Puede significar que el profesor/a no esté presente, que esté presente de forma remota a través de la red o que esté presente físicamente.
 - **graded:** Con este parámetro se hace referencia a si la tarea será evaluable o no y, en caso afirmativo, qué tipo de evaluación se le aplica. También está representado por un entero en formato de texto, con valores en el rango de 1 a 3. Estos representan que la tarea no será evaluada, que será auto evaluada y evaluada

respectivamente.

- **min:** Está representado por un entero en formato de texto. Representa la duración de la tarea en minutos. La suma en minutos de todas las tareas no podrá ser mayor al tiempo que dura la actividad.
- **description:** Característica representada por texto. En ella se describe la tarea a realizar, es opcional.
- **resources:** Es una lista de enteros. Contiene la referencia a los diferentes recursos utilizados, cada elemento de la lista es el número identificador de cada recurso.

Todas las características definidas, en la exportación de *edCrumble* a *Calendar*, aparecerán en la descripción del evento creado para cada tarea de la sesión. Estas aparecerían en el formato de texto explicado en el siguiente apartado 6.4. En la importación, tal y como se explica también en el apartado señalado, no es necesario definir cada una de estas características, ya que se pueden crear actividades sin tareas, o automáticamente se creará una tarea con los valores por defecto en el caso de que únicamente esté definido su número identificador y duración. Hay parámetros de las tareas que no se han definido ya que no forman parte de ninguno de los dos procesos de la herramienta, por tanto, en la exportación no se leerán y en la importación tendrán el valor por defecto que se aplica en *edCrumble*. Estas características son *FC*, *PBL* y *DB*.

6.4. Estructura del formato de evento

Para poder hacer los procesos, tanto de exportación, como de importación, se necesita definir un formato para los eventos de *Google Calendar*, ya que estos utilizan una interfaz diferente a la de *edCrumble* y tan solo deja la opción de convertir en texto la configuración de actividades, incluyendo todas sus tareas con sus respectiva configuración. Por ello se ha decidido utilizar para este fin el título y la descripción del evento.

- **Título del evento:** Tal y como se especifica en el apartado de diseño, el título del evento en *Google Calendar* será el título de la actividad de *edCrumble*. Además, en este se incluye una etiqueta de texto, la cual define el tipo de actividad. Esta etiqueta, podrá ser especificada como *in_class* o *out_class*. A continuación, se puede apreciar esta aplicada a un evento.

■ Class 1: in_class
Martes, 4 de mayo - 8:30 - 10:30am

Figura 14: Título de evento con etiqueta

- **Descripción del evento:** En la descripción se encontrarán todas las tareas de la actividad de *edCrumble*. Es aquí donde está la complejidad, ya que esta descripción deberá ser leíble tanto por el programa, como por el usuario, ya que este deberá ser capaz de configurar tareas utilizando el formato que se describa a continuación. Cabe destacar que este formato es el mismo utilizado en los dos procesos.

```
#Task:  
#Duration:  
#Task_type:  
#Students_work:  
#Students_group:  
#Teachers_presence:  
#Evaluation:  
#Description:  
#Resources:
```

Como se puede apreciar, se ha intentado que el formato sea lo más sencillo posible, destacando con el carácter almohadilla (#) el campo al que se hace referencia, e introduciendo el valor después de los dos puntos del campo, algo parecido al formato utilizado en *JSON*. A continuación, se define la configuración de cada uno de estos campos.

- En la primera línea del formato tenemos el campo *task*, que hace referencia a la creación de una tarea nueva, en el se especifica el número de la tarea dentro de la lista de tareas, por tanto es un entero.
- En cuanto al campo *duration*, el cual especifica la duración de la tarea, se podrá especificar con un número decimal o entero.
- *Task_type* especifica el tipo de tarea a realizar, el valor del cual deberá especificarse con una de las siguientes palabras: *Neutral, Remembering, Understanding, Applying, Analysing, Evaluating, Creating*.
- *Students_work* especifica el tipo de trabajo que deberán hacer los alumnos en la tarea, el valor de este parámetro deberá especificarse

con uno los valores pertenecientes a la siguiente lista: *Individual*, *Group*, *All_Class*.

- *Students_group* informa sobre el número de alumnos en un grupo, en el caso de que la tarea sea en grupo, se representará con un número entero.
- *Teachers_presence* indica cómo el docente está presente en la tarea, el valor de este parámetro deberá indicarse con *Teacher_not_present*, *Teacher_Available_Online* o *Teacher_Available_F2F*.
- *Evaluation* describe el tipo de evaluación de la tarea. Puede tener uno de los siguientes valores: *Not_Graded*, *Auto_Evaluation*, *Graded*.
- *Description* define la descripción de la tarea, puede ser un texto cualquiera.
- Con *Resources* se hace referencia a los recursos utilizados en la tarea, se define con una lista de enteros separados por comas, por ejemplo: 1,2,3.

A continuación, en la figura 15, se puede visualizar un evento de *Google Calendar* creado con el formato definido, el cual contiene varias tareas.

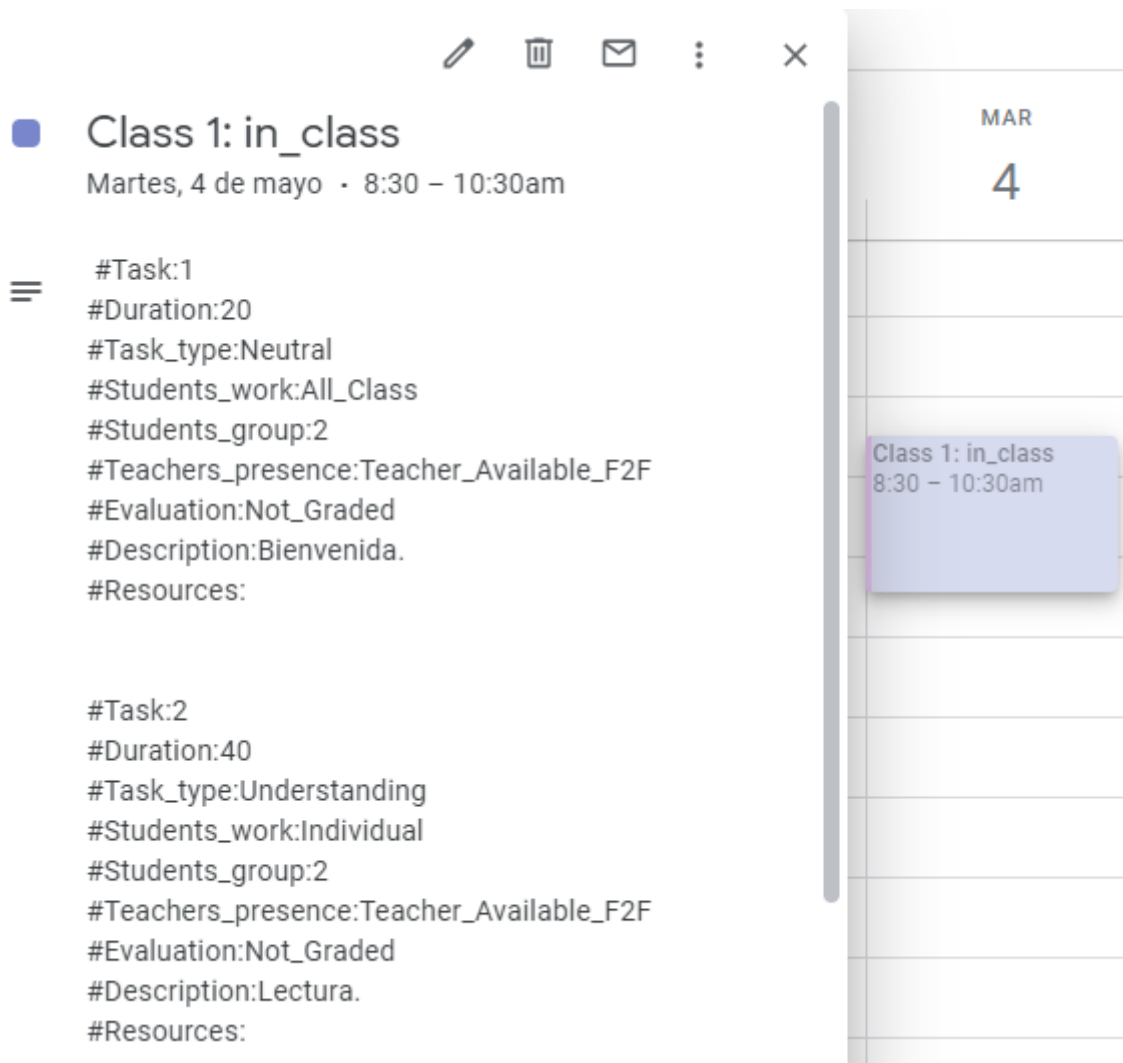


Figura 15: Evento con la configuración descrita

6.5. Estructura del código

Este apartado, es similar al apartado 5.3, ya que contiene aproximadamente la misma lista de clases, métodos y objetos, pero con la diferencia de que ahora, en vez de solo hacer una breve descripción de la función, se explica de forma detallada como ha sido implementada. A la hora de la implementación se han añadido funciones o implementado de una forma diferente alguna de las funciones globales definidas en el diseño. También cabe tener en cuenta que algunas de las funcionalidades descritas en el diseño estaban ya desarrolladas en la plantilla de *Google* utilizada, como por ejemplo el proceso de iniciar sesión. Estas funcionalidades ya han sido definidas en el apartado anterior de conexión con la interfaz de programación de *Google Calendar*, por tanto no se volverán a describir aquí. Cabe recalcar que los nombres utilizados en este apartado, aunque sí tienen relación directa, no son exactamente los definidos

en el diseño, sino los utilizados en la implementación.

- **Objetos globales:** Existen tres objetos globales en la herramienta, el exportador e importador, que se explican por separado más adelante ya que son clases complejas con varios métodos, y la base de datos, que es una estructura que contiene toda la información necesaria dentro del programa. Por ello ahora solo se definirá este último objeto.
 - **Base de datos:** Llamada *DB*, contiene todos los datos utilizados en la herramienta. Es un objeto de *JavaScript* formado por los campos:
 - ***events_to_export*:** Es una lista que contiene todas las actividades a exportar, cada una de estas actividades es un objeto extraído del archivo *JSON* del proyecto de *edCrumble*. Se almacenan aquí para que el acceso a estos datos sea más rápido e intuitivo. Cuando se ejecuta la aplicación, se lee de disco el archivo correspondiente y se introducen los datos de este campo.
 - ***events_to_import*:** Aquí tenemos el caso contrario al anterior, ya que esta lista está formada por los eventos a importar, donde cada uno de ellos es un objeto con el formato de evento de *Google Calendar*. Estos datos se introducen cuando se recibe respuesta de la petición de la lista de eventos dentro de un calendario.
 - ***calendars*:** También una lista, pero que contiene los calendarios de la cuenta del usuario, si se crea un calendario nuevo desde la herramienta también se añadirá a esta lista. De ella nos importa su número identificador y su nombre. Estos datos se introducen cuando se recibe respuesta de la petición de la lista de calendarios.
 - ***json*:** Es el mismo objeto que el archivo *JSON* de *edCrumble*. Se guarda en memoria para acceder a diferentes parámetros así como para poderlo modificar y exportar después a un fichero. Estos datos se introducen al iniciar la aplicación.
 - ***timeZone*:** Este parámetro contiene la zona horaria del usuario.
- **Funciones globales:** Aquí solo se definirán las funciones que no han sido extraídas de la plantilla proporcionada por *Google*, ya que, como se ha indicado anteriormente, la funcionalidad de estas está explicada en el segundo apartado de esta misma sección 6.2. El método que ha sido creado, aparte de aquellos que han sido aprovechados de la plantilla, es:
 - ***init_selector*:** En esta función se inicializa el selector de los calendarios que el usuario tenga en su cuenta de *Calendar*. En ella se crea el objeto correspondiente con formato *selector* para *HTML* y

se añade al cuerpo principal de la página. Después de haber sido creado, se hace una petición de la lista de calendarios utilizando la función llamada *list* perteneciente a la clase *calendarList* de la *API* de *Google Calendar*. El funcionamiento de esta función ha sido explicada anteriormente en apartado 6.2. Una vez se recibe la respuesta de la petición se obtiene la lista de calendarios, esta se almacena en la lista correspondiente del objeto base de datos, y a partir de su nombre se crea cada opción del selector y se añade a este. Finalmente se añade a la base de datos la zona horaria del primer calendario de la lista. Esta función no devuelve ningún valor o parámetro.

Además de esta función hay otra funcionalidad que se ha añadido, aunque no en forma de función. Esta es la lectura desde disco del archivo *JSON*. Esto se hace utilizando el método *fetch*, que es un método global de *JavaScript* par obtener recursos fácilmente. En este caso realizamos la petición del archivo, una vez se obtiene respuesta, utilizando una *callback* se introducen los datos correspondientes en la base de datos.

A continuación explicaremos por separado las clases que corresponden a las dos funcionalidades del plugin, ya que es donde se encuentran las funcionalidades centrales del propósito de la herramienta.

- **Exportador:** Tal y como indica su nombre, en esta se encuentran todos los métodos utilizados en el proceso de exportación. Este proceso es ligeramente más sencillo que la importación, en consecuencia se necesitan menos funciones para realizarlo. Estas son:
 - **parse_task:** Esta función se encarga de convertir la configuración de una tarea de *edCrumble*, la cual contiene todas las características definidas, a el formato de texto correspondiente definido en el apartado anterior, este último será el parámetro de salida de la función. Los parámetros que se introducen en ella son, la tarea en sí, que es un objeto de *JavaScript* con sus campos correspondientes, y el número de tarea, que corresponde a su índice dentro de la lista de tareas del archivo *JSON*. Al principio de este método se definen cuatro listas de texto, estas son *type_arr*, *student_role_arr*, *teacher_role_arr* y *graded_arr*. Como se puede deducir por su nombre, cada una de estas listas contiene, en texto, todas las opciones para estos parámetros de la tarea, los cuales se han definido en el apartado anterior 6.3. En este punto, se procede a crear el texto. Así que, siguiendo el formato, se van concatenando los valores de texto directamente definidos, con la configuración de la tarea, ya sea adjuntando directamente el valor del parámetro en caso de ser posible, o accediendo con el valor de este a las listas previamente

definidas en el inicio. Con la excepción de la lista de recursos, que se convierte directamente de *Array* a *String*.

- **create_event:** Este método, recibe como parámetros la actividad en el formato de objeto directamente extraído del archivo del proyecto, además de la descripción, que será la lista de tareas. Se encarga de crear o formatear un objeto de evento para *Calendar* a partir de los parámetros introducidos, para después poder hacer la petición de inserción. Este objeto será el parámetro de salida de la función, el cual tiene la siguiente configuración:
 - **summary:** En él se define el título o nombre del evento. Para ello se extrae el nombre de la actividad de entrada y se acompaña de la etiqueta que define de qué tipo es la actividad, la cual se extrae a partir del parámetro *group* de la actividad.
 - **colorID:** El color del evento también se define a partir de *group*, el cual será de un color morado claro si la actividad es dentro de clase o verde en caso de ser fuera de clase. Esto no es del todo relevante, pero se crea así para que a partir del aspecto visual haya una mayor diferenciación. No habrá problema en que el usuario cambie el color de los eventos desde su cuenta de *Calendar*.
 - **start y end:** Estos dos parámetros definen las fechas de comienzo y final del evento. Estos dos valores se extraen de la actividad de *edCrumble*, también se incluye la zona horaria, la cual ha sido extraída del calendario correspondiente.
 - **description:** La descripción del evento. Este parámetro tiene el valor de la descripción que se introduce como parámetro en esta función, la cual contiene las tareas procesadas por *parse_task*.
- **add_event:** En este método, el cual recibe como parámetro un índice que indica a que calendario queremos exportar, es donde se unen todas las funciones anteriores. Dentro de un bucle se extrae cada actividad a exportar, con *parse_event* se transforma cada una de las tareas de cada actividad a el formato de texto de la herramienta y se almacena en una variable, la cual será la descripción del evento. Seguidamente con *create_event*, función en la que se introducen el objeto que representa la actividad y la descripción del evento, se crea el evento, que posteriormente se incluirá a la petición de añadir, juntamente con el código identificador del calendario al que se exportará la actividad, utilizando la función de crear eventos de la *API* de *Calendar*.
- **create_calendar:** Esta función sólo se ejecutará si se decide crear un calendario a partir del proyecto de *edCrumble* en el que se está trabajando. En esta función se crea un objeto que representa un

calendario, este contiene el título del proyecto, extraído del archivo *JSON*, además de la zona horaria. Una vez creado, se procede a realizar la petición de inserción del calendario en el *Calendar* del usuario. Si esta petición se resuelve con éxito se recibe el objeto del calendario que se acaba de crear en la cuenta, por tanto, este se añade a la lista de calendarios de la aplicación, la cual está en la base de datos, y se crea una nueva opción dentro del selector para este. Finalmente, como esta función se ejecuta a la hora de exportar, con *add_event* se exportan todas las actividades del proyecto al nuevo calendario.

- **Importador:** Esta clase representa la funcionalidad de importar. En este proceso se requiere la lista de eventos de un calendario en concreto y todos estos eventos se convierten a el formato de actividad de *edCrumble*. Este proceso se hace a partir de interpretación de texto, lo que lo hace complejo y, por tanto, se necesitan varias funciones. Estas son:
 - **fieldInObject:** Este método se ha creado para asegurar que no se crean variables indefinidas, ya que se utiliza para comprobar si un cierto tipo de objeto tiene un cierto parámetro. La función recibe como parámetros un objeto y un texto que define el nombre del campo que se desea obtener. Si el campo existe dentro del objeto se retorna el valor correspondiente, en cambio, en caso de que no exista, se retorna un valor por defecto definido en el objeto. Cabe destacar, por este último punto, que los tipos de objetos que se introduzcan en la función deberán tener en común el parámetro por defecto que se indica.
 - **extractLineData:** Esta función recibe como parámetros una línea de texto, en formato *string*, y una palabra clave o *keyword*. Su función es, en caso de que la palabra clave indicada exista dentro de la línea de texto introducida, extraer el resto de componentes del texto excluyendo dicha palabra clave. Esto se hace para extraer datos utilizando el formato de texto para tareas de la herramienta. El valor a retornar son los datos extraídos de la línea o, -1, en el caso de no existir la *keyword* dentro del texto.
 - **create_task:** Inicializa una tarea de *edCrumble* con todos sus parámetros por defecto. Esto se hace para que no haya campos vacíos o errores cuando se haga la conversión, de tareas en la descripción de un evento de *Calendar*, a tareas con formato de *edCrumble*.
 - **parse_tasks:** Este método, es el encargado de, a partir de la descripción de un evento, la cual el parámetro de entrada, extraer todas las tareas existentes dentro de ella. Todo esto siguiendo el formato explicado en 6.4. Al principio de la función, se crean varias variables locales. Estas son:

- **tasks:** Lista en la que se almacenarán todas las tareas extraídas de un evento. Se inicializa, como es obvio, en una lista vacía. Este será el valor de salida de la función.
- **mapper:** Objeto en el que se definen algunos parámetros de configuración de tareas de *edCrumble*, es decir, parámetros pertenecientes a la configuración del archivo *JSON* de un proyecto de dicha plataforma de LD. Dentro de este objeto, hay más objetos que representan cada uno de los parámetros mencionados, dónde, dentro de ellos, están mapeadas cada una de las palabras clave, con las que se define el posible valor del parámetro correspondiente utilizando el formato de tareas creado para *Calendar*, con cada uno de sus posibles valores dentro de *edCrumble*. Concretamente, estos son, *type*, *student_role*, *teacher_role* y *graded*. A continuación, para entender mejor la estructura de la variable, se puede visualizar la configuración de *type* utilizando el formato de objeto descrito.

```
var mapper={
  type:{
    ':1,
    'Neutral':1,
    'Remembering':2,
    'Understanding':3,
    'Applying':4,
    'Analysing':5,
    'Evaluating':6,
    'Creating':7
  },
}
```

Figura 16: mapper: type

En la imagen anterior se puede observar que, el primer campo, cuyo nombre es definido únicamente por comillas simples, es el parámetro por defecto que tienen en común todos los objetos dentro de *mapper*, referido en la explicación de la función *fieldInObject*.

- **keywords:** No es una variable, sino un conjunto de variables. Cada una de ellas es de tipo *string*, contienen el nombre de cada campo de la tarea definido en la descripción del evento, tal y como se define en el apartado anterior 6.4.

Acto seguido, después de crear las variables locales, se separa, en líneas individuales, el texto de entrada a la función. Con ello se ob-

tiene una lista de textos, donde cada uno de sus valores corresponde a una línea de la descripción del evento en la que se han definido las tareas. Después, se crea un objeto 'tarea' utilizando *create_task*. Ahora, utilizando un bucle, para cada línea de la descripción, se comprueba si esta contiene alguna de las *keywords* definidas anteriormente y así configurar cada tarea con sus correspondientes valores. Esto se hace utilizando la función *extractLineData*. En el caso de que la palabra clave sea '#Task:', se añade la tarea a la lista local de tareas y se inicializa el objeto de nuevo para así configurar una nueva a partir de los valores extraídos. En caso de que el campo encontrado sea uno de los definidos dentro del objeto local *mapper*, se utiliza la función *fieldInObject* para comprobar que el valor extraído a partir del texto sea compatible con el formato. Para los demás parámetros, tan solo se extrae el valor, en formato *string* o convirtiéndolo a entero, y se incluye dentro de la tarea, excepto para la lista de referencias a los recursos, la cual se convierte de texto a lista de enteros.

- **download:** Con este método se logra descargar un archivo de texto y guardarlo en disco. Esto se hace creando un elemento dentro del cuerpo *HTML* de la herramienta y configurándolo para iniciar una descarga automática. En la función se introducen como parámetros el nombre del archivo, además de los datos que contendrá. El formato utilizado para codificar estos datos es *UTF-8*, ya que el archivo a descargar será de texto, concretamente, un archivo *JSON*.
- **convertEvents:** En esta función, se convierten e introducen, en el archivo *JSON* del proyecto, todos los eventos de *Calendar* a importar. Para cada evento a importar, se comprueba si este está creado dentro del período de tiempo del proyecto, si no lo está, este evento no se incluye. En el bucle se inicializa una actividad vacía, su *id* será la posición del evento dentro de la lista de eventos a importar, su título es el título del evento, posteriormente se comprueba, utilizando la etiqueta presente en el título, si la actividad será fuera o dentro de clase, si no existiese dicha etiqueta, la actividad sería, por defecto, dentro de clase. La lista de objetivos se marca por defecto a ningún objetivo para que no haya errores al leer el archivo en el proyecto de *edCrumble*, para que así el usuario pueda añadir posteriormente los objetivos en caso de que los haya creado. Posteriormente se lee del evento la descripción de la actividad, si dicha descripción existe, se utiliza la función *parse_tasks* para crear todas las tareas de la actividad. Cada una de estas actividades, convertidas ya a formato de *edCrumble*, se almacena en una lista inicializada en el inicio de la función. Finalmente, una vez recorrida la lista de eventos a importar, utilizando el archivo *JSON* original del proyecto, se sustituye la lista de actividades de este, llamada

itemsList, por la que se acaba de crear, por tanto no se preservarán las actividades creadas previamente en el proyecto. Después de este último paso, se introduce en la configuración del proyecto el nuevo último *id*, utilizado para añadir al proyecto, desde *edCrumble*, más actividades. Para acabar se utiliza la función *download* para descargar el archivo *JSON* modificado, ya que la herramienta aún no está integrada en la aplicación.

- **getEvents:** Finalmente esta es la función que contiene todas las demás. En ella se realiza el requerimiento de todos los eventos de un calendario. Si este requerimiento responde positivamente, se introducen todos los eventos a importar en la lista de eventos a importar y, posteriormente se ejecuta la función *convertEvents*.

Se puede acceder a todo el código descrito en esta sección utilizando la *URL* del anexo 8.3.

7. Evaluación

Esta sección está dedicada a la evaluación del plugin. Esta evaluación se ha hecho a partir de un cuestionario con una estructura definida, la cual ayudará a poder evaluar el término conocido como *usabilidad*, el cual es un neologismo con el que nos referimos a la facilidad con la que una persona puede utilizar una herramienta, además de poder extraer diferentes sugerencias para la mejora del plugin y realizar un análisis *DAFO* para evaluar las debilidades, amenazas, fortalezas y oportunidades de este. Cabe destacar que, por el gran formato y posibilidades que ofrece además de que es una opción muy práctica al permitir llegar a más gente de una forma sencilla y en línea, el cuestionario ha sido creado utilizando la herramienta *Google Forms*. El cuestionario ha sido añadido al anexo de este documento 8.3.

7.1. Plan de prueba de evaluación

En este apartado se definen las características principales y el contexto en el que realizará esta prueba, además de cómo está estructurado el cuestionario de evaluación. También se define por qué se ha utilizado esta estructura y los métodos que este incluye.

7.1.1. Características principales

La prueba se realiza en el contexto de este trabajo de fin de grado y tiene, como objetivo principal, evaluar la herramienta desarrollada. Por ello, la realizarán personas con un perfil concreto. Para la elección de los participantes, partiendo de que la herramienta ha sido implementada para una plataforma de LD, se tiene el objetivo de que la persona encuestada forme parte del entorno de la docencia o educación y estar familiarizada con herramientas TIC aplicadas en la educación. Las personas encuestadas también deberán conocer *edCrumble*, ya que el plugin es un añadido a dicha plataforma. El cuestionario, con una duración de entre ocho y doce minutos, se realiza de forma remota y el estudiante no está presente durante este, aunque existe la opción de contactar vía correo electrónico. Se ha optado por esta modalidad dado que permite a los participantes escoger el momento para realizar la prueba y es posible hacerla de una forma completamente autónoma. Para facilitar la evaluación y optimizar el tiempo del cuestionario, en vez de tener que utilizar de forma directa la herramienta, los participantes visualizarán dos vídeos para poder contestar a las preguntas. Por ello, se ha recurrido a la plataforma *Google Forms* para realizar el cuestionario, tal y como se ha indicado anteriormente, además de la plataforma *YouTube* para visualizar los vídeos correspondientes. En cuanto al rol del estudiante, éste se ha encargado, con el apoyo de la directora del trabajo, del diseño del cuestionario y la preparación de los vídeos.

Una vez introducidas las características principales de este proceso, a continuación, se describen todas las partes del formulario.

7.1.2. Declaración de consentimiento

En la segunda página del cuestionario, después de una breve bienvenida, se incluye una declaración de consentimiento. En esta declaración, se informa al usuario que realiza el cuestionario, del objetivo del estudio y las circunstancias de este, es decir, que el formulario se realiza para evaluar una herramienta realizada en el marco de un trabajo de fin de grado. A la persona que realiza el cuestionario se le solicita su nombre y su correo electrónico, para poder contactar con ella en caso de tener alguna duda sobre sus respuestas, además de la aceptación de los términos de esta declaración de consentimiento. En estos términos, de forma resumida, se comunica que la participación en el estudio es totalmente voluntaria, el consentimiento para tener en consideración su participación en los resultados del cuestionario puede ser revocado en cualquier momento, los datos solicitados serán exclusivamente utilizados para fines de investigación y que la identidad de la persona está completamente protegida además de especificar que sus datos serán tratados según marcan las leyes. Puede ver la declaración completa en el anexo 8.3.

7.1.3. Datos demográficos y conocimientos

Una vez el usuario ha aceptado los términos y da su consentimiento, este pasará a la sección de datos demográficos y conocimientos sobre la herramienta *edCrumble*. En esta sección se solicita el género de la persona, su franja de edad, los años de experiencia que tiene en docencia y finalmente cuantas veces ha utilizado la herramienta de *Learning Design* en cuestión. Se solicitan estos datos para entender la situación de la persona en relación a su conexión con el mundo académico o universitario, así como para saber las nociones que tiene sobre la aplicación web *edCrumble*, ya que para utilizar el plugin y llegar a entender su finalidad, sería positivo que la persona encuestada conociese, o al menos tuviese nociones de la herramienta.

7.1.4. Cuestionarios de usabilidad

Este apartado del cuestionario es el más importante y representativo, ya que es el que de verdad nos ayudará a entender la facilidad de uso del plugin desarrollado y si este es práctico. Para evaluar esta *usabilidad*, se ha decidido utilizar el método *System Usability Scale*, o simplemente SUS por sus siglas en inglés. Este método fué creado por John Brooke en 1986 y básicamente es una escala para medir la *usabilidad* de cualquier sistema, ya sea de hardware, software o aplicaciones web, como es el caso. Este ha sido utilizado durante muchos años y, aunque es relativamente sencillo, ha demostrado ser un proceso eficaz, válido y fiable en términos de *usabilidad* [24], [25]. Es muy sencillo de

responder por las personas cuestionadas, además de ser rápido de realizar y capaz de seguir siendo fiable aunque el número de participantes en el estudio no sea muy elevado, lo que lo hace ideal para el caso de este trabajo ya que no se prevé encuestar a un gran número de personas. El cuestionario está formado por diez sencillas afirmaciones, las cuales son las siguientes:

1. Creo que me gustaría utilizar este sistema con frecuencia.
2. El sistema me pareció innecesariamente complejo.
3. Creo que el sistema es fácil de usar.
4. Creo que necesitaría el apoyo de personal técnico para poder utilizar este sistema.
5. Me pareció que las distintas funciones de este sistema estaban bien integradas.
6. Me pareció que había demasiadas incoherencias en este sistema.
7. Me imagino que la mayoría de la gente aprendería a utilizar este sistema muy rápidamente.
8. Me pareció que el sistema era muy incómodo de utilizar.
9. Me sentí muy seguro o segura utilizando el sistema.
10. Tuve que aprender muchas cosas antes de ponerme en marcha con este sistema.

Como se puede observar el cuestionario intercala afirmaciones de ámbito positivo con otras de ámbito negativo, a las que el encuestado tan solo deberá responder a cada una de ellas utilizando una escala de cinco puntos, la cual va desde *Totalmente en desacuerdo* a *Totalmente de acuerdo* con estos dos incluidos. Cabe destacar que con este método no obtendremos una respuesta de qué está fallando o cuales son las debilidades en concreto del sistema, sino que proporciona unos resultados que nos orientan en su facilidad de uso y *practicidad*.

Dado que el plugin desarrollado tiene dos funcionalidades principales y la utilización de estas dos es bastante distinta, se ha decidido evaluarlas por separado. Esto permitirá analizar con más claridad los resultados y una mayor facilidad para encontrar errores. Lo ideal habría sido que cada usuario probase el sistema y después realizara el cuestionario, pero por motivos prácticos, ya que actualmente el plugin no está integrado en la plataforma, se ha decidido realizar dos vídeos en los que se muestran ambos procesos respectivamente, esto además ayudará a que el proceso de evaluación sea más eficiente, dado que la persona encuestada tardará menos en visualizarlos que en hacer las

pruebas con la herramienta. Los vídeos contienen una simulación que combinan el funcionamiento real y actual de la herramienta, con la estética y acabados que esta debería tener una vez integrada, además de una voz en *off*, todo ello ayuda a entender mejor cada proceso y entender el sentido que tiene el plugin dentro de *edCrumble*. A continuación, se puede apreciar el proceso donde se muestra de forma práctica las dos funcionalidades, para cada una se explica el contenido del vídeo, en el que se asume que el usuario ya ha iniciado sesión con su cuenta de *Google* y ha dado los correspondientes permisos a la herramienta, por lo que la conexión se ha realizado de forma exitosa. Cabe destacar que en los vídeos no se muestra la versión más actualizada de la herramienta, aunque esto no supone un problema para evaluar la *usabilidad* dado que los cambios son mínimos.

- **Exportación:** En el vídeo ² se puede apreciar la interfaz de dentro de un proyecto de *edCrumble*. A esta se le ha añadido un icono en la parte superior derecha de la línea de tiempo en la que se encuentran todas las actividades o sesiones del proyecto. Este icono corresponde al logotipo de *Google Calendar* y será con el que se abrirá el plugin. Se puede apreciar que varias tareas han sido creadas en el proyecto y se visualiza de forma rápida la configuración de una de estas. Al abrir el plugin, por defecto, estaremos en el modo de exportación, donde se puede visualizar un selector que contiene la lista de calendarios que podemos seleccionar para realizar el proceso además de tener la opción de crear un calendario nuevo a partir del proyecto. Se decide optar por esta última opción y hacer clic al botón de exportar. Tras una breve espera, todo el proceso ha sido realizado con éxito y se aprecia que se han creado en el calendario del usuario los diferentes eventos correspondientes a las sesiones, todos ellos con su correspondiente configuración tal y como se hicieron en la herramienta de *Learning Design*. Una vez el usuario lo ha visualizado, puede proceder a contestar el SUS de esta parte.
- **Importación:** El vídeo ³ empieza en la misma situación que en el comienzo del anterior caso. Pero ahora vamos al modo importación, al que podemos cambiar con un simple clic. Vemos que en este caso todos los elementos presentes en el modo exportación seguirán, aunque el botón ahora será para importar, y aparecerá un nuevo campo, el cual ofrece ayuda al usuario ya que contiene una plantilla con el formato utilizado para definir la configuración de tareas de una actividad dentro de un evento de *Calendar*. El usuario del vídeo copia esta plantilla y cambia de pestaña del navegador para dirigirse a un calendario dentro de su cuenta de *Google Calendar*. Una vez aquí, crea un evento, al que le pone el título, se elige el color para definir el tipo de actividad y en la descripción se pega la plantilla anteriormente copiada para crear y configurar dos

²<https://www.youtube.com/watch?v=o6BJnXPB34I>

³<http://youtube.com/watch?v=g8zyjH1wbeU>

tareas diferentes. Una vez creado el evento, se cambia a la pestaña del proyecto dentro de la herramienta de *Learning Design*, para escoger el calendario correspondiente del que se quiere importar, después tan solo hace clic al botón llamado *import* y tras una breve espera se puede ver que la actividad previamente creada como evento en *Calendar* aparece en la línea de tiempo del proyecto y, al abrirla, se puede apreciar que la configuración es exactamente como el usuario deseaba. Igual que en el caso anterior, ahora la persona encuestada pasaría a contestar el SUS correspondiente a esta sección.

7.1.5. Sugerencias de mejora

Dentro del cuestionario se requiere al usuario, de forma opcional, su opinión acerca de cada funcionalidad de la herramienta al final de cada una de las secciones correspondientes a ellas, justo después de cada SUS. Además de esto, en la última página del formulario, se requiere la opinión del plugin en general, así como poder marcar diferentes propuestas con las que le gustaría que contara la herramienta, estas son:

- Poder elegir por separado los eventos/actividades a importar/exportar.
- Tener la opción de crear automáticamente un proyecto totalmente nuevo al importar un calendario de Google Calendar.
- Poder seleccionar las franjas horarias o franja de días a importar/exportar.
- Poder seleccionar el tipo de actividades (fuera o dentro de clase) a importar/exportar.

Se ofrecen estas posibilidades para poder extraer más información a la hora de decidir diferentes mejoras o añadidos a la funcionalidad del plugin.

7.2. Participación

En este apartado se define la participación en el cuestionario y todo el proceso realizado para que se pudiese llevar a cabo. También, a partir de los datos obtenidos en el apartado de datos demográficos y conocimientos, se define el perfil de la persona encuestada.

En total, han participado nueve personas, las cuales han sido reclutadas para el estudio a partir de los contactos de la directora de este trabajo de fin de grado, buscando el perfil anteriormente definido. Con ello, se ha asegurado que todos los participantes tuviesen nociones de qué es *edCrumble*, aunque alguno de ellos no lo haya utilizado de forma directa. A todos ellos se les contactó a través de correo electrónico, en el cual se adjuntaba el formulario, del que

se ha hecho dos versiones, una en castellano y otra en inglés. En el mensaje se les hizo la petición de contestar el cuestionario en el caso de que tuviesen oportunidad. A continuación se mostrarán los datos demográficos obtenidos.

- **Procedencia:** De las nueve personas, seis provienen de la Universitat Pompeu Fabra de Barcelona, dos provienen del Centro de Estudios Dolmen en L'Hospitalet de Llobregat y una pertenece a la Universidad de Vigo, Galicia.
- **Género:** Cuatro de las personas encuestadas son de género femenino, mientras que las cinco personas restantes son de género masculino.

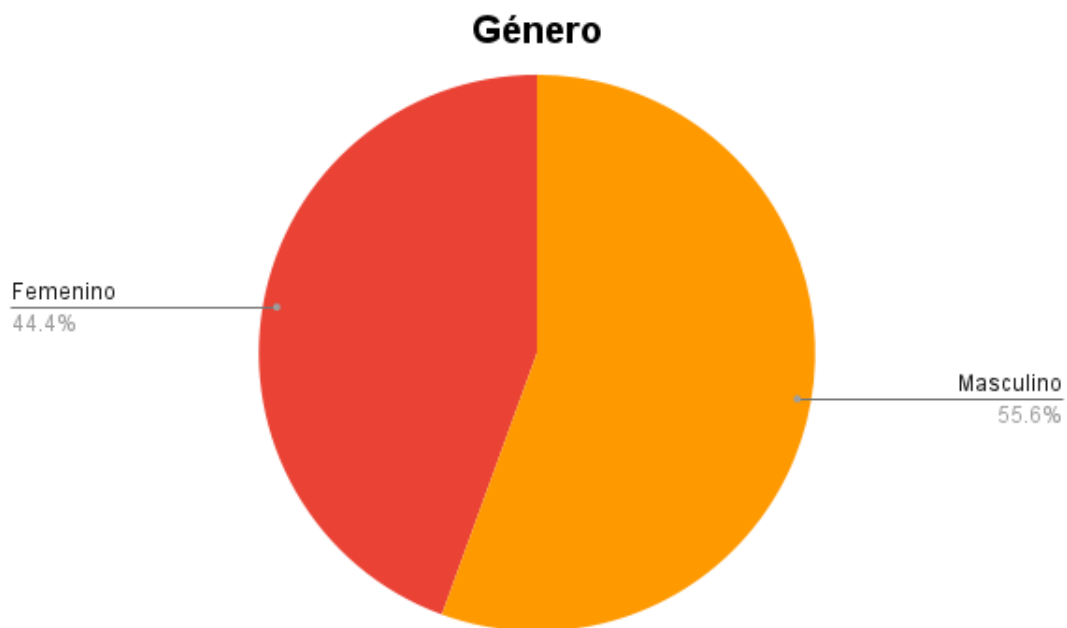


Figura 17: Género de los encuestados

- **Franja de edad:** La edad predominante de los encuestados se encuentra entre treinta y cuarenta años, ya que dos tercios pertenecen a esta franja de edad tal y como se puede apreciar en la siguiente gráfica.

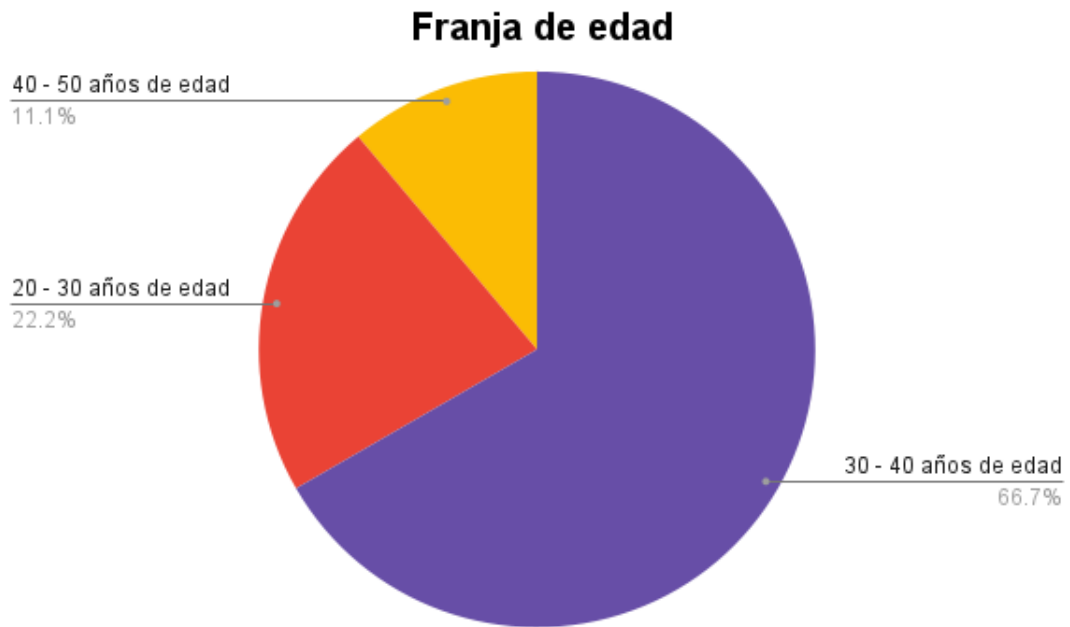


Figura 18: Franja de edad de los encuestados

- **Años de experiencia en docencia:** En las respuestas a esta pregunta se encuentra bastante variedad, ya que tres personas tienen más de diez años de experiencia en docencia, otras tres personas tienen entre seis y diez años de experiencia, dos personas tienen entre uno y cinco años de experiencia y tan solo una persona no tiene experiencia en docencia.



Figura 19: Experiencia en docencia de los encuestados

- **Utilización de *edCrumble*:** Obtenemos que dos tercios de estas personas han utilizado dos veces o más la herramienta, dos personas no la han utilizado nunca y una persona la utiliza de forma ocasional.



Figura 20: Experiencia de edCrumble de los encuestados

A partir de los datos obtenidos, podemos observar que las personas encuestadas encajan con el perfil deseado, ya que en su mayoría son personas con conocimientos de *edCrumble* y están ligadas al mundo de la docencia o la investigación.

7.3. Resultados

Una vez definido el número de participantes y el perfil mayoritario de estos, se puede proceder a interpretar y analizar los resultados del formulario.

7.3.1. Cuestionarios de usabilidad

Empezaremos por el resultado del cuestionario SUS. Para la evaluación de estos resultados, se han seguido las indicaciones de los recursos en los que se ha consultado la forma de realizar el cómputo [24], [25]. La respuesta a cada afirmación del cuestionario, tal y como se ha explicado anteriormente, será una de las cinco opciones posibles, se le dará un valor entre 1 y 5, siendo 1 el valor correspondiente a la opción *Totalmente en desacuerdo* y 5 el valor correspondiente a *Totalmente de acuerdo*. Dado que en el cuestionario se intercalan afirmaciones positivas, que son las afirmaciones impares, y negativas,

que corresponden a las afirmaciones pares, a la hora de hacer el cálculo se le resta un valor de 1 a las impares y el resultado de las pares se le resta a el número 5. Finalmente, todos estos valores son sumados y multiplicados por 2.5 para obtener un valor en la escala de 0 a 100. En este cuestionario se ha realizado este cálculo para cada uno de los participantes y después se ha hecho una media. Los resultados son los siguientes, no se deben interpretar como un porcentaje ya que no lo son.

- **Exportación:** 86.11 puntos en total, donde podemos ver los resultados de cada pregunta a continuación.

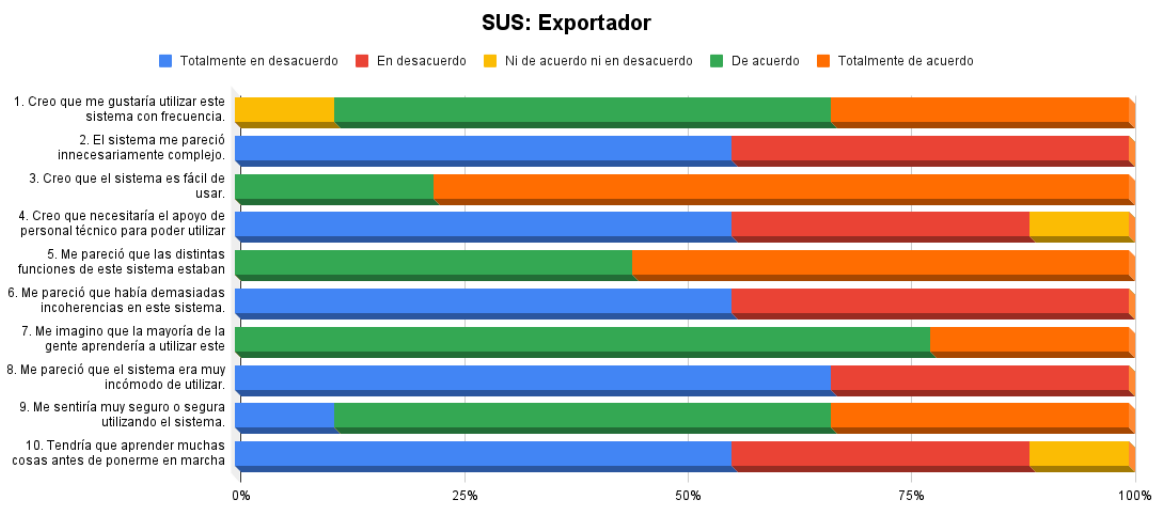


Figura 21: Resultados SUS, Exportador

- **Importación:** 76.39 puntos en total, donde podemos ver los resultados de cada pregunta a continuación.

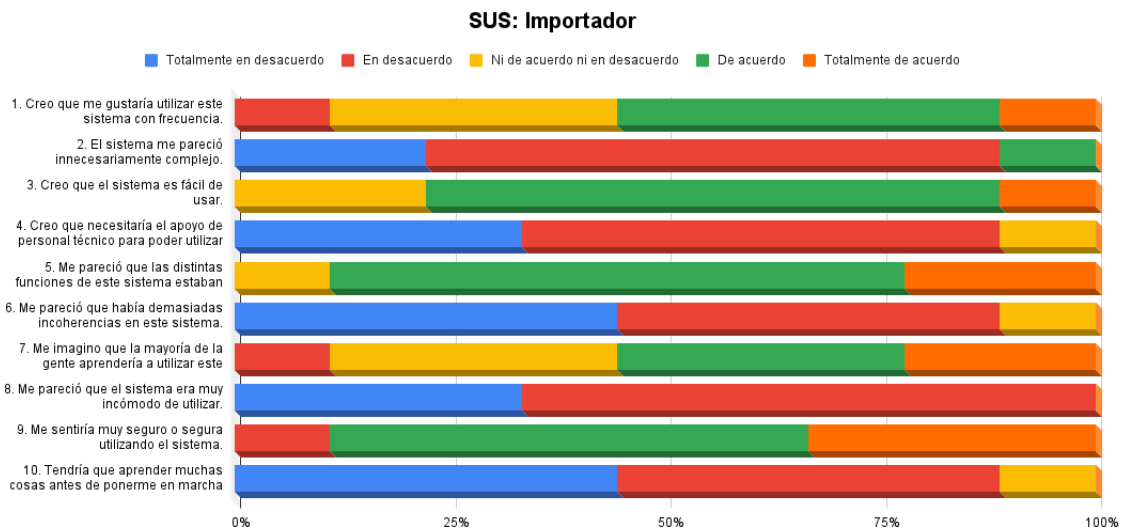


Figura 22: Resultados SUS, Importador

- **Global:** 81.25 puntos.

Para poder interpretar y medir estos resultados hace falta entender cual es la calificación de cada puntuación [25].

- **80.3 o más:** Este resultado se considera un sobresaliente, el sistema tiene un nivel de *usabilidad* muy alto.
- **68 o más:** Con estos parámetros se indica que el sistema está bien, está aprobado, pero necesita mejorar.
- **51 o menos:** Esta puntuación indica que el sistema no tiene *usabilidad*, por tanto, no es para nada práctico.

Tal y como podemos ver, los resultados obtenidos son buenos. Siendo el proceso de exportación el sistema mejor valorado. Entraremos más en detalle a la hora de hacer el análisis DAFO.

7.3.2. Sugerencias de mejora

En este apartado se analizarán las sugerencias de mejora que ha habido, además de extraer aquellas propuestas que más han votado las personas encuestadas. Esto se hace para extraer las mejoras que tengan más sentido de desarrollar tal y como está diseñado el plugin. Se separará por funcionalidades para poder hacer un mejor análisis.

- **Exportación:** Hay pocas sugerencias de mejoras para esta funcionalidad, en general la respuesta ha sido muy positiva. De todas formas entre las peticiones están la de poder acceder a los recursos a los que se hace referencia en cada tarea de cada actividad en los eventos de *Calendar* y otra es poder separar los eventos de fuera y dentro de clase en dos calendarios diferentes o, que se creen los dos a la hora de exportar para poder visualizar o un tipo de actividades u otro activando y desactivando los calendarios.
- **Importación:** Aquí no encontramos una respuesta tan positiva como en el proceso opuesto, aunque no se encuentran muchas más propuestas de mejora. Entre estas están la de cambiar el nombre al botón de ayuda que actualmente es *Help*, para dejar más claro que contiene esa opción, otra petición sería la de poder crear y configurar las tareas desde *Calendar* utilizando un formato más sencillo que el que se utiliza ahora, ya que se vería más útil evitar tener que depender de la plantilla que proporciona el plugin.

Propuestas de funcionalidades útiles más votadas por los participantes:

- Poder elegir por separado los eventos/actividades a importar/exportar: 6 votos.
- Poder seleccionar las franjas horarias o franja de días a importar/exportar: 6 votos.
- Poder seleccionar el tipo de actividades (fuera o dentro de clase) a importar/exportar: 6 votos.

7.3.3. Análisis DAFO

A partir de todos los datos obtenidos en la evaluación, es conveniente realizar un análisis DAFO, en el cual se analizan las debilidades, amenazas, fortalezas y oportunidades de la herramienta desarrollada. Así que procedemos a ello:

- **Debilidades:** Claramente la mayor debilidad del plugin es el modo de importación. Aunque el resultado del cuestionario SUS no es un resultado malo, en el cuestionario se han recibido, por parte de la mayoría de

personas, comentarios cuestionando el contexto en el que se utilizaría el modo de importación, ya que es algo más complejo que su funcionalidad opuesta, además, no gusta la dependencia de una plantilla para configurar tareas, ya que deberían abrir siempre un proyecto de *edCrumble* para poderlas modificar en *Calendar*.

- **Amenazas:** La debilidad de la falta de *usabilidad* del proceso de importación es una amenaza en sí para el sistema, ya que dicho proceso tiene la probabilidad de caer en desuso y que se haya desarrollado dicho sistema para que no se utilice supone un malgasto de recursos.
- **Fortalezas:** Obviamente la mayor fortaleza es el modo de exportación. Tiene un alto índice de *usabilidad* según los resultados obtenidos, factor que es muy importante dado que uno de los objetivos es que la herramienta sea fácil e intuitiva de utilizar. Se han recibido comentarios muy positivos acerca de esta funcionalidad y todos los encuestados lo utilizarían con frecuencia. Otra fortaleza del plugin es ofrecer los dos procesos tanto de exportación como importación.
- **Oportunidades:** Las mayores oportunidades de la herramienta pasan, sin duda, por seguir mejorándola. El hecho de que la función de exportación haya tenido tan buena acogida hace que, implementando más prestaciones y mejoras sugeridas se incentive aún más la utilización del plugin. Otra oportunidad es la de seguir también mejorando el proceso de importación para intentar hacerlo más fluido y sencillo, lo que supondría un incremento de *usabilidad*.

8. Conclusiones

Finalmente, después de los procesos de diseño, desarrollo y evaluación de la herramienta en la que se basa este trabajo, se pueden sacar las conclusiones pertinentes. Estas incluyen una reflexión sobre el éxito del trabajo, la valoración sobre el cumplimiento de los objetivos marcados para este TFG, las limitaciones encontradas en el marco de este trabajo y limitaciones de la herramienta, además de una valoración personal.

8.1. Conclusiones generales

En este apartado, se realiza una reflexión sobre los factores exitosos del proceso, además de valorar el cumplimiento de los objetivos indicados en la introducción.

En cuanto a las conclusiones sobre los aspectos de éxito, se deben mencionar, de forma general, los retos que han surgido durante el proyecto, así como su resolución utilizando las técnicas y tecnologías que han permitido el desarrollo de la herramienta.

Al empezar el proyecto, dado que este se debía realizar en un intervalo de tiempo definido, primero se procedió a planear todas sus partes de forma temporal, tal y como se puede apreciar en el diagrama de Gantt y su explicación en la sección 3. Esta planificación ha sido un éxito, ya que ha ayudado a mantener un orden correcto entre todas las fases y tareas.

En la primera fase, surge el problema de entender el entorno sobre el que se desarrolla el trabajo. Obviamente, como el objetivo es hacer un plugin para la plataforma *edCrumble*, surge la necesidad de entender el contexto general de la aplicación, las TIC aplicadas en la educación, especialmente el concepto de *Learning Design*. Esto se soluciona realizando una búsqueda de información, con el soporte de la directora, para profundizar sobre LD y diversas plataformas de diseño de cursos que siguen ese patrón de trabajo, la cual se detalla en la sección 2. Una vez se ha investigado y entendido el contexto, llega la fase de diseño del plugin. En esta, se plantea el reto más importante, el cual es entender la problemática principal a resolver y diseñar el sistema de solución. Gracias a la información extraída en el proceso anterior y varias reuniones con la directora, surgen los requerimientos que deberá tener la herramienta para solucionar el reto principal y, a partir de estos, se procede a realizar el diseño. El método utilizado resulta en buenos resultados, ya que se obtiene un diseño coherente que resuelve la problemática mencionada. Finalmente, el mayor reto ha sido desarrollar el diseño realizado. Para ello, por una parte, se ha buscado realizar una implementación lo más eficiente posible, aprovechando las posibilidades que proporciona *JavaScript* y, por otro lado, realizar una interfaz de usuario adecuada para *edCrumble* y

fácil de usar, tal y como se ha podido comprobar en la fase de evaluación. Por todo ello, se puede concluir, que las técnicas y tecnologías utilizadas han sido las adecuadas, ya que han ayudado a un correcto desarrollo del trabajo.

Ahora, desde la perspectiva de objetivos, se puede afirmar que el objetivo principal del proyecto, diseñar e implementar el plugin de conexión entre *edCrumble* y *Calendar*, se ha cumplido. Es cierto que, en términos de *usabilidad*, la función de exportación es mejor que la de importación, pero se llega a la conclusión de éxito dado que se ha llegado a implementar, a partir de un diseño que reúne todos los requerimientos de la herramienta, una versión funcional del plugin, cuya evaluación confirma que, en general, cumple su cometido principal de conectar las dos plataformas a las que se hace referencia, siendo una herramienta útil, en términos generales, para *edCrumble* y sus potenciales usuarios.

En cuanto a los objetivos secundarios, también pueden considerarse como cumplidos. Se ha desarrollado y gestionado un trabajo siguiendo la planificación prevista, todo esto con la ayuda de un diagrama de Gantt, que ha ayudado a secuenciar las fases de desarrollo del proyecto. Durante este, diferentes aptitudes adquiridas en el grado, tales como la programación web, creación de diagramas de clases y búsqueda de artículos académicos e información, han sido aplicadas y han sido clave para el trabajo. Se ha estudiado el entorno del marco del trabajo, tanto las TIC aplicadas a la educación, como herramientas de *Learning Design*, utilizando una estrategia de búsqueda adecuada, lo que ha ayudado a realizar un buen diseño de la herramienta.

Hace falta recalcar que, de momento, no se puede evaluar con certeza si la herramienta desarrollada mejora la *usabilidad* de *edCrumble*, dado que el plugin no está integrado en la plataforma principal. Esto es una limitación, la cual se describe, en detalle, a continuación.

8.2. Limitaciones y trabajo futuro

La mayor limitación de la herramienta, es que no ha sido integrada en *edCrumble*, lo que significa que, dado que no se ha probado la interacción directa con ella, no se ha podido evaluar en su totalidad. Una integración proporcionaría más oportunidades en cuanto a la expansión de funcionalidades de la herramienta, ya que ahora mismo no se pueden utilizar funcionalidades tales como, por ejemplo, la selección de las actividades en el *timeline* de *edCrumble*. Otra limitación podría ser que, en la importación, se sustituyen todos los eventos existentes en el proyecto por aquellos importados, además de tener que seguir obligatoriamente el formato indicado en los eventos de *Google Calendar* en el caso de querer configurar tareas desde su descripción.

Una última limitación, es que ahora tan solo se puede exportar e importar todas las actividades o eventos dentro del proyecto de un calendario, en vez de poder seleccionar cuales se quieren incluir en el proceso.

En cuanto a trabajo futuro, la mejor forma de seguir con este trabajo es integrando el plugin en la plataforma principal, *edCrumble*. Solucionaría varias limitaciones explicadas anteriormente y daría paso a poder hacer una evaluación de la herramienta que proporcione mayor fiabilidad. Siguiendo las sugerencias de mejora extraídas de la evaluación de la herramienta, también contaría como trabajo futuro añadir las siguientes funcionalidades o características al plugin.

- Acceder desde los eventos creados en la exportación, en *Calendar*, a los enlaces de los recursos utilizados en las tareas de una actividad. Se podría hacer, por ejemplo, poniendo el dirección de internet de cada uno de ellos, en el caso de que sean accesibles a través de *URL*.
- Permitir crear dos calendarios diferentes en el proceso de exportación. Uno para actividades de fuera de clase y otro para actividades en clase. Esto permitiría visualizar cada uno por separado. Sería cómodo para algunos usuarios.
- Rediseñar el proceso de importación, para hacerlo más intuitivo y así aumentar su *usabilidad*, aunque los resultados obtenidos no sean del todo desfavorables.
- Escoger separadamente, qué actividades quiere exportar o importar el usuario. Esto daría paso a que no tuviese que exportar o importar el proyecto entero al calendario deseado.
- Poder escoger el período temporal a exportar e importar. Parecido al punto anterior, se podrían escoger todas las actividades que se encuentren dentro de un período de tiempo.
- Escoger si, a la hora de importar, rehacer de nuevo el proyecto o mantener los eventos creados anteriormente y tan solo añadir aquellos nuevos que se importan.

8.3. Valoración personal

La realización de este proyecto, me ha ayudado a comprender mejor todo el trabajo que se encuentra detrás de las herramientas de software profesionales. También me ha abierto la posibilidad a aprender tecnologías con las que nunca había tenido contacto antes de este trabajo, tales como las tecnologías *frontend* y trabajar con tecnologías aplicadas a la educación o docencia. He mejorado mis capacidades de análisis, al tener que extraer información válida de diferentes

artículos académicos y páginas web, y planificación, al planificar, por fases, un trabajo de tanta duración. Me he sentido cómodo durante la realización de este también gracias al trato recibido por parte de la directora, siempre dispuesta a ayudar y aconsejar, además de haber tenido un gran grado de libertad a la hora de diseñar y desarrollar el sistema. En conclusión, la valoración es muy positiva, y este proceso seguro que me ayudará en los próximos proyectos en los que trabaje.

Referencias

- [1] María Pilar Colás Bravo. Juan de Pablos Pons. Javier Ballesta Pagán. (January 31, 2018). Incidencia de las TIC en la enseñanza en el sistema educativo español: una revisión de la investigación. Universidad de Sevilla. Universidad de Murcia. RED. Revista de Educación a Distancia. Núm. 56, Artic. 2.
- [2] Sims, Rod. (n.d.). Beyond Instructional Design: Making Learning Design a reality. Journal of Learning Design, Capella University.
- [3] Smart Sparrow. (n.d.). What is Learning Design? Smart Sparrow website. <https://www.smartsparrow.com/what-is-learning-design/>
- [4] Albó, L. Hernández-Leo, D. (2018). edCrumble: designing for learning with data analytics. Proceedings of the 13th European Conference on Technology Enhanced Learning, EC-TEL 2018, Leeds, UK, September 2018 (Accepted paper).
- [5] Smart Sparrow. (n.d.). Why We Prefer “Learning Design” Over “Instructional Design”. Smart Sparrow website. <https://www.smartsparrow.com/2016/07/12/why-we-prefer-learning-design/>
- [6] Koper, R., Olivier, B. (2004). Representing the Learning Design of Units of Learning. Educational Technology Society, 7 (3), 97-111.
- [7] Koper, R. (2006). Current Research in Learning Design. Educational Technology Society, 9 (1), 13-22.
- [8] Berlanga, A.J. García, F.J. Carabias, J. (n.d.). IMS Learning Design: Hacia la Descripción Estandarizada de los Procesos de Enseñanza. Universidad de Salamanca.
- [9] Learning Designer. (n.d.). Learning Designer website. <https://www.ucl.ac.uk/learning-designer/>
- [10] LAMS. (n.d.). LAMS international website. <https://www.lamsinternational.com/>
- [11] D. Hernández-Leo, J.I. Asensio-Pérez, M. Derntl, F. Pozzi, J. Chacón, L.P. Prieto, and D. Persico. (2018). “An Integrated Environment for Learning Design”. Frontiers in ICT, vol. 5, no. May, pp. 1–19.
- [12] Albó, L. Hernández-Leo, D. (2019). Co-creating a web-based visual representation model for authoring blended learning designs. Interaction Design and Architecture, 42, 164-182.

- [13] Dalziel J.: Implementing learning design: the Learning Activity Management System (LAMS). (2003). 20th Annual. Conf. Australas. Soc. Comput. Learn. Tert. Educ. 7–10, Dec.
- [14] Laurillard, D. et al. (2013). A constructionist learning environment for teachers to model learning designs. *J. Comput. Assist. Learn.*, 29(1), 15–30.
- [15] Jenny Preece, Yvonne Rogers, Helen Sharp. (2007). Interaction design : beyond human-computer interaction. 201-205.
- [16] MDN Web Docs. (n.d.). HTML: Lenguaje de etiquetas de hipertexto. MDN Web Docs webpage.
<https://developer.mozilla.org/es/docs/Web/HTML>
- [17] MDN Web Docs. (n.d.). CSS. MDN Web Docs webpage.
<https://developer.mozilla.org/es/docs/Web/CSS>
- [18] Google. (n.d.). API Reference. Google Calendar API website.
<https://developers.google.com/calendar/v3/reference>
- [19] Google. (n.d.). JavaScript Quickstart. Google Calendar API website.
<https://developers.google.com/calendar/quickstart/js>
- [20] Google. (n.d.). Discovery Document. Google API Discovery Service website.
<https://developers.google.com/discovery/v1/reference/apis>
- [21] Google. (n.d.). OAuth 2.0 Scopes for Google APIs. Google Identity website.
<https://developers.google.com/identity/protocols/oauth2/scopes>
- [22] FileInfo. (n.d.). .JSON File Extension. FileInfo website.
<https://fileinfo.com/extension/json>
- [23] Laia Albó. (2021, June 7). edCrumble learning design object: json structure specifications (Version 1). Zenodo.
<http://doi.org/10.5281/zenodo.4905491>
- [24] Usability.Gov. (n.d.). System Usability Scale (SUS). Usability.Gov website.
<https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html>

- [25] Nathan Thomas. (n.d.). How To Use The System Usability Scale (SUS) To Evaluate The Usability Of Your Website. Usability Geek.
<https://usabilitygeek.com/how-to-use-the-system-usability-scale-sus-to-evaluate-the-usability-of-your-website/>
- [26] MDN Web Docs. (n.d.). JavaScript. MDN Web Docs webpage.
<https://developer.mozilla.org/es/docs/Web/JavaScript>
- [27] Albó, L. Hernández-Leo, D. (2018). Identifying design principles for learning design tools: the case of edCrumble. Proceedings of the 13th European Conference on Technology Enhanced Learning, EC-TEL 2018, Leeds, UK, September 2018 (Accepted paper)
- [28] Equipo pedagógico de Campuseducación.com. (May 12, 2020). Taxonomía de Bloom. Campus educación website.
<https://www.campuseducacion.com/blog/recursos/articulos-campuseducacion/taxonomia-de-bloom/>

Anexos

Anexo 1: Código

<https://github.com/PabloAcedo/TFG-edCrumble-Google-Calendar-Plugin>

Anexo 2: Cuestionario de evaluación

Formulario evaluación de plugin de Google Calendar para edCrumble

Hola, soy Pablo, estudiante de Ingeniería de Sistemas Audiovisuales en la Universidad Pompeu Fabra.

Este es un formulario de evaluación sobre un plugin que he diseñado y desarrollado para mi Trabajo de Fin de Grado que conecta funcionalidades de la plataforma de Learning Design, edCrumble, con Google Calendar. Concretamente, este plugin es capaz de exportar a su Google Calendar, en forma de eventos, todas las actividades que haya añadido en la línea de tiempo de un proyecto de edCrumble. También es capaz de importar eventos creados en Google Calendar, siguiendo un determinado formato en su descripción, al timeline de un proyecto de edCrumble en forma de actividades. Cabe destacar que este plugin aún no está operativo en la plataforma al estar en fase de evaluación.

En la siguiente sección encontrará una declaración de consentimiento informado que deberá aceptar para poder realizar el formulario. Los datos que se le pidan serán utilizados únicamente para los resultados del estudio y, en el caso de su nombre y e-mail, para contactar con usted en caso de alguna duda sobre el resultado del cuestionario. Todos sus datos serán tratados de manera confidencial.

Estoy muy agradecido por su colaboración, para cualquier duda que tenga acerca de este estudio por favor, póngase en contacto conmigo a través de:

pablo.acedo01@estudiant.upf.edu

Muchas Gracias.

* **Obligatòria**

Protección de datos

Declaración de consentimiento

Título del Estudio: Evaluación de diseño del plugin de conexión de edCrumble con Google Calendar.

Objetivo del Estudio: En este estudio queremos medir la usabilidad de dicho plugin y valorar qué posibles mejoras necesitaría. Para ello, primero se le hará unas preguntas sobre datos demográficos y familiarización con edCrumble, posteriormente queremos analizar su opinión acerca de dos vídeos que muestran las funcionalidades del plugin mediante dos cuestionarios "SUS", los dos cuestionarios mencionados serán idénticos en estructura pero se estarán evaluando dos funcionalidades diferentes.

Responsable del estudio: Pablo Acedo Vázquez.

Este documento quiere informaros sobre un estudio al que os invitamos a participar. Este estudio lo lleva a cabo un estudiante en el marco de su trabajo de fin de grado y ha sido aprobado por la tutora responsable de este. Nuestra intención es que recibís la información correcta y suficiente para qué podáis decidir si aceptáis o no participar en este estudio. Os pedimos que leáis esta sección con atención y que nos formuláis las dudas que tengáis.

1. Nombre *

2. Correo electrónico *

MANIFIESTO QUE HE SIDO INFORMADO/DA DE LAS CUESTIONES SIGUIENTES RELACIONADAS CON EL ESTUDIO:

- Mi participación en este estudio es voluntaria y, si en cualquier momento deseo cambiar mi decisión, puedo retirar mi consentimiento en cualquier momento.

- La participación en este estudio consiste en medir la usabilidad de dicho plugin y valorar qué posibles mejoras necesitaría, para ello se le requerirá en algún momento de su nombre, correo electrónico y franja de edad. La persona Responsable del tratamiento de mis datos personales es Pablo Acedo Vázquez.

- Mis datos personales serán recogidos y tratados con finalidades exclusivamente docentes y de investigación y sin ánimo de lucro.

- Mis datos serán anonimizados, de manera que no se podrá conocer mi identidad a partir de los datos que se recojan.

- Se guardará secreto sobre la información personal que facilito, y solo se usará con finalidad docente y de investigación en el marco de este Estudio, de forma que no se me pueda identificar en los resultados del estudio.

- Siguiendo el principio de minimización, solo se recogerán los datos mínimos que sean necesarias para llevar a cabo el Estudio, y una vez haya acabado la finalidad docente o de investigación que se derive de este estudio, se destruirá toda la información de carácter personal que haya facilitado de forma definitiva.

- He sido informado/da mediante "Google Forms" sobre el Estudio, sobre su finalidad y sobre los datos que se recogerán, y he consentido a participar en este Estudio.

- El tratamiento de los datos de carácter personal de todos los participantes se ajustará al que se dispone al Reglamento General de Protección de Datos (UE) 2016/679 y a la Ley Orgánica 3/2018 de Protección de Datos Personales y Garantía de los Derechos Digitales. De acuerdo con el que se establece en esta legislación, podéis ejercer los derechos de acceso, modificación, oposición y supresión de vuestros datos de carácter personal dirigiéndoos al Responsable del tratamiento, identificado a continuación y a través de los canales de contacto establecidos.

3. Consentimiento *

Seleccioneu totes les opcions que corresponguin.

Acepto los términos de esta declaración de consentimiento.

Datos demográficos y conocimientos sobre edCrumble

Por favor, conteste las siguientes preguntas.

4. 1. Género *

Maqueu només un oval.

- Femenino
- Masculino
- Otro
- Prefiero no decirlo

5. 2. Franja de edad *

Maqueu només un oval.

- 20 - 30 años de edad
- 30 - 40 años de edad
- 40 - 50 años de edad
- 50 - 60 años de edad
- 60 - 70 años de edad
- Otra

6. 3. Años de experiencia en docencia *

Maqueu només un oval.

- Menos de 1 año
- Entre 1 y 5 años
- Entre 6 y 10 años
- Más de 10 años
- Sin experiencia en docencia

7. 4. ¿Cuántas veces ha utilizado edCrumble? *

Maquee només un oval.

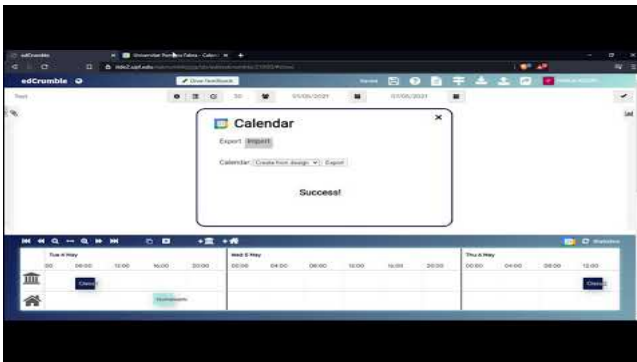
- Nunca he utilizado edCrumble
- Una sola vez
- Dos veces
- Más de dos veces
- Lo utilizo ocasionalmente
- Lo utilizo regularmente

Exportar
de
edCrumble
a Google
Calendar

A continuación deberá visualizar un vídeo en el que se muestra el actual modo de exportar actividades de un proyecto de edCrumble a Google Calendar. Una vez visualizado el vídeo, por favor, proceda a responder al cuestionario proporcionado.

*Indicación: Si no ve lo suficientemente grande el video haga click en el título del mismo o utilice este enlace: <https://youtu.be/o6BJnXPB34I>

Exportar: Vídeo de demostración



<http://youtube.com/watch?v=o6BJnXPB34I>

<http://youtube.com/watch?v=o6BJnXPB34I>

8. 1. Creo que me gustaría utilizar este sistema con frecuencia. *

Maquee només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

9. 2. El sistema me pareció innecesariamente complejo. *

Maqueeu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

10. 3. Creo que el sistema es fácil de usar. *

Maqueeu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

11. 4. Creo que necesitaría el apoyo de personal técnico para poder utilizar este sistema. *

Maqueeu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

12. 5. Me pareció que las distintas funciones de este sistema estaban bien integradas. *

Maqueeu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

13. 6. Me pareció que había demasiadas incoherencias en este sistema. *

Maqueeu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

14. 7. Me imagino que la mayoría de la gente aprendería a utilizar este sistema muy rápidamente. *

Maqueeu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

15. 8. Me pareció que el sistema era muy incómodo de utilizar. *

Maqueu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

16. 9. Me sentiría muy seguro o segura utilizando el sistema. *

Maqueu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

17. 10. Tendría que aprender muchas cosas antes de ponerme en marcha con este sistema. *

Maqueu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

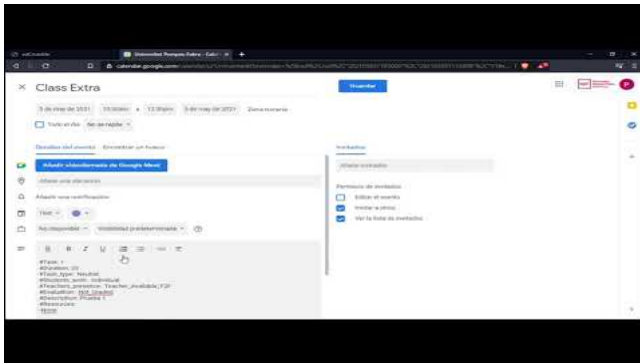
18. Comentario y/o sugerencias de mejora: Nos gustaría contar con su opinión o sugerencias para mejorar la función de exportar, si quiere puede redactarlas a continuación

Importar de Google Calendar a edCrumble

Siguiendo la misma metodología que en la sección anterior, por favor, visualice el vídeo que hay a continuación, el cual muestra el actual proceso de importar eventos de Google Calendar a edCrumble. Una vez visualizado proceda a responder las cuestiones proporcionadas.

*Indicación: Si no ve lo suficientemente grande el video haga click en el título del mismo o utilice este enlace: <https://youtu.be/g8zyjH1wbeU>

Importar: Vídeo de demostración



[http://youtube.com/watch?](http://youtube.com/watch?v=g8zyjH1wbeU)

[v=g8zyjH1wbeU](http://youtube.com/watch?v=g8zyjH1wbeU)

19. 1. Creo que me gustaría utilizar este sistema con frecuencia. *

Maqueu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

20. 2. El sistema me pareció innecesariamente complejo. *

Maqueu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

21. 3. Creo que el sistema es fácil de usar. *

Maqueu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

22. 4. Creo que necesitaría el apoyo de personal técnico para poder utilizar este sistema. *

Maqueu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

23. 5. Me pareció que las distintas funciones de este sistema estaban bien integradas. *

Maqueu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

24. 6. Me pareció que había demasiadas incoherencias en este sistema. *

Maqueu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

25. 7. Me imagino que la mayoría de la gente aprendería a utilizar este sistema muy rápidamente. *

Maqueu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

26. 8. Me pareció que el sistema era muy incómodo de utilizar. *

Maqueu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

27. 9. Me sentiría muy seguro o segura utilizando el sistema. *

Maqueu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

28. 10. Tendría que aprender muchas cosas antes de ponerme en marcha con este sistema. *

Maqueu només un oval.

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

29. Comentario y/o sugerencias de mejora: Nos gustaría contar con su opinión o sugerencias para mejorar la función de importar, si quiere puede redactarlas a continuación

Sugerencias de mejora

30. Nos gustaría saber su opinión acerca de el plugin en general (si le parece útil, si la interacción le parece bien diseñada, ...), si quiere puede dejar su opinión a continuación

31. Si quiere, marque las funcionalidades que consideraría útiles que tuviese el plugin evaluado

Selecioneu totes les opcions que corresponguin.

- Poder elegir por separado los eventos/actividades a importar/exportar
- Tener la opción de crear automáticamente un proyecto totalmente nuevo al importar un calendario de Google Calendar
- Poder seleccionar las franjas horarias o franja de días a importar/exportar
- Poder seleccionar el tipo de actividades (fuera o dentro de clase) a importar/exportar

Altres: _____

Final

Ha llegado al final de este formulario. Le estamos realmente agradecidos por haber dedicado parte de su tiempo para formar parte de este estudio, su opinión nos ha ayudará mucho para mejorar el plugin.

Recuerde que ante cualquier duda no dude en contactar conmigo a través de:
pablo.acedo01@estudiant.upf.edu

Muchísimas gracias, le deseamos un buen día.

Google no ha creat ni aprovat aquest contingut.

Google Formularis