



Universidad
Zaragoza

Trabajo Fin de Grado

ESCAPE ROOM DIGITAL SOBRE NORMATIVA DE TFE

Digital Escape Room on TFE Regulations

Autor

D.Francisco Benedicto Garzarán

Directoras

Dra. D^a.Piedad Garrido

Picazo

Dra. D^a.Ana María López

Torres

Escuela Universitaria Politécnica de Teruel

2024

Agradecimientos

Finalmente, tras todo el camino recorrido, llego al punto culminante de esta carrera con la realización de mi Trabajo Fin de Grado. Este proyecto representa no solo el cierre de una etapa, sino también el esfuerzo acumulado durante años de estudio y dedicación.

Quiero agradecer profundamente a mis padres por su apoyo incondicional y su constante ánimo, que me ha permitido mantenerme enfocado y motivado para llegar hasta aquí.

A Piedad y a Ana María López, mis tutoras, quienes me han guiado pacientemente durante el desarrollo de este proyecto. Sus orientaciones y sugerencias han sido esenciales para lograr el mejor resultado posible, desde las primeras reuniones hasta la exploración de nuevas ideas para el trabajo futuro.

También quiero expresar mi agradecimiento a todos aquellos profesores que, a lo largo de estos años, me han brindado su conocimiento y apoyo. Gracias a ellos, he podido aprender y crecer tanto a nivel académico como profesional.

Por último, quiero dar las gracias a mis compañeros de clase, que me han ayudado a superar asignaturas y a mis amigos, quienes con su compañía hicieron más amenos estos años. Sin ellos, este recorrido habría sido mucho más difícil.

Este logro no sería posible sin el conjunto de todas estas personas, a quienes estaré siempre agradecido.

Resumen y Palabras Clave

Resumen

El presente Trabajo Fin de Grado (TFG) se enfoca en el diseño y desarrollo de un escape room digital educativo, cuyo objetivo es motivar al alumnado a conocer la normativa de los Trabajos Fin de Estudios (TFE) de la Escuela Universitaria Politécnica de Teruel (EUPT). Utilizando la gamificación, se busca aumentar la motivación y el compromiso de los estudiantes mediante la resolución de una serie de desafíos interactivos. La aplicación ha sido desarrollada utilizando tecnologías web modernas, incluyendo React para la interfaz de usuario, Node.js y Express para el backend, y Socket.IO para la comunicación en tiempo real. Además, se ha integrado H5P para la creación de contenido interactivo y Ren'Py para un desafío basado en novelas visuales. La funcionalidad principal incluye la creación y gestión de salas de juego, sincronización de jugadores y una clasificación que refleja el desempeño de los jugadores al completar todos los desafíos.

Palabras clave: gamificación, escape room, normativa, desafío, trabajo fin de estudios.

Abstract

This Final Degree Project (TFG) focuses on the design and development of a digital escape room aimed at motivating students to understand the regulations of Final Degree Projects (TFE) at the Polytechnic University School of Teruel (EUPT). Using gamification, the project seeks to increase students' motivation and engagement through the completion of a series of interactive challenges. The application has been developed using modern web technologies, including React for the user interface, Node.js and Express for the backend, and Socket.IO for real-time communication. Additionally, H5P has been integrated for creating interactive content, and Ren'Py is used for a visual novel-based challenge. The main functionality includes the creation and management of game rooms, player synchronization, and a leaderboard that reflects player performance in completing all challenges.

Keywords: gamification, escape room, regulations, challenge, final degree project.

Tabla de Contenidos

1 INTRODUCCIÓN Y OBJETIVOS.....	1
1.1 INTRODUCCIÓN	1
1.2 OBJETIVOS	2
2 ESTADO DEL ARTE.....	3
3 ANÁLISIS, DISEÑO E IMPLEMENTACIÓN	7
3.1 ANÁLISIS.....	7
3.1.1 <i>Requisitos Funcionales (RF)</i>	7
3.1.2 <i>Requisitos No Funcionales (RNF)</i>	8
3.1.3 <i>Tecnologías Utilizadas</i>	9
3.2 DISEÑO.....	12
3.2.1 <i>Diagrama UML del sistema</i>	12
3.2.2 <i>Concepto general de los desafíos</i>	14
3.2.3 <i>Diseño de la base de datos (DBD)</i>	20
3.3.3 <i>Diseño físico de la Base de datos</i>	22
3.4 IDENTIDAD CORPORATIVA	22
4 PLANIFICACIÓN.....	23
5 IMPLEMENTACIÓN	24
5.1 INICIALIZACIÓN Y CONFIGURACIÓN DE LA BASE DE DATOS.....	24
5.2 DESARROLLO DE LAS PANTALLAS	25
5.2.1 <i>Pantalla Home</i>	25
5.2.2 <i>Pantalla "Elige una Opción"</i>	26
5.2.3 <i>Pantalla "Crear Sala" - Implementación Técnica</i>	26
5.2.4 <i>Pantalla "Unirse a Sala" - Implementación Técnica</i>	28
5.2.5 <i>Pantalla Desafio1</i>	29
5.2.6 <i>Pantalla Desafio2</i>	31
5.2.7 <i>Pantalla Desafio3</i>	33
5.2.8 <i>Pantalla Desafio4</i>	35
5.2.9 <i>Pantalla Clasificación / Leaderboard</i>	37
5.2.10 <i>Pantalla Challenge Menu</i>	39
5.3 ARQUITECTURA DEL SERVIDOR.....	41
6 PRINCIPIOS HEURÍSTICOS DE USABILIDAD	43
7 LICENCIA SOFTWARE Y DOCUMENTAL.....	45
8 CONCLUSIONES Y TRABAJO A FUTURO	46
8.1 CONCLUSIONES	46
8.2 TRABAJO FUTURO.....	47
BIBLIOGRAFÍA	48
ANEXO.....	49
ANEXO A: CAPTURAS DE PANTALLAS DE LA APLICACIÓN	49
ANEXO B: PRUEBAS DE VISUALIZACIÓN EN DIFERENTES NAVEGADORES	59
ANEXO C: PREGUNTAS DEL DESAFÍO 3	62
ANEXO D: INTERACCIONES DESAFÍO 4 – PRESENTACIÓN DEL TFG	63
ANEXO E: IDENTIDAD CORPORATIVA	66
ANEXO F: REPRESENTACIÓN XML DEL ESQUEMA DE DATOS	68

1 Introducción y objetivos

En esta sección se presentarán los objetivos del proyecto, así como el contexto en el que se desarrolla. Se hará una introducción general a los conceptos clave y a las motivaciones detrás del uso de gamificación y escape rooms en el ámbito educativo.

1.1 Introducción

El Trabajo Fin de Grado (TFG) titulado “Escape Room Digital sobre Normativa de TFE” se enmarca en la búsqueda de métodos educativos innovadores que motiven a los estudiantes y faciliten su comprensión de las normativas académicas. En este contexto, la gamificación, entendida como la aplicación de elementos de diseño de juegos en contextos no lúdicos, se ha consolidado como una metodología eficaz para aumentar la motivación y el compromiso de los estudiantes. Entre las diversas estrategias de gamificación, los “escape rooms” digitales se destacan como una herramienta educativa de gran potencial.

Un “escape room” es un juego en el que los participantes deben resolver una serie de acertijos y desafíos dentro de un tiempo limitado para “escapar” de una habitación virtual. Esta metodología puede ser adaptada para enseñar contenidos educativos de manera interactiva y atractiva. El objetivo de este TFG es diseñar y desarrollar un “escape room” digital que motive a los estudiantes a familiarizarse con la normativa de los Trabajos Fin de Estudios (TFE) de la Escuela Universitaria Politécnica de Teruel (EUPUT).

La idea de desarrollar este Trabajo Fin de Grado surgió a partir de la propuesta de mi tutora, Piedad, quien buscaba una manera innovadora de facilitar a los estudiantes la comprensión de los pasos necesarios para realizar un TFG. Ana, por su parte, colaboró en el proyecto proporcionando orientación sobre los aspectos técnicos del proceso y cómo afrontar los desafíos del diseño. Juntas, han contribuido a hacer de este proyecto una herramienta interactiva que ayudará a los estudiantes a abordar el TFG de una forma más clara y atractiva.

El uso de la gamificación, y en particular de los “escape rooms”, en el ámbito educativo, no solo busca hacer el aprendizaje más atractivo, sino también desarrollar competencias clave como el trabajo en equipo, la resolución de problemas y el pensamiento crítico. A través de esta propuesta, se espera que los estudiantes no solo comprendan mejor la normativa académica, sino que también superen el temor y la desmotivación asociados con la revisión de este tipo de documentación, habilidades que serán fundamentales en su futuro profesional.

Esta memoria está estructurada en varios capítulos. El capítulo 1 presenta los objetivos y el contexto del proyecto. El capítulo 2 analiza el estado del arte, revisando trabajos previos relacionados con la gamificación y los “escape rooms” educativos. El capítulo 3 se centra en el análisis, diseño e implementación del “escape room” digital. El capítulo 4 aborda la accesibilidad y usabilidad de la plataforma desarrollada. El capítulo 5 trata sobre la licencia del software y la documentación. Finalmente, el capítulo 6 concluye con los resultados obtenidos y sugiere líneas futuras de trabajo. Además, en el Anexo se incluyen capturas de pantalla que ilustran el diseño y las funcionalidades de la aplicación, así como pruebas de visualización en distintos navegadores para asegurar la compatibilidad de la plataforma.

1.2 Objetivos

El principal objetivo de este Trabajo de Fin de Grado (TFG) es diseñar y desarrollar un “escape room” digital educativo que ayude a los estudiantes a comprender y familiarizarse con la normativa de los Trabajos Fin de Estudios (TFE) de la Escuela Universitaria Politécnica de Teruel (EUPUT). Los objetivos específicos son los siguientes:

- **Incrementar la motivación y el compromiso de los estudiantes:** Utilizar la gamificación para hacer el aprendizaje de la normativa más atractivo y participativo.
- **Fomentar el aprendizaje activo:** Diseñar desafíos y acertijos que promuevan la participación activa de los estudiantes y la aplicación práctica de sus conocimientos.
- **Facilitar la comprensión de la normativa:** Crear una plataforma que explique la normativa de manera interactiva, reduciendo la carga del profesorado y ayudando a los estudiantes a internalizar mejor los conceptos.
- **Desarrollar habilidades técnicas:** Implementar el “escape room” utilizando herramientas y tecnologías como React y H5P, mejorando las competencias técnicas del desarrollador del proyecto.
- **Diseñar una plataforma modular y escalable:** Crear un sistema que pueda integrar nuevos desafíos, adaptar los existentes y escalar fácilmente, garantizando la estabilidad y sostenibilidad de la solución en el tiempo. La integración de tecnologías como React y Socket.IO permitirá un funcionamiento fluido, colaborativo y accesible desde cualquier navegador.

A través de estos objetivos, el proyecto busca no solo mejorar el conocimiento de la normativa académica entre los estudiantes, sino también desarrollar habilidades críticas que serán esenciales en su futuro profesional. La implementación de este “escape room” digital se desarrollará dentro del proyecto PIPOUZ (Programa de Incentivación del Plan de Orientación Universitaria en la UZ) 2024 5247 con el título “Implantación de herramientas de gamificación para la dinamización del Plan de Orientación Universitaria en la Escuela Universitaria Politécnica de Teruel”.

2 Estado del Arte

Antes de profundizar en las implementaciones específicas de este proyecto, es fundamental realizar una revisión exhaustiva del estado del arte en el ámbito de la gamificación y, en particular, en el uso de los “escape rooms” como herramienta educativa. Para ello, se han llevado a cabo búsquedas en los principales repositorios de publicaciones científicas orientadas a la educación, como IEEE Xplore, Google Scholar y otros recursos académicos relevantes. Estas búsquedas se han centrado en cómo se manejan y plantean los contenidos educativos y en las tecnologías utilizadas para implementar estos entornos de aprendizaje innovadores. Este Trabajo Fin de Grado (TFG) se centrará específicamente en los escapes rooms dentro del marco de la gamificación educativa.

La literatura académica identifica varios beneficios de la gamificación en la educación:

- **Motivación Intrínseca:** La gamificación puede aumentar la motivación intrínseca de los estudiantes al proporcionar recompensas inmediatas y feedback positivo, tal como se destaca en las investigaciones de Deci y Ryan sobre la autodeterminación del comportamiento humano (1).
- **Aprendizaje Activo:** Según J. P. Gee, los estudiantes involucrados en procesos de aprendizaje gamificados tienden a participar activamente en su educación, lo que puede llevar a una mayor retención del conocimiento y una comprensión más profunda de los conceptos (2).
- **Colaboración:** K. M. Kapp subraya que muchas actividades gamificadas fomentan la colaboración, mejorando las habilidades sociales y de trabajo en equipo entre los estudiantes, lo cual es esencial para su desarrollo académico y profesional (3).

Los escapes rooms digitales son una forma de gamificación que ha demostrado ser particularmente eficaz en el ámbito educativo. En estos juegos, los participantes deben resolver una serie de acertijos y desafíos en un tiempo determinado para "escapar" de una habitación virtual. Esta metodología puede adaptarse para enseñar una variedad de contenidos educativos, promoviendo tanto el aprendizaje activo como la colaboración entre los estudiantes.

Estudios previos han explorado el uso de los escapes rooms en diferentes contextos educativos. Por ejemplo:

Cerdeira Porras diseñó un escape room para enseñar matemáticas a estudiantes de 1º de primaria. Este trabajo es un Trabajo Fin de Grado y los resultados mostraron un aumento significativo en la motivación y el interés de los estudiantes por la asignatura (4).

Checiches propuso un escape room para la clase de educación física, destacando la mejora en la motivación de los alumnos y la efectividad de esta herramienta para enseñar conceptos físicos de manera interactiva y divertida. Este también fue un Trabajo Fin de Grado (5).

Además, se han encontrado trabajos relevantes en conferencias y revistas internacionales:

En la revista Education Sciences, una publicación con un factor de impacto de 2.5, se examina cómo los escape rooms en la enseñanza de STEM pueden motivar a los estudiantes y mejorar su comprensión de los conceptos clave (6).

El congreso Frontiers in Education (FIE), clasificado como Clase 3 (Core B), es un evento de gran relevancia en el campo educativo, donde se ha explorado el impacto de los escapes rooms en la educación científica, concluyendo que estas actividades fomentan el pensamiento crítico y la colaboración, cruciales en las disciplinas científicas (7).

En la Tabla 1, se recogen diversos trabajos que exploran el uso de los escapes rooms en la educación, indicando el objetivo educativo, el tipo de tareas y el enfoque de los desafíos que se presentan en cada propuesta.

Trabajo	Objetivo Educativo	Tipo de Tareas	Enfoque de los Desafíos
Enseñanza de las matemáticas a través de la gamificación (4).	Matemáticas en primaria	Puzzles Matemáticos	Problemas de cálculo y lógica adaptados a niños
Revisión literaria científica y propuesta de escape room (5).	Educación Física	Desafíos físicos	Pruebas de habilidades motoras y conocimiento físico
Learning through Challenges and Enigmas (8).	Motivación en educación superior	Enigmas y desafíos creativos	Pensamiento crítico y creatividad
Escape Rooms in STEM Teaching and Learning (9).	Educación STEM	Puzzles STEM	Problemas relacionados con ciencia y tecnología
Beyond the Early Adopters: Escape Rooms in Science Education (10).	Ciencias	Pensamiento crítico y colaboración	Actividades de resolución de problemas científicos

Tabla 1 Desarrollo de Contenidos en Escape Rooms Educativos

Por otro lado, en la Tabla 2, se detallan las tecnologías utilizadas en cada trabajo, que incluyen herramientas como H5P, Moodle, Unity, y otras plataformas específicas de gamificación educativa.

Trabajo	Tecnologías Utilizadas
Enseñanza de las matemáticas a través de la gamificación (4).	H5P, Moodle
Revisión literaria científica y propuesta de escape room (5).	Unity
Learning through Challenges and Enigmas (8).	Elementos físicos y digitales no especificados
Escape Rooms in STEM Teaching and Learning (9).	Herramientas educativas STEM generales
Beyond the Early Adopters: Escape Rooms in Science Education (10).	Plataformas de gamificación educativa

Tabla 2 Tecnologías Utilizadas en Escape Rooms Educativos

En cuanto a las tecnologías empleadas en las diferentes propuestas de escape rooms educativos, los trabajos más recientes (8), (9) y (10) aportan enfoques distintos sobre cómo integrar estas herramientas en el aula.

Learning through Challenges and Enigmas: Educational Escape Room as a Predictive Experience of Motivation in University Students - González-Yubero et al. (8): En este trabajo se destaca el uso de plataformas digitales interactivas que permiten tanto la creación como la gestión de escape rooms educativos de forma accesible. Utilizan tecnologías como H5P (Herramienta interactiva de contenido HTML5) dentro de sistemas de gestión del aprendizaje como Moodle para estructurar los desafíos y enigmas en un entorno virtual, permitiendo una interacción fluida con los estudiantes. H5P es clave para la creación de contenido interactivo adaptable a diferentes tipos de dispositivos, facilitando el acceso remoto a los estudiantes. También se menciona el uso de formularios de Google para la recolección de respuestas en algunos de los retos planteados.

Escape Rooms in STEM Teaching and Learning—Prospective Field or Declining Trend? A Literature Review - Lathwesen y Belova (9) :En este artículo de revisión, se analiza la implementación de escape rooms en el ámbito STEM. Si bien no se enfoca en una tecnología específica, los autores mencionan el uso de simulaciones y laboratorios virtuales como una parte fundamental de algunos escapes rooms para la enseñanza de ciencias. Se destacan herramientas de simulación como PhET Interactive Simulations, que permiten a los estudiantes realizar experimentos de física, química y biología en un entorno controlado. Además, mencionan el uso de juegos serios y plataformas de gamificación educativa como Kahoot y Quizizz para añadir una capa de interactividad y evaluación al escape rooms. Estas plataformas proporcionan un entorno lúdico que ayuda a los estudiantes a reflexionar sobre los problemas científicos que se plantean durante la actividad.

Beyond the Early Adopters: Escape Rooms in Science Education - Veldkamp et al. (10): Este estudio publicado en *Frontiers in Education* se centra en cómo los escape rooms pueden ser utilizados para mejorar la enseñanza de las ciencias. Los autores mencionan el uso de tecnologías basadas en realidad aumentada (RA) y realidad virtual (RV) para mejorar la inmersión en los escapes rooms, haciendo que los estudiantes se sientan parte del entorno que exploran. Se utilizan herramientas como CoSpaces Edu y ARKit para crear escenarios interactivos donde los estudiantes pueden manipular objetos tridimensionales o visualizar fenómenos científicos que no serían accesibles de otra manera. Estas tecnologías, aunque no son todavía ampliamente implementadas en todos los contextos educativos, tienen un potencial significativo para transformar la manera en que los estudiantes interactúan con los contenidos.

Los estudios revisados demuestran que los escape rooms educativos son efectivos para aumentar la motivación, promover el aprendizaje activo y fomentar la colaboración en una variedad de disciplinas, permitiendo a los estudiantes aprender de manera más interactiva y práctica, lo que mejora tanto sus habilidades cognitivas como sus habilidades sociales y de trabajo en equipo.

Los escapes rooms ofrecen un entorno lúdico donde los estudiantes pueden aplicar y consolidar sus conocimientos a través de la resolución de problemas y desafíos. A pesar de estos beneficios, no se han desarrollado enfoques específicos que utilicen escape rooms para guiar a los estudiantes en los pasos necesarios para realizar un Trabajo Fin de Grado (TFG), lo que representa una oportunidad única para explorar y desarrollar esta metodología innovadora. Este enfoque podría ayudar a los estudiantes a familiarizarse y comprender mejor los procesos y pasos del TFG mediante métodos de aprendizaje atractivos y efectivos, llenando así un vacío en la práctica educativa y ofreciendo nuevas estrategias para enseñar contenidos fundamentales de manera interactiva y motivadora.

3 Análisis, Diseño e Implementación

En este proyecto se ha desarrollado un juego de desafíos interactivo, comúnmente conocido como un "escape room" digital, utilizando tecnologías web modernas. La aplicación permite que múltiples jugadores se unan a salas de juego, completen una serie de desafíos secuenciales y sean evaluados en una clasificación que refleja el orden en el que completan todos los desafíos. El objetivo principal del proyecto es proporcionar una experiencia de aprendizaje y entretenimiento que sea interactiva y competitiva, a través de una aplicación accesible desde el navegador.

Para alcanzar este objetivo, se han empleado diversas tecnologías, como React, Node.js, Express, Socket.IO y MongoDB. El análisis y diseño del sistema se basan en los principios de la norma **ISO/IEC 25010** (11), la cual define un modelo de calidad para sistemas y software. Esta norma establece características clave como funcionalidad, fiabilidad, usabilidad, eficiencia y mantenibilidad, que fueron vistas en la asignatura de **Ingeniería Informática** y aplicadas en el diseño de esta aplicación para garantizar su efectividad y escalabilidad.

3.1 Análisis

3.1.1 Requisitos Funcionales (RF)

- **RF01.** El sistema debe permitir a los usuarios navegar entre diferentes desafíos a través de un menú de desafíos, proporcionando una interfaz de fácil acceso y uso intuitivo.
- **RF02.** Cada desafío debe comenzar con una pantalla de explicación que ofrezca una descripción clara del desafío, las reglas del juego, y un botón interactivo para iniciar.
- **RF03.** El sistema debe permitir a los usuarios interactuar con contenido H5P, proporcionando actividades como arrastrar y soltar, cuestionarios, juegos interactivos y actividades visuales.
- **RF04.** Los usuarios deben realizar un cuestionario relacionado con los procedimientos del Trabajo de Fin de Grado (TFG). Solo podrán avanzar a la siguiente pregunta si seleccionan la respuesta correcta.
- **RF05.** Al completar el último desafío, el sistema redirigirá al usuario a una página de clasificación o resultados.
- **RF06.** Los desafíos deben ser adaptables para funcionar en modo individual o en modo de grupo, utilizando un `roomCode` para identificar sesiones grupales y sincronizar el progreso de todos los jugadores.
- **RF07.** El sistema debe enviar eventos en tiempo real al servidor utilizando **Socket.IO** para indicar la finalización de los desafíos, cambios en el estado de los jugadores y otras acciones importantes.
- **RF08.** Cada pantalla de desafío debe incluir un enlace visible que redirija a los usuarios a la página informativa del TFG: <https://eupt.unizar.es/TFE>, para proporcionar contexto académico adicional.
- **RF09.** Los desafíos deben manejar situaciones de salida inesperada del juego o actividad, y el sistema debe enviar notificaciones al servidor que registren dicha salida.
- **RF10.** El sistema debe permitir a los usuarios tomar decisiones dentro de un juego interactivo de Ren'Py, y el sistema debe emitir eventos al servidor al finalizar el juego, registrando el progreso del jugador.
- **RF11.** El sistema debe permitir la creación y gestión de salas de juego, permitiendo a los usuarios invitar a otros jugadores a unirse mediante un código de sala único (`roomCode`), y gestionar su estado de preparación.

- **RF12.** La clasificación de los jugadores debe actualizarse en tiempo real para reflejar el orden de finalización y los tiempos de cada desafío, mostrando los resultados en una tabla comparativa.
- **RF13.** El sistema debe permitir la reconexión automática de los jugadores en caso de desconexión, asegurando que su progreso en los desafíos se mantenga intacto.
- **RF14.** Los usuarios deben tener la opción de saltarse un desafío después de 2 minutos de haber comenzado, lo que les permitirá avanzar a la siguiente pantalla.

3.1.2 Requisitos No Funcionales (RNF)

- **RNF01.** La interfaz de usuario debe ser totalmente responsiva, adaptándose de manera óptima a diferentes navegadores y manejando correctamente diferentes resoluciones.
- **RNF02.** El sistema debe utilizar **Socket.IO** para la comunicación en tiempo real entre el cliente y el servidor, asegurando una latencia baja y transmisión instantánea de eventos.
- **RNF03.** La aplicación debe cargar el contenido **H5P** de manera eficiente, utilizando referencias a `frameJs`, `frameCss` y `h5pJsonPath`, y asegurando que las actividades interactivas se carguen sin interrupciones.
- **RNF04.** Las páginas de explicación y las actividades de los desafíos deben estar estilizadas de manera consistente, utilizando un esquema de colores coherente, tipografía uniforme y un diseño visual que ofrezca una experiencia clara y sin distracciones.
- **RNF05.** El sistema debe ser compatible con los navegadores modernos (Chrome, Firefox, Edge y Safari) y garantizar que todas las funciones interactivas sean compatibles en estos navegadores.
- **RNF06.** Las transiciones entre pantallas de desafíos deben ser fluidas, sin interrupciones o recargas de página, para proporcionar una experiencia de usuario continua.
- **RNF07.** Los datos enviados al servidor (como la finalización de desafíos, respuestas de cuestionarios y tiempo de desafío) deben ser precisos y transmitidos en tiempo real sin pérdidas de información.
- **RNF08.** El enlace a la página de TFG debe abrirse en una nueva pestaña del navegador, para asegurar que el usuario no pierda su progreso en la aplicación y pueda continuar jugando.
- **RNF09.** La aplicación debe ser segura en cuanto a la gestión de sesiones y autenticación, asegurando que las conexiones solo se realicen desde fuentes autorizadas y verificadas.
- **RNF10.** Los errores de comunicación con el servidor deben ser gestionados correctamente, todos los errores deben ser registrados para facilitar la resolución de problemas.
- **RNF11.** El rendimiento de la aplicación debe ser óptimo en cuanto a tiempos de respuesta y carga, incluso bajo condiciones de múltiples jugadores y desafíos simultáneos.

3.1.3 Tecnologías Utilizadas

En el siguiente apartado se explicará detalladamente cada herramienta que se ha utilizado en el desarrollo de la aplicación web. La idea de partida era desarrollar una aplicación eficiente y robusta utilizando tecnologías modernas para ofrecer una experiencia de usuario óptima.

React (11): Es una biblioteca de JavaScript desarrollada por Facebook para la creación de interfaces de usuario, especialmente en aplicaciones de una sola página (SPA). Su arquitectura basada en componentes permite una fácil reutilización de código y facilita la organización del proyecto. React utiliza un Virtual DOM para mejorar la eficiencia en la actualización de la interfaz, haciendo que solo los componentes que han cambiado se vuelvan a renderizar. Esto contribuye a una experiencia de usuario más rápida y fluida.



Figura 1 Logo de React

Node.js (12): Es un entorno de ejecución para JavaScript en el servidor, basado en el motor V8 de Google Chrome. Node.js permite la construcción de aplicaciones de red escalables gracias a su modelo de entrada/salida no bloqueante y su capacidad para manejar múltiples conexiones simultáneamente. En este proyecto, Node.js se ha utilizado junto con el framework Express para desarrollar la lógica del servidor y crear APIs RESTful que interactúan con la base de datos.



Figura 2 Logo de Node.js

Express (13): Es un framework minimalista para Node.js que facilita la creación de servidores web y APIs. Express se encarga de manejar las solicitudes HTTP y de servir como puente entre la aplicación web y la base de datos, permitiendo así una comunicación fluida entre el cliente y el servidor.



Figura 3 Logo Express

Socket.IO (14): Es una biblioteca que permite la comunicación en tiempo real entre el servidor y los clientes. Ha sido fundamental para implementar funciones como la creación y gestión de salas de juego, la sincronización de estados entre jugadores y la actualización en tiempo real de la información compartida entre todos los usuarios conectados. Socket.IO soporta la comunicación bidireccional basada en eventos, lo que garantiza una experiencia de usuario interactiva y sin retrasos.



Figura 4 Logo Socket.IO

MongoDB (15): Es una base de datos NoSQL que almacena los datos en formato JSON, lo cual proporciona gran flexibilidad y escalabilidad. MongoDB se ha elegido como la base de datos principal del proyecto debido a su capacidad para manejar grandes volúmenes de datos no estructurados y su facilidad de integración con Node.js. MongoDB permite el almacenamiento eficiente de datos relacionados con las salas de juego y los jugadores, y facilita la recuperación rápida de información necesaria para la interacción en tiempo real.



Figura 5 Logo mongoDB

H5P (16): Es una plataforma que permite la creación, compartición y reutilización de contenido interactivo en HTML5. En este proyecto, H5P se ha utilizado para crear actividades interactivas y educativas que se pueden integrar fácilmente en la aplicación web, mejorando la experiencia del usuario mediante la inclusión de elementos multimedia y educativos.



Figura 6 Logo H5P

Ren'Py (17): Es un motor de creación de novelas visuales basado en Python, conocido por su facilidad de uso y flexibilidad. Ren'Py se ha integrado en el proyecto para desarrollar contenido narrativo interactivo, lo cual permite añadir elementos de gamificación y storytelling a la aplicación, enriqueciendo así la experiencia de usuario.



Figura 7 Logo Ren`Py

GitHub (18): Es una plataforma de control de versiones basada en Git que facilita la colaboración en proyectos de desarrollo. En este proyecto, GitHub se ha utilizado para alojar el código fuente y gestionar versiones.



Figura 8 Logo GitHub

3.2 Diseño

Esta sección cubrirá todos los aspectos relacionados con la conceptualización, diseño de desafíos y la estructura de la base de datos. Se enfoca en las decisiones iniciales de diseño y cómo se planificará la interacción de los usuarios con la aplicación.

3.2.1 Diagrama UML del sistema

Para ilustrar la arquitectura de la aplicación y las interacciones entre los diferentes componentes del sistema, se presenta a continuación el diagrama UML. Este diagrama proporciona una visión general de las clases, objetos y relaciones que conforman el proyecto "Escape Room", facilitando la comprensión de cómo se estructuran y comunican los módulos de la aplicación.

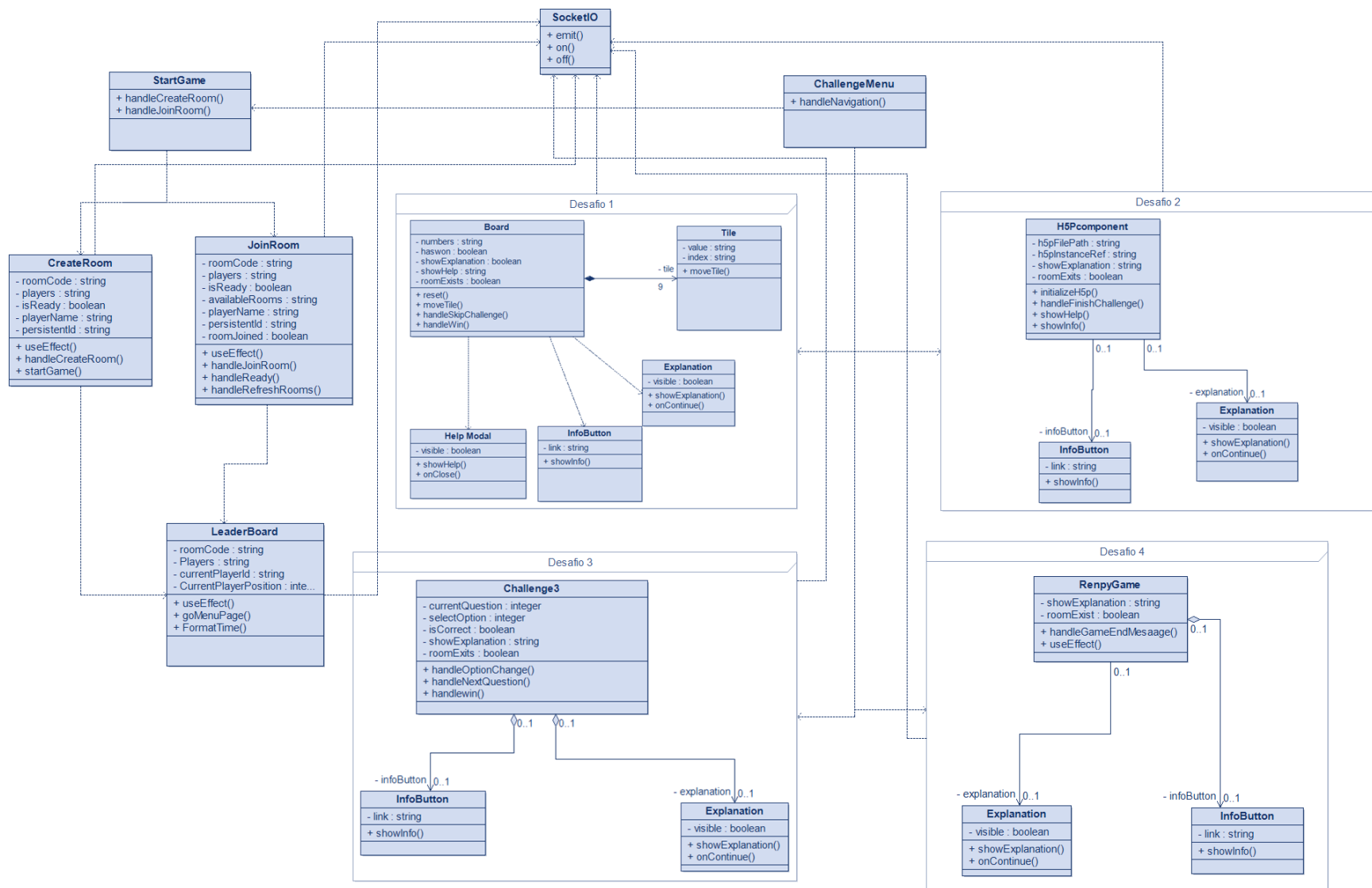


Figura 9 Diagrama de clase

3.2.2 Concepto general de los desafíos

Cada desafío en la aplicación "Escape Room" se diseñará para proporcionar una experiencia interactiva y educativa, aprovechando diversas tecnologías y enfoques de juego. A continuación, se describen los desafíos planeados.

3.2.2.1 Desafío 1 – Rompecabezas Numérico (con Imágenes)

Descripción: Este desafío presentará un tablero de rompecabezas deslizante en el que los jugadores deberán organizar imágenes en un orden específico, dejando una casilla vacía para facilitar el movimiento de las piezas. El objetivo será simular los pasos necesarios para completar un Trabajo de Fin de Grado (TFG) de manera correcta y secuencial.

Interacción del Usuario: Los jugadores interactuarán con el tablero moviendo las piezas adyacentes a la casilla vacía hasta que todas las imágenes estén en el orden correcto.

Elementos Adicionales de Interfaz:

- **Explicación del Desafío:** Al iniciar el desafío, se mostrará una explicación breve que orientará a los jugadores sobre el objetivo y las reglas del juego.
- **Botón de Ayuda:** Se proporcionará un botón de ayuda para guiar a los jugadores con pistas sobre cómo resolver el rompecabezas.
- **Botón de Información TFG:** Se incluirá un botón que redirigirá a una página web con información adicional sobre el TFG, proporcionando contexto académico.

Verificación de Victoria: El sistema verificará automáticamente si las imágenes están en el orden correcto. Si el jugador completa el desafío, se activará una acción para proceder al siguiente desafío o finalizar el juego.

Botón de Saltar Desafío: Se planificará un botón que se habilitará después de dos minutos de haber iniciado el desafío, permitiendo a los jugadores saltar al siguiente desafío si encuentran dificultades.

3.2.2.2 Desafío 2 – Drag and Drop (Arrastrar y soltar)

Descripción: En este desafío, los jugadores deberán arrastrar y soltar elementos en sus posiciones correctas, simulando la organización de componentes esenciales de una propuesta de TFG.

Interacción del Usuario: Los jugadores usarán una interfaz de arrastrar y soltar para mover diferentes elementos a sus lugares correctos, asegurando que los componentes del TFG estén adecuadamente organizados.

Elementos Adicionales de Interfaz:

- **Explicación del Desafío:** Se presentará una breve descripción inicial que guiará a los jugadores sobre cómo interactuar con la actividad.
- **Botón de Información TFG:** Se incluirá un botón que proporcionará acceso a una página con detalles sobre la normativa y estructura de los TFG, representado por un ícono de documento.

Verificación de Progreso: El sistema monitoreará el progreso del jugador. Una vez que todos los elementos estén correctamente posicionados, se registrará la finalización del desafío y el jugador podrá avanzar al siguiente.

Estado y Persistencia: El desafío verificará si la sala de juego es válida al cargar. Si la sala no es válida, el jugador será redirigido al menú de desafíos.

Botón de Saltar Desafío: Se planificará un botón que se habilitará después de dos minutos de haber iniciado el desafío, permitiendo a los jugadores saltar al siguiente desafío si encuentran dificultades.

3.2.2.3 Desafío 3 -Preguntas y Respuestas

Descripción: Este desafío consistirá en un cuestionario de opción múltiple diseñado para evaluar los conocimientos de los jugadores sobre los procedimientos del TFG.

Interacción del Usuario: Los jugadores seleccionarán una opción de respuesta para cada pregunta presentada. Solo podrán avanzar a la siguiente pregunta si seleccionan la respuesta correcta.

Elementos Adicionales de Interfaz:

- **Explicación Inicial del Desafío:** Se proporcionará una guía sobre el objetivo del desafío y cómo interactuar con las preguntas.
- **Botón de Información TFG:** Se facilitará el acceso a información adicional sobre los TFG para ayudar a los jugadores a entender mejor el contexto de las preguntas.

Verificación de Respuestas: El sistema verificará automáticamente si la opción seleccionada es correcta. Solo las respuestas correctas permitirán avanzar en el cuestionario.

Estado y Persistencia: Al igual que en otros desafíos, se verificará la validez de la sala al inicio para mantener la coherencia del juego.

Botón de Saltar Desafío: Se planificará un botón que se habilitará después de dos minutos de haber iniciado el desafío, permitiendo a los jugadores saltar al siguiente desafío si encuentran dificultades.

3.2.2.4 Desafío 4 – Juego con Ren`py

Descripción:

Este desafío sumergirá a los jugadores en una narrativa interactiva desarrollada con Ren'Py, simulando una defensa de TFG. Los jugadores tendrán que tomar decisiones para avanzar en la historia y completar el desafío.

Interacción del Usuario:

A través de una interfaz visual de novela, los jugadores interactuarán con personajes y tomarán decisiones clave que afectarán el desarrollo de la historia.

Elementos Adicionales de Interfaz:

- **Explicación del Desafío:** Se presentará una descripción antes de iniciar, que orientará a los jugadores sobre el contexto y objetivos del desafío.
- **Botón de Información TFG:** Dirigirá a los jugadores a una página web con más información sobre los TFG.

Verificación de Respuestas: El sistema verificará automáticamente si la opción seleccionada es correcta. Solo las respuestas correctas permitirán avanzar en el cuestionario.

Estado y Persistencia: Al igual que en otros desafíos, se verificará la validez de la sala al inicio para mantener la coherencia del juego.

Botón de Saltar Desafío: Se planificará un botón que se habilitará después de dos minutos de haber iniciado el desafío, permitiendo a los jugadores saltar al siguiente desafío si encuentran dificultades.

3.2.2.5 Clasificación

Descripción:

La sección de clasificación mostrará el rendimiento de los jugadores al completar los desafíos en el juego "Escape Room". La clasificación se basa en el tiempo total que cada jugador tarda en completar los desafíos y se actualiza en tiempo real para proporcionar una comparación clara entre los participantes.

Interacción del Usuario:

Los jugadores podrán ver su posición en la clasificación general, así como los tiempos que han tomado para completar cada uno de los desafíos. El sistema organizará la lista de jugadores por tiempo total de forma automática.

Elementos Adicionales de Interfaz:

- **Número del Jugador Actual:** Se mostrará un indicador visual que informa al jugador su posición actual en la clasificación.
- **Lista de Clasificación:** La lista de jugadores se presentará en orden de menor a mayor tiempo total, con sus nombres y tiempos totales.
- **Tabla de Tiempos por Desafío:** Para cada jugador, se proporcionará una tabla que muestra el tiempo tomado para completar cada uno de los desafíos, permitiendo a los jugadores analizar su rendimiento individual por desafío.
- **Botón de Retorno:** Un botón llevará al usuario de vuelta al menú principal.

Verificación de Progreso:

El sistema se conectará al servidor para recibir y actualizar en tiempo real los datos de clasificación de todos los jugadores.

Estado y Persistencia:

Cada vez que un jugador acceda a la página de clasificación, el sistema verificará la existencia de la sala de juego. Si la sala ha sido eliminada o es inválida, el jugador será redirigido al menú principal. Asimismo, la información sobre la clasificación se almacenará temporalmente en el servidor y se actualizará de manera dinámica a medida que los jugadores completen los desafíos.

3.2.2.6 Menú de selección de Desafíos

Descripción:

El menú de selección de desafíos será el punto de entrada para que los usuarios elijan los desafíos que desean realizar en la aplicación "Escape Room". Este menú estará diseñado para ser intuitivo, proporcionando a los jugadores una navegación clara entre los diferentes desafíos disponibles.

Interacción del Usuario:

- **Botones de Selección de Desafíos:** El menú contará con botones interactivos que permitirán al usuario seleccionar cualquiera de los cuatro desafíos disponibles. Al hacer clic en un botón, el usuario será redirigido al desafío correspondiente.
- **Botón de Menú Principal:** Se incluirá un botón para retornar fácilmente al menú principal de la aplicación, facilitando la navegación fluida entre las diferentes secciones del juego.
- **Enlace a Información sobre el TFG:** Se añadirá un enlace a la normativa del Trabajo Fin de Grado (TFG), que abrirá una página externa en una nueva pestaña del navegador. Esto permitirá a los jugadores consultar información adicional sin perder el progreso en el juego.

Elementos de Diseño:

- **Estructura Visual:** El diseño del menú será simple y accesible, con botones grandes y claramente etiquetados para cada desafío. Los botones estarán alineados en una cuadrícula o en un formato de lista vertical, dependiendo del espacio disponible en la interfaz de usuario.
- **Diseño Adaptativo:** El menú se diseñará para ser responsive, asegurando una visualización óptima tanto en dispositivos de escritorio como en dispositivos móviles.
- **Estilo y Temática:** Se utilizará una combinación de colores y fuentes que mantendrán la coherencia con el tema general del "Escape Room". Los botones y los enlaces tendrán un estilo visual que los destaque sin distraer al usuario.

Comportamiento del Sistema:

- **Navegación entre Desafíos:** Al seleccionar un desafío, el sistema comprobará si el jugador está autorizado para acceder al mismo, en función de su progreso actual. Si es válido, el jugador será redirigido al desafío. En caso contrario, se mostrará un mensaje indicando que el desafío no está disponible.
- **Persistencia de Progreso:** El sistema se encargará de mantener el estado del jugador, de manera que puedan retomar los desafíos en cualquier momento sin perder su progreso.
- **Acceso a Información del TFG:** El enlace a la información del TFG abrirá una nueva ventana del navegador para asegurar que la experiencia del juego no sea interrumpida.

Verificación de Navegación:

El sistema deberá verificar la validez de cada ruta de navegación antes de permitir el acceso al desafío correspondiente. Se diseñará un mecanismo que valide el acceso del jugador en función de su progreso y estado en el juego.

3.2.2.7 Gestión de salas

En la aplicación, la gestión de salas es un componente fundamental que permite a los jugadores interactuar entre sí en un entorno controlado. Esta funcionalidad facilita la creación de una experiencia de juego en grupo, la administración eficiente de los jugadores, la sincronización de sus acciones en tiempo real y la personalización mediante el ingreso de nombres de los jugadores. A continuación, se detalla el concepto y diseño de la funcionalidad de gestión de salas.

3.2.2.7.1 Creación de salas

Descripción:

Los jugadores podrán crear una nueva sala de juego con un solo clic. Cada sala tendrá un código único que identificará la sesión y permitirá que otros jugadores se unan. Además, el creador de la sala deberá **ingresar su nombre** antes de poder marcarse como "Listo" para empezar el juego.

Interacción del Usuario:

- **Generación de Código de Sala:** Al elegir crear una sala, el sistema generará y proporcionará un código de sala único al jugador creador, que este podrá compartir con otros jugadores interesados en unirse a la misma sesión de juego.
- **Ingreso del Nombre del Creador:** El jugador que crea la sala deberá introducir su nombre en un campo de entrada, que será visible para los demás jugadores en la lista de participantes.
- **Estado de Preparación:** El creador de la sala podrá indicar su estado de preparación marcándose como "Listo", pero no podrá iniciar el juego hasta que todos los jugadores también estén listos.

Comportamiento del Sistema:

- **Generación de Código:** El servidor generará un código de sala único y asignará al jugador creador un rol especial que le permitirá iniciar el juego.
- **Persistencia del Nombre:** El nombre del creador será almacenado temporalmente en el servidor para que otros jugadores puedan ver quién creó la sala.
- **Estado de la Sala:** La sala estará en un estado inicial que permitirá la entrada de nuevos jugadores hasta que se complete el número máximo permitido o el creador decida comenzar el juego.

3.2.2.7.2 Unión a salas

Descripción:

Los jugadores podrán unirse a una sala existente utilizando un código de sala proporcionado por el creador de la misma. Además, deberán **introducir su nombre** antes de poder marcarse como "Listos" para empezar el desafío.

Interacción del Usuario:

- **Introducción del Código de Sala:** Los jugadores introducirán el código de sala en un campo de entrada en la interfaz de usuario y seleccionarán la opción "Unirse a la Sala" para ingresar.
- **Ingreso del Nombre del Jugador:** Después de unirse a la sala, el jugador deberá introducir su nombre para ser identificado por los demás participantes. Este nombre se mostrará en la lista de jugadores junto a su estado de preparación.
- **Estado de Preparación:** Los jugadores deberán marcarse como "Listos" para que el creador pueda comenzar el juego. El sistema solo permitirá el inicio del juego cuando todos los jugadores en la sala estén listos.

Comportamiento del Sistema:

- **Validación del Código de Sala:** El servidor validará el código de sala y comprobará si la sala está abierta y no ha alcanzado el límite de jugadores. Si la validación es exitosa, el jugador se unirá a la sala y se le asignará un lugar en la lista de participantes.
- **Validación del Nombre:** El sistema verificará que el jugador haya ingresado un nombre antes de que pueda marcarse como "Listo". El nombre será persistente durante toda la sesión.
- **Actualización de la Interfaz:** El sistema notificará a los jugadores existentes sobre la llegada del nuevo jugador y actualizará la interfaz de usuario en tiempo real para reflejar el estado de los jugadores.

3.2.3 Diseño de la base de datos (DBD)

3.2.3.1 Diseño Conceptual

El diseño conceptual de la base de datos se representa mediante un Diagrama Entidad-Relación. Para el desarrollo de este proyecto, se utilizó MongoDB como sistema de base de datos, debido a su capacidad de manejar datos de manera flexible y eficiente en un formato de documento. A través de Mongoose, un ODM (Object Data Modeling) para MongoDB, se definieron los esquemas necesarios para almacenar la información clave del sistema, respetando el modelo no relacional propio de esta base de datos.

Esquema ChallengeTime

El esquema ChallengeTime permite registrar los tiempos que cada jugador tarda en completar diferentes desafíos. Este esquema incluye los siguientes campos:

- challengeName: El nombre del desafío, que es obligatorio para identificar el tipo de desafío.
- timeTaken: Tiempo que el jugador ha tardado en completar el desafío, inicializado en 0.
- startTime: Fecha y hora de inicio del desafío, con valor por defecto de la hora actual (Date.now).
- completedAt: Fecha y hora en la que el desafío fue completado, quedando en blanco hasta que el jugador termine el desafío.

Este esquema se utiliza como subdocumento en otros esquemas, permitiendo llevar un registro detallado de los tiempos de cada jugador en el contexto de su sala.

Esquema Player

El esquema Player representa a cada usuario dentro de una sala e incluye los siguientes atributos:

- id: Identificador único del jugador.
- persistentId: Identificador permanente que facilita el seguimiento de un jugador a lo largo del tiempo.
- name: Nombre del jugador.
- isReady: Estado del jugador (preparado o no), inicializado en false.
- totalTime: Tiempo total acumulado por el jugador en los desafíos de una sala.
- challengeTimes: Un array de subdocumentos ChallengeTime que almacena el tiempo tomado por el jugador en cada desafío.

Este diseño, en el que Player contiene un array de tiempos de desafíos (challengeTimes), facilita el acceso y almacenamiento directo de información relevante al rendimiento del jugador en cada sala.

Esquema Room

El esquema Room representa una instancia de juego, o sala, donde los jugadores interactúan y completan desafíos. Incluye los siguientes campos:

- token: Identificador único de la sala, que actúa como clave principal.
- players: Un array de subdocumentos Player, que contiene la información de cada jugador en la sala.
- maxPlayers: Número máximo de jugadores permitido en la sala, con un valor por defecto de 4.
- state: Estado de la sala (por ejemplo, “abierta” o “cerrada”).

Esquema Leaderboard

El esquema Leaderboard registra los tiempos y resultados de cada sala de manera consolidada.

Incluye:

- roomCode: Código de la sala al que está asociado el leaderboard.
- players: Un array de jugadores con sus tiempos totales y detalles de cada desafío.
- createdAt: Fecha de creación del leaderboard, con valor predeterminado Date.now.

El uso de subdocumentos en estos esquemas permite un modelo de datos altamente anidado y eficiente, alineado con la naturaleza no relacional de MongoDB, en la que la información relevante se almacena directamente dentro de los documentos, sin necesidad de hacer referencia a otras colecciones.

Para complementar la descripción de la base de datos y proporcionar una representación visual clara de los esquemas diseñados, se ha incluido una versión en formato XML. Este formato jerárquico refleja la estructura de los datos tal como se almacenan en MongoDB, permitiendo identificar las relaciones y anidamientos entre entidades como Room, Player, y ChallengeTime.

Dicha representación se encuentra detallada en el **Anexo F: Representación XML del Esquema de Datos**, el cual facilita la comprensión de los elementos y su organización dentro del sistema. Este anexo es particularmente útil para visualizar cómo se integran los subdocumentos y colecciones en MongoDB.

3.3.3 Diseño físico de la Base de datos

La aplicación almacenará los datos en la base de datos en la nube utilizando MongoDB Atlas. Esta elección se basa en la flexibilidad y escalabilidad que ofrece MongoDB, lo cual es esencial para manejar datos estructurados y no estructurados de manera eficiente. MongoDB es una base de datos NoSQL que permite la fácil gestión de datos de documentos en formato JSON, lo que se alinea bien con las necesidades dinámicas de la aplicación.

3.3.3.1 Integración del SGBD

A continuación, se detallará cómo se ha realizado la conexión de la base de datos MongoDB con la aplicación usando Node.js y Mongoose, junto con la configuración de Socket.IO para la comunicación en tiempo real.

1. Configuración Inicial de MongoDB Atlas:

- Acceder a la web de MongoDB Atlas y crear una cuenta si no se tiene una.
- Una vez dentro de MongoDB Atlas, se debe crear un nuevo cluster. Este es el entorno donde se alojarán las bases de datos.
- Después de crear el cluster, se debe configurar la base de datos, permitiendo las IPs necesarias y creando un usuario con los permisos adecuados para interactuar con la base de datos.

2. Conectar MongoDB Atlas con la Aplicación:

- Obtener el URI de conexión proporcionado por MongoDB Atlas. Este URI incluye la dirección del cluster y las credenciales de usuario.
- En el código de la aplicación, se debe usar la biblioteca mongoose para manejar la conexión

3. Configuración de Modelos con Mongoose:

- Definir los modelos de datos usando esquemas de Mongoose.

4. Uso de Socket.IO para la Comunicación en Tiempo Real

- La aplicación utiliza Socket.IO para permitir la comunicación en tiempo real entre el servidor y los clientes.

3.4 Identidad Corporativa

Para dar un toque de profesionalidad al proyecto, se ha diseñado un nombre y un logo que son representativos de la aplicación. El nombre elegido es "Escape Room", que permite a los usuarios identificar de inmediato la naturaleza del software: una aplicación interactiva centrada en desafíos y resolución de acertijos en un entorno de escape.

Ver **Anexo E**.

4 Planificación

La planificación fue un componente fundamental en la ejecución del "Escape Room Digital sobre Normativa de TFE". Con el objetivo de mantener el control del tiempo y los recursos, se utilizó la herramienta GanttProject (20) para crear un diagrama de Gantt que permitió estructurar y visualizar el desarrollo del proyecto en distintas fases. Este tipo de diagrama fue elegido debido a su utilidad para la gestión de proyectos, ya que facilita la representación gráfica de tareas, la identificación de dependencias y la monitorización del progreso. Además, su elección estuvo fundamentada en los conocimientos adquiridos en la asignatura de Proyecto Software, donde se estudiaron metodologías y herramientas clave para la gestión de proyectos.

El proyecto se dividió en cuatro fases principales: Análisis y diseño, Desarrollo, Despliegue y pruebas, y Realización de la memoria. A cada una de estas fases se le asignó un tiempo estimado de inicio y fin, con la posibilidad de realizar ajustes conforme a la evolución del trabajo. En algunos casos, surgieron discrepancias entre los tiempos planificados y los tiempos reales, principalmente debido a la complejidad técnica de las tecnologías involucradas y a las revisiones detalladas que fueron necesarias. No obstante, el diagrama de Gantt permitió identificar rápidamente los desvíos y reajustar las prioridades para cumplir con los objetivos establecidos.

Fases del proyecto:

1. **Análisis y diseño (5 de julio - 9 de julio):** Durante esta fase, se definieron las tecnologías que se utilizarían en el proyecto, el número de desafíos a incluir en el "Escape Room" y cómo se abordaría la estructura general del TFG. Se completó en los 5 días inicialmente planificados.
2. **Desarrollo (10 de julio - 6 de septiembre):** La fase de desarrollo abarcó la implementación del frontend utilizando **React**, la creación del backend con **Node.js**, y la integración de **MongoDB**, **H5P**, **Socket.IO** y **Redmi** para crear un entorno interactivo. Aunque el tiempo estimado para esta fase era de 53 días, se extendió a 59 días debido a complicaciones adicionales durante la integración de las tecnologías.
3. **Despliegue y pruebas (7 de septiembre - 20 de septiembre):** Esta fase se centró en el despliegue del sistema y la gestión del soporte técnico. Inicialmente estaba planificada para 7 días, pero la implementación tomó 14 días debido a ajustes técnicos y correcciones menores durante el despliegue.
4. **Realización de la memoria (10 de julio - 18 de octubre):** La realización de la memoria abarcó tanto la redacción como la corrección del documento final. Esta fase estaba planificada desde el **10 de julio de 2024** con una finalización estimada para el **11 de septiembre de 2024**. Sin embargo, debido a las revisiones y mejoras necesarias, la finalización real ocurrió el **18 de octubre de 2024**, acumulando un total de **101 días**.

A lo largo de todo el proyecto, el uso del diagrama de Gantt resultó fundamental para gestionar los tiempos y las dependencias entre las tareas. El diagrama permitió ajustar los plazos cuando fue necesario, facilitando la gestión del desarrollo y la documentación del proyecto.

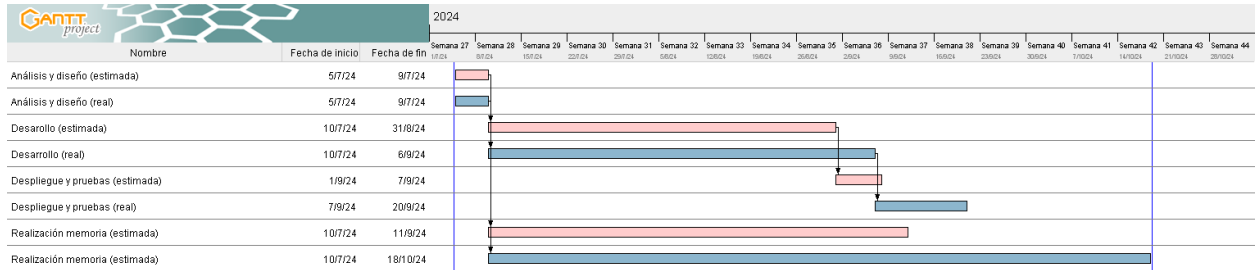


Figura 10 Diagrama de Gantt

5 Implementación

Tras haber desarrollado los prototipos de análisis y diseño, es momento de comenzar con el desarrollo de la aplicación "Escape Room". Esta etapa incluye la configuración de la base de datos y la implementación de las vistas de las pantallas con las que los usuarios interactuarán. La aplicación sigue un enfoque dinámico para la gestión de datos, creando y actualizando la información en tiempo real según las interacciones de los usuarios.

5.1 Inicialización y configuración de la Base de datos

La aplicación "Escape Room" utiliza un enfoque dinámico para gestionar la información de jugadores y salas de juego. Este enfoque permite que los datos se creen y modifiquen en tiempo real a medida que los usuarios interactúan con la aplicación. La base de datos se gestiona utilizando MongoDB Atlas en combinación con Mongoose, un ODM (Object Data Modeling) para Node.js que facilita la interacción con MongoDB.

```
_id: ObjectId('66d52499ae653085a03cf89b')
token: "wuo7n"
▼ players: Array (2)
  ▼ 0: Object
    id: "2pmOX0t8oRc_-0xFAABp"
    persistentId: "player-yqp8iyh98"
    name: "Jugador 1"
    isReady: true
    _id: ObjectId('66d52499ae653085a03cf89c')
  ▼ 1: Object
    id: "rDAR_zY1D4qXKVoYAAB4"
    persistentId: "player-09xs8o8jf"
    name: "Jugador 2"
    isReady: true
    _id: ObjectId('66d524a1ae653085a03cf8b2')
maxPlayers: 4
state: "InProgress"
▼ arrivalOrder: Array (2)
  0: "Jugador 2"
  1: "Jugador 1"
__v: 3
```

Figura 11 Ejemplo de Gestión de Datos en MongoDB Atlas para "Escape Room"

Esquema de Datos: La estructura de los datos se define mediante esquemas de Mongoose. Los esquemas roomSchema y playerSchema establecen cómo se organizan los datos en la base de datos. Por ejemplo, roomSchema define atributos como token, players, maxPlayers, state, y arrivalOrder, mientras que playerSchema incluye id, persistentId, name, e isReady.

5.2 Desarrollo de las pantallas

En esta sección se describen las diferentes pantallas de la aplicación "Escape Room Digital". Cada una ha sido diseñada para ser intuitiva y fácil de usar, siguiendo los principios de usabilidad y accesibilidad establecidos. A continuación, se detallan las características y funcionalidades de cada pantalla, comenzando con la Pantalla Principal. Las capturas de pantalla correspondientes pueden encontrarse en el **Anexo A**.

5.2.1 Pantalla Home

Descripción General: La Pantalla Home es la primera vista que los usuarios encuentran al iniciar la aplicación *Escape Room Digital*. Su propósito es ofrecer un punto de entrada sencillo e intuitivo desde el cual los usuarios puedan comenzar su experiencia de juego de forma rápida.

Interacciones del Usuario:

- **Iniciar Juego:** La pantalla muestra un mensaje de bienvenida junto con un botón prominente que invita al usuario a iniciar el juego. Al hacer clic en "Iniciar Juego", el usuario es redirigido automáticamente a la pantalla de selección de opciones, donde podrá elegir entre **Crear Sala** o **Unirse a una Sala**.

Ver captura en **Anexo A.1**.

Manejo de Navegación:

- **Navegación Dinámica:** La navegación se gestiona mediante enrutamiento dinámico utilizando React Router. Esto permite que la página no se recargue completamente al cambiar de una pantalla a otra, asegurando una experiencia fluida. Al seleccionar "Iniciar Juego", se desencadena un evento de redirección a la siguiente vista sin necesidad de recargar el contenido, manteniendo la aplicación reactiva y rápida.

Diseño de la Interfaz:

- **Estética Minimalista:** El diseño de la interfaz está orientado hacia una experiencia de usuario directa y sin distracciones. Los elementos visuales son mínimos pero efectivos, utilizando un **título destacado** que da la bienvenida al jugador y un **botón prominente** para guiar la acción principal. Esta estructura asegura que los usuarios comprendan inmediatamente qué deben hacer al iniciar la aplicación.

Conexión con Otras Pantallas:

- **Punto de Transición:** Desde un punto de vista arquitectónico, la Pantalla Home actúa como una página de transición hacia otras funcionalidades más interactivas de la aplicación. Aunque en esta pantalla no se gestionan salas ni desafíos directamente, la Home ofrece una navegación clara hacia el núcleo del juego: la creación de salas y la participación en desafíos. Al hacer clic en "Iniciar Juego", el usuario es llevado a la pantalla de selección de opciones.

5.2.2 Pantalla "Elige una Opción"

Descripción General: La pantalla *Elige una Opción* es un punto de decisión fundamental dentro de la aplicación *Escape Room Digital*, donde el usuario puede optar por **crear una sala de juego** o **unirse a una sala ya existente**. Esta funcionalidad permite al usuario iniciar la experiencia de juego de manera personalizada, dependiendo de si desea organizar una nueva sesión o sumarse a una partida existente.

Interacciones del Usuario:

- **Crear Sala:** Al hacer clic en el botón "Crear Sala", el usuario es redirigido a la pantalla donde puede generar una nueva sala y obtener un código que podrá compartir con otros jugadores.
- **Unirse a Sala:** Si el usuario selecciona la opción "Unirse a Sala", es llevado a una pantalla donde puede ingresar el código de una sala previamente creada por otro jugador. Esto le permitirá unirse a esa sesión de juego.

Ver captura en **Anexo A.2**.

Manejo de Navegación:

- **Navegación Dinámica:** Similar a otras pantallas, la navegación se gestiona mediante enrutamiento dinámico sin recargar la página completa. Cada opción de la pantalla redirige a rutas específicas predefinidas utilizando React Router. Cuando el usuario selecciona "Crear Sala", es redirigido a `/create`, mientras que "Unirse a Sala" lo lleva a `/join`. Esta navegación eficiente asegura una experiencia fluida y sin interrupciones.

Diseño de la Interfaz:

- **Interfaz Clara y Práctica:** El diseño de esta pantalla está orientado a la simplicidad y la claridad. Se presentan dos botones prominentes que permiten a los usuarios elegir entre **crear** o **unirse** a una sala. Estos botones están claramente diferenciados y estilizados para garantizar que el usuario pueda identificar rápidamente ambas opciones, promoviendo una experiencia de usuario directa y sin ambigüedades.

5.2.3 Pantalla "Crear Sala" - Implementación Técnica

Descripción General: La *Pantalla Crear Sala* es una de las funcionalidades centrales de la aplicación *Escape Room Digital*, ya que permite a los usuarios iniciar una nueva sesión de juego con varios jugadores. A través de la creación de una sala, el sistema genera un código único que se utiliza para que otros jugadores puedan unirse a la sesión.

Interacciones del Usuario:

1. Creación de Sala:

- Cuando el usuario selecciona la opción "Crear Sala", la aplicación envía una solicitud al servidor a través de **Socket.IO**. El servidor responde generando un código de sala único y aleatorio que se muestra en la pantalla. Este código puede ser compartido con otros jugadores para que se unan a la sala.

2. Manejo de Jugadores:

- Una vez creada la sala, los jugadores que se unan se listan en la pantalla, donde se muestra su **estado de preparación**. Cada jugador debe ingresar su nombre antes de poder marcarse como "Listo". El usuario que creó la sala también se une como el primer jugador automáticamente.

3. Estado de Preparación:

- El juego no puede comenzar hasta que todos los jugadores se marquen como "Listos". Para ello, los jugadores cuentan con un botón "Estoy Listo", el cual actualiza su estado en tiempo real.

Ver captura en **Anexo A.3**.

Manejo de la Conexión en Tiempo Real:

- La comunicación y sincronización entre los jugadores y el servidor se realiza mediante **Socket.IO**, lo que permite la actualización en tiempo real de los eventos clave de la sala. Los eventos que gestionan esta funcionalidad son:
 - **createRoom**: Al crear la sala, el servidor genera un código único y asigna el primer jugador.
 - **playerJoined**: Se activa cuando un nuevo jugador se une a la sala, actualizando la lista de jugadores en la pantalla para todos los usuarios conectados.
 - **playerReadyStatus**: Este evento se dispara cuando un jugador se marca como "Listo", actualizando el estado en la interfaz de todos los jugadores de la sala.
 - **startGame**: Una vez que todos los jugadores están listos, el creador de la sala puede iniciar el juego. Esto activa el evento de inicio, redirigiendo automáticamente a todos los jugadores al primer desafío.

Diseño de la Interfaz:

- La pantalla **Crear Sala** está diseñada bajo un enfoque minimalista para facilitar la usabilidad. Los elementos clave, como el código de sala, la lista de jugadores y el botón "**Estoy Listo**", están claramente visibles y destacados, lo que permite que los jugadores identifiquen rápidamente el estado de la sala y las interacciones necesarias.

Manejo de la Persistencia de Datos:

- Se utiliza un **ID persistente** para identificar a cada jugador. Este ID se genera la primera vez que el jugador crea o se une a una sala y se almacena en el almacenamiento local del navegador. Este ID permite que los jugadores sean identificados de forma consistente, incluso si se desconectan o vuelven a conectarse durante la sesión de juego.

Conexión con Otras Pantallas:

- Una vez que todos los jugadores están listos, el creador de la sala puede iniciar el juego. Al hacerlo, el sistema envía un evento al servidor que notifica a todos los jugadores conectados, redirigiéndolos automáticamente al **primer desafío**. La transición entre la pantalla de creación de sala y el inicio del juego es fluida y sincronizada mediante los eventos gestionados por **Socket.IO**.

5.2.4 Pantalla "Unirse a Sala" - Implementación Técnica

Descripción General: La pantalla *Unirse a Sala* permite que los jugadores se conecten a una sala de juego ya existente dentro de la aplicación *Escape Room Digital*. Esta funcionalidad es clave para permitir la colaboración en tiempo real entre varios jugadores. A través de esta pantalla, los jugadores pueden ingresar un código de sala proporcionado por el creador de la sala o seleccionar una sala abierta de una lista mostrada.

Interacciones del Usuario:

1. Unirse a una Sala Existente:

- Los jugadores pueden unirse a una sala existente de dos formas:
 - Ingresando un **código de sala** que el creador les haya compartido.
 - Seleccionando una **sala disponible** de una lista mostrada, que contiene salas abiertas con espacio disponible para más jugadores.
- Una vez que el jugador se une a la sala, es añadido a la **lista de jugadores** y puede ver el nombre de los otros participantes, así como su **estado de preparación** (Listo/No Listo).

2. Lista de Salas Disponibles:

- Al acceder a la pantalla *Unirse a Sala*, la aplicación se conecta al servidor mediante **Socket.IO** para obtener una lista de todas las salas abiertas que aún no han comenzado el juego.
- Esta lista muestra las salas disponibles, junto con el número de jugadores que hay en cada una. Los jugadores pueden elegir unirse a cualquier sala que no haya comenzado.
- La lista de salas puede ser actualizada manualmente mediante un botón, el cual solicita una nueva lista al servidor en tiempo real.

3. Estado de Preparación:

- Después de unirse a una sala, los jugadores deben ingresar su **nombre** en el campo de texto para ser identificados en el juego.
- Posteriormente, pueden marcarse como "**Listos**" mediante un botón interactivo. Este estado de preparación es visible para todos los jugadores en tiempo real, permitiendo que el creador de la sala vea quién está preparado y quién no.

Ver captura en **Anexo A.4**.

Manejo de la Conexión en Tiempo Real:

- La funcionalidad de esta pantalla depende en gran medida de **Socket.IO** para gestionar la comunicación en tiempo real entre el servidor y los jugadores conectados. Algunos de los eventos clave que gestionan esta interacción son:
 - **availableRooms:** Cuando el jugador accede a la pantalla, se envía una solicitud al servidor para obtener una lista actualizada de las salas disponibles. Este evento refresca la lista mostrada en la pantalla.
 - **joinRoom:** Cuando un jugador ingresa un código de sala o selecciona una sala de la lista, este evento se emite al servidor. El servidor verifica si la sala está abierta y, si es así, permite al jugador unirse.
 - **playerJoined:** Una vez que el jugador se une a la sala, este evento actualiza la lista de jugadores para todos los participantes conectados, mostrando el nuevo estado de la sala.

- **playerReadyStatus:** Este evento se encarga de actualizar el estado "Listo/No Listo" de cada jugador, sincronizando esta información en tiempo real para que todos los jugadores puedan ver quién está preparado.
- **startGame:** Cuando todos los jugadores están listos, el creador de la sala puede iniciar el juego. Este evento redirige a todos los jugadores de la sala al primer desafío de manera coordinada y simultánea.

Diseño de la Interfaz:

- La interfaz está diseñada para ser intuitiva y fácil de navegar, con un campo de texto donde los jugadores pueden ingresar el código de sala o unirse a una sala disponible de la lista.
- La lista de salas disponibles está claramente presentada, mostrando detalles importantes como el número de jugadores y el estado del juego (si ha comenzado o no).
- Los jugadores también pueden ingresar su nombre antes de marcarse como "Listos", asegurando que puedan ser identificados fácilmente en la partida.

Conexión con Otras Pantallas:

- Una vez que el jugador se une a la sala y está listo, el creador de la sala puede iniciar el juego. Esto redirige a todos los jugadores conectados a la **pantalla del primer desafío**, asegurando que la experiencia de juego colaborativa fluya sin interrupciones y de manera coordinada.

5.2.5 Pantalla Desafío1

Descripción General: La Pantalla *Desafío 1* es el primer desafío lógico en la aplicación *Escape Room Digital*. En este desafío, los jugadores deben resolver un rompecabezas de imágenes deslizantes, donde el objetivo es reorganizar las piezas para que formen una imagen completa. Esta pantalla está diseñada para ser visualmente atractiva y desafiante, incentivando la participación activa de los jugadores.

Características Técnicas:

1. **Juego de Tablero Deslizante con Imágenes:**
 - La mecánica principal de la pantalla es un **tablero deslizante** compuesto por fragmentos de una imagen. Los jugadores deben mover estas piezas en un espacio vacío para colocarlas en el orden correcto.
 - Las piezas del tablero están representadas como componentes en React que pueden ser reubicados. Los jugadores interactúan con las piezas al hacer clic sobre ellas, lo que provoca un movimiento si es válido (es decir, si la pieza está junto al espacio vacío).
2. **Verificación de Victoria:**
 - La pantalla tiene una **lógica de verificación automática** que revisa constantemente el estado del tablero. Una vez que las piezas están en el orden correcto, el sistema marca el desafío como completado.
 - El sistema compara las posiciones actuales de las piezas con las posiciones esperadas (la solución del rompecabezas). Si coinciden, el jugador recibe una notificación de éxito, y el sistema procede a redirigir a los jugadores al siguiente desafío.

3. Botón de Saltar Desafío:

- Para evitar la frustración de los jugadores, se incluye un **botón de "Saltar Desafío"**. Este botón solo se habilita después de dos minutos (120 segundos) desde el inicio del desafío.
- El temporizador que controla la activación de este botón se inicia en el momento en que el jugador comienza el desafío. Cuando se cumplen los dos minutos, el botón se vuelve visible, permitiendo al jugador optar por saltarse el desafío actual y avanzar al siguiente.

Elementos Adicionales de Interfaz:

- **Explicación del Desafío:**
 - Antes de iniciar el desafío, se muestra una breve **explicación** que guía al jugador sobre el objetivo del juego y cómo interactuar con las piezas del tablero. Esta explicación asegura que los jugadores entiendan las reglas antes de comenzar el desafío.
- **Botón de Ayuda:**
 - La pantalla incluye un **botón de ayuda** que abre un modal. Este modal contiene información sobre los pasos a seguir para completar un Trabajo de Fin de Grado (TFG), que es el tema que sigue el orden de las imágenes en el desafío. Este recurso adicional facilita la comprensión del contexto académico del rompecabezas.
- **Botón de Información TFG:**
 - También se proporciona un **botón de información adicional**, que al hacer clic redirige a una página externa con más detalles sobre el Trabajo de Fin de Grado. Esto ofrece un contexto académico y guía a los jugadores hacia recursos adicionales relevantes.
- **Verificación de Existencia de Sala:**
 - Al cargarse la pantalla, el sistema realiza una verificación para confirmar que la sala de juego aún existe. Esto se hace mediante un llamado al servidor usando **Socket.IO**. Si la sala no es válida o ya no existe, el jugador es redirigido automáticamente al menú de desafíos, evitando errores de conexión o desincronización.

Manejo de la Conexión en Tiempo Real:

- **Interacción en Tiempo Real con Socket.IO:**
 - **Socket.IO** se utiliza para sincronizar el progreso de todos los jugadores en tiempo real, asegurando que todos los participantes en la sala vean el mismo estado del desafío y que los eventos importantes (como la finalización del desafío) se compartan entre todos los jugadores.
 - Eventos principales:
 - **startChallenge:** Se emite cuando los jugadores comienzan el desafío, sincronizando el inicio del temporizador de 2 minutos para el botón de saltar desafío.
 - **finishChallenge:** Cuando el jugador resuelve el rompecabezas, este evento se emite al servidor para marcar la finalización del desafío y proceder al siguiente.
 - **skipChallenge:** Se emite si el jugador decide saltar el desafío después de los dos minutos, permitiéndole avanzar al siguiente reto.

Diseño de la Interfaz:

- **Juego de Tablero Deslizante:** La interfaz del tablero es visualmente clara y fácil de usar, con cada fragmento del rompecabezas representado de manera que los jugadores puedan identificar fácilmente qué pieza necesita ser movida.
- **Elementos Clave:** Los botones para **Saltar Desafío**, **Ayuda** e **Información TFG** están diseñados para ser accesibles y visibles, proporcionando al jugador opciones adicionales sin interrumpir la experiencia de juego.

Conexión con Otras Pantallas:

- Una vez que el jugador resuelve el desafío o elige saltarlo, el sistema redirige automáticamente a la siguiente pantalla de desafío, asegurando una transición fluida y manteniendo la experiencia de juego constante. Toda la interacción y navegación se gestiona en tiempo real a través de **Socket.IO**.

Ver captura en **Anexo A.5**.

5.2.6 Pantalla Desafío2

La *Pantalla Desafío 2* presenta un contenido interactivo basado en la tecnología **H5P**. Este desafío consiste en una actividad de arrastrar y soltar, en la que los jugadores deben organizar correctamente los elementos proporcionados en posiciones específicas. La actividad es tanto educativa como entretenida, ya que está diseñada para mejorar la comprensión de los procedimientos de creación de propuestas de fin de grado.

Características Técnicas:

1. Desafío de Arrastrar y Soltar:

- La pantalla implementa un contenido **H5P** que permite a los jugadores interactuar con el sistema arrastrando y soltando elementos en las posiciones correctas.
- Cada elemento representa un concepto o una parte del proceso necesario para elaborar una **propuesta de Trabajo de Fin de Grado (TFG)**. El objetivo es que el jugador empareje estos elementos de manera correcta, asegurando que comprendan los pasos fundamentales para la creación de una propuesta.
- El componente **H5P** está integrado en la aplicación mediante un contenedor en el que se carga la actividad interactiva. Al finalizar la actividad, el sistema utiliza los eventos de **xAPI** de H5P para monitorear el progreso y la puntuación de los jugadores.

2. Verificación de Progreso:

- El sistema monitorea en tiempo real el progreso del jugador mediante eventos **xAPI** proporcionados por **H5P**. Estos eventos permiten saber cuándo el jugador ha colocado todos los elementos en sus posiciones correctas.
- Si el jugador coloca correctamente todos los elementos y alcanza la **puntuación máxima**, el desafío se marca como completado, y el jugador es redirigido automáticamente al siguiente desafío o al menú de desafíos, según corresponda.

Elementos Adicionales de Interfaz:

- **Explicación del Desafío:**
 - Al iniciar la actividad, se presenta una breve **explicación** que guía a los jugadores sobre cómo interactuar con la mecánica de arrastrar y soltar. Esta explicación es esencial para asegurar que todos los jugadores comprendan tanto las reglas como el objetivo del desafío antes de comenzar.
 - Esta funcionalidad está diseñada para ser clara y comprensible, maximizando la efectividad del desafío como una herramienta tanto educativa como lúdica.
- **Botón de Información TFG:**
 - La interfaz incluye un botón de información que redirige a los jugadores a una página con **información adicional** sobre el Trabajo de Fin de Grado. Este botón está representado mediante un ícono de documento, proporcionando a los jugadores un contexto académico y recursos adicionales para entender mejor el proceso de elaboración de un TFG.
- **Verificación de Existencia de Sala:**
 - Al igual que en otros desafíos, la pantalla verifica la **existencia de la sala** antes de cargar el contenido de **H5P**. Esta verificación se realiza mediante una comunicación con el servidor a través de **Socket.IO**.
 - Si la sala no existe o ha sido eliminada, el jugador es redirigido automáticamente al menú de desafíos, evitando problemas de sincronización o de navegación entre las pantallas del juego.

Manejo de la Conexión en Tiempo Real:

- **Interacción en Tiempo Real mediante Socket.IO:**
 - La aplicación utiliza **Socket.IO** para manejar las interacciones en tiempo real entre los jugadores y el servidor. Esto asegura que el progreso del jugador se actualice de forma instantánea y que la sincronización entre los jugadores en la misma sala sea precisa.
 - Eventos clave de **Socket.IO** que se gestionan en la pantalla incluyen:
 - **startChallenge:** Este evento se emite cuando los jugadores comienzan la actividad de arrastrar y soltar, iniciando el registro del progreso y las interacciones de los jugadores en tiempo real.
 - **finishChallenge:** Al completar la actividad y alcanzar la puntuación máxima, este evento se emite al servidor, registrando la finalización exitosa del desafío.
 - **skipChallenge:** Si los jugadores optan por saltar el desafío después de los dos minutos establecidos, este evento les permite avanzar al siguiente desafío sin necesidad de completar el actual.

Diseño de la Interfaz:

- **Actividad de Arrastrar y Soltar:** La interfaz está diseñada para maximizar la **interactividad**. Los jugadores interactúan con los elementos de la pantalla al arrastrarlos y soltarlos en sus posiciones correctas. Los elementos se destacan visualmente, facilitando la interacción y minimizando la confusión.
- **Botones de Información y Saltar Desafío:** Los botones están posicionados de manera estratégica para ser fácilmente accesibles, asegurando una experiencia fluida y sin interrupciones.

Manejo de la Persistencia de Datos:

- Al igual que en otros desafíos, los jugadores están identificados mediante un **ID persistente** que se almacena localmente. Este **ID** es utilizado para rastrear el progreso del jugador, incluso si se desconectan temporalmente o cambian de dispositivo.

Conexión con Otras Pantallas:

- Una vez que el jugador completa correctamente el desafío o decide saltarlo, es redirigido al siguiente desafío o al menú de desafíos, dependiendo del flujo de juego. Esta transición es manejada mediante **Socket.IO**, asegurando una sincronización perfecta y manteniendo la continuidad en la experiencia del jugador.

Ver captura en **Anexo A.6**.

5.2.7 Pantalla Desafío3

Descripción General: La *Pantalla Desafío 3* es una actividad de tipo cuestionario de opción múltiple, diseñada para evaluar el conocimiento de los jugadores sobre los procedimientos y pasos necesarios para completar un Trabajo de Fin de Grado (TFG). Este desafío presenta preguntas con varias opciones, de las cuales solo una es correcta. Al finalizar correctamente el cuestionario, el jugador puede avanzar al siguiente desafío o al menú de desafíos.

Características Técnicas:

1. Cuestionario de Opción Múltiple:

- El desafío consiste en una serie de preguntas de **opción múltiple**. Cada pregunta ofrece varias respuestas, pero solo una es correcta.
- El sistema verifica automáticamente la respuesta seleccionada por el jugador. Si es correcta, se permite avanzar a la siguiente pregunta. Si no, el jugador debe intentar nuevamente o, si no lo logra en el tiempo límite, tiene la opción de saltar el desafío.

2. Verificación Automática de Respuestas:

- Cuando el jugador selecciona una respuesta, el sistema compara inmediatamente la elección con la respuesta correcta almacenada en el backend.
- Si la respuesta es correcta, se habilita el botón para avanzar a la siguiente pregunta o, si es la última, se completa el desafío.

Elementos Adicionales de Interfaz:

• Progreso a través de Preguntas:

- La pantalla muestra el número de preguntas que el jugador ha respondido correctamente, lo que proporciona una idea clara del progreso dentro del desafío. Este avance es visible en todo momento y actualiza dinámicamente a medida que el jugador responde.
- Si el jugador completa todas las preguntas correctamente, se considera que ha superado el desafío y puede pasar al siguiente nivel.

• Explicación Inicial del Desafío:

- Antes de comenzar el cuestionario, se presenta una explicación breve que detalla las instrucciones del desafío. Esta sección es crucial para asegurar que los

jugadores entiendan el objetivo del cuestionario y la mecánica de interacción con las preguntas.

- La explicación es clara y concisa, asegurando que los jugadores puedan comenzar el desafío sin confusiones.
- **Botón de Información TFG:**
 - Al igual que en otros desafíos, se incluye un botón que redirige a los jugadores a una página web con **información adicional** sobre el Trabajo de Fin de Grado (TFG). Este botón está representado por un ícono de documento y proporciona un contexto académico para el desafío, conectando el juego con la temática real del proyecto.
- **Botón de Saltar Desafío:**
 - El botón de "Saltar Desafío" está disponible después de dos minutos, permitiendo al jugador avanzar al siguiente desafío si no logra completar el cuestionario en el tiempo establecido. El temporizador se inicia desde el comienzo del desafío y el botón se activa automáticamente tras transcurrir 120 segundos, ofreciendo a los jugadores una opción para no quedarse bloqueados.

Manejo de la Conexión en Tiempo Real:

- **Interacción en Tiempo Real mediante Socket.IO:**
 - Al igual que en otros desafíos, la comunicación entre el cliente y el servidor se gestiona mediante **Socket.IO**, lo que permite sincronizar en tiempo real el progreso de los jugadores y el estado del cuestionario.
 - Los eventos principales que se manejan en esta pantalla incluyen:
 - **startChallenge:** Este evento se emite cuando el jugador comienza el cuestionario, registrando la actividad en el servidor.
 - **playerAnswered:** Cada vez que el jugador selecciona una respuesta, se emite este evento al servidor, registrando la respuesta en tiempo real y actualizando el estado del jugador.
 - **finishChallenge:** Una vez que el jugador ha respondido correctamente todas las preguntas o decide saltarse el desafío, este evento se emite al servidor, registrando el tiempo total y la finalización del desafío.
 - **skipChallenge:** Si el jugador decide saltar el desafío después de los dos minutos, el evento se emite para avanzar al siguiente nivel.

Diseño de la Interfaz:

- **Interfaz de Cuestionario:** La interfaz está diseñada para ser **intuitiva** y fácil de usar, con botones prominentes para cada opción de respuesta. Cada opción se presenta claramente, y la selección de la respuesta correcta se verifica automáticamente.
- **Elementos de Ayuda y Progreso:** El progreso del jugador se destaca en la parte superior de la pantalla, lo que ayuda a los jugadores a entender cuántas preguntas les faltan para completar el desafío. Además, la pantalla incluye los botones de información y el botón de saltar desafío, colocados de manera que sean fácilmente accesibles.

Manejo de la Persistencia de Datos:

- **ID Persistente:** Al igual que en otros desafíos, cada jugador cuenta con un **ID persistente** que se almacena localmente. Este ID permite mantener el progreso del jugador, incluso si se desconecta y vuelve a conectarse más tarde.

- El sistema también guarda el progreso de las preguntas respondidas y el tiempo transcurrido en el servidor, asegurando que los jugadores no pierdan su avance si surge alguna interrupción.

Conexión con Otras Pantallas:

- **Transición hacia el Siguiente Desafío:** Una vez que el jugador ha completado correctamente todas las preguntas o decide saltar el desafío, se redirige automáticamente al siguiente nivel o al menú de desafíos. Esta transición está gestionada por **Socket.IO**, lo que asegura una sincronización perfecta con los demás jugadores.
- **Control de Estado de Sala:** La pantalla también verifica si la sala de juego aún existe o ha sido eliminada. Si no se encuentra la sala, el jugador es redirigido al menú principal para evitar problemas de navegación.

Ver captura en **Anexo A.7**.

El conjunto de preguntas utilizadas en este desafío, así como sus opciones y respuestas correctas, se encuentran detalladas en el **Anexo C: Preguntas del Desafío 3**. Este anexo proporciona una visión completa del cuestionario implementado, facilitando la comprensión del contenido evaluado y su correlación con los objetivos del proyecto.

5.2.8 Pantalla Desafío4

La *Pantalla Desafío 4* introduce una experiencia interactiva más avanzada utilizando un juego desarrollado en **Ren'Py**, que simula un proceso de presentación de un Trabajo de Fin de Grado (TFG). A través de una narrativa, los jugadores deben tomar decisiones y completar tareas relacionadas con la defensa de un TFG. Este desafío está diseñado para ser inmersivo, y permite a los jugadores practicar la toma de decisiones en un contexto académico.

Características Técnicas:

1. **Juego Narrativo Interactivo:**
 - El desafío se ejecuta a través de un juego desarrollado en **Ren'Py**, que está incrustado en la pantalla mediante un `iframe`.
 - El jugador avanza en el desafío tomando decisiones en puntos clave de la narrativa. Estas decisiones simulan situaciones que un estudiante enfrentaría durante la presentación y defensa de su TFG.
 - Las decisiones afectan el flujo del juego y la conclusión de la presentación, lo que refuerza la conexión entre el contenido interactivo y el objetivo académico.
2. **Verificación Automática de Finalización:**
 - Durante el desafío, la aplicación escucha eventos generados por el juego de Ren'Py. Si el jugador completa el juego con éxito, el sistema emite un evento para finalizar el desafío y redirigir al jugador al siguiente nivel o al menú de desafíos.
 - En caso de un cierre inesperado o un error en el juego, el sistema maneja este evento enviando un mensaje al servidor y redirigiendo al jugador de vuelta a la pantalla de clasificación o al menú principal.

Elementos Adicionales de Interfaz:

- **Explicación del Desafío:**

- Antes de que el juego comience, se presenta una **explicación** sobre el objetivo del desafío, cómo interactuar con el juego y qué esperar de las decisiones tomadas. Esta guía inicial asegura que los jugadores comprendan el contexto del desafío y cómo avanzar en la narrativa.
- La explicación es esencial para introducir el desafío de manera clara, ya que el entorno interactivo puede ser nuevo para algunos jugadores.
- **Botón de Información TFG:**
 - Al igual que en los otros desafíos, se incluye un **botón de información TFG** que redirige a los jugadores a una página web con detalles adicionales sobre el Trabajo de Fin de Grado (TFG). Este botón proporciona contexto académico relevante y es fácilmente accesible a lo largo del desafío.
- **Botón de Saltar Desafío:**
 - El **botón de "Saltar Desafío"** se activa después de dos minutos de juego. Este temporizador se activa cuando comienza el desafío, y el botón se vuelve visible una vez que han transcurrido 120 segundos, permitiendo a los jugadores omitir el desafío si encuentran dificultades o prefieren no continuar.

Manejo de la Conexión en Tiempo Real:

- **Interacción en Tiempo Real mediante Socket.IO:**
 - La comunicación entre el cliente y el servidor se maneja en tiempo real mediante **Socket.IO**. Los eventos clave incluyen:
 - **startChallenge:** Se emite al iniciar el desafío, registrando el inicio de la actividad en el servidor y monitoreando el tiempo transcurrido.
 - **finishChallenge:** Este evento se emite cuando el jugador finaliza el juego o salta el desafío. Se registra la duración del desafío y el jugador avanza al siguiente nivel.
 - **skipChallenge:** Este evento se dispara si el jugador decide saltar el desafío después de los dos minutos, permitiéndole continuar con el siguiente desafío sin haber completado el juego.
 - **gameError:gameExitedUnexpectedly:** Si el juego de Ren'Py se cierra inesperadamente, este evento se captura para manejar el error y redirigir al jugador a una página adecuada, asegurando que no se quede atrapado en un estado erróneo.

Diseño de la Interfaz:

- **Integración de Ren'Py:**
 - El desafío se presenta en un **iframe** que contiene el juego de Ren'Py. Esto permite que el juego interactivo se ejecute dentro del entorno de la aplicación sin necesidad de que los jugadores descarguen ningún archivo adicional.
 - El iframe ocupa toda la pantalla, proporcionando una experiencia inmersiva en el desafío.
- **Elementos de Control y Navegación:**
 - Además del iframe, la pantalla contiene botones para acceder a la **información del TFG** y para **saltar el desafío**. Estos botones están claramente visibles y accesibles sin interrumpir la experiencia del jugador.
 - El botón de saltar desafío está diseñado para aparecer después de dos minutos de juego, similar a otros desafíos, garantizando la continuidad del flujo de juego si un jugador no desea completar el desafío.

Manejo de la Persistencia de Datos:

- **ID Persistente y Registro de Progreso:**
 - Cada jugador tiene un **ID persistente** que se almacena localmente, lo que permite que se mantenga el progreso incluso si el jugador se desconecta o si el juego de Ren'Py se cierra inesperadamente.
 - Los datos de progreso, incluyendo las decisiones tomadas y el tiempo total invertido en el desafío, se registran en el servidor en tiempo real. Esto asegura que, si el jugador abandona el juego, pueda retomar desde el mismo punto o continuar con el siguiente desafío.

Conexión con Otras Pantallas:

- **Transición al Siguiente Desafío o Pantalla de Clasificación:**
 - Cuando el jugador termina el juego de Ren'Py o decide saltar el desafío, el sistema emite un evento **finishChallenge**, que redirige automáticamente al siguiente desafío o al menú de clasificación, dependiendo de si la sala está en modo individual o multijugador.
 - En caso de un error en el juego, el jugador es redirigido al menú principal o a la pantalla de clasificación si el sistema detecta que la sala ya no está activa o ha sido eliminada.

Ver captura en **Anexo A.8**

Las interacciones completas, incluyendo los diálogos, decisiones y preguntas presentadas en este desafío, se encuentran documentadas en el **Anexo D: Interacciones del Desafío - Presentación de TFG**. Este anexo proporciona una descripción detallada del flujo del juego, las decisiones tomadas por el jugador y sus consecuencias, complementando la información aquí descrita.

5.2.9 Pantalla Clasificación / Leaderboard

Descripción General: La *Pantalla Leaderboard* presenta una clasificación en tiempo real que permite a los jugadores ver su posición relativa dentro de una sala, basada en el tiempo total invertido en completar los desafíos. Esta pantalla es clave para fomentar la competitividad y brindar una visión clara del progreso de cada jugador.

Características Técnicas:

1. **Visualización de Clasificación General:**
 - La pantalla muestra una lista ordenada de jugadores basada en el tiempo total que han tardado en completar los desafíos. Los jugadores son ordenados en la clasificación de menor a mayor tiempo, destacando a los jugadores más rápidos.
 - Cada jugador puede ver su posición en la tabla de clasificación en tiempo real, gracias a la actualización dinámica que se realiza mediante *Socket.IO*.
2. **Detalle de Tiempos por Desafío:**
 - Además del tiempo total, la pantalla incluye una tabla que desglosa el tiempo tomado por cada jugador en cada uno de los desafíos completados. Esto permite a los jugadores identificar en qué desafíos fueron más rápidos o más lentos.
 - Si un jugador no ha completado un desafío, en la tabla aparece la leyenda "No completado" para ese desafío.

Interacciones del Usuario:

- **Visualización de la Clasificación:**
 - Al finalizar los desafíos o cuando el jugador accede a la pantalla de *Leaderboard*, puede ver su posición en la lista. Se muestra su tiempo total acumulado y los tiempos específicos por desafío.
 - La posición del jugador se indica con un mensaje que dice "Eres el Jugador X", donde X es el número correspondiente en la clasificación.
- **Navegación:**
 - Se incluye un botón que permite regresar al menú principal de desafíos, ofreciendo al jugador la posibilidad de reiniciar el juego o participar en una nueva sesión.

Manejo de la Conexión en Tiempo Real:

- **Socket.IO y Actualización Dinámica:**
 - La interacción en esta pantalla se gestiona completamente mediante *Socket.IO*, lo que permite que los datos de la clasificación se actualicen en tiempo real sin necesidad de recargar la página.
 - Los eventos principales que se manejan incluyen:
 - **requestLeaderboard:** Al acceder a la pantalla, se envía una solicitud al servidor para obtener la lista actual de la clasificación.
 - **showLeaderboard:** El servidor responde enviando la lista completa de jugadores, sus tiempos y su posición en la sala, la cual se actualiza en la pantalla.
 - **updateLeaderboard:** Cuando un jugador completa un desafío o se produce un cambio en la sala, este evento se dispara para actualizar la clasificación en tiempo real.
 - **roomDeleted:** Si la sala es eliminada, este evento redirige automáticamente al jugador al menú principal.

Diseño de la Interfaz:

1. **Lista de Clasificación:**
 - Los jugadores se presentan en una lista, ordenados por el tiempo total invertido en completar los desafíos. Cada entrada de la lista incluye el nombre del jugador y su tiempo total.
 - La clasificación está diseñada para ser clara y fácilmente comprensible, destacando el orden de los jugadores con números (1, 2, 3, etc.) y tiempos en un formato legible (minutos y segundos).
2. **Tabla de Tiempos por Desafío:**
 - En la parte inferior de la pantalla se encuentra una tabla que muestra los tiempos de cada jugador desglosados por desafío (Desafío 1, Desafío 2, Desafío 3, y Desafío 4).
 - Si un jugador no ha completado un desafío, la tabla muestra "No completado" en lugar del tiempo.
3. **Botón de Navegación al Menú Principal:**
 - Un botón prominente en la parte inferior de la pantalla permite al jugador regresar al menú principal, facilitando la navegación después de revisar los resultados.

Manejo de la Persistencia de Datos:

- **ID Persistente para Seguimiento de Jugadores:**
 - Al igual que en otras pantallas, el sistema utiliza un **ID persistente** para identificar a los jugadores, lo que permite mostrar su progreso incluso si se desconectan y vuelven a conectarse.
 - El *ID persistente* es fundamental para asegurar que los datos de tiempo y posición se mantengan consistentes en todas las sesiones de juego.

Conexión con Otras Pantallas:

- **Transición desde los Desafíos:**
 - Una vez que un jugador completa un desafío o salta un desafío, es redirigido automáticamente a la pantalla de clasificación para ver su progreso y posición en comparación con los otros jugadores.
 - Si la sala es eliminada o se completa el ciclo de desafíos, el sistema redirige automáticamente a los jugadores al menú principal.

Ver captura en **Anexo A.9**.

5.2.10 Pantalla Challenge Menu

Descripción General: La *Pantalla Challenge Menu* actúa como un centro de navegación que permite a los jugadores seleccionar el desafío que desean jugar dentro de la aplicación "Escape Room Digital". También incluye opciones para volver al menú principal o acceder a información adicional sobre el Trabajo de Fin de Grado (TFG).

Características Técnicas:

1. **Selección de Desafíos:**
 - La pantalla muestra botones numerados que representan cada uno de los desafíos disponibles. Al hacer clic en un botón, el jugador es redirigido al desafío correspondiente mediante el sistema de enrutamiento de la aplicación.
 - Los desafíos están organizados de manera secuencial (Desafío 1, Desafío 2, Desafío 3, Desafío 4), ofreciendo a los jugadores un flujo claro para participar en los diferentes retos.

Interacciones del Usuario:

1. **Navegación entre Desafíos:**
 - El jugador puede seleccionar cualquiera de los desafíos disponibles haciendo clic en los botones correspondientes. Esta interacción dispara una acción de navegación que redirige al jugador a la pantalla del desafío seleccionado mediante el sistema de enrutamiento.
 - La funcionalidad `handleNavigation` se encarga de recibir el número del desafío seleccionado y redirigir al usuario a la ruta adecuada, por ejemplo, `/challenge1` o `/challenge2`.

2. Opciones de Menú:

- Además de los desafíos, la pantalla incluye opciones adicionales en forma de botones:
 - **Menú Principal:** Un botón que redirige al jugador de vuelta a la pantalla principal de la aplicación, permitiendo salir del menú de desafíos y explorar otras opciones.
 - **Información sobre el TFG:** Un enlace externo que lleva al jugador a la página de información del Trabajo de Fin de Grado (TFG). Este enlace se abre en una nueva pestaña del navegador, proporcionando contexto académico adicional al jugador.

Manejo de Navegación:

- **Enrutamiento Dinámico:**
 - El sistema de enrutamiento permite una navegación rápida y sin recargas. Cada botón de desafío está vinculado a una ruta específica dentro de la aplicación (por ejemplo, /challenge1, /challenge2), asegurando que los jugadores puedan navegar entre los desafíos sin interrupciones.
 - El botón de menú principal también utiliza el sistema de enrutamiento para regresar al usuario a la pantalla inicial de la aplicación.

Diseño de la Interfaz:

1. Botones de Desafíos:

- La interfaz presenta cuatro botones prominentes, cada uno etiquetado con el nombre del desafío correspondiente. Estos botones están diseñados para ser fácilmente accesibles y claramente distinguibles, facilitando la interacción para los jugadores.

2. Opciones de Menú:

- Las opciones del menú se presentan en una sección aparte, con un botón para regresar al menú principal y un enlace externo claramente marcado que dirige al jugador a la página del TFG.

Conexión con Otras Pantallas:

- **Navegación hacia los Desafíos:**
 - La pantalla *Challenge Menu* sirve como punto de transición hacia los desafíos específicos de la aplicación. Una vez que el jugador selecciona un desafío, se redirige automáticamente a la pantalla del desafío correspondiente.
- **Regreso al Menú Principal:**
 - Los jugadores pueden abandonar el menú de desafíos en cualquier momento y regresar al menú principal de la aplicación utilizando el botón de "Menú Principal". Esto les permite explorar otras funciones de la aplicación o regresar al flujo de inicio.

Diseño Adaptado a la Experiencia del Usuario:

- El diseño de esta pantalla se enfoca en facilitar la elección del desafío de forma rápida y eficiente. La organización de los botones, su tamaño y la claridad de las etiquetas aseguran que el jugador pueda navegar sin dificultades. Adicionalmente, el acceso a la información del TFG es un valor añadido que conecta el juego con el contexto académico, ofreciendo una experiencia más rica.

Ver captura en **Anexo A.10**.

5.3 Arquitectura del servidor

El servidor está construido con **Node.js**, **Express.js**, **Socket.IO** para manejar la comunicación en tiempo real y **MongoDB** como base de datos para almacenar información sobre las salas de juego, los jugadores y sus progresos en los desafíos.

Conexión a la Base de Datos

El servidor se conecta a **MongoDB** utilizando la URI proporcionada en las variables de entorno. Esta conexión es esencial para gestionar la creación, actualización y consulta de salas y de jugadores. La base de datos almacena información sobre:

- Salas activas.
- Jugadores en cada sala.
- Progreso de los jugadores en los desafíos (tiempos de inicio, tiempos totales y desafíos completados).

Socket.IO: Comunicación en Tiempo Real

El servidor utiliza **Socket.IO** para gestionar las comunicaciones en tiempo real entre el cliente (la aplicación del Escape Room) y el servidor. Esto permite a los jugadores interactuar en tiempo real y realizar las siguientes acciones:

1. **Crear Sala:** Un jugador puede crear una nueva sala, y el servidor asigna un código de sala único. Esta sala se almacena en la base de datos y está disponible para que otros jugadores se unan.
2. **Unirse a Sala:** Los jugadores pueden unirse a una sala existente ingresando el código. El servidor actualiza la lista de jugadores y sincroniza la información con todos los jugadores conectados a esa sala.
3. **Iniciar el Juego:** Cuando todos los jugadores de una sala están listos, el jugador que creó la sala puede iniciar el juego. El servidor emite un evento a todos los jugadores de la sala para comenzar los desafíos.
4. **Progreso en los Desafíos:** El servidor también rastrea el progreso de cada jugador en los desafíos, calculando el tiempo que toma completar cada uno de ellos. La información se almacena y se utiliza para generar la **clasificación** al final del juego.

Gestión de Desafíos y Clasificación

El servidor maneja eventos importantes relacionados con los desafíos:

- **Inicio de Desafíos:** Cuando un jugador inicia un desafío, el servidor registra el tiempo de inicio.
- **Finalización de Desafíos:** Una vez que el jugador completa un desafío, el servidor calcula el tiempo total transcurrido y lo almacena.
- **Clasificación en Tiempo Real:** A medida que los jugadores completan los desafíos, el servidor actualiza la clasificación en tiempo real, mostrando los resultados a todos los jugadores de la sala.

Control de Salas Disponibles

El servidor también proporciona a los jugadores la lista de salas disponibles mediante el uso de eventos de **Socket.IO**. Cada vez que se crea o se actualiza una sala, el servidor emite la nueva lista de salas para que otros jugadores puedan unirse.

6 Principios Heurísticos de Usabilidad

Para garantizar que la aplicación "Escape Room" sea usable y cuente con una buena accesibilidad, se ha llevado a cabo un análisis basado en los diez principios de usabilidad de Nielsen (19). Estos principios son esenciales para diseñar interfaces de usuario efectivas, asegurando que los sistemas sean intuitivos y satisfagan las expectativas de los usuarios.

Visibilidad del estado del sistema

El sistema debe mantener informado al usuario sobre lo que está sucediendo mediante retroalimentación adecuada en un tiempo razonable.

En la aplicación "Escape Room", antes de comenzar cada desafío, se muestra una descripción que informa a los usuarios sobre el contenido del desafío y su objetivo. Además, se indica claramente el número o nombre del desafío actual, lo que ayuda a los usuarios a orientarse dentro de la secuencia del juego. Aunque no hay retroalimentación constante durante el desafío, los jugadores reciben indicaciones claras al inicio de cada uno, asegurando que entienden el contexto y el objetivo antes de comenzar.

Correspondencia entre el sistema y el mundo real

El sistema debe utilizar un lenguaje y conceptos familiares para los usuarios, siguiendo convenciones del mundo real y presentando la información de manera lógica y natural.

En "Escape Room", se utiliza un lenguaje claro y comprensible para explicar los desafíos y las instrucciones. Aunque el contenido está relacionado con la normativa de los Trabajos Fin de Grado (TFG), se presenta de manera que sea accesible y fácil de entender para todos los jugadores, independientemente de su nivel de familiaridad con el tema.

Control y libertad del usuario

Los usuarios deben tener control sobre su navegación dentro del sistema y deben poder salir de situaciones no deseadas sin dificultad.

En la aplicación "Escape Room", los usuarios no pueden regresar al menú principal o cambiar de desafío sin completar el actual, lo cual es intencional para asegurar que cada desafío se aborde completamente antes de avanzar. Esta estructura mantiene el enfoque y la inmersión del jugador, promoviendo una experiencia de juego continua y coherente. La restricción en la navegación asegura que los usuarios sigan el flujo del juego tal como se ha diseñado, lo que refuerza el sentido de logro y progresión a medida que completan cada desafío.

Consistencia y estándares

Los usuarios no deben preguntarse si diferentes palabras, situaciones o acciones significan lo mismo. Es importante seguir las convenciones de la plataforma y utilizar un lenguaje coherente.

La aplicación mantiene una consistencia en el uso de colores, botones y tipografías a lo largo de todas las pantallas de los desafíos, lo que ayuda a los usuarios a reconocer acciones y funciones fácilmente.

Prevención de errores

Es preferible diseñar el sistema de manera que se eviten errores, en lugar de mostrar mensajes de error después de que estos ocurran.

En "Escape Room", se implementan validaciones para asegurar que los usuarios no puedan avanzar sin completar los requisitos de cada desafío, como contestar todas las preguntas de un cuestionario o completar una actividad de arrastrar y soltar.

Reconocimiento en lugar de recuerdo

Los objetos, acciones y opciones deben ser visibles y fácilmente accesibles, para que los usuarios no tengan que recordar información de una pantalla a otra.

La aplicación utiliza iconos descriptivos y etiquetas de texto claras en los botones para ayudar a los usuarios a identificar fácilmente las funciones y opciones disponibles en cada pantalla.

Flexibilidad y eficiencia de uso

El sistema debe ofrecer opciones para que tanto usuarios novatos como avanzados puedan interactuar de manera eficiente.

En "Escape Room" no incluye atajos ni opciones avanzadas para usuarios experimentados, ya que se enfoca en la simplicidad de uso para todos los jugadores.

Diseño estético y minimalista

Los diálogos y las interfaces no deben contener información irrelevante o rara vez necesaria. La simplicidad es clave para mejorar la usabilidad.

La aplicación "Escape Room" adopta un diseño limpio y minimalista, con un esquema de colores uniforme que facilita la lectura y comprensión de las instrucciones y desafíos, evitando la sobrecarga de información.

Ayuda a los usuarios a reconocer, diagnosticar y recuperarse de errores

Los mensajes de error deben ser claros, indicar con precisión el problema y sugerir una solución constructiva.

En la aplicación "Escape Room", actualmente no se implementan mensajes de error detallados para situaciones como la desconexión de la sala o problemas de sincronización. Esto representa un área de mejora, ya que proporcionar mensajes de error claros ayudaría a los usuarios a entender qué ha salido mal y qué pasos pueden tomar para resolver el problema.

Ayuda y documentación

Aunque es ideal que el sistema sea intuitivo y no requiera documentación extensa, siempre es útil proporcionar ayuda accesible.

En "Escape Room", cada pantalla de desafío incluye una breve explicación inicial y un botón de ayuda que proporciona información adicional. Para una mejora futura, se podría considerar la integración de un sistema de ayuda más completo o un asistente virtual para responder a preguntas frecuentes.

7 Licencia Software y Documental

Llegados a este apartado, se va a proceder a comentar tanto la licencia de software como la licencia documental.

En cuanto a la licencia de software se va a emplear Berkeley Software Distribution (BSD). Se trata de una licencia de software libre permisiva como puede ser OpenSSL o la MIT License. Existen diferentes tipos de licencias, en el caso de este TFG se ha utilizado la licencia “BSD modificada”, “BSD revisada”, “BSD-3” o “BSD de 3 cláusulas”.



Figura 12 Logotipo de la licencia BSD

Al igual que sucede en el mundo del software, se tienen que buscar formas de garantizar las libertades asociadas al trabajo elaborado y su inviolabilidad futura. Para garantizar que la libertad esté asociada al documento se buscan métodos, uno de ellos es la licencia GNU Free Documentation License (GFDL).



Figura 13 Logotipo de la licencia GNU

El propósito de esta Licencia es hacer que en el caso de este TFG sea 'gratuito' en el sentido de libertad: para asegurar a todos la libertad efectiva de copiarlo y redistribuirlo, con o sin modificarlo, ya sea comercial o no comercialmente. En segundo lugar, esta licencia preserva para el autor y el editor una forma de obtener crédito por su trabajo, sin ser considerado responsable de las modificaciones realizadas por otros. Es una especie de 'copyleft', lo que significa que las obras derivadas del documento deben ser libres en el mismo sentido. Si por algún motivo se emplea este documento y se modifica, debe realizar una serie de acciones indicadas en el sitio web oficial de GNU.

Tampoco hay que olvidar que este documento, por defecto, está al amparo de la licencia 7.3, por su inclusión en el Repositorio Institucional de Documentos de la Universidad de Zaragoza: ZAGUAN



Figura Z. Licencia de ZAGUAN

8 Conclusiones y Trabajo a futuro

En este apartado se presentan las conclusiones generales derivadas del desarrollo del proyecto "Escape Room Digital sobre Normativa de TFE" y se reflexiona sobre los logros técnicos, conceptuales y personales alcanzados. Asimismo, se exponen las áreas que ofrecen oportunidades de mejora y expansión en futuros desarrollos, tomando en cuenta las lecciones aprendidas durante el proceso de implementación y las necesidades que podrían surgir en el uso continuo de la aplicación.

8.1 Conclusiones

Este proyecto tenía como objetivo crear una aplicación capaz de motivar a los estudiantes a familiarizarse mejor con la normativa de los Trabajos Fin de Estudios (TFE) en la Escuela Universitaria Politécnica de Teruel (EUPUT). En lugar de utilizar tecnologías tradicionales o específicas de plataforma, como Android y Java, se optó por una solución basada en tecnologías web modernas. Esta decisión permitió ofrecer una mayor accesibilidad y una experiencia más enriquecedora para los usuarios. La implementación de herramientas como React para la interfaz, Node.js y Socket.IO para el backend y la comunicación en tiempo real, junto con H5P para contenido interactivo y Ren'Py para narrativas visuales, facilitó la creación de una aplicación dinámica accesible desde cualquier navegador (ver Anexo B para detalles sobre las pruebas de visualización en navegadores).

El uso de estas tecnologías ha permitido la creación de una plataforma modular y escalable, capaz de integrar nuevos desafíos o modificar los existentes sin comprometer la estabilidad del sistema. Además, la integración de Socket.IO ha facilitado la interacción en tiempo real entre los usuarios, enriqueciendo la dinámica de los desafíos y proporcionando una experiencia de juego más fluida y colaborativa. La estructura modular y la elección de tecnologías también han mejorado la capacidad del sistema para manejar múltiples sesiones simultáneas, asegurando una experiencia de usuario consistente y sin interrupciones.

Este proyecto no solo permitió explorar diferentes aspectos del desarrollo de software, como la gestión del estado en React o la comunicación en tiempo real en entornos multiusuario, sino que también subrayó la importancia de la usabilidad y la accesibilidad en el diseño de aplicaciones educativas. A lo largo del desarrollo, se adquirieron conocimientos valiosos tanto en la implementación técnica como en la conceptualización y el diseño de experiencias de aprendizaje efectivas.

El éxito en el desarrollo de esta aplicación se debe en gran medida a los conocimientos adquiridos en las asignaturas clave del plan de estudios. La asignatura de Ingeniería del Software proporcionó los principios necesarios para gestionar el ciclo de vida del proyecto, desde la planificación hasta la entrega final, garantizando un desarrollo coherente y estructurado.

Las asignaturas de Bases de Datos 1 y Bases de Datos 2 fueron fundamentales para el diseño y la gestión eficiente del sistema de almacenamiento de datos utilizando MongoDB, lo que permitió que la aplicación manejara de manera efectiva la información de los usuarios y los desafíos.

Asimismo, la asignatura de Ingeniería Web fue clave para la implementación del frontend en React, facilitando la creación de una interfaz interactiva y adaptable a diversas plataformas. Esta asignatura también resultó crucial para comprender y aplicar tecnologías web modernas, asegurando la correcta integración entre el frontend y el backend de la aplicación.

En última instancia, el proyecto "Escape Room" demuestra cómo la combinación de tecnologías modernas y principios de diseño centrados en el usuario puede crear aplicaciones educativas innovadoras y efectivas. La aplicación ha demostrado que es posible ofrecer una experiencia educativa significativa que sea tanto entretenida como instructiva, utilizando la gamificación para facilitar la comprensión de conceptos complejos y mejorar el compromiso de los estudiantes. Este enfoque no solo enriquece el aprendizaje, sino que también abre nuevas posibilidades para el desarrollo de futuras aplicaciones educativas basadas en tecnologías emergentes.

8.2 Trabajo futuro

A pesar del éxito en la implementación de los desafíos y la arquitectura general de la aplicación, hay varios aspectos que pueden mejorarse y expandirse en futuros desarrollos. Algunas áreas clave de trabajo futuro incluyen:

- **Accesibilidad Mejorada:** Si bien se han implementado prácticas básicas de accesibilidad, algunos desafíos pueden no ser completamente accesibles para usuarios con discapacidades. En futuras iteraciones, se debería trabajar en la implementación de controles y alternativas de navegación que no dependan exclusivamente del uso del ratón, como comandos de voz o teclado.
- **Ampliación de Desafíos:** Se podrían desarrollar y agregar más desafíos, utilizando diferentes tecnologías y metodologías, para ofrecer una experiencia de juego más diversa y rica. Cada nuevo desafío debería seguir los principios de modularidad para facilitar su integración.
- **Optimización del Rendimiento:** A medida que se agregan más características y desafíos, es crucial asegurar que el rendimiento de la aplicación se mantenga ágil. Se deben considerar técnicas de optimización de carga, como la carga diferida de componentes y la optimización de la gestión de estado en React.
- **Funcionalidades Adicionales:** Implementar características como la posibilidad de reanudar desafíos desde el último punto guardado, un sistema de puntos o recompensas, y la integración con redes sociales para compartir resultados puede aumentar el compromiso del usuario.
- **Mejoras en la Seguridad:** A medida que la aplicación crece y potencialmente maneja más datos de usuarios, la seguridad se convierte en una prioridad. Implementar medidas más robustas de seguridad y autenticación garantizará que los datos de los usuarios y la integridad de la aplicación estén protegidos.
- **Soporte Multilenguaje:** Ampliar la accesibilidad de la aplicación a usuarios que hablen diferentes idiomas. Implementar un sistema multilingüe permitirá a la aplicación llegar a una audiencia global.
- **Integración de Inteligencia Artificial:** En futuros desarrollos, la inteligencia artificial podría utilizarse para personalizar la dificultad de los desafíos o proporcionar retroalimentación en tiempo real a los usuarios, mejorando la experiencia de aprendizaje y juego.

Bibliografía

1. *The “What” and “Why” of Goal Pursuits: Human Needs and the Self-Determination of Behavior.* **Deci, E. L. y Ryan, R. M.** 4, 2000, *Psychological Inquiry*, Vol. 11, págs. 227-268.
2. **Gee, J. P.** *What Video Games Have to Teach Us About Learning and Literacy.* s.l. : Palgrave Macmillan, 2003.
3. **Kapp, K. M.** *The Gamification of Learning and Instruction: Game-based Methods and Strategies for Training and Education.* s.l. : Pfeiffer, 2012.
4. **Cerdeira Porras, María.** *Enseñanza de las matemáticas a través de la gamificación. Diseño de un escape room para 1º de primaria.* Universidad de Sevilla. 2020.
5. **Checiches, Adela Rodica.** *Revisión literaria científica y propuesta de escape room en el aula de educación física.* Universidad de Sevilla. 2019.
6. *Escape Rooms in STEM Teaching and Learning—Prospective Field or Declining Trend? A Literature Review.* **Lathwesen, Chantal y Belova, Nadja.** 6, 2021, *Education Sciences*, Vol. 11, pág. 308.
7. *Beyond the Early Adopters: Escape Rooms in Science Education.* **Veldkamp, Alice, Knippels, Marie-Christine P. J. y van Joolingen, Wouter R.** 2021. *Frontiers in Education*.
8. *Earning through Challenges and Enigmas: Educational Escape Room as a Predictive Experience of Motivation in University Students.* **González-Yubero, Sara, y otros.** 17, 2023, *Sustainability*, Vol. 15, pág. 13001.
9. *Escape Rooms in STEM Teaching and Learning—Prospective Field or Declining Trend? A Literature Review.* **Lathwesen, Chantal y Belova, Nadja.** 6, 2021, *Education Sciences*, Vol. 11, pág. 308.
10. *Beyond the Early Adopters: Escape Rooms in Science Education.* **Veldkamp, Alice, Knippels, Marie-Christine P. J. y van Joolingen, Wouter R.** 2020, *Frontiers in Education*.
11. **Inc., Facebook.** *React: A JavaScript Library for Building User Interfaces.* s.l. : Facebook Open Source, 2013.
12. **Documentation, Node.js.** *Node.js Documentation.* s.l. : OpenJS Foundation, 2024.
13. **Documentation, Express.** *Express - Node.js web application framework.* 2024.
14. **Documentation, Socket.IO.** *Socket.IO - Documentation.* 2024.
15. **Documentation, MongoDB.** *MongoDB Documentation.* 2024.
16. **Documentation, H5P.** *H5P - Create and Share Interactive Content.* 2024.
17. **Documentation, Ren'Py.** *Ren'Py Visual Novel Engine Documentation.* 2024.
18. **Documentation, GitHub.** *GitHub Documentation.* 2024.
19. **Nielsen, Jakob.** 10 Usability Heuristics for User Interface Design. *Nielsen Norman Group.* [En línea] 1995. <https://www.nngroup.com/articles/ten-usability-heuristics/>.

Anexo

Anexo A: Capturas de Pantallas de la Aplicación

En este anexo se incluyen las capturas de pantalla de las diferentes vistas de la aplicación "**Escape Room Digital**", que ilustran el diseño de la interfaz y las interacciones en cada una de las pantallas descritas en el desarrollo de la aplicación. Cada captura está numerada para su referencia en el documento principal, permitiendo una fácil correlación entre el texto y la interfaz.

A.1 Pantalla Home

La **Pantalla Home** es el primer punto de interacción para el usuario, donde se presenta un mensaje de bienvenida y la opción de comenzar el juego. Esta pantalla incluye un botón destacado que redirige a las opciones de crear o unirse a una sala de juego.

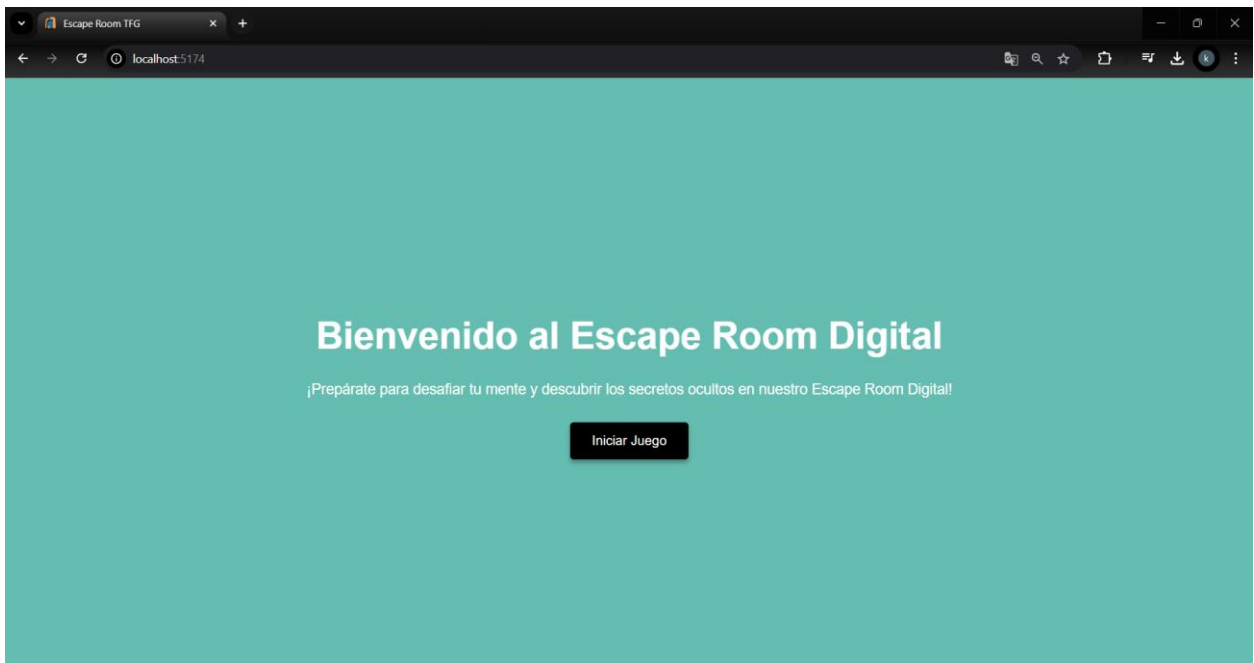


Figura 14 Pantalla Home

A.2 Pantalla “Elige una opción”

En la **Pantalla "Elige una opción"**, el usuario puede seleccionar entre dos opciones principales: **Crear Sala** o **Unirse a una Sala**. Esta pantalla actúa como puente entre la pantalla de bienvenida y las funcionalidades de gestión de salas.

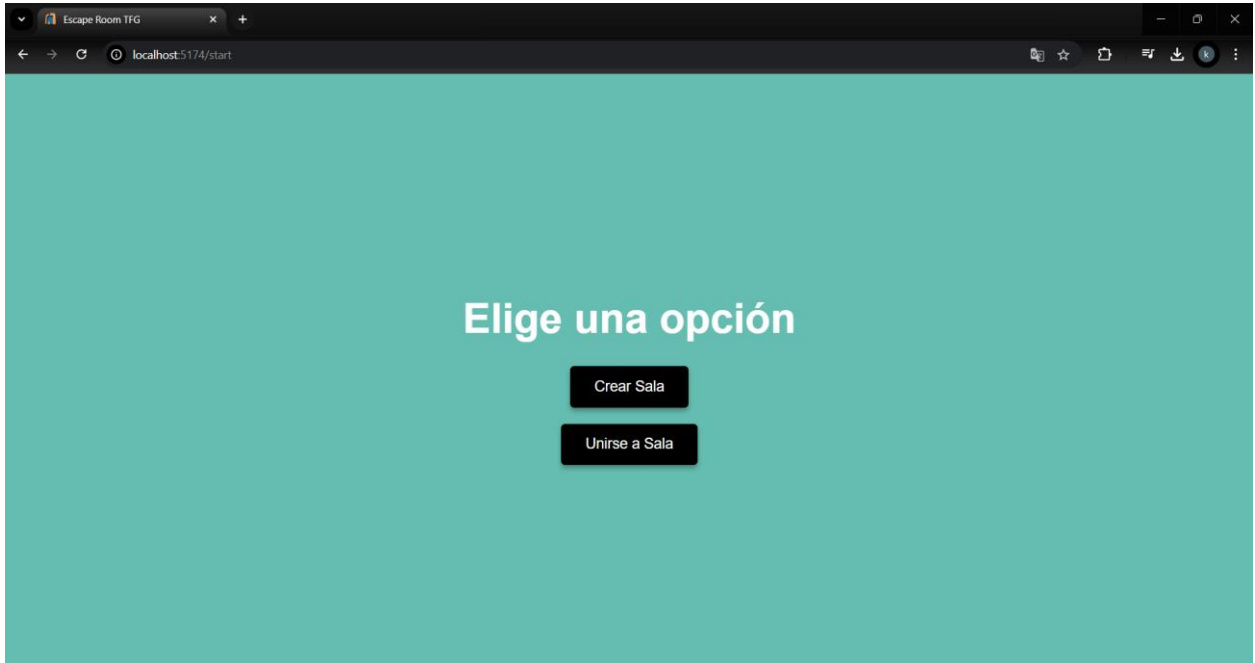


Figura 15 Pantalla "Elige una opcion"

A.3 Pantalla crear Sala

La **Pantalla Crear Sala** permite al usuario generar una nueva sesión de juego. Se muestra un código único que puede ser compartido con otros jugadores, y los jugadores que se unen son listados junto con su estado de preparación (Listo/No Listo).

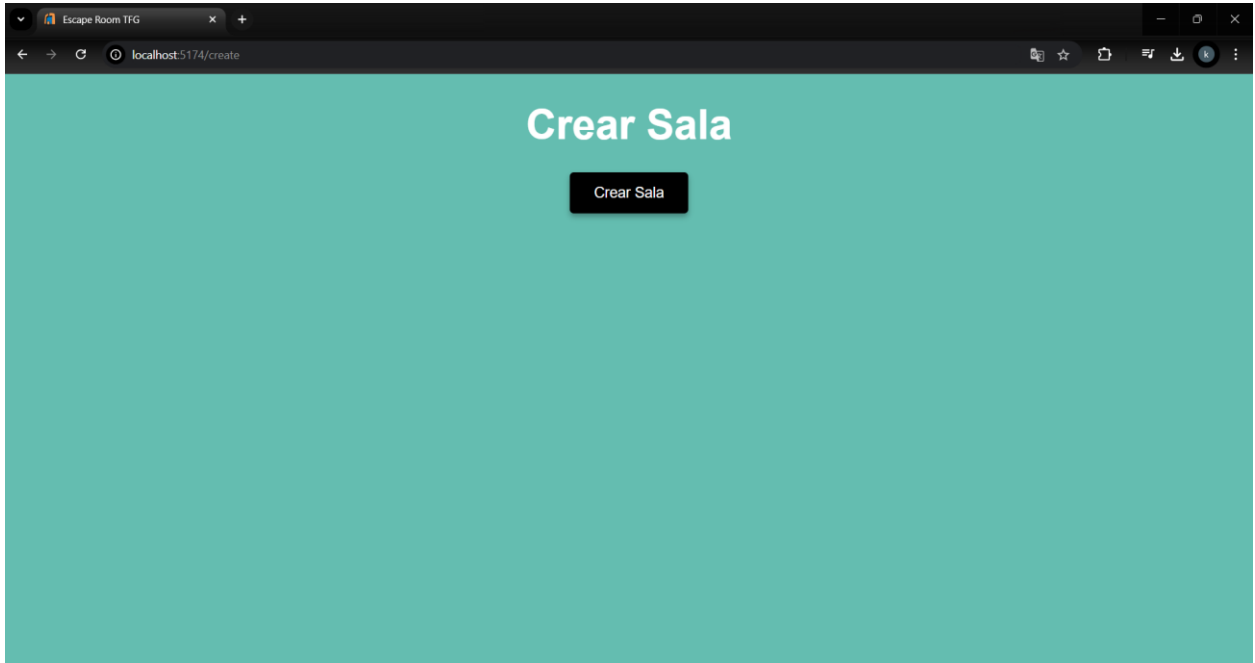


Figura 16 Pantalla Crear Sala - Generar Código

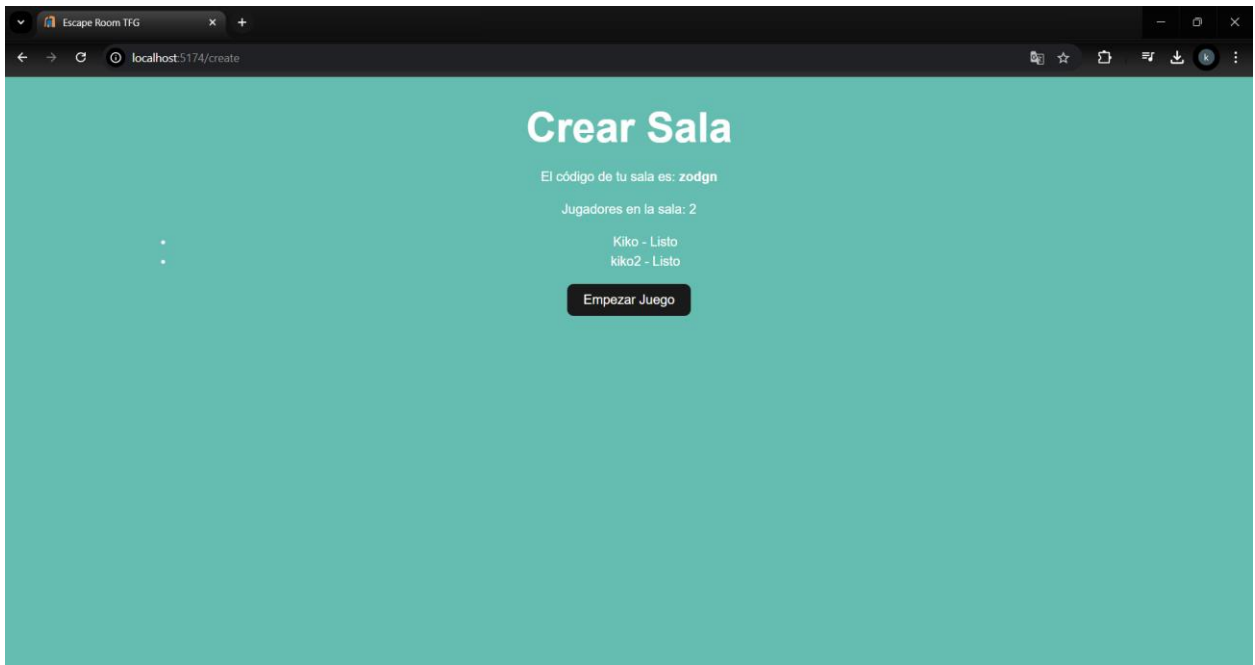


Figura 17 Pantalla Crear Sala - Código de sala Generado

A.4 Pantalla Unirse a Sala

En la Pantalla Unirse a Sala, los jugadores pueden ingresar un código de sala o seleccionar una sala disponible de la lista mostrada. Una vez unidos, pueden ver la lista de jugadores conectados y su estado de preparación.

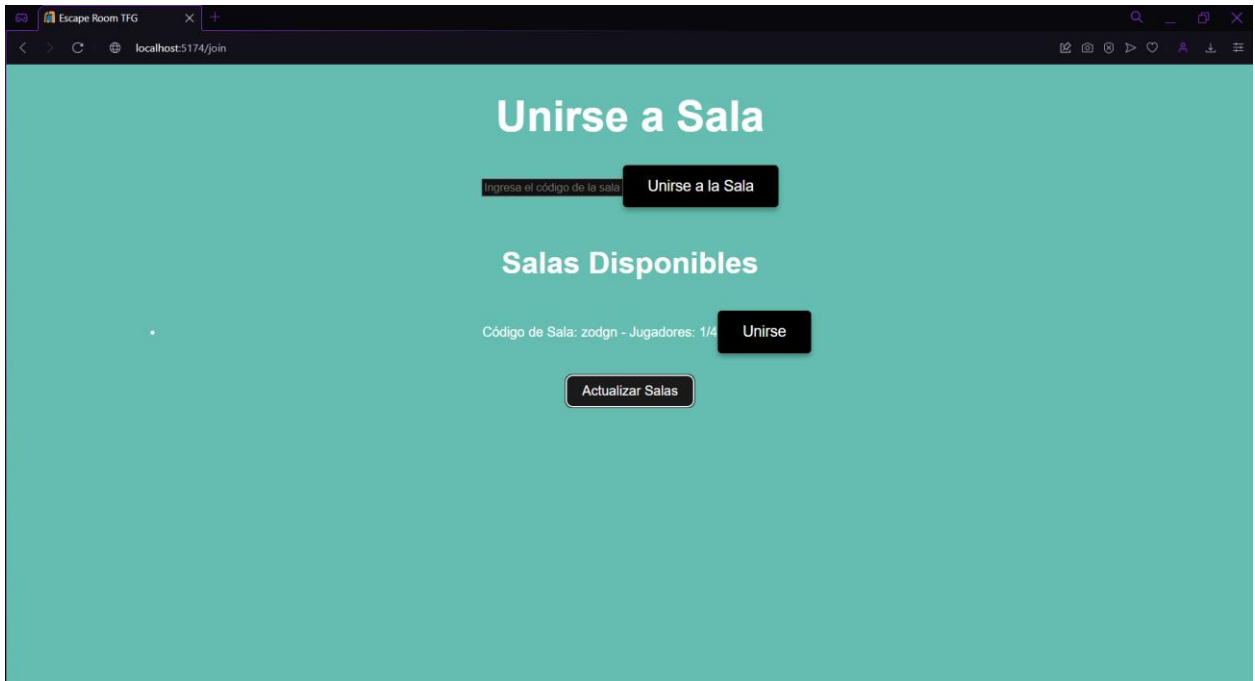


Figura 18 Pantalla Unirse a sala - Introducir Código

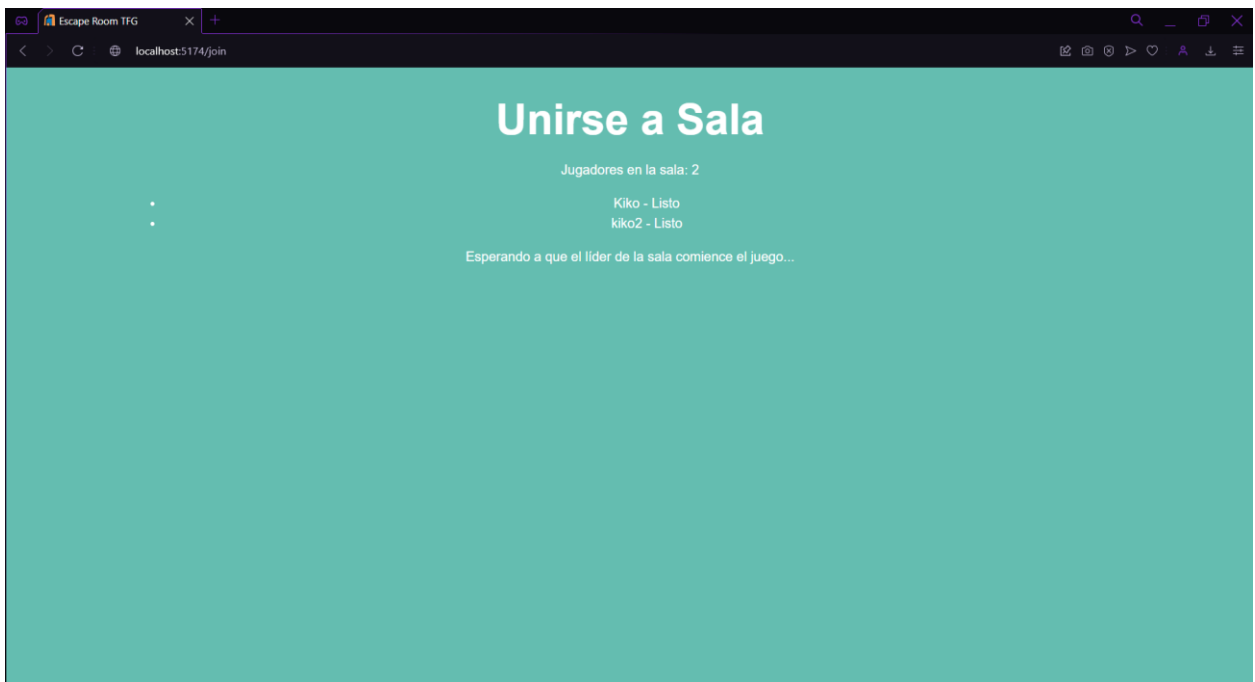


Figura 19 Pantalla Unirse a Sala - Jugadores Conectados

A.5 Pantalla Desafío 1

La Pantalla Desafío 1 muestra un juego de imágenes deslizantes. El objetivo del usuario es ordenar las piezas correctamente para completar el desafío. Incluye un temporizador de dos minutos, tras el cual aparece un botón para saltar el desafío.

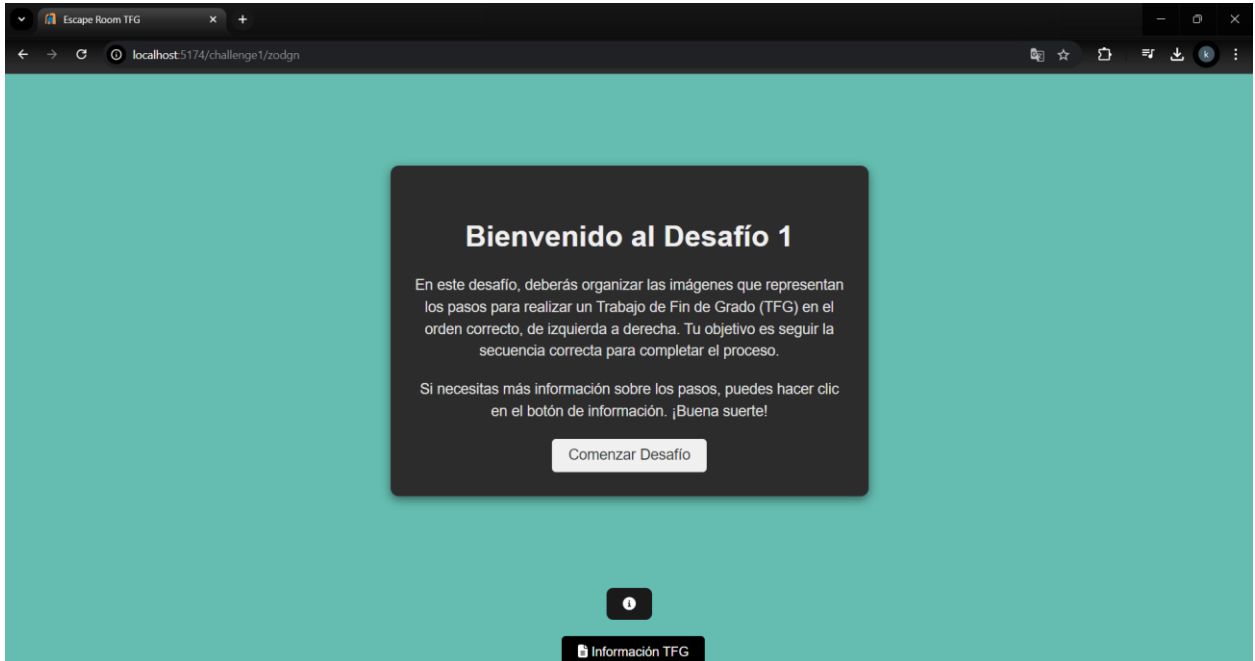


Figura 20 Pantalla Desafío 1 - Introducción

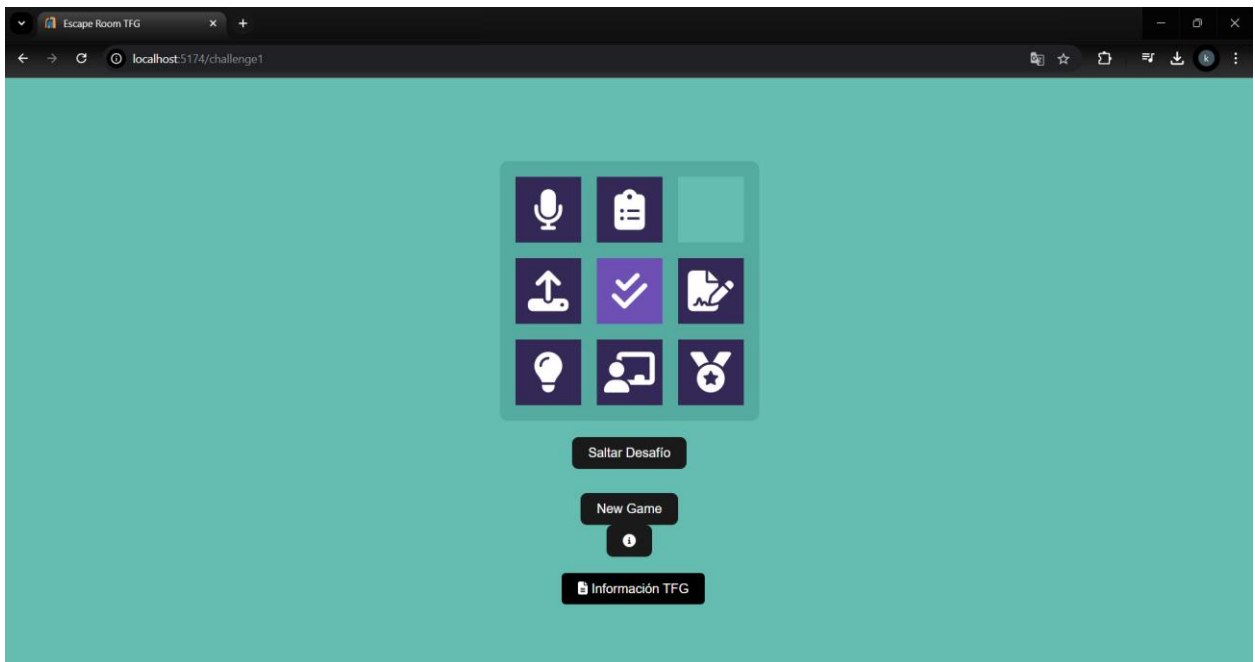


Figura 21 Pantalla Desafío 1 - Desafío

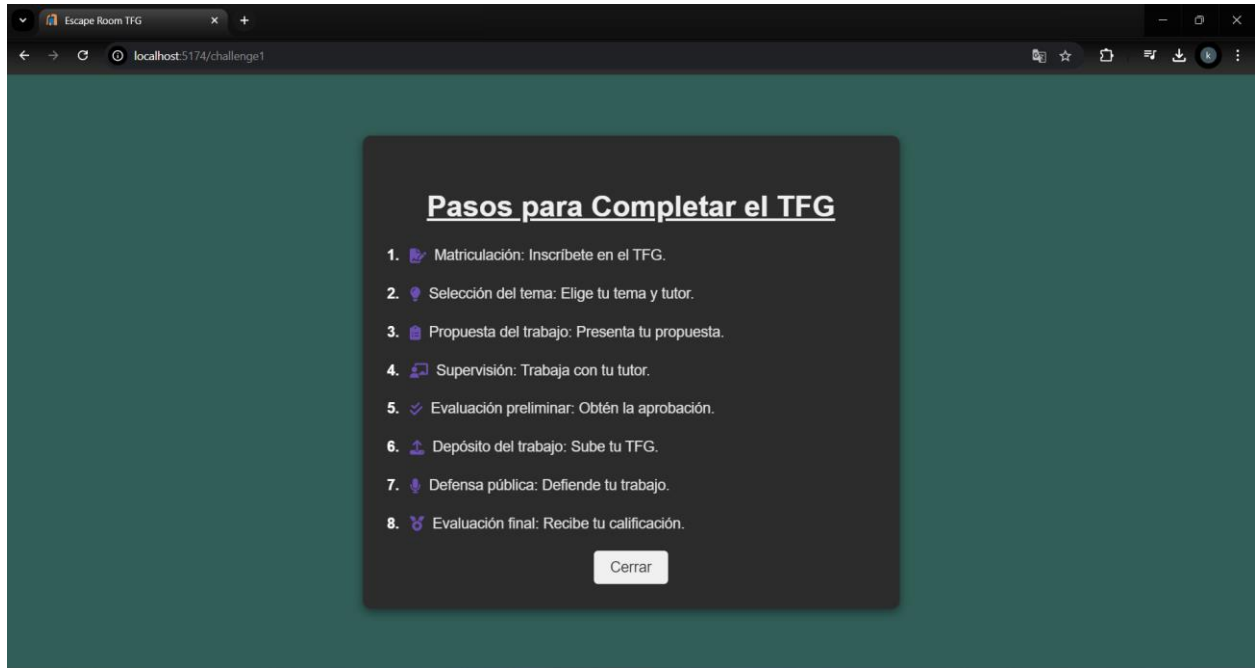


Figura 22 Pantalla Desafío 1 - Pantalla ayuda

A.6 Pantalla Desafío 2

La Pantalla Desafío 2 presenta un contenido interactivo H5P de arrastrar y soltar, en el que el jugador debe emparejar conceptos. Se incluye una verificación automática que evalúa si los elementos están en las posiciones correctas.



Figura 23 Pantalla Desafío 2 - Introducción

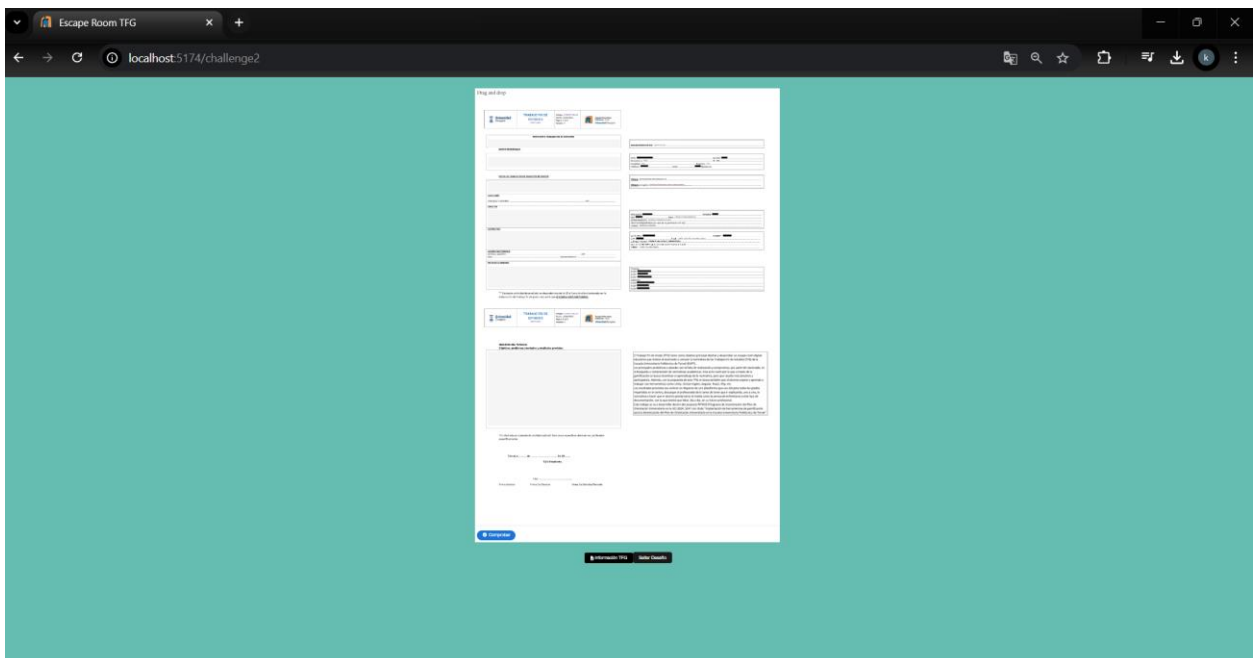


Figura 24 Pantalla Desafío 2 - Desafío

A.7 Pantalla Desafío 3

La Pantalla Desafío 3 ofrece un cuestionario interactivo. El usuario debe responder correctamente las preguntas de opción múltiple relacionadas con el proceso del Trabajo de Fin de Grado (TFG) para avanzar. Las preguntas deben responderse de manera correcta para desbloquear la siguiente pregunta.



Figura 25 Pantalla Desafío 3 - Introducción



Figura 26 Pantalla Desafío 3 - Desafío

A.8 Pantalla Desafío 4

La Pantalla Desafío 4 utiliza un juego interactivo creado con Ren'Py. En este desafío, el usuario debe tomar decisiones a lo largo de una narrativa interactiva. Se registran las decisiones y se permite saltar el desafío tras dos minutos de inactividad.

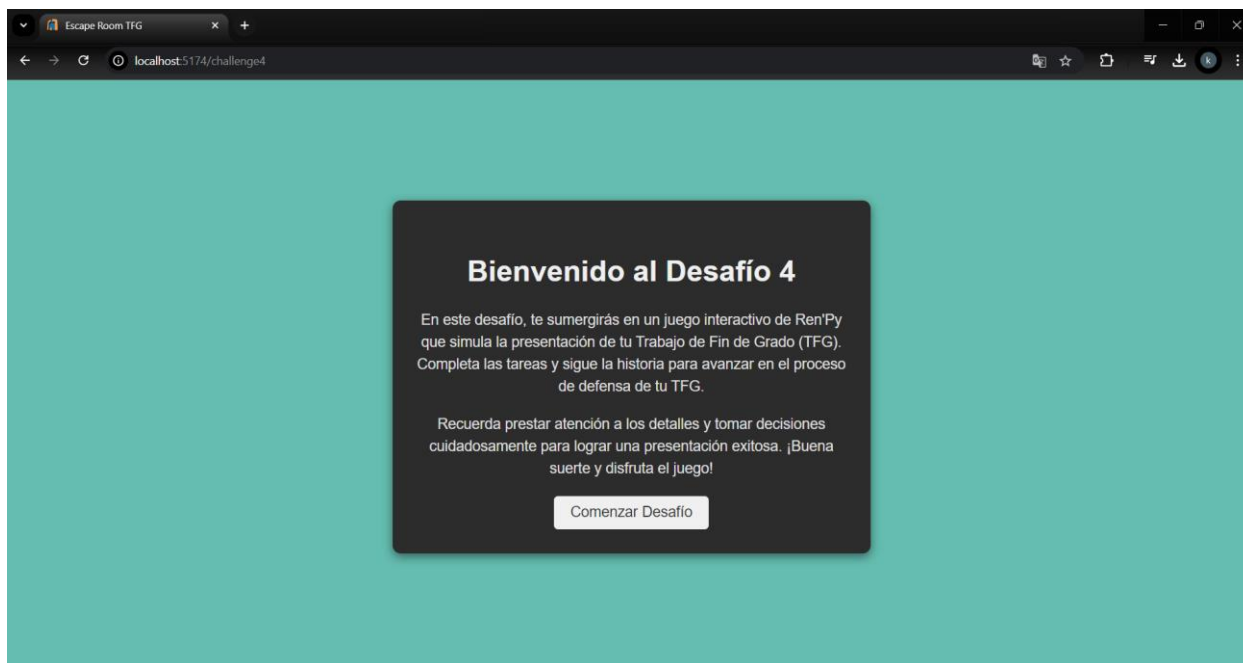


Figura 27 Pantalla Desafío 4 - Introducción

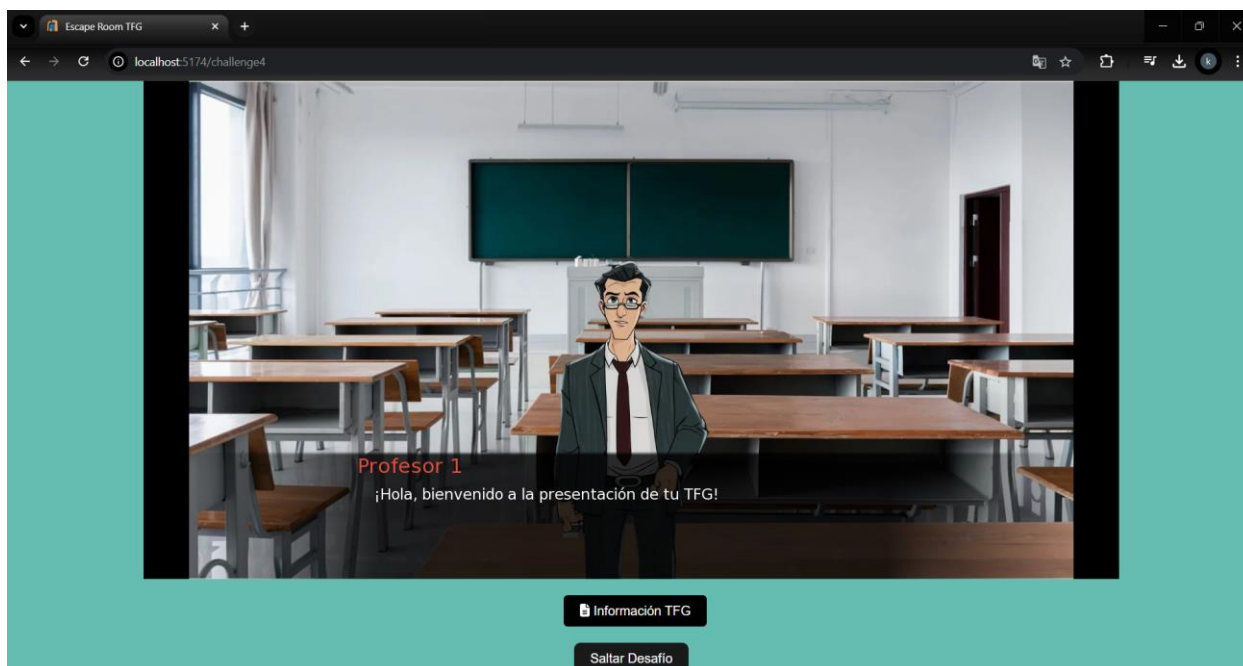
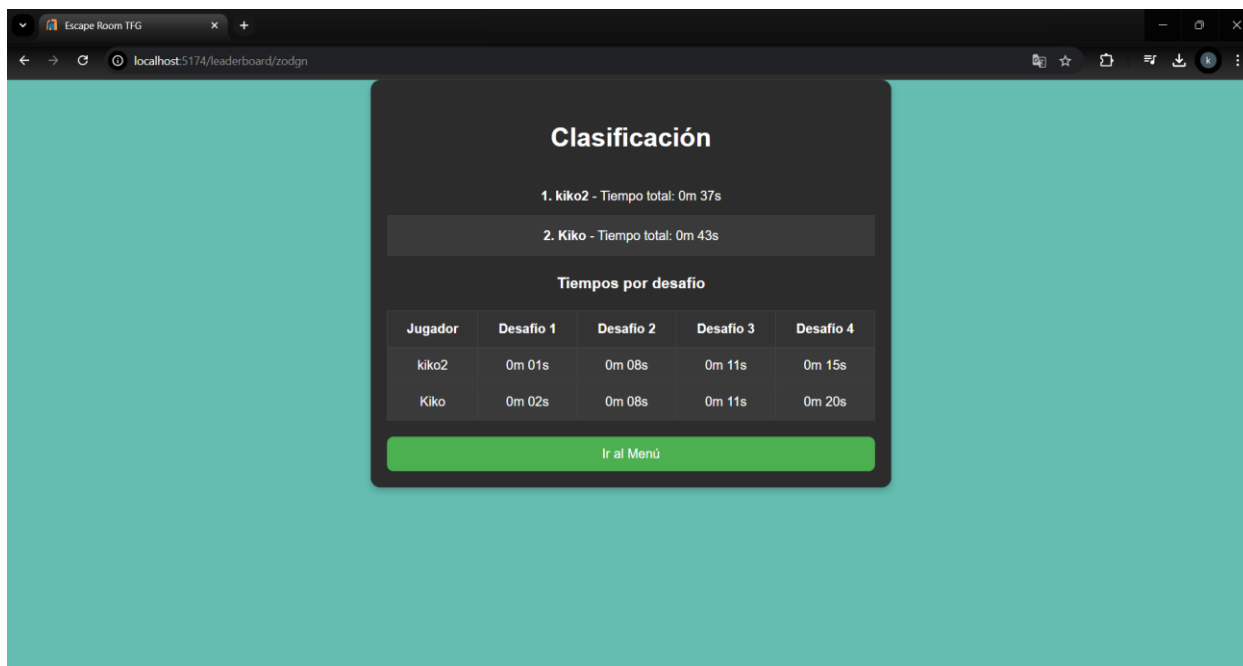


Figura 28 Pantalla Desafío 4 - Desafío

A.9 Pantalla Clasificación

La Pantalla Clasificación muestra los resultados finales de los jugadores tras completar los desafíos. Se actualiza en tiempo real y permite visualizar los tiempos de cada desafío y el tiempo total de cada jugador.



Clasificación				
1. kiko2 - Tiempo total: 0m 37s				
2. Kiko - Tiempo total: 0m 43s				
Tiempos por desafío				
Jugador	Desafío 1	Desafío 2	Desafío 3	Desafío 4
kiko2	0m 01s	0m 08s	0m 11s	0m 15s
Kiko	0m 02s	0m 08s	0m 11s	0m 20s
Ir al Menú				

Figura 29 Pantalla Clasificación

A.10 Pantalla Menú Desafíos

El Menú Desafíos permite al usuario seleccionar el desafío que desea jugar. Se presenta como una lista de botones que redirigen a los diferentes desafíos.

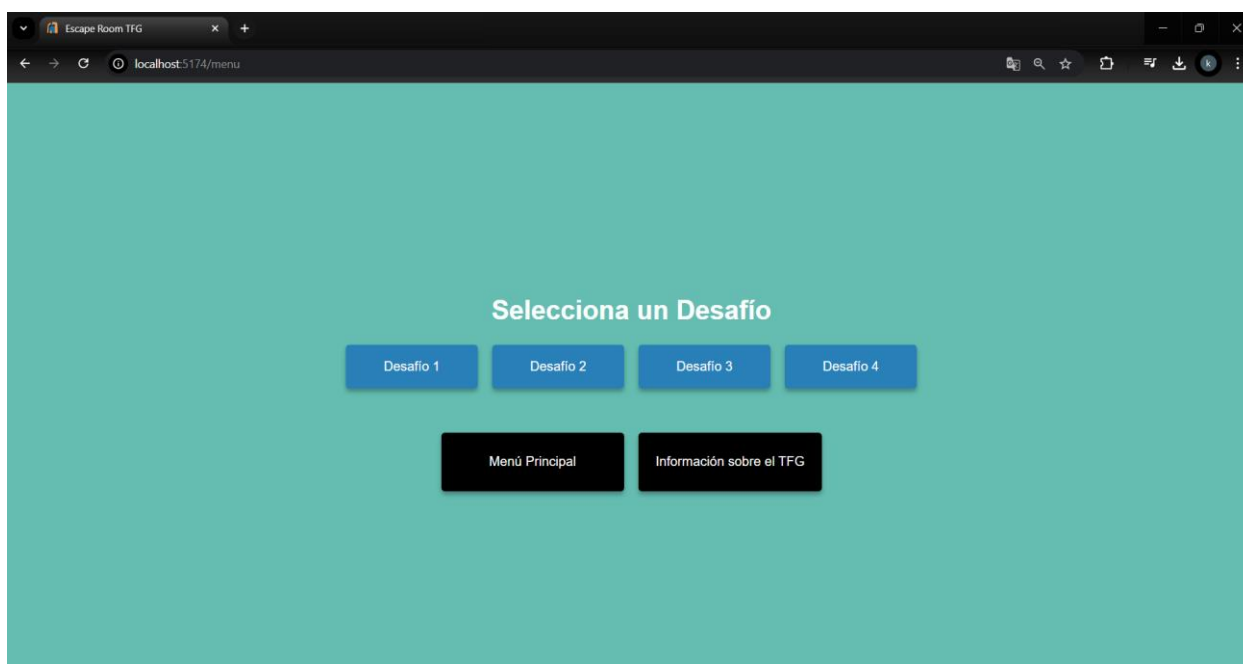


Figura 30 Pantalla Menú Desafíos

Anexo B: Pruebas de Visualización en Diferentes Navegadores

En este anexo se incluyen los resultados de las pruebas de visualización realizadas en diferentes navegadores para asegurar la compatibilidad y correcto funcionamiento de la aplicación "Escape Room Digital" en diversos entornos.

B.1 Prueba en Safari

Se realizó una prueba de visualización en Safari, asegurando que la interfaz mantiene su integridad visual, así como la funcionalidad del botón de inicio.

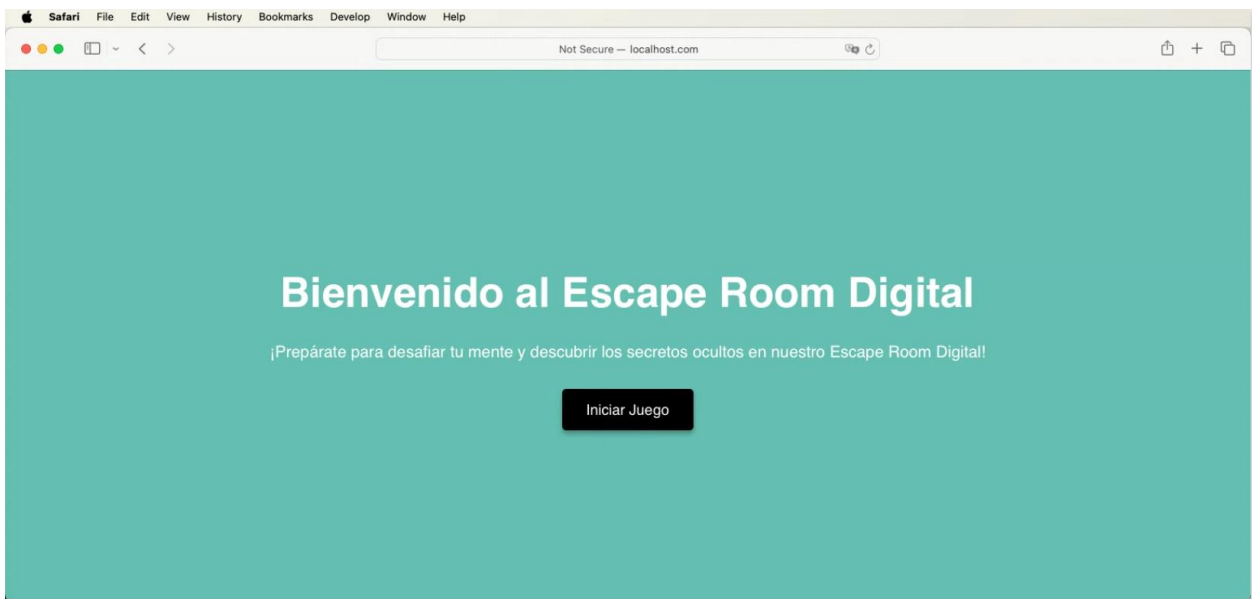


Figura 31 Prueba visualización Safari

B.2 Prueba en Mozilla Firefox

En esta prueba, se visualiza la aplicación en Mozilla Firefox, comprobando que los elementos gráficos y las funcionalidades son consistentes y que no hay problemas de renderización.

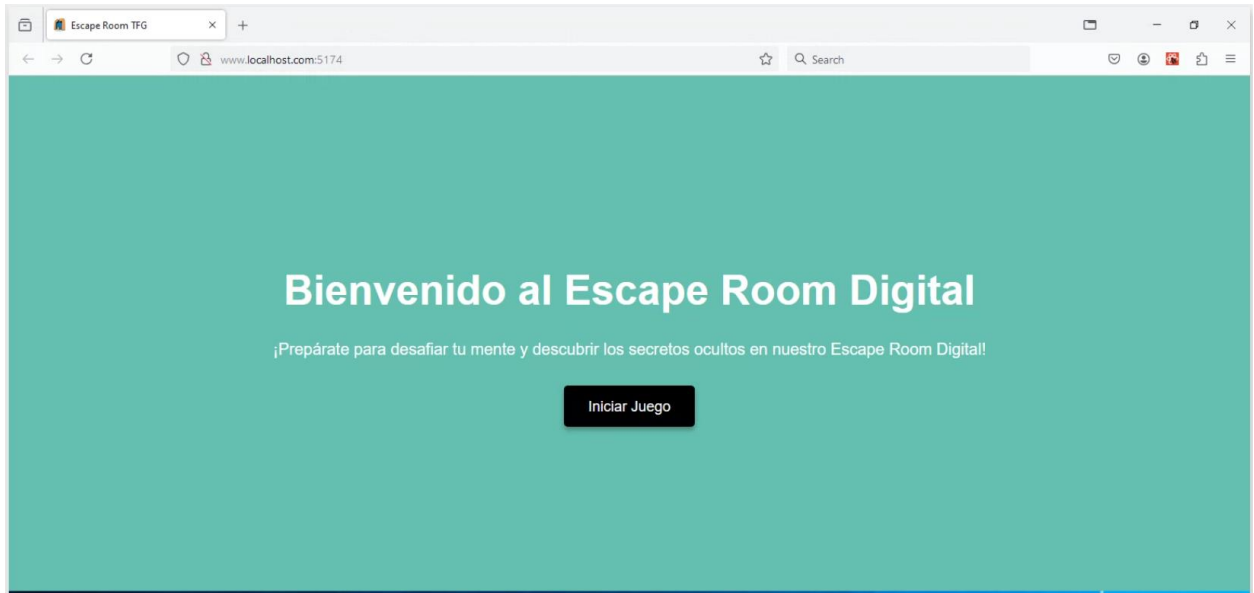


Figura 32 Prueba visualización Mozilla Firefox

B.3 Prueba en Chrome

Esta prueba muestra la Pantalla Home ejecutada en Google Chrome, donde se verifica que los elementos de la interfaz, como el título y el botón, se visualizan correctamente sin distorsiones en la presentación.

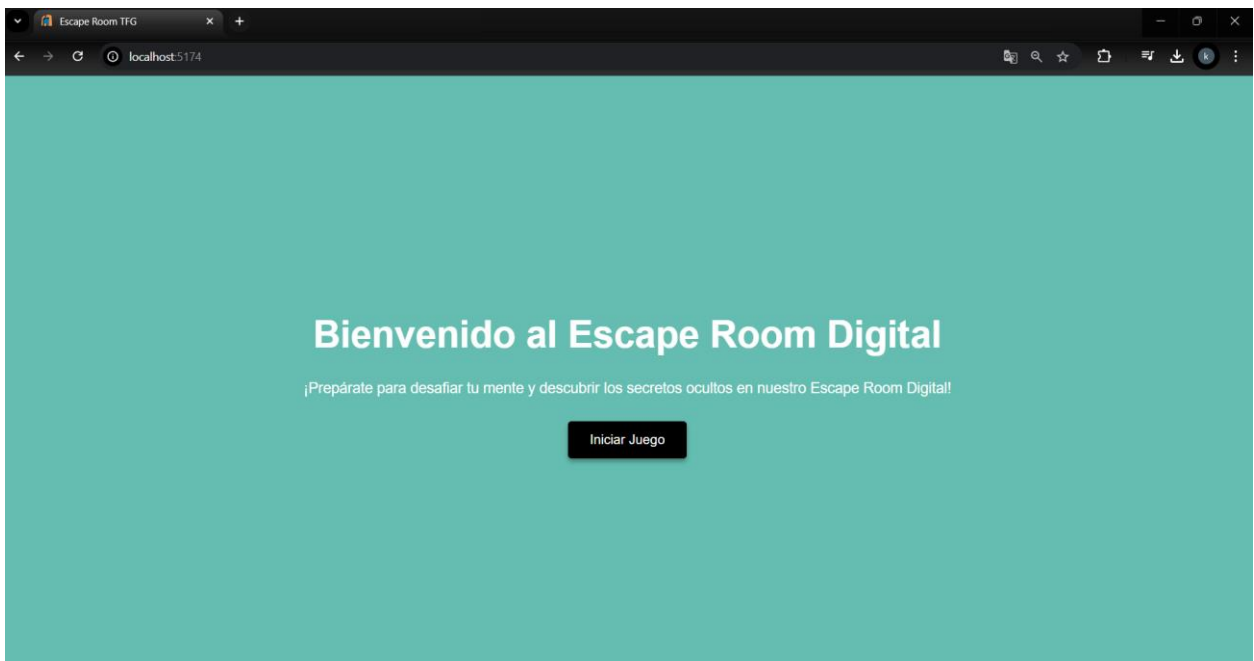


Figura 33 Prueba visualización Chrome

B.4 Prueba en Microsoft Edge

En Microsoft Edge, la Pantalla Home también se visualiza correctamente, manteniendo la estructura y diseño adecuados para una experiencia fluida.

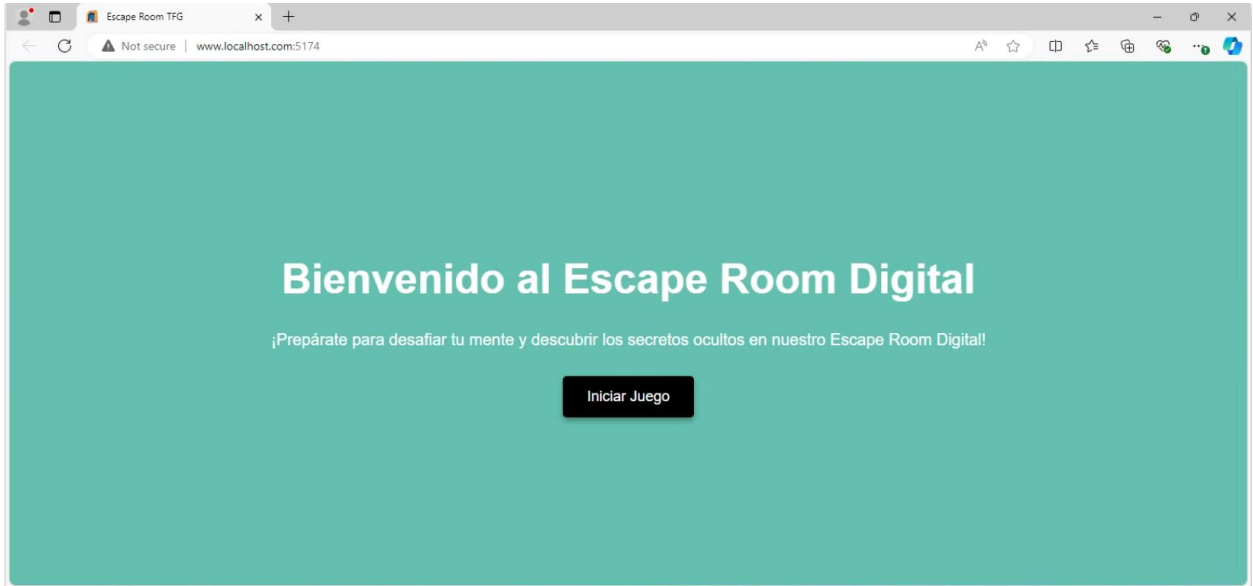


Figura 34 Prueba visualización Microsoft Edge

B.3 Prueba en Opera

Finalmente, se probó la visualización en el navegador Opera, donde la aplicación mostró el mismo nivel de calidad gráfica y funcionalidad que en los otros navegadores.

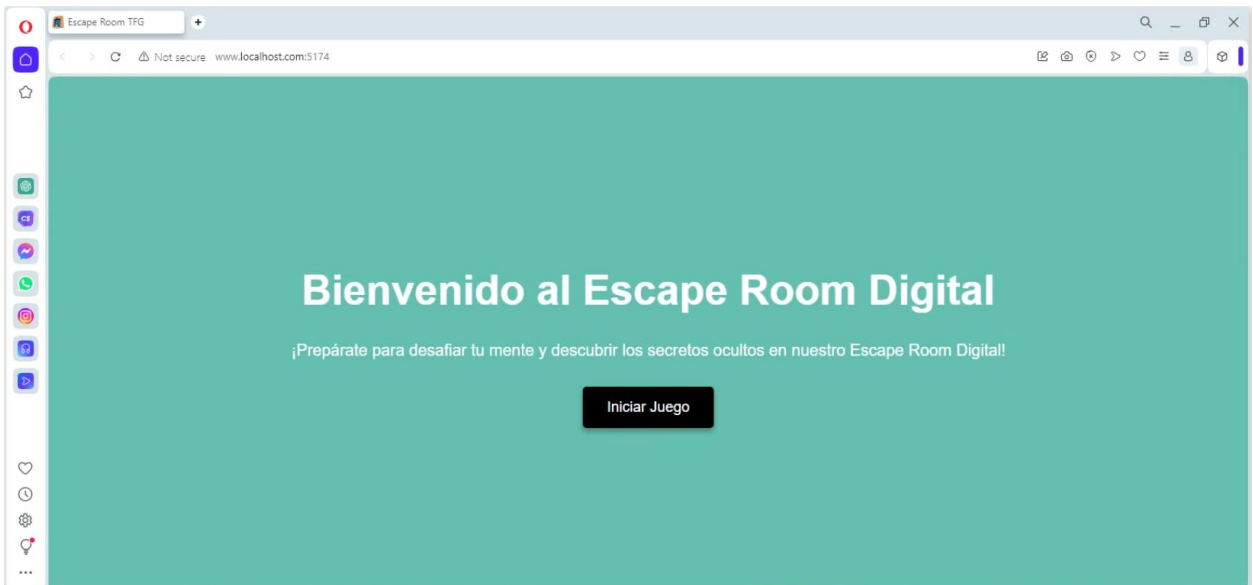


Figura 35 Prueba visualización Opera

Anexo C: Preguntas del Desafío 3

En este anexo se incluyen las preguntas del Desafío 3 de la aplicación "Escape Room Digital". Estas preguntas están diseñadas como un cuestionario de opción múltiple para evaluar el conocimiento de los usuarios sobre los procedimientos y normativas del Trabajo de Fin de Grado (TFG). Cada pregunta incluye varias opciones, indicando cuál es la respuesta correcta. Este contenido complementa la descripción del Desafío 3 en el documento principal, facilitando la consulta detallada de las interacciones evaluativas.

¿Qué debe hacer el alumno para entregar la memoria de su TFG?

- Enviar el TFG por correo electrónico al tutor.
- Depositar el TFG en la plataforma DEPOSITA y enviar la solicitud de defensa. *(Correcta)*
- Entregar el TFG impreso en la Secretaría de la Facultad.

¿Qué documentos debe aportar el tutor para el depósito?

- El plan de estudios del alumno.
- El documento Autorización del Director debidamente firmado. *(Correcta)*
- Las calificaciones del alumno en todas las asignaturas.

¿Qué es lo que tiene que hacer el tribunal?

- Proporcionar una lista de temas para el TFG.
- Organizar los horarios de las clases.
- Evaluar el TFG a partir de la memoria y la defensa y emitir una calificación. *(Correcta)*

¿Qué formato debe seguir la memoria del TFG?

- Cumplir con las normas de presentación establecidas en la normativa de la Escuela. *(Correcta)*
- Ser escrito a mano en folios.
- Ser presentado en un formato libre elegido por el alumno.

¿Qué debe hacer el alumno para la defensa del TFG?

- Solo asistir el día de la defensa sin preparación previa.
- Preparar una presentación para la defensa pública ante el tribunal evaluador. *(Correcta)*
- Enviar una grabación de video en lugar de asistir a la defensa.

¿Qué plazos debe respetar el alumno en el proceso de entrega del TFG?

- No hay plazos específicos, el alumno puede entregar cuando desee.
- Debe respetar el plazo de presentación de la propuesta y esperar a una de las fechas de depósitos públicas en la web de la EUPT. *(Correcta)*
- Puede entregar el TFG en cualquier momento antes de graduarse.

¿Cuáles de los siguientes documentos deben adjuntarse junto con la memoria del TFG según la diapositiva?

- Autorización del Director, Pantalla encuesta egresados, Documento de resguardo.
- Correo de confirmación del repositorio, Pantalla encuesta egresados, Declaración de autoría, Documento de resguardo.
- Autorización del Director, Correo de confirmación del repositorio, Pantalla encuesta egresados, Declaración de autoría, Documento de resguardo. *(Correcta)*

Anexo D: Interacciones Desafío 4 – Presentación del TFG

En este anexo se presentan todas las interacciones del Desafío de Presentación de TFG en la aplicación "Escape Room Digital". Este desafío simula una defensa de TFG ante un tribunal, incluyendo preguntas interactivas, opciones de respuesta y sus respectivos efectos en el desarrollo del desafío. Además, se detalla el diálogo con los personajes, las sugerencias proporcionadas por el tribunal y el proceso de deliberación de la calificación. Este contenido complementa la descripción del Desafío en el documento principal, proporcionando una visión completa de la narrativa y las mecánicas.

Bienvenida del tribunal:

- El tribunal se presenta:
 - Profesor 1: "¡Hola, bienvenido a la presentación de tu TFG!"
 - Profesor 2: "Nos aseguraremos de evaluar cada detalle de tu trabajo."
 - Presidente del Tribunal: "Soy el Profesor 3 y actuaré como presidente del tribunal. Dirigiré esta sesión. ¿Estás listo para comenzar?"

Pregunta inicial sobre el tiempo de exposición:

- **Pregunta:** ¿Cuántos minutos puedes estar exponiendo tu TFG?
 - Opciones:
 - Menos de 30 minutos. (*Correcta*)
 - Respuesta del tribunal: "Correcto, el tiempo de exposición debe ser menos de 30 minutos. Puedes proceder con tu presentación."
 - 30 a 45 minutos.
 - Respuesta del tribunal: "Eso es incorrecto. La presentación no debe durar tanto. Volvamos a empezar."
 - Más de 45 minutos.
 - Respuesta del tribunal: "No, no puedes estar tanto tiempo exponiendo. Debemos volver a empezar."

Opción de acercarse al ordenador o salir:

- **Decisión:**
 - Acercarse al ordenador. (*Continúa la presentación.*)
 - Salir de la clase.
 - Respuesta: "Decidí salir de la clase y no hacer el TFG."
 - Profesor 1: "No, así no. Debes hacer la presentación de tu TFG."

Presentación del TFG:

- **Diálogos:**
 - Alumno: "Gracias, comenzaré con mi presentación de TFG."
 - **Proyección del PowerPoint y prototipos:**
 - "Aquí podemos ver los aspectos más importantes de mi proyecto..."
 - "A continuación, procederé a mostrar el prototipo que he desarrollado."

Preguntas del tribunal:

- **Pregunta del Profesor 1:**

- ¿Por qué elegiste esta metodología en lugar de otras alternativas, y cómo se relaciona con los objetivos de tu TFG?
 - Opciones:
 - "Elegí esta metodología porque se alinea mejor con los objetivos de investigación, y me permite recopilar datos más relevantes." *(Correcta, +2 puntos)*
 - "No tuve mucho tiempo para considerar otras metodologías, así que opté por esta." *(Incorrecta, -1 punto)*

- **Pregunta del Profesor 2:**

- ¿Podrías explicar cómo tu TFG contribuye al campo de estudio existente? ¿Qué novedad o aporte consideras que has hecho?
 - Opciones:
 - "Mi TFG introduce un nuevo enfoque para resolver un problema conocido, lo que podría mejorar la eficiencia de las soluciones actuales." *(Correcta, +2 puntos)*
 - "Creo que mi TFG es similar a otros trabajos, pero al menos comprueba algunas teorías." *(Incorrecta, -1 punto)*

- **Pregunta del Presidente del Tribunal:**

- Como presidente, quiero preguntarte sobre la relevancia práctica de tu trabajo. ¿Cómo crees que tu TFG podría ser aplicado en el mundo real?
 - Opciones:
 - "Mi TFG puede ser aplicado en la industria para optimizar procesos y reducir costes, beneficiando así a las empresas." *(Correcta, +2 puntos)*
 - "No estoy seguro de su aplicación práctica, es más bien un estudio teórico." *(Incorrecta, -1 punto)*

- **Pregunta del Alumno 2:**

- Tengo una pregunta sobre la presentación de resultados: ¿Por qué decidiste presentar los resultados de esta manera y cómo crees que esto ayuda a la comprensión del lector?
 - Opciones:
 - "Elegí este formato porque es claro y facilita la comprensión de los datos, lo que es fundamental para la toma de decisiones." *(Correcta, +2 puntos)*
 - "No le di mucha importancia a cómo presentar los resultados, solo quería mostrar los datos." *(Incorrecta, -1 punto)*

Sugerencia del tribunal:

- **Diálogo fijo:**
 - Presidente: "Creo que podrías considerar utilizar una arquitectura más modular para facilitar futuras actualizaciones. Esto permitirá que tu proyecto se adapte mejor a las necesidades cambiantes."
 - Alumno: "¡Gracias por la sugerencia! Lo tendré en cuenta para la futura mejora del proyecto."

Deliberación y calificación:

- **Cálculo de la calificación basada en la puntuación acumulada:**
 - **Sobresaliente:** Si la puntuación es mayor o igual a 7 (Máximo alcanzable:8).
 - Presidente: "Después de nuestra deliberación, hemos decidido darte una calificación de Sobresaliente. ¡Felicidades!"
 - Alumno: "¡Muchas gracias por su tiempo y consideración!"
 - **Notable:** Si la puntuación está entre 5 y 6.
 - Presidente: "Después de nuestra deliberación, hemos decidido darte una calificación de Notable. Buen trabajo."
 - Alumno: "Gracias, trabajaré en las sugerencias para mejorar."
 - **Aprobado:** Si la puntuación es menor que 5.
 - Presidente: "Después de nuestra deliberación, hemos decidido darte una calificación de Aprobado. Hay áreas que requieren mejoras."
 - Alumno: "Agradezco su evaluación y trabajaré para mejorar."

Nota: El sistema está diseñado con un puntaje máximo de 8 puntos, no 10. Esto garantiza que las calificaciones se distribuyan dentro de los rangos establecidos, en lugar de basarse en una escala absoluta.

Anexo E: Identidad Corporativa

En este anexo se describe la identidad corporativa aplicada en la aplicación "Escape Room Digital". Este apartado incluye los elementos visuales y estilísticos clave que aseguran una apariencia coherente, profesional y alineada con los objetivos académicos del proyecto.

E.1 Diseño del logo

El logo utilizado en la aplicación es el de la Escuela Universitaria Politécnica de Teruel, el cual ha sido elegido para representar la conexión académica del proyecto. Este logo consiste en un diseño que utiliza una combinación de colores cálidos y confiables, reflejando tanto innovación tecnológica como un vínculo con la educación.



Figura 36: Logo de la aplicación "Escape Room"

E.2 Colores de la aplicación

La aplicación hace uso de una paleta de colores cuidadosamente seleccionada para proporcionar una experiencia de usuario coherente y atractiva. Los colores principales incluyen tonos de púrpura, verde agua, gris y blanco, que se utilizan estratégicamente en distintos elementos de la interfaz para mejorar la estética y la funcionalidad:

Púrpura Oscuro (#342956): Utilizado como color principal para los elementos clave de la interfaz, como fondos y títulos.



Figura 37 Púrpura Oscuro (#342956)

Púrpura Claro (#6e50b4): Empleado en botones y elementos interactivos para proporcionar un contraste visible y facilitar la interacción del usuario.



Figura 38 Púrpura Claro (#6e50b4)

Verde Agua (#64bdb0): Este color se utiliza como fondo general de la aplicación, aportando frescura y serenidad, lo cual es fundamental para mantener una experiencia de usuario relajante y agradable.



Figura 39 Verde Agua (#64bdb0)

Gris Oscuro (#2c2c2c): Utilizado en contenedores de texto, como en la clase .explanation, para reducir la dureza del negro puro y proporcionar un fondo suave que no fatigue la vista.



Figura 40 Gris Oscuro (#2c2c2c)

Blanco (#ffffff) y Gris Claro (#f0f0f0, #): Usados en textos y botones para asegurar una buena legibilidad y contraste. El blanco se utiliza para los textos principales, mientras que los tonos de gris claro se emplean para suavizar el contraste en títulos y párrafos.

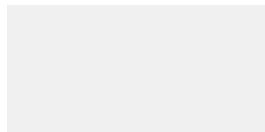


Figura 42 Blanco (#ffffff)

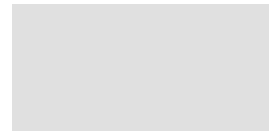


Figura 41 Gris Claro (#e0e0e0)

Figura 43 Gris Claro (#f0f0f0)

Gris Oscuro para Texto de Botón (#333): Usado en botones con fondos claros para garantizar que el texto sea fácilmente legible y contraste adecuadamente.



Figura 44 Gris Oscuro (#333)

E.3 Tipografía

Para mantener una apariencia consistente y profesional, la aplicación utiliza la fuente Lexend Deca para todos los textos, garantizando claridad y legibilidad.

Anexo F: Representación XML del Esquema de datos

```
<Room>
  <token>unique_token_string</token>
  <maxPlayers>4</maxPlayers>
  <state>open</state>
  <players>
    <Player>
      <id>player1_id</id>
      <persistentId>unique_persistent_id</persistentId>
      <name>Player One</name>
      <isReady>false</isReady>
      <totalTime>120</totalTime>
      <challengeTimes>
        <ChallengeTime>
          <challengeName>Challenge 1</challengeName>
          <timeTaken>60</timeTaken>
          <startTime>2024-11-12T08:30:00Z</startTime>
          <completedAt>2024-11-12T08:31:00Z</completedAt>
        </ChallengeTime>
        <ChallengeTime>
          <challengeName>Challenge 2</challengeName>
          <timeTaken>0</timeTaken>
          <startTime>2024-11-12T08:45:00Z</startTime>
        </ChallengeTime>
      </challengeTimes>
    </Player>
    <Player>
      <id>player2_id</id>
      <persistentId>unique_persistent_id_2</persistentId>
      <name>Player Two</name>
      <isReady>true</isReady>
      <totalTime>150</totalTime>
      <challengeTimes>
        <ChallengeTime>
          <challengeName>Challenge 1</challengeName>
          <timeTaken>70</timeTaken>
          <startTime>2024-11-12T08:32:00Z</startTime>
          <completedAt>2024-11-12T08:33:10Z</completedAt>
        </ChallengeTime>
      </challengeTimes>
    </Player>
  </players>
</Room>

<Leaderboard>
  <roomCode>unique_token_string</roomCode>
  <createdAt>2024-11-12T08:00:00Z</createdAt>
  <players>
    <Player>
      <name>Player One</name>
      <totalTime>120</totalTime>
      <challengeTimes>
        <ChallengeTime>
          <challengeName>Challenge 1</challengeName>
          <timeTaken>60</timeTaken>
          <startTime>2024-11-12T08:30:00Z</startTime>
          <completedAt>2024-11-12T08:31:00Z</completedAt>
        </ChallengeTime>
      </challengeTimes>
    </Player>
  </players>
</Leaderboard>
```

<Room>: Representa una sala de juego. Contiene información sobre el número máximo de jugadores (`<maxPlayers>`), el estado de la sala (`<state>`), y una lista de jugadores (`<players>`).

<Player> dentro de <Room>: Cada jugador tiene un `id`, `persistentId`, `name`, `isReady` y `totalTime`. Los `<challengeTimes>` de cada jugador están anidados, permitiendo ver cada desafío y su respectivo tiempo.

<ChallengeTime>: Incluye `challengeName`, `timeTaken`, `startTime`, y `completedAt` para cada desafío realizado.

<Leaderboard>: Se refiere a la tabla de clasificación, vinculada a una sala mediante `roomCode`. Muestra una lista de jugadores (`<players>`) con sus `challengeTimes`.