



**Escuela Superior
de Ingeniería y Tecnología**
Universidad de La Laguna

Trabajo de Fin de Grado

**Aplicación de Juego basada en Balderdash
con integración de IA**

Game Application based on Balderdash with AI integration

Óscar Ignacio Pozo Fernández

La Laguna, 23 de mayo de 2024

D. **Francisco Javier Rodríguez González**, profesor Asociado adscrito al Departamento Ingeniería Informática y de Sistemas. Área: Ciencia de la Computación e Inteligencia Artificial de la Universidad de La Laguna, como tutor.

D. **Alejandro Pérez Navas**, profesor Asociado adscrito al Departamento Ingeniería Informática y de Sistemas. Área: Ciencia de la Computación e Inteligencia Artificial de la Universidad de La Laguna, como cotutor.

C E R T I F I C A (N)

Que la presente memoria titulada:

"Aplicación de Juego basada en Balderdash con integración de IA"

ha sido realizada bajo su dirección por D. **Óscar Ignacio Pozo Fernández**.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 23 de mayo de 2024

Agradecimientos

Les doy las gracias principalmente a mi familia por apoyarme todos estos años, dándome las energías para mantenerme constante con las cosas importante. A mis amigos por su apoyo incondicional e interés en mi pequeño proyecto.

Licencia



© Esta obra está bajo una licencia de Creative Commons Reconocimiento-
NoComercial-CompartirIgual 4.0 Internacional.

Resumen

Los juegos de cartas, de mesa y familiares son formatos de ocio que llevan existiendo muchos años en todos los núcleos sociales y familiares.

Antes la única manera de obtener estos juegos era en una tienda física. Hoy en día, aunque este modelo sigue presente, muchos de estos juegos proponen adaptarse al mercado digital para seguir siendo una opción de ocio para las nuevas generaciones.

El prototipo desarrollado está inspirado en el juego de mesa 'Balderdash' [5], el cual es un juego de 3 a 6 jugadores donde los participantes ganan puntos inventándose definiciones de palabras, iniciales, películas, etc, que es muy probable que no conozcan. Estas invenciones se mezclan con el significado real de dicha cuestión y los jugadores votan la que creen que es verdad. Los jugadores que aciertan reciben un punto, y también pueden recibir puntos si otros jugadores votan su definición inventada.

En este proyecto se realiza un estudio y prototipo de un videojuego de adivinanzas para dispositivos Android y navegador web. El objetivo es permitir a grupos de amigos el jugar partidas rápidas donde se premia el conocimiento y el uso astuto de la palabra.

Para evaluar la viabilidad económica del proyecto, se ha realizado un estudio de viabilidad económica del proyecto, el cual contempla el coste y el tiempo de desarrollo del proyecto final, las posibles fuentes de ingreso y comercialización de la aplicación y un estudio del retorno de inversión (ROI).

Palabras clave: Juego de mesa, Juego de cartas, Multijugador, Party Game ¹

¹**Party Games:** juegos multijugador pensados para jugarse grupos de amigos

Abstract

In all social and family groups there's always been some games that can be played by the whole group: Board, Card and Family Games. Previously you could only get these games in physical stores, but now many of these games are trying to adapt to the digital market, offering versions for PC and smartphone, trying to appeal to new audiences.

The developed prototype is heavily inspired in the board game "Balderdash", which is a game for 3 to 6 players where each one must come up with a made up definition for a word, initials, films, etc, that is very rare or unknown. These made up definitions are shown later along with the real definition and each player must vote for the one they believe it's true. Players that guessed it right receive a point, and each vote for any other definition is added to the score of the respective player.

This document will provide a study and explanation of the developed prototype of this guessing game for Android devices and web browsers. The final goal is to provide these groups of players with a game about knowledge, cunning and the smart use of words. Furthermore, to evaluate the project sustainability there's also an economic viability plan that discusses the money and time costs of the development, plausible sources of income, commercialization and Return of Investment (ROI).

Keywords: Card Game, Board Game, Multiplayer, Party Game.

Índice general

1. Introducción	1
1.1. Definición del problema	1
1.2. Justificación	2
1.3. Objetivo	3
1.4. Juegos similares	4
1.5. Estado actual del mercado	5
2. Estudio Previo	7
2.1. Frontend	7
2.1.1. Aplicación móvil	7
2.2. Backend	8
2.2.1. Servidor	8
2.2.2. Base de datos	9
2.3. Inteligencia Artificial	10
2.4. Lenguajes	11
2.5. Elección	12
2.6. Otras tecnologías	12
3. Desarrollo	14
3.1. Backend	14
3.1.1. API REST	14
3.1.2. Sockets	16
3.1.3. GameLoop	16
3.2. Creación base de datos MongoDB	17
3.3. Frontend	17
3.3.1. Componentes	18
3.3.2. Context	18
3.3.3. Navigator	18
3.3.4. Vistas	18
3.4. Implementación de IA	24
4. Estudio de viabilidad económica del proyecto	25
4.1. Modelo de comercialización	25
4.1.1. Fuentes de ingresos	25
4.1.2. Elección de modelo de comercialización	26
4.2. Desarrollo del proyecto	27
4.2.1. Análisis	27

4.2.2. Diseño	27
4.2.3. Desarrollo	27
4.2.4. Fase de Pruebas	28
4.2.5. Despliegue	28
4.3. Estudio del Retorno de inversión	28
5. Conclusiones y líneas futuras	37
6. Summary and Conclusions	38
7. Presupuesto	39

Índice de Figuras

1.1. Logotipo de Balderdash	1
1.2. Logotipo de Cartas contra la Humanidad	1
1.3. Crecimiento estimado del mercado de juegos de mesa global. Fuente: Words Rated.[43]	2
1.4. Ingresos del mercado móvil en España. Fuente: Statista.[38]	2
1.5. Interés a nivel global de 'Chatbot'.	3
1.6. Juegos similares	4
1.7. Interés desde 2021 hasta hoy sobre Wolvesville en todo el mundo	5
1.8. Interés a lo largo de 5 años sobre Pinturillo 2	5
1.9. Interés desde 2020 hasta la actualidad sobre Gartic Phone	5
1.10 Interés a lo largo del último año sobre Make it Meme	6
1.11 Interés a lo largo de 5 años sobre Juegos Multijugador	6
2.1. Propuestas para la aplicación de Android	8
2.2. Logotipos Stack MERN	9
2.3. Propuestas para la Base de Datos	9
2.4. Propuestas para la IA	10
2.5. Propuestas de Lenguajes de programación	11
2.6. Otras tecnologías utilizadas	13
3.1. Diagrama Bucle de Juego.	17
3.2. Pantalla del Menú Principal.	19
3.3. Pantalla con inicio como user anónimo.	19
3.4. Pantalla del registro.	20
3.5. Pantalla del Lobby.	20
3.6. Pantalla de la Sesión de Juego.	21
3.7. Pantalla del Bucle de Juego. Recibe el término.	22
3.8. Pantalla Bucle de Juego. Selección de definición.	23
3.9. Pantalla Bucle de Juego. Se muestra al Ganador.	24
4.1. Tabla de planificación de la duración y costo del proyecto. ProjectLibre.	29
4.2. Diagrama de Gantt. Parte 1.	30
4.3. Diagrama de Gantt. Parte 2.	31
4.4. Tabla trabajadores.	32
4.5. Tabla usuarios y visitas.	34
4.6. Tabla cálculo de ingresos.	35
4.7. Gráfico de gastos e ingresos.	36

Capítulo 1

Introducción

1.1. Definición del problema

En la actualidad existe un género de juegos multijugador llamado 'Party Games', destinados a jugar en familia y con amigos. Estos juegos clásicos como el 'Pictionary'[34], 'la Mafia/Hombre lobo' [19] o 'el Telefonillo roto'[7] tienen actualmente versiones digitales para jugador en web y dispositivos móviles: 'Pinturillo'[2], 'Werewolf Online'[29] y 'Gartic Phone'[33].

Sin embargo, otro de los pilares de este tipo de juegos es 'Balderdash', o también conocido como 'Dictionary Game', juego de adivinar significados de palabras o términos. A pesar de que juegos como 'Cartas contra la Humanidad'[14] que se han inspirado en este y ha sido fenómeno mundial, todavía 'Balderdash' no ha tenido impacto en la escena digital de 'Party Games', menos aún en español.



Figura 1.1: Logotipo de Balderdash



Figura 1.2: Logotipo de Cartas contra la Humanidad

1.2. Justificación

Los 'Party Games' online han crecido en popularidad estos últimos años. Son jugados habitualmente por grupos de amigos de manera casual en todo el mundo.

Al ser un juego multijugador para todas las edades, el público del juego abarca desde niños y niñas que han comenzado a usar un dispositivo inteligente, como una tablet, hasta adultos que buscan una partida casual en la que desafiar sus conocimientos.

La tendencia de mercado, tanto de juegos móviles como de mesa, demuestra una curva ascendente a nivel mundial. Los siguiente gráficos revelan que estos dos sectores a los que pertenece, en mayor o menor medida, la aplicación siguen en constante crecimiento, tanto a nivel de ingresos como de número de jugadores.



Figura 1.3: Crecimiento estimado del mercado de juegos de mesa global. Fuente: Words Rated.[43]



Figura 1.4: Ingresos del mercado móvil en España. Fuente: Statista.[38]

A parte de los ya mencionados, los cuales son adaptaciones a digital de juegos clásicos, también han surgido nuevos juegos de esta misma categoría a buscar su hueco en el mercado con propuestas nuevas y originales.

Como ejemplo reciente tenemos 'Make it Meme'[20], que es otro multijugador online que invita a grupos de amigos a rellenar el texto de plantillas de memes, puntuando la creatividad y el humor. Una muestra que sigue existiendo mercado para nuevos 'Party Games'.

Todos estos juegos se centran en la comunicación y astucia de los jugadores, sin embargo, como 'Balderdash' está regido a un terreno estrictamente escrito, supone un terreno de pruebas ideal para comprobar la potencia de los Chatbots, intentando desdibujar la línea entre quién es un jugador real o una IA cuando únicamente la parte escrita es la esencial del juego.

Otra propósitos que pueden tener los Chatbots en esta aplicación es la posibilidad de entrenar a uno con las respuestas de todos los jugadores. Concretamente, esta IA se podría especializar en realizar explicaciones y definiciones que usen un lenguaje más informal.

Para justificar su uso, se ha comprobado el aumento en popularidad que han tenido las IAs, en general, durante los últimos años. Las grandes empresas como Google, Meta y Microsoft ya han lanzado y continúan desarrollando sus propios Chatbot, con mayor interés y mejores resultados que nunca, y el término en sí ya forma parte del conocimiento popular como una herramienta que ha aparecido para quedarse.

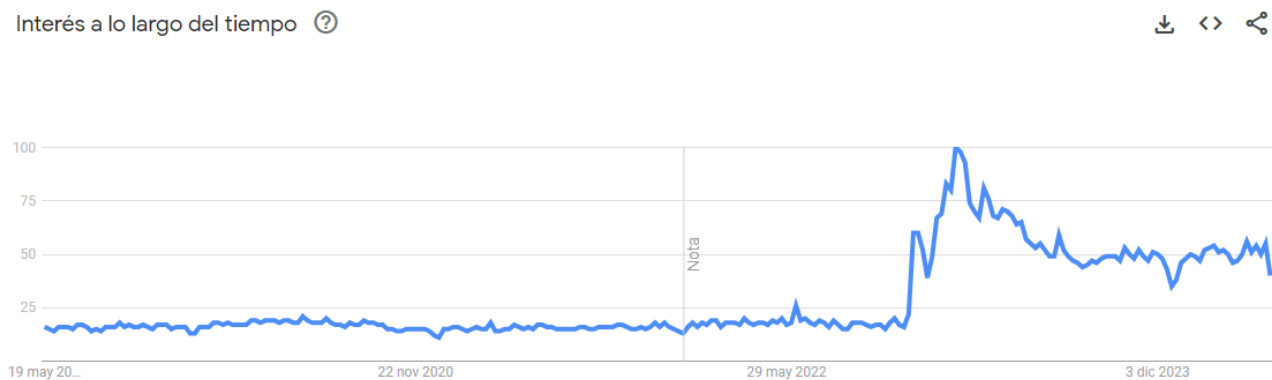


Figura 1.5: Interés a nivel global de 'Chatbot'.

1.3. Objetivo

El propósito de la aplicación es la de ofrecer a los jugadores web y móvil el juego 'Balderdash' en español e inglés para partidas multijugador con posibilidad de competir contra sus amigos y/o contra una o varias IAs.

La idea es facilitar el acceso al juego para que los grupos de amigos o jugadores individuales puedan comenzar una partida lo más rápido posible. Es por esto que, tanto en la versión web como en la app, tendrán la posibilidad de crear o unirse a una sala con un usuario temporal, con la posibilidad de crear una cuenta.

Esta intención de abarcar al mayor público posible es la que decanta el prototipo a tener principalmente una versión Android, donde más usuarios casuales hay, y otra para navegador web.

El nombre comercial de la aplicación, y el que se usará en este documento, es 'Disparates'.

1.4. Juegos similares

A continuación se presenta un listado de juegos de móvil y navegador web que tienen un mismo tipo de jugadores: jugadores casuales que buscan pasar un buen rato con sus amigos y/o familia.

- **Balderdash.**[5] Aunque existe 'Balderdash' como juego de navegador web, solo está disponible para jugar en web y en Inglés. No existe una aplicación decente de 'Balderdash' para dispositivos móviles, ni tampoco versión móvil ni web en español.
- **Wolvesville - Werewolf Online.**[29] Juego multijugador de hasta 16 jugadores, para web y móvil. 'Party Game' donde los jugadores tienen roles secretos, pudiendo pertenecer al bando del pueblo o de los lobos. Gana el bando que consiga acabar con el otro.
- **Pinturillo 2.**[2] Juego multijugador para web y móvil. Consiste en adivinar una palabra a través del dibujo realizado por alguno de los jugadores. Su modelo y despliegue es el más parecido al objetivo final de este proyecto.
- **Gartic Phone.**[33] Juego multijugador principalmente para web. Juego del telefonillo roto pero con frases y dibujos en vez de transmitir una frase usando el habla.
- **Make It Meme.**[20] Última incorporación a este tipo de juegos. Juego en tono humorístico que reta a sus jugadores a crear memes introduciendo el texto en una plantilla.



(a) Logo Wolvesville



(b) Logo Pinturillo 2



(c) Logo Gartic Phone



(d) Logo Make it Meme

Figura 1.6: Juegos similares

1.5. Estado actual del mercado

Las diferentes tiendas digitales para dispositivos móviles tienen mucha variedad en cuanto a juegos se refiere. Sin embargo, juegos multijugador, casuales, rápidos y para todas las edades son más escasos. Respecto a juegos web que no requieran descarga ni instalación, tenemos también una cierta variedad que se trata en Juegos Similares (1.5).

A través de Google Trends[41] podemos comprobar el interés a lo largo del tiempo que tienen varios términos de búsqueda que puedan resultar relevantes.

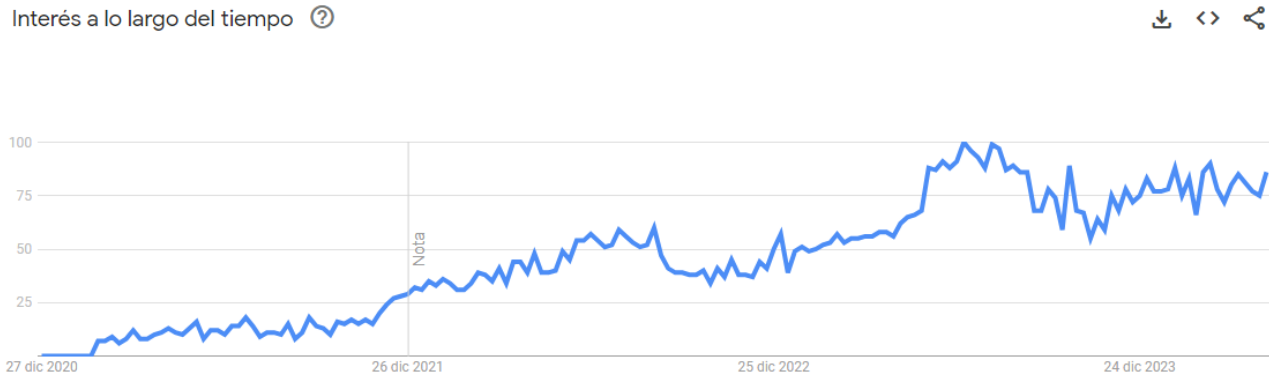


Figura 1.7: Interés desde 2021 hasta hoy sobre Wolvesville en todo el mundo



Figura 1.8: Interés a lo largo de 5 años sobre Pinturillo 2



Figura 1.9: Interés desde 2020 hasta la actualidad sobre Gartic Phone



Figura 1.10: Interés a lo largo del último año sobre Make it Meme

Otro indicador de mercado son las descargas en la Play Store. Para este caso, tomamos de referencia algunos de los Juegos Similares:

■ **Wolvesville**

- **Wolvesville - Werewolf Online**[30]: más de 10 millones de descargas. Es la versión gratuita con publicidad.
- **Wolvesville Classic**[6]: más de 1 millón de descargas. Versión para jugar en local con publicidad.
- **Wolvesville Classic Pro**[35]: más de 5 mil descargas. Versión local sin publicidad y contenido adicional.

■ **Pinturillo 2**[39]: más de 1 millón de descargas.

■ **Gartic.io**[9]: más de 10 millones de descargas. Copia de Pinturillo.

■ **Evil Apples: Funny as ******[1]: más de 5 millones de descargas. Copia de 'Cartas contra la Humanidad'.

Que estos juegos tengan un flujo de jugadores continuo junto a picos de popularidad, nos indica también que '*Disparates*' puede compartir mercado y clientes con los juegos mencionados.



Figura 1.11: Interés a lo largo de 5 años sobre Juegos Multijugador

Capítulo 2

Estudio Previo

En este apartado se exponen y explican las diferentes propuestas de tecnologías y herramientas tomadas en consideración al inicio del proyecto. Al final del apartado también se mostrarán las propuestas seleccionadas para el desarrollo del proyecto.

2.1. Frontend

El 'Front' representa toda la construcción de la interfaz y experiencias de usuario. Todos los elementos gráficos que forman la aplicación con la que el usuario va a interactuar para su uso forman parte del 'Frontend'.

Para este proyecto se ha priorizado la versión 'Frontend' para dispositivos móviles Android, que generará una APK¹ del prototipo desarrollado. Algunos motivos para esta decisión son: que el desarrollo móvil supone un nuevo formato y plataforma de desarrollo para el autor, el mercado móvil es en el que más cantidad de jugadores potenciales hay y donde se puede prever mayor éxito comercial.

La versión web está propuesta para un desarrollo profesional del proyecto.

2.1.1. Aplicación móvil

Para el desarrollo móvil del proyecto se valoraron únicamente dos opciones:

- **Android Studio**[40]. Es el entorno de desarrollo oficial para Android. Tiene un entorno de desarrollo integrado (IDE²) para la creación de Apps. Como lenguaje de programación tiene dos opciones: Kotlin y Java.

Permite crear diseños dinámicos y animaciones con plantillas. Tiene un editor de texto inteligente y previsualización de diseños. También incluye un emulador de multitud de dispositivos Android: teléfonos, tablets, TVs, etc.

- **React Native**[26]. Es un framework³ que usa Javascript⁴ para crear código nativo tanto para Android como iOS⁵. Esta particularidad permite usar un mismo código

¹**APK**: es un formato de archivo 'Android Application Package' ó 'Paquete de Aplicación Android' específico para Android con el que se puede instalar todo el código de un programa.

²**IDE**: es una aplicación de software que ayuda a los programadores a desarrollar código de software de manera eficiente, pudiendo crear, editar, probar y empaquetar software en una sola aplicación.

³**Framework**: conjunto de herramientas, guías y estructuras predefinidas que se utilizan para desarrollar y organizar software de manera eficiente.

⁴**Javascript**: lenguaje de programación muy centrado en el desarrollo web.

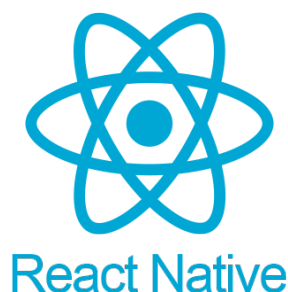
⁵**iOS**: sistema operativo de los dispositivos portátiles de Apple.

para desarrollar el grueso de una aplicación móvil y poder diferenciar dentro del código las particularidades para cada uno de estos dos sistemas operativos.

También nos permite visualizar los cambios de una rápida. Tiene integración con React[37], por lo cual tiene acceso a la gran cantidad de recursos que tiene esta biblioteca para interfaces web.



(a) Logo Android Studio



(b) Logo React Native

Figura 2.1: Propuestas para la aplicación de Android

2.2. Backend

El Backend de la aplicación es todo el código e infraestructura que almacena, procesa y transforma los datos para el correcto funcionamiento de la aplicación. Es el encargado de recibir los datos y peticiones del Frontend y devolver una respuesta.

2.2.1. Servidor

Para el servidor de la aplicación ya se tenían seleccionadas una serie de tecnologías desde el principio por conocimientos previos y gusto personal.

- **Stack MERN.** Es un conjunto de tecnologías utilizados para el desarrollo de aplicaciones web. Es un paquete completo para este propósito, debido a que cada una de las tecnologías se encarga de una parte esencial de una página web completa: base de datos, desarrollo y conexión web, frontend y backend.
 - **MongoDB.**[24] Es la base de datos NoSQL⁶ más popular. Almacena la información en colecciones, pudiendo crear subestructuras y esquemas más flexibles y similares a un JSON⁷.
 - **ExpressJS.**[8] Es una infraestructura de aplicaciones web Node.js mínima y flexible que proporciona un conjunto sólido de características para las aplicaciones web y móviles.
 - **React.**[37] Es una biblioteca de interfaces de usuario web y nativas para el desarrollo Front. Trabaja con piezas individuales de código modular llamadas

⁶**NoSQL:** tipo de base de datos no relacional, es decir, que no usa tablas y ni sus estructuras basada en reglas. En cambio, suelen ser más flexibles.

⁷**JSON:** es un formato de texto sencillo para el intercambio de datos, que significa 'JavaScript Object Notation'.

'Componentes' para crear las pantallas. También permite integrar código JavaScript para añadir condicionales e interactividad a la aplicación. Está mantenido por Facebook/Meta.

- **NodeJS**. [27] Es un entorno de ejecución para JavaScript que puede permitirle ejecutar JavaScript del lado servidor y no en un navegador. Utiliza módulos para instalar en el proyecto las bibliotecas de recursos específicos para las necesidades del desarrollo.

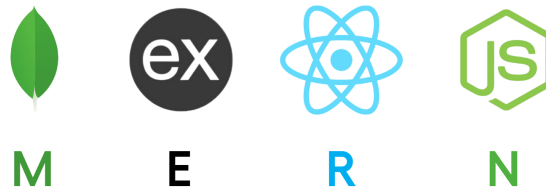


Figura 2.2: Logotipos Stack MERN

2.2.2. Base de datos

Se sopesaron las siguientes opciones:

- **MySQL**. [25] Es el sistema de gestión de datos relacional más popular del mundo. Nos permite realizar consultas rápidas a la base de datos, la cual tiene estructura de tablas con reglas que especifican el tipado de cada dato almacenado. Sus filas y columnas permiten almacenar, actualizar, eliminar, buscar y recuperar información de la base de datos.
- **MongoDB**. [24] Es un sistema de base de datos que no usa SQL. En su lugar, almacena la información en un formato similar a JSON, el cual permite estructuras de datos flexibles y personalizables.



(a) Logo MySQL



(b) Logo MongoDB

Figura 2.3: Propuestas para la Base de Datos

2.3. Inteligencia Artificial

La integración de la sección de la Inteligencia Artificial (IA) es quizás la parte más curiosa del proyecto. Para el desarrollo de este apartado, se expondrán los diferentes tipos de ChatBots⁸ que se tuvieron en cuenta.

La función principal de esta IA es la de comportarse como un participante humano en el juego, es decir, inventar definiciones de términos y participar en la votación. El objetivo final es que los demás jugadores no se den cuenta de que están jugando con una Inteligencia Artificial.

Las tres opciones sopesadas fueron las siguientes:

- **API de ChatGPT.**[31] El ChatBot más popular del momento. Está desarrollado por OpenAI. Tiene una API que nos permite integrar su funcionalidad en el prototipo, sin embargo, es de pago por la cantidad de tokens⁹ que se usen.
- **OLlama 2.**[21] ChatBot de Facebook/Meta. Concretamente se seleccionó la versión 7B, que requiere de 3.8Gb de procesamiento, cantidad inferior a la mayoría de sus otras versiones. Su API es gratuita y nos permite crear clones gratuitos de los diferentes modelos y entrenarlos.
- **Fireworks IA.**[3] Es una plataforma que ofrece el uso de APIs de IAs generativas como un servicio en la nube y bajo demanda, en el que puedes importar ese modelo en tu máquina. Además de los Chatbots, también ofrece el uso de otro tipo de IAs generativas más específicas, como las que generan imágenes, resumen textos o asisten en el desarrollo de código. Su cobro es por cantidad de tokens usados, sin embargo, tiene una cantidad de uso gratuito.



(a) Logo ChatGPT



(b) Logo oLlama



(c) Logo Fireworks AI

Figura 2.4: Propuestas para la IA

⁸**ChatBot:** Bot conversacional que usa inteligencia artificial y procesamiento del lenguaje natural para comprender las preguntas de los clientes y automatizar las respuestas, simulando la conversación humana.

⁹**Token:** trozo de una frase a transmitir. Su objetivo es medir la cantidad de texto de cada mensaje.

2.4. Lenguajes

Teniendo en cuenta las diferentes tecnologías, los lenguajes de programación que fueron valorados son los siguientes:

- **Javascript.**[16] Es el lenguaje de programación usado por todos los navegadores web modernos. Es el único lenguaje que entiende de forma nativa los navegadores. Esto le permite actualizar información de manera dinámica, realizar animaciones y mejorar la interfaz y experiencia de usuario general.
- **Typescript.**[42] Es un lenguaje de programación que puede ejecutar código Javascript, con el objetivo de resolver muchos de los problemas que tiene en proyectos grandes. Añade tipado, es decir, fuerza a que las variables sean y reciban datos del tipo específico asignado para funcionar. También incluye una potente herramienta de autocompletado para agilizar la escritura de código y detectar mejor los errores.
- **Java.**[15] Es un lenguaje de programación muy utilizado para la codificación de páginas web. Tiene una larga trayectoria, lo cual le ha permitido adaptarse a muchos campos especializados, como IA, desarrollo de videojuegos, computación en la nube, etc. La flexibilidad y cantidad de recursos que tiene lo convierten en un lenguaje confiable y versátil.
- **Kotlin.**[18] Es un lenguaje de programación específico para desarrollar aplicaciones en sistemas operativos Android. Es más moderno y simple que Java, aunque tiene interoperabilidad, es decir, puede interactuar con código escrito en ese lenguaje. También resuelve algunos problemas de su lenguaje padre, como la sintaxis, la complejidad del lenguaje y mejor control de excepciones y tipado.



(a) Logo Javascript



(b) Logo Typescript



(c) Logo Java



(d) Logo Kotlin

Figura 2.5: Propuestas de Lenguajes de programación

2.5. Elección

La selección final de tecnologías fue la siguiente:

■ Frontend

- **React Native.** La posibilidad de utilizar Javascript como lenguaje permite usar los mismos recursos y mantener consistencia con el backend. Además es el lenguaje con el que tengo más experiencias y familiaridad.

■ Backend. Stack MERN.

- **ExpressJS.** Estructura para la creación de routers¹⁰ que se comunican con las aplicaciones web y móviles.
- **React.** A pesar de que pertenece al paquete, concretamente se usa su versión 'Native'¹¹ para el desarrollo 'Frontend' en Android: 'React Native'.
- **NodeJS.** Para el desarrollo de toda la lógica y manejo de peticiones del servidor de la aplicación.

■ Base de Datos

- **MongoDB.** Debido a la elección del Stack MERN, también es consecuente el uso de MongoDB para almacenar la información. Su formato de almacenaje BSON¹² es complementaria con el formato JSON tan utilizado en Javascript.

■ Inteligencia Artificial

- **Fireworks AI.** Un problema que se logró anticipar fue la limitación de procesamiento del servidor en el despliegue. Originalmente, se iba a utilizar un modelo entrenado de Ollama2 versión 7B, debido a que es el que menos potencia requiere. Sin embargo, su necesidad de procesamiento sigue siendo superior a la ofrecida por cualquier plataforma de hosting gratuito. Es por esto que se decidió buscar una alternativa serverless¹³. Otro punto a favor es la posibilidad de usar cualquiera de las APIs de los modelos de forma gratuita si no se llega a superar un límite de tokens.

2.6. Otras tecnologías

- **Git.**[10] Es un sistema de control de versiones distribuido, lo que significa que se crea un clon local del proyecto como un repositorio de control de versiones completo.
- **Github.**[11] Es una plataforma de alojamiento de código y control de versiones Git en la nube. Los repositorios son el lugar donde almacenar el código desarrollado, pudiendo colaborar con otros miembros en la aportación de nuevo código.

¹⁰**Router:** En Express un router es manejador de rutas montables y modulares.

¹¹**Native:** Término para referirse a la versión de dispositivo móvil de una herramienta de desarrollo.

¹²**BSON:** conocido también como 'Binary JSON', es un formato de intercambio de datos usado principalmente para su almacenamiento y transferencia en la base de datos MongoDB.

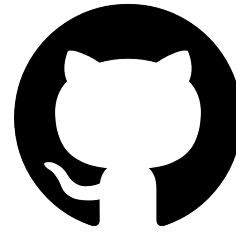
¹³**Serverless:** 'sin servidor' en español, indica que todos los modelos están alojados en Fireworks y todo el procesamiento se realiza en sus servidores.

- **Overleaf**. [32] Aplicación para generar la memoria del proyecto. Es una herramienta online que usa 'Latex', el cual es un sistema de creador de tipados pensado para documentos técnicos y científicos. Permite formatear el contenido en tablas, figuras, expresiones matemáticas, imágenes, bibliografías, citas, etc.



git

(a) Logo Git



(b) Logo Github



(c) Logo Overleaf

Figura 2.6: Otras tecnologías utilizadas

Capítulo 3

Desarrollo

El desarrollo del *'Disparates'* se dividió en 4 fases, cada una conteniendo una parte importante y distintiva del mismo: desarrollo Backend, desarrollo Frontend, implementación de IA y despliegue.

3.1. Backend

El desarrollo Backend consiste en la definición e implementación de una API REST¹ para el manejo de peticiones de la aplicación, el uso de Sockets² para un intercambio de información más rápido entre los jugadores y servidor, el Bucle de juego (GameLoop³) con sus normas y restricciones, y la conexión con la Base de Datos.

La aplicación móvil y la versión web usan este mismo Backend para permitir a jugadores de las dos principales plataformas de entretenimiento, a nivel de números, el poder jugar una partida.

3.1.1. API REST

- **Models:** Contiene los archivos que definen la estructura de los modelos de datos de la aplicación. Usando Mongoose podemos definir estos objetos usando su clase *'Schema'*, que nos permite especificar tipos, validaciones, valores, etc, para cada uno de nuestros modelos. Estos modelos generarán los objetos finales usados por la aplicación y que estarán almacenados en la Base de Datos de MongoDB. Los modelos implementados son los siguientes:
 - **Games:** Es la estructura de datos para las partidas realizadas en el juego. Contiene los siguientes parámetros:
 - Nombre sesión
 - Creador
 - Array de Participantes
 - Array de Puntuaciones

¹**API REST:** mecanismos que permite a dos componentes de software el poder comunicarse. En este caso, usando una transferencia de estado representacional

²**Socket:** canal de comunicación bidireccional que permite a dos procesos el intercambiar información rápidamente.

³**GameLoop:** término para referirse a la serie de acciones que se irán repitiendo a lo largo del juego y/o las partidas

- Número de rondas
- Contraseña de sesión
- Estatus de la partida
- Ganador

Su complejidad reside en permitir de manera dinámica un número flexible de jugadores. Esto se consigue usando un array de tipo que almacena el ID de los participantes. Los demás parámetros son los mínimos indispensables para poder controlar el flujo de una partida. A parte de los datos usuales, el *Estatus* indica si esta partida debe aparecer en el listado pública o no.

- **Players:** Modelo sencillo que representa la información de cada cuenta de usuario registrado en la aplicación.
 - Nombre de usuario (único)
 - Correo electrónico (único)
 - Contraseña encriptada
- **Terms:** Representa los términos usados para el juego, cuyo significado se debe inventar y luego adivinar. Los parámetros son los siguientes:
 - Nombre del término
 - Tipo de término
 - Significado

Actualmente el juego solo contempla usar palabras raras del diccionario como términos, pero se plantea ampliar el tipado a fechas, personajes desconocidos, siglas, libros y películas poco conocidas, etc.

- **Routes:** para las operaciones CRUD (Create, Read, Update, Delete) se han generado una serie de rutas para cada uno de los modelos, separadas en un fichero por cada modelo.
 - **Player**
 - /login: inicio de sesión en la aplicación
 - /register: registro de usuario en la aplicación, comprobando que tanto nombre de usuario como correo sean únicos.
 - /player: petición 'get' para la búsqueda específica de un usuario.
 - **Games**
 - /partidas: petición 'post' crea nuevas partidas (sesiones de juego) y petición 'get' recoge todas las partidas activas públicas a las que un jugador puede unirse.
 - /updateRounds: petición 'patch' para actualizar el número de rondas de la partida.
 - /updatePlayers: petición 'patch' que actualiza la lista de jugadores participantes cuando estos se unen a la sesión.
 - **/terms:** manejo de las llamadas 'get' y 'post'. Devuelve la cantidad de términos aleatorios que se vayan a usar en esa partida. Permite añadir un nuevo término a la colección.

- **Middleware:** Paso intermedio de comprobación de identidad a través de un token. Restringe el acceso a las rutas más sensibles: las que contengan información privada de la partida o usuarios, y/o que modifiquen o participen en el flujo de juego.
- **Config:** Resto de información sensible: secreto de encriptado.

3.1.2. Sockets

Para la implementación relacionada a la partida, se decidió por emplear los Sockets, puesto establecemos una conexión fija y bidireccional que nos permite intercambiar información rápidamente entre cliente y servidor.

La propia naturaleza de las partidas supone en que tenemos un corto periodo de tiempo (menos de un minuto) para pensar y enviar una respuesta al servidor, teniendo que realizar una rápida elección cuando el servidor reciba todas las respuestas de manera asíncrona. Es por este motivo que el uso de Sockets se ajusta bien a nuestras necesidades.

Dentro del fichero 'partidasWebSocket' se define el **WebSocketServer** como un servidor de sockets que contiene el manejo de todas los mensajes de todos los clientes que están conectados.

Una peculiaridad es que le añade un ID aleatorio a cada cliente para poder encontrarlo más fácilmente en otros casos. Su complejidad reside en el apartado de cuando recibe un petición de tipo 'mensaje', donde tiene que discriminar qué tipo de mensaje ha recibido, de quién y cómo afecta eso a la partida que están jugando: puede ser un mensaje en el chat, una respuesta a la pregunta, un voto, etc.

3.1.3. GameLoop

Es la función donde se aplican las reglas del juego y se lleva a cabo todo el flujo lógico de la partida.

Consiste en un bucle principal que representa el número de rondas. En cada una de estas rondas se recoge un término al azar de la base de datos, y se le presenta el nombre de este término a cada uno de los jugadores.

Después de recibir la respuesta con la definición de todos los jugadores o tras haberse agotado el tiempo, se muestra a cada jugador todas esas definiciones recogidas más la definición correcta.

Tras una breve cuenta atrás, todos los jugadores han tenido que enviar sus votos, y el servidor pasa a realizar una comprobación de qué ha votado cada jugador, actualizando el contador de puntos de cada jugador en consecuencia. Una vez finalizado este paso, se vuelve a comenzar una ronda nueva.

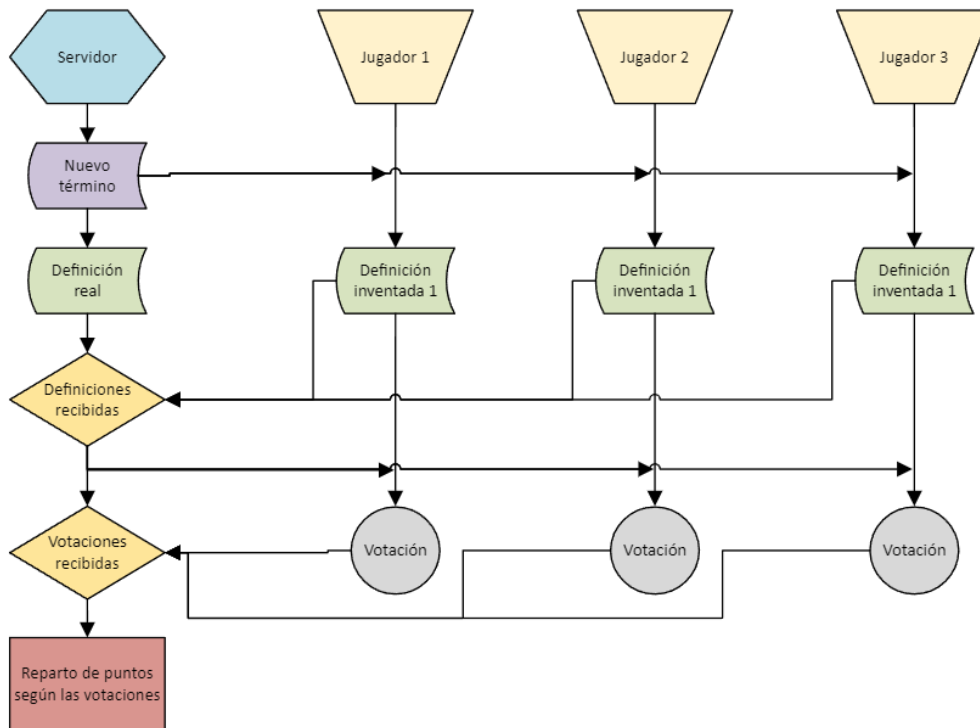


Figura 3.1: Diagrama Bucle de Juego.

3.2. Creación base de datos MongoDB

Para almacenar toda la información del proyecto, se decidió por usar la base de datos NoSQL conocida como MongoDB.

MongoDB nos permite crear un 'cluster' compartido de manera gratuita con nuestra cuenta. Nos permite elegir entre AWS[4], Google Cloud[13] y Microsoft Azure[23] para hostear nuestra base de datos.

Dentro de sus diferentes Tier, podemos elegir diferentes versiones con sus características y su precio asociado. Para nuestro caso gratuito, el cluster⁴ 'Sandbox' nos ofrece 500 conexiones máximo, un máximo de 100 databases y un máximo de 500 colecciones.

A diferencia de las bases de datos relacionales, las no relacionales se almacenan en colecciones, que se dividen en sí en 'databases', que cada una contiene todas las instancias de cada uno de los modelos que definimos en el proyecto. De esta manera, en vez de usar filas, tenemos instancias con IDs únicos que podemos referenciar para obtener el objeto.

3.3. Frontend

El Frontend de la aplicación móvil y de la web son dos versiones muy similares que buscan un diseño minimalista y directo para que los jugadores puedan acceder rápidamente a una sala y jugar las partidas que ellos deseen. Se ha apostado por un diseño de blanco sobre negro, con detalles sencillos que recuerden a las cartas de 'Cartas contra la Humanidad', del que se inspira.

La organización de todo este apartado se ha dividido en las siguientes secciones.

⁴**Cluster:** es un servidor que permite crear y mantener múltiples bases de datos de MongoDB.

3.3.1. Componentes

Aquí se almacenan los diferentes componentes visuales personalizados que se han implementado. Principalmente son recursos obtenidos de internet que luego se han ajustado a las necesidades del proyecto. Concretamente tenemos:

- **CustomButton:** botón pulsable que permite centralizar el diseño de todos los botones en un mismo modelo.
- **CustomInput:** cajetilla donde introducir texto que no solo permite un diseño coherente en toda la aplicación, si no que permite un uso más sencillo de formularios usando los 'Hooks'⁵ de 'React'.

3.3.2. Context

Uso del 'Context' de 'React' nos permite pasar información a componentes hijos sin necesidad de pasarlos de manera explícita. Esto nos permite mantener un estado global de lo que nos interesa.

El principal uso ha sido para definir una interfaz para el Socket de usuario, donde se define la conexión establecida con el server, los tipos de respuesta que puede enviar el usuario y se maneja los tipos de respuesta que recibe del servidor.

3.3.3. Navigator

Para poder saltar entre las diferentes pantallas, se ha tenido que instalar un Navigator de React de la biblioteca 'react-navigation'.

Este Navegador se declara en su propio fichero, en el cual definimos las diferentes pantallas de nuestra aplicación. También es necesario especificar qué información se debe recibir cada vez que navegamos a otra pantalla.

3.3.4. Vistas

Las vistas son todos los elementos que se muestra en la pantalla de nuestra aplicación. Cada grupo de elementos visuales forma una vista, que contiene todos estos componentes interactivables e información.

Las vistas desarrolladas de la versión móvil del proyecto son las siguientes:

⁵**Hooks:** es una función que recibe un componente o estado y que nos permite acceder a ese recurso durante toda la ejecución.



Figura 3.2: Pantalla del Menú Principal.

Es la primera pantalla que ven siempre los usuarios al entrar a la aplicación. En ella se encuentra un menú sencillo donde escoger entre diferentes opciones de identificación:

- **Jugar Ya:** Escoges un nombre de usuario y directamente accedes al 'Lobby'. Tu información de usuario se pierde al cerrar la sesión y no es recuperable. Es una opción rápida y directa para los usuarios casuales que únicamente quieren jugar con sus amigos.
- **Iniciar Sesión:** inicio de sesión con tu nombre de usuario y contraseña. Una vez autenticado, te otorga tu token de usuario.
- **Registro:** registro de usuario. Es necesario introducir un nombre de usuario que no esté en uso, un correo que no haya sido registrado y tu contraseña. Una vez validado, se registra el usuario en la base de datos y accede a la aplicación, recibiendo también su token.



Figura 3.3: Pantalla con inicio como user anónimo.



Figura 3.4: Pantalla del registro.

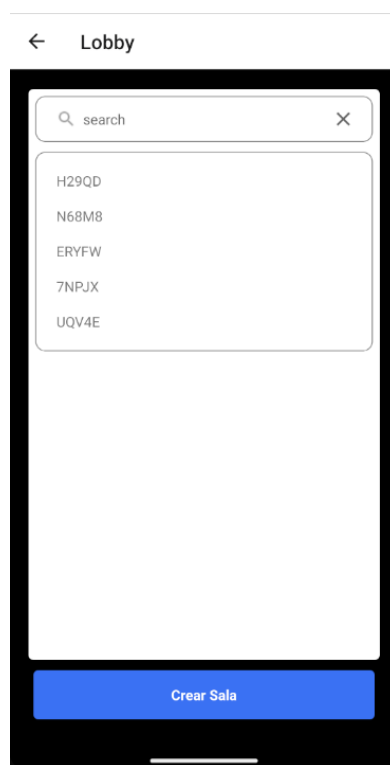


Figura 3.5: Pantalla del Lobby.

Esta segunda pantalla es la que contiene la información de todas sesiones públicas que están ahora mismo activas. A través de un menú desplegable, se puede seleccionar cualquier sesión y unirse con un clic. Después de esta acción se navega a la pantalla "Sesión de Juego".

Si se quiere crear una sala propia a la que invitar a tus amigos, se puede pulsar el botón:

- **Crear sala:** Botón para crear una sala. Al clicarlo, navegas automáticamente a la pantalla de "Sesión de Juego".



Figura 3.6: Pantalla de la Sesión de Juego.

En esta pantalla se encuentran todos los datos relacionados a la sesión de juego:

- Creador. Jugador creador de la sesión de juego.
- ID de la sesión.
- Número de rondas. Por defecto es igual al número de jugadores.
- Contraseña de la sesión.
- Jugadores. Listado de todos los jugadores participantes. Límite de 6.

Para los jugadores invitados estos serán todos los componentes que aparecerán en pantalla. La lista de jugadores y el número de rondas son los únicos datos que se actualizarán si ocurre alguna de las siguientes cosas: entra algún jugador, sale algún jugador o el creador actualiza el número de rondas jugables.

Los jugadores invitados se podrán unir encontrando el grupo en la pestaña de Lobby anterior o introduciendo la contraseña de la sesión.

Para compartir la contraseña, suponemos que el creador tiene un medio de comunicación adicional para jugar con sus invitados: ya sea de manera presencial o con algún software de comunicación externo.



Figura 3.7: Pantalla del Bucle de Juego. Recibe el término.

La pantalla de 'Gameloop' es el lugar donde los jugadores escriben su definición. Consiste en una pantalla principal donde se presenta el término a definir y una cajetilla donde introducir la definición inventada.

Una vez se acabe el tiempo de escritura y/ó todos los participantes hayan enviado su respuesta, la tarjeta se actualizará. Se ocultará la cajetilla y en su lugar aparece un menú desplegable con todas las respuestas enviadas junto a la verdadera.

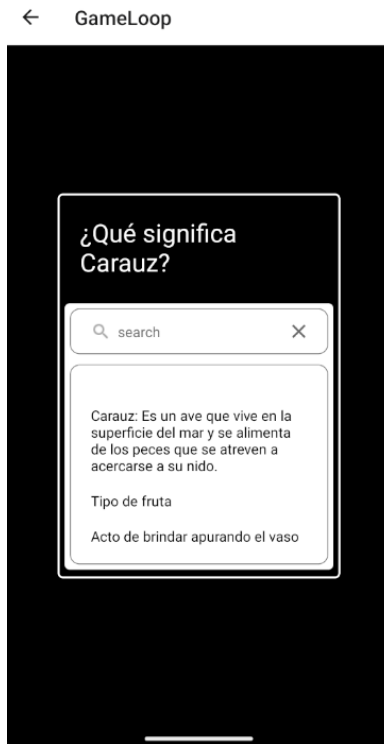


Figura 3.8: Pantalla Bucle de Juego. Selección de definición.

Se les otorga 30 segundos de reflexión a los jugadores, donde tendrán que emitir un voto sobre la definición que creen correcta. Una vez se obtengan todos los votos o se acabe la cuenta, se mostrará la definición correcta y se sumarán los puntos de manera acorde.

Una vez finalicen todas las rondas, se mostrará en el centro de la pantalla al jugador con más puntos, es decir, al ganador de la partida.

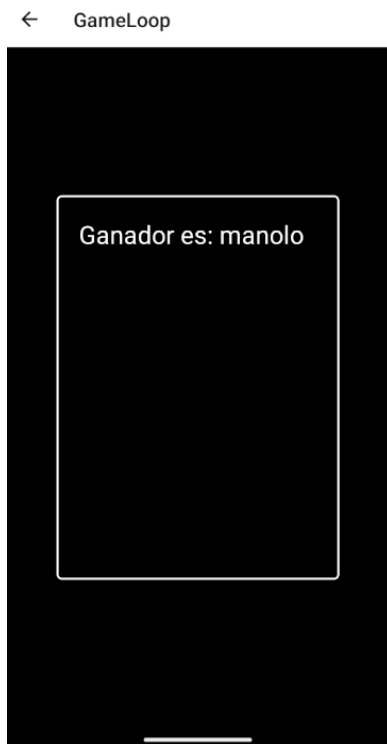


Figura 3.9: Pantalla Bucle de Juego. Se muestra al Ganador.

3.4. Implementación de IA

Para la implementación de la IA se decantó por utilizar uno de los modelos ofrecidos por Fireworks AI. Los motivos para ellos son que disponemos de gran cantidad de Chatbots⁶ con los que trabajar y que todos ellos disponen de modelo 'serverless', es decir, que podemos realizar peticiones a la API y que el procesamiento se realice en sus servidores. Seleccionamos el modelo Ollama2 70B para ser utilizado en el proyecto, ya que es el más potente.

Para realizar las llamadas a la API de Fireworks se necesitó crear una cuenta en la página, pues a ella se le asigna un token de autorización para comunicarse con las APIs. Este token se debe añadir a las opciones de la petición. Por último, se realiza una petición http simple a la que se añade el mensaje. Esta petición queda envuelta en una Promesa⁷ que nos permite esperar a la respuesta en el hilo principal de ejecución.

Para poder comunicar la respuesta recibida por el Chatbot se creó un Cliente del WebSocketServer. Este ClienteIA tiene un filtro para cuando recibe un mensaje de otras fuentes, que indica qué tipo de acción requiere de su parte. Algunas de estas acciones son: contestar al chat, felicitar al ganador, inventarse una definición y seleccionar una respuesta.

Tanto para cuando necesitamos obtener una definición inventada como una selección de una respuesta, se llama y espera a la Promesa. Una vez se recibe la respuesta, esta se envía a través del socket al servidor, donde será procesada. Esto ocurre las veces que sea necesario durante la partida.

⁶**Chatbot:** es una inteligencia artificial especializada en la comprensión y procesamiento del lenguaje natural. Su objetivo es dialogar con personas, creando sus respuestas en base a lo que el humano pregunte.

⁷**Promesa:** Representa un valor futuro que se puede obtener o no. Se ejecuta de manera asíncrona y solo devuelve la información cuando se la otorgan.

Capítulo 4

Estudio de viabilidad económica del proyecto

Este capítulo está dedicado a tres apartados que dictan la viabilidad económica del producto, presentando un modelo atractivo para clientes e inversores, el alcance del proyecto y el estudio del Retorno de inversión.

4.1. Modelo de comercialización

'Disparates' es un juego multijugador que requiere conocimientos generales muy básicos, por lo cual está pensado para todos los públicos. Esto supone un abanico grande de posibilidades donde exponer y publicitar el juego.

La presentación ante el público es crucial para este tipo de productos software, más si tenemos en cuenta que es una aplicación clasificada como ocio para todos los públicos, lo cual supone un nivel de exigencia superior en Interfaz de Usuario (UI)¹ y Experiencia de Usuario (UX)².

Otro factor interesante del proyecto es el uso de Inteligencias Artificiales, más concretamente ChatBots, como jugadores robot que pueden participar en las partidas. Estos Chatbots que están tan centrados en aprender del lenguaje humano, pueden suponer una capa de estudio en el proyecto.

A través de un chat de grupo y las definiciones entregadas por los jugadores humanos, estos Chatbots pueden almacenar y procesar esta información para otorgar definiciones o comentarios cada vez más verosímiles. Puede suponer otra ruta de estudio para los Chatbots.

Otra idea es intentar desdibujar la línea entre un participante real y una IA, pues es solo necesario el lenguaje escrito para jugar. El destacar que existe uno o varios jugadores robot en una partida puede ser atractivo para jugadores casuales e interesados en IA, que pueden encontrar curioso o entretenido el intentar reconocer las definiciones inventadas por el Chatbot.

4.1.1. Fuentes de ingresos

Una vez lanzada la aplicación, se deben elegir una forma de recuperar la inversión inicial. Debido a la naturaleza de videojuego del proyecto, es posible elegir más de una opción de monetización.

¹UI: *User Interface* en inglés. Se refiere a los elementos gráficos que crean una interfaz clara e intuitiva.

²UX: *User eXperience* en inglés. Se refiere a las sensaciones y opiniones que el uso de la aplicación genera en los usuarios.

- **Aplicación de pago.** La opción más inmediata es la de vender la aplicación en tiendas digitales de Android (y/o más plataformas) por un precio adecuado. Debido a que es un videojuego, el único indicador es el número de copias vendidas.
- **Publicidad.** Otra opción dentro de los videojuegos para móvil es el de ofrecer la aplicación gratis y añadir publicidad. Los ingresos se obtendrían a través de esta monetización de la publicidad con Google Ads[12].
- **Dos versiones.** Es común encontrar también videojuegos que apuestan por lanzar a la tienda digital las dos versiones anteriores. La idea es exponer y posicionar la versión gratis con publicidad, para obtener la mayor cantidad de usuarios e ingresos por publicidad. Esos usuarios que quieran acceder a la versión sin publicidad de la aplicación pueden pagar por la versión de pago, que puede ser igual o tener algún extra, respecto a la versión gratuita, para incentivar la compra.
- **Expansiones.** Otra posibilidad que ofrecen los juegos son las expansiones con nuevos términos, modos de juegos o temáticas para jugar. Esta forma de monetización consiste en invertir un tiempo de desarrollo menor al inicial en sacar nuevo contenido temático (Halloween, Navidad, sobre culturas, etc) y vender por un precio adecuado este nuevo contenido a los jugadores.
- **Ampliaciones futuras con metas.** Otra posibilidad del prototipo es la de incluir nuevos tipos de términos a adivinar. Actualmente el juego solo incluye palabras del diccionario, pero el juego original también contempla adivinar Siglas, Fechas o Personajes. Se pueden añadir metas estilo 'Kickstarter'[17]³: que a partir de un número de ventas de la aplicación o ingresos conseguidos, liberarían estas nuevas categorías de juego.
- **Expansión de la IA.** Añadir contenido de pago relacionado al uso de la inteligencia artificial. Poder añadir más IAs por partida, colaborar con organizaciones que estudien el procesamiento del lenguaje, desarrollar un Chatbot propio y comercializarlo; todas son ideas con las que sacar rentabilidad de esta funcionalidad.

4.1.2. Elección de modelo de comercialización

Debido a las diferentes maneras en las que es posible comercializar la aplicación, se ha decidido realizar una selección del modelo más interesante para el proyecto inicial, con la intención de realizar el Estudio del Retorno de Inversión usando esta referencia.

El formato elegido ha sido el de 'Dos versiones': '*Disparates Free*' y '*Disparates Pro*', sin embargo, se ha decidido en crear una versión adicional con mayores beneficios para aquellos entusiastas de la IA: '*Disparates Premium*'.

³**Kickstarter:** web de patronaje donde exponer tu proyecto y alcanzar una meta monetaria para financiarlo.

Versión	Precio	Publicidad	Contenido	IAs por partida
Disparates Free	Gratis	Sí	Solo definiciones de términos	1
Disparates Pro	5€	No	Definiciones de términos + primera expansión (Siglas y Fechas).	2
Disparates Premium	10€	No	Definiciones de términos + primera expansión (Siglas y Fechas).	5

Respecto a las demás opciones, esto no quiere decir que queden descartas, al contrario, pueden ser buenas estrategias para ampliar la vida del producto y conseguir ingresos a medio y largo plazo.

4.2. Desarrollo del proyecto

Durante esta sección se detallarán las fases y tareas que se deben llevar a cabo para realizar el proyecto. Siguiendo un esquema secuencial, se establecerán los costes, recursos y duración del proyecto.

4.2.1. Análisis

En esta primera fase se realiza un estudio de mercado completo, donde se analizará la tendencia de mercado, la competencia, el público objetivo, etc. A parte, se especificarán los requisitos y objetivos que tiene la aplicación en su fase inicial. Una vez aclarada la viabilidad del proyecto, se podrán planificar las tareas, fechas y recursos de las fases posteriores y el alcance total del proyecto.

4.2.2. Diseño

Esta segunda fase pone el foco en la selección de tecnologías para desarrollar el prototipo, creación de las estructuras de software y base de datos y especial énfasis en crear mockups con diseño de interfaces y estilos que sean juveniles y dinámicos, para ser lo más atractivos posibles.

4.2.3. Desarrollo

La fase de desarrollo consisten en la programación del software del proyecto. Se implementa el backend de la aplicación: sus rutas, modelos, seguridad y autenticación,

también la estructura y creación de la base de datos y el frontend de la aplicación. Debido a que la aplicación está categorizada como ocio es, aún si cabe, más necesario el invertir en la interfaz y experiencia de usuario junto a un diseño colorido y entretenido.

4.2.4. Fase de Pruebas

Una vez se desarrolla una versión completa sólida de la aplicación, se puede comenzar con la fase de pruebas. En este periodo se comprueba que la app cumple con los requisitos y funcionalidades declarados durante el Análisis, se cumplan las medidas de seguridad y los estándares de calidad.

Debido a la naturaleza lúdica del videojuego, se requiere realizar un tipo de testado especializado dentro de los estándares de calidad. Este consiste en realizar pruebas de juego con usuarios que representen nuestro público objetivo, y que estos juzguen si la aplicación es lo suficientemente divertida y/o interesante. Aunque sea una aplicación sólida y de calidad, si no es divertida, los usuarios no querrán jugar más de un par de veces.

A la mitad de esta etapa se debe comenzar con el marketing y publicidad del juego en redes sociales y entornos lúdicos, centrados principalmente en el público juvenil. Es posible utilizar el contenido generado en los tests lúdicos para empezar a generar expectación.

4.2.5. Despliegue

Es la fase final del proyecto, en la que se despliegan el backend y el frontend de la aplicación y se sube la app a la Play Store⁴.

En este nuevo entorno de producción se realizan los ajustes, pruebas y tests finales. También es cuando se empieza a invertir de manera importante en publicidad, ya sea con anuncios en varias plataformas, colaboraciones pagadas, etc, intentando continuar este momento que se ha ido generando de expectación y popularidad por el proyecto.

4.3. Estudio del Retorno de inversión

El retorno de inversión (ROI, en inglés) es una medida financiera utilizada para evaluar la rentabilidad del proyecto. En él se agrupan los gastos iniciales junto a los de mantenimiento, partiendo desde la fecha inicial hasta un periodo usual de 2 a 3 años. Se contrasta con el grupo de ingresos que obtiene la aplicación en ese tiempo para determinar si el proyecto es viable.

Los gastos del proyecto vienen divididos en dos grupos: gastos puntuales por licencias, hardware y compras, y los gastos de mantenimiento: trabajadores a nuestro cargo, suscripciones y servicios.

En la siguiente tabla se detallan los gastos de pago único:

⁴**Play Store**: tienda digital de aplicaciones Android.

Materiales	Unidades	Coste individual	Coste total
Portátil	6	1000€	6000€
Ratón	6	10€	60€
Teclado	6	120€	120€
Móvil Android	6	250	1.500€
Licencia Play Store	1	25	25€
Publicidad Influencers ⁵	6	250	1.500€

Haciendo uso de la herramienta Project Libre[36] se ha realizado una tabla de planificación que desglosa las diferentes fases en las tareas más importantes para completar el proyecto. A cada una de ellas se les ha asignado una serie de recursos para calcular el coste de tiempo y dinero de cada tarea, fase y del proyecto.

	Nombre	Duración	Inicio	Terminado	Costo	Trabajo
1	☐ Desarrollo del proyecto	272 days	20/05/24 8:00	3/06/25 17:00	163312,99 €	7.819,999 ...
2	☐ 1. Fase de análisis	22 days	20/05/24 8:00	18/06/24 17:00	24605,00 €	736 horas
3	1.1. Estudio de mercado y competencia	14 days	20/05/24 8:00	6/06/24 17:00	1904,00 €	112 horas
4	1.2. Análisis de requisitos y objetivos	10 days	20/05/24 8:00	31/05/24 17:00	3840,00 €	160 horas
5	1.3. Crear documento de especializaciones técnicas	4 days	3/06/24 8:00	6/06/24 17:00	1536,00 €	64 horas
6	1.4. Definición del alcance del proyecto, tareas y fases	4 days	7/06/24 8:00	12/06/24 17:00	1536,00 €	64 horas
7	1.5. Definición del equipo de desarrollo	4 days	13/06/24 8:00	18/06/24 17:00	9261,00 €	64 horas
8	☐ 2. Fase de diseño	34 days	19/06/24 8:00	5/08/24 17:00	22235,99 €	1.020 horas
9	2.1. Selección de tecnologías	3 days	19/06/24 8:00	21/06/24 17:00	2232,00 €	104 horas
10	2.2. Diseño de la base de datos	7 days	24/06/24 8:00	2/07/24 17:00	4704,00 €	224 horas
11	2.3. Diseño de la arquitectura	7 days	24/06/24 8:00	2/07/24 17:00	2688,00 €	112 horas
12	2.4. Diseño UI/UX	21 days	24/06/24 8:00	22/07/24 17:00	2772,00 €	140 horas
13	2.5. Creación de Mockups	10 days	23/07/24 8:00	5/08/24 17:00	3120,00 €	160 horas
14	☐ 3. Fase de desarrollo	77 days	6/08/24 8:00	20/11/24 17:00	50687,99 €	2.600 horas
15	3.1. Creación de la Base de datos	7 days	6/08/24 8:00	14/08/24 17:00	3360,00 €	168 horas
16	☐ 3.2. Backend	42 days	6/08/24 8:00	2/10/24 17:00	19680,00 €	1.016 horas
17	3.2.1. Creación modelos Base de datos	7 days	6/08/24 8:00	14/08/24 17:00	2016,00 €	112 horas
18	3.2.2. Creación servidor y API	21 days	6/08/24 8:00	3/09/24 17:00	10080,00 €	504 horas
19	3.2.3. Seguridad y Autenticación	7 days	4/09/24 8:00	12/09/24 17:00	3360,00 €	168 horas
20	3.2.4. Testeo	14 days	13/09/24 8:00	2/10/24 17:00	4032,00 €	224 horas
21	☐ 3.3. Frontend Android	46 days	4/09/24 8:00	6/11/24 17:00	12168,00 €	624 horas
22	3.3.1. Implementación de pantallas	25 days	4/09/24 8:00	8/10/24 17:00	7800,00 €	400 horas
23	3.3.2. Conectividad con el Backend	7 days	9/10/24 8:00	17/10/24 17:00	2352,00 €	112 horas
24	3.3.3. Testeo	14 days	18/10/24 8:00	6/11/24 17:00	2016,00 €	112 horas
25	☐ 3.4. Frontend Web	56 days	4/09/24 8:00	20/11/24 17:00	15288,00 €	784 horas
26	3.4.1. Implementación de pantallas	35 days	4/09/24 8:00	22/10/24 17:00	10920,00 €	560 horas
27	3.4.2 Conectividad con el Backend	7 days	23/10/24 8:00	31/10/24 17:00	2352,00 €	112 horas
28	3.4.3. Testeo	14 days	1/11/24 8:00	20/11/24 17:00	2016,00 €	112 horas
29	☐ 4. Fase de pruebas	58 days	21/11/24 8:00	10/02/25 17:00	24960,00 €	1.328 horas
30	4.1. Realización de prueba de funcionalidad	30 days	21/11/24 8:00	1/01/25 17:00	8640,00 €	480 horas
31	4.2. Conectividad entre plataformas	7 days	2/01/25 8:00	10/01/25 17:00	3192,00 €	168 horas
32	4.3. Pruebas lúdicas con usuarios finales	7 days	13/01/25 8:00	21/01/25 17:00	4200,00 €	224 horas
33	4.4. Documentación y corrección de errores	7 days	22/01/25 8:00	30/01/25 17:00	4536,00 €	224 horas
34	4.5. Pruebas lúdicas con usuarios finales 2	7 days	31/01/25 8:00	10/02/25 17:00	4200,00 €	224 horas
35	☐ 5. Fase de lanzamiento	81 days	11/02/25 8:00	3/06/25 17:00	40824,00 €	2.136 horas
36	5.1. Preparación de la campaña de marketing	30 days	11/02/25 8:00	24/03/25 17:00	7680,00 €	480 horas
37	5.2. Despliegue de Servidor y Web	7 days	25/03/25 8:00	2/04/25 17:00	3696,00 €	168 horas
38	5.3. Subida de la App a las tiendas digitales	1 day	3/04/25 8:00	3/04/25 17:00	336,00 €	16 horas
39	5.4. Pruebas finales en producción	7 days	4/04/25 8:00	14/04/25 17:00	5880,00 €	280 horas
40	5.5. Publicación de la aplicación	0 days	14/04/25 17:00	14/04/25 17:00	0,00 €	0 horas
41	5.6. Arranque y promoción de la campaña de marketing	15 days	15/04/25 8:00	5/05/25 17:00	3840,00 €	240 horas
42	5.7. Corrección de problemas urgentes	14 days	6/05/25 8:00	23/05/25 17:00	11760,00 €	560 horas
43	5.8. Planificación de nuevas actualizaciones y contenido	7 days	26/05/25 8:00	3/06/25 17:00	7440,00 €	384 horas

Figura 4.1: Tabla de planificación de la duración y costo del proyecto. ProjectLibre.

⁵**Influencer**: persona conocida o destacable en redes sociales e internet

Haciendo uso de estos datos, se estima que la duración total es de 272 días, es decir, casi 9 meses para la puesta en marcha.

El Diagrama de Gantt que se muestra a continuación se ha creado también con ProjectLibre. Representa de manera gráfica las fases estipuladas a lo largo de un calendario.

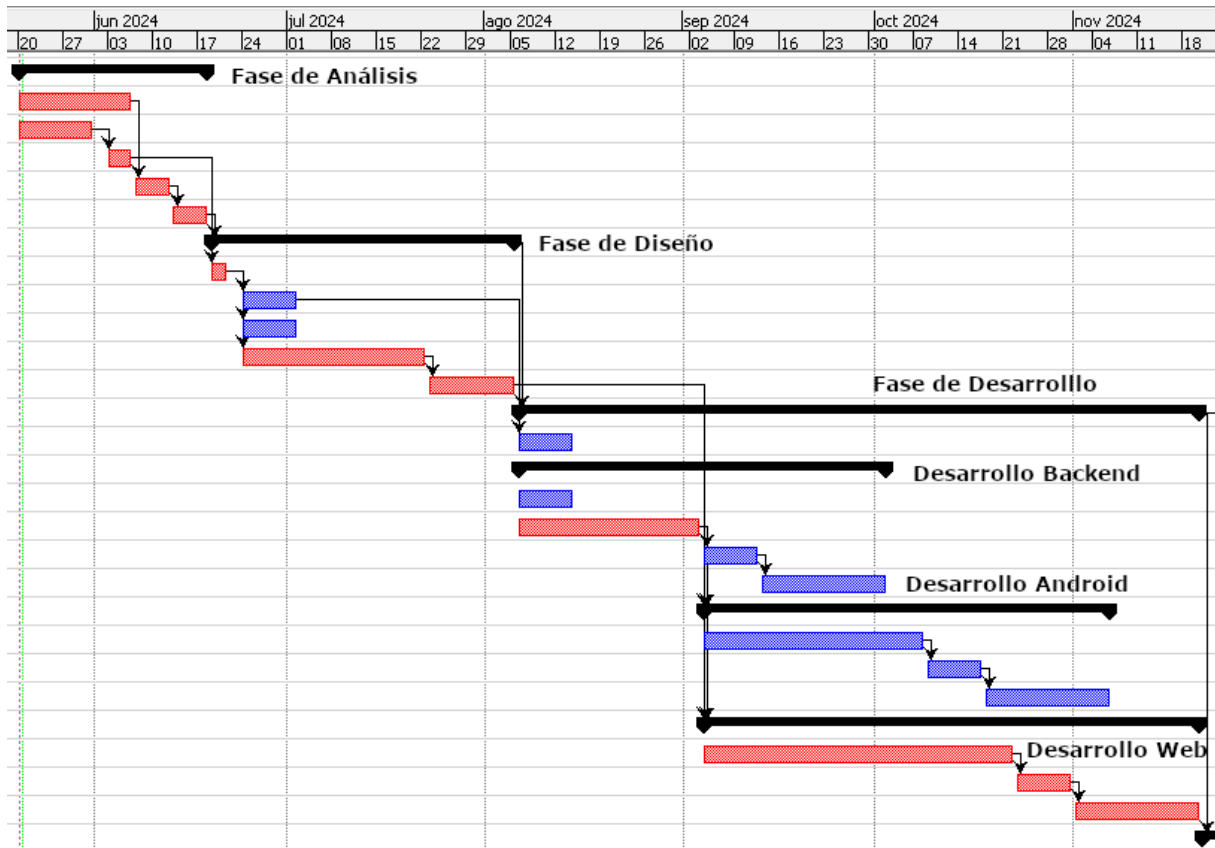


Figura 4.2: Diagrama de Gantt. Parte 1.

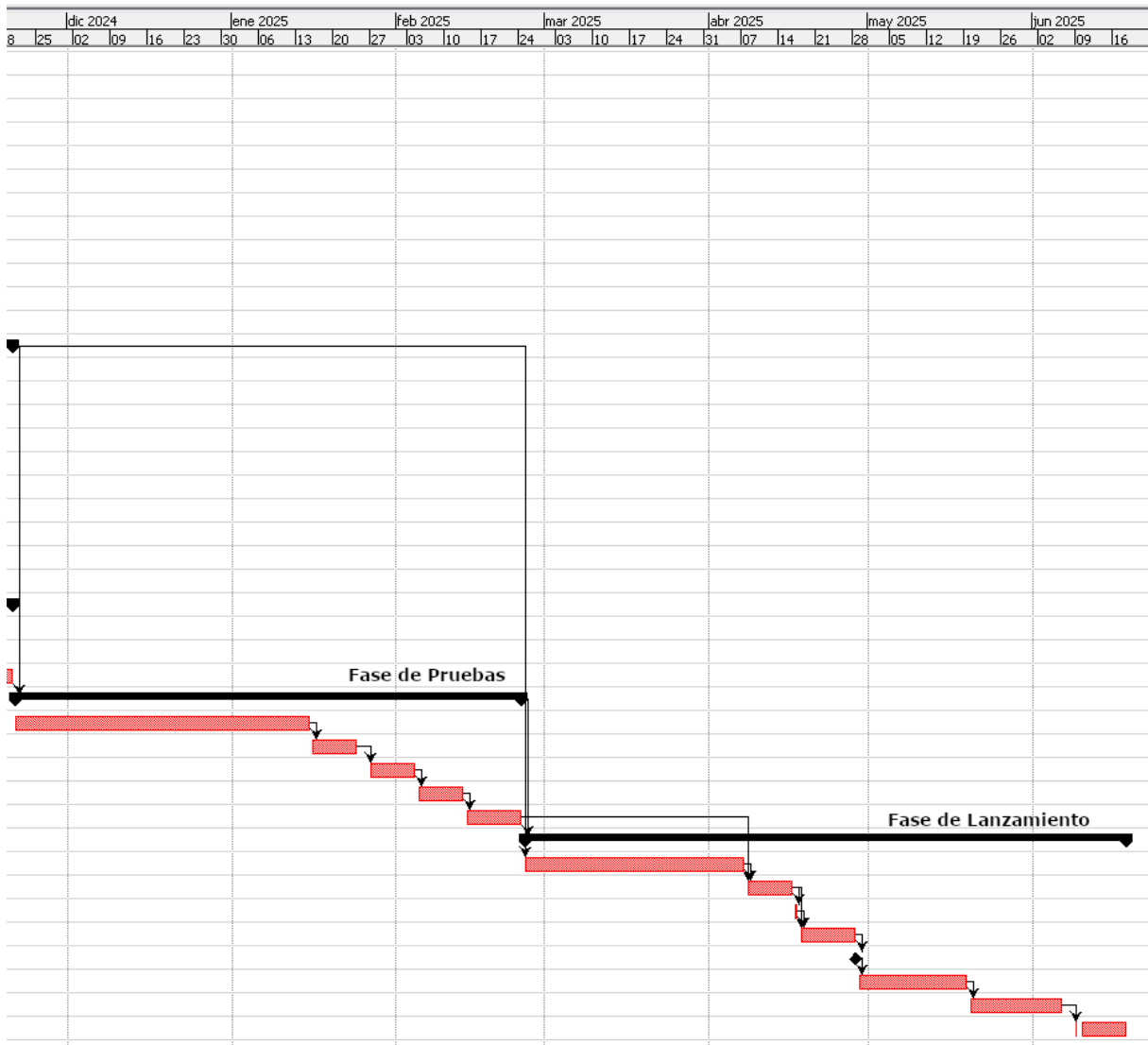


Figura 4.3: Diagrama de Gantt. Parte 2.

Adicional a lo concretado en la Figura 4.1, falta añadir el coste mensual de los servicios de hosting, de dominio, de la base de datos y el uso de IA. Para realizar este cálculo se ha hecho uso de la herramienta Azure Pricing Calculator[22] y los precios de Fireworks AI:

Servicio	Coste desarrollo	Coste producción
Microsoft Azure: Dominio + Hosting + Soporte	30€	162€
Cosmos DB	27€	110€
Fireworks AI. Tier 1.	Variable bajo demanda	50€ + variabilidad

Se estima que durante 7 de los 9 meses de ejecución del proyecto se emplean los costes de desarrollo, que suponen 57€ al mes. Aplicando este precio podemos calcular que los servicios cuestan un total de 399€ durante este periodo. A partir de la finalización de la Fase de pruebas se transiciona al coste de producción: 322€ al mes.

Los actores que forman parte activa del proyecto toman diferentes roles dentro del mismo para lograr una jerarquía clara. Estos perfiles son imprescindibles para desarrollar

el proyecto y cumplir con los objetivos. La estructura del equipo es la siguiente:

- **Jefe de Proyecto.** Supervisor del proyecto y encargado de coordinar todos los perfiles y fases para cumplir con los plazos y lograr los objetivos.
- **Arquitecto de Software.** Especialista en el diseño de software que crea unas bases sólidas en términos de arquitectura técnica, que aseguren una mayor escalabilidad, detección de errores, calidad del código, etc.
- **Desarrollador Fullstack.** Encargados de desarrollar las funcionalidades tanto Frontend como Backend.
- **Desarrollador Frontend.** Encargado de desarrollar únicamente el frontend. Debido al énfasis que se pone en tener un diseño atractivo, es necesario contratar un perfil con cierta experiencia que se enfoque únicamente en este campo.
- **Diseñador UI/UX.** Perfil responsable de la creación de las interfaces de usuario, las cuales deben ser dinámicas, funcionales, intuitivas y atractivas.
- **Experto en Márketing Digital.** Persona experta en estudiar y reconocer las tendencias de negocio en Internet. También debe ser capaz de proponer campañas de publicidad y reconocer posibilidades de exposición online.
- **Community Manager.** Encargado de llevar las redes sociales de la aplicación. Debe tener altas capacidades de comunicación, saber desenvolverse online y cierto conocimiento de marketing digital con intención de crear tendencias.

Nombre	Tasa Estandar
Desarrollador Fullstack	18,00 €/hora
Desarrollador Fullstack	18,00 €/hora
Desarrollador Fronted	21,00 €/hora
Diseñador UI/UX	18,00 €/hora
Arquitecto de software	24,00 €/hora
Jefe de proyecto	24,00 €/hora
Community Manager	15,00 €/hora
Experto en Márketing Digital	17,00 €/hora

Figura 4.4: Tabla trabajadores.

Una vez se lanza el proyecto de manera exitosa y se asienta la nueva metodología de trabajo de producción, es necesario prescindir de algunos perfiles. En este caso, se ha decidido prescindir del Arquitecto de Software y el Experto en Marketing Digital, pues ambos perfiles ya han aportado su valor al proyecto.

Respecto al número de usuarios de la aplicación, se espera un crecimiento estable de los mismos a medida que *'Disparates'* se da a conocer en nuevos grupos online.

Se esperan picos de popularidad en fechas festivas, campañas de publicidad, actualizaciones del juego y a la llegada de nuevas comunidades (por países, seguidores de cierto influencer, de otros videojuegos, etc). Durante los periodos intermedios se estima un flujo menor pero estable de usuarios fieles. Cada nuevo pico de popularidad aumentará esta cantidad de usuarios usuales.

Para estudiar este crecimiento se han creado los siguientes grupos:

- **Usuarios activos.** Media de usuarios activos que acceden a la página de manera usual durante el mes.
- **Usuarios Pro.** Número de usuarios que han pagado la versión Pro este mes. Se estima que un 10 % de los usuarios activos compra esta versión cada mes.
- **Usuarios Premium.** Número de usuarios que han pagado la versión Premium este mes. Se estima que un 5 % de los usuarios activos compra esta versión cada mes.
- **Visitas.** Número de visitas a la página al mes. Se estima que cada usuario activo accede una media de 10 veces al mes. Adicionalmente, existe una cantidad de usuarios esporádicos que acceden a la página y se estima que sus visitas suponen un incremento del 30 %. Es decir, el total de visitas se calcula:

$$Visitas = (Activos * 10) * 1,3$$

En la siguiente tabla se plasman estos resultados mes a mes:

Usuarios Activos	Nuevos Usuarios Pro	Nuevos Usuarios Premium	Visitas
14.000	1400	700	210000
8000	800	400	120000
8500	850	425	127500
8500	850	425	127500
9000	900	450	135000
14000	1400	700	210000
14000	1400	700	210000
10000	1000	500	150000
10500	1050	525	157500
10500	1050	525	157500
11000	1100	550	165000
18000	1800	900	270000
12500	1250	625	187500
12500	1250	625	187500
13000	1300	650	195000
13000	1300	650	195000
13500	1350	675	202500
13500	1350	675	202500
16000	1600	800	240000
14000	1400	700	210000
14000	1400	700	210000
14500	1450	725	217500
14500	1450	725	217500
19000	1900	950	285000
16000	1600	800	240000
16500	1650	825	247500
17000	1700	850	255000
17000	1700	850	255000
17000	1700	850	255000
21000	2100	1050	315000

Figura 4.5: Tabla usuarios y visitas.

En este modelo, los ingresos suponen la suma de lo generado por las compras de las versiones de pago combinado con los pagos por publicidad de Google AdSense. Usando los datos de la tabla anterior podemos calcular las ganancias percibidas usando la Calcula de AdSense de OmniCalculator[28]:

Ingresos Visitas (€)	Ingresos Usuarios (€)	Ingresos Mensuales (€)	Ingresos Acumulados (€)
25.200	14000	39.200	39.200
14.400	8000	22.400	61.600
15.300	8500	23.800	85.400
15.300	8500	23.800	109.200
16.200	9000	25.200	134.400
25.200	14000	39.200	173.600
25.200	14000	39.200	212.800
18.000	10000	28.000	240.800
18.900	10500	29.400	270.200
18.900	10500	29.400	299.600
19.800	11000	30.800	330.400
32.400	18000	50.400	380.800
22.500	12500	35.000	415.800
22.500	12500	35.000	450.800
23.400	13000	36.400	487.200
23.400	13000	36.400	523.600
24.300	13500	37.800	561.400
24.300	13500	37.800	599.200
28.800	16000	44.800	644.000
25.200	14000	39.200	683.200
25.200	14000	39.200	722.400
26.100	14500	40.600	763.000
26.100	14500	40.600	803.600
34.200	19000	53.200	856.800
28.800	16000	44.800	901.600
29.700	16500	46.200	947.800
30.600	17000	47.600	995.400
30.600	17000	47.600	1.043.000
30.600	17000	47.600	1.090.600
37.800	21000	58.800	1.149.400

Figura 4.6: Tabla cálculo de ingresos.

Por último, en el siguiente gráfico se presentan los gastos e ingresos del proyecto a lo largo de 30 meses a partir del lanzamiento de la aplicación. Como se puede observar, en el mes 24 se alcanza el punto en el que se comienzan a tener beneficios.

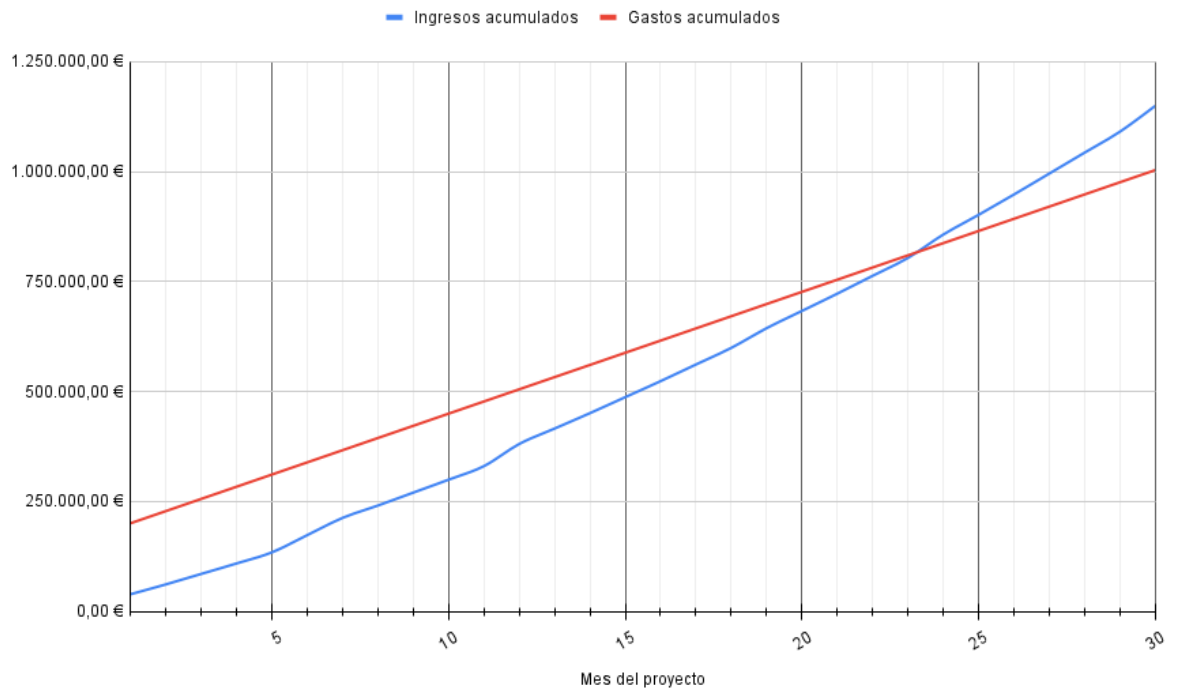


Figura 4.7: Gráfico de gastos e ingresos.

Capítulo 5

Conclusiones y líneas futuras

Como conclusión, el prototipo desarrollado ha logrado implementar *'Disparates'*, una versión multijugador digital contra IA del juego de mesa *'Balderdash'* en español. Es una versión simple tanto en diseño como funcionalidades, pero supone una base sobre la que construir un proyecto profesional.

Posibles ampliaciones y mejoras a futuro podrían ser las siguientes:

- **Expansiones originarias** de *'Balderdash'*, que incluyan las otras cuatro categorías: Siglas inverosímiles, Personajes peculiares, Películas sorprendentes y Leyes divertidas.
- **Expansiones de nuevas categorías.** En la misma línea que las anteriores: libros desconocidos, dichos o frases hechas, fechas, etc.
- **Expansiones temáticas.** Que incluyan una mezcla de palabras raras, libros, películas e incluso preguntas sobre la temática, como Navidad, Halloween, otras culturas, etc.
- **Añadir contenido cosmético.** Pagando dinero real o una moneda del juego, permitir conseguir nuevos cosméticos como fotos de perfil, colores de fondo, animaciones, etc.
- **Expansión de la IA.** Incrementar el número de IAs por partida, crear un Chatbot propio, entrenarlo con las respuestas de jugadores reales, realizar estudios sobre el procesamiento del lenguaje natural, etc.

Como último punto, el éxito de este proyecto reside en la capacidad de atraer y mantener una base de jugadores. Uno de los posibles objetivos puede suponer el crear una comunidad de fans casual y entusiastas de la inteligencia artificial.

Capítulo 6

Summary and Conclusions

As a conclusion, the developed prototype has successfully implemented '**Disparates**', a digital multiplayer against AI version of 'Balderdash' in Spanish. This version is simpler in design and functionality, compared to other commercial titles, but is a solid base for a future professional software project.

Some possible expansions and improvements for the launched version could include:

- **Core expansions** from 'Balderdash', that include the other four categories included in the base game: initials, peculiar characters, marvelous movies and laughable laws.
- **Brand new expansions** for new kind of categories, similar to the base ones. As an example: unknown books, sayings and proverbs, dates, etc.
- **Thematic expansions.** Theses could include a mix of different words, movies, books or even questions about a certain theme, like Christmas, Halloween, foreign cultures, etc.
- **Add new cosmetic content.** Either by paying real money or an in-game currency, give the players cosmetics such as profile pictures, background colors, animations, etc.
- **AI expansion.** Increasing the number of AIs per game session, build our own Chatbot, train them using the answers from real players, conduct studies about natural language processing, etc.

As a last note, the success of this future project resides in the ability to attract and maintain a base of recurrent players. One of the objectives could include creating a community around the game that includes casual players and AI enthusiasts.

Capítulo 7

Presupuesto

La realización de este Trabajo de Final de Grado también supone un coste añadido para el proyecto, debido a que declara y modela las bases, requisitos y objetivos. Así pues, en la siguiente tabla se desglosan las tareas realizadas, su tiempo estimado en horas y el precio asociado a cada tarea.

Tarea	Tiempo invertido (en horas)	Coste asociado (€/hora)
Análisis del problema y estudio previo	20	14€ / hora
Selección de las tecnologías	10	17€ / hora
Desarrollo Backend	65	17€ / hora
Desarrollo Frontend	110	17€ / hora
Testing	10	14€ / hora
Creación de la memoria	80	14€ / hora

En total se han invertido 295 horas, que implican un coste total de 4.545€.

Bibliografía

- [1] Evil Apples: Funny as ****. url: <https://play.google.com/store/apps/details?id=com.evilapples.app>.
- [2] Pinturillo 2. url: <https://pinturillo2.com/es/>.
- [3] Fireworks AI. url: <https://fireworks.ai/>.
- [4] Amazon. *Amazon Web Services*. url: <https://aws.amazon.com/>.
- [5] Balderdash. url: <https://en.wikipedia.org/wiki/Balderdash>.
- [6] Wolvesville Classic. url: <https://play.google.com/store/apps/details?id=org.faudroids.werewolf>.
- [7] Teléfono descompuesto. url: https://es.wikipedia.org/wiki/Tel%C3%A9fono_descompuesto.
- [8] ExpressJS. url: <https://expressjs.com/es/>.
- [9] Gartic.io. url: <https://play.google.com/store/apps/details?id=io.gartic.Gartic>.
- [10] Git. url: <https://git-scm.com/>.
- [11] GitHub. url: <https://github.com/>.
- [12] Google. *Google AdSense*. url: <https://ads.google.com/>.
- [13] Google. *Google Cloud*. url: <https://cloud.google.com/>.
- [14] Cartas contra la Humanidad. url: https://es.wikipedia.org/wiki/Cartas_contra_la_humanidad.
- [15] Java. url: <https://www.java.com/es/>.
- [16] Javascript. url: <https://es.wikipedia.org/wiki/JavaScript>.
- [17] Kickstarter. url: <https://www.kickstarter.com/>.
- [18] Kotlin. url: <https://kotlinlang.org/>.
- [19] Mafia. url: [https://en.wikipedia.org/wiki/Mafia_\(juego\)](https://en.wikipedia.org/wiki/Mafia_(juego)).
- [20] Make It Meme. url: <https://makeitmeme.com/es/>.
- [21] Meta. *OLLama2*. url: <https://ollama.com/>.
- [22] Microsoft. *Azure Pricing Calculator*. url: <https://azure.microsoft.com/en-us/pricing/calculator/>.
- [23] Microsoft. *Microsoft Azure*. url: <https://azure.microsoft.com/>.
- [24] MongoDB. url: <https://www.mongodb.com/>.
- [25] MySQL. url: <https://www.mysql.com/>.

- [26] React Native. url: <https://reactnative.dev/>.
- [27] NodeJS. url: <https://nodejs.org/en>.
- [28] OmniCalculator. url: <https://www.omnicalculator.com/finance/google-adsense>.
- [29] Wolvesville Online. url: <https://www.wolvesville.com/>.
- [30] Wolvesville - Werewolf Online. url: <https://play.google.com/store/apps/details?id=com.werewolfapps.online>.
- [31] OpenAI. *ChatGPT*. url: <https://chatgpt.com/>.
- [32] Overleaf. url: <https://www.overleaf.com/>.
- [33] Gartic Phone. url: <https://garticphone.com/es>.
- [34] Pictionary. url: <https://en.wikipedia.org/wiki/Pictionary>.
- [35] Wolvesville Classic Pro. url: <https://play.google.com/store/apps/details?id=org.faudroids.werewolf.pro>.
- [36] ProjectLibre. url: <https://www.projectlibre.com/>.
- [37] React. url: <https://es.react.dev/>.
- [38] Statista. url: <https://es.statista.com/>.
- [39] Pinturillo 2 App Store. url: <https://play.google.com/store/apps/details?id=air.com.pinturillo2.pinturillo2>.
- [40] Android Studio. url: <https://developer.android.com/>.
- [41] Google Trends. url: <https://trends.google.es/trends/>.
- [42] Typescript. url: <https://www.typescriptlang.org/>.
- [43] WordsRated. url: <https://wordsrated.com/>.