

# Learn2Learn: Uso de aprendizaje computacional y estrategias de gamificación para el diseño de una herramienta de soporte al aprendizaje

Joan Muñoz Riera

**Resumen–** El nuevo contexto de aprendizaje presenta como uno de sus retos principales la observación más cercana del progreso del alumno. Obtener feedback rápido y preciso puede ser clave para solventar fallos del aprendizaje. En este proyecto planteamos el desarrollo de una herramienta de soporte al aprendizaje que incluye funcionalidades de gamificación, con el objetivo de que ésta sea atractiva para el uso frecuente por parte de los estudiantes. Una de las claves de esta aplicación es el uso de aprendizaje computacional para poder plantearle al alumno preguntas similares en el caso de que no las responda correctamente y así reforzar sobre la marcha su aprendizaje. El resultado de este proyecto es una aplicación móvil completamente funcional que incluye módulos de gestión de usuarios (registro e ingreso), gestión de preguntas (creación y modificación) así como una primera versión completa de un juego.

**Palabras clave–** Gamificación, e-learning, Bag of Words, Aprendizaje Computacional, Bases de datos no relacionales, Unity

**Abstract–** The new learning context we are in presents as one of its main challenges the closest observation of student progress. Getting fast and accurate feedback can be key to solving learning failures. In this project we propose the development of a learning support tool that includes gamification functionalities, being one of its goals to be attractive for frequent use by students. One of the main features of this application is the use of machine learning in order to propose students similar questions in the event that they are not answered correctly, thus reinforcing the progress of their learning. The result of this project is a fully functional mobile application that includes user management modules (registration and login), question management (creation and modification) as well as a first full version of a game.

**Keywords–** Gamification, e-learning, Bag of Words, Machine Learning, Non relational databases, Unity

---

## 1 INTRODUCCIÓN

ESTE documento muestra el proceso de desarrollo de una aplicación de soporte al aprendizaje que emplea técnicas de gamificación y aprendizaje computacional. El objeto de esta herramienta es ayudar a los estudiantes

- E-mail de contacto: joan.muñozr@e-campus.uab.cat
- Mención realizada: Computación
- Trabajo tutorizado por: Jorge Bernal (Ciencias de la Computación)
- Curso 2019/20

a asimilar los conceptos del temario de la asignatura así como generar datos que puedan ser usados en el futuro para observar mejor el aprendizaje de los alumnos por parte de los docentes.

### 1.1. Situación Actual

Debido al cambio en el paradigma educativo en el cual la enseñanza se centra en el estudiante, es necesario realizar cambios en cómo éste se relaciona con el material docente del que dispone para enfrentar una asignatura. Actualmente los estudiantes basan su aprendizaje en el estudio y consul-

ta de powerpoints y documentos de texto pero no reciben feedback sobre el nivel de comprensión de conceptos hasta la realización de pruebas de evaluación. Del mismo modo, el profesor no sabe si los alumnos están asimilando los conceptos hasta los períodos de evaluación, no pudiendo corregir carencias del aprendizaje hasta que ya es tarde.

Mediante experiencias de innovación docente realizadas en cursos anteriores, se ha visto que la experiencia de aprendizaje es más dinámica en aquellas asignaturas que emplean aplicaciones móviles o plataformas web durante las clases para testear el nivel de conocimiento de los alumnos sobre un tema concreto [1].

Sí que es cierto que estas herramientas sólo se han empleado durante las clases, sin permitir al alumno reintentar aquellas preguntas que ha fallado o simplemente practicar con otras nuevas. En este proyecto trataremos el desarrollo de una aplicación móvil que permitirá al alumno poner a prueba sus conocimientos dentro y fuera del aula, ofreciendo partes de gamificación que esperemos le motiven a entrar en la aplicación de manera continua y, por tanto, mantener el interés en la asignatura.

En un contexto como el actual donde la enseñanza presencial va perdiendo peso (sobre todo en la situación reciente causada por la COVID-19), herramientas como la propuesta pueden usarse como un incentivo de cara al estudiante para que vaya auto-evaluando su progreso de manera continua.

## 1.2. Estado del arte

Existen muchas herramientas que permiten a los profesores interactuar con los alumnos. Por ejemplo, Kahoot! [2] (ver Fig. 1) y Socrative [3] (ver Fig. 1) se usan mayoritariamente durante las clases para resolver pequeñas cuestiones teóricas. Preguntados (ver Fig. 3) es un juego de estilo trivial sobre conocimientos generales que puede ser jugado en cualquier momento.

Kahoot! y Socrative son de fácil uso para el profesorado pero su utilidad se limita a las actividades realizadas en el aula, no permite a los usuarios tener un histórico de las preguntas realizadas (ni reintentárlas) ni integran ningún tipo de aprendizaje computacional para proponer preguntas personalizadas. Preguntados es una aplicación orientada a un ámbito de uso general con un componente competitivo pero, en este contexto, puede ser usado por los estudiantes para aumentar el nivel de conocimientos de categoría en la que se juega.

La aplicación que queremos desarrollar sería un híbrido entre Preguntados en el sentido de que integra funciones competitivas y puede ser usada en cualquier momento con las funcionalidades que da Kahoot! para poder obtener feedback durante la clase de manera instantánea. La novedad de Learn2Learn (L2L) radica en la inclusión de características de aprendizaje computacional de modo que los estudiantes tengan una experiencia de aprendizaje personalizada que evolucione según lo hagan ellos en la asignatura.

## 1.3. Objetivos del proyecto

El objetivo principal de este TFG es el desarrollo de una aplicación móvil (L2L) de soporte al aprendizaje que integra funcionalidades de gamificación y que emplea técnicas



Fig. 1: Kahoot!

Fig. 2: Socrative

Fig. 3: Preguntados

de aprendizaje computacional. Para poder llevar a cabo este objetivo, se definen los siguientes objetivos secundarios:

1. Diseño BBDD básica con usuarios y preguntas.
2. Diseño de la interfaz de usuario básica.
3. Diseño del módulo de acceso de usuarios.
4. Diseño del módulo para introducir de manera ágil y sencilla nuevo material docente en la aplicación.
5. Integración básica de herramientas de inteligencia artificial para el desarrollo de un sistema recomendación

de preguntas.

## 6. Diseño de un módulo de juego de prueba.

### 1.4. Herramientas Empleadas

La herramienta básica empleada para el desarrollo de la aplicación será Unity [4]. Las razones que han llevado a escoger esta herramienta es su gran versatilidad tanto de diseño como de dispositivos a exportar el proyecto y por la gran comunidad que tiene. Se exploraron otras alternativas como Unreal Engine [5] o Game Maker [6], pero se descartaron por las siguientes razones: Unreal fue descartado por su mayor curva de aprendizaje mientras que Game Maker fue descartado por su excesiva simplicidad.

Para gestionar la base de datos se ha escogido Firebase (Google), que tiene una API lista para usar con Unity. Dentro de Firebase usaremos RealTime DataBase [7] que es una Base de Datos no relacional. En este sentido, se exploró el uso de otros entornos de bases de datos como Amazon Web Services [8] pero en este caso se ha preferido escoger aquel entorno que permite una más fácil integración con la herramienta de desarrollo de videojuegos. Se han empleado bases de datos no relacionales [9] con el objetivo de permitir mayor flexibilidad a la hora de almacenar el material docente que se empleará en la aplicación.

### 1.5. Requisitos del proyecto

#### 1.5.1. Requisitos HW

Para usar Unity en Windows:

- OS: Windows 7 o Windows 10, 64-bit.
- CPU: Arquitectura X64 con soporte para instrucciones SSE2.
- GPU: Soporte para DX10, DX11, o DX12.

#### 1.5.2. Requisitos SW

- Unity 2019.3 [4] .
- Firebase SDK [10] .
- Chrome/Firefox para acceder a la consola de Firebase.

### 1.6. Requisitos funcionales

Requisitos funcionales: Hay dos tipos de usuarios principales en el sistema: el profesor y el estudiante. El profesor puede hacer lo siguiente:

- Acceder al sistema
- Introducir nuevo material docente
- Consultar resultados
- Revisar progreso individual de estudiante

El estudiante podrá hacer lo siguiente:

- Acceder al sistema
- Jugar a un juego

- Consultar resultados propios de juego
- Ver perfil usuario

La figura 4 muestra una versión general del diagrama de casos de uso de nuestra aplicación.

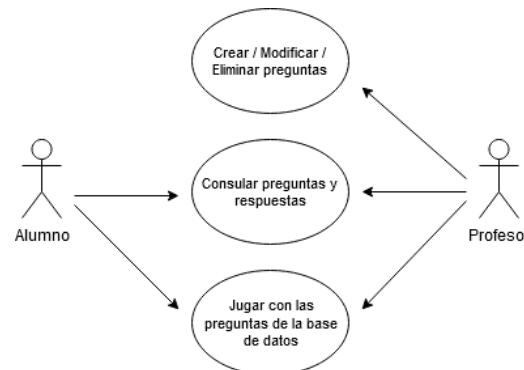


Fig. 4: Capacidades de los usuarios

### 1.7. Requisitos no funcionales

- Poder integrar el módulo de IA en Unity.
- Que el usuario pueda usar la aplicación en modo “offline”.
- Tiempo máximo de respuesta a la BBDD de 1 s
- Número máximo de conexiones permitidas a la vez de 300 usuarios
- La aplicación será responsive y tendrá modo claro
- La aplicación estará disponible en castellano

### 1.8. Riesgos del proyecto

En la TABLA 1 se muestran los riesgos del proyecto.

### 1.9. Metodología

Para realizar de manera correcta este proyecto se ha apostado por una metodología incremental que incluye reuniones semanales con el tutor. Debido a estos requerimientos, se ha optado, dentro del conjunto de metodologías Agile, por la metodología agile Kanban [11]. Esta metodología se basa en el desarrollo iterativo e incremental, donde los requisitos y soluciones evolucionan con el tiempo según la necesidad del proyecto. Esta metodología tiene como objetivo gestionar de manera general cómo se van completando las tareas.

### 1.10. Planificación del proyecto

El proyecto se ha realizado en las siguientes etapas temporales:

1. Diseño de la Base de Datos: se realizó un primer esquema de la base de datos teniendo en cuenta las funcionalidades previstas en el proyecto, dejando puerta a poder implementar fácilmente otras en el futuro.

TABLA 1: RIESGOS DEL PROYECTO

| ID | Descripción                                       | Efecto   | Posibilidad | Impacto | Contingencia                                       |
|----|---|--|-------------|---------|--|
| R1 | Mala planificación.                               | No llegar a desarrollar las tareas a tiempo.                       | Media       | Critico | Revisar la planificación y priorización de tareas  |
| R2 | Falta de disponibilidad de preguntas.             | Bajo rendimiento sistema recomendador, aplicación poco interesante | Media       | Medio   | Adquirir preguntas nuevas o cambiar de BBDD        |
| R3 | Falta de disponibilidad de recursos informáticos. | Imposibilidad de progresar con TFG                                 | Baja        | Crítico | Solicitar soporte para adquirir o reparar material |

2. Desarrollo de módulo de acceso de usuarios: para poder realizar pruebas con usuarios reales de L2L se implementará un sistema de registro y login en la plataforma web.
3. Diseño de interfaz de usuario: se realiza un primer diseño de la GUI en paralelo al módulo de acceso de usuarios para comenzar a integrar funcionalidades del sistema.
4. Diseño módulo de introducción y modificación de preguntas: basándonos siempre en el diseño acordado de la BBDD, se implementará la funcionalidad que permite revisar las preguntas ya existentes, dando la opción de modificarlas si así se considera así como la creación de preguntas nuevas.
5. Desarrollo de módulo de inteligencia artificial: se implementa un sistema de búsqueda de preguntas cercanas basado en Bag of Words que permita, en caso de fallo, proponer una cuestión parecida a la que no se ha acertado.
6. Implementación primer juego: a modo de demostración de las posibilidades de la plataforma creada, se implementa un primer juego de preguntas.

## 2 DESARROLLO FUNDAMENTAL DEL PROYECTO

Este documento muestra el desarrollo de la interfaz de usuario básica para poder iniciar sesión como usuario de la aplicación, tanto como alumno o profesor. También se incluye una explicación de como podemos gestionar los usuarios y como está estructurada (y se accede) a la base de datos, que son herramientas de Firebase (Aplicación de Google).

### 2.1. Estructuración del proyecto

Para organizar el proyecto se ha utilizado la estructura singleton [12], esta estructura nos permite garantizar una sola instanciación de los gestores, tanto del gestor de la aplicación como del gestor del menú y también de funcionalidades como la comunicación de usuarios mediante el módulo de autentificación de Firebase.

Tanto los gameobjects con estos gestores como el gameobject menú se intanciarán como `dontdestroyonload()`.

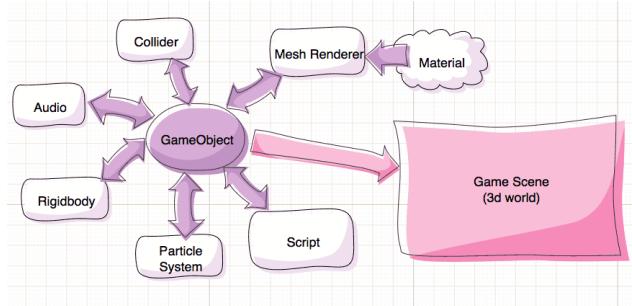


Fig. 5: Diagrama basico de los elementos del engine de Unity [14].

Esto permitirá que no se destruyan al cargar nuevas escenas y así poder movernos entre escenas que contengan diferentes configuraciones o gameobjects, por ejemplo para un mini-juego, sin perder estos elementos, es decir para mantener los gestores y el menú en todo momento sin importar escenas.

#### 2.1.1. Interfaz de usuario

Unity es un engine que contiene diferentes escenas como se puede ver en la Fig. 5. Estas escenas representan un espacio en el cual se encuentran gameobjects [13] (elementos del juego) los cuales pueden ser tanto elementos 3d como 2d. En cada escena podemos tener diferentes distribuciones de gameobjects, y más adelante podemos instanciar las diferentes escenas en el momento que creamos oportuno.

Para añadirle funcionalidades a los gameobjects, se les añaden scripts creados por el usuario. Un gameobject es capaz de existir en la escena sin que ello suponga que aparezca asociado a él ningún elemento visible, es decir, forma parte de la escena pero es transparente al usuario.

A parte de gameobjects creados por el usuario, también hay elementos que forman parte del engine que nos aportan funcionalidades del mismo. Ejemplos de estos elementos son la cámara, sistemas de iluminación o el Canvas. El elemento Canvas [15] es un área donde se disponen los diversos componentes de la interfaz de usuario. Unity permite añadir al Canvas diferentes elementos como pueden ser botones, deslizadores, texto, etc. El Canvas nos permite indicar si queremos que se adapte a la resolución de pantalla o si, en cambio, ha de tener una dimensión fija. También nos permite seleccionar entre varios tipos de renderizado.

En nuestro caso, usamos Canvas para definir la interfaz de usuario a partir de la cual se crearán los diferentes menús.

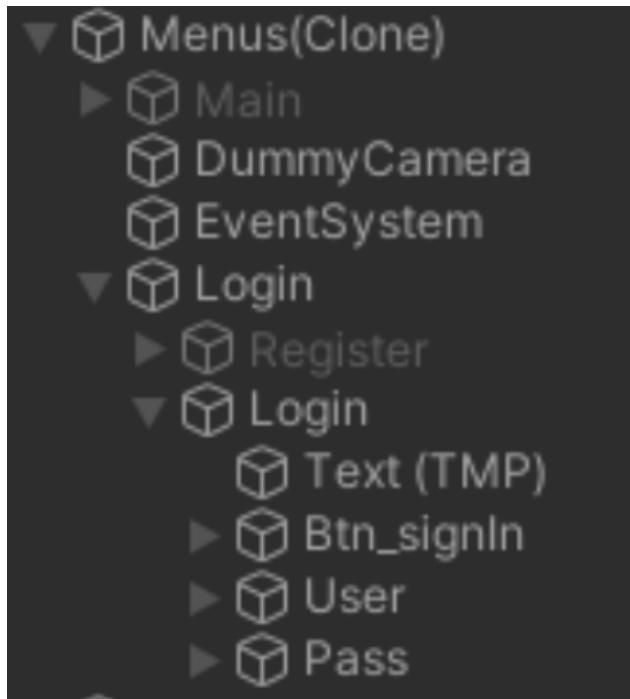


Fig. 6: Jerarquía del menú

Se ha creado un sistema de menús jerárquico que consiste en tres elementos diferentes:

- 1. Gestor del Menú
- 2. Grupo de Pantallas
- 3. Pantalla

Definimos pantalla como un elemento básico que permite mostrar información al usuario. Este elemento nos permite implementar diferentes tipos de pantallas: desde una pantalla estática hasta una pantalla que se muestre durante un tiempo determinado y sirva de transición.

El grupo de pantallas nos permite tener un control sobre las pantallas que pertenecen a él, para añadir funcionalidades como 'volver a la pantalla anterior'.

El gestor del menú (MenuManager) une todos los grupos de pantallas y contiene referencias a otros gestores, como por ejemplo el gestor del juego o el gestor de autenticación de usuarios. Este elemento nos permite añadir las funcionalidades que afectan a los distintos grupos de pantallas. Por ejemplo, si queremos cambiar un grupo de pantallas por otro, la función no estará llamada por el grupo que se cambia sino que será gestionada por el Gestor del Menú.

El gestor del menú 'Menus' en la jerarquía de la Fig.6, gestiona los grupos de pantallas 'Main' y 'Login', y en el caso del último contiene las pantallas 'Register' y 'Login'. Como se puede ver en la figura, 'Menus' contiene asimismo otros dos elementos: DummyCamera y EventSystem. El primero es necesario para poder visualizar las pantallas mientras que el segundo es el encargado de la gestión de eventos tales como introducción de datos por parte del usuario o simplemente recoger los clics del ratón. Ambos son imprescindibles para el funcionamiento de nuestro sistema.

## 2.2. Diseño de la base de datos

Para la aplicación se usará la RealTime Database de Firebase, una base de datos no relacional. El hecho de usar una base de datos sea no relacional nos aporta flexibilidad y escalabilidad. Esta base de datos además es accesible sin conexión dado que el SDK de Firebase Realtime Database hace que los datos persistan en el disco.

Para el proyecto que estamos presentando, se ha usado como base una serie de preguntas tipo test proporcionadas por el tutor del proyecto. Para poder interactuar con su contenido, se ha realizado una modificación de los datos para adaptarlos al formato JSON necesario para interactuar con Firebase.

La estructura inicial de la base de datos sería la siguiente:

- Temas
  - TemaA
    - Preguntas
      - 1. PreguntaA
        - ◊ Enunciado
        - ◊ Respuestas
      - 2. PreguntaB
    - TemaB
      - Preguntas
        - ◊ PreguntaA
        - ◊ PreguntaB

Esta estructura se transforma de modo que sea más acorde a una base de datos no relacional donde se evita el uso excesivo de indexaciones:

- Temas
  - TemaA
  - TemaB
- Preguntas
  - PreguntaA
  - PreguntaB

Una vez modificada la estructura de datos, estos se importarán a la plataforma de Firebase (RealTime Database) para más adelante poder acceder a esos datos.

## 3 DESARROLLO APLICACIÓN L2L

Llegados a este punto, ya empezamos a diseñar la aplicación final. Esta constará de los siguientes módulos:

- Registro y Autentificación de usuario (Módulo de Gestión de Usuarios)
- Creación, Eliminación y modificación de preguntas (Módulo de Gestión de Preguntas)
- Módulo juego contra la máquina

Respecto al último módulo, se han implementado actualmente las funcionalidades auxiliares de crear pregunta desde cero y crear pregunta a partir de una existente. Para poder tener esta última funcionalidad, se ha diseñado un submódulo que permite ver todas las preguntas existentes en la base de datos, así como la distinción de usuario, para determinar si es alumno o profesor.

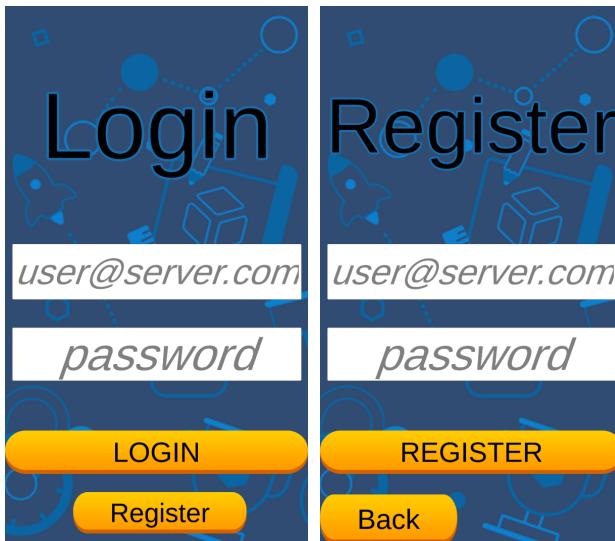


Fig. 7: Resumen interfaz usuario. La figura de la izquierda muestra la pantalla de login (pantalla inicial de la aplicación) mientras que la de la derecha muestra la de registro.

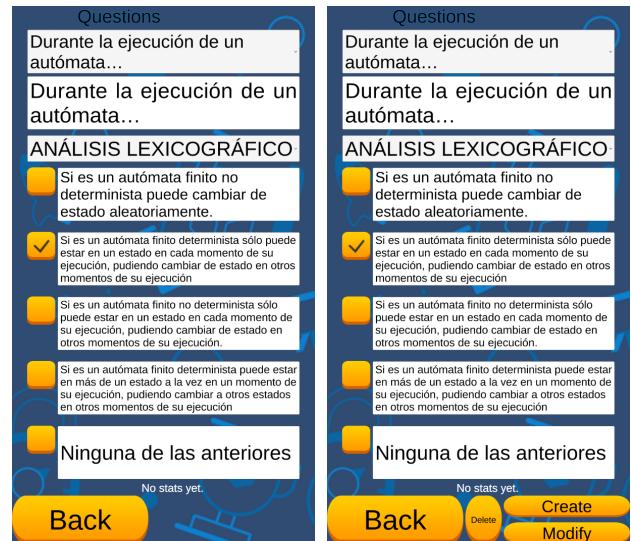


Fig. 8: Funcionalidades de gestión de preguntas. La figura de la izquierda muestra la interfaz de selección de preguntas para alumnos mientras que la de la derecha muestra la misma vista pero para el perfil de profesor.

### 3.1. Módulo de gestión de usuarios

Firebase ofrece un módulo de autenticación desde el cual se pueden gestionar los usuarios de nuestra aplicación. Este entorno permite elegir distintos tipos de autenticaciones entre las cuales se encuentran: e-mail, teléfono, cuenta de google, etc. Asimismo nos permite añadir restricciones como que solo exista un único e-mail por cuenta, y gestionar la verificación de correo o el cambio de contraseña.

Con el fin de comunicar Firebase con nuestra aplicación es necesario generar un script que haga de pasarela. En nuestro caso hemos usado como ejemplo el proyecto de autenticación de Firebase para Unity [16]. Dicho proyecto contiene los scripts para comunicarnos con el modulo de autenticación de Firebase, incluyendo funciones de acceso y de registro. Nuestro trabajo aquí ha sido integrar dichas funcionalidades en nuestro proyecto de Unity.

En nuestro proyecto usaremos la estructura de Canvas mencionada anteriormente para incluir los scripts que realizarán las distintas funcionalidades. Una de ellas es la gestión de usuarios. En este caso, el gestor del menú puede implementar funcionalidades que necesiten de la interacción entre las diferentes pantallas y relacionarlas con scripts que tengan otras funciones, como por ejemplo un script generado por el usuario que gestione un mini-juego. En nuestro caso, el gestor del menú implementa las funciones de autenticación en las cuales recibe los datos recogidos de la pantalla, como puede ser el input del usuario (nombre y contraseña), y estos son enviados al script de autenticación de Firebase para comunicarse con el módulo de autenticación del servicio de Firebase.

Para ello, hemos de integrar los paquetes correspondientes del SDK de Firebase, en este caso el paquete de autenticación, siguiendo la guía [10].

Usando las funcionalidades de Firease se ha implementado que el usuario pueda hacer loguin y registro usando un correo y una contraseña.

Mantendremos una interfaz similar tanto para login Fig. 7 (a) como para registro Fig. 7 (b). En este caso requerire-

mos que el usuario introduzca su correo y la contraseña para la misma. Una vez el usuario presione registrar”se creará un usuario en la plataforma de Firebase y se le hará login en la misma aplicación. Firebase se ocupa de comprobar que no existe otro usuario con el mismo e-mail y de hacer hash de las contraseñas para que solo el usuario tenga conocimientos de la misma.

Se ha elegido e-mail y contraseña ya que de esa manera se pueda elegir el correo institucional de la facultad y poder gestionar las materias a las que tiene acceso el usuario.

Una vez el usuario se ha registrado, podrá hacer login con su usuario (e-mail) y contraseña, siendo esta pantalla será la que recibirá al usuario al abrir la aplicación en el caso de que no lo haya hecho antes o haya decidido explicitamente hacer login. En este caso, la aplicación retiene los datos de login y en caso de cerrar y abrir la aplicación, esta misma gestionará el login.

### 3.2. Módulo de Gestión de preguntas

Para la gestión de preguntas se ha creado una pantalla en la cual si el usuario tiene perfil de alumno se le mostrará un desplegable en el cual el alumno podrá seleccionar cualquier pregunta de la base de datos, ordenadas por tema, y al seleccionar una de ellas podrá observar las respuestas, tanto correctas como incorrectas. En versiones futuras de la aplicación, sólo podrá acceder a las preguntas que ya haya contestado, a modo de herramienta de repaso.

En caso de que el usuario se trate de un profesor entonces en esa misma pantalla tendrá también acceso a todas las preguntas ordenadas por tema y además después de seleccionar tendrá las opciones de:

- Eliminar: Elimina la pregunta completamente de la base de datos, modificar
- Modificar: Modifica los datos de la pregunta y sobre-escribe la pregunta en la base de datos.
- Crear: Crea una nueva entrada en la base de datos.

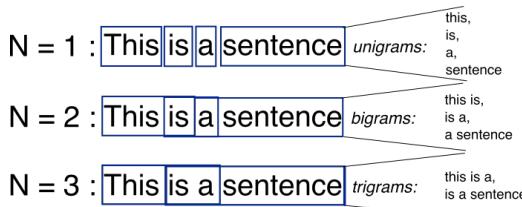


Fig. 9: Así es como quedaría cada bag of words para diferentes n-grams. [18]

Para facilitar la gestión de preguntas todo ello se hace en la misma pantalla, en la cual se puede editar la pregunta seleccionada o se puede crear una nueva pregunta a partir de la base de la pregunta seleccionada. De esta manera facilitamos la creación de nuevas preguntas y la creación de variantes de las mismas.

También se muestran las estadísticas para la pregunta seleccionada, las veces que se ha respondido correctamente y las veces que se ha respondido incorrectamente. La figura 8 muestra ejemplos de las vistas del gestor de preguntas tanto para el alumno como para el profesor.

### 3.3. Algoritmo de búsqueda de preguntas similares

Con el fin de poder plantear al jugador una pregunta similar a la fallada se ha seguido una estrategia de caracterización de preguntas basada en la técnica Bag of Words [17]. En este caso se ha creado un diccionario que contiene todas las preguntas de la base de datos y, para caracterizar cada una de ellas, se han descompuesto en n-gramas. Un n-grama es una subsecuencia de n elementos de una secuencia dada; en nuestro caso la secuencia será la pregunta y el conjunto de n-gramas serán las diferentes combinaciones de palabras que contiene esa pregunta.

Para determinar que preguntas son cercanas.<sup>a</sup> otras hemos creado un histograma de n-grams para cada pregunta, el cual separa las frases en n palabras de las cuales hemos quitado los artículos y preposiciones. Usando la distancia de intersección de histogramas [19] podemos seleccionar las preguntas más cercanas.<sup>a</sup> la pregunta fallada por el usuario.

Con el fin de facilitar la implementación en el juego y evitar que el usuario tenga que correr el sistema de clasificación en su dispositivo, en esta primera versión de la aplicación se ha incluido un atributo ligado a cada pregunta que indica las preguntas cercanas/parecidas a la misma (usando los identificadores únicos de las mismas). De este modo, en caso de que el usuario falle la aplicación pueda mostrarle preguntas similares y poder reforzar esa área de conocimiento.

Para este algoritmo hemos usado como distancia la intersección entre preguntas, y con ello se ha comprobado que n-grams daban mejores resultados empleando una métrica que comprueba la media de preguntas cercanas que pertenecen al mismo tema que la pregunta a la que hacen referencia. Hemos usado esta métrica ya que normalmente las preguntas más parecidas formarán parte del mismo tema.

Como se puede observar en la Fig 10 el mejor resultado se obtiene cuando ngram es igual a 1 así que eso significa que cogemos palabra por palabra, ya que al crear combina-

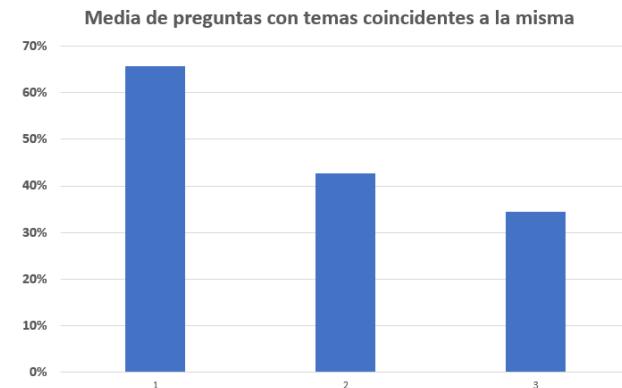


Fig. 10: Porcentaje de preguntas cercanas que pertenecen al mismo tema que la pregunta a la que referencian, variando ngrams.

ciones de 2 o 3 los resultados obtenidos descienden drásticamente.

Como ejemplo de funcionamiento de este módulo, veamos qué pasa si usamos como pregunta base “Un autómata finito determinista acepta una secuencia de caracteres cuando”. En este caso, L2L nos presenta como preguntas similares las siguientes (ordenadas según la similitud):

1. Un autómata finito acepta una secuencia de caracteres si ...
2. ¿ Cuándo un autómata acepta una secuencia de caracteres de entrada ?
3. Un autómata finito determinista esta formado por los siguientes elementos ...

Podemos comprobar como el sistema acierta al proponer preguntas diferentes cubriendo conceptos similares.

### 3.4. Módulo de juego

Una vez explicada la gestión de usuarios, pasamos a explicar en qué consiste la experiencia de aprendizaje gamificado. Se ha diseñado un sistema de juego sencillo en el cual el usuario va contestando preguntas y acumulando puntos (restándose puntos en caso de respuesta incorrecta y aumentando la cantidad de puntos a acumular en caso de aciertos consecutivos).

El juego añade 200 puntos al usuario por responder una pregunta correctamente y resta 50 puntos en caso contrario, de esta manera se asimila a un examen tipo test real.

Para incentivar al usuario cada vez que se responde correctamente de manera consecutiva aumenta la racha lo cual hace que en vez de añadirse 200 puntos, se añade un multiplicador el cual aumenta por un 25 % cada vez. En el caso de que falle una pregunta, dicha racha se corta. Aparte de mostrar la puntuación obtenida por el jugador, se muestra a los usuarios una tabla de puntuaciones Fig. 11 para despertar el espíritu competitivo de los mismos.

Como se observa en la Fig. 12 el usuario ha respondido 5 preguntas, 3 correctamente (en verde) y 2 incorrectamente (en rojo). Encima de las preguntas respondidas correctamente se muestra la racha del usuario en ese momento, y podemos inferir que ha respondido correctamente la pregunta anterior. A la derecha de la racha del usuario se muestran

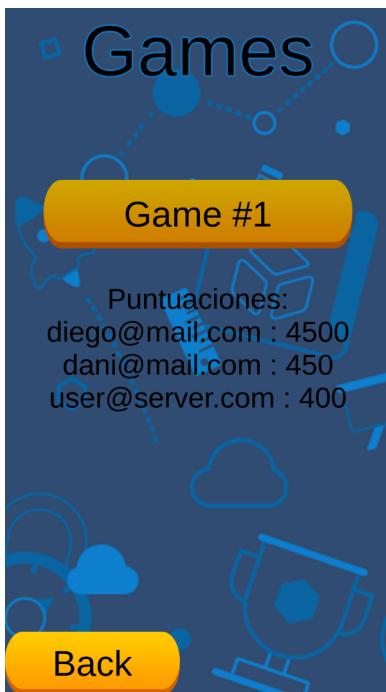


Fig. 11: Pantalla de selección de juegos en la cual se pueden observar las puntuaciones

los puntos adquiridos de la partida; el usuario puede parar en cualquier momento, y no hay límite de preguntas a responder.

Cada vez que el usuario responde una pregunta se tiene en cuenta si ha sido correcta o incorrectamente para mostrarle otra pregunta, en caso correcto se le mostrará una pregunta aleatoria de la base de datos y en caso incorrecto se le mostrará una pregunta similar a la que ha respondido para que el usuario refuerce sus conocimientos en ese ámbito.

En las siguientes partes del informe explicamos la herramienta de aprendizaje computacional de soporte que se ha empleado. Dicha herramienta se incluye para favorecer el aprendizaje del estudiante a base de mostrarle (en caso de fallo) preguntas similares a la misma.

#### 4 DEMOSTRACIÓN DEL JUEGO

En el siguiente enlace se muestra un vídeo demostrando las funcionalidades básicas de L2L. El vídeo incluye aspectos como la gestión de usuarios así como enseña el desarrollo de una breve partida de juego. Finalmente muestra cómo funciona el módulo de gestión de preguntas.

<https://youtu.be/0-Tmxz881bc>

#### 5 CONCLUSIONES

Este proyecto ha servido para poder aplicar los conceptos aprendidos a lo largo de la carrera, ya que ha sido un esfuerzo tanto por aprender nuevas cosas como por aplicar el conocimiento previo. También se ha podido desarrollar lo que puede ser el principio de una herramienta docente que ayude a reforzar el aprendizaje autónomo por parte de los estudiantes, y que también ayude a los docentes a saber qué partes del temario es más complejo para sus estudiantes.

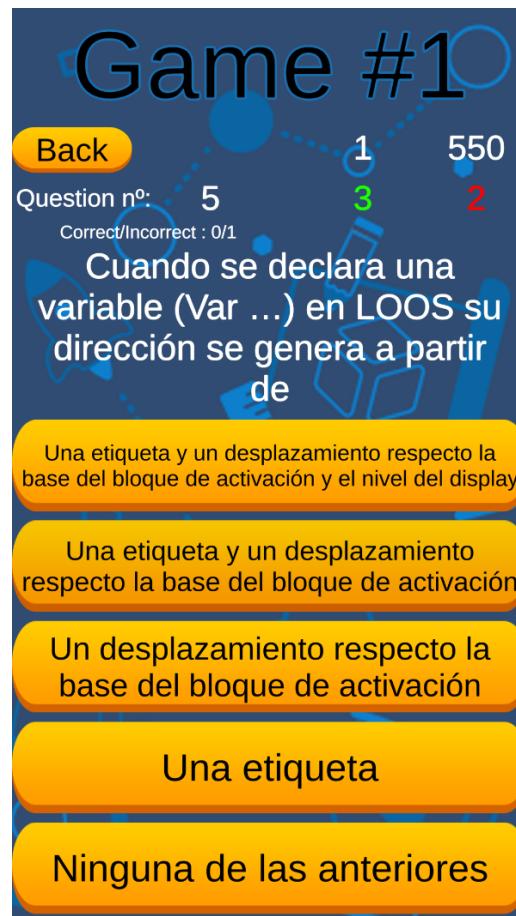


Fig. 12: Pantalla de juego en la cual se han respondido multiples preguntas

#### 5.1 Conclusiones generales del proyecto

Con esta primera versión de la aplicación se ha logrado todo lo propuesto, que se puedan registrar los nuevos alumnos y que estos puedan tanto consultar las preguntas como demostrar sus conocimientos jugando. Dentro de la parte más lúdica de la aplicación, el alumno puede retarse a sí mismo para evaluar su nivel de conocimientos y se ve recompensado su esfuerzo mediante un sistema de puntuaciones y un ranking competitivo que le incite a seguir usando la aplicación. También hemos conseguido que los profesores de las mismas materias puedan consultar qué partes del temario presentan más dificultades a los alumnos y así poder actuar de cara a las siguientes clases y poder resolver dudas de una forma más eficaz.

También hemos podido incorporar un algoritmo que permite que el alumno refuerze sus conocimientos de manera personalizada y eficaz, mostrándole preguntas similares a las que falle. De esta manera podemos presentar al usuario con un reto que se ajuste a sus necesidades.

#### 5.2 Revisión de la planificación

Durante el proyecto han surgido muchos problemas en cuanto al engine de Unity ya que se ha tenido que aprender el funcionamiento del mismo mientras se desarrollaba la misma aplicación. Ésto ha comprendido aprender desde cómo diseñar una interfaz de usuario y hacerla adaptable a los diferentes tamaños de pantalla hasta cómo gestionar los

diferentes menús. No solo eso sino que también se ha tenido que tener en cuenta la gestión de escenas, aunque en esta primera versión no se acabe de utilizar completamente.

También han surgido problemas con la unión de Firebase a Unity ya que tampoco se tenían conocimientos de como integrar correctamente Firebase a un engine que gestiona los procesos diferente de como lo puede hacer un servicio web. No solo eso sinó que uno de los mayores problemas ha sido el de los datos recibidos en Unity por parte de Firebase ya que no queda claro en la documentación cómo hacer el tratamiento de los mismos de manera correcta, y dependiendo de cada tipo de dato se tiene que hacer una serie de conversiones de tipos de datos desde algunos más abstractos a otros más restrictivos hasta que se llega a aquellos que se querían extraer.

Gran parte de los problemas de Unity se han podido solucionar a la documentación y tutoriales (tanto oficiales como no oficiales), dado que Unity tiene una gran comunidad que lleva desarollando nuevas aplicaciones desde hace años.

Los problemas de Firebase se han solucionado con un conocimiento básico del uso de la consola del engine de Unity que permite hacer Logs en los scripts que se van ejecutando, pero mayormente ha sido ensayo y error, en algún momento llegando a costar más tiempo del que habría sido estimado en el planteamiento de las tareas. Hay que añadir que la documentación y tutoriales oficiales sobre la integración de Firebase a Unity son escasos y la mayoría de tutoriales no oficiales son muy específicos a un uso concreto de la herramienta.

### 5.3. Aprendizaje personal

Ha sido todo un reto desarrollar una aplicación con el engine Unity y conectarlo con Firebase pero ha sido muy enriquecedor en cuanto a aprendizaje. No sólo eso sino que siento que este proyecto puede llegar a ser una herramienta que forme parte del inicio del uso generalizado de herramientas del estilo en el aprendizaje de los alumno en un furuto cercano.

Realizar este TFG me ha permitido integrar conocimientos adquiridos en diversas de la carrera (Bases de Datos, asignaturas de programación, Tecnologías de Desarrollo para Internet y Web, Aprendizaje Computacional e IA) lo cual ha sido muy importante para mí ya que me ha permitido asentar todo lo que he ido aprendiendo a lo largo de los años y ver cómo puedo usarlo para realizar un proyecto completo por mi mismo.

### 5.4. Trabajo futuro

Dado que esta es la primera versión de L2L, está claro que queda mucho camino con el cual ampliarla y mejorarla. Algo principal es ampliar la base de datos, es decir, añadir más contenido, nuevas matérias y más preguntas para las mismas, así como permitir la adición de imágenes para las preguntas y comentarios que puedan explicar por qué la respuesta es la correcta.

Otro campo de estudio es mejorar las estadísticas que se ofrecen dado que actualmente solo podemos saber las veces que se ha respondido correctamente e incorrectamente una pregunta globalmente, pero se podría mejorar para mostrar

estadísticas por usuario, para así ofrecer un aprendizaje que se adapte más al alumno.

De cara a integrar más a los alumnos en la aplicación, se podría proponer un sistema para que los mismos alumnos puedan introducir preguntas y que si son aprobadas por un profesor de la matéría estas se añadan a los mini-juegos. Esto sería premiado al alumno con puntos y lo profesores podrían aumentar la base de datos preguntas solo teniendo que revisarlas y/o modificarlas.

Asimismo, para que los alumnos se vean más implicados en la aplicación se puede implementar un sistema de transacción entre puntos y cósmeticos/ventajas virtuales (mini-juegos) para el alumnado. Así se propicia la obtención de puntos poniendo a prueba sus conocimientos y el uso de la aplicación entre el círculo de compañeros de la misma matéría.

De cara a su uso masivo dentro de un curso, habría que estudiar más profundamente las transacciones de obtención y modificación de datos para la base de datos, ya que eso es algo muy importante para que la aplicación pueda soportar a todo el alumnado concurrentemente.

Finalmente y de cara a favorecer el uso de la herramienta para el estudio y repaso, se podría plantear una versión off-line de la misma mediante la precarga de temas en el dispositivo de uso, en este caso sin las funcionalidades completas de gamificación.

## AGRADECIMIENTOS

Me gustaría expresar mi más sincero agradecimiento a mi tutor Jorge Bernal que me ha estado ayudando desde el inicio del proyecto motivándome y ofreciéndose a resolver cualquier duda que tuviera; sin él este TFG no hubiera sido posible. También a Diego Velázquez uno de mis compañeros que me ha ayudado cuando numpy se mostraba reticente. Y también gracias a mi familia y amigos que han mostrado su apoyo y han estado testeando la aplicación.

## REFERENCIAS

- [1] F. J. Sánchez and J. Bernal, “Use of software tools for real-time monitoring of learning processes: Application to compilers subject.,” in *4th International Conference on Higher Education Advances (HEAD’18)*, pp. 1359–1366, Editorial Universitat Politècnica de València, 2018.
- [2] Kahoot!, “Kahoot! webpage.” <https://kahoot.com/>.
- [3] Socrative, “Socrative webpage.” <https://socrative.com/>.
- [4] Unity, “Unity engine.” <https://unity.com/>, 2019. [Online; Setiembre-2019].
- [5] U. Engine, “Unreal engine webpage.” <https://www.unrealengine.com>.
- [6] Y. Games, “Gamemaker webpage.” <https://www.yoyogames.com/gamemaker>.
- [7] Google, “Firebase realtime database.” <https://firebase.google.com/docs/database/>.

- [8] Amazon, “Amazon base de datos relacional.”  
<https://aws.amazon.com/es/rds/free/>.
- [9] Wikipedia, “Base de datos no relacional.”  
<https://en.wikipedia.org/wiki/NoSQL>.
- [10] Google, “Firebase.”  
<https://firebase.google.com/docs/unity/setup>, 2019.  
[Online; Noviembre-2019].
- [11] G. S. Matharu, A. Mishra, H. Singh, and P. Upadhyay, “Empirical study of agile software development methodologies: A comparative analysis,” *ACM SIGSOFT Software Engineering Notes*, vol. 40, no. 1, pp. 1–6, 2015.
- [12] Wikipedia, “Singleton” wikipedia.”  
<https://es.wikipedia.org/wiki/Singleton>.
- [13] Unity, “Unity gameobject.”  
<https://docs.unity3d.com/es/current/Manual/GameObjects.html>.
- [14] M. Todorov, “Diagrama elementos engine unity.”  
<https://www.raywenderlich.com/3038-how-to-make-a-2-5d-game-with-unity-tutorial-part-2>.
- [15] Unity, “Unity canvas.”  
<https://docs.unity3d.com/es/current/Manual/UICanvas.html>.
- [16] Google, “Firebase.”  
<https://firebase.google.com/docs/samples/>, 2019.  
[Online; Noviembre-2019].
- [17] Wikipedia, “Bag of words.”  
[https://en.wikipedia.org/wiki/Bag-of-words<sub>m</sub>odel](https://en.wikipedia.org/wiki/Bag-of-words_model).
- [18] recognize-speech.com (deprecated), “Frase en ngrams.” <http://recognize-speech.com/language-model/n-gram-model/comparison>.
- [19] @crossentropy, “Distancia de historigramas.”  
<http://blog.datadive.net/histogram-intersection-for-change-detection/:text=Histogram intersection,it can be calculated by.>

## APÉNDICE

La figura 13 muestra la planificación del proyecto.

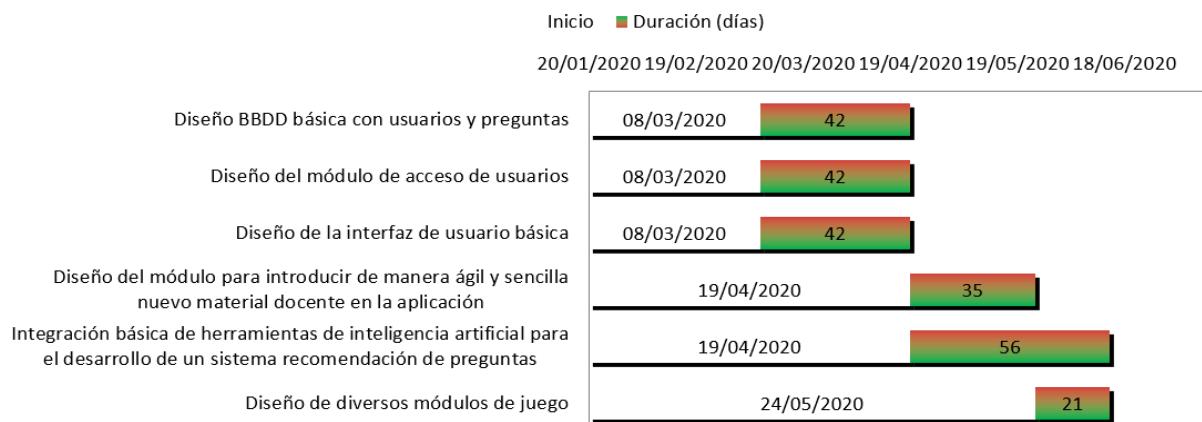


Fig. 13: Planificación implementaciones