

# **MEMORIA TFG**

## **REUNIÓN INICIAL (24-10-2024)**

En la reunión inicial del proyecto de TFG EducaQuest, sentamos las bases de lo que queríamos que fuera la aplicación. Decidimos que se trataría de un juego dividido en niveles, donde cada nivel contendría tareas o deberes específicos que los usuarios tendrían que completar, como resolver 5 multiplicaciones en un día. La idea principal era que estos ejercicios sumaran puntos, de modo que al final del periodo establecido, el usuario con más puntos ganaría un premio. Este enfoque buscaba combinar el aprendizaje con la diversión, incentivando a los estudiantes a participar activamente.

Durante esa primera reunión, también surgieron preguntas clave que debíamos resolver para seguir avanzando. Discutimos temas como la edad del público objetivo de la aplicación, si el premio para el ganador sería monetario o de otro tipo, y si la plataforma se desarrollaría como una aplicación web o una aplicación móvil. Estas cuestiones eran fundamentales para definir mejor el alcance y los recursos necesarios para el proyecto.

También, durante esta primera reunión, se me sugirió desde la parte de HP (Jaime) que fuera pensando con qué lenguaje quería hacer mi aplicación, por lo tanto, tendría que pensar con qué haría el front-end y el back-end.

## **AVANCE DE ESTA REUNIÓN:**

En la primera reunión se me había pedido que investigara sobre lenguajes para aplicaciones, y tras hacerlo, y buscar información sobre aplicaciones similares a la que se desean desarrollar, decidí usar Angular como framework para el front-end. Además de ser ampliamente utilizado en proyectos de gran envergadura, observé que otras aplicaciones con características parecidas habían optado por Angular, lo que me dio más confianza en su capacidad para gestionar proyectos de este tipo. Otra razón que influyó en mi elección fue la experiencia de Jaime, mi tutor de HP, quien tiene un amplio conocimiento en este framework y podría proporcionarme apoyo técnico valioso. Entre las ventajas adicionales de Angular se encuentran su estructura bien definida y su capacidad para manejar aplicaciones complejas.

con alta interactividad. Me decanté por Angular en lugar de React debido a su enfoque más completo y sus herramientas integradas, así como frente a otros lenguajes por su sólida comunidad y documentación.

Durante la fase de investigación, decidí buscar aplicaciones educativas exitosas que pudieran ofrecer una referencia para el diseño y la funcionalidad de EducaQuest. Dos aplicaciones que me sirvieron de inspiración fueron Matific y Reading Eggs, ambas altamente valoradas por su capacidad para combinar el aprendizaje con la diversión, adaptándose al público infantil.

Matific es una aplicación interactiva que ofrece una amplia variedad de actividades matemáticas para estudiantes de primaria. Lo que me llamó la atención de esta aplicación fue su enfoque en la gamificación del aprendizaje. A través de juegos matemáticos, los estudiantes pueden resolver problemas en un entorno atractivo y dinámico, lo que fomenta la motivación y el compromiso. Algunas características de Matific que me inspiraron fueron:

- Gamificación: Los juegos tienen un componente de recompensas que mantiene el interés de los estudiantes, similar al sistema de puntos que implementaré en EducaQuest.
- Progresión por niveles: Los usuarios avanzan a través de diferentes niveles que se vuelven progresivamente más difíciles, lo que facilita la personalización del aprendizaje.
- Interactividad: Las actividades están diseñadas para involucrar a los estudiantes de manera activa, lo que asegura que no solo aprendan, sino que disfruten el proceso.



Reading Eggs es una plataforma centrada en el aprendizaje de la lectura y la escritura para niños. Se destaca por su enfoque basado en historias y actividades que motivan a los estudiantes a mejorar sus habilidades lingüísticas. Algunas de las características de Reading Eggs que me inspiraron incluyen:

- Aprendizaje adaptativo: La plataforma adapta las actividades en función del progreso del estudiante, ofreciendo un enfoque más personalizado.
- Recompensas y logros: Reading Eggs utiliza un sistema de premios para mantener a los estudiantes motivados, ofreciendo medallas, trofeos y otros incentivos.
- Interfaz amigable: La interfaz de la aplicación es visualmente atractiva y fácil de usar para los niños, un aspecto clave que también quiero implementar en EducaQuest para garantizar que sea accesible y disfrutable para estudiantes de primaria.



Ambas aplicaciones me ayudaron a comprender mejor cómo estructurar la experiencia del usuario en términos de accesibilidad, motivación y progresión en el aprendizaje. La combinación de elementos de gamificación, un sistema de recompensas y una progresión clara son aspectos fundamentales que se desean integrar en EducaQuest para asegurar que los estudiantes no solo aprendan, sino que disfruten y se sientan motivados a seguir utilizando la aplicación.

#### REUNIÓN Nº 2 (31-10-2024)

En la segunda reunión, realizada después de un primer sprint de una semana, se me asignó la tarea de investigar y seleccionar un lenguaje de back-end para la próxima reunión. Además, se me pidió que comenzara a practicar con ese lenguaje para asegurar un desarrollo más fluido y eficiente en las próximas etapas del proyecto.

En esta reunión, la Asociación Montemadrid ayudó a despejar diversas dudas. Confirmaron que la aplicación estaría enfocada en estudiantes de 5º y 6º de primaria, asegurando que la experiencia educativa se adaptara a este rango de edad. También se definió que el premio no sería monetario, sino una experiencia especial para motivar a los participantes. Por último, se aclaró que la aplicación debía ser web responsive, lo que permitiría su uso tanto en ordenadores como en dispositivos móviles, ampliando así su accesibilidad y versatilidad.

## AVANCE DE ESTA REUNIÓN:

Tras investigar sobre posibles opciones para realizar la parte de Back-End, he elegido Node.js para el desarrollo de mi aplicación, porque, junto con Angular, constituye una combinación poderosa que ofrece varias ventajas para el desarrollo de aplicaciones web modernas.

Node.js es un entorno de ejecución de JavaScript basado en el servidor, que se destaca por su rendimiento y escalabilidad, lo que lo convierte en una excelente opción para manejar múltiples usuarios simultáneamente, como en el caso de EducaQuest, que tendrá muchos estudiantes haciendo sus tareas a la vez. Al utilizar Angular para el front-end, que es un framework robusto y eficiente, y Node.js para el back-end, puedo mantener todo el proyecto en JavaScript, lo que facilita la comunicación entre ambos componentes y mejora la eficiencia del desarrollo.

En cuanto a la base de datos, he optado por usar una base de datos relacional, como MySQL, para gestionar los datos de los usuarios de manera estructurada. Los datos que se desean almacenar en EducaQuest incluyen información crítica sobre los alumnos, profesores y tutores, y las bases de datos relacionales son ideales para manejar este tipo de información. Las bases de datos relacionales permiten estructurar los datos de manera clara, utilizando tablas interconectadas, lo que facilita la gestión de usuarios y sus roles. Además, garantizan la integridad de los datos, lo cual es fundamental para asegurar que no haya datos inconsistentes o perdidos.

Por ejemplo, puedo tener tablas separadas para usuarios, roles (alumno, profesor, tutor), tareas y puntuaciones, con claves primarias y foráneas que aseguren que la información se relacione de forma correcta y eficiente. Esto también facilita la realización de consultas complejas, como obtener el progreso de un alumno en función de sus tareas completadas, lo cual es clave para el seguimiento del aprendizaje en una plataforma educativa.

## REUNIÓN Nº 3 (13-11-2024)

Para el tercer sprint del proyecto se me ha encomendado desarrollar un “happy path” básico que permita implementar un inicio de sesión funcional, conectando las tres capas del sistema: el frontend (Angular), el backend (Node.js) y la base de datos (MySQL).

En el frontend, se deberá crear una interfaz que permita al usuario ingresar sus credenciales (correo y contraseña), las cuales serán enviadas al backend. En el backend, se validarán las credenciales y se gestionará la conexión con la base de datos, donde los datos del usuario se almacenarán o validarán en una tabla llamada “usuarios”. Dicho inicio de sesión será un prototipo a la espera de ver cómo funcionará la privacidad del usuario, información que aún no conocemos de parte de la asociación.

Además, se me ha solicitado diseñar un croquis o esquema de la base de datos, que incluya las tablas necesarias y sus respectivas relaciones, para definir cómo estarán conectadas entre sí.

También debo crear y organizar las tareas específicas de cada sprint en el repositorio de GitLab del proyecto, desglosándolas en subtareas claras y manejables para focalizar mejor mis objetivos. Esto me permitirá estructurar el desarrollo de forma eficiente, priorizar tareas y cumplir con los objetivos establecidos para cada sprint.

### AVANCE DE ESTA REUNIÓN:

Implementé una interfaz funcional en Angular que permite al usuario ingresar sus credenciales (correo y contraseña).

En el backend (Node.js), desarrollé las rutas necesarias para validar las credenciales recibidas y conectarlas con la base de datos.

Sin embargo no conseguí establecer conexión entre el Front-End y el Back-End correctamente, por lo que utilicé la aplicación “Postman” para simular el envío de los datos

del formulario del Front-End y así comprobar si mi Back-End funciona correctamente, así como mi base de datos.

En cuanto a la base de datos, elaboré un croquis detallado del modelo de base de datos, incluyendo la tabla “usuarios” con campos como id, nombre, correo, contraseña (encriptada) y fechas de creación/modificación.

Para la gestión de tareas en el repositorio GitLab, organicé y documenté todas las tareas del sprint en el repositorio, dividiéndolas en subtareas claras y manejables. Este avance ha sido útil puesto que ahora no se me olvidan las tareas que tengo que hacer, y las que completo, tan solo tengo que moverlas de la columna de “abiertas” a “cerradas” (acabadas).

#### REUNIÓN Nº 4 (28-11-2024)

Para esta reunión se me propuso acabar el “happy path” que no había conseguido terminar en el anterior sprint.

También se me propuso hacer un dibujo de la base de datos construida para que se viera gráficamente por todas las partes del proyecto.

Por último se me propuso hacer commits en el repositorio GitLab del proyecto para ver los avances en el código.

#### AVANCE DE ESTA REUNIÓN:

En esta reunión, se me propuso finalizar el "happy path" que no había logrado completar en el sprint anterior. Trabajé en ello y conseguí concluirlo satisfactoriamente, estableciendo la conexión básica entre el front-end, el back-end y la base de datos. Además, se me pidió realizar un diagrama de la base de datos para visualizar cómo se organizarían las tablas y sus relaciones. El diagrama fue completado y presentado a todas las partes con el objetivo de

recibir un feed-back y modificarlo si fuera necesario. Finalmente, también se me pidió hacer commits frecuentes en el repositorio GitLab para mantener un control adecuado del progreso del código.

#### REUNIÓN Nº 5 (19-12-2024):

En la reunión de este día, discutimos una importante mejora en el flujo de trabajo de la aplicación. Decidimos que, para cada tarea realizada por los alumnos, un tutor asignado sería el encargado de determinar si la respuesta es correcta o incorrecta. Esta decisión surgió como una medida para optimizar el proceso, ya que incluir la resolución de cada actividad en la base de datos podría generar problemas de interpretación o complicaciones con actividades cuya resolución tuviera un formato que no fuera fácilmente almacenable en bases de datos. Un ejemplo de esto podría ser el análisis sintáctico de una frase, que es difícil de representar en un formato adecuado para una base de datos.

Como consecuencia de esta decisión, se me encomendó la creación de un área de trabajo diferenciada para tutores y alumnos. Esta área estaría accesible mediante un login diferenciado, según el rol del usuario. En este nuevo espacio, los alumnos podrían insertar sus respuestas a las actividades, mientras que los tutores tendrían acceso a una lista con las respuestas de cada alumno por actividad, para poder evaluarlas y determinar su validez. Esta estructura permitirá una gestión más eficiente y simplificada de las tareas, mejorando la experiencia tanto para los estudiantes como para los tutores

#### AVANCE DE ESTA REUNIÓN:

Para este sprint complete satisfactoriamente la diferenciación de área de trabajo entre el alumno y el tutor.

Además, tras la presentación inicial del diseño de la base de datos, se me hicieron algunas sugerencias para modificarla ligeramente. Siguiendo esas indicaciones, realicé los ajustes necesarios y añadí la base de datos a MySQL Workbench.

Actualmente, estoy trabajando con varias tablas, además de la tabla "usuarios", lo que me permite gestionar de manera más estructurada los datos relacionados con los estudiantes, tutores, actividades y tareas.

En cuanto a mi experiencia con GitLab, me he familiarizado aún más con la plataforma. Ahora organizo mejor mis avances y realizo commits diferenciados en ramas de back-end y front-end, lo que me permite llevar un control más eficiente de los cambios en ambas partes del proyecto. Esta metodología me ha ayudado a organizar mejor el desarrollo y a asegurar que los avances en ambos lados del proyecto se mantengan separados y claramente documentados.